THE HONORABLE JAMES L. ROBART

UNITED STATES DISTRICT COURT
WESTERN DISTRICT OF WASHINGTON
AT SEATTLE

SRC LABS, LLC & SAINT REGIS MOHAWK TRIBE, Plaintiffs, v. AMAZON WEB SERVICES, INC., AMAZON.COM, INC., & VADATA INC. Defendants.	Case No.: 2:18-cv-00317-JLR **JOINT CLAIM CHART AND PREHEARING STATEMENT** **JURY TRIAL DEMANDED**
SRC LABS, LLC & SAINT REGIS MOHAWK TRIBE, Plaintiffs, v. MICROSOFT CORPORATION, Defendant.	Case No.: 2:18-cv-00321-JLR **JURY TRIAL DEMANDED**

Plaintiffs SRC Labs, LLC & Saint Regis Mohawk Tribe (collectively, "SRC"), Defendants Amazon Web Services, Inc., Amazon.com, Inc., and VADATA Inc. (collectively "Amazon") and Defendant Microsoft Corp. ("Microsoft") submit this Joint Claim Chart and Prehearing Statement per Local Patent Rule 132, the Court's Standing Order for Patent Cases, the Court's May 23, 2018 Order consolidating *SRC Labs, LLC v. Amazon Web Servs. Inc.*, No. 2:18-cv-00317-JLR ("the

KELLER ROHRBACK L.L.P.
1201 Third Avenue, Suite 3200
Seattle, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Amazon case") and *SRC Labs, LLC v. Microsoft Corporation*, No.: 2:18-cv-00321-JLR ("the Microsoft case") for a *Markman* hearing and *Markman*-related pretrial matters (Dkt. 96[1]), and the Court's October 23, 2018 Order Modifying the Claim Construction Schedule (Dkt. 110).

SRC asserts four patents against Amazon and six patents against Microsoft. Two of the patents—U.S. Patent No. 7,225,324 (the "'324 patent") and U.S. Patent No. 7,620,800 (the "'800 patent")—are asserted against both Amazon and Microsoft. The remaining two patents that SRC asserts against Amazon are U.S. Patent No. 7,149,867 (the "'867 patent") and U.S. Patent No. 9,153,311 (the "'311 patent"). The remaining four patents that SRC asserts against Microsoft are U.S. Patent Nos. 6,434,687 (the "'687 patent"), 6,076,152 (the "'152 patent"), 6,247,110 (the "'110 patent"), and 7,421,524 (the "'524 patent").

I. SRC'S ALLEGATIONS OF INFRINGEMENT

A. Against Amazon

The table below identifies each claim of each patent-in-suit that SRC asserts in the Amazon case, and each apparatus, product, device, process, method, act, or other instrumentality that SRC accuses of infringing each asserted claim:

Patent Number	Asserted Claims	Applicable subsection of § 271	Accused Devices
7,149,867	1, 3, 4	§271(a)	EC2 F1 Instance
7,225,324	1, 17	§271(b)	Zebra and other similar applications designed to run on EC2 F1 Instance
7,620,800	1, 17	§271(b)	Zebra and other similar applications designed to run on EC2 F1 Instance
9,153,311	1, 3, 9, 10	§271(a)	EC2 F1 Instance

SRC believes each claim limitation is literally present in the accused devices. Amazon denies that it infringes any valid and enforceable claim directly or indirectly, literally or otherwise.

[1] Unless otherwise indicated, docket citations refer to the docket number in *SRC Labs, LLC v. Amazon Web Servs. Inc.*, No. 2:18-cv-00317-JLR.

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

B. Against Microsoft

The table below identifies each claim of each patent in suit that Plaintiffs allege Microsoft is infringing, each accused apparatus, product, device, process, method, act, or other instrumentality accused of infringing each asserted claim, including for each claim the applicable statutory subsections of 35 U.S.C. § 271 asserted:

Patent Number	Asserted Claims	Applicable subsection of § 271	Accused Devices
6,076,152	1-7, 11, 12, 15, 21	§271(a)	Catapult v2 (Pikes Peak, Storey Peak), Catapult v3 (Dragontail Peak, Longs Peak, Nicholas Peak[2]), Catapult v4 (Storm Peak[3]).
6,247,110	1-7, 11, 12, 15, 21	§271(a)	Catapult v2 (Pikes Peak, Storey Peak), Catapult v3 (Dragontail Peak, Longs Peak, Nicholas Peak[4]), Catapult v4 (Storm Peak[5]).
6,434,687	1-5, 10-13, 18, 25	§271(a)	Bing (Ranking, Selection, DNN, CNN).
7,225,324	1, 8, 9, 17, 18, 21, 22, 23	§271(a)	Bing (Ranking, Selection, DNN, CNN), Brainwave, Azure Accelerated Networking, Compression (Xpress9 Level 6, Express8 Level 5), decompression, JPEG & video compression; LZ77 data compression, all applications running on the role or soft-shell portion of an FPGA in a Catapult Board.
7,421,524	1, 2, 13, 15	§271(a)	Catapult v2 (Pikes Peak, Storey Peak), Catapult v3 (Dragontail Peak, Longs Peak, Nicholas Peak), Catapult v4 (Storm Peak).
7,620,800	1, 8, 9, 17, 18, 21, 22, 23	§271(a)	Bing (Ranking, Selection, DNN, CNN), Brainwave, Azure Accelerated Networking, Compression (Xpress9 Level 6, Express8 Level 5), decompression, JPEG & video compression; LZ77 data compression, all applications running on the role or soft-shell portion of an FPGA in a Catapult Board.

Plaintiffs contend that each claim limitation is literally present in the Accused Devices. To the extent certain elements are not found to be literally present in the Accused Devices, and depending on the Court's construction, Plaintiffs contend the following elements may also be present

[2] Only accused to the extent it was deployed before December 17, 2017.
[3] Only accused to the extent it was deployed before December 17, 2017.
[4] Only accused to the extent it was deployed before December 17, 2017.
[5] Only accused to the extent it was deployed before December 17, 2017.

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

under the doctrine of equivalents:

Patent Number	Claim	Term
6,076,152	1, 3, 11	Memory bank
6,247,110	1, 3, 11	Memory bank
6,076,152	1, 11	Memory addressable
6,247,110	1, 11	Memory addressable
7,421,524	1, 15	Memory module bus

Copies of the eight patents in suit are attached as Exhibits A-H and excerpts from the file histories are attached as Exhibits I-P, which are shown in the Table Below:

Exhibit A:	U.S. Patent No. 6,247,110
Exhibit B:	U.S. Patent No. 6,076,152
Exhibit C:	U.S. Patent No. 9,153,311
Exhibit D:	U.S. Patent No. 7,225,324
Exhibit E:	U.S. Patent No. 7,421,524
Exhibit F:	U.S. Patent No. 6,434,687
Exhibit G:	U.S. Patent No. 7,620,800
Exhibit H:	U.S. Patent No. 7,149,867
Exhibit I:	'110 File History (excerpted pages)
Exhibit J:	'152 File History (excerpted pages)
Exhibit K:	'311 File History (excerpted pages)
Exhibit L:	'324 File History (excerpted pages)
Exhibit M:	'524 File History (excerpted pages)
Exhibit N:	'687 File History (excerpted pages)
Exhibit O:	'800 File History (excerpted pages)
Exhibit P:	'867 File History (excerpted pages)

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

II. AMAZON'S ALLEGATIONS OF INVALIDITY

Amazon contends that all asserted claims of each of the patents that SRC asserts against Amazon—the '324 patent, the '800 patent, the '867 patent, and the '311 patent—are invalid. The table below identifies the claims that Amazon alleges are invalid. Amazon served Invalidity Contentions served on July 9, 2018 and Amended Invalidity Contentions served on October 3, 2018. Included in the table below are the bases of the alleged invalidity and the relevant prior art as provided in those Contentions:

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
'324 patent	1, 17	The claims are invalid under 35 U.S.C. §§ 102, 103, 112 as anticipated, obvious, indefinite, and lacking written description and/or enablement.	• U.S. Patent No. 6,438,747 B1 ("Schreiber") • U.S. Patent No. 5,361,367 ("Fijany") • U.S. Patent No. 7,139,743 B2 ("Indeck") • U.S. Patent No. 6,675,187 B1 ("Greenberger") • "Building and Using a Highly Parallel Programmable Logic Array," Gokhale et al., January 1991 ("Splash") • An FPGA Implementation of Walsh-Hadamard Transforms for Signal Processing," A. Amira et al., 2001 ("Amira") • U.S. Patent No. 5,757,959 ("Lopresti") • U.S. Patent No. 4,698,751 ("Parvin") • "Artificial Neural Network Implementation on a single FPGA of a Pipelined On-Line Backpropagation," R. Gadea et al, Proceedings of International Symposium on Systems Synthesis, 2000 ("Gadea") • "Searching Genetic Databases on Splash 2," D. Hoang, 1993 ("Hoang") • "Mapping Nested Loops to Field Programmable Gate Array Based Systems," J. Spillane and J.S.N. Jean, NAECON 1995 ("Spillane") • "Splash 2: FPGAs in a Custom Computing Machine," D.A. Buell et al., 1996 ("Buell")

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
			• "Evaluation of the Streams-C C-to-FPGA Compiler: An Applications Perspective," J. Frigo et al., February 11-13, 2001 ("Streams-C") • "PCI-based WILDFIRE Reconfigurable Computing Engines," by B. K. Fross et al., October 21, 1996 ("Fross")
'800 patent	1, 17	The claims are invalid under 35 U.S.C. §§ 102, 103, 112 as anticipated, obvious, indefinite, and lacking written description and/or enablement.	• U.S. Patent No. 5,361,367 ("Fijany") • U.S. Patent No. 7,139,743 B2 ("Indeck") • U.S. Patent No. 6,438,747 B1 ("Schreiber") • U.S. Patent No. 6,675,187 B1 ("Greenberger") • "Building and Using a Highly Parallel Programmable Logic Array," Gokhale et al., January 1991 ("Splash") • An FPGA Implementation of Walsh-Hadamard Transforms for Signal Processing," A. Amira et al., 2001 ("Amira") • U.S. Patent No. 5,757,959 ("Lopresti") • U.S. Patent No. 4,698,751 ("Parvin") • "Artificial Neural Network Implementation on a single FPGA of a Pipelined On-Line Backpropagation," R. Gadea et al, Proceedings of International Symposium on Systems Synthesis, 2000 ("Gadea") • "Searching Genetic Databases on Splash 2," D. Hoang, 1993 ("Hoang") • "Mapping Nested Loops to Field Programmable Gate Array Based Systems," J. Spillane and J.S.N. Jean, NAECON 1995 ("Spillane") • "Splash 2: FPGAs in a Custom Computing Machine," D.A. Buell et al., 1996 ("Buell") • "Evaluation of the Streams-C C-to-FPGA Compiler: An Applications Perspective," J. Frigo et al., February 11-13, 2001 ("Streams-C") • "PCI-based WILDFIRE Reconfigurable Computing Engines," by B. K. Fross et al., October 21, 1996 ("Fross")

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
'311 patent	1, 3, 9, 10	The claims are invalid under 35 U.S.C. §§ 102, 103, 112 as anticipated, obvious, and indefinite.	• U.S. Patent Application Publication No. 2004/0034732 A1 ("Valin") • U.S. Patent Application Publication No. 2003/0200382 A1 ("Wells") • U.S. Patent No. 6,119,200 ("George") • U.S. Patent Application Publication No. 2014/0043918 A1 ("Ellis") • External Memory Interface Handbook Volume 3: Section III. DDR2 and DDR3 SDRAM Controller with UniPHY User • Guide, Altera, June 2011 ("UniPHY") • DDR2SOFT DDR2 Memory Controller VHDL Source Code Overview, ComBlock, September 22, 2010 ("ComBlock") • Atria DDR I/II DRAM Controller Core, Atria Logic Inc., 2009 ("Atria") • Xilinx MIG Spartan-6 MCB ("Spartan-6") • U.S. Patent No. 8,683,166 B1 ("Flateau") • ZedBoard System with Zync-7000 Processor ("ZedBoard + Zynq") • U.S. Patent Application Publication No. 2011/0264934 A1 ("Branover") • "System- and application-level support for runtime hardware reconfiguration on SoC platforms," Syrivelis et al., 2006 ("Syrivelis") • U.S. Patent No. 7,836,331 B1 ("Totolos") • "SUZAKU Hardware Manual," Atmark Techno, Inc., December 14, 2004 ("SUZAKU") • U.S. Patent No. 8,476,926 B2 ("Brunham") • "JEDEC Standard," JEDEC Solid State Technology Association, November 2008 ("JEDEC")
'867 patent	1, 3, 4	The claims are invalid under 35 U.S.C. §§ 102, 103, 112 as anticipated, obvious, and indefinite.	• U.S. Patent No. 6,822,959 B2 ("Galbi") • U.S. Patent No. 7,055,016 B2 ("Phelps") • "Memory Access Schemes for Configurable Processors," H. Lange and A. Koch, FPL 2000: Field-Programmable Logic and

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
			Applications: The Roadmap to Reconfigurable Computing, 2000 ("Lange") • "Architectural Adaptation for Application-Specific Locality Optimizations," Xingbin Zhang et al., International Conference on Computer Design VLSI in Computers and Processors, 1997 ("Zhang") • U.S. Patent No. 6,662,285 B1 ("Douglass") • U.S. Patent No. 6,981,099 B2 ("Paulraj") • "An FPGA Implementation of Triangle Mesh Decompression," by Tulika Mitra ("Mitra") • U.S. Patent No. 6,182,206 B1 ("Baxter")

III. MICROSOFT'S ALLEGATIONS OF INVALIDITY

Microsoft contends that all asserted claims of each of the patents that SRC asserts against Microsoft—the '152 Patent, the '110 patent, the '324 patent, the '800 patent, the '687 patent, and the '524 patent—are invalid. The table below identifies the claims that Microsoft alleges are invalid, along with the bases of the alleged invalidity and the relevant prior art:

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
'152 patent	1-7, 11, 12, 15, 18, 21	The claims are invalid under 35 U.S.C. §§ 101, 102, 103, 112 as directed to an abstract idea, anticipated, obvious, indefinite, and lacking written description and/or enablement.	• US Patent No. 5,574,930 to Halverson et al. (issued November 12, 1996). • Halverson, Richard Peyton, Jr., Ph.D, The Functional Memory Approach to the Design of Custom Computing Machines, August 1994. • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the machine described in the two references cited above. • U.S. Patent No. 5,678,021 by Pawate et al. (issued October 14, 1997). • U.S. Patent No. 6,185,704 by Pawate et al. (issued February 6, 2001, provisional filed

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
			April 11, 1997). • "YARDS: FPGA/MPU Hybrid Architecture for Telecommunication Data Processing" Tsutsui et al. • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the YARDS and/or ANT devices described in the reference cited above. • U.S. Patent No. 6,470,380 by Yoshizawa et al. (issued October 22, 2002, filed October 21, 1997). • "Splash 2: FPGAs in a custom Computing Machine," D.A. Buell, J. M. Arnold, and W. J. Kleinfelder, IEEE Computer Society, 1996. • The pre-critical date or pre-invention date, public knowledge, offer for sale, sale or prior invention of any version of the Splash 2 system described in the reference cited above. • Prism II: "PRISM-II Compiler and Architecture," M. Wazlowski et al., in Proceedings of the IEEE Workshop on FPGAs for Custom Computing Machines. 5-7 April, 1993. • The pre-critical date or pre-invention date pubic use, public knowledge, offer for sale, sale or prior invention of any version of the PRISM-II system described in the reference cited above. • U.S. Patent No. 5,671,355 to Collins (issued September 23, 1997). • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the Collins system described in the reference cited above. • U.S. Patent No. 5,835,734 to Alkalaj (issued November 10, 1998). • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
			of the Alkalaj system described in the reference cited above • "Programmable Active Memories: Reconfigurable Systems Come of Age" by Vuillemin et al. ("Vuillemin")
'110 patent	1-7, 11, 12, 15, 18, 21	The claims are invalid under 35 U.S.C. §§ 101, 102, 103, 112 as directed to an abstract idea, anticipated, obvious, indefinite, and lacking written description and/or enablement.	• US Patent No. 5,574,930 to Halverson et al. (issued November 12, 1996). • Halverson, Richard Peyton, Jr., Ph.D, The Functional Memory Approach to the Design of Custom Computing Machines, August 1994. • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the machine described in the two references cited above. • U.S. Patent No. 5,678,021 by Pawate et al. (issued October 14, 1997). • U.S. Patent No. 6,185,704 by Pawate et al. (issued February 6, 2001, provisional filed April 11, 1997). • "YARDS: FPGA/MPU Hybrid Architecture for Telecommunication Data Processing" Tsutsui et al. • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the YARDS and/or ANT devices described in the reference cited above. • U.S. Patent No. 6,470,380 by Yoshizawa et al. (issued October 22, 2002, filed October 21, 1997). • "Splash 2: FPGAs in a custom Computing Machine," D.A. Buell, J. M. Arnold, and W. J. Kleinfelder, IEEE Computer Society, 1996. • The pre-critical date or pre-invention date, public knowledge, offer for sale, sale or prior invention of any version of the Splash 2 system described in the reference cited above.

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
			• Prism II: "PRISM-II Compiler and Architecture," M. Wazlowski et al., in Proceedings of the IEEE Workshop on FPGAs for Custom Computing Machines. 5-7 April, 1993. • The pre-critical date or pre-invention date pubic use, public knowledge, offer for sale, sale or prior invention of any version of the PRISM-II system described in the reference cited above. • U.S. Patent No. 5,671,355 to Collins (issued September 23, 1997). • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the Collins system described in the reference cited above. • U.S. Patent No. 5,835,734 to Alkalaj (issued November 10, 1998). • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the Alkalaj system described in the reference cited above • "Programmable Active Memories: Reconfigurable Systems Come of Age" by Vuillemin et al. ("Vuillemin")
'324 patent	1, 8, 9, 17, 18, 21, 22, 23	The claims are invalid under 35 U.S.C. §§ 101, 102, 103, 112 as directed to an abstract idea, anticipated, obvious, indefinite, and lacking written description and/or enablement.	• "Splash 2: FPGAs in a custom Computing Machine," D.A. Buell, J. M. Arnold, and W. J. Kleinfelder, IEEE Computer Society, 1996. • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the Splash 2 system described in the reference cited above. • "Evaluation of the Streams-C C-to-FPGA Compiler: An Applications Perspective," J. Frigo et al., in Proceedings of the Association for Computing Machinery (ACM), February 11-13, 2001. • "PCI-based WILDFIRE Reconfigurable Computing Engines," B. K. Fross et al., in

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
			Proceedings of the International Society for Optics and Photonics (SPIE) Vol. 2914, October 21, 1996. • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the WILDFIRE system described in the reference cited above. • U.S. Patent No. 6,182,206,"Dynamically Reconfigurable Computing using a Processing Unit Having Changeable Internal Hardware Organization," filed February 26, 1998; priority date April 17, 1995. • "REMARC: Reconfigurable Multimedia Array Coprocessor," T. Miyamori and K. Olukotun, in IEICE Transactions on Information and Systems E82-D, Volume 82, pages 389-397, 1998. • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the REMARC system described in the reference cited above. • "Computing Multidimensional DFTs Using Xilinx FPGAs," C. Dick, in Proceedings of the 8th International Conference on Signal Processing Applications and Technology, September 13-16, 1998 • "The Fast Fourier Transform on a Reconfigurable Processor," G. Donohoe, J. Purviance, and P. Yeh, in Proceedings of the NASA Earth Sciences Technology Conference, June 11-13, 2002. • U.S. Patent No. 6,883,084 (issued on April 19, 2005, provisional filed July 25, 2001). • Mapping Applications to the RaPiD Configurable Architecture to C. Ebeling et al. • Data-Driven Multicomputers to J. Gaudiot • Automated Target Recognition on Splash 2, Rencher, et al.

KELLER ROHRBACK L.L.P.
1201 Third Avenue, Suite 3200
Seattle, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
			• Development of a parallel molecular dynamics code on SIMD Computers: Algorithm for use of pair list criterion to Roccatano, et al.
'800 patent	1, 8, 9, 17, 18, 21, 22, 23	The claims are invalid under 35 U.S.C. §§ 101, 102, 103, 112 as directed to an abstract idea, anticipated, obvious, indefinite, and lacking written description and/or enablement.	• "Splash 2: FPGAs in a custom Computing Machine," D.A. Buell, J. M. Arnold, and W. J. Kleinfelder, IEEE Computer Society, 1996. • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the Splash 2 system described in the reference cited above. • "Evaluation of the Streams-C C-to-FPGA Compiler: An Applications Perspective," J. Frigo et al., in Proceedings of the Association for Computing Machinery (ACM), February 11-13, 2001. • "PCI-based WILDFIRE Reconfigurable Computing Engines," B. K. Fross et al., in Proceedings of the International Society for Optics and Photonics (SPIE) Vol. 2914, October 21, 1996. • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the WILDFIRE system described in the reference cited above. • U.S. Patent No. 6,182,206,"Dynamically Reconfigurable Computing using a Processing Unit Having Changeable Internal Hardware Organization," filed February 26, 1998; priority date April 17, 1995. • "REMARC: Reconfigurable Multimedia Array Coprocessor," T. Miyamori and K. Olukotun, in IEICE Transactions on Information and Systems E82-D, Volume 82, pages 389-397, 1998. • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the REMARC system described in the

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
			reference cited above. • "Computing Multidimensional DFTs Using Xilinx FPGAs," C. Dick, in Proceedings of the 8th International Conference on Signal Processing Applications and Technology, September 13-16, 1998 • "The Fast Fourier Transform on a Reconfigurable Processor," G. Donohoe, J. Purviance, and P. Yeh, in Proceedings of the NASA Earth Sciences Technology Conference, June 11-13, 2002. • U.S. Patent No. 6,883,084 (issued on April 19, 2005, provisional filed July 25, 2001). • Mapping Applications to the RaPiD Configurable Architecture to C. Ebeling et al. • Data-Driven Multicomputers to J. Gaudiot • Automated Target Recognition on Splash 2, Rencher, et al. • Development of a parallel molecular dynamics code on SIMD Computers: Algorithm for use of pair list criterion to Roccatano, et al.
'687 patent	1-5, 10-13, 18, 25	The claims are invalid under 35 U.S.C. §§ 101, 102, 103, 112 as directed to an abstract idea, anticipated, obvious, indefinite, and lacking written description and/or enablement.	• "The Architecture of the Obelix - An Improved Internet Search Engine," P. Knezevic et al., Naval Postgraduate School, Proceedings of the 33rd Annual Hawaii International Conference on System Sciences (HICSS) Jan. 4-7, 2000, Maui, HI, USA, pp. 2145-2155 • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale, or prior invention or, or derivation from, of any version of the Obelix system described in the three references cited above. • U.S. Patent No. 6,326,806 H. Fallside et al. (issued December 4, 2001, filed March 29, 2000). • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale, or prior invention of any version

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
			of the Fallside system described in the reference cited above. • "A Web Based Multiuser Operating System for Reconfigurable Computing," to O. Diessel et al. in Proceedings of the Association for Computing Machinery (ACM), published during a conference held on April 12-16, 1999 • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale, or prior invention of any version of the Space 2 system described in the reference above. • "Networking Requirements and Solutions for a TV WWW Browser," to T. David et al., submitted to the Virginia Polytechnic Institute and State University on September 17, 1997 • U.S. Patent No. 6,370,527 to A. Singhal et al. (issued April 9, 2002, filed December 29, 1998) • U.S. Patent No. 6,795,448 to P. Lee et al. (issued September 21, 2004, filed March 2, 2000) • U.S. Patent No. 5,887,165 to S. Martel et al. (issued March 23, 1999) • U.S. Patent No. 6,101,180 to P. Donahue et al. (issued August 8, 2000) • U.S. Patent No. 5,870,769 to Y. Freund (issued February 9, 1999) • A press release entitled "Xilinx Unveils Internet Reconfigurable Logic," ("Xilinx Press Release"), published on November 10, 1998. • "An Open Platform for Development of Network Processing Modules in Reprogrammable Hardware," to J. Lockwood in IEC DesignCon 2001 ("FPX"), published during a conference held in January of 2001. • A dynamic reconfiguration run-time system to J. Burns et al. • U.S. Patent No. 7,072,888 to A. Perkins

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
			(issued July 4, 2006, filed June 16, 1999). • U.S. Patent No. 6,230,307 to D. Davis et al. (issued May 8, 2001). • U.S. Patent No. 6,098,065 to R. Skillen et al. (issued August 1, 2000).
'524 patent	1, 2, 13, 14, 15	The claims are invalid under 35 U.S.C. §§ 101, 102, 103, 112 as directed to an abstract idea, anticipated, obvious, indefinite, and lacking written description and/or enablement.	• "YARDS: FPGA/MPU Hybrid Architecture for Telecommunication Data Processing" Akihiro Tsutsui and Toshiaki Miyazaki, FPGA '97 ("Tsutsui"), presented and published at the 1997 ACM Fifth International Symposium on Field-Programmable Gate Arrays in California on February 9-11, 1997. • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the YARDS and/or ANT devices described in the reference cited above. • "Pilchard—A Reconfigurable Computing Platform With Memory Slot Interface," P. H. W. Leong, M. P. Leong, O. Y. H. Cheung, et al.,, in Proceedings of the 9th Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM '01), pp. 170–179, Rohnert Park, California, USA, April 29, 2001-May 2, 2001. • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the Pilchard system described in the reference cited above. • U.S. Patent No. 5,671,355 to Collins (issued September 23, 1997). • The pre-critical date or pre-invention date public use, public knowledge, offer for sale, sale or prior invention of any version of the Collins system described in the reference cited above. • U.S. Patent No. 5,835,734 to Alkalaj et al. (issued on November 10, 1998, filed on September 23, 1997)

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Patent	Claims	Bases of Invalidity	Relevant Prior Art Under §§ 102, 103
			• Special purpose FPGA for High-Speed Digital Telecommunication Systems to Tsutsui, et al. • Reconfigurable Real-Time Signal Transport System using Custom FPGAs to Hyashi et al. • U.S. Patent No. 5,857,109 to Taylor • Microcomputer Interfacing to Harold Stone

IV. AGREED CONSTRUCTIONS

SRC and Amazon have agreed to the constructions of the following claim terms:

Claim Terms	Agreed Constructions
"A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor, the reconfigurable processor comprising a plurality of functional units, said method comprising" – '324 patent, claim 1	The preamble is limiting.
"A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor, the reconfigurable processor comprising a plurality of functional units, said method comprising" –'800 patent, claim 1	The preamble is limiting.
"a data driven calculation" –'800 Patent, claim 1	Computation triggered by the availability of input data
"reconfigurable logic device" – '311 patent, claims 1, 3, 9, 10	FPGA, hybrid devices, such as a reconfigurable logic device with partial reconfiguration capabilities or an application specific integrated circuit (ASIC) device with reprogrammable regions contained within the chip

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

"A reconfigurable processor that instantiates an algorithm as hardware comprising" – '867 patent, claim 1	The preamble is limiting.

SRC and Microsoft have agreed to the constructions of the following claim terms:

Claim Terms	Agreed Constructions
Preamble of Claim 1 - `324 Patent and `800 Patent	The preamble is a claim limitation.
"Addressable" - `152 Patent: 1 & `110 Patent: 1	Accessible using normal memory access protocols
"Addressed" - `152 Patent: 1 & `110 Patent: 1	Accessed using normal memory access protocols
"Address" - `152 Patent: 1 & `110 Patent: 1	To access using normal memory access protocols
"a single system image of an operating system." - `687 Patent: 10, 11	an operating system that hides the heterogeneous and distributed nature of the available resources and presents them to the user and applications as a single unified computing resource
"Plurality" - `152 Patent: 1-7, 11, 12, 15, 18 and 21 & `110 Patent: 1-7, 11, 12, 15, 18 and 21	More than one
"instantiating" - `687 Patent: 1, 11	configuring

V. DISPUTED CLAIM TERMS

C. Patents Common to the Amazon Case and the Microsoft Case

With respect to the commonly asserted '324 and '800 patents, the parties in both cases dispute the claim constructions for the following three terms:

1. **"systolic" and "systolically"** – '324 patent: claim 1
2. **"pass computed data seamlessly"** – '324/'800 patents: claim 1

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

3. **"instantiating," "instantiated," and "instantiation"** – '324 patent: claim 1

D. Patents Asserted Against Amazon Only

With respect to the two patents asserted against Amazon only (the '311 patent and the '867 patent), SRC and Amazon dispute the claim constructions for the following four terms:

1. **"a data maintenance block"** – '311 patent: claim 1
2. **"a data prefect unit"** – '867 patent: claims 1, 3, 4
3. **"a data prefetch unit coupled to the memory, wherein the data prefetch unit retrieves only computational data required by the algorithm from a second memory of second characteristic memory bandwidth and/or memory utilization and places the retrieved computational data in the first memory"** – '867 patent: claim 1
4. **"at least the first memory and data prefetch unit are configured to conform to needs of the algorithm"** – '867 patent: claims 1, 3, 4

E. Patents Asserted Against Microsoft Only

With respect to the four patents asserted against Microsoft only (the '687 patent, the '152 patent, the'110 patent, and the '524 patent), SRC and Microsoft dispute the claim constructions for the following ten terms:

1. **A data driven calculation** – '800 patent: claim 1
2. **Memory bank** – '152/'110 patents claims: 1, 3, 11, 20
3. **Memory algorithm processor/reconfigurable memory algorithm processor** – '152/'110 patents claims: 1, 3, 11
4. **Means connecting** … - '152/'110 patents: claim 1
5. **Means coupling** … - '152/'110 patents: claim 11
6. **Memory module bus** – '524 patent: claim 1
7. **Providing said altered data directly from said memory module bus to an external device coupled thereto** – '524 patent claim 1
8. **The order of the steps recited in claim 1** – '687 patent

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

9. **At an internet site** – ’687 patent: claim 1, 18

10. **Demographic data** – ’687 patent: claim 5, 12, 13

Attached as Exhibit R is the Joint Claim Chart of SRC and Amazon and attached as Exhibit Q is the Joint Claim Chart of SRC and Microsoft. Each chart contains the corresponding parties’ proposed constructions for each disputed claim term, phrase, or clause, together with an identification of intrinsic and extrinsic evidence on which each party intends to rely to support or oppose the proposed constructions.

VI. THE TEN MOST IMPORTANT DISPUTED CLAIM TERMS

SRC and Amazon dispute the claim constructions of seven terms only.

SRC and Microsoft dispute the claim constructions of thirteen terms, three of which are common to Amazon. A list of the ten most important disputed claim terms is below:

1. **Means connecting** … - ’152/’110 patents: claim 1
2. **Means coupling** … - ’152/’110 patents: claim 11
3. **“instantiating,” “instantiated,” and “instantiation”** – ’324/’800 patents: claim 1
4. **“Providing said altered data directly from said memory module bus to an external device coupled thereto** – ’524 patent claim 1
5. **The order of the steps recited in claim 1** – ’687 patent
6. **At an internet site** – ’687 patent: claim 1, 18
7. **Memory algorithm processor/reconfigurable memory algorithm processor** – ’152/’110 patents claims: 1, 3, 11
8. “**memory bank”** – ’152/’110 patents claims 1, 3, 11, 20
9. **“systolic” and “systolically”** – ’324 patent: claim 1
10. **“memory module bus** – ’524 patent claim 1

VII. THE CLAIM CONSTRUCTION HEARING

Anticipated Length: Pursuant to this Court’s Minute Order (Dkt. 96), the parties antici-

pate that the Claim Construction hearing will take place over two days, with three portions addressing the following patents: (1) the commonly-asserted patents (the '324 patent and the '800 patent); (2) the patents asserted only in the Amazon case (the '311 patent and the '867 patent); and (3) the patents asserted only in the Microsoft case (the '687 patent, the '152 patent, the '110 patent, and the '524 patent).

The parties anticipate that portion (1) of the Claim Construction Hearing will last approximately 2 hours, to be evenly allocated among the parties. Counsel for each party (SRC, Amazon, Microsoft) intends to present argument for each disputed term from the commonly-asserted patents.

SRC and Amazon anticipate that portion (2) of the Claim Construction Hearing will last approximately 2 hours, to be evenly allocated among the parties.

SRC and Microsoft believe that portion (3) of the Claim Construction Hearing will last approximately 5 hours, to be evenly allocated among the parties.

Proposed Order of Presentation: The parties propose that the hearing proceed on a term-by-term basis, according to the order in the list of disputed terms provided above, and for each term, SRC presents first followed by Amazon and/or Microsoft.

Live Testimony: The parties agree that live testimony at the Claim Construction Hearing is not necessary.

Technical Tutorials: The parties believe a technical tutorial on the subject matter of the patents would be helpful for the Court. SRC would like to present its technical tutorial through the live presentation of one of the inventors. Amazon may present its technical tutorial through the live presentation of its technical expert, Dr. Brad Hutchings. Microsoft believes that a technical tutorial can be provided by counsel the first day of the Markman hearing, but to the extent the Court would hear a presentation from SRC's inventor, Microsoft may present its technical tutorial through a live presentation by its technical expert, Dr. Henry Houh. The parties agree that any tutorial is not evidence for claim construction.

SRC and Amazon agree that the tutorial should take place before the Claim Construction

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

Hearing, but they have no preference as to whether it is scheduled for December 20, 2018 (the first day of the Claim Construction Hearing) or at another day prior to that. SRC and Amazon agree that 45 minutes is sufficient for each party to present its tutorial in its case. Microsoft prefers that the technical tutorial take place on December 20, 2018, but otherwise agrees that believes that 45 minutes is sufficient for each party to present its tutorial in its case

Prehearing Conference: SRC believes that a pre-hearing conference is necessary to address issues the parties anticipate may arise during or after claim construction briefing.

Amazon and Microsoft do not anticipate any issues to arise during or after claim construction briefing that may require a pre-hearing conference, but do not oppose SRC's request. Should the Court be so inclined, any issues can also be addressed (along with the tutorial) on December 20, 2018 before the Claim Construction Hearing commences on that day.

Independent Expert: The parties do not think the appointment of an independent expert is necessary.

Dated: November 1, 2018

By: *s/ Karin B. Swope*
Mark A. Griffin, WSBA #16296
Karin B. Swope, WSBA #24015
KELLER ROHRBACK L.L.P.
1201 Third Avenue, Suite 3200
Seattle, WA 98101
mgriffin@KellerRohrback.com
kswope@KellerRohrback.com
T: (206) 623-1900
F: (206) 623-3384

Michael W. Shore* (mshore@shorechan.com)
Alfonso G. Chan* (achan@shorechan.com)
Christopher Evans* (cevans@shorechan.com)
Ari B. Rafilson* (arafilson@shorechan.com)
Paul T. Beeler* (pbeeler@shorechan.com)
SHORE CHAN DEPUMPO LLP
901 Main Street, Suite 3300
Dallas, Texas 75202
T: (214) 593-9110
F: (214) 593-9111

Attorneys for Plaintiffs SRC Labs, LLC & Saint Regis Mohawk Tribe in the Amazon case

By */s/ Scott M. Border*
Scott M. Border*
sborder@sidley.com
Joseph A. Micallef*
jmicallef@sidley.com
SIDLEY AUSTIN LLP
1501 K St. NW, Suite 600
Washington, DC 20005
Tel: (202) 736-8000

Patty A. Eakes, WSBA 18888
pattye@calfoeakes.com
Emily D. Powell, WSBA 49351
emilyp@calfoeakes.com
CALFO EAKES & OSTROVSKY PLLC
1301 Second Avenue, Suite 2800
Seattle, Washington 98101
(206_407-2200

David E. Killough, WSBA 40,185
davkill@microsoft.com
1 Microsoft Way
Redmond, WA 98052
Tel: (425) 703-8865

Attorneys for Plaintiffs SRC Labs, LLC and Saint Regis Mohawk Tribe in the Microsoft case

Richard A. Cederoth*
rcederoth@sidley.com
Nathaniel C. Love*
nlove@sidley.com
SIDLEY AUSTIN LLP
One South Dearborn Street
Chicago, Illinois 60603
Tel: (312) 853-7000

* Admitted pro hac vice
Attorneys for Defendant Microsoft Corporation

CERTIFICATE OF SERVICE

I hereby certify that on this 1st day of November 2018, I electronically filed the foregoing with the Clerk of the Court using the CM/ECF system, which will send notification of such filing to all counsel of record.

Dated: November 1, 2018 *s/ Karin B. Swope*

KELLER ROHRBACK L.L.P.
1201 THIRD AVENUE, SUITE 3200
SEATTLE, WA 98101-3052
TELEPHONE: (206) 623-1900
FACSIMILE: (206) 623-3384

EXHIBIT A

US006247110B1

(12) **United States Patent**
Huppenthal et al.

(10) **Patent No.: US 6,247,110 B1**
(45) **Date of Patent: *Jun. 12, 2001**

(54) **MULTIPROCESSOR COMPUTER ARCHITECTURE INCORPORATING A PLURALITY OF MEMORY ALGORITHM PROCESSORS IN THE MEMORY SUBSYSTEM**

(75) Inventors: **Jon M. Huppenthal**; **Paul A. Leskar**, both of Colorado Springs, CO (US)

(73) Assignee: **SRC Computers, Inc.**, Colorado Springs, CO (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **09/481,902**

(22) Filed: **Jan. 12, 2000**

Related U.S. Application Data

(63) Continuation of application No. 08/992,763, filed on Dec. 17, 1997, now Pat. No. 6,076,152.

(51) **Int. Cl.**[7] **G06F 15/80**

(52) **U.S. Cl.** **712/15**

(58) **Field of Search** 712/2, 13, 15, 712/21, 22, 28; 708/232

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,230,057		7/1993	Shido et al.	712/22
5,892,962		4/1999	Cloutier	712/16
6,076,152	*	6/2000	Huppenthal et al.	712/15

OTHER PUBLICATIONS

Albaharna, Osama, et al., "On the viability of FPGA–based integrated coprocessors", © 1996 IEEE, Publ. No. 0–8186–7548–9/96, pp. 206–215.

Barthel, Dominique Aug. 25–26, 1997, "PVP a Parallel Video coProcessor", Hot Chips IX, pp. 203–210.

Bittner, Ray, et al., "Computing kernels implemented with a wormhole RTR CCM", © 1997 IEEE, Publ. No. 0–8186–8159–4/97, pp. 98–105.

Bertin, Patrice, et al., "Programmable active memories: a performance assessment", © 1993 Massachusetts Institute of Technology, pp. 88–102.

Culbertson, W. Bruce, et al., "Exploring architectures for volume visualization on the Teramac custom computer", © 1996 IEEE, Publ. No. 0–8186–7548–9/96, pp. 80–88.

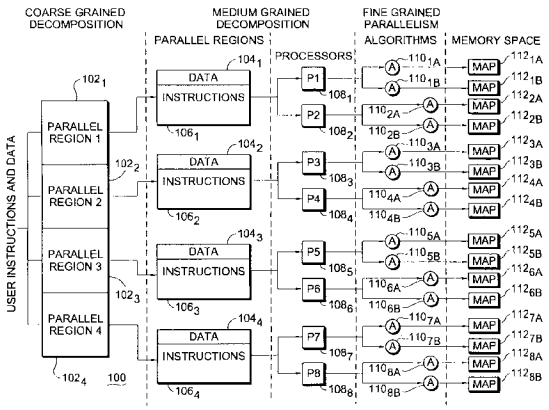
(List continued on next page.)

Primary Examiner—John A. Follansbee
(74) *Attorney, Agent, or Firm*—William J. Kubida; Hogan & Hartson LLP

(57) **ABSTRACT**

A multiprocessor computer architecture incorporating a plurality of programmable hardware memory algorithm processors ("MAP") in the memory subsystem. The MAP may comprise one or more field programmable gate arrays ("FPGAs") which function to perform identified algorithms in conjunction with, and tightly coupled to, a microprocessor and each MAP is globally accessible by all of the system processors for the purpose of executing user definable algorithms. A circuit within the MAP signals when the last operand has completed its flow thereby allowing a given process to be interrupted and thereafter restarted. Through the use of read only memory ("ROM") located adjacent the FPGA, a user program may use a single command to select one of several possible pre-loaded algorithms thereby decreasing system reconfiguration time. A computer system memory structure MAP disclosed herein may function in normal or direct memory access ("DMA") modes of operation and, in the latter mode, one device may feed results directly to another thereby allowing pipelining or parallelizing execution of a user defined algorithm. The system of the present invention also provides a user programmable performance monitoring capability and utilizes parallelizer software to automatically detect parallel regions of user applications containing algorithms that can be executed in the programmable hardware.

22 Claims, 4 Drawing Sheets



OTHER PUBLICATIONS

Culbertson, W. Bruce, et al., "Defect tolerance on the Teramac custom computer", © 1997 IEEE, Publ. No. 0–8186–8159–4/97, pp. 116–123.

Chan, Pak, et al., "Architectural tradeoffs in field–programmable–device–based computing systems", © 1993 IEEE, Publ. No. 0–8186–3890–7/93, pp. 152–161.

Clark, David, et al., "Supporting FPGA microprocessors through retargetable software tools", © 1996 IEEE, Publ. No. 0–8186–7548–9/96, pp. 195–103.

Cuccaro, Steven, et al., "The CM–2X: a hybrid CM–2/Xilink prototype", © 1993 IEEE, Publ. No. 0–8186–3890–7/93, pp. 121–130.

Dehon, Andre, "DPGA–Coupled microprocessors: commodity IC for the early 21st century", © 1994 IEEE, Publ. No. 0–8186–5490–2/94, pp. 31–39.

Dhaussy, Philippe, et al., "Global control synthesis for an MIMD/FPGA machine", © 1994 IEEE, Publ. No. 0–8186–5490–2/94, pp. 72–81.

Elliott, Duncan, et al., "Computational Ram: a memory–SIMD hybrid and its application to DSP", © 1992 IEEE, Publ. No. 0–7803–0246–X/92, pp. 30.6.1–30.6.4.

Fortes, Jose, et al., "Systolic arrays, a survey of seven projects", © 1987 IEEE, Publ. No. 0018–9162/87/0700–0091, pp. 91–103.

Hauser, John, et al.: "GARP: a MIPS processor with a reconfigurable co–processor", © 1997 IEEE, Publ. No. 0–08186–8159–4/97, pp. 12–21.

Hartenstein, Reiner, et al., "A reconfigurable data–driven ALU for Xputers", © 1994 IEEE, Publ. No. 0–8186–5490–2/94, pp. 139–146.

Hayes, John, et al., "A microprocessor–based hypercube, supercomputer", © 1986 IEEE, Publ. No. 0272–1732/86/1000–0006, pp. 6–17.

Hagiwara, Hiroshi, et al., "A dynamically microprogrammable computer with low–level parallelism", © 1980 IEEE, Publ. No. 0018–9340/80/07000–0577, pp. 577–594.

King, William, et al., "Using MORRPH in an industrial machine vision system". © 1996 IEEE, Publ. No. 08186–7548–9/96, pp. 18–26.

Manohar, Swaminathan, et al., "A pragmatic approach to systolic design", © 1988 IEEE, Publ. No. CH2603–9/88/0000/0463, pp. 463–472.

Mauduit, Nicolas, et al., "Lneuro 1.0: a piece of hardware LEGO for building neural network systems," © 1992 IEEE, Publ. No. 1045–9227/92, pp. 414–422.

Patterson, David, et al., "A case for intelligent DRAM: IRAM", Hot Chips VIII, Aug. 19–20, 1996, pp. 75–94.

Peterson, Janes, et al., "Scheduling and partitioning ANSI–C programs onto multi–FPGA CCM architectures", © 1996 IEEE, Publ. No. 0–8186–7548–9/96, pp. 178–187.

Schmit, Herman, "Incremental reconfiguration for pipelined applications," © 1997 IEEE, Publ. No. 0–8186–8159–4/97, pp. 47–55.

Villasenor, John, et al., "Configurable computing", © 1997 Scientific American, Jun. 1997.

Stone, Harold, "A logic–in–memory computer", © 1970 IEEE, IEEE Transactions on Computers, pp. 73–78, Jan. 1990.

Trimberger, Steve, et al., "A time–multiplexed FPGA", © 1997, IEEE, Publ. No. 0–8186–8159–4/97, pp. 22–28.

Thornburg, Mike, et al., "Transformable Computers", © 1994 IEEE, Publ. No. 0–8186–5602–6/94, pp. 674–679.

Tangen, Uwe, et al., "A parallel hardware evolvable computer POLYP extended abstract", © 1997 IEEE, Publ. No. 0–8186–8159/4/97, pp. 238–239.

Tomita, Shinji, et al., "A computer low–level parallelism QA–2", © 1986 IEEE, Publ. No. 0–0384–7495/86/0000/0280, pp. 280–289.

Ueda, Hirotada, et al., "A multiprocessor system utilizing enhanced DSP's for image processing", © 1988 IEEE, Publ. No. CH2603–9/88/0000/0611, pp. 611–620.

Wang, Quiang, et al., "Automated field–programmable compute accelerator design using partial evaluation", © 1997 IEEE, Publ. No. 0–8186–8159–4/97, pp. 145–154.

Wirthlin, Michael, et al., "The Nano processor: a low resource reconfigurable processor", © 1994 IEEE, Publ. No. 0–8186–5490–2/94, pp. 23–30.

Wittig, Ralph, et al., "One Chip: An FPGA processor with reconfigurable logic", © 1996 IEEE, Publ. No. 0–8186–7548–9/96, pp. 126–135.

Wirthlin, Michael, et al., "A dynamic instruction set computer", © 1995 IEEE, Publ. No. 0–8186–7086–X/95, pp. 99–107.

Yamauchi, Tsukasa, et al., "SOP: A reconfigurable massively parallel system and its control–data flow based compiling method", © 1996 IEEE, Publ. No. 0–8186–7548–9/96, pp. 148–156.

W.H. Mangione–Smith and B.L. Hutchings. Configurable computing: The Road Ahead. In Proceedings of the Reconfigurable Architectures Workshop (RAW'97), pp. 81–96, 1997.

Mirsky, Ethan A., "Coarse–Grain Reconfigurable Computing", Massachusetts Institute of Technology, Jun. 1996.

Amerson, Rick, et al., "Teramac—Configurable Custom Computing", © 1995 IEEE, Publ. No. 0/8186–7086–X/95, pp. 32–38.

Casselman, Steven, "Virtual Computing and The Virtual Computer", © 1993 IEEE, Publ. No. 0–8186–3890–7/93, pp. 43–48.

Cuccaro, Steven, et al., "The CM–2X: A Hybrid CM–2 / Xilinx Prototype", © 1993 IEEE, Publ. No. 0–8186–3890–7/93, pp. 121–130.

Gunther, Bernard, et al., "Assessing Document Relevance with Run–Time Reconfigurable Machines", © 1996 IEEE, Publ. No. 0–8186–7548–9/96, pp. 10–17.

Hogl, H., et al., "Enable++: A second generation FPGA processor", © 1995 IEEE, Publ. No. 0–8186–7086–X/95, pp. 45–53.

Herpel, H.–J., et al., "A Reconfigurable Computer for Embedded Control Applications", © 1993 IEEE, Publ. No. 0–8186–3890–7/93, pp. 111–120.

Mirsky, Ethan, et al., "MATRIX: A Reconfigurable Computing Architecture with Configurable Instruction Distribution and Deployable Resources", © 1996 IEEE, Publ. No. 0–8186–7548–9/96, pp. 157–166.

Morley, Robert E., Jr., et al., "A Massively Parallel Systolic Array Processor System", © 1988 IEEE, Publ. No. CH2603–9/88/0000/0217, pp. 217–225.

Sitkoff, Nathan, et al., "Implementing a Genetic Algorithm on a Parallel Custom Computing Machine", Publ. No. 0–8186–7086–X/95, pp. 180–187.

Gokhale, M., et al., "Processing in Memory: The Terasys Massively Parallel PIM Array" © Apr. 1995, IEEE, pp. 23–31.

DeHon, A., et al., "Matrix A Reconfigurable Computing Device with Configurable Instruction Distribution", Hot Chips IX, Aug. 25–26, 1997, Stanford, California, MIT Artificial Intelligence Laboratory.

Buell, D., et al. "Splash 2: FPGAs in a Custom Computing Machine—Chapter 1—Custom Computing Machines: An Introduction", pp. 1–11, http://www.computer.org/espress/catalog/bp07413/spls–ch1.html (originally believed published in J. of Supercomputing, vol. IX, 1995, pp. 219–230.

Agarwal, A., et al., "The Raw Compiler Project", pp. 1–12, http://cag–www.lcs.mit.edu/raw, Proceedings of the Second SUIF Compiler Workshop, Aug. 21–23, 1997.

Hartenstein, R.W., et al. "A General Approach in System Design Integrating Reconfigurable Accelerators," http://xputers.informatik.uni–kl.de/papers/paper026–1.html, IEEE 1996 Conference, Austin, TX, Oct. 9–11, 1996.

* cited by examiner

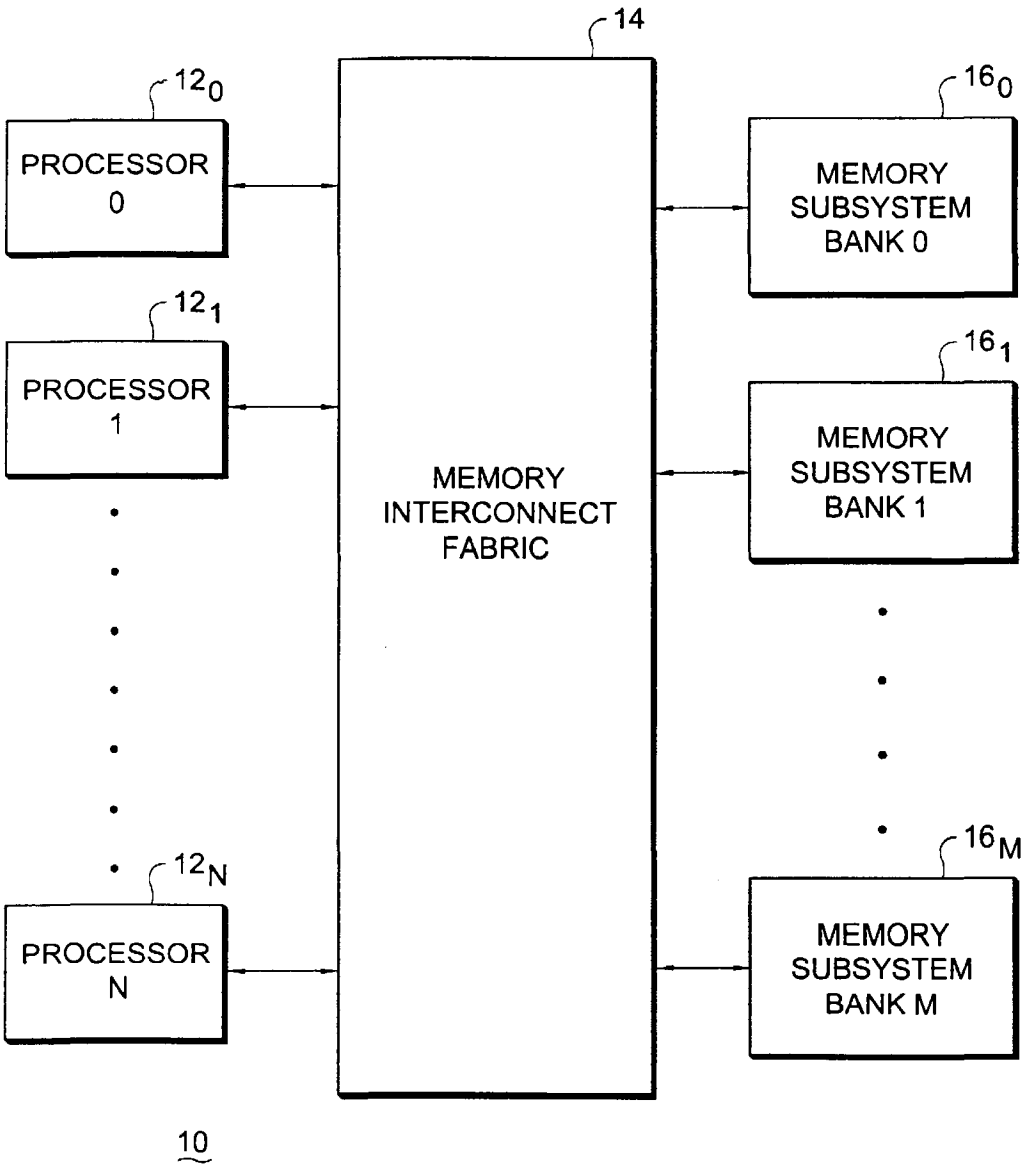

14
12_0
PROCESSOR 0
12_1
PROCESSOR 1
12_N
PROCESSOR N
MEMORY INTERCONNECT FABRIC
16_0
MEMORY SUBSYSTEM BANK 0
16_1
MEMORY SUBSYSTEM BANK 1
16_M
MEMORY SUBSYSTEM BANK M
10

FIG. 1

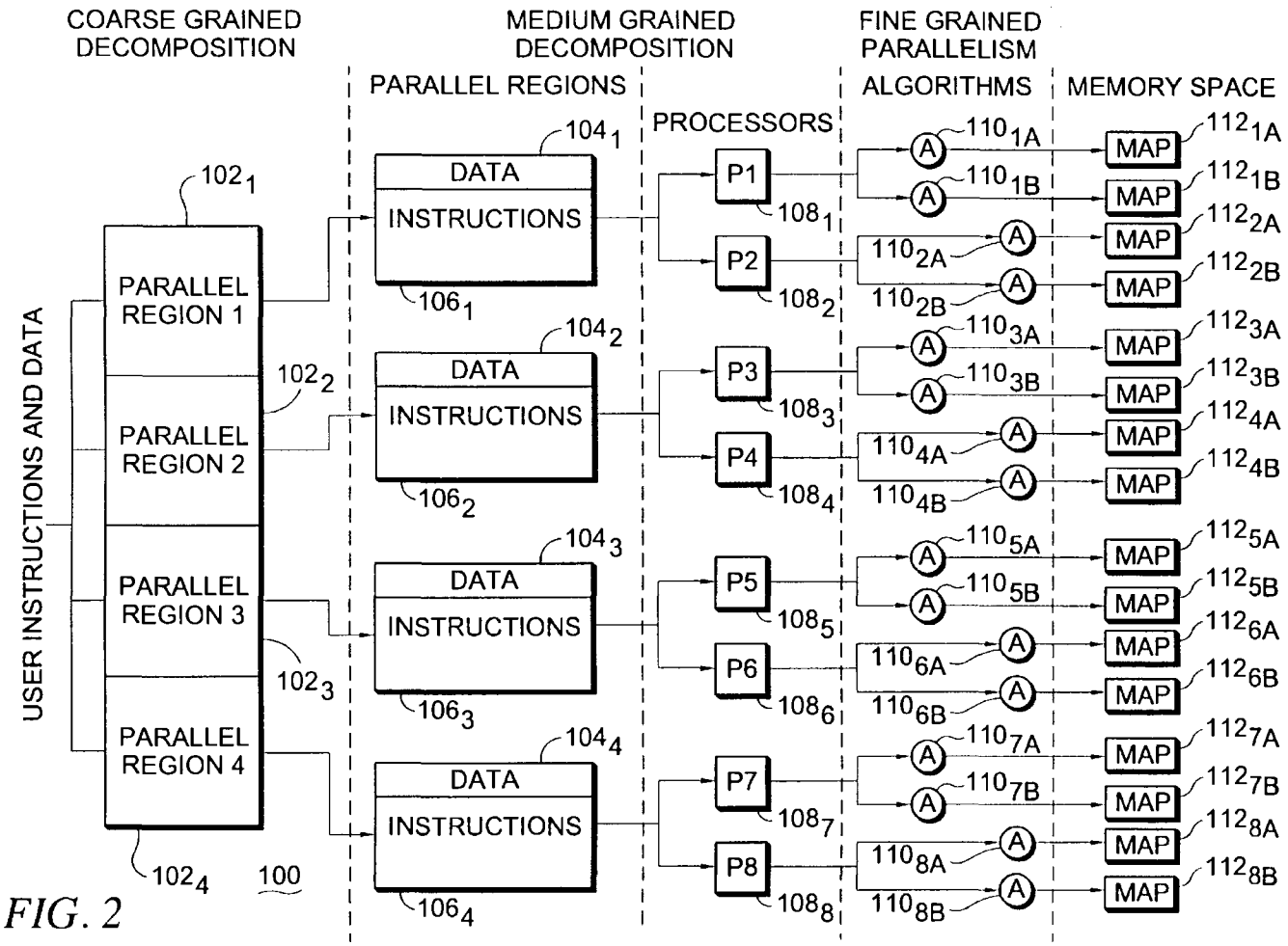

COARSE GRAINED DECOMPOSITION
MEDIUM GRAINED DECOMPOSITION
FINE GRAINED PARALLELISM
PARALLEL REGIONS
PROCESSORS
ALGORITHMS
MEMORY SPACE
USER INSTRUCTIONS AND DATA
PARALLEL REGION 1
PARALLEL REGION 2
PARALLEL REGION 3
PARALLEL REGION 4
DATA
INSTRUCTIONS
P1
P2
P3
P4
P5
P6
P7
P8
MAP
A
100

FIG. 2

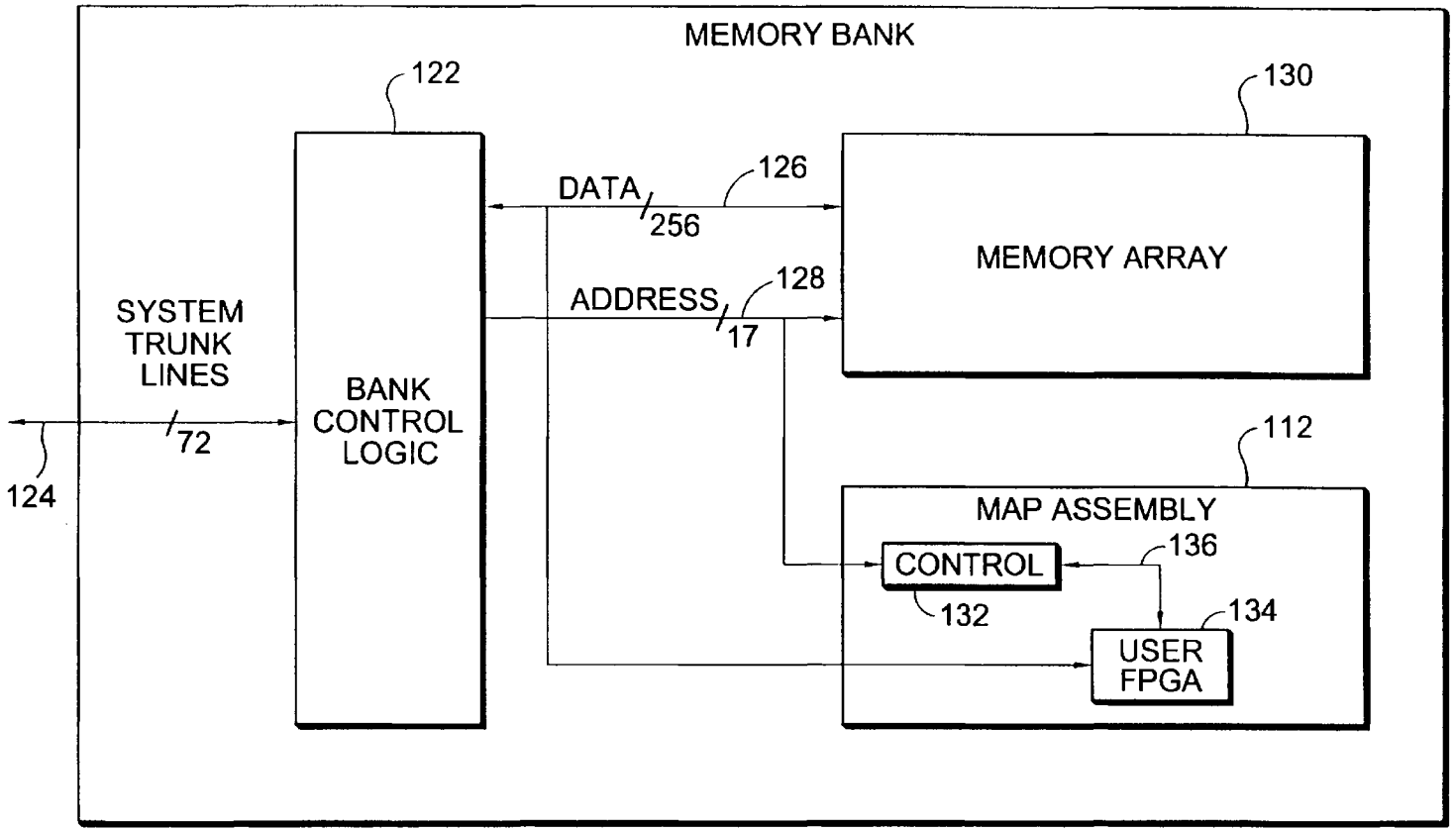

MEMORY BANK
122
130
DATA
256
126
MEMORY ARRAY
ADDRESS
17
128
SYSTEM
TRUNK
LINES
72
124
BANK
CONTROL
LOGIC
112
MAP ASSEMBLY
CONTROL
136
132
134
USER
FPGA
120

FIG. 3

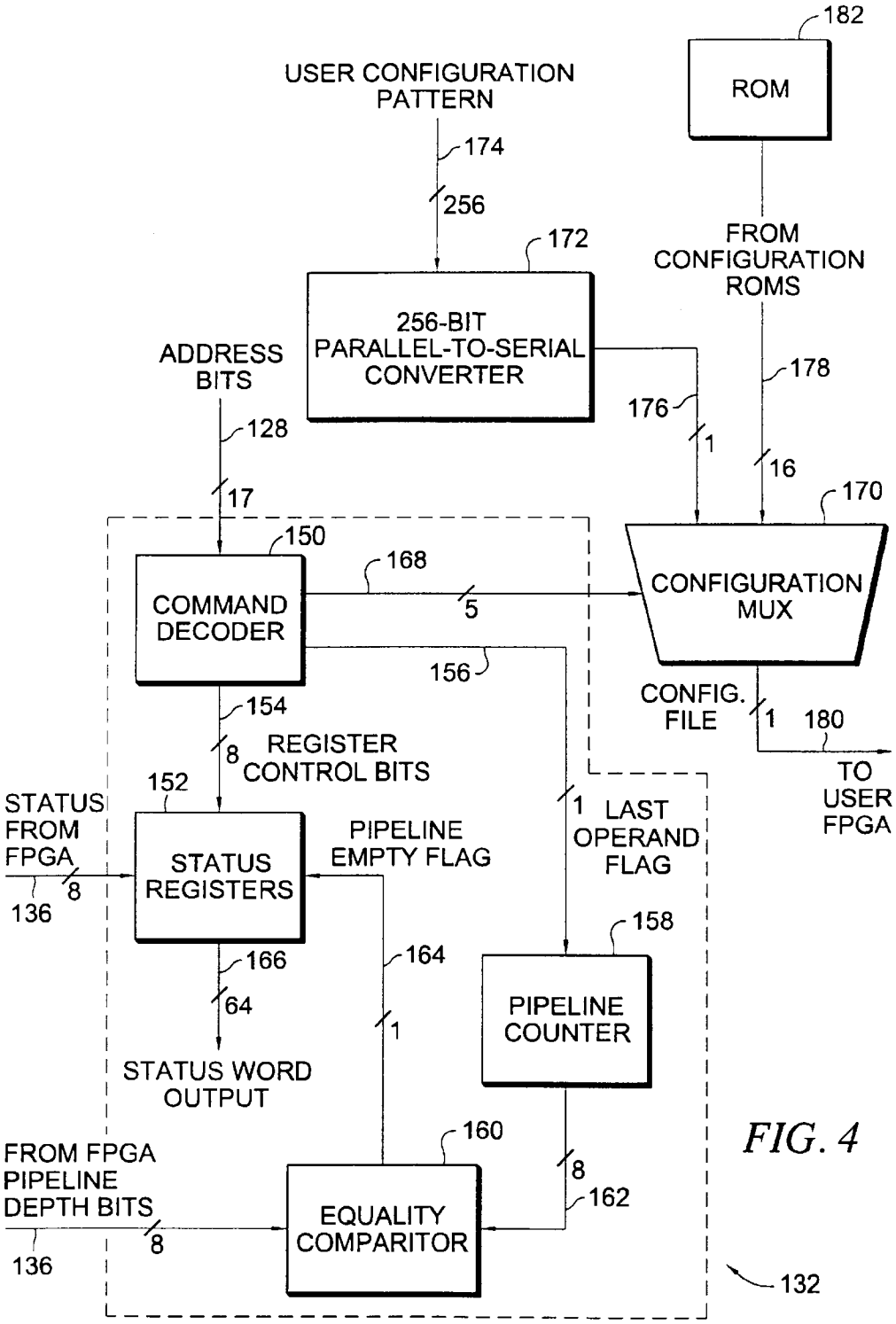

182
ROM
USER CONFIGURATION PATTERN
174
256
172
FROM CONFIGURATION ROMS
256-BIT PARALLEL-TO-SERIAL CONVERTER
ADDRESS BITS
128
176
178
1
16
17
170
150
168
COMMAND DECODER
5
CONFIGURATION MUX
156
CONFIG. FILE
1
180
TO USER FPGA
154
8
REGISTER CONTROL BITS
152
STATUS FROM FPGA
8
136
STATUS REGISTERS
PIPELINE EMPTY FLAG
1
LAST OPERAND FLAG
158
164
166
64
PIPELINE COUNTER
1
STATUS WORD OUTPUT
160
FROM FPGA PIPELINE DEPTH BITS
8
136
EQUALITY COMPARITOR
8
162
132

FIG. 4

MULTIPROCESSOR COMPUTER ARCHITECTURE INCORPORATING A PLURALITY OF MEMORY ALGORITHM PROCESSORS IN THE MEMORY SUBSYSTEM

RELATED APPLICATION

The present application is a continuation of U.S. patent application Ser. No. 08/992,763 filed Dec. 17, 1997 now U.S. Pat. No. 6,076,152, incorporated herein by referenced, which is assigned to the assignee of the present application.

BACKGROUND OF THE INVENTION

The present invention relates, in general, to the field of computer architectures incorporating multiple processing elements. More particularly, the present invention relates to a multiprocessor computer architecture incorporating a number of memory algorithm processors in the memory subsystem to significantly enhance overall system processing speed.

All general purpose computers are based on circuits that have some form of processing element. These may take the form of microprocessor chips or could be a collection of smaller chips coupled together to form a processor. In any case, these processors are designed to execute programs that are defined by a set of program steps. The fact that these steps, or commands, can be rearranged to create different end results using the same computer hardware is key to the computer's flexibility. Unfortunately, this flexibility dictates that the hardware then be designed to handle a variety of possible functions, which results in generally slower operation than would be the case were it able to be designed to handle only one particular function. On the other hand, a single function computer is inherently not a particularly versatile computer.

Recently, several groups have begun to experiment with creating a processor out of circuits that are electrically reconfigurable. This would allow the processor to execute a small set of functions more quickly and then be electrically reconfigured to execute a different small set. While this accelerates some program execution speeds, there are many functions that cannot be implemented well in this type of system due to the circuit densities that can be achieved in reconfigurable integrated circuits, such as 64-bit floating point math. In addition, all of these systems are presently intended to contain processors that operate alone. In high performance systems, this is not the case. Hundreds or even tens of thousands of processors are often used to solve a single problem in a timely manner. This introduces numerous issues that such reconfigurable computers cannot handle, such as sharing of a single copy of the operating system. In addition, a large system constructed from this type of custom hardware would naturally be very expensive to produce.

SUMMARY OF THE INVENTION

In response to these shortcomings, SRC Computers, Inc., Colorado Springs, Colo., assignee of the present invention, has developed a Memory Algorithm Processor ("MAP") multiprocessor computer architecture that utilizes very high performance microprocessors in conjunction with user reconfigurable hardware elements. These reconfigurable elements, referred to as MAPs, are globally accessible by all processors in the systems. In addition, the manufacturing cost and design time of a particular multiprocessor computer system is relatively low inasmuch as it can be built using industry standard, commodity integrated circuits and, in a preferred embodiment, each MAP may comprise a Field Programmable Gate Array ("FPGA") operating as a reconfigurable functional unit.

Particularly disclosed herein is the utilization of one or more FPGAs to perform user defined algorithms in conjunction with, and tightly coupled to, a microprocessor. More particularly, in a multiprocessor computer system, the FPGAs are globally accessible by all of the system processors for the purpose of executing user definable algorithms.

In a particular implementation of the present invention disclosed herein, a circuit is provided either within, or in conjunction with, the FPGAs which signals, by means of a control bit, when the last operand has completed its flow through the MAP, thereby allowing a given process to be interrupted and thereafter restarted. In a still more specific implementation, one or more read only memory ("ROM") integrated circuit chips may be coupled adjacent the FPGA to allow a user program to use a single command to select one of several possible algorithms pre-loaded in the ROM thereby decreasing system reconfiguration time.

Still further provided is a computer system memory structure which includes one or more FPGAs for the purpose of using normal memory access protocol to access it as well as being capable of direct memory access ("DMA") operation. In a multiprocessor computer system, FPGAs configured with DMA capability enable one device to feed results directly to another thereby allowing pipelining or parallelizing execution of a user defined algorithm located in the reconfigurable hardware. The system and method of the present invention also provide a user programmable performance monitoring capability and utilizes parallelizer software to automatically detect parallel regions of user applications containing algorithms that can be executed in programmable hardware.

Broadly, what is disclosed herein is a computer including at least one data processor for operating on user data in accordance with program instructions. The computer includes at least one memory array presenting a data and address bus and comprises a memory algorithm processor associated with the memory array and coupled to the data and address buses. The memory algorithm processor is configurable to perform at least one identified algorithm on an operand received from a write operation to the memory array.

Also disclosed herein is a multiprocessor computer including a first plurality of data processors for operating on user data in accordance with program instructions and a second plurality of memory arrays, each presenting a data and address bus. The computer comprises a memory algorithm processor associated with at least one of the second plurality of memory arrays and coupled to the data and address bus thereof. The memory algorithm processor is configurable to perform at least one identified algorithm on an operand received from a write operation to the associated one of the second plurality of memory arrays.

BRIEF DESCRIPTION OF THE DRAWINGS

The aforementioned and other features and objects of the present invention and the manner of attaining them will become more apparent and the invention itself will be best understood by reference to the following description of a preferred embodiment taken in conjunction with the accompanying drawings, wherein:

FIG. **1** is a simplified, high level, functional block diagram of a standard multiprocessor computer architecture;

FIG. **2** is a simplified logical block diagram of a possible computer application program decomposition sequence for use in conjunction with a multiprocessor computer architecture utilizing a number of memory algorithm processors ("MAPs") in accordance with the present invention;

FIG. **3** is a more detailed functional block diagram of an individual one of the MAPs of the preceding figure and illustrating the bank control logic, memory array and MAP assembly thereof; and

FIG. **4** is a more detailed functional block diagram of the control block of the MAP assembly of the preceding illustration illustrating its interconnection to the user FPGA thereof.

DESCRIPTION OF A PREFERRED EMBODIMENT

With reference now to FIG. **1**, a conventional multiprocessor computer **10** architecture is shown. The multiprocessor computer **10** incorporates N processors $\mathbf{12}_O$ through $\mathbf{12}_N$ which are bi-directionally coupled to a memory interconnect fabric **14**. The memory interconnect fabric **14** is then also coupled to M memory banks comprising memory bank subsystems $\mathbf{16}_0$ (Bank **0**) through $\mathbf{16}_M$ (Bank M).

With reference now to FIG. **2**, a representative application program decomposition for a multiprocessor computer architecture **100** incorporating a plurality of memory algorithm processors in accordance with the present invention is shown. The computer architecture **100** is operative in response to user instructions and data which, in a coarse grained portion of the decomposition, are selectively directed to one of (for purposes of example only) four parallel regions $\mathbf{102}_1$ through $\mathbf{102}_4$ inclusive. The instructions and data output from each of the parallel regions $\mathbf{102}_1$ through $\mathbf{102}_4$ are respectively input to parallel regions segregated into data areas $\mathbf{104}_1$ through $\mathbf{104}_4$ and instruction areas $\mathbf{106}_1$ through $\mathbf{106}_4$. Data maintained in the data areas $\mathbf{104}_1$ through $\mathbf{104}_4$ and instructions maintained in the instruction areas $\mathbf{106}_1$ through $\mathbf{106}_4$ are then supplied to, for example, corresponding pairs of processors $\mathbf{108}_1$, $\mathbf{108}_2$ (P1 and P2); $\mathbf{108}_3$, $\mathbf{108}_4$ (P3 and P4); $\mathbf{108}_5$, $\mathbf{108}_6$ (P5 and P6); and $\mathbf{108}_7$, $\mathbf{108}_8$ (P7 and P8) as shown. At this point, the medium grained decomposition of the instructions and data has been accomplished.

A fine grained decomposition, or parallelism, is effectuated by a further algorithmic decomposition wherein the output of each of the processors $\mathbf{108}_1$ through $\mathbf{108}_8$ is broken up, for example, into a number of fundamental algorithms $\mathbf{110}_{1A}$, $\mathbf{110}_{1B}$, $\mathbf{110}_{2A}$, $\mathbf{110}_{2B}$ through $\mathbf{110}_{8B}$ as shown. Each of the algorithms is then supplied to a corresponding one of the MAPs $\mathbf{112}_{1A}$, $\mathbf{112}_{1B}$, $\mathbf{112}_{2A}$, $\mathbf{112}_{2B}$ through $\mathbf{112}_{8B}$ in the memory space of the computer architecture **100** for execution therein as will be more fully described hereinafter.

With reference additionally now to FIG. **3**, a preferred implementation of a memory bank **120** in a MAP system computer architecture **100** of the present invention is shown for a representative one of the MAPs **112** illustrated in the preceding figure. Each memory bank **120** includes a bank control logic block **122** bi-directionally coupled to the computer system trunk lines, for example, a 72 line bus **124**. The bank control logic block **122** is coupled to a bi-directional data bus **126** (for example 256 lines) and supplies addresses on an address bus **128** (for example 17 lines) for accessing data at specified locations within a memory array **130**.

The data bus **126** and address bus **128** are also coupled to a MAP assembly **112**. The MAP assembly **112** comprises a control block **132** coupled to the address bus **128**. The control block **132** is also bi-directionally coupled to a user field programmable gate array ("FPGA") **134** by means of a number of signal lines **136**. The user FPGA **134** is coupled directly to the data bus **126**. In a particular embodiment, the FPGA **134** may be provided as a Lucent Technologies OR3T80 device.

The computer architecture **100** comprises a multiprocessor system employing uniform memory access across common shared memory with one or more MAPs **112** located in the memory subsystem, or memory space. As previously described, each MAP **112** contains at least one relatively large FPGA **134** that is used as a reconfigurable functional unit. In addition, a control block **132** and a preprogrammed or dynamically programmable configuration read-only memory ("ROM" as will be more fully described hereinafter) contains the information needed by the reconfigurable MAP assembly **112** to enable it to perform a specific algorithm. It is also possible for the user to directly download a new configuration into the FPGA **134** under program control, although in some instances this may consume a number of memory accesses and might result in an overall decrease in system performance if the algorithm was short-lived.

FPGAs have particular advantages in the application shown for several reasons. First, commercially available, off-the-shelf FPGAs now contain sufficient internal logic cells to perform meaningful computational functions. Secondly, they can operate at speeds comparable to microprocessors, which eliminates the need for speed matching buffers. Still further, the internal programmable routing resources of FPGAs are now extensive enough that meaningful algorithms can now be programmed without the need to reassign the locations of the input/output ("I/O") pins.

By placing the MAP **112** in the memory subsystem or memory space, it can be readily accessed through the use of memory read and write commands, which allows the use of a variety of standard operating systems. In contrast, other conventional implementations propose placement of any reconfigurable logic in or near the processor. This is much less effective in a multiprocessor environment because only one processor has rapid access to it. Consequently, reconfigurable logic must be placed by every processor in a multiprocessor system, which increases the overall system cost. In addition, MAP **112** can access the memory array **130** itself, referred to as Direct Memory Access ("DMA"), allowing it to execute tasks independently and asynchronously of the processor. In comparison, were it were placed near the processor, it would have to compete with the processors for system routing resources in order to access memory, which deleteriously impacts processor performance. Because MAP **112** has DMA capability, (allowing it eo write to memory), and because it receives its operands via writes to memory, it is possible to allow a MAP **112** to feed results to another MAP **112**. This is a very powerful feature that allows for very extensive pipelining and parallelizing of large tasks, which permits them to complete faster.

Many of the algorithms that may be implemented will receive an operand and require many clock cycles to produce a result. One such example may be a multiplication that takes 64 clock cycles. This same multiplication may also need to be performed on thousands of operands. In this situation, the incoming operands would be presented sequentially so that while the first operand requires 64 clock cycles to produce results at the output, the second operand, arriving one clock cycle later at the input, will show results

one clock cycle later at the output. Thus, after an initial delay of 64 clock cycles, new output data will appear on every consecutive clock cycle until the results of the last operand appears. This is called "pipelining".

In a multiprocessor system, it is quite common for the operating system to stop a processor in the middle of a task, reassign it to a higher priority task, and then return it, or another, to complete the initial task. When this is combined with a pipelined algorithm, a problem arises (if the processor stops issuing operands in the middle of a list and stops accepting results) with respect to operands already issued but not yet through the pipeline. To handle this issue, a solution involving the combination of software and hardware is disclosed herein.

To make use of any type of conventional reconfigurable hardware, the programmer could embed the necessary commands in his application program code. The drawback to this approach is that a program would then have to be tailored to be specific to the MAP hardware. The system of the present invention eliminates this problem. Multiprocessor computers often use software called parallelizers. The purpose of this software is to analyze the user's application code and determine how best to split it up among the processors. The present invention provides significant advantages over a conventional parallelizer and enables it to recognize portions of the user code that represent algorithms that exist in MAPs **112** for that system and to then treat the MAP **112** as another computing element. The parallelizer then automatically generates the necessary code to utilize the MAP **112**. This allows the user to write the algorithm directly in his code, allowing it to be more portable and reducing the knowledge of the system hardware that he has to have to utilize the MAP **112**.

With reference additionally now to FIG. **4**, a block diagram of the MAP control block **132** is shown in greater detail. The control block **132** is coupled to receive a number of command bits (for example, 17) from the address bus **128** at a command decoder **150**. The command decoder **150** then supplies a number of register control bits to a group of status registers **152** on an eight bit bus **154**. The command decoder **150** also supplies a single bit last operand flag on line **156** to a pipeline counter **158**. The pipeline counter **158** supplies an eight bit output to an equality comparitor **160** on bus **162**. The equality comparitor **160** also receives an eight bit signal from the FPGA **134** on bus **136** indicative of the pipeline depth. When the equality comparitor determines that the pipeline is empty, it provides a single bit pipeline empty flag on line **164** for input to the status registers **152**. The status registers are also coupled to receive an eight bit status signal from the FPGA **134** on bus **136** and it produces a sixty four bit status word output or bus **166** in response to the signals on bus **136**, **154** and line **164**.

The command decoder **150** also supplies a five bit control signal to a configuration multiplexer ("MUX") **170** as shown. The configuration mux **170** receives a single bit output of a 256 bit parallel-serial converter **172** on line **176**. The inputs of the 256 bit parallel-to-serial converter **172** are coupled to a 256 bit user configuration pattern bus **174**. The configuration mux **170** also receives sixteen single bit inputs from the configuration ROMs (illustrated as ROM **182**) on bus **178** and provides a single bit configuration file signal on line **180** to the user FPGA **134** as selected by the control signals from the command decoder **150** on the bus **168**.

In operation, when a processor **108** is halted by the operating system, the operating system will issue a last operand command to the MAP **112** through the use of command bits embedded in the address field on bus **128**. This command is recognized by the command decoder **150** of the control block **132** and it initiates a hardware pipeline counter **158**. When the algorithm was initially loaded into the FPGA **134**, several output bits connected to the control block **132** were configured to display a binary representation of the number of clock cycles required to get through its pipeline (i.e. pipeline "depth") on bus **136** input to the equality comparitor **160**. After receiving the last operand command, the pipeline counter **158** in the control block **132** counts clock cycles until its count equals the pipeline depth for that particular algorithm. At that point, the equality comparitor **160** in the control block **132** de-asserts a busy bit on line **164** in an internal group of status registers **152**. After issuing the last operand signal, the processor **108** will repeatedly read the status registers **152** and accept any output data on bus **166**. When the busy flag is de-asserted, the task can be stopped and the MAP **112** utilized for a different task. It should be noted that it is also possible to leave the MAP **112** configured, transfer the program to a different processor **108** and restart the task where it left off.

In order to evaluate the effectiveness of the use of the MAP **112** in a given application, some form of feedback to the use is required. Therefore, the MAP **112** may be equipped with internal registers in the control block **132** that allow it to monitor efficiency related factors such as the number of input operands versus output data, the number of idle cycles over time and the number of system monitor interrupts received over time. One of the advantages that the MAP **112** has is that because of its reconfigurable nature, the actual function and type of function that are monitored can also change as the algorithm changes. This provides the user with an almost infinite number of possible monitored factors without having to monitor all factors all of the time.

While there have been described above the principles of the present invention in conjunction with a specific multiprocessor architecture it is to be clearly understood that the foregoing description is made only by way of example and not as a limitation to the scope of the invention. Particularly, it is recognized that the teachings of the foregoing disclosure will suggest other modifications to those persons skilled in the relevant art. Such modifications may involve other features which are already known per se and which may be used instead of or in addition to features already described herein. Although claims have been formulated in this application to particular combinations of features, it should be understood that the scope of the disclosure herein also includes any novel feature or any novel combination of features disclosed either explicitly or implicitly or any generalization or modification thereof which would be apparent to persons skilled in the relevant art, whether or not such relates to the same invention as presently claimed in any claim and whether or not it mitigates any or all of the same technical problems as confronted by the present invention. The applicants hereby reserve the right to formulate new claims to such features and/or combinations of such features during the prosecution of the present application or of any further application derived therefrom.

What is claimed is:

1. In a computer system having at least one data processor for executing an application program by operating on user data in accordance with application program instructions, said computer system having at least one memory bank with a data bus and an address bus connected to said at least one data processor, the improvement comprising:

a plurality of reconfigurable memory algorithm processors within individually addressable portions of said memory bank,

means connecting said plurality of memory algorithm processors to said data bus and to said address bus such that said plurality of memory algorithm processors are individually memory addressable by said at least one data processor at said one data processor executes said application program; and

said plurality of memory algorithm processors being configured as individual data processing elements to perform data processing related to said application program in accordance with an identified algorithm, said data processing being performed on at least one operand that is received directly from said at least one data processor.

2. The improvement of claim **1** wherein each of said plurality of memory algorithm processors comprises a field programmable gate array.

3. The improvement of claim **1** wherein each of said plurality of memory algorithm processors is operative to memory address said memory bank independent of said at least one data processor.

4. The improvement of claim **1** wherein an identified algorithm is preprogrammed into each of said plurality of memory algorithm processors.

5. The improvement of claim **4** wherein a plurality of identified algorithms are preprogrammed into a memory device that is associated with said plurality of memory algorithm processors.

6. The improvement of claim **5** wherein said memory device comprises at least one read only memory device.

7. The improvement of claim **1** wherein any given one of said plurality of memory algorithm processors is operative to pass a data processing result of an operand that has been processed by an identified algorithm to another of said plurality of memory algorithm processors.

8. The improvement of claim **1** wherein said plurality of memory algorithm processors comprise a memory algorithm processor assembly, said memory algorithm processor assembly including:

a control block having a command decoder coupled to said address bus and having a pipeline counter coupled to said command decoder;

said command decoder for providing a last operand flag to said pipeline counter in response to a last operand command from an operating system of said at least one data processor.

9. The improvement of claim **8** wherein said control block further includes:

at least one status register; and

an equality comparator coupled to receive a pipeline depth signal and an output of said pipeline counter, said equality comparator for providing a pipeline empty flag to said at least one status register.

10. The improvement of claim **9** wherein said at least one status register is coupled to said command decoder to receive a register control signal and is coupled to said plurality of memory algorithm processors to receive a status signal, said at least one status register providing a status word output signal.

11. A multiprocessor computer system comprising:

a plurality of data processors for executing at least one application program by operating on user data in accordance with program instructions;

a memory bank having a data bus and an address bus connected to said plurality of data processors;

a plurality of reconfigurable memory algorithm processors within said memory bank at plurality of individual memory addressable memory locations;

means coupling said plurality of individual memory algorithm processors to said data bus and to said address bus;

said plurality of reconfigurable memory algorithm processors being individually memory addressable by all of said plurality of data processors; and

said plurality of memory algorithm processors being individually configurable to perform an identified algorithm on an operand that is received from a write operation by one of said plurality of data processors to said memory bank as said at least one of said plurality of data processors executes said at least one application program.

12. The multiprocessor computer system of claim **11** wherein all of said plurality of memory algorithm processors are memory addressable by all of said plurality of data processors.

13. The multiprocessor computer system of claim **12** wherein all of said plurality of memory algorithm processors are mutually memory addressable.

14. The multiprocessor computer system of claim **13** wherein said plurality of memory algorithm processors collectively comprises a memory algorithm processor assembly, said memory algorithm processor assembly including:

a control block operative to provide a last operand flag in response to a last operand having been processed by said memory algorithm processor assembly.

15. The multiprocessor computer system of claim **11** including:

at least one memory device associated with said plurality of memory algorithm processors for storing a plurality of pre loaded identified algorithms.

16. The multiprocessor computer system of claim **15** wherein said at least one memory device is responsive to a predetermined command from a data processor and operates in response thereto to selected one of said plurality of pre-loaded identified algorithms to be implemented by an addressed one of said plurality of memory algorithm processors.

17. The multiprocessor computer system of claim **16** wherein said at least one memory device comprises at least one read only memory device.

18. The multiprocessor computer system of claim **11** wherein each of said plurality of memory algorithm processors comprises a field programmable gate array.

19. The multiprocessor computer system of claim **11** wherein each of said plurality of memory algorithm processors is memory accessible through normal memory access protocol.

20. The multiprocessor computer system of claim **11** wherein each of said plurality of memory algorithm processors has direct memory access capability to said memory bank.

21. The multiprocessor computer system of claim **11** wherein each of said plurality of memory algorithm processors is operative to pass a result of a processed operand to another memory algorithm processor.

22. The multiprocessor computer system of claim **11** operative to detect at least one parallel region of said at least one application program, wherein at least one of said plurality of memory algorithm processors is configured as a function of said detected at least one parallel region of said at least one application program.

* * * * *

EXHIBIT B

US006076152A

United States Patent [19]

Huppenthal et al.

[11] **Patent Number: 6,076,152**

[45] **Date of Patent: Jun. 13, 2000**

[54] **MULTIPROCESSOR COMPUTER ARCHITECTURE INCORPORATING A PLURALITY OF MEMORY ALGORITHM PROCESSORS IN THE MEMORY SUBSYSTEM**

[75] Inventors: **Jon M. Huppenthal**; **Paul A. Leskar**, both of Colorado Springs, Colo.

[73] Assignee: **SRC Computers, Inc.**, Colorado Springs, Colo.

[21] Appl. No.: **08/992,763**

[22] Filed: **Dec. 17, 1997**

[51] **Int. Cl.**[7] **G06F 15/80**

[52] **U.S. Cl.** **712/15**

[58] **Field of Search** 712/13, 15, 2, 712/21, 22, 28; 708/232

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,230,057	7/1993	Shido et al.	712/22
5,892,962	4/1999	Cloutier	712/16

OTHER PUBLICATIONS

Yun et al., A Distributed Memory MIMD Multi–Computer with Reconfigurable Custom Computing Capabilities IEEE, Dec. 13, 1997, pp. 8/13 especially pp. 9–11.

Gokhale, M., et al., "Processing in Memory: The Terasys Massively Parellel PIM Array"©Apr. 1995, IEEE, pp. 23–31.

DeHon, A., et al., "MATRIX A Reconfigurable Computing Device with Configurable Instruction Distribution", Hot Chips IX, Aug. 25–26, 1997, Stanford, California, MIT Artifical Intelligence Laboratory.

Buell, D., et al., "Splash 2: FPGAs in a Custom Computing Machine–Chapter 1–Custom Computing Machines: An Introduction", pp. 1–11, *http://ww.computer.org/espress/catalog/bp07413/spls–ch1.html*(originally believed published in J.of Supercomputing, voil. IX, 1995, pp. 219–230.

Agarwal, A, et al., "The Raw Compiler Project", pp. 1–12, *http://cag–www.lcs.mit.edu/raw*, Proceedings of the Second SUIF Compiler Workshop, Aug. 21–23,1997.

Hartenstein, R.W. et al. "A General Approach in System Design Intergrated Reconfigurable Accelerators", *http://xputers.informatik.uni–kl.de/papers026–1.html*, IEEE 1996 Conference, Austin, TX, Oct. 9–11, 1996.

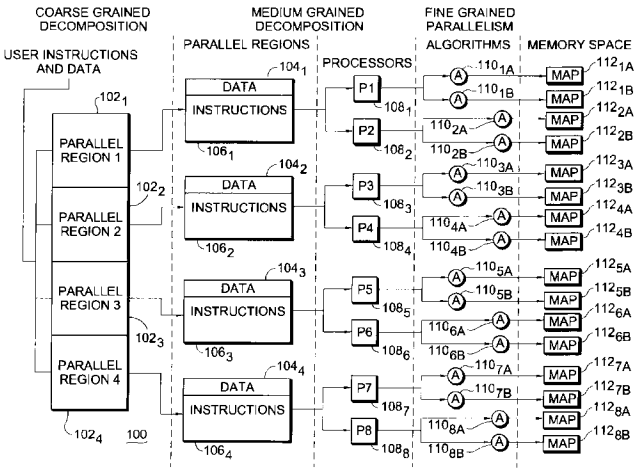
Primary Examiner—John A. Follansbee

Attorney, Agent, or Firm—William J. Kubida, Esq.; Hogan & Hartgon

[57] **ABSTRACT**

A multiprocessor computer architecture incorporating a plurality of programmable hardware memory algorithm processors ("MAP") in the memory subsystem. The MAP may comprise one or more field programmable gate arrays ("FPGAs") which function to perform identified algorithms in conjunction with, and tightly coupled to, a microprocessor and each MAP is globally accessible by all of the system processors for the purpose of executing user definable algorithms. A circuit within the MAP signals when the last operand has completed its flow thereby allowing a given process to be interrupted and thereafter restarted. Through the use of read only memory ("ROM") located adjacent the FPGA, a user program may use a single command to select one of several possible pre-loaded algorithms thereby decreasing system reconfiguration time. A computer system memory structure MAP disclosed herein may function in normal or direct memory access ("DMA") modes of operation and, in the latter mode, one device may feed results directly to another thereby allowing pipelining or parallelizing execution of a user defined algorithm. The system of the present invention also provides a user programmable performance monitoring capability and utilizes parallelizer software to automatically detect parallel regions of user applications containing algorithms that can be executed in the programmable hardware.

22 Claims, 4 Drawing Sheets



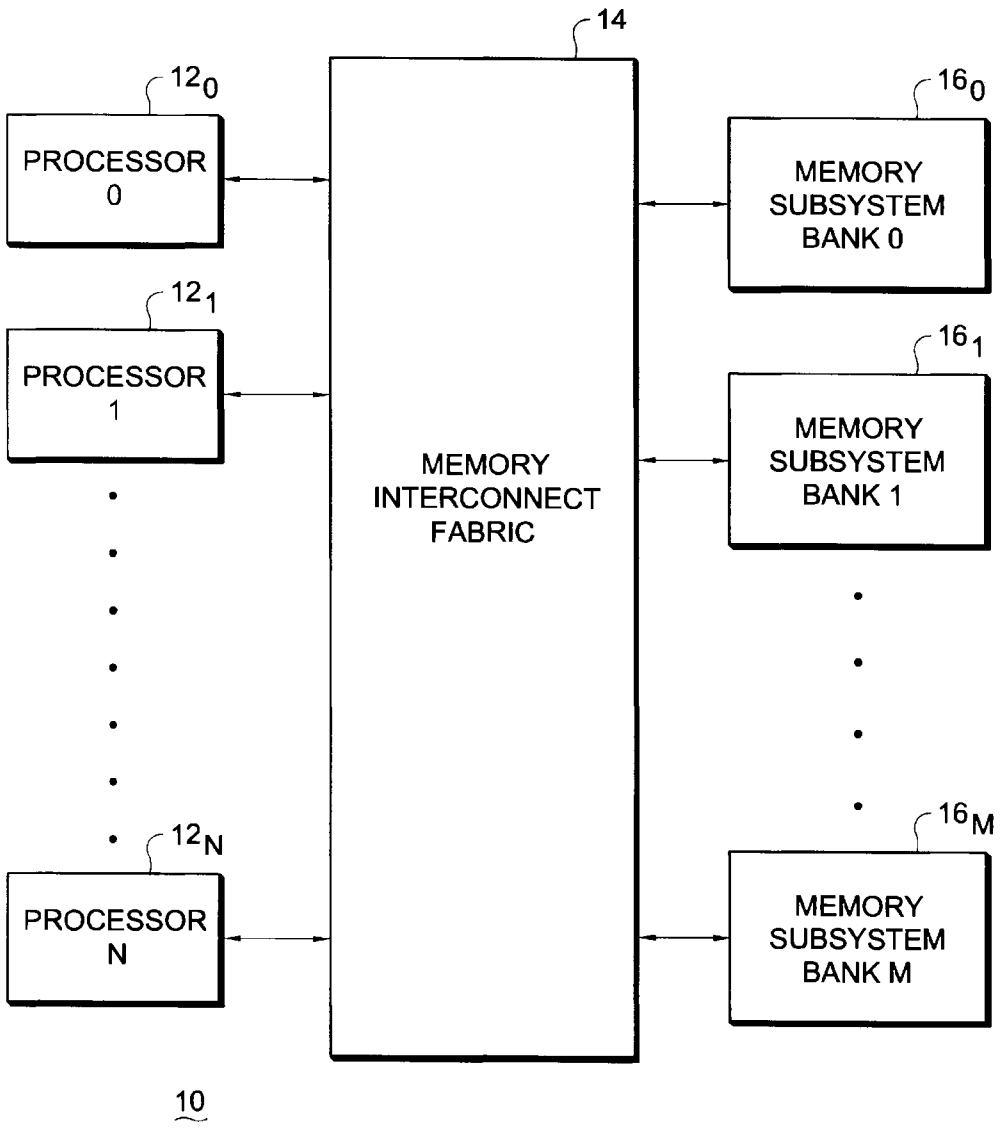

14
12 0
PROCESSOR 0
12 1
PROCESSOR 1
12 N
PROCESSOR N
MEMORY INTERCONNECT FABRIC
16 0
MEMORY SUBSYSTEM BANK 0
16 1
MEMORY SUBSYSTEM BANK 1
16 M
MEMORY SUBSYSTEM BANK M
10

FIG. 1

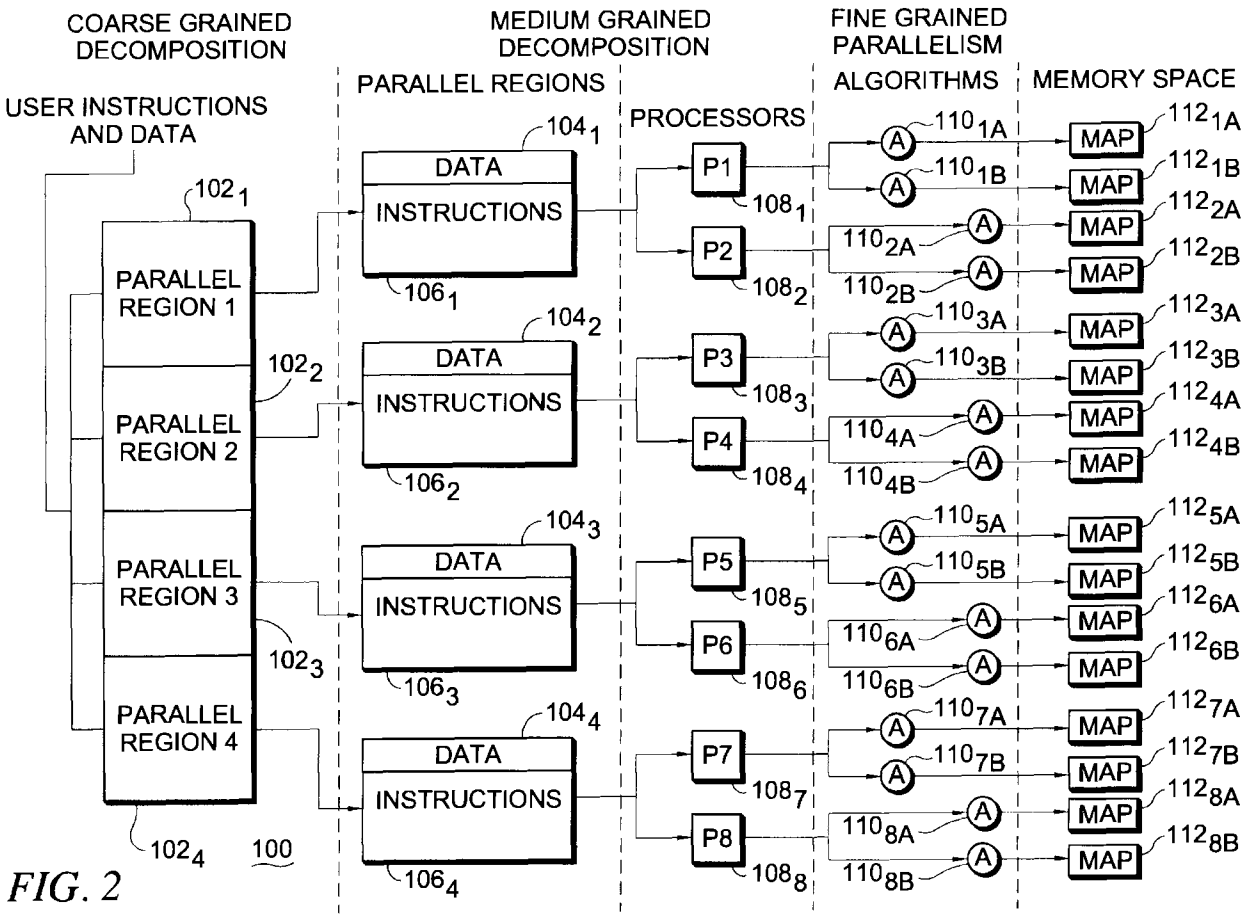

COARSE GRAINED DECOMPOSITION
MEDIUM GRAINED DECOMPOSITION
FINE GRAINED PARALLELISM
USER INSTRUCTIONS AND DATA
PARALLEL REGIONS
PROCESSORS
ALGORITHMS
MEMORY SPACE
PARALLEL REGION 1
PARALLEL REGION 2
PARALLEL REGION 3
PARALLEL REGION 4
DATA
INSTRUCTIONS
P1
P2
P3
P4
P5
P6
P7
P8
A
MAP
100

FIG. 2

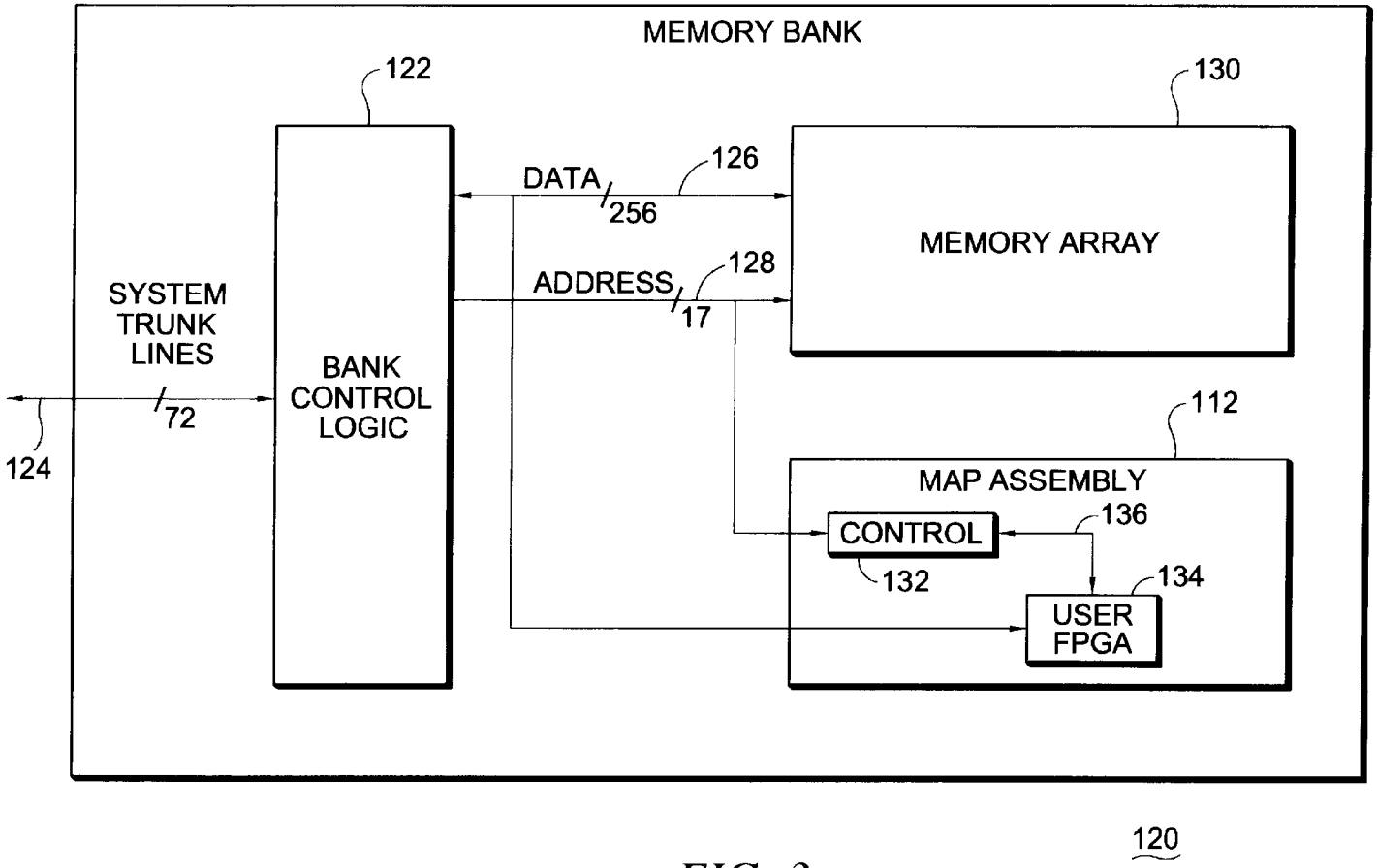

MEMORY BANK
122
130
DATA
256
126
MEMORY ARRAY
ADDRESS
17
128
SYSTEM
TRUNK
LINES
72
124
BANK
CONTROL
LOGIC
112
MAP ASSEMBLY
CONTROL
136
132
134
USER
FPGA
120

FIG. 3

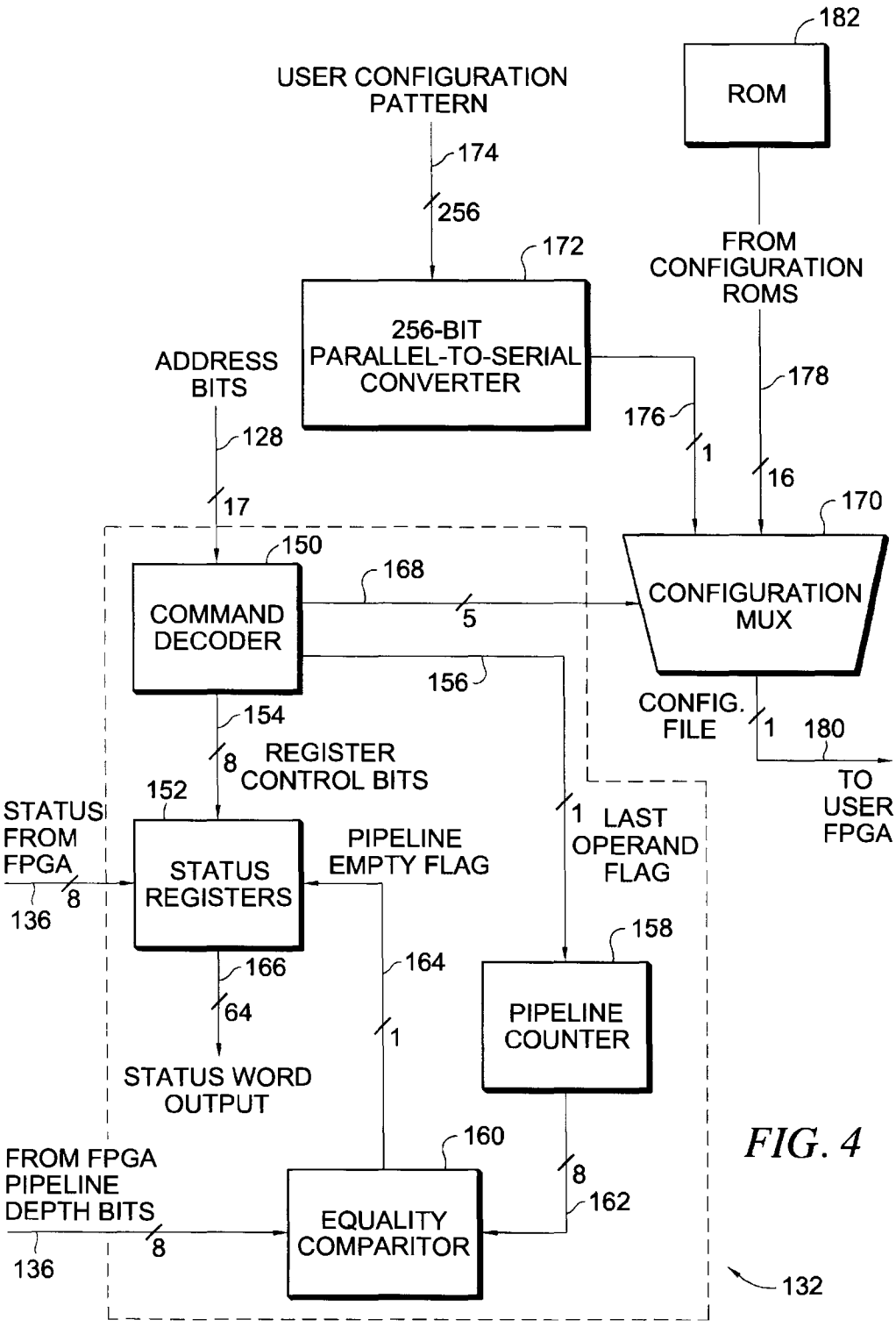

182
ROM
USER CONFIGURATION PATTERN
174
256
172
256-BIT PARALLEL-TO-SERIAL CONVERTER
FROM CONFIGURATION ROMS
178
ADDRESS BITS
128
17
176
1
16
170
150
COMMAND DECODER
168
5
CONFIGURATION MUX
156
CONFIG. FILE
1
180
TO USER FPGA
154
8
REGISTER CONTROL BITS
152
STATUS FROM FPGA
8
136
STATUS REGISTERS
PIPELINE EMPTY FLAG
1
LAST OPERAND FLAG
158
166
64
164
1
PIPELINE COUNTER
STATUS WORD OUTPUT
160
FROM FPGA PIPELINE DEPTH BITS
8
136
EQUALITY COMPARITOR
8
162
132

FIG. 4

MULTIPROCESSOR COMPUTER ARCHITECTURE INCORPORATING A PLURALITY OF MEMORY ALGORITHM PROCESSORS IN THE MEMORY SUBSYSTEM

BACKGROUND OF THE INVENTION

The present invention relates, in general, to the field of computer architectures incorporating multiple processing elements. More particularly, the present invention relates to a multiprocessor computer architecture incorporating a number of memory algorithm processors in the memory subsystem to significantly enhance overall system processing speed.

All general purpose computers are based on circuits that have some form of processing element. These may take the form of microprocessor chips or could be a collection of smaller chips coupled together to form a processor. In any case, these processors are designed to execute programs that are defined by a set of program steps. The fact that these steps, or commands, can be rearranged to create different end results using the same computer hardware is key to the computer's flexibility. Unfortunately, this flexibility dictates that the hardware then be designed to handle a variety of possible functions, which results in generally slower operation than would be the case were it able to be designed to handle only one particular function. On the other hand, a single function computer is inherently not a particularly versatile computer.

Recently, several groups have begun to experiment with creating a processor out of circuits that are electrically reconfigurable. This would allow the processor to execute a small set of functions more quickly and then be electrically reconfigured to execute a different small set. While this accelerates some program execution speeds, there are many functions that cannot be implemented well in this type of system due to the circuit densities that can be achieved in reconfigurable integrated circuits, such as 64-bit floating point math. In addition, all of these systems are presently intended to contain processors that operate alone. In high performance systems, this is not the case. Hundreds or even tens of thousands of processors are often used to solve a single problem in a timely manner. This introduces numerous issues that such reconfigurable computers cannot handle, such as sharing of a single copy of the operating system. In addition, a large system constructed from this type of custom hardware would naturally be very expensive to produce.

SUMMARY OF THE INVENTION

In response to these shortcomings, SRC Computers, Inc., Colorado Springs, Colo., assignee of the present invention, has developed a Memory Algorithm Processor ("MAP") multiprocessor computer architecture that utilizes very high performance microprocessors in conjunction with user reconfigurable hardware elements. These reconfigurable elements, referred to as MAPs, are globally accessible by all processors in the systems. In addition, the manufacturing cost and design time of a particular multiprocessor computer system is relatively low inasmuch as it can be built using industry standard, commodity integrated circuits and, in a preferred embodiment, each MAP may comprise a Field Programmable Gate Array ("FPGA") operating as a reconfigurable functional unit.

Particularly disclosed herein is the utilization of one or more FPGAs to perform user defined algorithms in conjunction with, and tightly coupled to, a microprocessor. More particularly, in a multiprocessor computer system, the FPGAs are globally accessible by all of the system processors for the purpose of executing user definable algorithms.

In a particular implementation of the present invention disclosed herein, a circuit is provided either within, or in conjunction with, the FPGAs which signals, by means of a control bit, when the last operand has completed its flow through the MAP, thereby allowing a given process to be interrupted and thereafter restarted. In a still more specific implementation, one or more read only memory ("ROM") integrated circuit chips may be coupled adjacent the FPGA to allow a user program to use a single command to select one of several possible algorithms pre-loaded in the ROM thereby decreasing system reconfiguration time.

Still further provided is a computer system memory structure which includes one or more FPGAs for the purpose of using normal memory access protocol to access it as well as being capable of direct memory access ("DMA") operation. In a multiprocessor computer system, FPGAs configured with DMA capability enable one device to feed results directly to another thereby allowing pipelining or parallelizing execution of a user defined algorithm located in the re-configurable hardware. The system and method of the present invention also provide a user programmable performance monitoring capability and utilizes parallelizer software to automatically detect parallel regions of user applications containing algorithms that can be executed in programmable hardware.

Broadly, what is disclosed herein is a computer including at least one data processor for operating on user data in accordance with program instructions. The computer includes at least one memory array presenting a data and address bus and comprises a memory algorithm processor associated with the memory array and coupled to the data and address buses. The memory algorithm processor is configurable to perform at least one identified algorithm on an operand received from a write operation to the memory array.

Also disclosed herein is a multiprocessor computer including a first plurality of data processors for operating on user data in accordance with program instructions and a second plurality of memory arrays, each presenting a data and address bus. The computer comprises a memory algorithm processor associated with at least one of the second plurality of memory arrays and coupled to the data and address bus thereof. The memory algorithm processor is configurable to perform at least one identified algorithm on an operand received from a write operation to th e associated one of the second plurality of memory arrays.

BRIEF DESCRIPTION OF THE DRAWINGS

The aforementioned and other features and objects of the present invention and the manner of attaining them will become more apparent and the invention itself will be best understood by reference to the following description of a preferred embodiment taken in conjunction with the accompanying drawings, wherein:

FIG. **1** is a simplified, high level, functional block diagram of a standard multiprocessor computer architecture;

FIG. **2** is a simplified logical block diagram of a possible computer application program decomposition sequence for use in conjunction with a multiprocessor computer architecture utilizing a number of memory algorithm processors ("MAPs") in accordance with the present invention;

FIG. **3** is a more detailed functional block diagram of an individual one of the MAPs of the preceding figure and

illustrating the bank control logic, memory array and MAP assembly thereof; and

FIG. **4** is a more detailed functional block diagram of the control block of the MAP assembly of the preceding illustration illustrating its interconnection to the user FPGA thereof.

DESCRIPTION OF A PREFERRED EMBODIMENT

With reference now to FIG. **1**, a conventional multiprocessor computer **10** architecture is shown. The multiprocessor computer **10** incorporates N processors $\mathbf{12}_O$ through $\mathbf{12}_N$ which are bi-directionally coupled to a memory interconnect fabric **14**. The memory interconnect fabric **14** is then also coupled to M memory banks comprising memory bank subsystems $\mathbf{16}_0$ (Bank **0**) through $\mathbf{16}_M$ (Bank M).

With reference now to FIG. **2**, a representative application program decomposition for a multiprocessor computer architecture **100** incorporating a plurality of memory algorithm processors in accordance with the present invention is shown. The computer architecture **100** is operative in response to user instructions and data which, in a coarse grained portion of the decomposition, are selectively directed to one of (for purposes of example only) four parallel regions $\mathbf{102}_1$ through $\mathbf{102}_4$ inclusive. The instructions and data output from each of the parallel regions $\mathbf{102}_1$ through $\mathbf{102}_4$ are respectively input to parallel regions segregated into data areas $\mathbf{104}_1$ through $\mathbf{104}_4$ and instruction areas $\mathbf{106}_1$ through $\mathbf{106}_4$. Data maintained in the data areas $\mathbf{104}_1$ through $\mathbf{104}_4$ and instructions maintained in the instruction areas $\mathbf{106}_1$ through $\mathbf{106}_4$ are then supplied to, for example, corresponding pairs of processors $\mathbf{108}_1$, $\mathbf{108}_2$ (P1 and P2); $\mathbf{108}_3$, $\mathbf{108}_4$ (P3 and P4); $\mathbf{108}_5$, $\mathbf{108}_6$ (P5 and P6) ; and $\mathbf{108}_7$, $\mathbf{108}_8$ (P7 and P8) as shown. At this point, the medium grained decomposition of the instructions and data has been accomplished.

A fine grained decomposition, or parallelism, is effectuated by a further algorithmic decomposition wherein the output of each of the processors $\mathbf{108}_1$ through $\mathbf{108}_8$ is broken up, for example, into a number of fundamental algorithms $\mathbf{110}_{1A}$, $\mathbf{110}_{1B}$, $\mathbf{110}_{2A}$, $\mathbf{110}_{2B}$ through $\mathbf{110}_{8B}$ as shown. Each of the algorithms is then supplied to a corresponding one of the MAPs $\mathbf{112}_{1A}$, $\mathbf{112}_{1B}$, $\mathbf{112}_{2A}$, $\mathbf{112}_{2B}$ through $\mathbf{112}_{8B}$ in the memory space of the computer architecture **100** for execution therein as will be more fully described hereinafter.

With reference additionally now to FIG. **3**, a preferred implementation of a memory bank **120** in a MAP system computer architecture **100** of the present invention is shown for a representative one of the MAPs **112** illustrated in the preceding figure. Each memory bank **120** includes a bank control logic block **122** bi-directionally coupled to the computer system trunk lines, for example, a 72 line bus **124**. The bank control logic block **122** is coupled to a bi-directional data bus **126** (for example 256 lines) and supplies addresses on an address bus **128** (for example 17 lines) for accessing data at specified locations within a memory array **130**.

The data bus **126** and address bus **128** are also coupled to a MAP assembly **112**. The MAP assembly **112** comprises a control block **132** coupled to the address bus **128**. The control block **132** is also bi-directionally coupled to a user field programmable gate array ("FPGA") **134** by means of a number of signal lines **136**. The user FPGA **134** is coupled directly to the data bus **126**. In a particular embodiment, the FPGA **134** may be provided as a Lucent Technologies OR3T80 device.

The computer architecture **100** comprises a multiprocessor system employing uniform memory access across common shared memory with one or more MAPs **112** located in the memory subsystem, or memory space. As previously described, each MAP **112** contains at least one relatively large FPGA **134** that is used as a reconfigurable functional unit. In addition, a control block **132** and a preprogrammed or dynamically programmable configuration read-only memory ("ROM" as will be more fully described hereinafter) contains the information needed by the reconfigurable MAP assembly **112** to enable it to perform a specific algorithm. It is also possible for the user to directly download a new configuration into the FPGA **134** under program control, although in some instances this may consume a number of memory accesses and might result in an overall decrease in system performance if the algorithm was short-lived.

FPGAs have particular advantages in the application shown for several reasons. First, commercially available, off-the-shelf FPGAs now contain sufficient internal logic cells to perform meaningful computational functions. Secondly, they can operate at speeds comparable to microprocessors, which eliminates the need for speed matching buffers. Still further, the internal programmable routing resources of FPGAs are now extensive enough that meaningful algorithms can now be programmed without the need to reassign the locations of the input/output ("I/O") pins.

By placing the MAP **112** in the memory subsystem or memory space, it can be readily accessed through the use of memory read and write commands, which allows the use of a variety of standard operating systems. In contrast, other conventional implementations propose placement of any reconfigurable logic in or near the processor. This is much less effective in a multiprocessor environment because only one processor has rapid access to it. Consequently, reconfigurable logic must be placed by every processor in a multiprocessor system, which increases the overall system cost. In addition, MAP **112** can access the memory array **130** itself, referred to as Direct Memory Access ("DMA"), allowing it to execute tasks independently and asynchronously of the processor. In comparison, were it were placed near the processor, it would have to compete with the processors for system routing resources in order to access memory, which deleteriously impacts processor performance. Because MAP **112** has DMA capability, (allowing it to write to memory), and because it receives its operands via writes to memory, it is possible to allow a MAP **112** to feed results to another MAP **112**. This is a very powerful feature that allows for very extensive pipelining and parallelizing of large tasks, which permits them to complete faster.

Many of the algorithms that may be implemented will receive an operand and require many clock cycles to produce a result. One such example may be a multiplication that takes 64 clock cycles. This same multiplication may also need to be performed on thousands of operands. In this situation, the incoming operands would be presented sequentially so that while the first operand requires 64 clock cycles to produce results at the output, the second operand, arriving one clock cycle later at the input, will show results one clock cycle later at the output. Thus, after an initial delay of 64 clock cycles, new output data will appear on every consecutive clock cycle until the results of the last operand appears. This is called "pipelining".

In a multiprocessor system, it is quite common for the operating system to stop a processor in the middle of a task, reassign it to a higher priority task, and then return it, or

another, to complete the initial task. When this is combined with a pipelined algorithm, a problem arises (if the processor stops issuing operands in the middle of a list and stops accepting results) with respect to operands already issued but not yet through the pipeline. To handle this issue, a solution involving the combination of software and hardware is disclosed herein.

To make use of any type of conventional reconfigurable hardware, the programmer could embed the necessary commands in his application program code. The drawback to this approach is that a program would then have to be tailored to be specific to the MAP hardware. The system of the present invention eliminates this problem. Multiprocessor computers often use software called parallelizers. The purpose of this software is to analyze the user's application code and determine how best to split it up among the processors. The present invention provides significant advantages over a conventional parallelizer and enables it to recognize portions of the user code that represent algorithms that exist in MAPs **112** for that system and to then treat the MAP **112** as another computing element. The parallelizer then automatically generates the necessary code to utilize the MAP **112**. This allows the user to write the algorithm directly in his code, allowing it to be more portable and reducing the knowledge of the system hardware that he has to have to utilize the MAP **112**.

With reference additionally now to FIG. **4**, a block diagram of the MAP control block **132** is shown in greater detail. The control block **132** is coupled to receive a number of command bits (for example, **17**) from the address bus **128** at a command decoder **150**. The command decoder **150** then supplies a number of register control bits to a group of status registers **152** on an eight bit bus **154**. The command decoder **150** also supplies a single bit last operand flag on line **156** to a pipeline counter **158**. The pipeline counter **158** supplies an eight bit output to an equality comparitor **160** on bus **162**. The equality comparitor **160** also receives an eight bit signal from the FPGA **134** on bus **136** indicative of the pipeline depth. When the equality comparitor determines that the pipeline is empty, it provides a single bit pipeline empty flag on line **164** for input to the status registers **152**. The status registers are also coupled to receive an eight bit status signal from the FPGA **134** on bus **136** and it produces a sixty four bit status word output on bus **166** in response to the signals on bus **136**, **154** and line **164**.

The command decoder **150** also supplies a five bit control signal to a configuration multiplexer ("MUX") **170** as shown. The configuration mux **170** receives a single bit output of a 256 bit parallel-serial converter **172** on line **176**. The inputs of the 256 bit parallel-to-serial converter **172** are coupled to a 256 bit user configuration pattern bus **174**. The configuration mux **170** also receives sixteen single bit inputs from the configuration ROMs (illustrated as ROM **182**) on bus **178** and provides a single bit configuration file signal on line **180** to the user FPGA **134** as selected by the control signals from the command decoder **150** on the bus **168**.

In operation, when a processor **108** is halted by the operating system, the operating system will issue a last operand command to the MAP **112** through the use of command bits embedded in the address field on bus **128**. This command is recognized by the command decoder **150** of the control block **132** and it initiates a hardware pipeline counter **158**. When the algorithm was initially loaded into the FPGA **134**, several output bits connected to the control block **132** were configured to display a binary representation of the number of clock cycles required to get through its pipeline (i.e. pipeline "depth")on bus **136** input to the equality comparitor **160**. After receiving the last operand command, the pipeline counter **158** in the control block **132** counts clock cycles until its count equals the pipeline depth for that particular algorithm. At that point, the equality comparitor **160** in the control block **132** de-asserts a busy bit on line **164** in an internal group of status registers **152**. After issuing the last operand signal, the processor **108** will repeatedly read the status registers **152** and accept any output data on bus **166**. When the busy flag is de-asserted, the task can be stopped and the MAP **112** utilized for a different task. It should be noted that it is also possible to leave the MAP **112** configured, transfer the program to a different processor **108** and restart the task where it left off

In order to evaluate the effectiveness of the use of the MAP **112** in a given application, some form of feedback to the use is required. Therefore, the MAP **112** may be equipped with internal registers in the control block **132** that allow it to monitor efficiency related factors such as the number of input operands versus output data, the number of idle cycles over time and the number of system monitor interrupts received over time. One of the advantages that the MAP **112** has is that because of its reconfigurable nature, the actual function and type of function that are monitored can also change as the algorithm changes. This provides the user with an almost infinite number of possible monitored factors without having to monitor all factors all of the time.

While there have been described above the principles of the present invention in conjunction with a specific multiprocessor architecture it is to be clearly understood that the foregoing description is made only by way of example and not as a limitation to the scope of the invention. Particularly, it is recognized that the teachings of the foregoing disclosure will suggest other modifications to those persons skilled in the relevant art. Such modifications may involve other features which are already known per se and which may be used instead of or in addition to features already described herein. Although claims have been formulated in this application to particular combinations of features, it should be understood that the scope of the disclosure herein also includes any novel feature or any novel combination of features disclosed either explicitly or implicitly or any generalization or modification thereof which would be apparent to persons skilled in the relevant art, whether or not such relates to the same invention as presently claimed in any claim and whether or not it mitigates any or all of the same technical problems as confronted by the present invention. The applicants hereby reserve the right to formulate new claims to such features and/or combinations of such features during the prosecution of the present application or of any further application derived therefrom.

What is claimed is:

1. In a computer system having at least one data processor for executing an application program by operating on user data in accordance with application program instructions, said computer system having at least one memory bank with a data bus and an address bus connected to said at least one data processor, the improvement comprising:

a plurality of memory algorithm processors within individually addressable portions of said memory bank;

means connecting said plurality of memory algorithm processors to said data bus and to said address bus such that said plurality of memory algorithm processors are individually memory addressable by said at least one data processor as said at least one data processor executes said application program; and

said plurality of memory algorithm processors being configured as individual data processing machines that

can be memory addressed to perform data processing related to said application program in accordance with an identified algorithm, said data processing being performed on at least one operand that is received as a result of a write operation to said memory bank by said at least one data processor.

2. The improvement of claim **1** wherein each of said plurality of memory algorithm processors comprises a field programmable gate array.

3. The improvement of claim **1** wherein each of said plurality of memory algorithm processors is operative to memory address said memory bank independent of said at least one data processor.

4. The improvement of claim **1** wherein an identified algorithm is preprogrammed into each of said plurality of memory algorithm processors.

5. The improvement of claim **4** wherein a plurality of identified algorithms are preprogrammed into a memory device that is associated with said plurality of memory algorithm processors.

6. The improvement of claim **5** wherein said memory device comprises at least one read only memory device.

7. The improvement of claim **1** wherein any given one of said plurality of memory algorithm processors is operative to pass a data processing result of an operand that has been processed by an identified algorithm to another of said plurality of memory algorithm processors.

8. The improvement of claim **1** wherein said plurality of memory algorithm processors comprise a memory algorithm processor assembly, said memory algorithm processor assembly including:

a control block having a command decoder coupled to said address bus and having a pipeline counter coupled to said command decoder;

said command decoder for providing a last operand flag to said pipeline counter in response to a last operand command from an operating system of said at least one data processor.

9. The improvement of claim **8** wherein said control block further includes:

at least one status register; and

an equality comparator coupled to receive a pipeline depth signal and an output of said pipeline counter, said equality comparator for providing a pipeline empty flag to said at least one status register.

10. The improvement of claim **9** wherein said at least one status register is coupled to said command decoder to receive a register control signal and is coupled to said plurality of memory algorithm processors to receive a status signal, said at least one status register providing a status word output signal.

11. A multiprocessor computer system comprising:

a plurality of data processors for executing at least one application program by operating on user data in accordance with program instructions;

a memory bank having a data bus and an address bus connected to said plurality of data processors;

a plurality of memory algorithm processors within said memory bank at a plurality of individual memory addressable memory locations;

means coupling said plurality of individual memory algorithm processors to said data bus and to said address bus;

said plurality of individual memory algorithm processors being individually memory addressable by all of said plurality of data processors; and

said plurality of memory algorithm processors being individually configurable to perform an identified algorithm on an operand that is received from a write operation by said at least one data processor to said memory bank as said at least one data processor executes said at least one application program.

12. The multiprocessor computer system of claim **11** wherein all of said plurality of memory algorithm processors are memory addressable by all of said plurality of data processors.

13. The multiprocessor computer system of claim **12** wherein all of said plurality of memory algorithm processors are mutually memory addressable.

14. The multiprocessor computer system of claim **13** wherein said plurality of memory algorithm processors collectively comprises a memory algorithm processor assembly, said memory algorithm processor assembly including:

a control block operative to provide a last operand flag in response to a last operand having been processed by said memory algorithm processor assembly.

15. The multiprocessor computer system of claim **11** including:

at least one memory device associated with said plurality of memory algorithm processors for storing a plurality of pre loaded identified algorithms.

16. The multiprocessor computer system of claim **15** wherein said at least one memory device is responsive to a predetermined command from a data processor and operates in response thereto to selected one of said plurality of pre-loaded identified algorithms to be implemented by an addressed one of said plurality of memory algorithm processors.

17. The multiprocessor computer system of claim **16** wherein said at least one memory device comprises at least one read only memory device.

18. The multiprocessor computer system of claim **11** wherein each of said plurality of memory algorithm processors comprises a field programmable gate array.

19. The multiprocessor computer system of claim **11** wherein each of said plurality of memory algorithm processors is memory accessible through normal memory access protocol.

20. The multiprocessor computer system of claim **11** wherein each of said plurality of memory algorithm processors has direct memory access capability to said memory bank.

21. The multiprocessor computer system of claim **11** wherein each of said plurality of memory algorithm processors is operative to pass a result of a processed operand to another memory algorithm processor.

22. The multiprocessor computer system of claim **11** operative to detect at least one parallel region of said at least one application program, wherein at least one of said plurality of memory algorithm processors is configured as a function of said detected at least one parallel region of said at least one application program.

* * * * *

EXHIBIT C

US009153311B1

(12) **United States Patent**
Tewalt

(10) **Patent No.: US 9,153,311 B1**
(45) **Date of Patent: Oct. 6, 2015**

(54) **SYSTEM AND METHOD FOR RETAINING DRAM DATA WHEN REPROGRAMMING RECONFIGURABLE DEVICES WITH DRAM MEMORY CONTROLLERS**

(71) Applicant: **SRC Computers, LLC.**, Colorado Springs, CO (US)

(72) Inventor: **Timothy J. Tewalt**, Larkspur, CO (US)

(73) Assignee: **SRC Computers, LLC**, Colorado Springs, CO (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **14/288,094**

(22) Filed: **May 27, 2014**

(51) **Int. Cl.**
G11C 7/00 (2006.01)
G11C 11/406 (2006.01)

(52) **U.S. Cl.**
CPC ***G11C 11/40615*** (2013.01)

(58) **Field of Classification Search**
USPC 365/222
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,026,459	A	2/2000	Huppenthal
6,076,152	A	6/2000	Huppenthal et al.
6,247,110	B1	6/2001	Huppenthal et al.
6,295,598	B1	9/2001	Bertoni et al.
6,339,819	B1	1/2002	Huppenthal et al.
6,356,983	B1	3/2002	Parks
6,434,687	B1	8/2002	Huppenthal
6,594,736	B1	7/2003	Parks
6,836,823	B2	12/2004	Burton
6,941,539	B2	9/2005	Hammes
6,961,841	B2	11/2005	Huppenthal et al.
6,964,029	B2	11/2005	Poznanovic et al.
6,983,456	B2	1/2006	Poznanovic et al.
6,996,656	B2	2/2006	Burton
7,003,593	B2	2/2006	Huppenthal et al.
7,124,211	B2	10/2006	Dickson et al.
7,134,120	B2	11/2006	Hammes
7,149,867	B2	12/2006	Poznanovic et al.
7,155,602	B2	12/2006	Poznanovic
7,155,708	B2	12/2006	Hammes et al.
7,167,976	B2	1/2007	Poznanovic
7,197,575	B2	3/2007	Huppenthal et al.
7,225,324	B2	5/2007	Huppenthal et al.
7,237,091	B2	6/2007	Huppenthal et al.
7,299,458	B2	11/2007	Hammes

(Continued)

OTHER PUBLICATIONS

Allan, Graham, "DDR IP Integration: How to Avoid Landmines in this Quickly Changing Landscape", Chip Design, Jun./Jul. 2007, pp. 20-22.

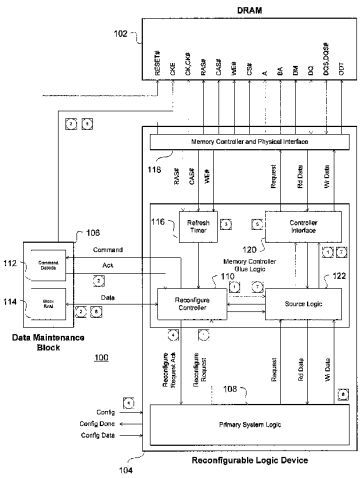
(Continued)

Primary Examiner — Hoai V Ho
(74) *Attorney, Agent, or Firm* — Peter J. Meza; William J. Kubida; Hogan Lovells US LLP

(57) **ABSTRACT**

A system and method for retaining dynamic random access memory (DRAM) data when reprogramming reconfigurable devices with DRAM memory controllers such as field programmable gate arrays (FPGAs). The DRAM memory controller is utilized in concert with an internally or externally located data maintenance block wherein the FPGA drives the majority of the DRAM input/output (I/O) and the data maintenance block drives the self-refresh command inputs. Even though the FPGA reconfigures and the majority of the DRAM inputs are tri-stated, the data maintenance block provides stable input levels on the self-refresh command inputs.

19 Claims, 2 Drawing Sheets



(56) **References Cited**

U.S. PATENT DOCUMENTS

7,373,440	B2	5/2008	Huppenthal et al.	
7,406,573	B2	7/2008	Huppenthal et al.	
7,421,524	B2	9/2008	Huppenthal et al.	
7,424,552	B2	9/2008	Burton	
7,565,461	B2	7/2009	Huppenthal et al.	
7,620,800	B2	11/2009	Huppenthal et al.	
7,680,968	B2	3/2010	Burton	
7,703,085	B2	4/2010	Poznanovic et al.	
7,890,686	B2	2/2011	Conner	
8,589,666	B2	11/2013	Hammes	
8,713,518	B2	4/2014	Pointer et al.	
2012/0117318	A1	5/2012	Burton et al.	
2013/0157639	A1	6/2013	Huppenthal et al.	
2014/0211579	A1*	7/2014	Lovelace	365/200

OTHER PUBLICATIONS

Wilson, Ron, "DRAM Controllers for System Designers", Altera Corporation Articles, 2012, 8 pages.

* cited by examiner

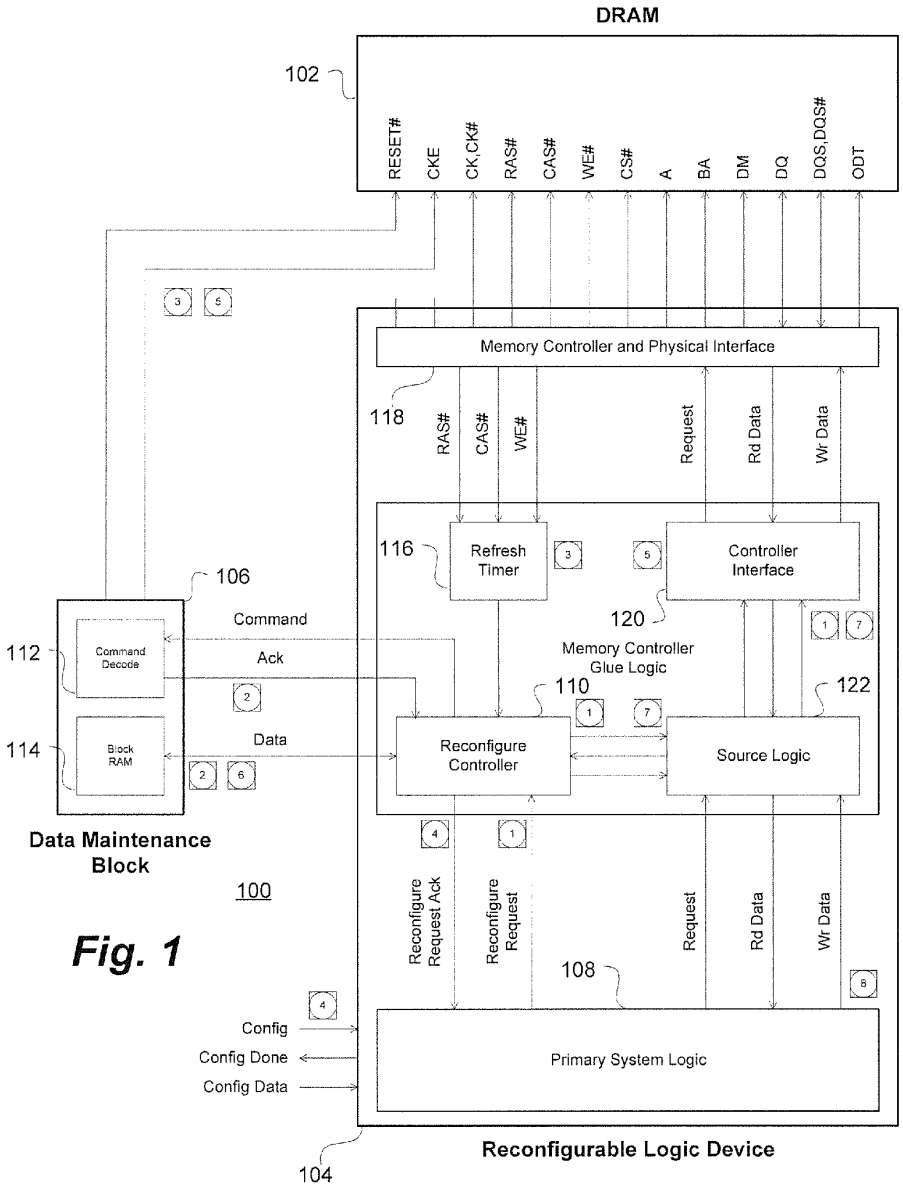

DRAM
102
RESET#
CKE
CK,CK#
RAS#
CAS#
WE#
CS#
A
BA
DM
DQ
DQS,DQS#
ODT
Memory Controller and Physical Interface
118
RAS#
CAS#
WE#
Request
Rd Data
Wr Data
116
Refresh Timer
Controller Interface
120
Memory Controller Glue Logic
110
122
Reconfigure Controller
Source Logic
106
Command
Ack
Data
112
Command Decode
114
Block RAM
Data Maintenance Block
100
Reconfigure Request Ack
Reconfigure Request
Request
Rd Data
Wr Data
108
Config
Config Done
Config Data
Primary System Logic
Reconfigurable Logic Device
104

Fig. 1

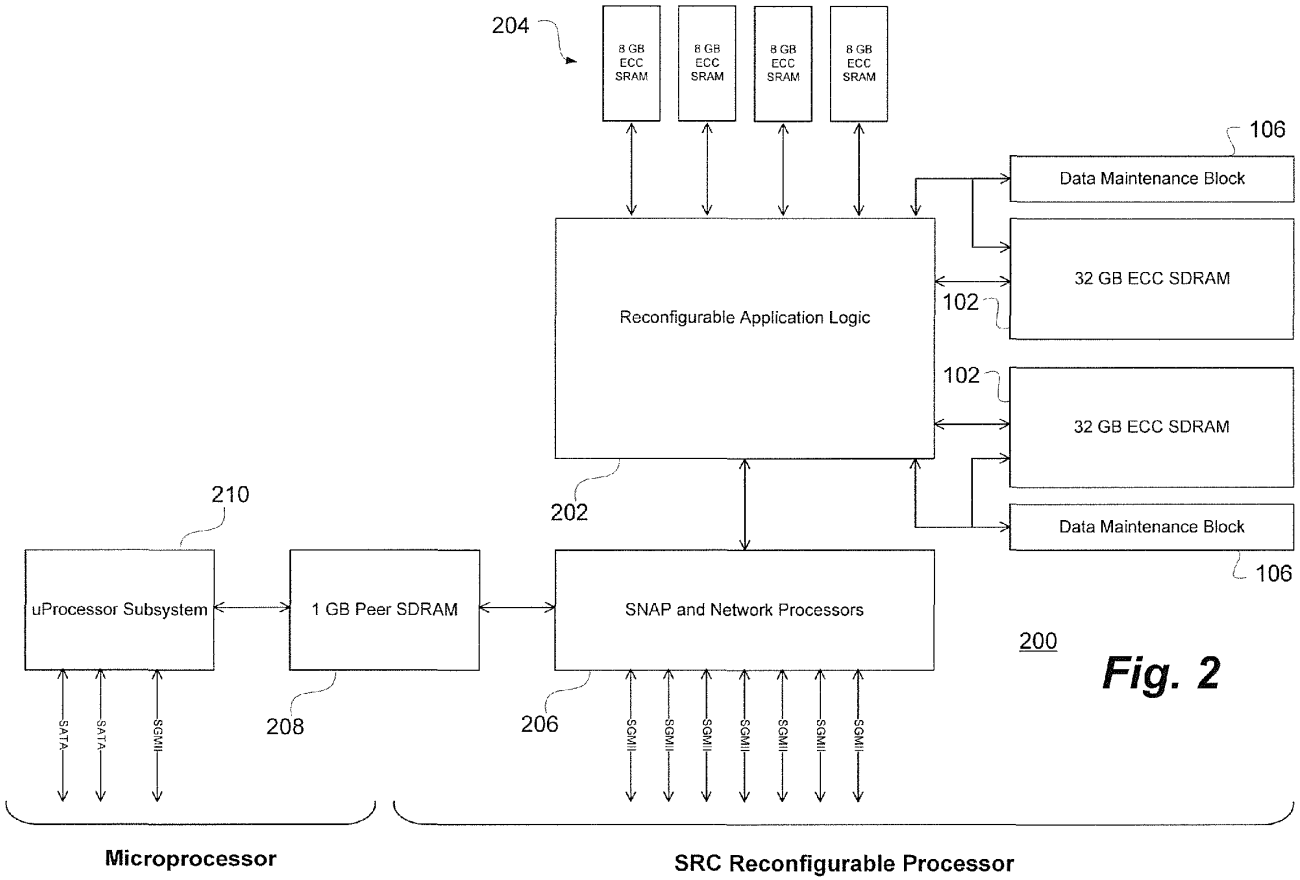

204
8 GB ECC SRAM
8 GB ECC SRAM
8 GB ECC SRAM
8 GB ECC SRAM
106
Data Maintenance Block
32 GB ECC SDRAM
102
Reconfigurable Application Logic
102
32 GB ECC SDRAM
202
210
Data Maintenance Block
106
uProcessor Subsystem
1 GB Peer SDRAM
SNAP and Network Processors
200
208
206
SATA
SATA
SGMII
SGMII
SGMII
SGMII
SGMII
SGMII
SGMII
Microprocessor
SRC Reconfigurable Processor

Fig. 2

SYSTEM AND METHOD FOR RETAINING DRAM DATA WHEN REPROGRAMMING RECONFIGURABLE DEVICES WITH DRAM MEMORY CONTROLLERS

BACKGROUND OF THE INVENTION

The present invention relates, in general, to the field of reconfigurable computing systems. More particularly, the present invention relates to a system and method for retaining dynamic random access memory (DRAM) data when reprogramming reconfigurable devices with DRAM memory controllers.

The majority of today's programmable logic designs include a DRAM based memory solution at the heart of their memory subsystem. Today's DRAM devices are significantly faster than previous generation's, albeit at the cost of requiring increasingly complex and resource intensive memory controllers. One example is in double data rate 3 and 4 (DDR3 and DDR4) controllers which require read and write calibration logic. This added logic was not necessary when using previous versions of DRAM (e.g. DDR and DDR2. As a result, companies are forced to absorb substantial design costs and increased project completion times when designing proprietary DRAM controllers utilizing modern DRAM technology.

In order to mitigate design engineering costs and verification time, it is very common for field programmable gate array (FPGA) designers to implement vendor provided memory controller intellectual property (IP) when including DRAM based memory solutions in their designs. See, for example, Allan, Graham; "DDR IP Integration: How to Avoid Landmines in this Quickly Changing Landscape"; Chip Design, June/July 2007; pp 20-22 and Wilson, Ron; "DRAM Controllers for System Designers"; Altera Corporation Articles, 2012.

FPGA designers tend to choose device manufacturer IP designs because they are proven, tested and have the incredible benefit of significantly reduced design costs and project completion times. Many times there is the added benefit of exploiting specialized circuitry within the programmable device to increase controller performance, which is not always readily apparent when designing a controller from scratch.

The downside to using factory supplied IP memory controllers is that there is little flexibility when trying to modify operating characteristics. A significant problem arises in reconfigurable computing when the FPGA is reprogrammed during a live application and the memory controller tri-states all inputs and outputs (I/O) between the FPGA device and the DRAM. The result is corrupted data in the memory subsystem. Therefore, dynamically reconfigurable processors are excluded as viable computing options, especially in regard to database applications or context switch processing. The reason for this is that the time it takes to copy the entire contents of DRAM data and preserve it in another part of the system, reconfigure the processor, then finally retrieve the data and restore it in DRAM is just too excessive.

SUMMARY OF THE INVENTION

Disclosed herein is a system and method for preserving DRAM memory contents when a reconfigurable device, for example an FPGA having a DRAM memory controller, is reconfigured, reprogrammed or otherwise powered down. When an FPGA is reprogrammed, the DRAM inputs are tri-stated including self-refresh command signals. Indeterminate states on the reset or clock enable inputs results in DRAM data corruption.

In accordance with the system and method of the present invention, an FPGA based DRAM controller is utilized in concert with an internally or externally located data maintenance block. In operation, the FPGA drives the majority of the DRAM input/output (I/O) and the data maintenance block drives the self-refresh command inputs. Even though the FPGA reconfigures and the majority of the DRAM inputs are tri-stated, the data maintenance block provides stable input levels on the self-refresh command inputs.

Functionally, the data maintenance block does not contain the memory controller and therefore has no point of reference for when and how to initiate the self-refresh commands, particularly the DRAM self-refresh mode. As also disclosed herein, a communication port is implemented between the FPGA and the data maintenance block that allows the memory controller in the FPGA to direct the self-refresh commands to the DRAM via the data maintenance block. Specifically, this entails when to put the DRAM into self-refresh mode and preserve the data in memory.

At this point, the DRAM data has been preserved throughout the FPGA reconfiguration via the self-refresh mode initiated by the data maintenance block, but the DRAM controller must now re-establish write/read timing windows and will corrupt specific address contents with guaranteed write and read data required during the calibration/leveling process. Consequently, using the self-refresh capability of DRAM alone is not adequate for maintaining data integrity during reconfiguration. (It should be noted that the memory addresses used during calibration/leveling are known and typically detailed in the controller IP specification).

In order to effectuate this, the system transmits a "reconfiguration request" to the DRAM controller. Once received, glue logic surrounding the FPGA vendor provided memory controller IP issues read requests to the controller specifying address locations used during the calibration/leveling process. As data is retrieved from the DRAM, it is transmitted via the communication port from the FPGA device to a block of storage space residing within the data maintenance block itself or another location in the system.

Once the process is complete, the data maintenance block sends a self-refresh command to the DRAM and transmits an acknowledge signal back to the FPGA. The data maintenance block recognizes this as an FPGA reconfiguration condition versus an FPGA initial power up condition and retains this state for later use.

Once the FPGA has been reprogrammed, the DRAM controller has re-established calibration settings and several specific addresses in the DRAM have been corrupted with guaranteed write/read data patterns. At this point, glue logic surrounding the vendor memory controller IP is advised by the data maintenance block (through the communication port) that it has awakened from either an initial power up condition or a reconfiguration condition. If a reconfiguration condition is detected, and before processing incoming DMA requests, the controller retrieves stored DRAM data from the data maintenance block (again through the communication port) and writes it back to the specific address locations corrupted during the calibration/leveling process. Once complete, the DRAM controller in the FPGA is free to begin servicing system memory requests in the traditional fashion.

Among the benefits provided in conjunction with the system and method of the present invention is that since the data maintenance block functions to hold the DRAM in self-refresh mode, the FPGA is free to be reprogrammed to perform

a very application-specific computing job that may not require DRAM. This means all the device resources previously reserved for creating a DRAM controller are now free to be used for different functions.

Further, the overall computer system benefits from the present invention because data previously stored in DRAM has now been preserved and is available for use by the next application that needs it. This leads to the fact that computing solutions requiring a series of specific data manipulation tasks now have the ability to be implemented in a small reconfigurable processor. Each application performs its intended function and data is passed from application to application between reconfiguration periods via the DRAM.

Importantly, it should also be noted that the DRAM data contents are retained even if the reconfigurable device is powered down. This is especially critical, for example, when the system and method of the present invention is implemented in mobile devices.

Particularly disclosed herein is a system and method for preserving DRAM data contents when reconfiguring a device containing one or more DRAM controllers. Also particularly disclosed herein is a system and method for preserving DRAM data contents in a reconfigurable computing environment when the programmable device is reconfigured with a new design that does not include a DRAM controller. Further disclosed herein is a system and method for passing DRAM data between sequential computing tasks in a reconfigurable computing environment as well as system and method for preserving DRAM contents when the reconfigurable device is powered down.

Also particularly disclosed herein is a computer system which comprises a DRAM memory, a reconfigurable logic device having a memory controller coupled to selected inputs and outputs of the DRAM memory and a data maintenance block coupled to the reconfigurable logic device and self-refresh command inputs of the DRAM memory. The data maintenance block is operative to provide stable input levels on the self-refresh command inputs while the reconfigurable logic device is reconfigured.

Still further particularly disclosed herein is a method for preserving the contents of a DRAM memory associated with a reconfigurable device having a memory controller. The method comprises providing a data maintenance block coupled to the reconfigurable device, coupling the data maintenance block to self-refresh command inputs of the DRAM memory, storing data received from the reconfigurable device at the data maintenance block and maintaining stable input levels on the self-refresh command inputs while the reconfigurable logic device is reconfigured.

BRIEF DESCRIPTION OF THE DRAWINGS

The aforementioned and other features and objects of the present invention and the manner of attaining them will become more apparent and the invention itself will be best understood by reference to the following description of a preferred embodiment taken in conjunction with the accompanying drawings, wherein:

FIG. **1** is a functional block diagram of a computer subsystem comprising a reconfigurable logic device having a reconfigurable DRAM controller with associated DRAM memory and illustrating the data maintenance block of the present invention for retaining DRAM data when the logic device is reconfigured; and

FIG. **2** is a block diagram of a reconfigurable computer system, such as that available from SRC Computers, LLC, assignee of the present invention, incorporating a pair of data maintenance blocks and DRAM memory in accordance with the system and method of the present invention in association with reconfigurable application logic.

DESCRIPTION OF A REPRESENTATIVE EMBODIMENT

With reference now to FIG. **1**, a functional block diagram of a computer subsystem **100** comprising a DRAM memory **102** and reconfigurable logic device **104** is shown. In a representative embodiment of the present invention, the reconfigurable logic device **104** may comprise a field programmable gate array (FPGA). However, it should be noted that the reconfigurable logic device **104** may comprise any and all forms of reconfigurable logic devices including hybrid devices, such as a reconfigurable logic device with partial reconfiguration capabilities or an application specific integrated circuit (ASIC) device with reprogrammable regions contained within the chip.

Also illustrated is a data maintenance block **106** in accordance with the present invention for retaining DRAM memory **102** data when the logic device **104** is reconfigured during operation of the computer subsystem **100**. In a representative embodiment of the present invention, the data maintenance block **106** may be conveniently provided as a complex programmable logic device (CPLD) or other separate integrated circuit device or, in alternative embodiments, may be provided as a portion of an FPGA comprising the reconfigurable logic device **104**.

As illustrated, the reconfigurable logic device **104** comprises a primary system logic block **108** which issues a reconfigure request command to a reconfigure controller **110** and receives a reconfigure request acknowledgement (Ack) signal in return. The reconfigure controller **110**, in turn, issues a command to the command decode block **112** of the data maintenance block **106** and receives an acknowledgement (Ack) signal in return. A block RAM portion **114** of the data maintenance block **106** exchanges data with the reconfigure controller **110**.

The reconfigure controller **110** receives an input from a refresh timer **116** which is coupled to receive row address select (RAS#), column address select (CAS#) and write enable (WE#) signals from a memory controller and physical interface block **118**. The memory controller and physical interface block **118** also provides the RAS#, CAS# and WE# signals to the DRAM memory **102** as well as clock (CR, CK#), chip select (CS#), address (A), bank address (BA), data mask (DM) and on-die termination (ODT) input signals. Bididirectional data (DQ) input/output (I/O) and differential data strobe signals (DQS/DQS#) are exchanged between the DRAM memory **102** and the memory controller and physical interface block **118** as shown. The data maintenance block **106** is coupled to the DRAM memory **102** to supply reset (RESET#) and clock enable (CKE#) signals thereto.

The memory controller and physical interface block **118** responds to a request from the controller interface **120** to provide data read from the DRAM memory **102** (Rd Data) and to receive data to be written to the DRAM memory **102** (Wr Data) as shown. A source logic block **122** is coupled to the controller interface **120** as well as the reconfigure controller **110** as also illustrated. The source logic block **122** receives a data request from the primary system logic block **108** and supplies data read from the DRAM memory **102** while receiving data to be written thereto.

As indicated by the operation at numeral **1**, a reconfiguration request is received at the reconfigure controller **110** from the primary system logic block **108** of the reconfigurable

logic device **104**. The reconfigure controller **110** initiates direct memory access (DMA) read requests to memory addresses used in a calibration/leveling sequence after the reconfigurable logic device **104** is reconfigured. Returned data is stored in a small section of block RAM (not shown) in the reconfigure controller **110**.

As indicated by the operation at numeral **2**, the reconfigure Controller **110** stores its block RAM contents in another small section of block RAM **114** located in the data maintenance block **106**. When complete, the data maintenance block **106** asserts an acknowledge signal from its command decode block **112**. At the operation indicated by numeral **3**, the reconfigure controller **110** detects a refresh command from the refresh timer **116**, waits a refresh cycle time (t_{RFC}) and instructs the data maintenance block **106** to de-assert CKE to the DRAM memory **102**.

The reconfigure controller **110** asserts the Reconfigure Request Ack signal at the operation indicated by numeral **4** and the reconfigurable logic device **104** is reconfigured. As indicated by the operation at numeral **5**, the reconfigure controller **110** recognizes a post-reconfigure condition (Ack=High), holds the memory controller and physical interface **118** in reset and instructs the data maintenance block **106** to assert CKE to the DRAM memory **102**. The memory controller and physical interface **118** is then released from reset and initializes the DRAM memory **102**.

At the operation indicated by numeral **6**, the reconfigure controller **110** retrieves the data maintenance block **106** block RAM **114** contents and stores it in a small section of block RAM (not shown) in the reconfigure controller **110**. The reconfigure controller **110** detects that the memory controller and physical interface **118** and DRAM memory **102** initialization is complete at the operation indicated by numeral **7** and initiates DMA write requests to restore the memory contents corrupted during the calibration/leveling sequence with the data values read prior to reconfiguration. At the operation indicated by numeral **8**, the memory controller and physical interface **118** glue logic (comprising reconfigure controller **110**, refresh timer **116**, controller interface **120** and source logic block **122**) resumes DMA activity with the primary system logic **108** in a conventional fashion.

It should be noted certain of the aforementioned operational steps may, in fact, operate substantially concurrently. Further, while functionally accurate, some of the operational steps enumerated have been listed out of order to provide logical continuity to the overall operation and to facilitate comprehensibility of the process. In a particular implementation of the system and method of the present invention, one or more of the operational steps disclosed may be conveniently re-ordered to increase overall hardware efficiency. Moreover, steps which can serve to facilitate relatively seamless integration in an active application can be provided in addition to those described as may be desired.

With reference additionally now to FIG. **2**, a block diagram of a reconfigurable computer system **200** is illustrated incorporating a pair of data maintenance blocks **106** and DRAM memory **102** in accordance with the system and method of the present invention in association with reconfigurable application logic **202**. In this representative embodiment of a reconfigurable computer system **200**, the DRAM memory **102** is illustrated in the form of 32 GB error correction code (ECC) synchronous dynamic random access memory (SDRAM).

The reconfigurable application logic **202** is coupled to the data maintenance blocks **106** and DRAM memory **102** as depicted and described previously with respect to the preceding figure and is also illustrated as being coupled to a number of 8 GB ECC static random access memory (SRAM) memory modules **204**. The reconfigurable application logic **202** is also coupled to an SRC Computers, LLC SNAP™ and network processors block **206** having a number of serial gigabit media independent interface (SGMII) links as shown. It should be noted that the DRAM memory **102** controller in the reconfigurable application block **202** may be omitted upon subsequent reconfigurations as the DRAM memory **102** data contents will be maintained in the data maintenance blocks **106**.

The SNAP and network processors block **206** shares equal read/write access to a 1 GB peer SDRAM system memory **208** along with a microprocessor subsystem **210**. The microprocessor subsystem **210**, as illustrated, also comprises an SGMII link as well as a pair of serial advanced technology attachment (SATA) interfaces.

For continuity and clarity of the description herein, the term "FPGA" has been used in conjunction with the representative embodiment of the system and method of the present invention and refers to just one type of reconfigurable logic device. However, it should be noted that the concept disclosed herein is applicable to any and all forms of reconfigurable logic devices including hybrid devices, inclusive of reconfigurable logic devices with partial reconfiguration capabilities or an ASIC device with reprogrammable regions contained within the chip.

Representative embodiments of dynamically reconfigurable computing systems incorporating the DRAM memory **102**, reconfigurable logic device **104**, associated microprocessors and programming techniques are disclosed in one or more of the following United States Patents and United States Patent Publications to SRC Computers LLC, assignee of the present invention, the disclosures of which are herein specifically incorporated by this reference in their entirety: U.S. Pat. No. 6,026,459; U.S. Pat. No. 6,076,152; U.S. Pat. No. 6,247,110; U.S. Pat. No. 6,295,598; U.S. Pat. No. 6,339,819; U.S. Pat. No. 6,356,983; U.S. Pat. No. 6,434,687; U.S. Pat. No. 6,594,736; U.S. Pat. No. 6,836,823; U.S. Pat. No. 6,941,539; U.S. Pat. No. 6,961,841; U.S. Pat. No. 6,964,029; U.S. Pat. No. 6,983,456; U.S. Pat. No. 6,996,656; U.S. Pat. No. 7,003,593; U.S. Pat. No. 7,124,211; U.S. Pat. No. 7,134,120; U.S. Pat. No. 7,149,867; U.S. Pat. No. 7,155,602; U.S. Pat. No. 7,155,708; U.S. Pat. No. 7,167,976; U.S. Pat. No. 7,197,575; U.S. Pat. No. 7,225,324; U.S. Pat. No. 7,237,091; U.S. Pat. No. 7,299,458; U.S. Pat. No. 7,373,440; U.S. Pat. No. 7,406,573; U.S. Pat. No. 7,421,524; U.S. Pat. No. 7,424,552; U.S. Pat. No. 7,565,461; U.S. Pat. No. 7,620,800; U.S. Pat. No. 7,680,968; U.S. Pat. No. 7,703,085; U.S. Pat. No. 7,890,686; U.S. Pat. No. 8,589,666; U.S. Pat. Pub. No. 2012/0117318; U.S. Pat. Pub. No. 2012/0117535; and U.S. Pat. Pub. No. 2013/0157639.

While there have been described above the principles of the present invention in conjunction with specific apparatus and methods, it is to be clearly understood that the foregoing description is made only by way of example and not as a limitation to the scope of the invention. Particularly, it is recognized that the teachings of the foregoing disclosure will suggest other modifications to those persons skilled in the relevant art. Such modifications may involve other features which are already known per se and which may be used instead of or in addition to features already described herein. Although claims have been formulated in this application to particular combinations of features, it should be understood that the scope of the disclosure herein also includes any novel feature or any novel combination of features disclosed either explicitly or implicitly or any generalization or modification thereof which would be apparent to persons skilled in the relevant art, whether or not such relates to the same invention as presently claimed in any claim and whether or not it miti-

gates any or all of the same technical problems as confronted by the present invention. The applicants hereby reserve the right to formulate new claims to such features and/or combinations of such features during the prosecution of the present application or of any further application derived therefrom.

As used herein, the terms "comprises", "comprising", or any other variation thereof, are intended to cover a non-exclusive inclusion, such that a process, method, article, or apparatus that comprises a recitation of certain elements does not necessarily include only those elements but may include other elements not expressly recited or inherent to such process, method, article or apparatus. None of the description in the present application should be read as implying that any particular element, step, or function is an essential element which must be included in the claim scope and THE SCOPE OF THE PATENTED SUBJECT MATTER IS DEFINED ONLY BY THE CLAIMS AS ALLOWED. Moreover, none of the appended claims are intended to invoke paragraph six of 35 U.S.C. Sect. 112 unless the exact phrase "means for" is employed and is followed by a participle.

What is claimed is:

1. A computer system comprising:

a DRAM memory;

a reconfigurable logic device having a memory controller coupled to selected inputs and outputs of said DRAM memory; and

a data maintenance block coupled to said reconfigurable logic device and self-refresh command inputs of said DRAM memory, said data maintenance block operative to provide stable input levels on said self-refresh command inputs while said reconfigurable logic device is reconfigured.

2. The computer system of claim **1** wherein said DRAM memory comprises DDR3 compliant memory devices.

3. The computer system of claim **1** wherein said reconfigurable logic device comprises an FPGA.

4. The computer system of claim **1** wherein said data maintenance block comprises a command decode portion coupled to a reconfigure controller of said reconfigurable logic device.

5. The computer system of claim **4** wherein said command decode portion of said data maintenance block is operative in response to a command from said reconfigure controller and provides an acknowledgement signal in response.

6. The computer system of claim **1** wherein said data maintenance block comprises a memory block coupled to a reconfigure controller of said reconfigurable logic device.

7. The computer system of claim **6** wherein said memory block is operative to retain data received from said reconfigure controller of said reconfigurable logic device.

8. The computer system of claim **1** wherein said data maintenance block comprises a CPLD.

9. The computer system of claim **1** wherein said reconfigurable logic device comprises said data maintenance block.

10. The computer system of claim **1** wherein said data maintenance block is operable to hold said DRAM memory in self-refresh mode while said reconfigurable logic device is reconfigured.

11. A method for preserving contents of a DRAM memory associated with a reconfigurable device having a memory controller comprising:

providing a data maintenance block coupled to said reconfigurable device;

coupling said data maintenance block to self-refresh command inputs of said DRAM memory;

storing data received from said reconfigurable device at said data maintenance block; and

maintaining stable input levels on said self-refresh command inputs while said reconfigurable logic device is reconfigured.

12. The method of claim **11** wherein said step of providing comprises:

providing a command decode portion of said data maintenance block coupled to receive commands from said reconfigurable device and return acknowledgment signals in response thereto.

13. The method of claim **11** wherein said step of storing comprises:

providing a memory block in said data maintenance block for storing said data received from said reconfigurable device and returning said data to said reconfigurable device upon completion of a reconfiguration function.

14. The method of claim **11** wherein said step of storing comprises:

providing a memory block in said data maintenance block for storing said data received directly from said DRAM memory and returning said data directly to said DRAM memory upon completion of a reconfiguration function.

15. The method of claim **11** wherein said step of providing said data maintenance block comprises:

providing a portion of said reconfigurable device as said data maintenance block.

16. The method of claim **11** wherein said step of providing said data maintenance block comprises:

providing a CPLD as said data maintenance block.

17. The method of claim **11** wherein said step of providing said data maintenance block comprises:

providing a block RAM for storing said data received from said reconfigurable device; and

providing a command decode portion responsive to said reconfigurable device and coupled to said reset and lock enable inputs of said DRAM memory.

18. The method of claim **11** further comprising:

passing said data between sequential computing tasks in a reconfigurable computing environment.

19. The method of claim **11** further comprising:

preserving said data at said data maintenance block while said reconfigurable logic device is powered down.

* * * * *

EXHIBIT D

US007225324B2

(12) **United States Patent**
Huppenthal et al.

(10) **Patent No.: US 7,225,324 B2**
(45) **Date of Patent: May 29, 2007**

(54) **MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS**

(75) Inventors: **Jon M. Huppenthal**, Colorado Springs, CO (US); **David E. Caliga**, Colorado Springs, CO (US)

(73) Assignee: **SRC Computers, Inc.**, Colorado Springs, CO (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 550 days.

(21) Appl. No.: **10/285,318**

(22) Filed: **Oct. 31, 2002**

(65) **Prior Publication Data**

US 2004/0088527 A1 May 6, 2004

(51) **Int. Cl.**
G06F 17/00 (2006.01)

(52) **U.S. Cl.** **712/226**

(58) **Field of Classification Search** 712/15, 712/19, 226, 215
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,727,503	A		2/1988	McWhirter	
4,872,133	A	*	10/1989	Leeland	708/509
4,962,381	A	*	10/1990	Helbig, Sr.	342/372
5,020,059	A		5/1991	Gorin et al.	
5,072,371	A	*	12/1991	Benner et al.	712/11
5,230,057	A		7/1993	Shido et al.	
5,274,832	A	*	12/1993	Khan	708/424
5,471,627	A		11/1995	Means et al.	
5,477,221	A		12/1995	Chang et al.	
5,570,040	A		10/1996	Lytle et al.	
5,640,586	A	*	6/1997	Pechanek et al.	712/13
5,737,766	A		4/1998	Tan	
5,784,108	A	*	7/1998	Skaletzky et al.	375/240.15
5,892,962	A		4/1999	Cloutier	
5,903,771	A		5/1999	Sgro et al.	
5,915,123	A	*	6/1999	Mirsky et al.	712/16
5,956,518	A	*	9/1999	DeHon et al.	712/15
6,023,755	A		2/2000	Casselman	
6,052,773	A		4/2000	DeHon et al.	
6,061,706	A	*	5/2000	Gai et al.	708/491
6,076,152	A		6/2000	Huppenthal et al.	
6,192,439	B1		2/2001	Grunewald et al.	
6,215,898	B1	*	4/2001	Woodfill et al.	382/154
6,226,776	B1		5/2001	Panchul et al.	
6,289,440	B1	*	9/2001	Casselman	712/227
6,385,757	B1	*	5/2002	Gupta et al.	716/1

OTHER PUBLICATIONS

Rosenberg, J. M., Dictionary of Computers , Information Processing &Telecommunications, 1984, John Wiley&Sons, 2ed, pp. 496.*

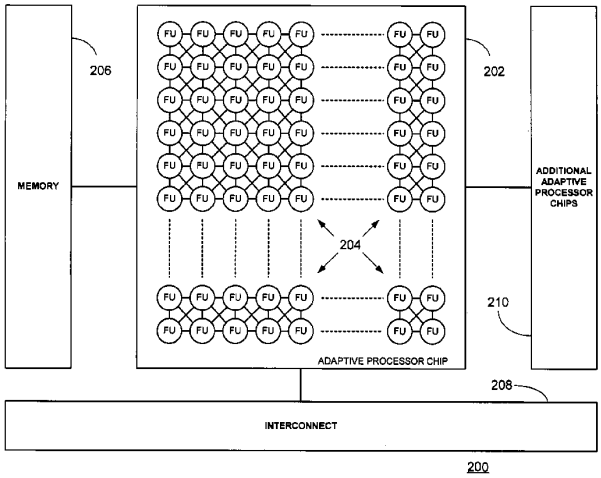
(Continued)

Primary Examiner—Eric Coleman
(74) *Attorney, Agent, or Firm*—William J. Kubida; Michael C. Martensen; Hogan & Hartson LLP

(57) **ABSTRACT**

Multi-adaptive processing systems and techniques for enhancing parallelism and performance of computational functions are disclosed which can be employed in a myriad of applications including multi-dimensional pipeline computations for seismic applications, search algorithms, information security, chemical and biological applications, filtering and the like as well as for systolic wavefront computations for fluid flow and structures analysis, bioinformatics etc. Some applications may also employ both the multi-dimensional pipeline and systolic wavefront methodologies disclosed.

52 Claims, 20 Drawing Sheets



OTHER PUBLICATIONS

Agarwal, A., et al., "The Raw Compiler Project", pp. 1-12, http://cag-www.lcs.mit.edu/raw, Proceedings of the Second SUIF Compiler Workshop, Aug. 21-23, 1997.

Albaharna, Osama, et al., "On the viability of FPGA-based integrated coprocessors", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 206-215.

Amerson, Rick, et al., "Teramac—Configurable Custom Computing", © 1995 IEEE, Publ. No. 0-8186-7086-X/95, pp. 32-38.

Barthel, Dominique Aug. 25-26, 1997, "PVP a Parallel Video coProcessor", Hot Chips IX, pp. 203-210.

Bertin, Patrice, et al., "Programmable active memories: a performance assessment", © 1993 Massachusetts Institute of Technology, pp. 88-102.

Bittner, Ray, et al., "Computing kernels implemented with a wormhole RTR CCM", © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 98-105.

Buell, D., et al. "Splash 2: FPGAs in a Custom Computing Machine—Chapter 1—Custom Computing Machines: An Introduction", pp. 1-11, http://www.computer.org/espress/catalog/bp07413/spls-ch1.html (originally believed published in J. of Supercomputing, vol. IX, 1995, pp. 219-230.

Casselman, Steven, "Virtual Computing and The Virtual Computer", © 1993 IEEE, Publ. No. 0-8186-3890-7/93, pp. 43-48.

Chan, Pak, et al., "Architectural tradeoffs in field-programmable-device-based computing systems", © 1993 IEEE, Publ. No. 0-8186-3890-7/93, pp. 152-161.

Clark, David, et al., "Supporting FPGA microprocessors through retargetable software tools", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 195-103.

Cuccaro, Steven, et al., "The CM-2X: a hybrid CM-2/Xilink prototype", © 1993 IEEE, Publ. No. 0-8186-3890-7/93, pp. 121-130.

Culbertson, W. Bruce, et al., "Exploring architectures for volume visualization on the Teramac custom computer", © 1996 IEEE, Pub. No. 0-8186-7548-9/96, pp. 80-88.

Culbertson, W. Bruce, et al., "Defect tolerance on the Teramac custom computer", © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 116-123.

Dehon, Andre, "DPGA-Coupled microprocessors: commodity IC for the early 21st century", © 1994 IEEE, Publ. No. 0-8186-5490-2/94, pp. 31-39.

Dehon, A., et al., "MATRIX A Reconfigurable Computing Device with Configurable Instruction Distribution", Hot Chips IX, Aug. 25-26, 1997, Stanford, California, MIT Artificial Intelligence Laboratory.

Dhaussy, Philippe, et al., "Global control synthesis for an MIMD/FPGA machine", © 1994 IEEE, Publ. No. 0-8186-5490-2/94, pp. 72-81.

Elliott, Duncan, et al., "Computational Ram: a memory-SIMD hybrid and its application to DSP", © 1992 IEEE, Publ. No. 0-7803-0246-X/92, pp. 30.6.1-30.6.4.

Fortes, Jose, et al., "Systolic arrays, a survey of seven projects", © 1987 IEEE, Publ. No. 0018-9162/87/0700-0091, pp. 91-103.

Gokhale, M., et al., "Processing in Memory: The Terasys Massively Parallel PIM Array" © Apr. 1995, IEEE, pp. 23-31.

Gunther, Bernard, et al., "Assessing Document Relevance with Run-Time Reconfigurable Machines", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 10-17.

Hagiwara, Hiroshi, et al., "A dynamically microprogrammable computer with low-level parallelism", © 1980 IEEE, Publ. No. 0018-9340/80/07000-0577, pp. 577-594.

Hartenstein, R. W., et al. "A General Approach in System Design Integrating Reconfigurable Accelerators," http://xputers.informatik.uni-kl.de/papers/paper026-1.html, IEEE 1996 Conference, Austin, TX, Oct. 9-11, 1996.

Hartenstein, Reiner, et al., "A reconfigurable data-driven ALU for Xputers", © 1994 IEEE, Publ. No. 0-8186-5490-2/94, pp. 139-146.

Hauser, John, et al.: "GARP: a MIPS processor with a reconfigurable co-processor", © 1997 IEEE, Publ. No. 0-08186-8159-4/97, pp. 12-21.

Hayes, John, et al., "A microprocessor-based hypercube, supercomputer", © 1986 IEEE, Publ. No. 0272-1732/86/1000-0006, pp. 6-17.

Herpel, H.-J., et al., "A Reconfigurable Computer for Embedded Control Applications", © 1993 IEEE, Publ. No. 0-8186-3890-7/93, pp. 111-120.

Hogl, H., et al., "Enable++: A second generation FPGA processor", © 1995 IEEE, Publ. No. 0-8186-7086-X/95, pp. 45-53.

King, William, et al., "Using MORRPH in an industrial machine vision system", © 1996 IEEE, Publ. No. 08186-7548-9/96, pp. 18-26.

Manohar, Swaminathan, et al., "A pragmatic approach to systolic design", © 1988 IEEE, Publ. No. CH2603-9/88/0000/0463, pp. 463-472.

Mauduit, Nicolas, et al., "Lneuro 1.0: a piece of hardware LEGO for building neural network systems," © 1992 IEEE, Publ. No. 1045-9227/92, pp. 414-422.

Mirsky, Ethan A., "Coarse-Grain Reconfigurable Computing", Massachusetts Institute of Technology, Jun. 1996.

Mirsky, Ethan, et al., "MATRIX: A Reconfigurable Computing Architecture with Configurable Instruction Distribution and Deployable Resources", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 157-166.

Morley, Robert E., Jr., et al., "A Massively Parallel Systolic Array Processor System", © 1988 IEEE, Publ. No. CH2603-9/88/0000/0217, pp. 217-225.

Patterson, David, et al., "A case for intelligent DRAM: IRAM", Hot Chips VIII, Aug. 19-20, 1996, pp. 75-94.

Peterson, Janes, et al., "Scheduling and partitioning ANSI-C programs onto multi-FPGA CCM architectures", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 178-187.

Schmit, Herman, "Incremental reconfiguration for pipelined applications," © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 47-55.

Sitkoff, Nathan, et al., "Implementing a Genetic Algorithm on a Parallel Custom Computing Machine", Publ. No. 0-8186-7086-X/95, pp. 180-187.

Stone, Harold, "A logic-in-memory computer", © 1970 IEEE, IEEE Transactions on Computers, pp. 73-78, Jan. 1990.

Tangen, Uwe, et al., "A parallel hardware evolvable computer POLYP extended abstract", © 1997 IEEE, Publ. No. 0-8186-8159/4/97, pp. 238-239.

Thornburg, Mike, et al., "Transformable Computers", © 1994 IEEE, Publ. No. 0-8186-5602-6/94, pp. 674-679.

Tomita, Shinji, et al., "A computer low-level parallelism QA-2", © 1986 IEEE, Publ. No. 0-0384-7495/86/0000/0280, pp. 280-289.

Trimberger, Steve, et al., "A time-multiplexed FPGA", © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 22-28.

Ueda, Hirotada, et al., "A multiprocessor system utilizing enhanced DSP's for image processing", © 1988 IEEE, Publ. No. CH2603-9/88/0000/0611, pp. 611-620.

Villasenor, John, et al., "Configurable computing", © 1997 Scientific American, Jun. 1997.

Wang, Quiang, et al., "Automated field-programmable compute accelerator design using partial evaluation", © 1997 IEEE, Publ. No. 0-8186-8159-4/97. pp. 145-154.

W.H. Manglone-Smith and B.L. Hutchings. Configurable computing: The Road Ahead. In Proceedings of the Reconfigurable Architectures Workshop (RAW'97), pp. 81-96, 1997.

Wirthlin, Michael, et al., "The Nano processor: a low resource reconfigurable processor", © 1994 IEEE, Publ. No. 0-8186-5490-2/94, pp. 23-30.

Wirthlin, Michael, et al., "A dynamic instruction set computer", © 1995 IEEE, Publ. No. 0-8186-7086-X/95, pp. 99-107.

Wittig, Ralph, et al., "One Chip: An FPGA processor with reconfigurable logic", © 1996 IEEE, Publ. 0-8186-7548-9/96, pp. 126-135.

Yamauchi, Tsukasa, et al., "SOP: A reconfigurable massively parallel system and its control-data flow based compiling method", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 148-156.

"Information Brief", PCI Bus Technology, © IBM Personal Computer Company, 1997, pp. 1-3.

Yun, Hyun-Kyu and Silverman, H. F.; "A distributed memory MIMD multi-computer with reconfigurable custom computing capabilities", Brown University, Dec. 10-13, 1997, pp. 7-13.
Hoover, Chris and Hart, David; "San Diego Supercomputer Center, Timelogic and Sun Validate Ultra-Fast Hidden Markov Model Analysis-One DeCypher-accelerated Sun Fire 6800 beats 2,600 CPUs running Linux-", San Diego Supercomputer Center, http://www.sdsc.edu/Press/02/050802_markovmodel.html, May 8, 2002, pp. 1-3.
Caliga, David and Barker, David Peter, "Delivering Acceleration: The Potential for Increased HPC Application Performance Using Reconfigurable Logic", SRC Computers, Inc., Nov. 2001, pp. 20.
Hammes, J.P., Rinker, R. E.; McClure, D, M., Böhm, A. P. W., Najjar, W. A., "The SA-C Compiler Dataflow Description", Colorado State University, Jun. 21, 2001, pp. 1-25.
Callahan, Timothy J. and Wawrzynek, John, "Adapting Software Pipelining for Reconfigurable Computing", University of California at Berkeley, Nov. 17-19, 2000, pp. 8.
Ratha, Nalini K., Jain, Anil K. and Rover, Diane T., "An FPGA-based Point Pattern Matching Processor with Application to Fingerprint Matching", Michigan State University, Department of Computer Science, pp. 8.
Dehon, André, "Comparing Computing Machines", University of California at Berkeley, Proceedings of SPIE vol. 3526, Nov. 2-3, 1998, pp. 11.
Vemuri, Ranga R. and Harr, Randolph E., "Configurable Computing: Technology and Applications", University of Cincinnati and Synopsys Inc., IEEE, Apr. 2000, pp. 39-40.
Dehon, André, "The Density Advantage of Configurable Computing", California Institute of Technology, IEEE, Apr. 2000. pp. 41-49.
Haynes, Simon D., Stone, John, Cheung, Peter Y.K. and Luk, Wayne, "Video Image Processing with the Sonic Architecture", Sony Broadcast & Professional Europe, Imperial College, University of London, IEEE, Apr. 2000, pp. 50-57.
Platzner , Marco, "Reconfigurable Accelerators for Combinatorial Problems", Swiss Federal Institute of Technology (ETH) Zurich, IEEE, Apr. 2000, pp. 58-60.
Callahan, Timothy J., Hauser, John R. And Wawrzynek, John, "The Garp Architecture and C Compiler", University of California, Berkeley, IEEE, April 2000, pp. 62-69.
Goldstein, Seth Copen, Schmit, Herman, Budiu , Mihai, Cadambi, Srihari, Moe, Matt and Taylor, R. Reed, "PipeRench: A Reconfigurable Architecture and Compiler", Carnegie Mellon University, IEEE, Apr. 2000, pp. 70-76.
Muchnick, Steven S., "Advanced Compiler Design and Implementation", Morgan Kaufmann Publishers, pp. 217.
Hammes, Jeffrey P., Dissertation "Compiling SA-C To Reconfigurable Computing Systems", Colorado State University, Department of Computer Science, Summer 2000, pp. 179.
Automatic Target Recognition, Colorado State University & USAF, http://www.cs.colostate.edu/cameron/applications.html, pp. 1-3.
Chodowiec, Pawel, Khuon, Po, Gaj, Kris, Fast Implementations of Secret-Key Block Ciphers Using Mixed Inner- and Outer-Round Pipelining, George Mason University, Feb. 11-13, 2001, pp. 9.
Miyamori, Takashi, "REMARC: Reconfigurable Multimedia Array Coprocessor", IEICE Transactions on Information and Systems, Information & Systems Society, Tokyo, JP, vol. E82-D, No. 2, Feb. 1999, pp. 389-397, XP000821922.
Gross Thomas, et al., "Compilation for a High-performance Systolic Array", Sigplan Notices USA, vol. 21, No. 7, Jul. 1986, pp. 27-38, XP002418625.
Rauchwerger, Lawrence, et al., "The LRPD Test: Speculative Run-Time Parallelization of Loops with Privatization and Reduction Parallelization", IEEE Transaction on Parallel and Distributed Systems, IEEE Service Center, Los Alamitos, CA, vol. 10, No. 2, Feb. 1999, pp. 160-180, XP000908318.
Arnold Jeffrey M. et al., "The Splash 2 Processor and Applications", Computer Design: VLSI in Computers and Processors, 1993, ICCD '93 Proceedings, 1993 IEEE International Conference on Cambridge, MA, Oct. 3-6 1993, Los Alamitos, CA, IEEE Comput. Soc., Oct. 3, 1993, pp. 482-485, XP010134571.
Hwang, Kai, "Computer Architecture and Parallel Processing", Data Flow Computers and VLSI Computations, 1985, McGraw Hill, Chapter 10, pp. 732-807, XP-002418655.
Hartenstein, Reiner W., et al. "A Synthesis System for Bus-based Wavefront Array Architectures", Proceedings, International Conference on Application-Specific Systems, Architecture and Processors, 1996, pp. 274-283, XP002132819.
Alexander, Thomas, et al. "A Reconfigurable Approach To A Systolic Sorting Architecture", ISCAS 89, May 8, 1989, pp. 1178-1182, XP010084477.
Wu, Youfeng, et al. "Better Exploration of Region-Level Value Locality with Integrated Computation Reuse and Value Prediction", Proceedings of the 28th International Symposium on Computer Architecture, ISCA 2001, Goteberg, Sweden, Jun. 30-Jul. 4, 2001, International Symposium on Computer Architecture, (ISCA), Los Alamitos, CA, IEEE Comp. Soc, US, Jun. 30, 2001, pp. 93-103, XP010552866.

* cited by examiner

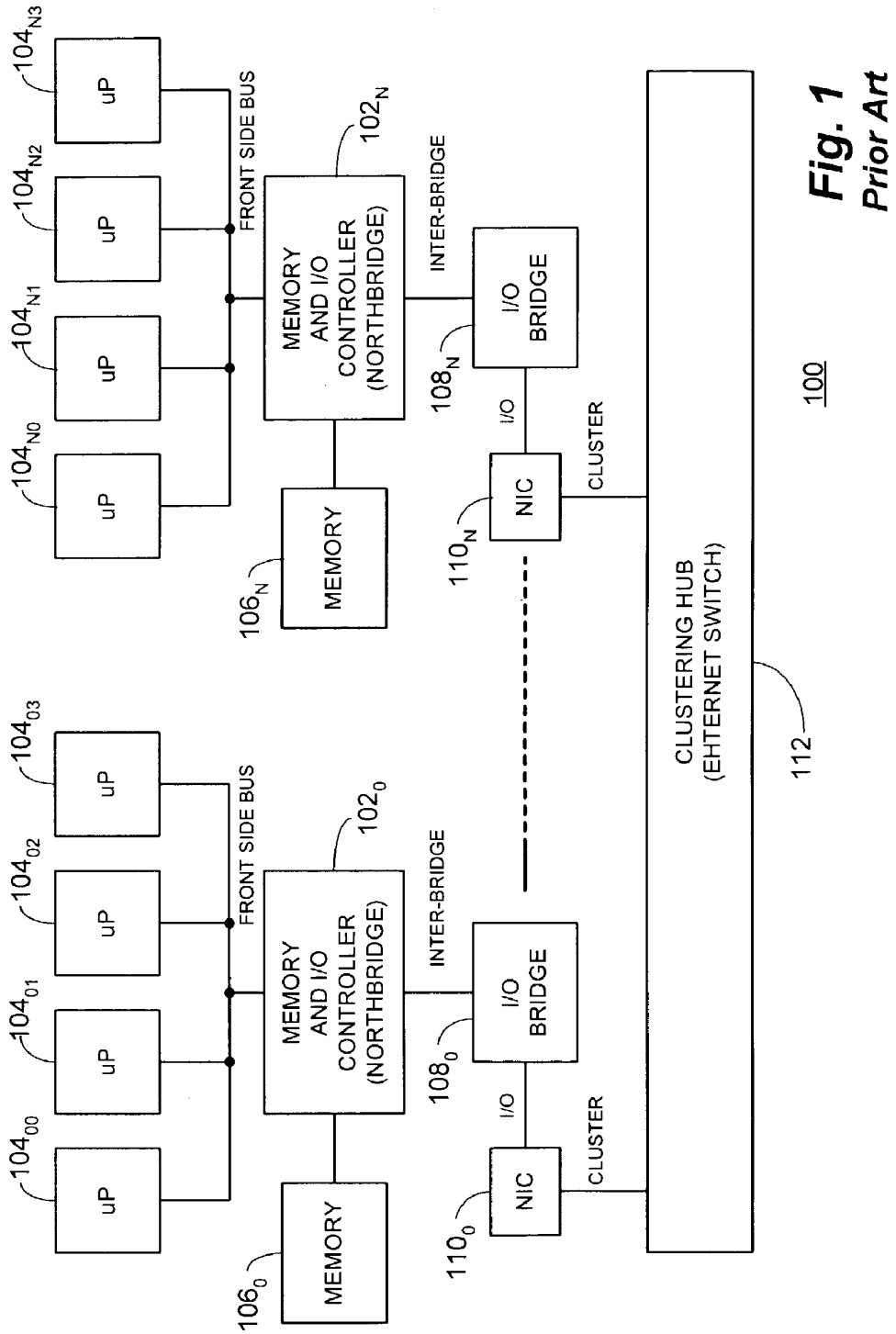


Fig. 1
Prior Art

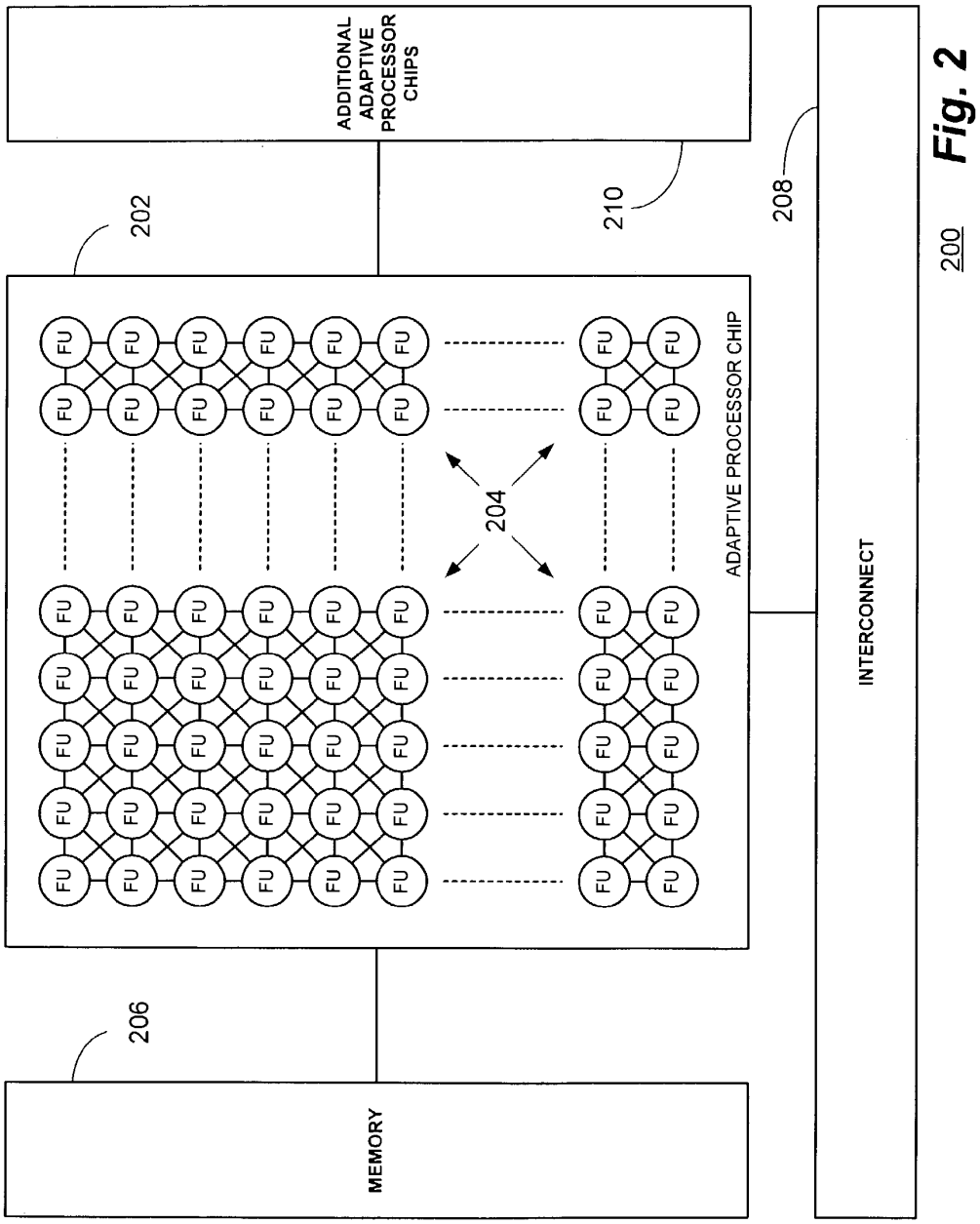

ADDITIONAL ADAPTIVE PROCESSOR CHIPS
210
208
202
200
ADAPTIVE PROCESSOR CHIP
FU
204
INTERCONNECT
206
MEMORY

Fig. 2

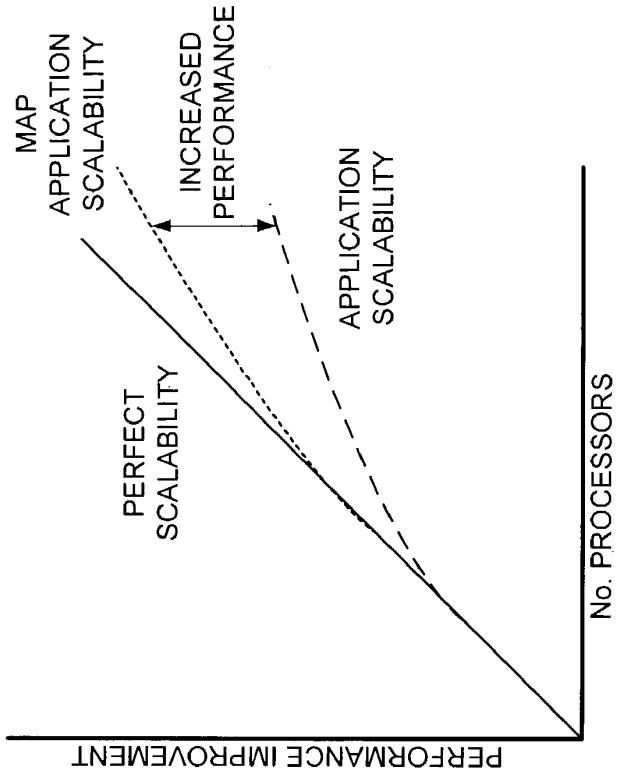


Fig. 3B

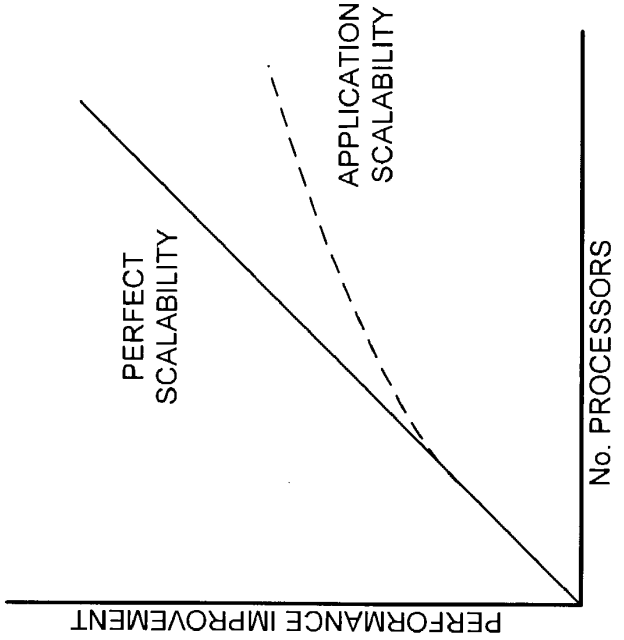


Fig. 3A
PRIOR ART

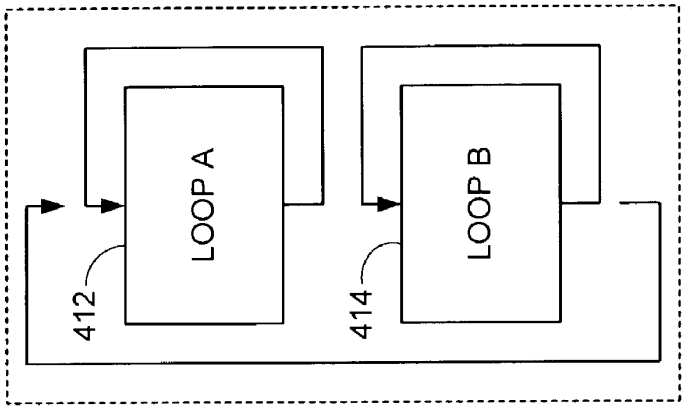


Fig. 4B

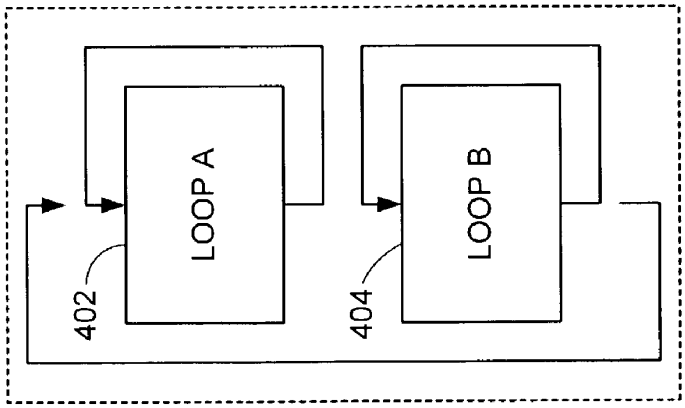


Fig. 4A
Prior Art

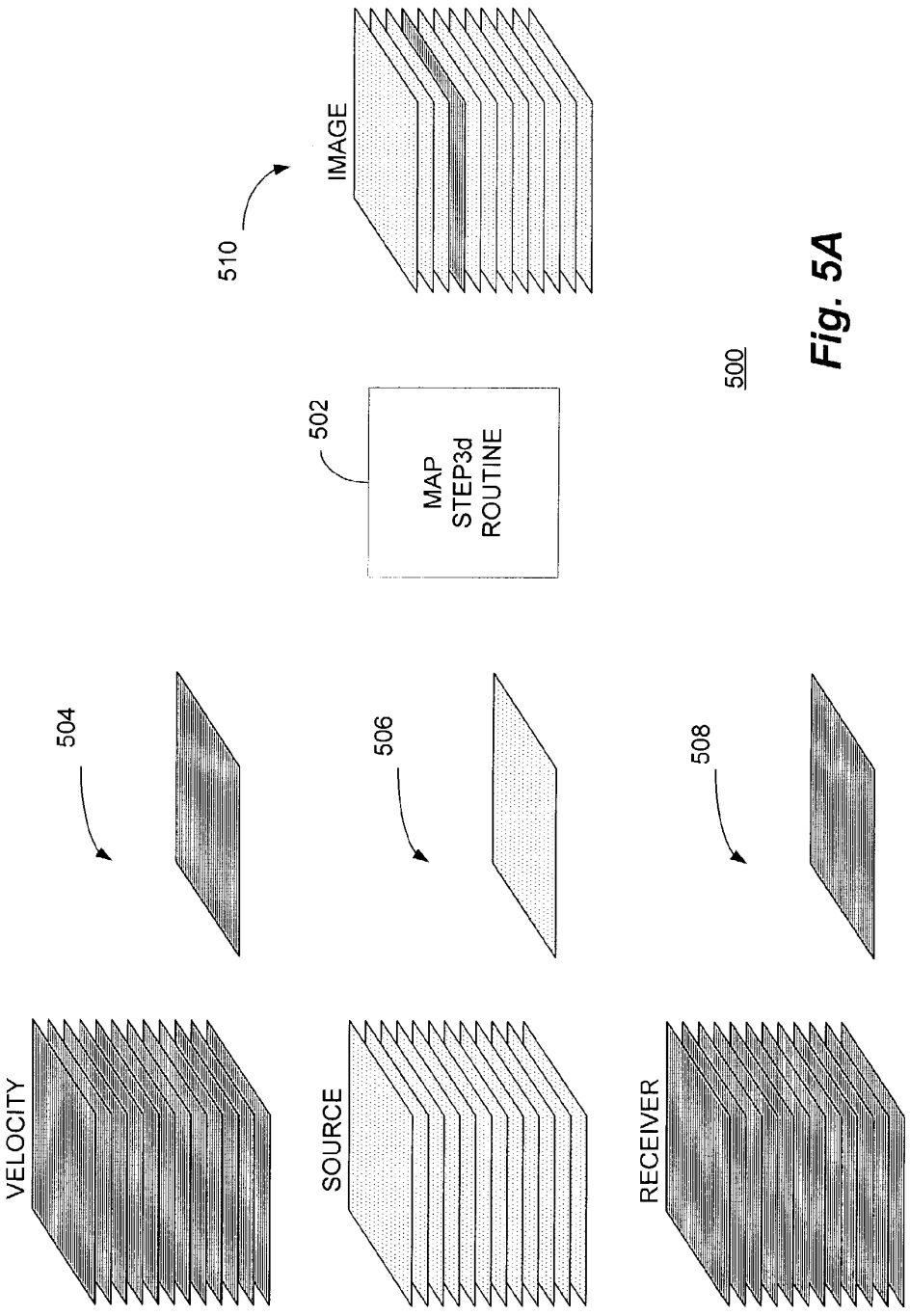
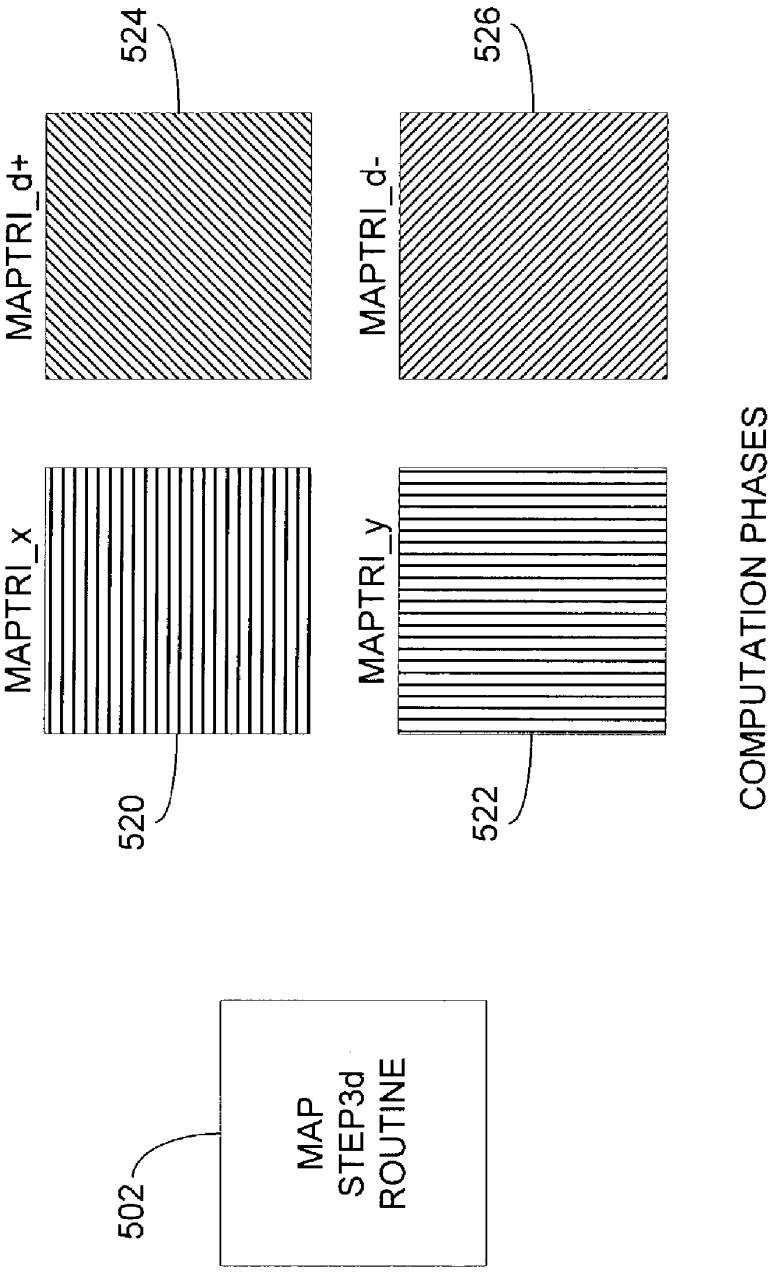

500
510
IMAGE
502
MAP
STEP3d
ROUTINE
504
506
508
VELOCITY
SOURCE
RECEIVER

Fig. 5A

Fig. 5B


524
526
MAPTRI_d+
MAPTRI_d-
MAPTRI_x
MAPTRI_y
520
522
COMPUTATION PHASES
502
MAP STEP3d ROUTINE

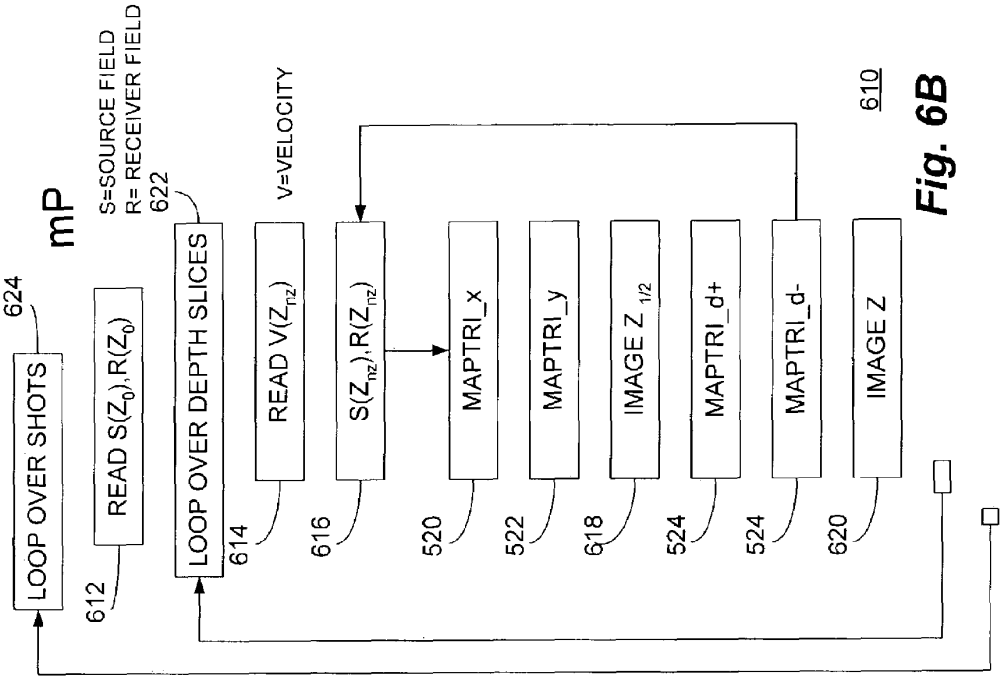

mP
S=SOURCE FIELD
R= RECEIVER FIELD
V=VELOCITY
LOOP OVER SHOTS
READ S(Z0),R(Z0)
LOOP OVER DEPTH SLICES
READ V(Znz)
S(Znz),R(Znz)
MAPTRI_x
MAPTRI_y
IMAGE Z 1/2
MAPTRI_d+
MAPTRI_d-
IMAGE Z
624
612
622
614
616
520
522
618
524
524
620
610

Fig. 6B

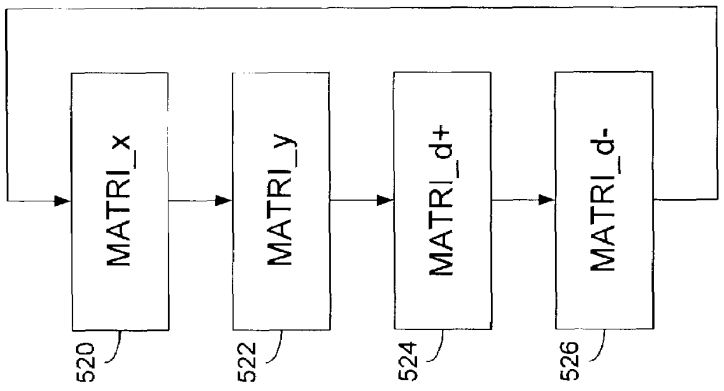

MATRI_x
MATRI_y
MATRI_d+
MATRI_d-
520
522
524
526
600

Fig. 6A

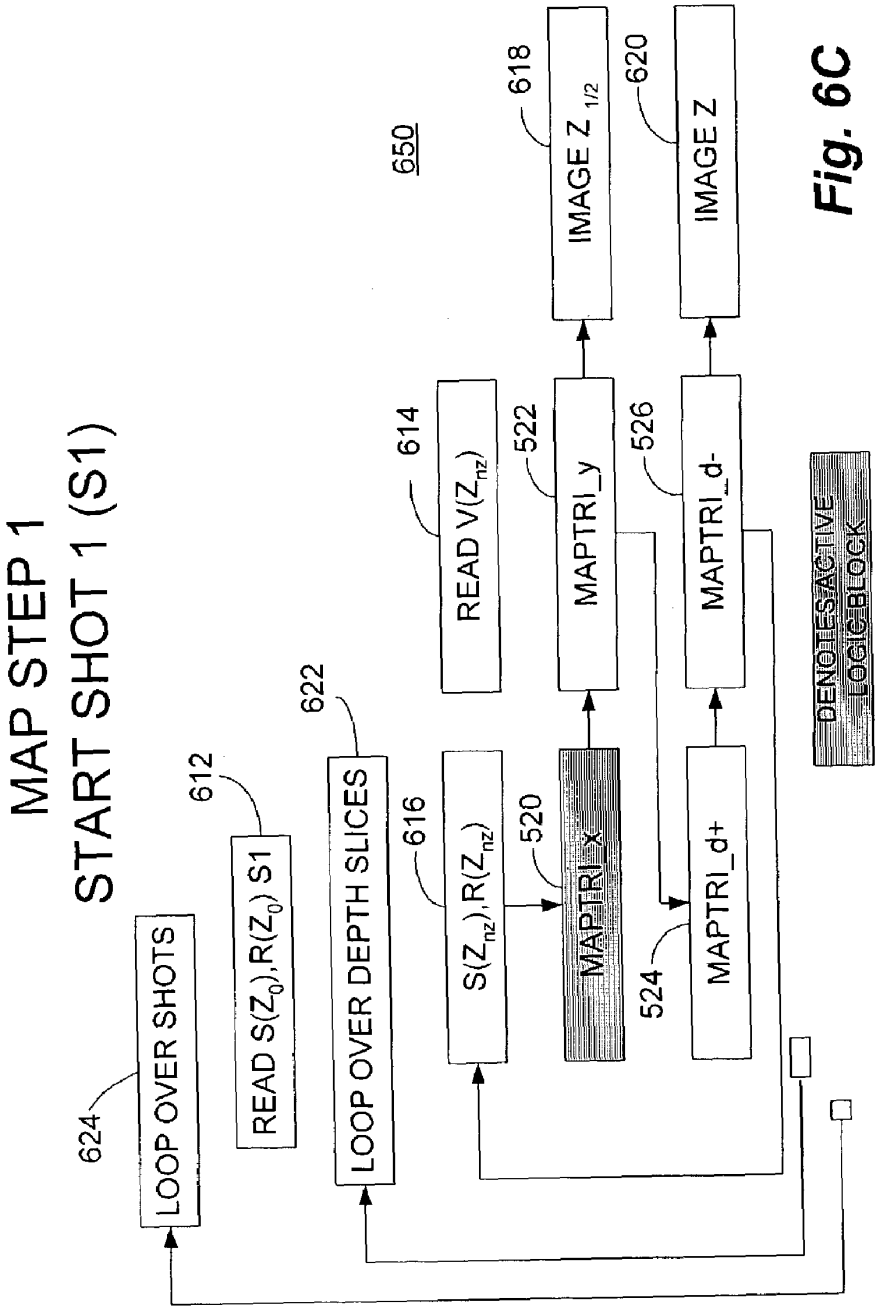

MAP STEP 1
START SHOT 1 (S1)
650
618
IMAGE Z 1/2
620
IMAGE Z
614
READ V(Znz)
522
MAPTRI_y
526
MAPTRI_d-
622
LOOP OVER DEPTH SLICES
616
S(Znz),R(Znz)
520
MAPTRI_x
524
MAPTRI_d+
612
READ S(Z0),R(Z0) S1
624
LOOP OVER SHOTS
DENOTES ACTIVE LOGIC BLOCK

Fig. 6C

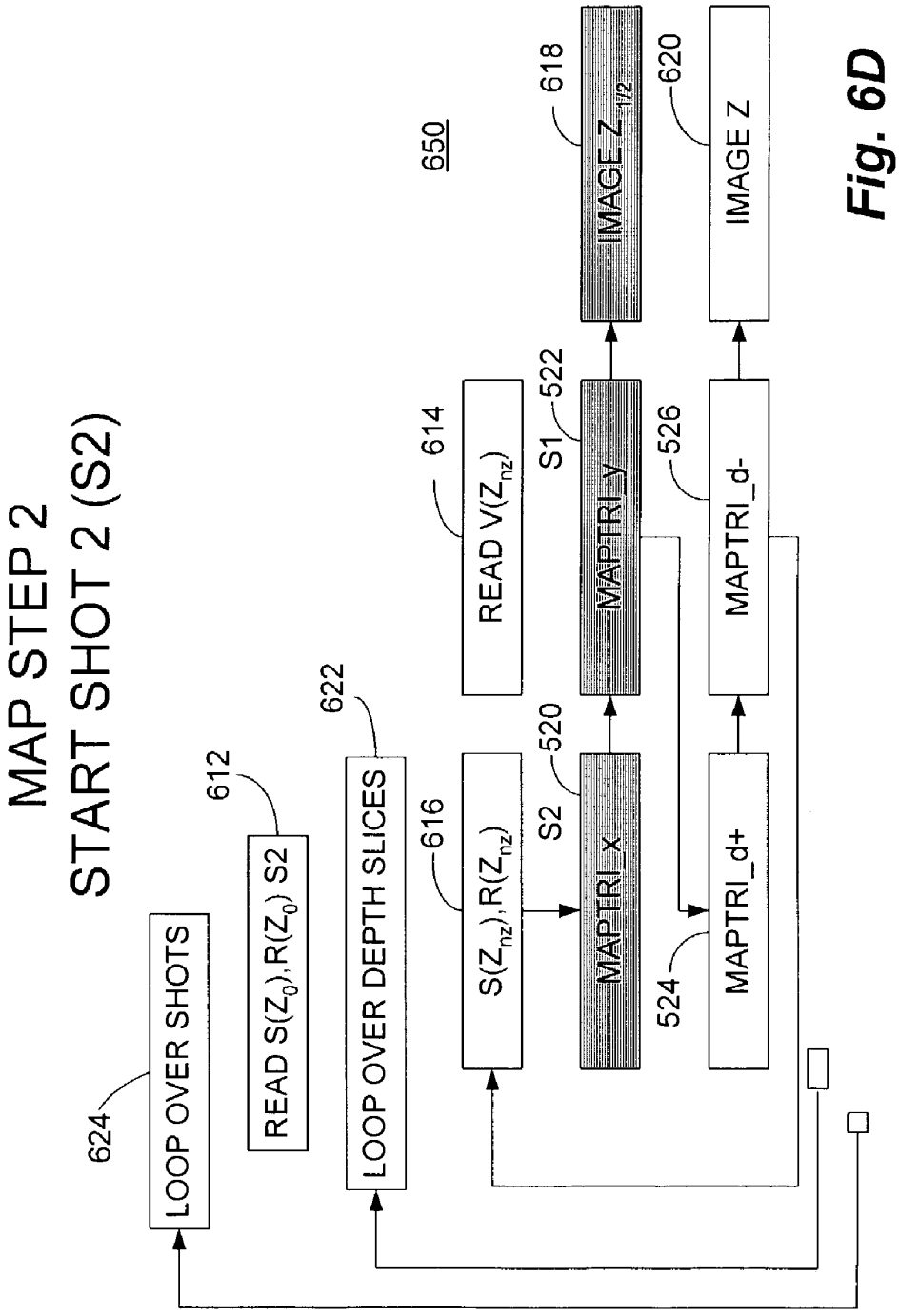

MAP STEP 2
START SHOT 2 (S2)
650
LOOP OVER SHOTS
624
READ S(Z0),R(Z0) S2
612
LOOP OVER DEPTH SLICES
622
S(Znz),R(Znz)
616
READ V(Znz)
614
MAPTRI_x
S2
520
MAPTRI_y
S1
522
MAPTRI_d+
524
MAPTRI_d-
526
IMAGE Z 1/2
618
IMAGE Z
620

Fig. 6D

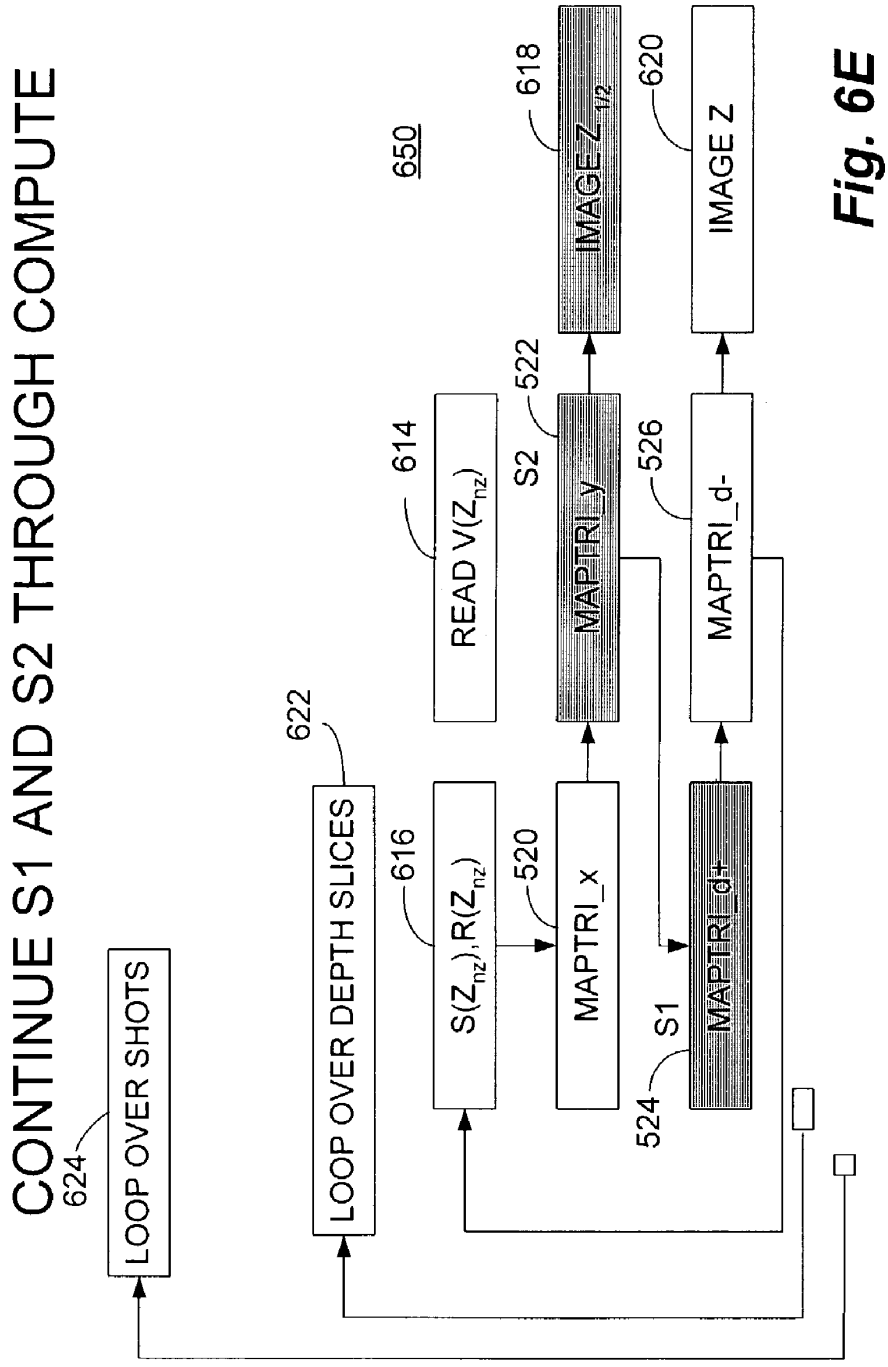

CONTINUE S1 AND S2 THROUGH COMPUTE
650
624
LOOP OVER SHOTS
622
LOOP OVER DEPTH SLICES
616
$S(Z_{nz}), R(Z_{nz})$
614
READ $V(Z_{nz})$
520
MAPTRI_x
522
S2
MAPTRI_y
618
IMAGE $Z_{1/2}$
524
S1
MAPTRI_d+
526
MAPTRI_d-
620
IMAGE Z

Fig. 6E

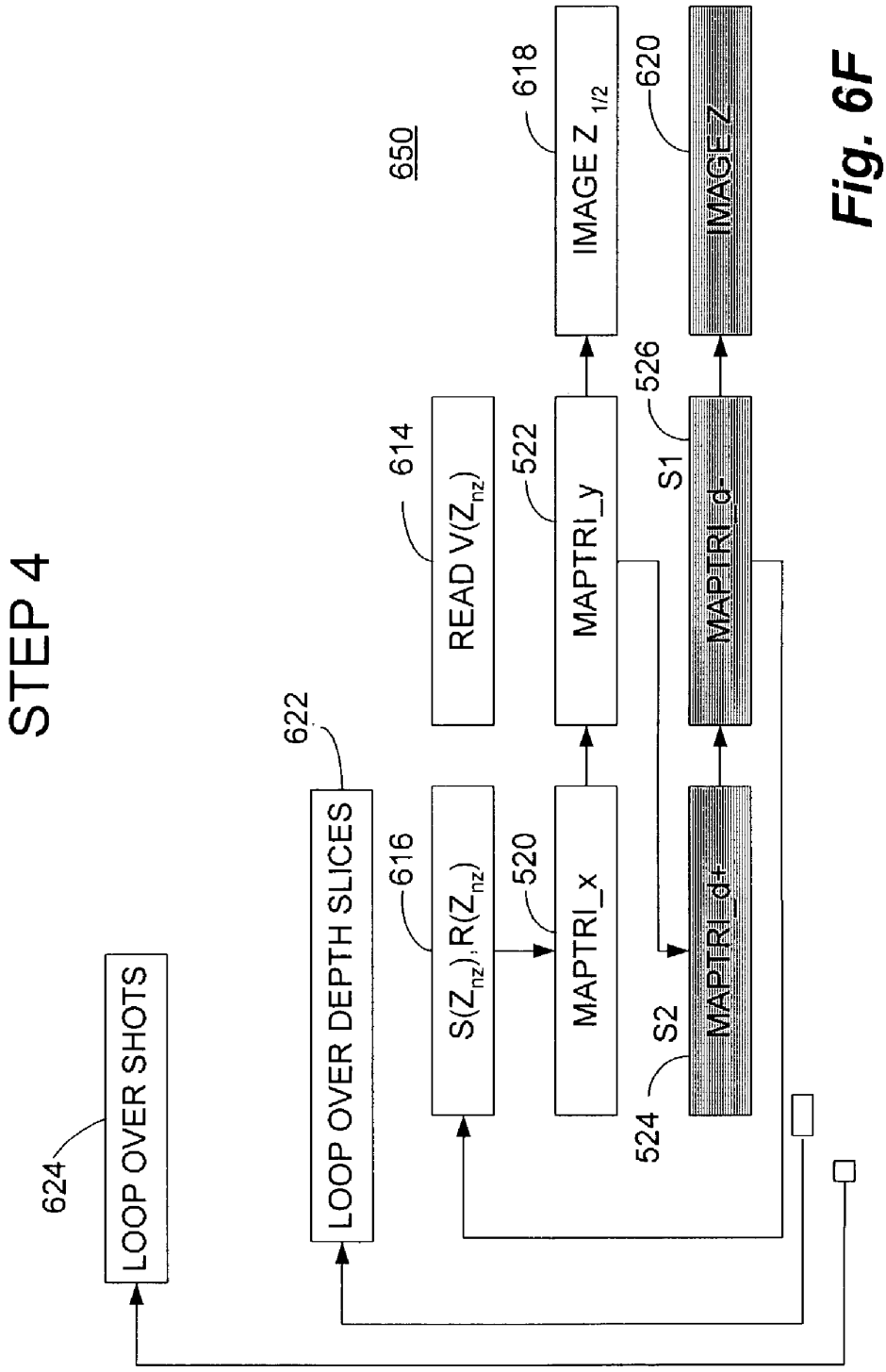

STEP 4
650
618
IMAGE Z 1/2
620
IMAGE Z
614
READ V(Znz)
522
MAPTRI_y
526
S1
MAPTRI_d-
622
LOOP OVER DEPTH SLICES
616
S(Znz), R(Znz)
520
MAPTRI_x
S2
MAPTRI_d+
524
624
LOOP OVER SHOTS

Fig. 6F

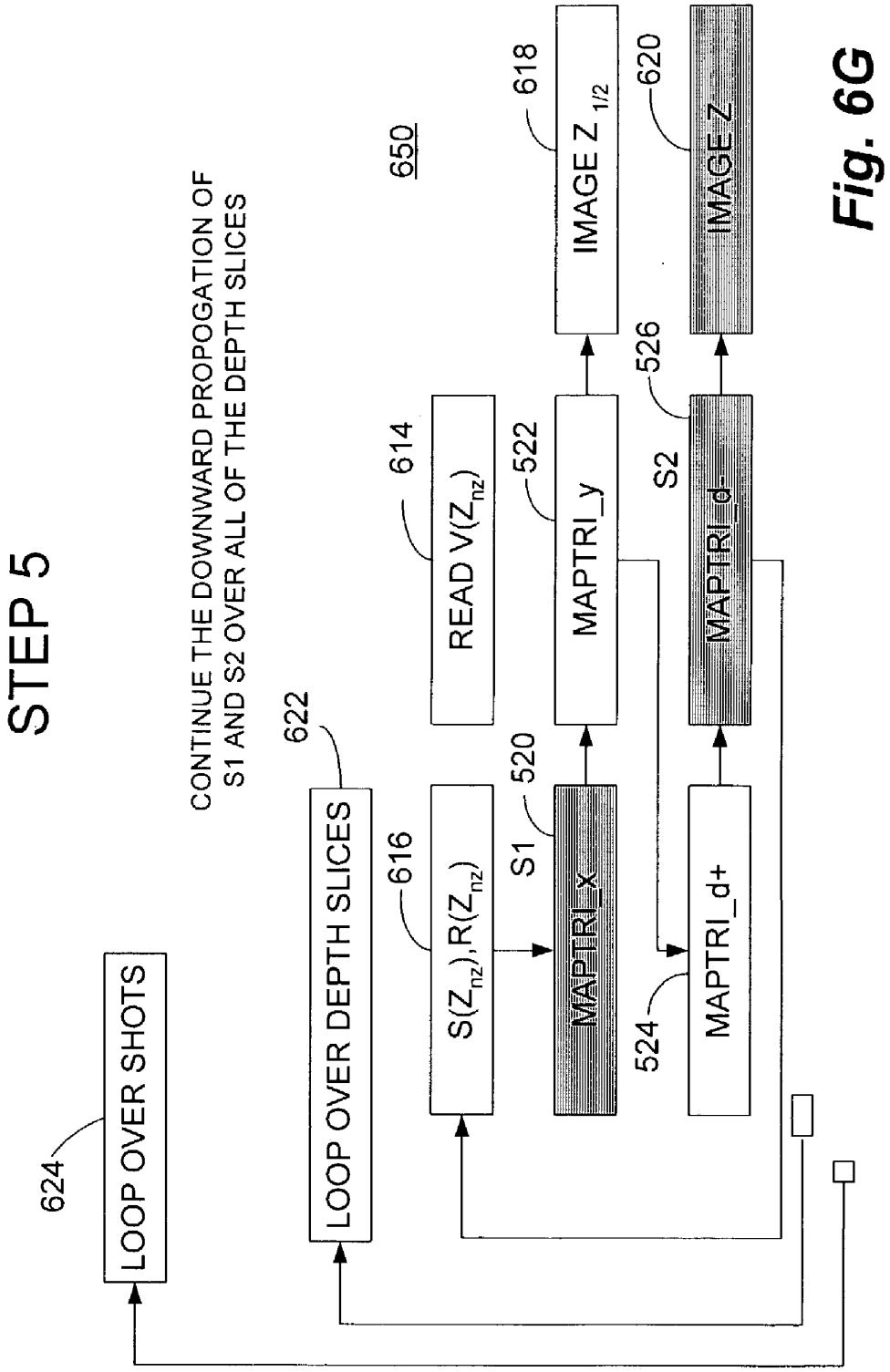

STEP 5
CONTINUE THE DOWNWARD PROPOGATION OF S1 AND S2 OVER ALL OF THE DEPTH SLICES
650
624
LOOP OVER SHOTS
622
LOOP OVER DEPTH SLICES
616
S(Z_nz),R(Z_nz)
614
READ V(Z_nz)
S1
520
MAPTRI_x
522
MAPTRI_y
524
MAPTRI_d+
S2
526
MAPTRI_d-
618
IMAGE Z 1/2
620
IMAGE Z

Fig. 6G

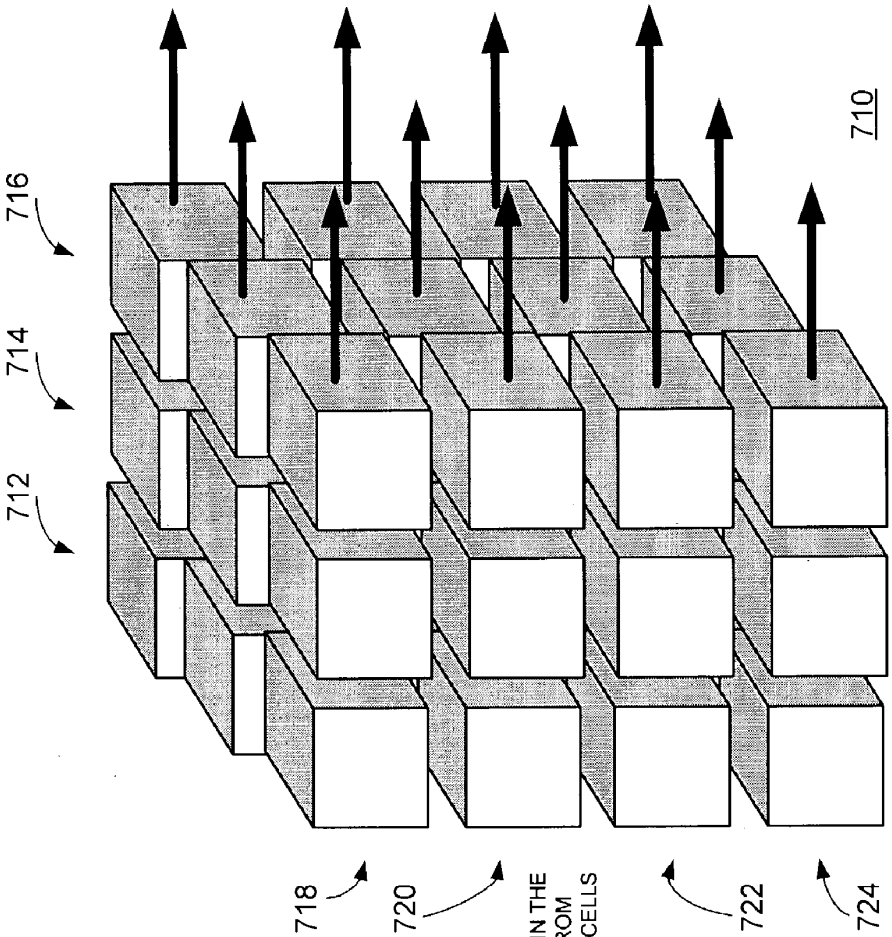

710
716
714
712
718
720
722
724

Fig. 7B

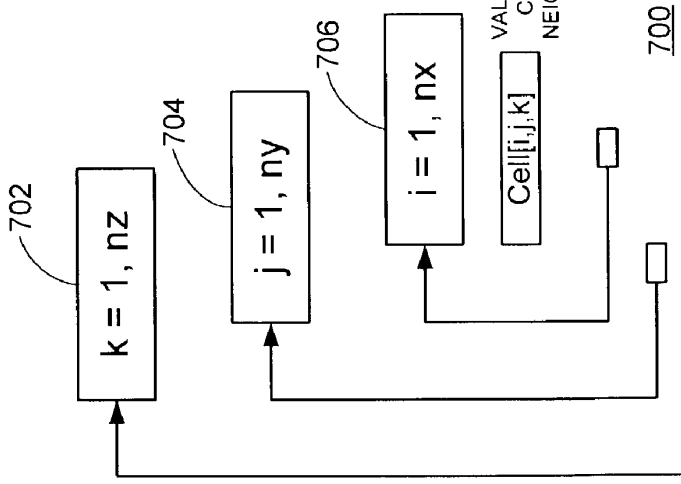

k = 1, nz
j = 1, ny
i = 1, nx
Cell[i,j,k]
VALUES USED IN THE COMPUTE FROM NEIGHBORING CELLS IN i,j,k
700
702
704
706

Fig. 7A

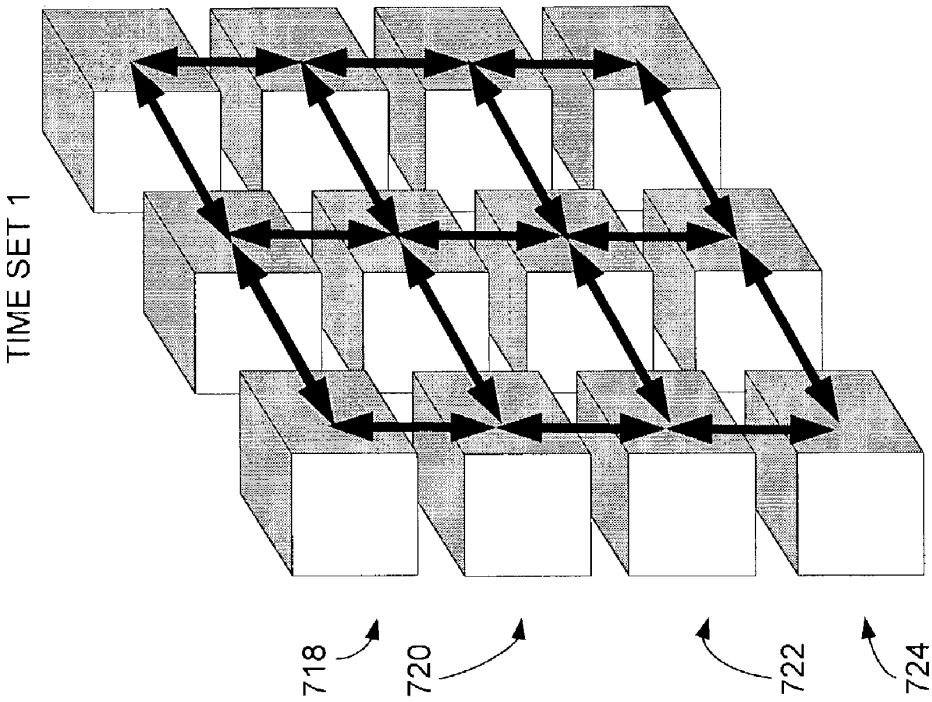


Fig.7C

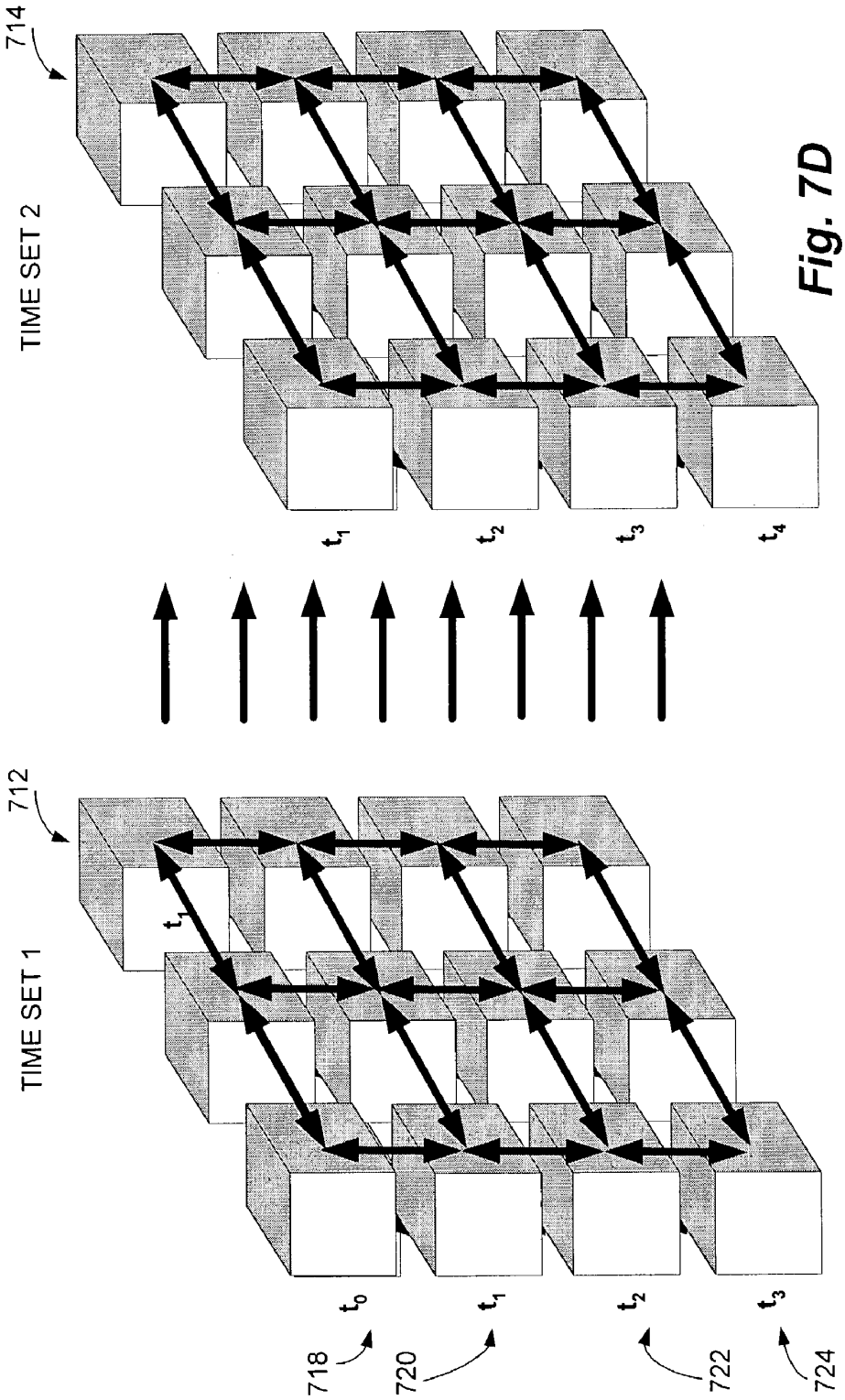

714
TIME SET 2
t_1
t_2
t_3
t_4
712
TIME SET 1
t_1
t_0
t_1
t_2
t_3
718
720
722
724

Fig. 7D

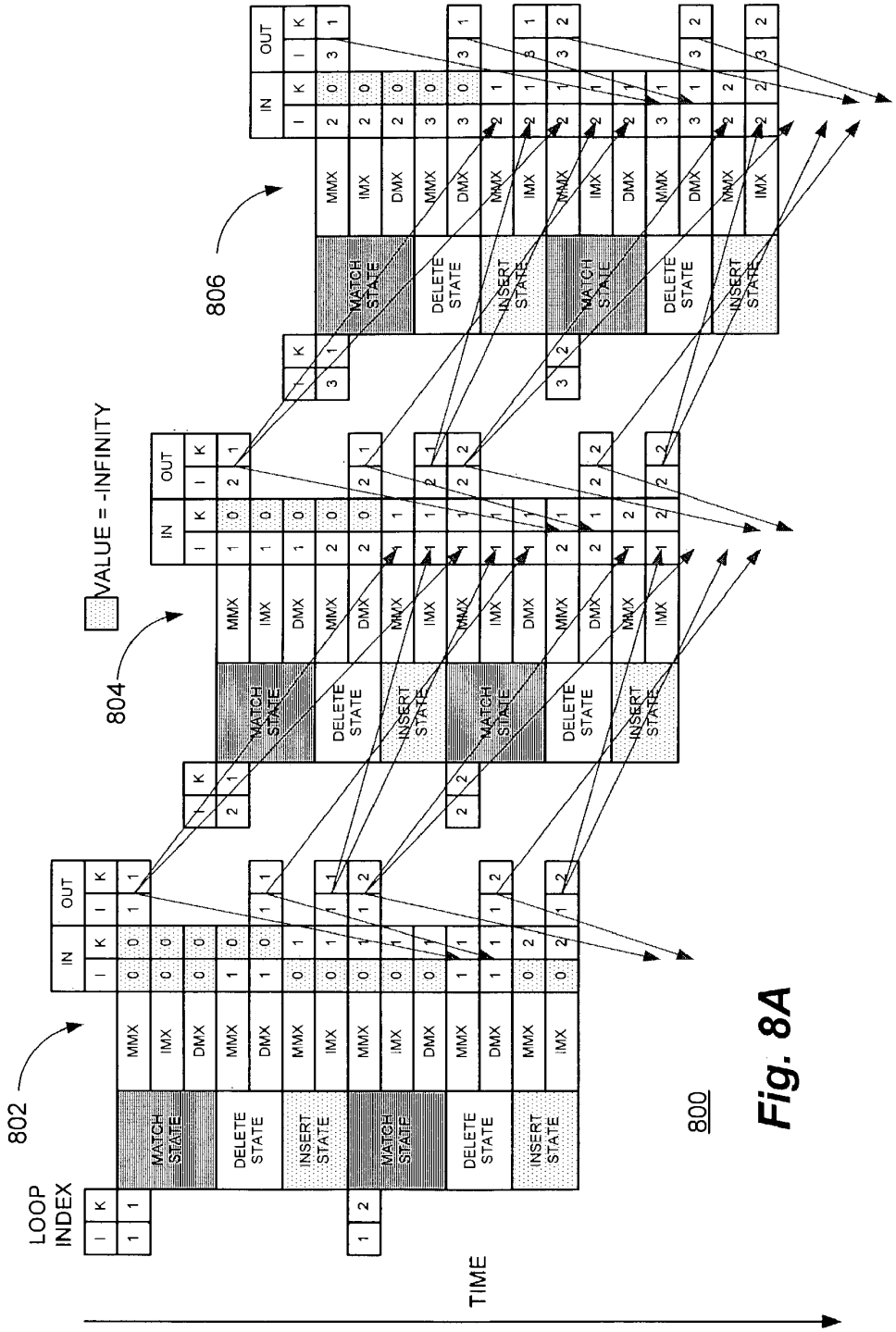

VALUE = -INFINITY
802
804
806
800
LOOP INDEX
TIME
MATCH STATE
DELETE STATE
INSERT STATE
MMX
IMX
DMX
IN
OUT
I
K

Fig. 8A

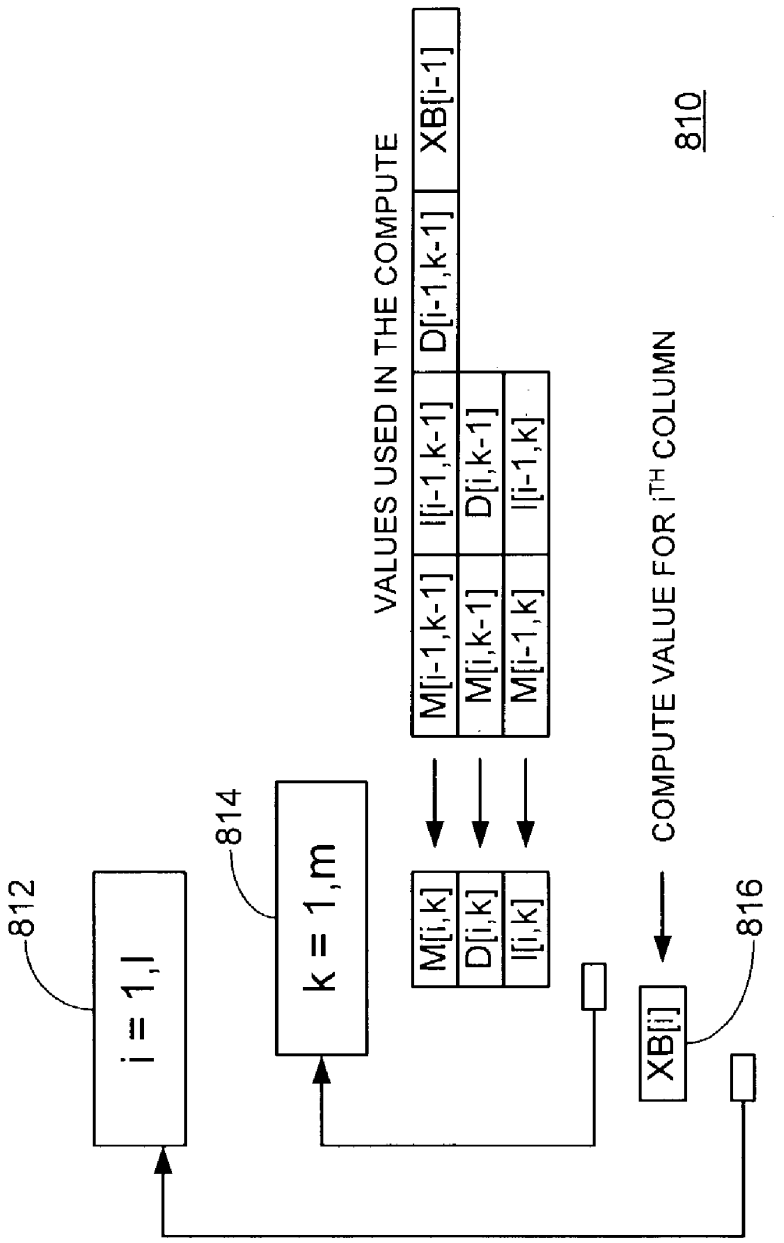
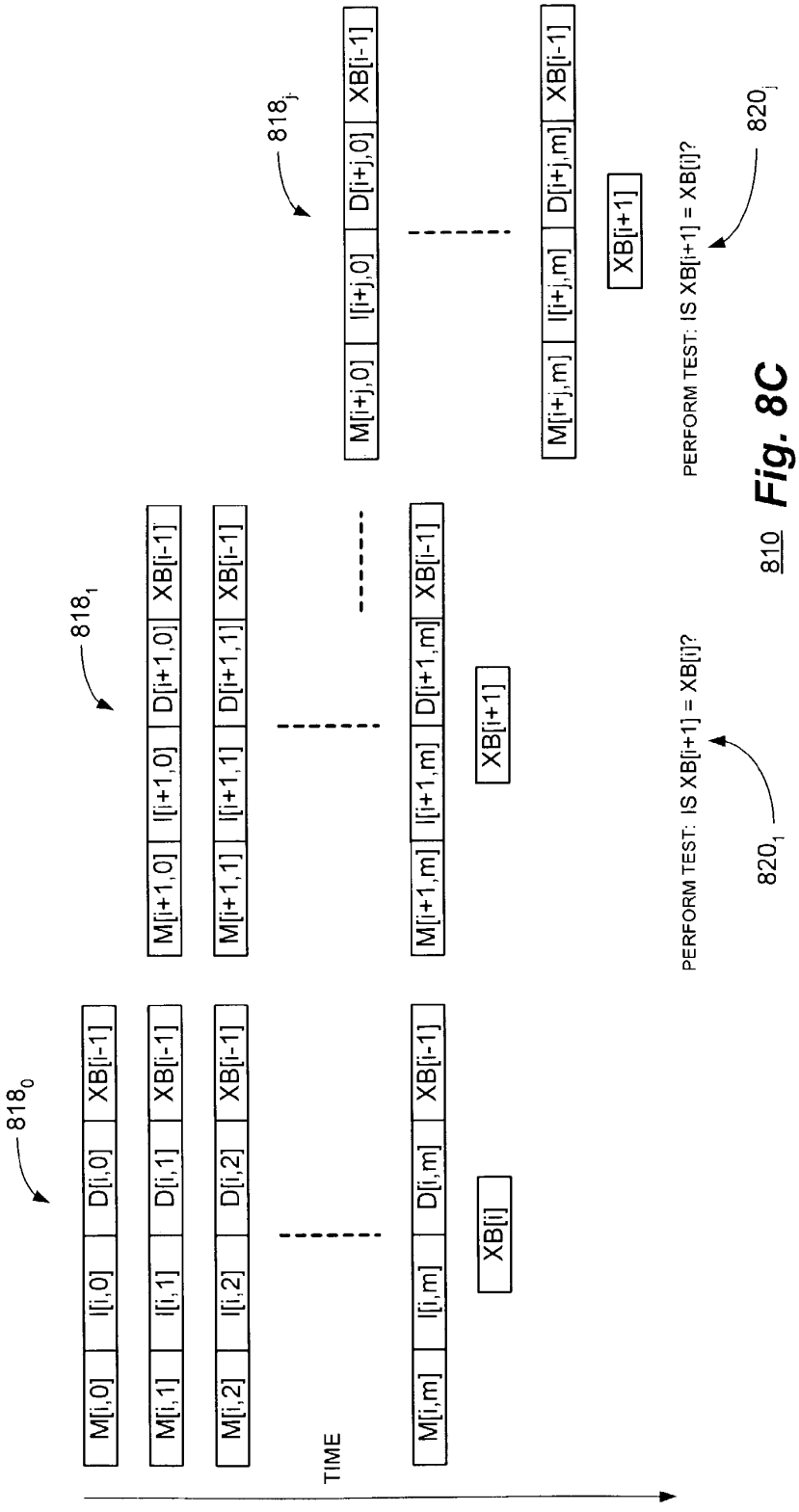
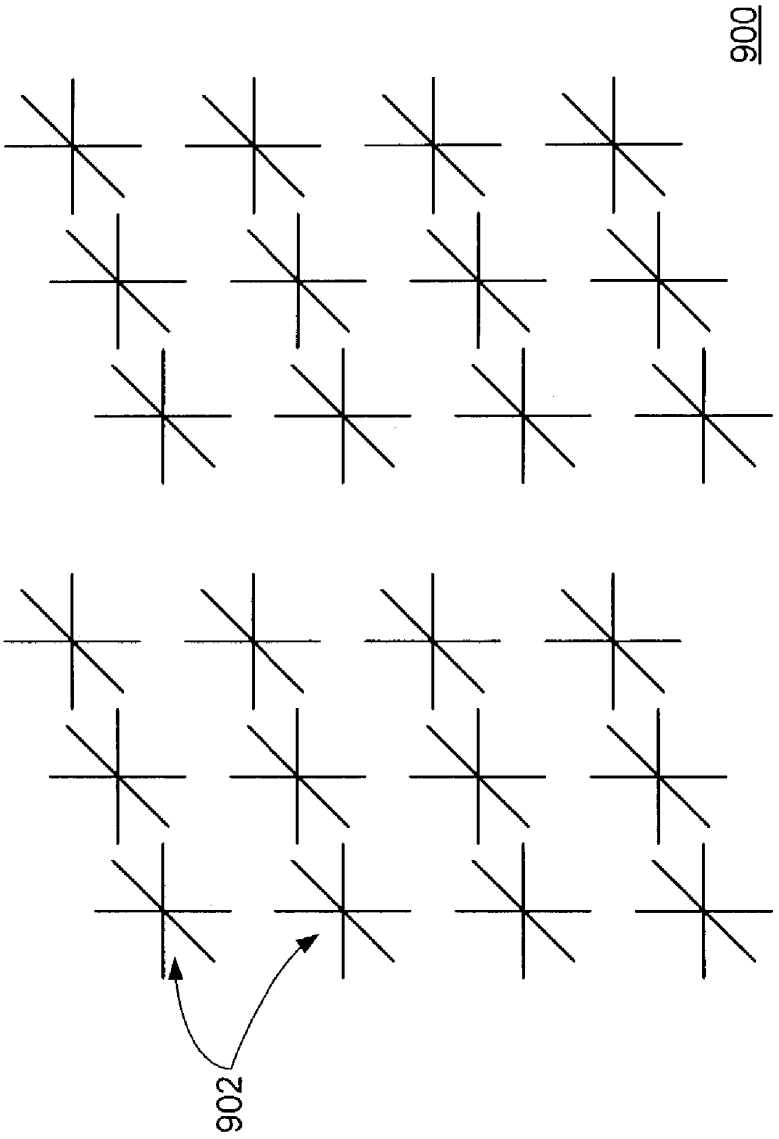

812
i = 1,I
814
k = 1,m
M[i,k]
D[i,k]
I[i,k]
VALUES USED IN THE COMPUTE
M[i-1,k-1]
M[i,k-1]
M[i-1,k]
I[i-1,k-1]
D[i,k-1]
I[i-1,k]
D[i-1,k-1]
XB[i-1]
COMPUTE VALUE FOR iTH COLUMN
XB[i]
816
810

Fig. 8B


818_0
M[i,0] I[i,0] D[i,0] XB[i-1]
M[i,1] I[i,1] D[i,1] XB[i-1]
M[i,2] I[i,2] D[i,2] XB[i-1]
M[i,m] I[i,m] D[i,m] XB[i-1]
XB[i]
TIME
818_1
M[i+1,0] I[i+1,0] D[i+1,0] XB[i-1]
M[i+1,1] I[i+1,1] D[i+1,1] XB[i-1]
M[i+1,m] I[i+1,m] D[i+1,m] XB[i-1]
XB[i+1]
PERFORM TEST: IS XB[i+1] = XB[i]?
820_1
818_j
M[i+j,0] I[i+j,0] D[i+j,0] XB[i-1]
M[i+j,m] I[i+j,m] D[i+j,m] XB[i-1]
XB[i+1]
PERFORM TEST: IS XB[i+1] = XB[i]?
820_j

810 *Fig. 8C*

Fig. 9A



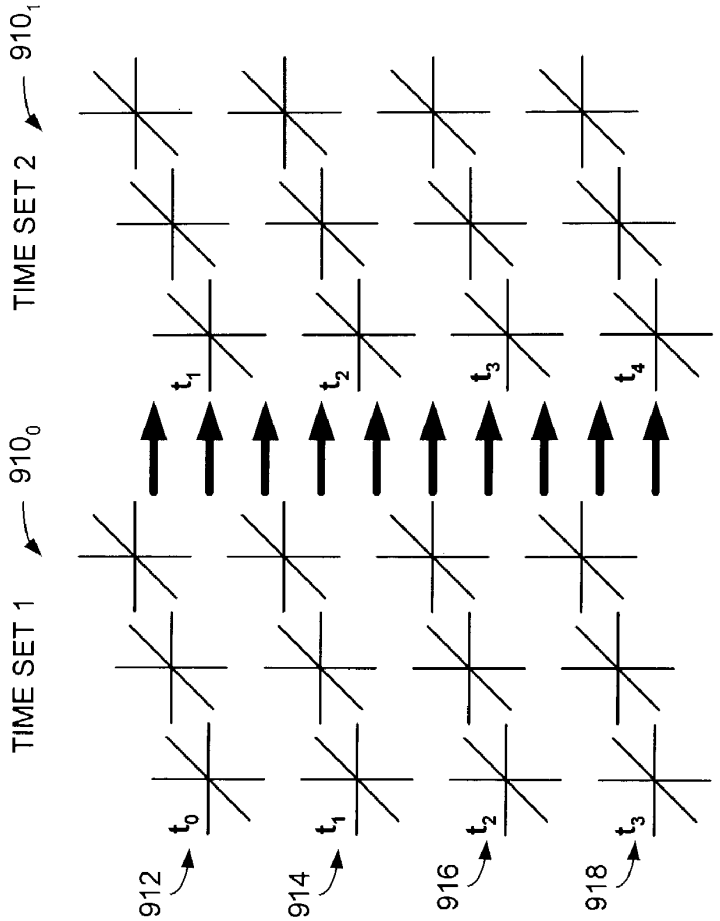

TIME SET 1
TIME SET 2
910_0
910_1
t_0
t_1
t_2
t_3
t_4
912
914
916
918

Fig. 9C

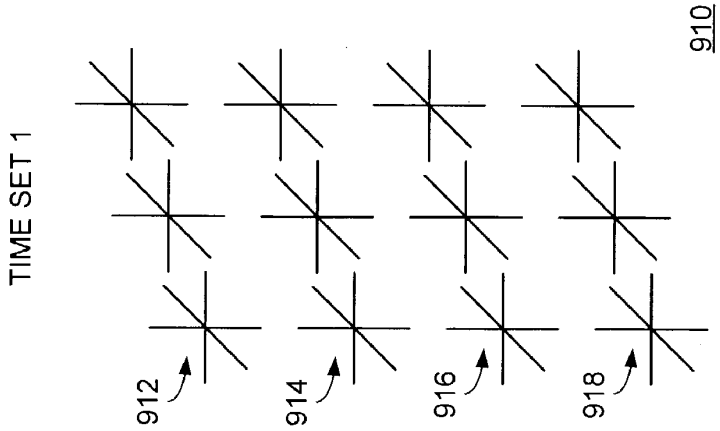

TIME SET 1
910
912
914
916
918

Fig. 9B

MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS

CROSS REFERENCE TO RELATED PATENT APPLICATIONS

The present invention is related to the subject matter of U.S. patent application Ser. No. 09/755,744 filed Jan. 5, 2001 for: "Multiprocessor Computer Architecture Incorporating a Plurality of Memory Algorithm Processors in the Memory Subsystem" and is further related to the subject matter of U.S. Pat. No. 6,434,687 for: "System and Method for Accelerating Web Site Access and Processing Utilizing a Computer System Incorporating Reconfigurable Processors Operating Under a Single Operating System Image", all of which are assigned to SRC Computers, Inc., Colorado Springs, Colo. and the disclosures of which are herein specifically incorporated in their entirety by this reference.

COPYRIGHT NOTICE/PERMISSION

BACKGROUND OF THE INVENTION

The present invention relates, in general, to the field of computing systems and techniques. More particularly, the present invention relates to multi-adaptive processing systems and techniques for enhancing parallelism and performance of computational functions.

Currently, most large software applications achieve high performance operation through the use of parallel processing. This technique allows multiple processors to work simultaneously on the same problem to achieve a solution in a fraction of the time required for a single processor to accomplish the same result. The processors in use may be performing many copies of the same operation, or may be performing totally different operations, but in either case all processors are working simultaneously.

The use of such parallel processing has led to the proliferation of both multi-processor boards and large scale clustered systems. However, as more and more performance is required, so is more parallelism, resulting in ever larger systems. Clusters exist today that have tens of thousands of processors and can occupy football fields of space. Systems of such a large physical size present many obvious downsides, including, among other factors, facility requirements, power, heat generation and reliability.

SUMMARY OF THE INVENTION

However, if a processor technology could be employed that offers orders of magnitude more parallelism per processor, these systems could be reduced in size by a comparable factor. Such a processor or processing element is possible through the use of a reconfigurable processor.

Reconfigurable processors instantiate only the functional units needed to solve a particular application, and as a result, have available space to instantiate as many functional units as may be required to solve the problem up to the total capacity of the integrated circuit chips they employ.

At present, reconfigurable processors, such as multi-adaptive processor elements (MAP™, a trademark of SRC Computers, Inc.) can achieve two to three orders of magnitude more parallelism and performance than state-of-the-art microprocessors. Through the advantageous application of adaptive processing techniques as disclosed herein, this type of reconfigurable processing parallelism may be employed in a variety of applications resulting in significantly higher performance than that which can now be achieved while using significantly smaller and less expensive computer systems.

However, in addition to these benefits, there is an additional much less obvious one that can have even greater impact on certain applications and has only become available with the advent of multi-million gate reconfigurable chips. Performance gains are also realized by reconfigurable processors due to the much tighter coupling of the parallel functional units within each chip than can be accomplished in a microprocessor based computing system.

In a multi-processor, microprocessor-based system, each processor is allocated but a relatively small portion of the total problem called a cell. However, to solve the total problem, results of one processor are often required by many adjacent cells because their cells interact at the boundary and upwards of six or more cells, all having to interact to compute results, would not be uncommon. Consequently, intermediate results must be passed around the system in order to complete the computation of the total problem. This, of necessity, involves numerous other chips and busses that run at much slower speeds than the microprocessor thus resulting in system performance often many orders of magnitude lower than the raw computation time.

On the other hand, in the use of an adaptive processor-based system, since ten to one thousand times more computations can be performed within a single chip, any boundary data that is shared between these functional units need never leave a single integrated circuit chip. Therefore, data moving around the system, and its impact on reducing overall system performance, can also be reduced by two or three orders of magnitude. This will allow both significant improvements in performance in certain applications as well as enabling certain applications to be performed in a practical timeframe that could not previously be accomplished.

Particularly disclosed herein is a method for data processing in a reconfigurable computing system comprising a plurality of functional units. The method comprises: defining a calculation for the reconfigurable computing system; instantiating at least two of the functional units to perform the calculation; utilizing a first of the functional units to operate upon a subsequent data dimension of the calculation and substantially concurrently utilizing a second of the functional units to operate upon a previous data dimension of the calculation.

Further disclosed herein is a method for data processing in a reconfigurable computing system comprising a plurality of functional units. The method comprises: defining a first systolic wall comprising rows of cells forming a subset of the plurality of functional units; computing a value at each of the cells in at least a first row of the first systolic wall; communicating the values between cells in the first row of the cells to produce updated values; communicating the updated values to a second row of the first systolic wall; and

substantially concurrently providing the updated values to a first row of a second systolic wall of rows of cells in the subset of the plurality of functional units.

Also disclosed herein is a method for data processing in a reconfigurable processing system which includes setting up a systolic processing form employing a speculative processing strategy.

BRIEF DESCRIPTION OF THE DRAWINGS

The aforementioned and other features and objects of the present invention and the manner of attaining them will become more apparent and the invention itself will be best understood by reference to the following description of a preferred embodiment taken in conjunction with the accompanying drawings, wherein:

FIG. **1** is a simplified functional block diagram of typical clustered inter-processor communications path in a conventional multi-processor computing system;

FIG. **2** is a functional block diagram of an adaptive processor communications path illustrating the many functional units ("FU") interconnected by reconfigurable routing resources within the adaptive processor chip;

FIG. **3**A is a graph of the actual performance improvement versus the number of processors utilized and illustrating the deviation from perfect scalability of a particular application utilizing a conventional multi-processor computing system such as that illustrated in FIG. **1**;

FIG. **3**B is a corresponding graph of the actual performance improvement versus the number of processors utilized and illustrating the performance improvement over a conventional multi-processor computing system utilizing an adaptive processor-based computing system such as that illustrated in FIG. **2**;

FIG. **4**A is a simplified logic flowchart illustrating a conventional sequential processing operation in which nested Loops A and B are alternately active on different phases of the process;

FIG. **4**B is a comparative, simplified logic flowchart illustrating multi-dimensional processing in accordance with the technique of the present invention wherein multiple dimensions of data are processed by both Loops A and B such that the computing system logic is operative on every clock cycle;

FIG. **5**A is illustrative of a general process for performing a representative multi-dimensional pipeline operation in the form of a seismic migration imaging function utilizing the parallelism available in the utilization of the adaptive processing techniques of the present invention;

FIG. **5**B is a follow-on illustration of the computation phases employed in implementing the exemplary seismic migration imaging function of the preceding figure;

FIG. **6**A is a simplified logic flowchart for a particular seismic migration imaging application illustrative of the parallelism provided in the use of an adaptive processor-based computing system;

FIG. **6**B illustrates the computational process which may be employed by a microprocessor in the execution of the seismic imaging application of the preceding figure;

FIG. **6**C illustrates the first step in the computational process which may be employed by an adaptive processor in the execution of the seismic imaging application of FIG. **6**A in which a first shot (S**1**) is started;

FIG. **6**D illustrates the second step in the same computational process for the execution of the seismic imaging application of FIG. **6**A in which a second shot (S**2**) is started;

FIG. **6**E illustrates the third step in the same computational process for the execution of the seismic imaging application of FIG. **6**A in which the operation on the first and second shots is continued through compute;

FIG. **6**F illustrates the fourth step in the same computational process showing the subsequent operation on shots S**1** and S**2**;

FIG. **6**G illustrates the fifth step in the same computational process followed by the continued downward propagation of shots S**1** and S**2** over all of the depth slices;

FIG. **7**A illustrates a process for performing a representative systolic wavefront operation in the form of a reservoir simulation function also utilizing the parallelism available in the utilization of the adaptive processing techniques of the present invention;

FIG. **7**B illustrates the general computation of fluid flow properties in the reservoir simulation of the preceding figure which are communicated to neighboring cells;

FIG. **7**C illustrates the creation of a systolic wall of computation at Time Set **1** which has been started for a vertical wall of cells and in which communication of values between adjacent rows in the vertical wall can occur without storing values to memory;

FIG. **7**D is a follow on illustration of the creation of a systolic wall of computation at Time Set **1** and Time Set **2** showing how a second vertical wall of cells is started after the computation for cells in the corresponding row of the first wall has been completed;

FIG. **8**A illustrates yet another process for performing a representative systolic wavefront operation in the form of the systolic processing of bioinformatics also utilizing the parallelism available in the utilization of the adaptive processing techniques of the present invention;

FIG. **8**B illustrates a systolic wavefront processing operation which further incorporates a speculative processing strategy based upon an evaluation of the rate of change of XB;

FIG. **8**C is a further illustration of the systolic wavefront processing operation of the preceding figure incorporating speculative processing;

FIG. **9**A illustrates still another process for performing a representative systolic wavefront operation in the form of structure codes calculating polynomials at grid intersections, again utilizing the parallelism available in the utilization of the adaptive processing techniques of the present invention;

FIG. **9**B illustrates the computation start for a vertical wall of grid points at Time Set **1** for a polynomial evaluation performed on grid intersections wherein calculations between rows are done in a stochastic fashion using values from a previous row; and

FIG. **9**C is a further illustration of the polynomial evaluation performed on grid intersections of the preceding figure wherein a second wall is started after the cells in the corresponding row of the first wall have been completed.

DESCRIPTION OF A REPRESENTATIVE EMBODIMENT

This application incorporates by reference the entire disclosure of Caliga, D. et al. "Delivering Acceleration: "The Potential for Increased HPC Application Performance Using Reconfigurable Logic", SC2001, November 2001, ACM 1-58113-293-X/01/0011.

With reference now to FIG. **1**, a simplified functional block diagram of typical clustered inter-processor communications path in a conventional multi-processor computing system **100** is shown. The computer system comprises a

number of memory and input/output ("I/O" controller integrated circuits ("ICs") **102**$_0$ through **102**$_N$, (e.g. "North Bridge") **102** such as the P4X333/P4X400 devices available from VIA Technologies, Inc.; the M1647 device available from Acer Labs, Inc. and the 824430X device available from Intel Corporation. The North Bridge IC **102** is coupled by means of a Front Side Bus ("FSB") to one or more microprocessors **104**$_{00}$ though **104**$_{03}$ and **104**$_{N0}$ through **104**$_{N3}$ such as one of the Pentium® series of processors also available from Intel Corporation.

The North Bridge ICs **102**$_0$ through **102**$_N$ are coupled to respective blocks of memory **106**$_0$ through **106**$_N$ as well as to a corresponding I/O bridge element **108**$_0$ through **108**$_N$. A network interface card ("NIC") **110**$_0$ through **210**$_N$ couples the I/O bus of the respective I/O bridge **108**$_0$ through **108**$_N$ to a cluster bus coupled to a common clustering hub (or Ethernet Switch) **112**.

Since typically a maximum of four microprocessors **104**, each with two or four functional units, can reside on a single Front Side Bus, any communication to more than four must pass over the Front Side Bus, inter-bridge bus, input/output ("I/O") bus, cluster interconnect (e.g. an Ethernet clustering hub **112**) and then back again to the receiving processor **104**. The I/O bus is typically an order of magnitude lower in bandwidth than the Front Side Bus, which means that any processing involving more than the four processors **104** will be significantly throttled by the loose coupling caused by the interconnect. All of this is eliminated with a reconfigurable processor having hundreds or thousands of functional units per processor.

With reference additionally now to FIG. **2**, a functional block diagram of an adaptive processor **200** communications path for implementing the technique of the present invention is shown. The adaptive processor **200** includes an adaptive processor chip **202** incorporates a large number of functional units ("FU") **204** interconnected by reconfigurable routing resources. The adaptive processor chip **202** is coupled to a memory element **206** as well as an interconnect **208** and a number of additional adaptive processor chips **210**.

As shown, each adaptive processor chip **202** can contain thousands of functional units **204** dedicated to the particular problem at hand. Interconnect between these functional units is created by reconfigurable routing resources inside each chip **202**. As a result, the functional units **204** can share or exchange data at much higher data rates and lower latencies than a standard microprocessor **104** (FIG. **1**). In addition, the adaptive processor chips **202** can connect directly to the inter-processor interconnect **208** and do not require the data to be passed through multiple chips in a chipset in order to communicate. This is because the adaptive processor can implement whatever kind of interface is needed to accomplish this connection.

With reference additionally now to FIG. **3**A, a graph of the actual performance improvement versus the number of processors utilized in a conventional multi-processor computing system **100** (FIG. **1**) is shown. In this figure, the deviation from perfect scalability of a particular application is illustrated for such a system.

With reference additionally now to FIG. **3**B, a corresponding graph of the actual performance improvement versus the number of processors utilized in an adaptive processor-based computing system **200** (FIG. **2**) is shown. In this figure, the performance improvement provided with an adaptive processor-based computing system **200** over that of a conventional multi-processor computing system **100** is illustrated.

With reference additionally now to FIG. **4**A, a simplified logic flowchart is provided illustrating a conventional sequential processing operation **400** in which nested Loops A (first loop **402**) and B (second loop **404**) are alternately active on different phases of the process.

As shown, the standard implementation of applications that have a set of nested loops **402**,**404** is to complete the processing of the first loop **402** before proceeding to the second loop **404**. The problem inherent in this approach, particularly when utilized in conjunction with field programmable gate arrays ("FPGAs") is that all of the logic that has been instantiated is not being completely utilized.

With reference additionally now to FIG. **4**B, a comparative, simplified logic flowchart is shown illustrating a multi-dimensional process **410** in accordance with the technique of the present invention. The multi-dimensional process **410** is effectuated such that multiple dimensions of data are processed by both Loops A (first loop **412**) and B (second loop **414**) such that the computing system logic is operative on every clock cycle.

In contrast to the sequential processing operation **400** (FIG. **4**A) the solution to the problem of most effectively utilizing available resources is to have an application evaluate a problem in a data flow sense. That is, it will "pass" a subsequent dimension of a given problem through the first loop **412** of logic concurrently with the previous dimension of data being processed through the second loop **414**. In practice, a "dimension" of data can be: multiple vectors of a problem, multiple planes of a problem, multiple time steps in a problem and so forth.

With reference additionally now to FIG. **5**A, a general process for performing a representative multi-dimensional pipeline operation is shown in the form of a seismic migration imaging function **500**. The process **500** can be adapted to utilize the parallelism available in the utilization of the adaptive processing techniques of the present invention in the form of a multi-adaptive processor (MAP™, a trademark of SRC Computers, Inc., assignee of the present invention) STEP**3**d routine **502**. The MAP STEP**3**d routine **502** is operation to utilize velocity data **504**, source data **506** and receiver data **508** to produce a resultant image **510** as will be more fully described hereinafter.

With reference additionally now to FIG. **5**B, the MAP STEP**3**d routine **502** of the preceding figure is shown in the various computational phases of: MAPTRI_x **520**, MAPTRI_y **522**, MAPTRI_d+ **524** and MAPTRI_d− **526**.

With reference additionally now to FIG. **6**A, a simplified logic flowchart for a particular seismic migration imaging application **600** is shown. The seismic migration imaging application **600** is illustrative of the parallelism provided in the use of an adaptive processor-based computing system **200** such as that shown in FIG. **2**. The representative application **600** demonstrates a nested loop parallelism in the tri-diagonal solver and the same logic can be implemented for the multiple tri-diagonal solvers in the x, y, d+ and d− directions. The computational phases of: MAPTRI_x **520**, MAPTRI_y **522**, MAPTRI_d+ **524** and MAPTRI_d− **526** are again illustrated.

With reference additionally now to FIG. **6**B, a computational process **610** is shown which may be employed by a microprocessor ("mP") in the execution of the seismic imaging application **600** of the preceding figure. The process **610** includes the step **612** of reading the source field [$S(Z_0)$] and receiver field [$R(Z_0)$] as well as the velocity field [$V(Z_0)$] at step **614**. At step **616** values are computed for $S(Z_{nz})$, $R(Z_{nz})$ which step is followed by the phases MAPTRI_x **520** and MAPTRI_y **522**. At step **618**, the image of

$Z_{1/2}$ is computed. This is followed by the phases MAPTRI_d+ **524** and MAPTRI_d− **526** to produce the resultant image Z at step **620**. The process **610** loops over the depth slices as indicated by reference number **622** and loops over the shots as indicated by reference number **624**.

With reference additionally now to FIG. **6**C, the first step in a computational process **650** in accordance with the technique of the present invention is shown in which a first shot (S**1**) is started. The process **650** may be employed by an adaptive processor (e.g. a MAP™ adaptive processor) as disclosed herein in the execution of the seismic imaging application **600** of FIG. **6**A. As indicated by the shaded block, the phase MAPTRI_x **520** is active.

With reference additionally now to FIG. **6**D, the second step in the computational process **650** is shown at a point at which a second shot (S**2**) is started. Again, as indicated by the shaded blocks, the phase MAPTRI_x **520** is active for S**2**, the phase MAPTRI_y **522** is active for S**1** and image $Z_{1/2}$ has been produced at step **618**. As shown, adaptive processors in accordance with the disclosure of the present invention support computation pipelining in multiple dimensions and the parallelism in Z and shots is shown at step **612**.

With reference additionally now to FIG. **6**E, the third step in the computational process **650** is shown in which the operation on the first and second shots is continued through compute. As indicated by the shaded blocks, the phase MAPTRI_d+ **524** is active for S**1**, the phase MAPTRI_y **522** is active for S**2** and image $Z_{1/2}$ has been produced at step **618**.

With reference additionally now to FIG. **6**F, the fourth step in the computational process **650** is shown illustrating the subsequent operation on shots S**1** and S**2**. The phase MAPTRI_d+ **524** is active for S**2**, the phase MAPTRI_d− **526** is active for S**1** and image Z has been produced at step **620**.

With reference additionally now to FIG. **6**G, the fifth step in the computational process **650** is shown as followed by the continued downward propagation of shots S**1** and S**2** over all of the depth slices. The phase MAPTRI_x **520** is active for S**1**, the phase MAPTRI_d− **526** is active for S**2** and image Z has been produced at step **620**.

With reference additionally now to FIG. **7**A, a process **700** for performing a representative systolic wavefront operation in the form of a reservoir simulation function is shown which utilizes the parallelism available in the adaptive processing techniques of the present invention. The process **700** includes a "k" loop **702**, "j" loop **704** and "i" loop **706** as shown.

With reference additionally now to FIG. **7**B, the general computation of fluid flow properties in the reservoir simulation process **700** of the preceding figure are illustrated as values are communicated between a group of neighboring cells **710**. The group of neighboring cells **710** comprises, in the simplified illustration shown, first, second and third walls of cells **712**, **714** and **716** respectively. Each of the walls of cells includes a corresponding number of first, second, third and fourth rows **718**, **720**, **722** and **724** respectively.

As shown, the computation of fluid flow properties are communicated to neighboring cells **710** and, importantly, this computation can be scheduled to eliminate the need for data storage. In accordance with the technique of the present invention, a set of cells can reside in an adaptive processor and the pipeline of computation can extend across multiple adaptive processors. Communication overhead between multiple adaptive processors may be advantageously minimized through the use of MAP™ adaptive processor chain ports as disclosed in U.S. Pat. No. 6,339,819 issued on Jan. 15, 2002 for: "Multiprocessor With Each Processor Element Accessing Operands in Loaded Input Buffer and Forwarding Results to FIFO Output Buffer", assigned to SRC Computers, Inc., assignee of the present invention, the disclosure of which is herein specifically incorporated by this reference.

With reference additionally now to FIG. **7**C, the creation of a systolic wall **712** of computation at Time Set **1** is shown. The systolic wall **712** has been started for a vertical wall of cells and communication of values between adjacent rows **718** through **724** in the vertical wall can occur without storing values to memory.

With reference additionally now to FIG. **7**D, a follow on illustration of the creation of a systolic wall **712** of computation at Time Set **1** and a second systolic wall **714** at Time Set **2** is shown. In operation, a second vertical wall of cells is started after the computation for cells in the corresponding row of the first wall has been completed. Thus, for example, at time t_0, the first row **718** of systolic wall **712** is completed and the results passed to the first row **718** of the second systolic wall **714**. At time t_1, the second row **720** of the first systolic wall **712** and the first row **718** of the second systolic wall **714** are computed. Thereafter, at time t_2, the third row **722** of the first systolic wall **712** and the second row **720** of the second systolic wall **714** are computed. The process continues in this manner for all rows and all walls.

With reference additionally now to FIG. **8**A, yet another process **800** for performing a representative systolic wavefront operation is shown. The process **800** is in the form of the systolic processing of bioinformatics and also utilizes the parallelism available in the adaptive processing techniques of the present invention. As shown, systolic processing in the process **800** can pass previously computed data down within a column (e.g. one of columns **802**, **804** and **806**) as to subsequent columns as well (e.g. from column **802** to **804**; from column **804** to **806** etc.) The computational advantage provided is the processing of the second column **804** can begin after only a few clock cycles following the start of the processing of the first column **802** to compute the first "match" state.

With reference additionally now to FIG. **8**B, a systolic wavefront processing operation **810** is shown. The processing operation **810**, comprising "i" loop **812** and "k" loop **814** now further incorporates a speculative processing strategy based upon an evaluation of the rate of change of XB.

A straightforward systolic processing operation could be used for performing the operation **810** but for the problem inherent in the computation of XB as its value XB[i] **816** can not be known until the completion of the entire "k" loop **814**. After evaluating the rate of change of XB, it was determined that a speculative processing strategy could be used for the problem. A normal systolic form is set up and the value of XB is held constant for the set of columns computed in the systolic set. At the bottom of each column, the value of XB[i] **816** is then computed.

With reference additionally now to FIG. **8**C, a further illustration of the systolic wavefront processing operation **810** incorporating speculative processing of the preceding figure is shown. The speculative processing includes "j" columns $\mathbf{818}_0$ through $\mathbf{818}_j$ as shown. Each of the columns **818** assumes that XB[i+j] has a constant value. A test is conducted at the bottom of each of the columns **818** to determine with the XB value changes as indicated at steps $\mathbf{820}_1$, through $\mathbf{820}_j$. If the value of XB changes at the i+n column, the process is then restarted at that column **818**. Since the rate of change of XB is relatively slow, the "cost" of the compute operation can be greatly reduced.

With reference additionally now to FIG. **9**A, another process **900** for performing a representative systolic wavefront operation is shown in the form of structure codes calculating polynomials at grid intersections **902**. The process **900** advantageously utilizes the parallelism available in the adaptive processing techniques of the present invention.

With reference additionally now to FIGS. **9**B and **9**C, the computation start for a vertical wall **910** of grid points at Time Set **1** is shown for a polynomial evaluation performed on grid intersections **902** (FIG. **9**A) wherein calculations between rows **912**, **914**, **016** and **918** are done in a stochastic fashion using values from a previous row. As shown, a polynomial evaluation is performed on the grid intersections **902** such that a second wall $\mathbf{910}_1$ is started after the cells in the corresponding row of the first wall $\mathbf{910}_0$ have been completed.

As can be determined from the foregoing, the multi-adaptive processing systems and techniques for enhancing parallelism and performance of computational functions disclosed herein can be employed in a myriad of applications including multi-dimensional pipeline computations for seismic applications, search algorithms, information security, chemical and biological applications, filtering and the like as well as for systolic wavefront computations for fluid flow and structures analysis, bioinformatics etc. Some applications may also employ both the multi-dimensional pipeline and systolic wavefront methodologies.

Following are representative applications of the techniques for adaptive processor based computation disclosed herein:

Imaging

Seismic: These applications, typically used in the oil and gas exploration industries, process echo data to produce detailed analysis of subsurface features. The applications use data collected at numerous points and consisting of many repeated parameters. Due to this, these programs are ideal candidates to take advantage of parallel computing. In addition, because the results of the computation on one data point are used in the computation of the next, these programs will particularly benefit from the tight parallelism that can be found in the use of adaptive or reconfigurable processors.

Synthetic Aperture Radar ("SAR"): These applications are typically used in geographical imaging. The applications use data collected in swaths. Processing consists of repeated operations on data that has been sectioned in cells. These programs are also ideal candidates to take advantage of parallel computing and in particular to benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

JPEG Image compression: These applications partition an image into numerous blocks. These blocks then have a set of operations performed on them. The operations can be parallelized across numerous blocks. The combination of the set of operations and the parallelism will particularly benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

MPEG Image compression: These applications partition a frame into numerous blocks. These blocks then have a set of operations performed on them. The operations can be parallelized across numerous blocks. In addition, there are numerous operations that are performed on adjacent frames. The combination of the set of operations and the parallelism will particularly benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Fluid Flow

Reservoir Simulation: These applications, also typically used in the oil and gas production industries, process fluid flow data in the oil and gas subsurface reservoirs to produce extraction models. The application will define a three dimensional ("3d") set of cells that contain the oil and gas reservoir. These programs are ideal candidates to take advantage of parallel or adaptive computing because there are repeated operations on each cell. In addition, information computed for each cell is then passed to neighboring cells. These programs will particularly benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Weather prediction: Such an application will partition the forecast area into logical grid cells. The computational algorithms will then perform calculations that have polynomials that have nodes associated with the grid cells. These programs are ideal candidates to take advantage of adaptive or parallel computing because there are repeated operations on each cell associated with the set of times computed in the forecast.

Automotive: These applications investigate the aerodynamics of automobile or other aerodynamic structures. The application generally divides the space surrounding the automobile structure into logical cells that are associated with nodes in computational polynomials. These programs are ideal candidates to take advantage of adaptive or parallel computing because there are repeated operations on each cell associated with the set of wind velocities computed in the forecast. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Aerospace: These applications investigate the aerodynamics of aerospace/airplane structures. The application divides the space surrounding the aerospace/airplane structure into logical cells that are associated with nodes in computational polynomials. These programs are ideal candidates to take advantage of parallel computing because there are repeated operations on each cell associated with the set of wind velocities computed in the forecast. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Plastic Injection Molding: These applications investigate the molding parameters of injecting liquid plastic into molds. The application divides the space inside the mold into logical cells that are also associated with nodes in computational polynomials. These programs are ideal candidates to take advantage of parallel computing because there are repeated operations on each cell associated with the set of injection parameters. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Structures

Crash Analysis: These applications are typically used in the automotive or aviation industry. The application will partition the entire automobile into components. These components are then subdivided into cells. The application will analyze the effect of a collision on the structure of the automobile. These programs are ideal candidates for parallel computing because there are repeated operations on each cell and they receive computed information from their neighboring cells. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Structural Analysis: These applications investigate the properties of structural integrity. The application divides the

structure into logical cells that are associated with nodes in computational polynomials. These programs are ideal candidates to take advantage of parallel computing because there are repeated operations on each cell associated with load and stress. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Search Algorithms

Image searches: These applications are typically used in the security industry for fingerprint matching, facial recognition and the like. The application seeks matches in either a collection of subsets of the total image or the total image itself. The process compares pixels of the model to pixels of a record from an image database. These programs are ideal candidates for parallel computing because of the correlation of comparison results that exist for each pixel in the subsets or entire image. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Data mining: These applications are typically used in commercial market spaces. The application seeks matches in a set of search information (e.g. character strings) in each record in a database. The application then produces a match correlation for all data records. A match correlation is produced from the comparison results for each set of search information with all characters in a database record. These programs are ideal candidates for parallel computing because of the repeated comparison operations that exist all character comparisons of the set of search information with each character in the database record. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Finance

Financial modeling: The application creates numerous strategies for each decision step in the modeling process. The results of a computational step are feed into another set of strategies for subsequence modeling steps. These programs are ideal candidates to take advantage of parallel computing because there are repeated operations on each strategy within a modeling step. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Information Security

Encryption/Decryption: The application applies an algorithm that converts the original data into an encrypted, or "protected", form. The process is applied to each set of N bits in the original data. Decryption reverses the process to deliver the original data. These programs are ideal candidates for parallel computing because there are repeated operations on each N bits of data. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Chemistry/Biology

Genetic pattern matching: These applications are typically used in the bioinformatics industry. The application looks for matches of a particular genetic sequence (or model) to a database of genetic records. The application compares each character in the model to the characters in genetic record. These programs are ideal candidates for parallel computing because of the repeated comparison operations that exist for all character comparisons of the model with each character in the genetic record. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Protein Folding: These applications are typically used by pharmaceutical companies. The application investigates the dynamics of the deformation of the protein structure. The application uses a set of equations which are recomputed at various "time" intervals to model the protein folding. These programs are ideal candidates for parallel computing because of the repeated computations on a large set of time intervals in the modeling sequence. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors

Organic structure interaction: These applications are typically used by chemical and drug companies. The application investigates the dynamics of organic structures as they are interacting. The application uses a set of equations which are recomputed at various "time" intervals to model how the organic structure interact. These programs are ideal candidates for parallel computing because of the repeated computations on a large set of time intervals in the modeling sequence. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors

Signals

Filtering: Applications often utilize filtering techniques to "clean-up" a recorded data sequence. This technique is utilized in a wide variety of industries. The application generally applies a set of filter coefficients to each data point in the recorded sequence. These programs are ideal candidates for parallel computing because of the repeated computations to all data points in the sequence and all sequences. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

While there have been described above the principles of the present invention in conjunction with specific, exemplary applications for the use of adaptive processor-based systems in the implementation of multi-dimensional pipeline and systolic wavefront computations, it is to be clearly understood that the foregoing descriptions are made only by way of example and not as a limitation to the scope of the invention. Particularly, it is recognized that the teachings of the foregoing disclosure will suggest other modifications to those persons skilled in the relevant art. Such modifications may involve other features which are already known per se and which may be used instead of or in addition to features already described herein. Although claims have been formulated in this application to particular combinations of features, it should be understood that the scope of the disclosure herein also includes any novel feature or any novel combination of features. disclosed either explicitly or implicitly or any generalization or modification thereof which would be apparent to persons skilled in the relevant art, whether or not such relates to the same invention as presently claimed in any claim and whether or not it mitigates any or all of the same technical problems as confronted by the present invention. The applicants hereby reserve the right to formulate new claims to such features and/or combinations of such features during the prosecution of the present application or of any further application derived therefrom.

The invention claimed is:

1. A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor, the reconfigurable processor comprising a plurality of functional units, said method comprising:

transforming an algorithm into a calculation that is systolically implemented by said reconfigurable computing system at the at least one reconfigurable processor;

instantiating at least two of said functional units at the at least one reconfigurable processor to perform said calculation wherein only functional units needed to solve the calculation are instantiated and wherein each instantiated functional unit at the at least one reconfig-

urable processor interconnects with each other instantiated functional unit at the at least one reconfigurable processor based on reconfigurable routing resources within the at least one reconfigurable processor as established at instantiation, and wherein systolically linked lines of code of said calculation are instantiated as clusters of functional units within the at least one reconfigurable processor;

utilizing a first of said instantiated functional units to operate upon a subsequent data dimension of said calculation forming a first computational loop; and

substantially concurrently utilizing a second of said instantiated functional units to operate upon a previous data dimension of said calculation forming a second computational loop wherein said systolic implementation of said calculation enables said first computational loop and said second computational loop execute concurrently and pass computed data seamlessly between said computational loops.

2. The method of claim **1** wherein said subsequent and previous data dimensions of said calculation comprise multiple vectors in said calculation.

3. The method of claim **1** wherein said subsequent and previous data dimensions of said calculation comprise multiple planes in said calculation.

4. The method of claim **1** wherein said subsequent and previous data dimensions of said calculation comprise multiple time steps in said calculation.

5. The method of claim **1** wherein said subsequent an previous data dimensions of said calculation comprise multiple grid points in said calculation.

6. The method of claim **1** wherein said calculation comprises a seismic imaging calculation.

7. The method of claim **1** wherein said calculation comprises a synthetic aperture radar imaging calculation.

8. The method of claim **1** wherein said calculation comprises a JPEG image compression calculation.

9. The method of claim **1** wherein said calculation comprises an MPEG image compression calculation.

10. The method of claim **1** wherein said calculation comprises a fluid flow calculation for a reservoir simulation.

11. The method of claim **1** wherein said calculation comprises a fluid flow calculation for weather prediction.

12. The method of claim **1** wherein said calculation comprises a fluid flow calculation for automotive applications.

13. The method of claim **1** wherein said calculation comprises a fluid flow calculation for aerospace applications.

14. The method of claim **1** wherein said calculation comprises a fluid flow calculation for an injection molding application.

15. The method of claim **1** wherein instantiating includes establishing a stream communication connection between functional units.

16. The method of claim **1** wherein said calculation is comprises a structures calculation for structural analysis.

17. The method of claim **1** wherein said calculation comprises a search algorithm for an image search.

18. The method of claim **1** wherein said calculation comprises a search algorithm for data mining.

19. The method of claim **1** wherein said calculation comprises a financial modeling application.

20. The method of claim **1** wherein said calculation comprises an encryption algorithm.

21. The method of claim **1** wherein said calculation comprises a genetic pattern matching function.

22. The method of claim **1** wherein said calculation comprises a protein folding function.

23. The method of claim **1** wherein said calculation comprises an organic structure interaction function.

24. The method of claim **1** wherein said calculation comprises a signal filtering application.

25. A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor comprising a plurality of functional units, said method comprising:

transforming an algorithm into a calculation that is systolically implemented by said reconfigurable computing system at the at least one reconfigurable processor wherein systolically linked lines of code of said calculation are instantiated as walls of functional units within the at least one reconfigurable processor;

defining a first systolic wall comprising rows of cells forming a subset of said plurality of functional units;

computing at the at least one reconfigurable processor a value at each of said cells in at least a first row of said first systolic wall substantially concurrently;

communicating said values between cells in said first row of said cells to produce updated values, wherein communicating said values is based on reconfigurable routing resources within the at least one reconfigurable processor;

communicating said updated values substantially concurrently to a second row of said first systolic wall, wherein communicating said updated values is based on reconfigurable routing resources within the at least one reconfigurable processor; and

communicating said updated values substantially concurrently to a first row of a second systolic wall of rows of cells in said subset of said plurality of functional units, wherein communicating said updated values is based on reconfigurable routing resources within the at least one reconfigurable processor and wherein said first systolic wall of rows of cells and said second wall of rows of systolic cells execute substantially concurrently and pass computed data seamlessly between said systolic walls.

26. The method of claim **25** wherein said values correspond to vectors in a computation.

27. The method of claim **25** wherein said values correspond to planes in a computation.

28. The method of claim **25** wherein said values correspond to time steps in a computation.

29. The method of claim **25** wherein said values correspond to grid points in a computation.

30. The method of claim **25** wherein said step of communicating said updated values to a second row of said first systolic wall is carried out without storing said updated values in an extrinsic memory.

31. The method of claim **25** wherein said values correspond to a seismic imaging calculation.

32. The method of claim **25** wherein said values correspond to a synthetic aperture radar imaging calculation.

33. The method of claim **25** wherein said values correspond to a JPEG image compression calculation.

34. The method of claim **25** wherein said values correspond to an MPEG image compression calculation.

35. The method of claim **25** wherein said values correspond to a fluid flow calculation for a reservoir simulation.

36. The method of claim **25** wherein said values correspond to a fluid flow calculation for weather prediction.

37. The method of claim **25** wherein said values correspond to a fluid flow calculation for automotive applications.

38. The method of claim **25** wherein said values correspond to a fluid flow calculation for aerospace applications.

39. The method of claim **25** wherein said values correspond to a fluid flow calculation for an injection molding application.

40. The method of claim **25** wherein defining includes establishing a stream communication connection between functional units and wherein only functional units needed to solve the calculations are instantiated.

41. The method of claim **25** wherein said values correspond to a structures calculation for structural analysis.

42. The method of claim **25** wherein said values correspond to a search algorithm for an image search.

43. The method of claim **25** wherein said values correspond to a search algorithm for data mining.

44. The method of claim **25** wherein said values correspond to a financial modeling application.

45. The method of claim **25** wherein said values correspond to an encryption algorithm.

46. The method of claim **25** wherein said values correspond to a genetic pattern matching function.

47. The method of claim **25** wherein said values correspond to a protein folding function.

48. The method of claim **25** wherein said values correspond to an organic structure interaction function.

49. The method of claim **25** wherein said values correspond to a signal filtering application.

50. The method of claim **25** wherein said reconfigurable computing system comprises at least one microprocessor.

51. A method for data processing in a reconfigurable computing system, the reconfigurable computer system comprising at least one reconfigurable processor comprising a plurality of functional units, said method comprising:

transforming an algorithm into a calculation that is systolically implemented by said reconfigurable computing system at the at least one reconfigurable processor wherein systolically linked lines of code of said calculation are instantiated as subsets of said plurality of functional units within the at least one reconfigurable processor forming columns of said calculation;

performing said calculation at the at least one reconfigurable processor by said subsets of said plurality of functional units to produce computed data;

exchanging said computed data between a first column of said calculation and a next column in said calculation, wherein said exchanging is based on reconfigurable routing resources within the at least one reconfigurable processor and wherein execution of said subsets of said plurality of function units occurs concurrently and said computed data is seamlessly passed between said first column of said calculation and said second column of said calculation;

evaluating a rate of change in at least one variable for each of said columns in said calculation;

continuing said calculation when said variable does not change for a particular column of said calculation; and

restarting said calculation at said column of said calculation where said variable does change.

52. The method of claim **51** wherein how many functional units comprise the subset and functional type of each functional unit in said subset is based on the calculation.

* * * * *

EXHIBIT E

US007421524B2

(12) **United States Patent**
Huppenthal et al.

(10) **Patent No.: US 7,421,524 B2**
(45) **Date of Patent: Sep. 2, 2008**

(54) **SWITCH/NETWORK ADAPTER PORT FOR CLUSTERED COMPUTERS EMPLOYING A CHAIN OF MULTI-ADAPTIVE PROCESSORS IN A DUAL IN-LINE MEMORY MODULE FORMAT**

(75) Inventors: **Jon M. Huppenthal**, Colorado Springs, CO (US); **Thomas R. Seeman**, Colorado Springs, CO (US); **Lee A. Burton**, Divide, CO (US)

(73) Assignee: **SRC Computers, Inc.**, Colorado Springs, CO (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 387 days.

(21) Appl. No.: **10/996,016**

(22) Filed: **Nov. 23, 2004**

(65) **Prior Publication Data**

US 2005/0091434 A1 Apr. 28, 2005

Related U.S. Application Data

(60) Division of application No. 09/932,330, filed on Aug. 17, 2001, which is a continuation-in-part of application No. 09/755,744, filed on Jan. 5, 2001, now abandoned, which is a division of application No. 09/481,902, filed on Jan. 12, 2000, now Pat. No. 6,247,110, which is a continuation of application No. 08/992,763, filed on Dec. 17, 1997, now Pat. No. 6,076,152.

(30) **Foreign Application Priority Data**

Oct. 17, 2003 (JP) 2003-357726
Feb. 3, 2004 (JP) 2004-027337

(51) **Int. Cl.**
G06F 13/12 (2006.01)

(52) **U.S. Cl.** **710/72**; 710/2; 710/62; 709/250; 716/16

(58) **Field of Classification Search** 710/2, 710/62, 72; 716/16
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,763,294 A 8/1988 Fong

(Continued)

FOREIGN PATENT DOCUMENTS

EP 0 571 099 A 11/1993

OTHER PUBLICATIONS

Agarwal, A., et al., "The Raw Compiler Project", pp. 1-12, http://cag-www.lcs.mit.edu/raw, Proceedings of the Second SUIF Compiler Workshop, Aug. 21-23, 1997.

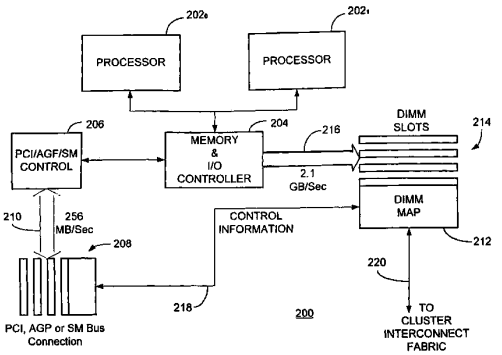
(Continued)

Primary Examiner—Alford W. Kindred
Assistant Examiner—Eron J Sorrell
(74) *Attorney, Agent, or Firm*—William J. Kubida; Michael C. Martensen; Hogan & Hartson LLP

(57) **ABSTRACT**

A switch/network adapter port ("SNAP") for clustered computers employing multi-adaptive processor ("MAP™", a trademark of SRC Computers, Inc.) elements in a dual in-line memory module ("DIMM") or Rambus™ in-line memory module ("RIMM") format to significantly enhance data transfer rates over that otherwise available through use of the standard peripheral component interconnect ("PCI") bus. Particularly disclosed is a microprocessor based computer system utilizing either a DIMM or RIMM physical format processor element for the purpose of implementing a connection to an external switch, network, or other device. In a particular embodiment, connections may be provided to either the PCI, accelerated graphics port ("AGP") or system maintenance ("SM") bus for purposes of passing control information to the host microprocessor or other control chips. The field programmable gate array ("FPGA") based processing elements have the capability to alter data passing through it to and from an external interconnect fabric or device.

15 Claims, 5 Drawing Sheets



U.S. PATENT DOCUMENTS

4,783,730	A		11/1988	Fischer
4,972,457	A		11/1990	O'Sullivan
5,230,057	A		7/1993	Shido et al.
5,265,218	A		11/1993	Testa et al.
5,295,246	A		3/1994	Bischoff et al.
5,509,134	A		4/1996	Fandrich et al.
5,570,040	A		10/1996	Lytle et al.
5,673,204	A	*	9/1997	Klingelhofer 348/578
5,737,766	A		4/1998	Tan
5,802,290	A		9/1998	Casselman
5,857,109	A	*	1/1999	Taylor 712/37
5,889,959	A		3/1999	Whittaker et al.
5,892,962	A		4/1999	Cloutier
5,903,771	A		5/1999	Sgro et al.
5,911,778	A	*	6/1999	Garnett 726/21
5,915,104	A		6/1999	Miller
5,923,682	A		7/1999	Seyyedy et al.
5,953,502	A		9/1999	Helbig, Sr.
5,966,534	A		10/1999	Cooke et al.
5,978,862	A	*	11/1999	Kou et al. 710/14
6,023,755	A		2/2000	Casselman
6,026,478	A		2/2000	Dowling et al.
6,038,431	A		3/2000	Fukutani et al.
6,047,343	A		4/2000	Olarig et al.
6,052,134	A		4/2000	Foster
6,052,773	A		4/2000	DeHon et al.
6,076,152	A		6/2000	Huppenthal et al.
6,094,532	A	*	7/2000	Acton et al. 712/28
6,108,730	A		8/2000	Dell et al.
6,148,356	A		11/2000	Archer et al.
6,192,439	B1		2/2001	Grunewald et al.
6,202,111	B1		3/2001	Wallach et al.
6,295,571	B1		9/2001	Scardamalia et al.
6,326,973	B1		12/2001	Behrbaum et al.
6,446,192	B1		9/2002	Narasimhan et al.
6,452,700	B1		9/2002	Mays, Jr.
6,480,014	B1		11/2002	Li et al.
6,577,621	B1		6/2003	Balachandran
6,581,157	B1		6/2003	Chiles et al.
6,598,199	B2		7/2003	Tetrick
6,633,945	B1		10/2003	Fu et al.
6,704,816	B1		3/2004	Burke
6,721,884	B1		4/2004	De Oliveira Kastrup Pereira et al.
6,799,252	B1		9/2004	Bauman
6,810,121	B1		10/2004	Ikesue
2001/0010057	A1		7/2001	Yamada
2004/0030816	A1		2/2004	Knight et al.

OTHER PUBLICATIONS

Albaharna, Osama, et al., "On the viability of FPGA-based integrated coprocessors", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 206-215.

Amerson, Rick, et al., "Teramac—Configurable Custom Computing", © 1995 IEEE, Publ. No. 0-8186-7086-X/95, pp. 32-38.

Barthel, Dominique Aug. 25-26, 1997, "PVP a Parallel Video coProcessor", Hot Chips IX, pp. 203-210.

Bertin, Patrice, et al., "Programmable active memories: a performance assessment", © 1993 Massachusetts Institute of Technology, pp. 88-102.

Bittner, Ray, et al., "Computing kernels implemented with a wormhole RTR CCM", © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 98-105.

Buell, D., et al. "Splash 2: FPGAs in a Custom Computing Machine—Chapter 1—Custom Computing Machines: An Introduction", pp. 1-11, http://www.computer.org/espress/catalog/bp07413/spls-ch1.html (originally believed published in J. of Supercomputing, vol. IX, 1995, pp. 219-230.

Casselman, Steven, "Virtual Computing and The Virtual Computer", © 1993 IEEE, Publ. No. 0-8186-3890-7/93, pp. 43-48.

Chan, Pak, et al., "Architectural tradeoffs in field-programmable-device-based computing systems", © 1993 IEEE, Publ. No. 0-8186-3890-7/93, pp. 152-161.

Clark, David, et al., "Supporting FPGA microprocessors through retargetable software tools", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 195-103.

Cuccaro, Steven, et al., "The CM-2X: a hybrid CM-2/Xilink prototype", © 1993 IEEE, Publ. No. 0-8186-3890-7/93, pp. 121-130.

Culbertson, W. Bruce, et al., "Exploring architectures for volume visualization on the Teramac custom computer", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 80-88.

Culbertson, W. Bruce, et al., "Defect tolerance on the Teramac custom computer", © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 116-123.

Dehon, Andre, "DPGA-Coupled microprocessors: commodity IC for the early 21st century", © 1994 IEEE, Publ. No. 0-8186-5490-2/94, pp. 31-39.

Dehon, A., et al., "Matrix A Reconfigurable Computing Device with Configurable Instruction Distribution", Hot Chips IX, Aug. 25-26, 1997, Stanford, California, MIT Artificial Intelligence Laboratory.

Dhaussy, Philippe, et al., "Global control synthesis for an MIMD/FPGA machine", © 1994 IEEE, Publ. No. 0-8186-5490-2/94, pp. 72-81.

Elliott, Duncan, et al., "Computational Ram: a memory-SIMD hybrid and its application to DSP", © 1992 IEEE, Publ. No. 0-7803-0246-X/92, pp. 30.6.1-30.6.4.

Fortes, Jose, et al., "Systolic arrays, a survey of seven projects", © 1987 IEEE, Publ. No. 0018-9162/87/0700-0091, pp. 91-103.

Gokhale, M., et al.,, "Processing in Memory: The Terasys Massively Parallel PIM Array" © Apr. 1995, IEEE, pp. 23-31.

Gunther, Bernard, et al., "Assessing Document Relevance with Run-Time Reconfigurable Machines",© 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 10-17.

Hagiwara, Hiroshi, et al., "A dynamically microprogrammable computer with low-level parallelism", © 1980 IEEE, Publ. No. 0018-9340/80/07000-0577, pp. 577-594.

Hartenstein, R. W., et al. "A General Approach in System Design Integrating Reconfigurable Accelerators," http://xputers.informatik.uni-kl.de/papers/paper026-1.html, IEEE 1996 Conference, Austin, TX, Oct. 9-11, 1996.

Hartenstein, Reiner, et al., "A reconfigurable data-driven ALU for Xputers", © 1994 IEEE, Publ. No. 0-8186-5490-2/94, pp. 139-146.

Hauser, John, et al.: "GARP: a MIPS processor with a reconfigurable co-processor", © 1997 IEEE, Publ. No. 0-08186-8159-4/97, pp. 12-21.

Hayes, John, et al., "A microprocessor-based hypercube, supercomputer", © 1986 IEEE, Publ. No. 0272-1732/86/1000-0006, pp. 6-17.

Herpel, H.—J., et al., "A Reconfigurable Computer for Embedded Control Applications", © 1993 IEEE, Publ. No. 0-8186-3890-7/93, pp. 111-120.

Hogl, H., et al., "Enable++: A second generation FPGA processor", © 1995 IEEE, Publ. No. 0-8186-X/95, pp. 45-53.

King, William, et al., "Using MORRPH in an industrial machine vision system". © 1996 IEEE, Publ. No. 08186-7548-9/96, pp. 18-26.

Manohar, Swaminathan, et al., "A pragmatic approach to systolic design", © 1988 IEEE, Publ. No. CH2603-9/88/0000/0463, pp. 463-472.

Mauduit, Nicolas, et al., "Lneuro 1.0: a piece of hardware LEGO for building neural network systems," © 1992 IEEE, Publ. No. 1045-9227/92, pp. 414-422.

Mirsky, Ethan A., "Coarse-Grain Reconfigurable Computing", Massachusetts Institute of Technology, Jun. 1996.

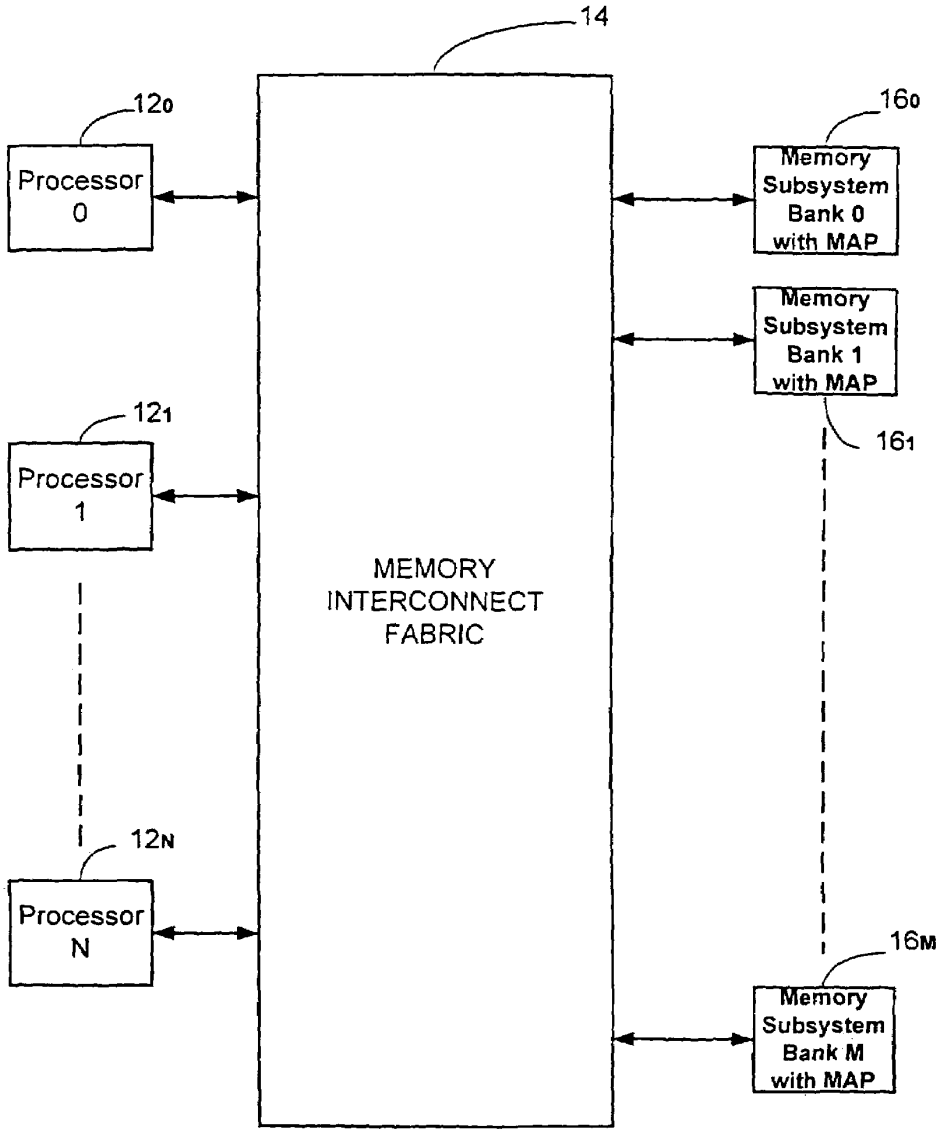
Mirsky, Ethan, et al., "Matrix: A Reconfigurable Computing Architecture with Configurable Instruction Distribution and Deployable Resources", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 157-166.

Morley, Robert E., Jr., et al., "A Massively Parallel Systolic Array Processor System", © 1988 IEEE, Publ. No. CH2603-9/88/0000/0217, pp. 217-225.

Patterson, David, et al., "A case for intelligent DRAM: IRAM", Hot Chips VIII, Aug. 19-20, 1996, pp. 75-94.

Peterson, Janes, et al., "Scheduling and partitioning ANSI-C programs onto multi-FPGA CCM architectures", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 178-187.
Schmit, Herman, "Incremental reconfiguration for pipelined applications," © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 47-55.
Sitkoff, Nathan, et al., "Implementing a Genetic Algorithm on a Parallel Custom Computing Machine", Publ. No. 0-8186-7086-X/95, pp. 180-187.
Stone, Harold, "A logic-in-memory computer", © 1970, IEEE Transactions on Computers, pp. 73-78, Jan. 1990.
Tangen, Uwe, et al., "A parallel hardware evolvable computer POLYP extended abstract", © 1997 IEEE, Publ. No. 0-8186-8159/4/97, pp. 238-239.
Thornburg, Mike, et al., "Transformable Computers", © 1994 IEEE, Publ. No. 0-8186-5602-6/94, pp. 674-679.
Tomita, Shinji, et al., "A computer low-level parallelism QA-2", © 1986 IEEE, Publ. No. 0-0384-7495/86/0000/0280, pp. 280-289.
Trimberger, Steve, et al., "A time-multiplexed FPGA", © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 22-28.
Ueda, Hirotada, et al., "A multiprocessor system utilizing enhanced DSP's for image processing", © 1998 IEEE, Publ. No. CH2603-9/88/0000/0611, pp. 611-620.
Villasensor, John, et al., "Configurable computing", © 1997 Scientific American, Jun. 1997.
Wang, Quiang, et al., "Automated field-programmable compute accelerator design using partial evaluation", © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 145-154.
W.H. Manigone-Smith and B.L. Hutchings. Configurable computing: The Road Ahead. In Proceedings of the Reconfigurable Architectures Workshop (RAW'97), pp. 81-96, 1997.
Wirthlin, Michael, et al., "The Nano processor: a low resource reconfigurable processor", © 1994 IEEE, Publ. No. 0-8186-5490-2/94, pp. 23-30.
Wirthlin, Michael, et al., "A dynamic instruction set computer", © 1995 IEEE, Publ. No. 0-8186-7086-X/95, pp. 99-107.
Wittig, Ralph, et al., "One Chip: An FPGA processor with reconfigurable logic", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 126-135.
Yamauchi, Tsukasa, et al., "SOP: A reconfigurable massively parallel system and its control-data flow based compiling method", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 148-156.
"Information Brief", PCI Bus Technology, © IBM Personal Computer Company, 1997, pp. 1-3.
Yun, Hyun-Kyu and Silverman, H. F.; "A distributed memory MIMD multi-computer with reconfigurable custom computing capabilities", Brown University, Dec. 10-13, 1997, pp. 7-13.
Vemuri, Ranga R. et al., "Configurable Computing Technology and Applications", Apr. 2000, Computer, pp. 39-40.
DeHon, Andre, "The Density Advantage of Configurable Computing", Apr. 2000, Computer, pp. 41-49.
Haynes, Simon D. et al., "Video Image Processing with the Sonic Architecture", Apr. 2000, Computer, pp. 50-57.
Platzner, Marco, "Reconfigurable Accelerators for Combinatorial Problems", Apr. 2000, Computer, pp. 58-60.
Callahan, Timothy J. et al., "The Garp Architecture and C Compiler", Apr. 2000, Computer, pp. 62-69.
Goldstein, Seth Copen et al., "PipeRench: A Reconfigurable Architecture and Compiler", Apr. 2000, Computer, pp. 70-76.

* cited by examiner


14
12₀
Processor 0
12₁
Processor 1
12N
Processor N
MEMORY INTERCONNECT FABRIC
16₀
Memory Subsystem Bank 0 with MAP
Memory Subsystem Bank 1 with MAP
16₁
16M
Memory Subsystem Bank M with MAP

10

Fig. 1

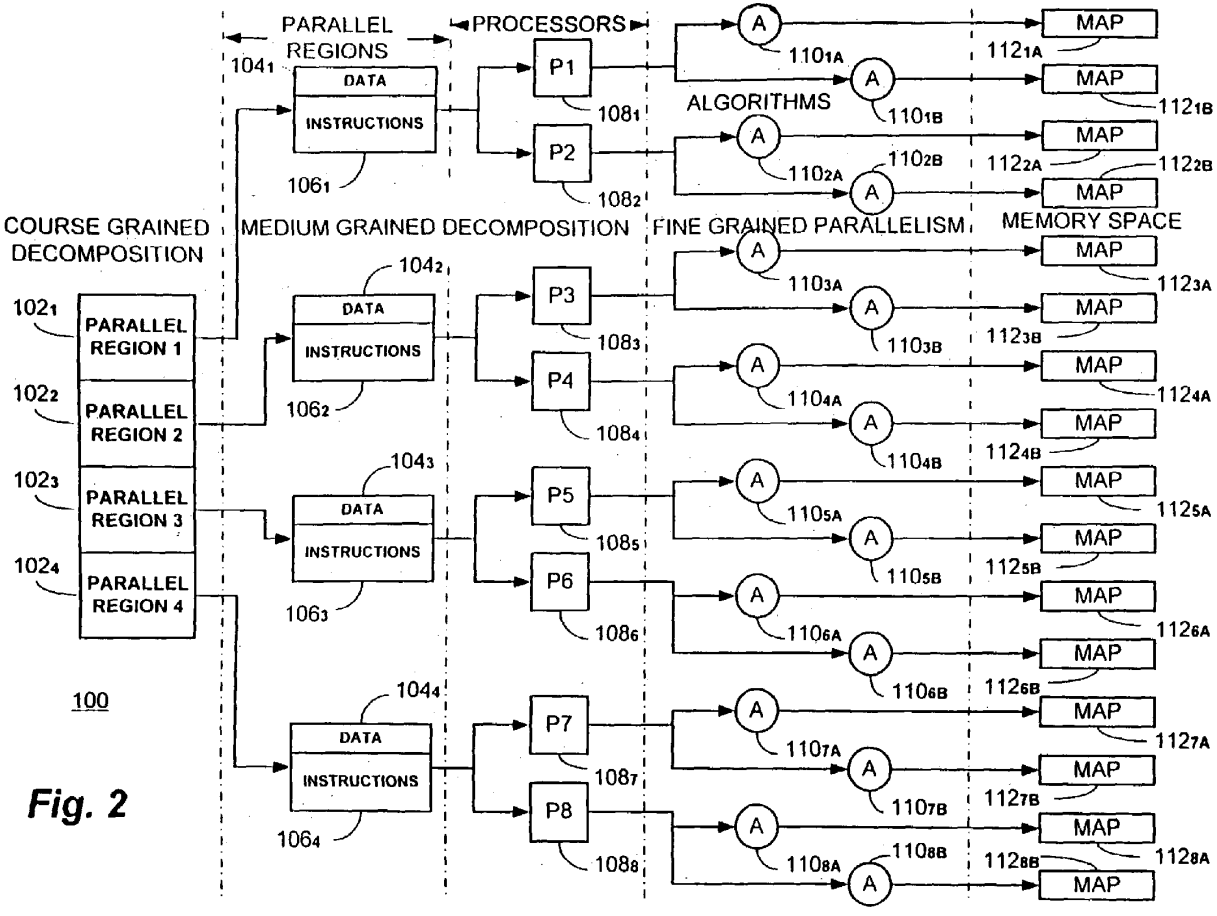

PARALLEL REGIONS
PROCESSORS
COURSE GRAINED DECOMPOSITION
MEDIUM GRAINED DECOMPOSITION
FINE GRAINED PARALLELISM
ALGORITHMS
MEMORY SPACE
PARALLEL REGION 1
PARALLEL REGION 2
PARALLEL REGION 3
PARALLEL REGION 4
DATA
INSTRUCTIONS
P1
P2
P3
P4
P5
P6
P7
P8
A
MAP
100

Fig. 2

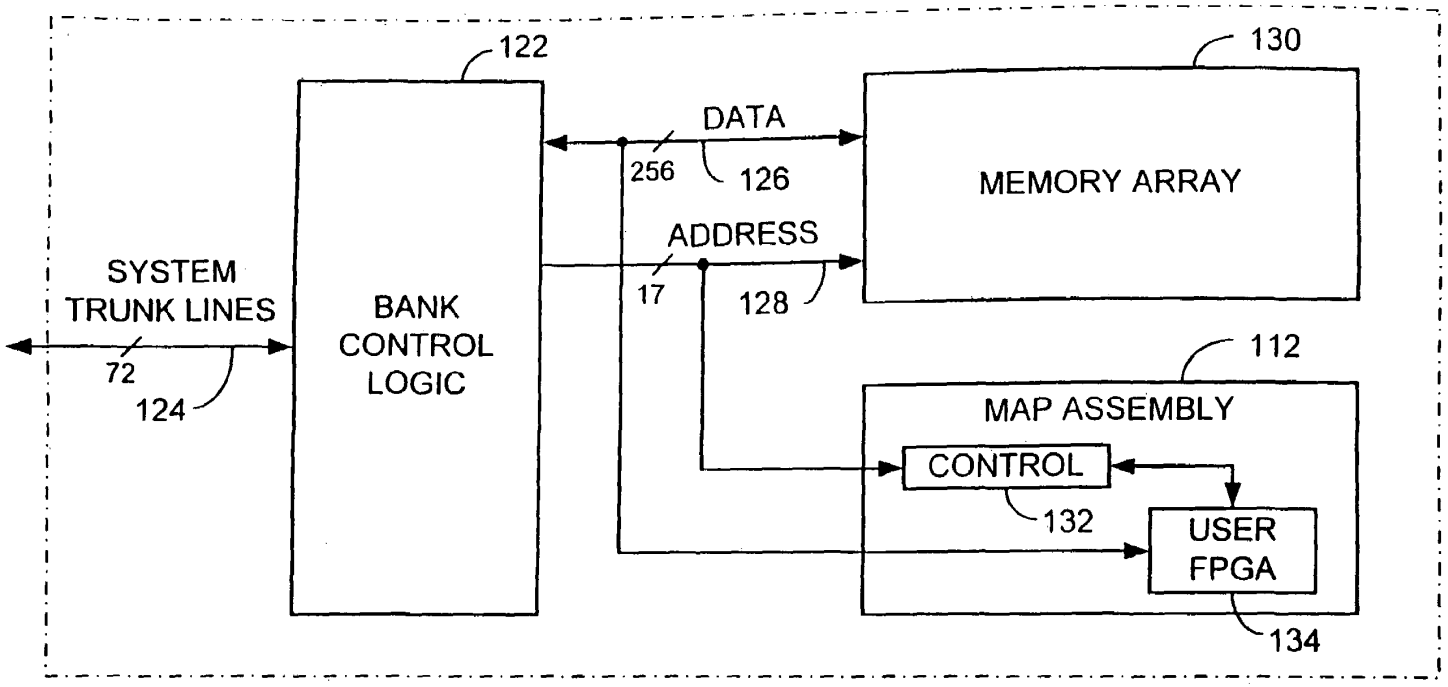

122
130
DATA
256
126
MEMORY ARRAY
ADDRESS
17
128
SYSTEM
TRUNK LINES
72
124
BANK
CONTROL
LOGIC
112
MAP ASSEMBLY
CONTROL
132
USER
FPGA
134
120

Fig. 3

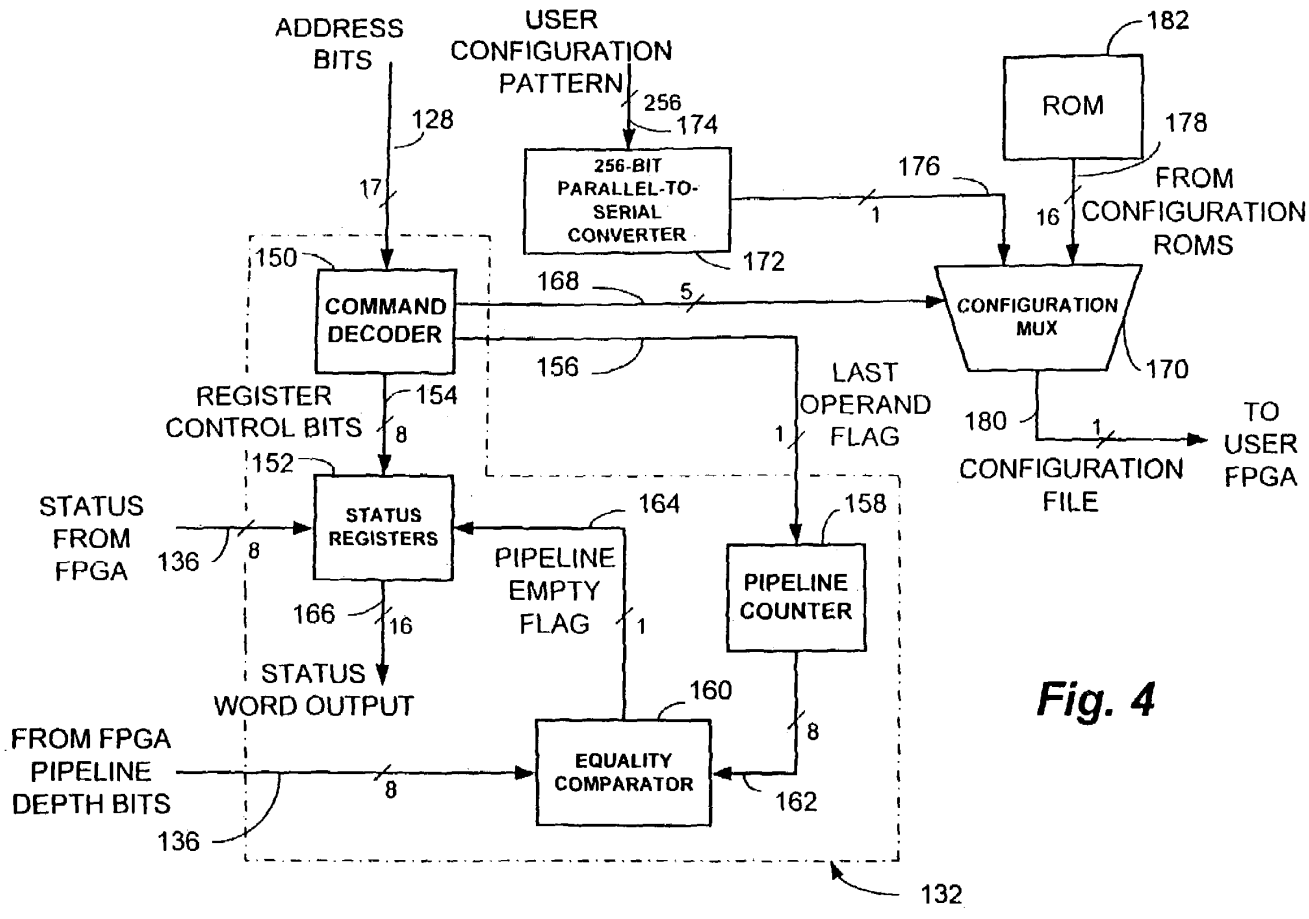

ADDRESS BITS
128
17
USER CONFIGURATION PATTERN
256
174
256-BIT PARALLEL-TO-SERIAL CONVERTER
172
176
1
ROM
182
178
16
FROM CONFIGURATION ROMS
150
COMMAND DECODER
168
5
CONFIGURATION MUX
170
156
LAST OPERAND FLAG
1
180
1
TO USER FPGA
CONFIGURATION FILE
REGISTER CONTROL BITS
154
8
152
STATUS FROM FPGA
136
8
STATUS REGISTERS
164
158
PIPELINE EMPTY FLAG
1
PIPELINE COUNTER
166
16
STATUS WORD OUTPUT
160
8
FROM FPGA PIPELINE DEPTH BITS
8
136
EQUALITY COMPARATOR
162
132

Fig. 4

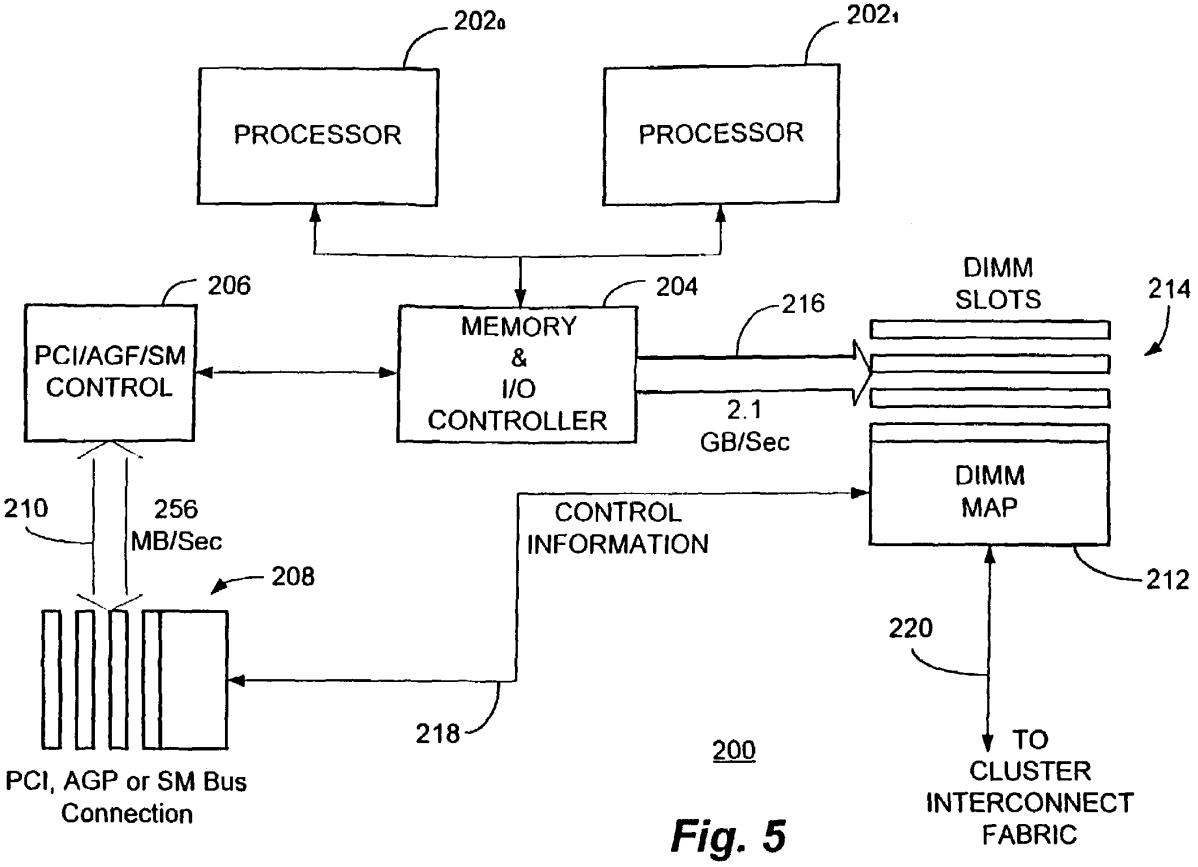

202₀
202₁
PROCESSOR
PROCESSOR
206
204
216
DIMM SLOTS
214
PCI/AGF/SM CONTROL
MEMORY & I/O CONTROLLER
2.1 GB/Sec
DIMM MAP
210
256 MB/Sec
CONTROL INFORMATION
208
212
220
218
200
PCI, AGP or SM Bus Connection
TO CLUSTER INTERCONNECT FABRIC

Fig. 5

SWITCH/NETWORK ADAPTER PORT FOR CLUSTERED COMPUTERS EMPLOYING A CHAIN OF MULTI-ADAPTIVE PROCESSORS IN A DUAL IN-LINE MEMORY MODULE FORMAT

CROSS REFERENCE TO RELATED PATENT APPLICATIONS

The present invention is a divisional patent application of U.S. patent application Ser. No. 09/932,330 filed Aug. 17, 2001, which is a continuation-in-part patent application of U.S. patent application Ser. No. 09/755,744 filed Jan. 5, 2001, now abandoned, which is a divisional patent application of U.S. patent application Ser. No. 09/481,902 filed Jan. 12, 2000, now U.S. Pat. No. 6,247,110, which is a continuation of U.S. patent application Ser. No. 08/992,763 filed Dec. 17, 1997 for: "Multiprocessor Computer Architecture Incorporating a Plurality of Memory Algorithm Processors in the Memory Subsystem", now U.S. Pat. No. 6,076,152, assigned to SRC Computers, Inc., Colorado Springs, Colo., assignee of the present invention, the disclosures of which are herein specifically incorporated by this reference.

BACKGROUND OF THE INVENTION

The present invention relates, in general, to the field of computer architectures incorporating multiple processing elements. More particularly, the present invention relates to a switch/network adapter port ("SNAP") for clustered computers employing a chain of multi-adaptive processors ("MAP™", a trademark of SRC Computers, Inc.) in a dual in-line memory module ("DIMM") format to significantly enhance data transfer rates over that otherwise available from the peripheral component interconnect ("PCI") bus.

Among the most currently promising methods of creating large processor count, cost-effective computers involves the clustering together of a number of relatively low cost microprocessor based boards such as those commonly found in personal computers ("PCs"). These various boards are then-operated using available clustering software to enable them to execute, in unison, to solve one or more large problems. During this problem solving process, intermediate computational results are often shared between processor boards.

Utilizing currently available technology, this sharing must pass over the peripheral component interconnect ("PCI") bus, which is the highest performance external interface bus, commonly found on today's PCs. While there are various versions of this bus available, all are limited to less than 1 GB/sec. bandwidth and, because of, their location several levels of chips below the processor bus, they all. exhibit a very high latency. In low cost PCs, this bus typically offers only on the order of 256 MB/sec. of bandwidth.

These factors, both individually and collectively can significantly limit the overall effectiveness of the cluster and, if a faster interface could be found, the ability of clusters to solve large problems would be greatly enhanced. Unfortunately, designing a new, dedicated chip set that could provide such a port is not only very expensive, it would also have to be customized for each type of clustering interconnect encountered. This would naturally lead to relatively low potential sale volumes for any one version of the chipset, thus rendering it cost ineffective.

SUMMARY OF THE INVENTION

In accordance with the technique of the present invention a system and method is provided: which enables an existing, standard PC memory bus to be utilized in conjunction with a multi-adaptive processor ("MAP™", a trademark of SRC Computers, Inc.) to solve this data transfer rate problem in a universally applicable way. To this end, disclosed herein is a switch/network adapter port for clustered computers employing a chain of multi-adaptive processors in a DIMM format to significantly enhance data transfer rates over that otherwise available from the PCI bus.

One of the most commonly used memory formats in PCs today is the dual inline memory module ("DIMM") format. These modules are-presently available in what is called a double data rate ("DDR") format and PCs using this format incorporate a memory bus that can provide up to 1.6 GB/sec. of bandwidth today. In the near future, this bus will be further expanded to support quad data rate ("QDR") DIMMs having up to 3.2 GB/sec. of bandwidth. A currently available alternative form of memory is the Rambus DIMM ("RIMM"). The basic features of RIMM are similar to that of the standard DIMM so, for purposes of the preceding discussion and ensuing disclosure, the term DIMM shall be utilized to denote both forms of memory.

Since the DIMM memory comprises-the primary storage location for the PC microprocessor, it is designed to be electrically very "close" to the processor bus and thus exhibit very low latency and it is not uncommon for the latency associated with the DIMM to be on the order of only 25% of that of the PCI bus. By, in essence, harnessing this bandwidth as an interconnect between computers, greatly increased cluster performance may be realized.

To this end, by placing a MAP element (in, for example, a DIMM physical format) in one of the PC's DIMM slots, it's field programmable gate array ("FPGA") could accept the normal memory "read" and "write" transactions and convert them to a format used by an interconnect switch or network.

As disclosed in the aforementioned patents and patent applications, each MAP element may include chain ports to enable it to be coupled to other MAP elements. Through the utilization of the chain port to connect to the external clustering fabric, data packets can then be sent to remote nodes where they can be received by an identical board. In this particular application, the MAP element would extract the data from the packet and store it until needed by the receiving processor.

This technique results in the provision of data transfer rates several times higher than that of any currently available PC interface. However, the electrical protocol of the DIMMs is such that once the data arrives at the receiver, there is no way for a DIMM module to signal the microprocessor that it has arrived, and without this capability, the efforts of the processors would have to be synchronized through the use of a continued polling of the MAP elements to determine if data has arrived. Such a technique would totally consume the microprocessor and much of its bus bandwidth thus stalling all other bus agents.

To avoid this situation, the DIMM MAP element may be further provided with a connection to allow it to communicate with the existing PCI bus and could then generate communications control packets and send them via the PCI bus to the processor. Since these packets would account for but a very small percentage of the total data moved, the low bandwidth effects of the PCI bus are minimized and conventional PCI interrupt signals could also be utilized to inform the processor that data has arrived. In accordance with another implemen-

tation of the present invention, the system maintenance ("SM") bus could also be used to signal the processor. The SM bus is a serial current mode bus that conventionally allows various devices on the processor board to interrupt the processor.

With a MAP element associated with what might be an entire DIMM slot, the PC will allocate a large block of addresses, typically on the order of 1 GB, for use by the MAP element. While some of these can be decoded as commands, (as disclosed in the aforementioned patents and patent applications) many can still be used as storage. By having at least as many address locations as the normal input/output ("I/O") block size used to transfer data from peripherals, the conventional Intel™ chip sets used in most PCs will allow direct I/O transfers into the MAP element. This then allows data to arrive from, for example, a disk and to pass directly into a MAP element. It then may be altered in any fashion desired, packetized and transmitted to a remote node. Because both the disk's PCI port and the MAP element DIMM slots are controlled by the PC memory controller, no processor bus bandwidth is consumed by this transfer.

It should also be noted that in certain PCs, several DIMMs may be interleaved to provide wider memory access capability in order to increase memory bandwidth. In these systems, the previously described technique may also be utilized concurrently in several DIMM slots. Nevertheless, regardless of the particular implementation chosen, the end result is a DIMM-based MAP element having one or more connections to the PCI bus and an external switch or network which results in many times the performance of a PCI-based connection alone as well as the ability to process data as it passes through the interconnect fabric.

Particularly disclosed herein is a microprocessor based computer system utilizing either a DIMM or RIMM based MAP element for the purpose of implementing a connection to an external switch, network, or other device. Further disclosed herein is a DIMM or RIMM based MAP element having connections to the either the PCI or SM bus for purposes of passing control information to the host microprocessor or other control chips. Still further disclosed herein is a DIMM or RIMM based MAP element having the capability to alter data passing through it to and from an external interconnect fabric or device.

BRIEF DESCRIPTION OF THE DRAWINGS

The aforementioned and other features and objects of the present invention and the manner of attaining them will become more apparent and the invention itself will be best understood by reference to the following description of a preferred embodiment taken in conjunction with the accompanying drawings, wherein:

FIG. **1** is a simplified, high level, functional block diagram of a multiprocessor computer architecture employing multi-adaptive processors ("MAP") in accordance with the disclosure of the aforementioned patents and patent applications in a particular embodiment wherein direct memory access ("DMA") techniques may be utilized to send commands to the MAP elements in addition to data;

FIG. **2** is a simplified logical block diagram of a possible computer application program decomposition sequence for use in conjunction with a multiprocessor computer architecture utilizing a number of MAP elements located, for example, in the computer system memory space;

FIG. **3** is a more detailed functional block diagram of an exemplary individual one of the MAP elements of the preceding figures and illustrating the bank control logic, memory array and MAP assembly thereof;

FIG. **4** is a more detailed functional block diagram of the control block of the MAP assembly of the preceding illustration illustrating its interconnection to the user FPGA thereof in a particular embodiment; and

FIG. **5** is a functional block diagram of an exemplary embodiment of the present invention comprising a switch/network adapter port for clustered computers employing a chain of multi-adaptive processors in a DIMM format to significantly enhance data transfer rates over that otherwise available from the peripheral component interconnect ("PCI") bus.

DESCRIPTION OF AN EXEMPLARY EMBODIMENT

With reference now to FIG. **1**, a multiprocessor computer **10** architecture in accordance with the disclosures of the foregoing patents and patent applications is shown. The multiprocessor computer **10** incorporates N processors $\mathbf{12}_0$ through $\mathbf{12}_N$ which are bi-directionally coupled to a memory interconnect fabric **14**. The memory interconnect fabric **14** is then also coupled to M memory banks comprising memory bank subsystems $\mathbf{16}_0$ (Bank **0**) through **16**M (Bank M). A number of multi-adaptive processor elements ("MAP™") **112** (as shown with more particularity in the following figure) are associated with one or more of the memory banks **16**. The MAP elements **112** may include chain ports as also disclosed in the aforementioned patents and patent applications.

With reference now to FIG. **2**, a representative application program decomposition for a multiprocessor computer architecture **100** incorporating a plurality of multi-adaptive processor elements **112** in accordance with the present invention is shown. The computer architecture **100** is operative in response to user instructions and data which, in a coarse grained portion of the decomposition, are selectively directed to one of (for purposes of example only) four parallel regions $\mathbf{102}_1$ through $\mathbf{102}_4$ inclusive. The instructions and data output from each of the parallel regions $\mathbf{102}_1$ through $\mathbf{102}_4$ are respectively input to parallel regions segregated into data areas $\mathbf{104}_1$ through $\mathbf{104}_4$ and instruction areas $\mathbf{106}_1$ through $\mathbf{106}_4$. Data maintained in the data areas $\mathbf{104}_1$ through $\mathbf{104}_4$ and instructions maintained in the instruction areas $\mathbf{106}_1$ through $\mathbf{106}_4$ are then supplied to, for example, corresponding pairs of processors $\mathbf{108}_1$, $\mathbf{108}_2$ (P**1** and P**2**); $\mathbf{108}_3$, $\mathbf{108}_4$ (P**3** and P**4**); $\mathbf{108}_5$, $\mathbf{108}_6$ (P**5** and P**6**); and $\mathbf{108}_7$, $\mathbf{108}_8$ (P**7** and P**8**) as shown. At this point, the medium grained decomposition of the instructions and data has been accomplished.

A fine grained decomposition, or parallelism, is effectuated by a further algorithmic decomposition wherein the output of each of the processors $\mathbf{108}_1$ through $\mathbf{108}_8$, is broken up, for example, into a number of fundamental algorithms $\mathbf{110}_{1A}$, $\mathbf{110}_{1B}$, $\mathbf{110}_{2A}$, $\mathbf{110}_{2B}$ through $\mathbf{110}_{8B}$ as shown. Each of the algorithms is then supplied to a corresponding one of the MAP elements $\mathbf{112}_{1A}$, $\mathbf{112}_{1B}$, $\mathbf{112}_{2A}$, $\mathbf{112}_{2B}$, through $\mathbf{112}_{8B}$ which may be located in the memory space of the computer architecture **100** for execution therein as will be more fully described hereinafter.

With reference additionally now to FIG. **3**, an exemplary implementation of a memory bank **120** in a MAP element-based system computer architecture **100** is shown for a representative one of the MAP elements **112** illustrated in the preceding figure. Each memory bank **120** includes a bank control logic block **122** bi-directionally coupled to the computer system trunk lines, for example, a 72 line bus **124**. The bank control logic block **122** is coupled to a bi-directional

data bus **126** (for example 256 lines) and supplies addresses on an address bus **128** (for example 17 lines) for accessing data at specified locations within a memory array **130**.

The data bus **126** and address bus **128** are also coupled to a MAP element **112**. The MAP element **112** comprises a control block **132** coupled to the address bus **128**. The control block **132** is also bi-directionally coupled to a user field programmable gate array ("FPGA") **134** by means of a number of signal lines **136**. The user FPGA **134** is coupled directly to the data bus **126**. In a particular embodiment, the FPGA **134** may be provided as a Lucent Technologies OR3T80 device.

The exemplary computer architecture **100** comprises a multiprocessor system employing uniform memory access across common shared memory with one or more MAP elements **112** which may be located in the memory subsystem, or memory space. As previously described, each MAP element **112** contains at least one relatively large FPGA **134** that is used as a reconfigurable functional unit. In addition, a control block **132** and a preprogrammed or dynamically programmable configuration ROM (as will be more fully described hereinafter) contains the information needed by the reconfigurable MAP element **112** to enable it to perform a specific algorithm. It is also possible for the user to directly download a new configuration into the FPGA **134** under program control, although in some instances this may consume a number of memory accesses and might result in an overall decrease in system performance if the algorithm was short-lived.

FPGAs have particular advantages in the application shown for several reasons. First, commercially available FPGAs now contain sufficient internal logic cells to perform meaningful computational functions. Secondly, they can operate at bus speeds comparable to microprocessors, which eliminates the need for speed matching buffers. Still further, the internal programmable routing resources of FPGAs are now extensive enough that meaningful algorithms can now be programmed without the need to reassign the locations of the input/output ("1/0") pins.

By, for example, placing the MAP element **112** in the memory subsystem or memory space, it can be readily accessed through the use of memory "read" and "write" commands, which allows the use of a variety of standard operating systems. In contrast, other conventional implementations may propose placement of any reconfigurable logic in or near the processor, however these conventional implementations are generally much less effective in a multiprocessor environment because only one processor may have rapid access to it. Consequently, reconfigurable logic must be placed by every processor in a multiprocessor system, which increases the overall system cost. Because a MAP element **112** has DMA capability, (allowing it to write to memory), and because it receives its operands via writes to memory, it is possible to allow a MAP element **112** to feed results to another MAP element **112** through use of a chain port. This is a very powerful feature that allows for very extensive pipelining and parallelizing of large tasks, which permits them to complete faster.

Many of the algorithms that may be implemented will receive an operand and require many clock cycles to produce a result. One such example may be a multiplication that takes 64 clock cycles. This same multiplication may also need to be performed on thousands of operands. In this situation, the incoming operands would be presented sequentially so that while the first operand requires 64 clock cycles to produce results at the output, the second operand, arriving one clock cycle later at the input, will show results one clock cycle later at the output. Thus, after an initial delay of 64 clock cycles, new output data will appear on every consecutive clock cycle until the results of the last operand appears. This is called "pipelining".

In a multiprocessor system, it is quite common for the operating system to stop a processor in the middle of a task, reassign it to a higher priority task, and then return it, or another, to complete the initial task. When this is combined with a pipelined algorithm, a problem arises (if the processor stops issuing operands in the middle of a list and stops accepting results) with respect to operands already issued but not yet through the pipeline. To handle this issue, a solution involving the combination of software and hardware as disclosed in the aforementioned patents and patent applications.

To make use of any type of conventional reconfigurable hardware, the programmer could embed the necessary commands in his application program code. The drawback to this approach is that a program would then have to be tailored to be specific to the MAP hardware. The system disclosed eliminates this problem. Multiprocessor computers often use software called parallelizers. The purpose of this software is to analyze the user's application code and determine how best to split it up among the processors. The technique disclosed provides significant advantages over a conventional parallelizer and enables it to recognize portions of the user code that represent algorithms that exist in MAP elements **112** for that system and to then treat the MAP element **112** as another computing element. The parallelizer then automatically generates the necessary code to utilize the MAP element **112**. This allows the user to write the algorithm directly in his code, allowing it to be more portable and reducing the knowledge of the system hardware that he has to have to utilize the MAP element **112**.

With reference additionally now to FIG. **4**, a block diagram of the MAP control block **132** is shown in greater detail. The control block **132** is coupled to receive a number of command bits (for example, 17) from the address bus **128** at a command decoder **150**. The command decoder **150** then supplies a number of register control bits to a group of status registers iS2 on an eight bit bus **154**. The command decoder **150** also supplies a single bit last operand flag on line **156** to a pipeline counter **158**. The pipeline counter **158** supplies an eight bit output to an equality comparator **160** on bus **162**. The equality comparator **160** also receives an eight bit signal from the FPGA **134** on bus **136** indicative of the pipeline depth. When the equality comparator **160** determines that the pipeline is empty, it provides a single bit pipeline empty flag on line **164** for input to the status registers **152**. The status registers **152** are also coupled to receive an eight bit status signal from the FPGA **134** on bus **136** and it produces a sixty four bit status word output on bus **166** in response to the signals on bus **136**, **154** and line **164**.

The command decoder **150** also supplies a five bit control signal on line **168** to a configuration multiplexer ("MUX") **170** as shown. The configuration MUX **170** receives a single bit output of a 256 bit parallel-serial converter **172** on line **176**. The inputs of the 256 bit parallel-to-serial converter **172** are coupled to a 256 bit user configuration pattern bus **174**. The configuration MUX **170** also receives sixteen single bit inputs from the configuration ROMs (illustrated as ROM **182**) on bus **178** and provides a single bit configuration file signal on line **180** to the user FPGA **134** as, selected by the control signals from the command decoder **150** on the bus **168**.

In operation, when a processor **108** is halted by the operating system, the operating system will issue a last operand command to the MAP element **112** through the use of command bits embedded in the address field on bus **128**. This

command is recognized by the command decoder **150** of the control block **132** and it initiates a hardware pipeline counter **158**. When the algorithm was initially loaded into the FPGA **134**, several output bits connected to the control block **132** were configured to display a binary representation of the number of clock cycles required to get through its pipeline (i.e. pipeline "depth") on bus **136** input to the equality comparator **160**. After receiving the last operand command, the pipeline counter **158** in the control block **132** counts clock cycles until its count equals the pipeline depth for that particular algorithm. At that point, the equality comparator **160** in the control block **132** de-asserts a busy bit on line **164** in an internal group of status registers **152**. After issuing the last operand signal, the processor **108** will repeatedly read the status registers **152** and accept any output data on bus **166**. When the busy flag is de-asserted, the task can be stopped and the MAP element **112** utilized for a different task. It should be noted that it is also possible to leave the MAP element **112** configured, transfer the program to a different processor **108** and restart the task where it left off.

In order to evaluate the effectiveness of the use of the MAP element **112** in a given application, some form of feedback to the use is required. Therefore, the MAP element **112** may be equipped with internal registers in the control block **132** that allow it to monitor efficiency related factors such as the number of input operands versus output data, the number of idle cycles over time and the number of system monitor interrupts received over time. One of the advantages that the MAP element **112** has is that because of its reconfigurable nature, the actual function and type of function that are monitored can also change as the algorithm changes. This provides the user with an almost infinite number of possible monitored factors without having to monitor all factors all of the time.

With reference additionally now to FIG. **5**, a functional block diagram of an exemplary embodiment of a computer system **200** in accordance with the present invention is shown comprising a switch/network adapter port for clustered computers employing a chain of multi-adaptive processors in a DIMM format to significantly enhance data transfer rates over that otherwise available from the peripheral component interconnect ("PCI") bus.

In the particular embodiment illustrated, the computer system **200** includes one or more processors $\mathbf{202}_0$ and $\mathbf{202}_1$ which are coupled to an associated PC memory and I/O controller **204**. In operation, the controller **204** sends and receives control information from a PCI control block **206**. It should be noted that in alternative implementations of the present invention, the control block **206** may also be an AGP or SM control block. The PCI control block **206** is coupled to one or more PCI card slots **208** by means of a relatively low bandwidth PCI bus **210** which allows data transfers at a rate of substantially 256 MB/sec. In the alternative embodiments of the present invention mentioned above, the card slots **208** may alternatively comprise accelerated graphics port ("AGP") or system maintenance ("SM") bus connections.

The controller **204** is also conventionally coupled to a number of DIMM slots **214** by means of a much higher bandwidth DIMM bus **216** capable of data transfer rates of substantially 2.1 GB/sec. or greater. In accordance with a particular implementation of the present invention, a DIMM MAP element **212** is associated with, or physically located within, one of the DIMM slots **214**. Control information to or from the DIMM MAP element **212** is provided by means of a connection **218** interconnecting the PCI bus **210** and the DIMM MAP element **212**. The DIMM MAP element **212** then may be coupled to another clustered computer MAP element by means of a cluster interconnect fabric connection **220** connected to the MAP chain ports. As previously noted, the DIMM MAP element **212** may also comprise a RIMM MAP element.

Since the DIMM memory located within the DIMM slots **214** comprises the primary storage location for the PC microprocessor(s) $\mathbf{202}_0$, $\mathbf{202}_1$, it is designed to be electrically very "close" to the processor bus and thus exhibit very low latency. As noted previously, it is not uncommon for the latency associated with the DIMM to be on the order of only 25% of that of the PCI bus **210**. By, in essence, harnessing this bandwidth as an interconnect between computer systems **200**, greatly increased cluster performance may be realized.

To this end, by placing the DIMM MAP element **212** in one of the PC's DIMM slots **214**, its FPGA **134** (FIG. **3**) could accept the normal memory "read" and "write" transactions and convert them to a format used by an interconnect switch or network. As disclosed in the aforementioned patents and patent applications, each MAP element **212** includes chain ports to enable it to be coupled to other MAP elements **212**. Through the utilization of the chain port to connect to the external clustering fabric over connection **220**, data packets can then be sent to remote, nodes where they can be received by an identical board. In this particular application, the DIMM MAP element **212** would extract the data from the packet and store it until needed by the receiving processor **202**.

This technique results in the provision of data transfer rates several times higher than that of any currently available PC interface such as the PCI bus **210**. However, the electrical protocol of the DIMMs is such that once the data arrives at the receiver, there is no way for a DIMM module within the DIMM slots **214** to signal the microprocessor **202** that it has arrived, and without this capability, the efforts of the processors **202** would have to be synchronized through the use of a continued polling of the DIMM MAP elements **212** to determine if data has arrived. Such a technique would totally consume the microprocessor **202** and much of its bus bandwidth thus stalling all other bus agents.

To avoid this situation, the DIMM MAP element **212** may be further provided with the connection **218** to allow it to communicate with the existing PCI bus **210** which could then generate communications packets and send them via the PCI bus **210** to the processor **202**. Since these packets would account for but a very small percentage of the total data moved, the low bandwidth effects of the PCI bus **210** are minimized and conventional PCI interrupt signals could also be utilized to inform the processor **202** that data has arrived. In accordance with another implementation of the present invention, the system maintenance ("SM") bus (not shown) could also be used to signal the processor **202**. The SM bus is a serial current mode bus that conventionally allows various devices on the processor board to interrupt the processor **202**. In an alternative embodiment, the accelerated graphics port ("AGP") may also be utilized to signal the processor **202**.

With a DIMM MAP element **212** associated with what might be an entire DIMM slot **214**, the PC will allocate a large block of addresses, typically on the order of 1 GB, for use by the DIMM MAP element **212**. While some of these can be decoded as commands, (as disclosed in the aforementioned patents and patent applications) many can still be used as storage. By having at least as many address locations as the normal input/output ("I/O") block size used to transfer data from peripherals, the conventional Intel™ chip sets used in most PCs (including controller **204**) will allow direct I/O transfers into the DIMM MAP element **212**. This then allows data to arrive from, for example, a disk and to pass directly into a DIMM MAP element **212**. It then may be altered in any

fashion desired, packetized and transmitted to a remote node over connection **220**. Because both the disk's PCI bus **210** and the DIMM MAP element **212** and DIMM slots **214** are controlled by the PC memory controller **204**, no processor bus bandwidth is consumed by this transfer.

It should also be noted that in certain PCs, several DIMMs within the DIMM slots **214** may be interleaved to provide wider memory access capability in order to increase memory bandwidth. In these systems, the previously described technique may also be utilized concurrently in several DIMM slots **214**. Nevertheless, regardless of the particular implementation chosen, the end result is a DIMM-based MAP element **212** having one or more connections to the PCI bus **210** and an external switch or network over connection **220** which results in many times the performance of a PCI-based connection alone as well as the ability to process data as it passes through the interconnect fabric.

While there have been described above the principles of the present invention in conjunction with a specific computer architecture, it is to be clearly understood that the foregoing description is made only by way of example and not as a limitation to the scope of the invention. Particularly, it is recognized that the teachings of the foregoing disclosure will suggest other modifications to those persons skilled in the relevant art. Such modifications may involve other features which are already known per se and which may be used instead of or in addition to features already described herein. Although claims have been formulated in this application to particular combinations of features, it should be understood that the scope of the disclosure herein also includes any novel feature or any novel combination of features disclosed either explicitly or implicitly or any generalization or modification thereof which would be apparent to persons skilled in the relevant art, whether or not such relates to the same invention as presently claimed in any claim and whether or not it mitigates any or all of the same technical problems as confronted by the present invention. The applicants hereby reserve the right to formulate new claims to such features and/or combinations of such features during the prosecution of the present application or of any further application derived therefrom.

What is claimed is:

1. A processor element for a memory module bus of a computer system, said processor element comprising:

a field programmable gate array configurable to perform an identified algorithm on an operand provided thereto and operative to alter data provided directly thereto on said memory module bus; and

a direct data connection coupled to said field programmable gate array for providing said altered data directly from said memory module bus to an external device coupled thereto.

2. The processor element of claim **1** further comprising:

a control connection coupled to said processor element for indicating to a processor of said computer system an arrival of data on said data connection from said external device.

3. The processor element of claim **2** wherein said control connection indicates said arrival of data to said processor by means of a peripheral bus.

4. The processor element of claim **3** wherein said peripheral bus comprises a PCI bus.

5. The processor element of claim **2** wherein said control connection indicates said arrival of data to said processor by means of a graphics bus.

6. The processor element of claim **5** wherein said graphics bus comprises an AGP bus.

7. The processor element of claim **2** wherein said control connection indicates said arrival of data to said processor by means of a system maintenance bus.

8. The processor element of claim **7** wherein said system maintenance bus comprises an SM bus.

9. The processor element of claim **1** wherein said memory module bus comprises a DIMM bus.

10. The processor element of claim **9** wherein said processor element comprises a DIMM physical format.

11. The processor element of claim **1** wherein said memory module bus comprises a in-line memory module serial interface bus.

12. The processor element of claim **11** wherein said processor element comprises a in-line memory module serial interface physical format.

13. The processor element of claim **1** wherein said external device comprises one of another computer system, switch or network.

14. The processor element of claim **1** wherein said processor of said computer system comprises a plurality of processors.

15. The processor element of claim **1** wherein said field programmable gate array is further operative to alter data provided thereto from said external device on said data connection and providing said altered data on said memory module bus.

* * * * *

EXHIBIT F

US006434687B1

(12) **United States Patent**
Huppenthal

(10) **Patent No.: US 6,434,687 B1**
(45) **Date of Patent: Aug. 13, 2002**

(54) **SYSTEM AND METHOD FOR ACCELERATING WEB SITE ACCESS AND PROCESSING UTILIZING A COMPUTER SYSTEM INCORPORATING RECONFIGURABLE PROCESSORS OPERATING UNDER A SINGLE OPERATING SYSTEM IMAGE**

(75) Inventor: **Jon M. Huppenthal**, Colorado Springs, CO (US)

(73) Assignee: **SRC Computers, Inc.**, Colorado Springs, CO (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/888,276**

(22) Filed: **Jun. 22, 2001**

Related U.S. Application Data

(63) Continuation-in-part of application No. 09/563,561, filed on May 3, 2000, now Pat. No. 6,339,819, which is a continuation-in-part of application No. 09/481,902, filed on Jan. 12, 2000, now Pat. No. 6,247,110, which is a continuation of application No. 08/992,763, filed on Dec. 17, 1997, now Pat. No. 6,076,152.

(51) **Int. Cl.**[7] **G06F 15/16**

(52) **U.S. Cl.** **712/32**; 707/501.1; 707/513; 709/203; 709/219

(58) **Field of Search** 707/501.1, 513; 709/203, 219; 712/32

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,671,377	A	*	9/1997	Bleidt et al.	345/723
5,715,453	A	*	2/1998	Stewart	707/513
5,867,706	A	*	2/1999	Martin et al.	709/105
5,987,480	A	*	11/1999	Donohue et al.	707/501
6,009,410	A	*	12/1999	LeMole et al.	705/14
6,128,663	A	*	10/2000	Thomas	709/228

OTHER PUBLICATIONS

Albaharna, Osama, et al., "On the viability of FPGA–based integrated coprocessors", © 1996 IEEE, Publ. No. 0–8186–7548–9/96, pp. 206–215.

Barthel, Dominique Aug. 25–26, 1997, "PVP a Parallel Video coProcessor", Hot Chips IX, pp. 203–210.

Bittner, Ray, et al., "Computing kernels implemented with a wormhole RTR CCM", © 1997 IEEE, Publ. No. 0–8186–8159–4/97, pp. 98–105.

Babb, Jonathan, et al., "Parallelizing applications into silicon", © 1999 IEEE.

Bertin, Patrice, et al., "Programmable active memories: a performance assessment", © 1993 Massachusetts Institute of Technology, pp. 88–102.

Culbertson, W. Bruce, et al., "Exploring architectures for volume visualization on the Teramac custom computer", © 1996 IEEE, Publ. No. 0–8186–7548–9/96, pp. 80–88.

Culbertson, W. Bruce, et al., "Defect tolerance on the Teramac custom computer", © 1997 IEEE, Publ. No. 0–8186–8159–4/97, pp. 116–123.

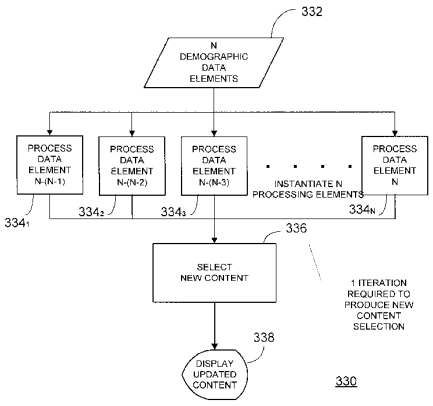
(List continued on next page.)

Primary Examiner—Kenneth S. Kim
(74) *Attorney, Agent, or Firm*—William J. Kubida; Hogan & Hartson LLP

(57) **ABSTRACT**

A system and method for accelerating web site access and processing utilizing a multiprocessor computer system incorporating reconfigurable and standard microprocessors as the web site server. One or more reconfigurable processors may be utilized, for example, in accelerating site visitor demographic data processing, real time web site content updating, database searches and other processing associated with e-commerce applications. In a particular embodiment disclosed, all of the reconfigurable and standard microprocessors may be controlled by a single system image of the operating system, although cluster management software may be utilized to cause a cluster of microprocessors to appear to the user as a single copy of the operating system.

25 Claims, 14 Drawing Sheets



OTHER PUBLICATIONS

Chan, Pak, et al., "Architectural tradeoffs in field–programmable–device–based computing systems", © 1993 IEEE, Publ. No. 0–8186–3890–7/93, pp. 152–161.

Clark, David, et al., "Supporting FPGA microprocessors through retargetable software tools", © 1996 IEEE, Publ. No. 0–8186–7548–9/96, pp. 195–103.

Cuccaro, Steven, et al., "The CM–2X: a hybrid CM–2/Xilink prototype", © 1993 IEEE, Publ. No. 0–8186–3890–7/93, pp. 121–130.

Dehon, Andre, "DPGA–Coupled microprocessors: commodity IC for the early 21st century", © 1994 IEEE, Publ. No. 0–8186–5490–2/94, pp. 31–39.

Dhaussy, Philippe, et al., "Global control synthesis for an MIMD/FPGA machine", © 1994 IEEE, Pub. No. 0–8186–5490–2/94, pp. 72–81.

Deshpande, Deepali, et al., "Hybrid data/configuration caching for striped FPGAs" © 1999 IEEE.

Elliott, Duncan, et al., "Computational Ram: a memory–SIMD hybrid and its application to DSP",© 1992 IEEE, Publ. No. 0–7803–0246–X/92, pp. 30.6.1–30.6.4.

Fortes, Jose, et al., "Systolic arrays, a survey of seven projects", © 1987 IEEE, Publ. No. 0018–9162/87/0700–0091, pp. 91–103.

Purna, Karthikeya, et al., "Temporal partitioning and scheduling data flow graphs for reconfigurable computers", © 1999 IEEE, Publ. No. 0018–9340/99 pp. 579–590.

Gibbs, W. Wayt, "Blitzing bits", © 1999 Scientific American Presents, pp. 57–61.

Gonzalez, Ricardo, "Configurable and extensible processors change system design", Aug. 15–17, 1999, Hot Chips 11 Tutorials, pp. 135–146.

Graham, Paul, et al., ."FPGA–based sonar processing", © 1998 ACM 0–89791–978–5/98, pp. 201–208.

Hauser, John, et al.: "GARP: a MIPS processor with a reconfigurable co–processor", © 1997 IEEE, Pub. No. 0–08186–8159–4/97, pp. 12–21.

Hammond, Lance, et al., "The Stanford Hydra CMP", Aug. 15–17, 1999 Hot Chips 11 Tutorials, pp. 23–31.

Hartenstein, Reiner, et al., "A reconfigurable data–driven ALU for Xputers", © 1994 IEEE, Publ. No. 0–8186–5490–2/94, pp. 139–146.

Hayes, John, et al., "A microprocessor–based hypercube, supercomputer", © 1986 IEEE, Publ. No. 0272–1732/86/1000–0006, pp. 6–17.

Hagiwara, Hiroshi, et al., "A dynamically microprogrammable computer with low–level parallelism", © 1980 IEEE, Pub. No. 0018–9340/80/07000–0577, pp. 577–594.

Hasebe, A., et al., "Architecture of SIPS, a real time image processing system," © 1988 IEEE, Publ. No. CH2603–9/88/0000/0621, pp. 621–630.

Jean, Jack, et al., "Dynamic reconfiguration to support concurrent applications", © 1999 IEEE, Publ. No. 0018–9340/99, pp. 591–602.

Kastrup, Bernardo, et al., "Concise: a compiler–driven CPLD–based instruction set accelerator", © 1999 IEEE.

King, William, et al., "Using MORRPH in an industrial machine vision system", © 1996 IEEE, Pub. No. 08186–7548–9/96, pp. 18–26.

Manohar, Swaminathan, et al., "A pragmatic approach to systolic design", © 1988 IEEE, Publ. No. CH2603–9/88–0000/0463, pp. 463–472.

Motomura, Masato, et al., "An embedded DRAM–FPGA chip with instantaneous logic reconfiguration", © 1998 IEEE, Publ. No. 0–8186–8900–5/98, pp. 264–266.

McConnell, Ray, "Massively parallel computing on the FUZION chip", Aug. 15–17, 1999, Hot Chips 11 Tutorials, pp. 83–94.

McShane, Erik, et al., "Functionally integrated systems on a chip: technologies, architectures, CAD tools, and applications", © 1998 IEEE, Publ. No. 8–8186–8424–0/98, pp. 67–75.

Mauduit, Nicolas, et al., "Lneuro 1.0: a piece of hardware LEGO for building neural network systems," © 1992 IEEE, Publ. No. 1045–9227/92 pp. 414–422.

Patterson, David, et al., "A case for intelligent DRAM: IRAM", Hot Chips VIII, Aug. 19–20, 1996, pp. 75–94.

Peterson, Janes, et al., "Scheduling and partitioning ANSI–C programs onto multi–FPGA CCM architectures", © 1996 IEEE, Publ. No. 0–8186–7548–9/96, pp. 178–187.

Rupp, Charley, et al., "The NAPA adaptive processing architecture", © 1998 the Authors, pp. 1–10.

Saito, Osamu, et al., "A 1M synapse self learning digital neural network chip", © 1998 IEEE, Publ. No. 0–7803–4344/1/98, pp. 94–95.

Schott, Brian, et al., "Architectures for system–level applications of adaptive computing", © 1999 IEEE.

Schmit, Herman, "Incremental reconfiguration for pipelined applications," © 1997 IEEE, Publ. No. 0–8186–8159–4/97, pp. 47–55.

Villasenor, John, et al., "Configurable computing", © 1997 Scientific American, Jun. 1997.

Stone, Harold, "A logic–in–memory computer", © 1970 IEEE, IEEE Transactions on Computers, pp. 73–78, Jan. 1990.

Trimberger, Steve, et al., "A time–multiplexed FPGA", © 1997 IEEE, Publ. No. 0–8186–8159–4/97, pp. 22–28.

Thornburg, Mike, et al., "Transformable Computers", © 1994 IEEE, Publ. No. 0–8186–5602–6/94, pp. 674–679.

Tangen, Uwe, et al., "A parallel hardware evolvable computer POLYP extended abstract", © 1997 IEEE, Publ. No. 0–8186–8159/4/97, pp. 238–239.

Tomita, Shinji, et al., "A computer low–level parallelism QA–2", © 1986 IEEE, No. 0–0384–7495/86/0000/0280, pp. 280–289.

Ueda, Hirotada, et al., "A multiprocessor system utilizing enhanced DSP's for image processing", © 1988 IEEE, Publ. No. CH2603–9/88/0000/0611, pp. 611–620.

Wang, Quiang, et al., "Automated field–programmable compute accelerator design using partial evaluation", © 1997 IEEE, Publ. No. 0–8186–8159–4/97, pp. 145–154.

Wirthlin, Michael, et al., "The Nano processor: a low resource reconfigurable processor", © 1994 IEEE, Publ. No. 0–8186–5490–2/94, pp. 23–30.

Wittig, Ralph, et al., "One Chip: An FPGA processor with reconfigurable logic", © 1996 IEEE, Publ. No. 0–8186–7458–9/96, pp. 126–135.

Wirthlin, Michael, et al., "A dynamic instruction set computer", © 1995 IEEE, Publ. No. 0–8186–7086–X/95, pp. 99–107.

Yamauchi, Tsukasa, et al., "SOP: A reconfigurable massively parallel system and its control–data flow based compiling method", © 1996 IEEE, Publ. No. 0–8186–7548–9/96, pp. 148–156.

Mencer, Oskar, et al., "PAM–Blox: High Performance FPGA Design for Adaptive Computing", © 1998 IEEE, Conference Paper, INSPEC Abstract Nos. B9811–1265B–044, C9811–5210–009.

Miyamori, Takashi, et al., "A quantitative analysis of reconfigurable coprocessors for multimedia applications", © 1998 IEEE, Conference Paper, INSPEC Abstract Nos. B9811–1265F–011, C9811–5310–010.

W.H. Mangione–Smith and B.L. Hutchings. Configurable computing: The Road Ahead. In Proceedings of the Reconfigurable Architectures Workshop (RAW'97), pp. 81–96, 1997.

Mirsky, Ethan A., "Coarse–Grain Reconfigurable Computing", Massachusetts Institute of Technology, Jun. 1996.

Vemuri, Ranga R. et al., "Configurable Computing: Technology and Applications", Apr. 2000, Computer, pp. 39–40.

DeHon, Andre, "The Density Advantage of Configurable Computing", Apr. 2000, Computer, pp. 41–49.

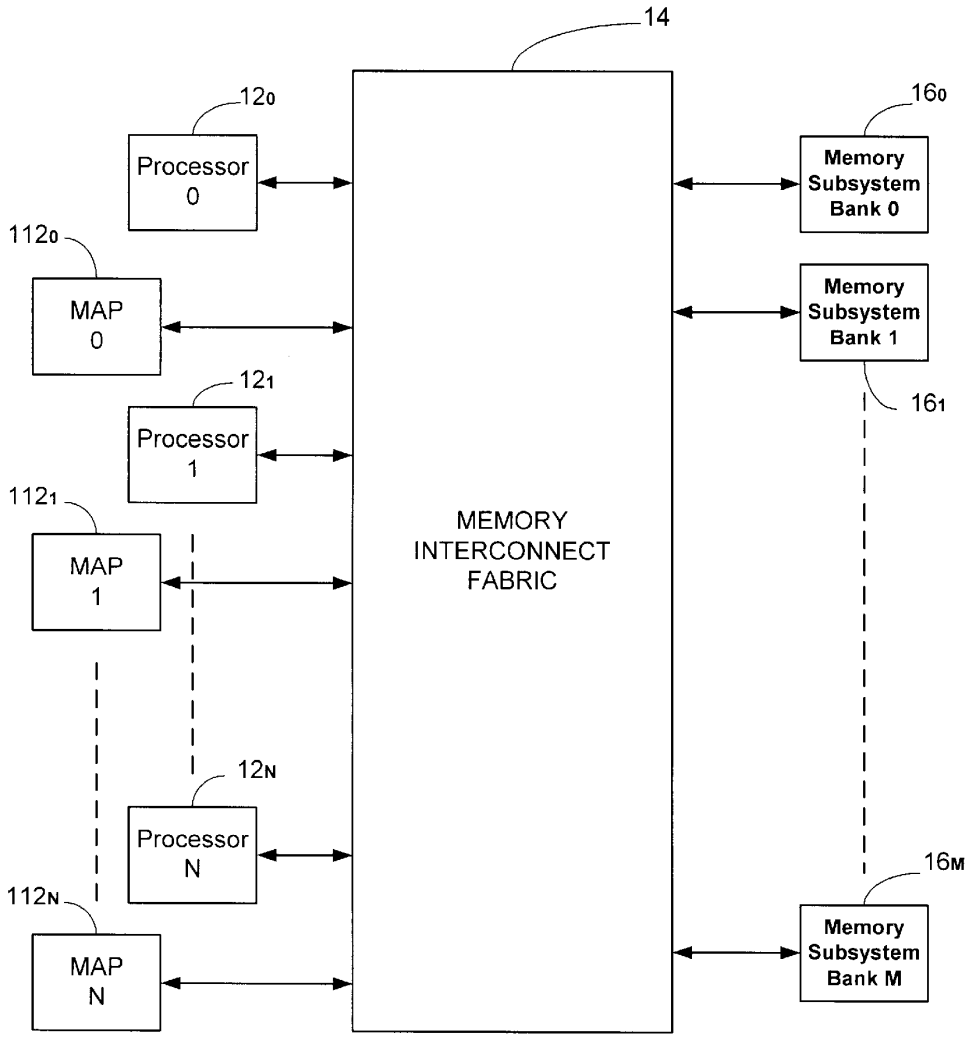
Haynes, Simon D. et al., "Video Image Processing with the Sonic Architecture", Apr. 2000, Computer, pp. 50–57.

Platzner, Marco, "Reconfigurable Accelerators for Combinatorial Problems", Apr. 2000, Computer, pp. 58–60.

Callahan, Timothy J. et al., "The Garp Architecture and C Compiler", Apr. 2000, Computer, pp. 62–69.

Goldstein, Seth Copen et al., "PipeRench: A Reconfigurable Architecture and Compiler", Apr. 2000, Computer, pp. 70–76.

* cited by examiner


14
12₀
Processor 0
112₀
MAP 0
12₁
Processor 1
112₁
MAP 1
12N
Processor N
112N
MAP N
MEMORY INTERCONNECT FABRIC
16₀
Memory Subsystem Bank 0
Memory Subsystem Bank 1
16₁
16M
Memory Subsystem Bank M

10

Fig. 1

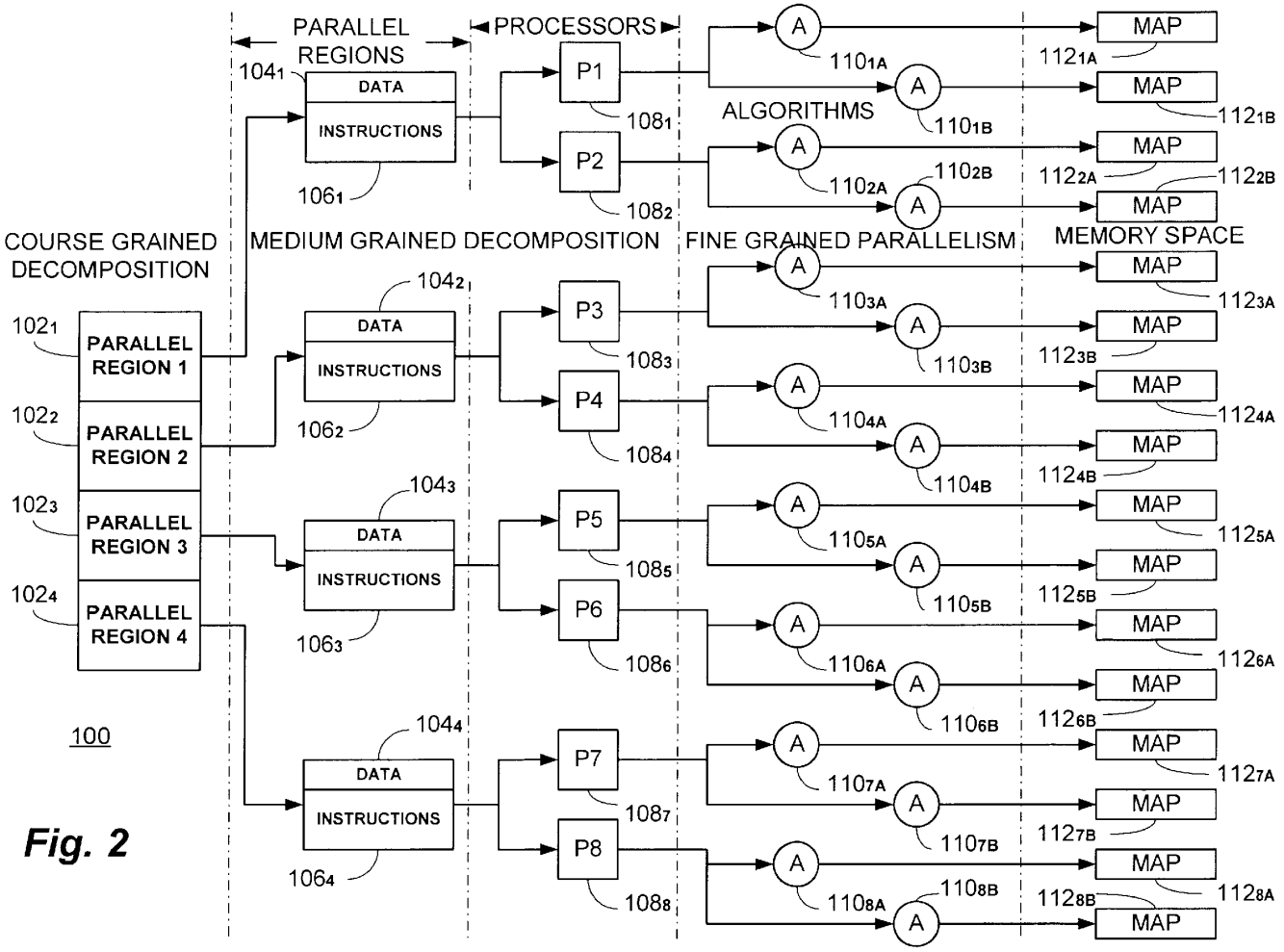

PARALLEL REGIONS
PROCESSORS
COURSE GRAINED DECOMPOSITION
MEDIUM GRAINED DECOMPOSITION
FINE GRAINED PARALLELISM
MEMORY SPACE
ALGORITHMS
PARALLEL REGION 1
PARALLEL REGION 2
PARALLEL REGION 3
PARALLEL REGION 4
DATA
INSTRUCTIONS
P1
P2
P3
P4
P5
P6
P7
P8
A
MAP
100

Fig. 2

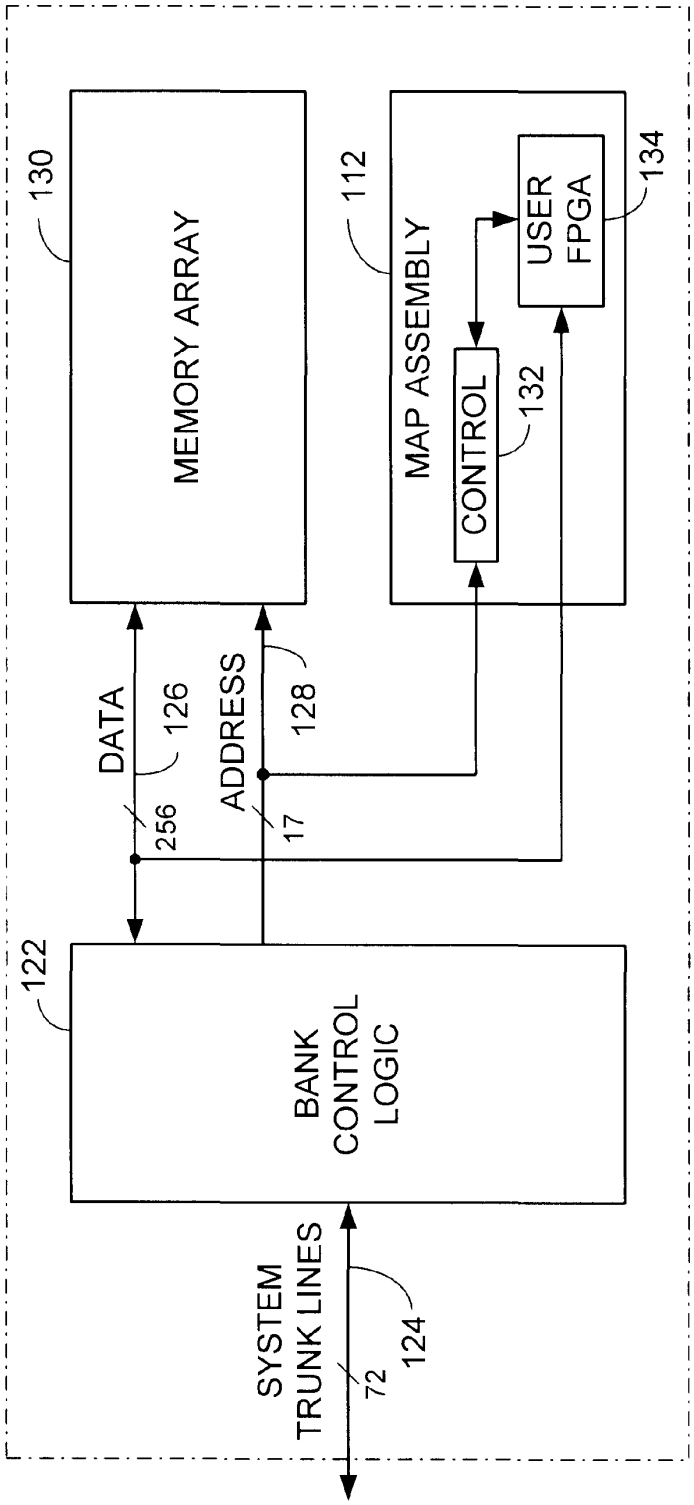

120
130
MEMORY ARRAY
112
MAP ASSEMBLY
USER FPGA
134
CONTROL
132
DATA
126
256
ADDRESS
128
17
122
BANK CONTROL LOGIC
SYSTEM TRUNK LINES
72
124

Fig. 3

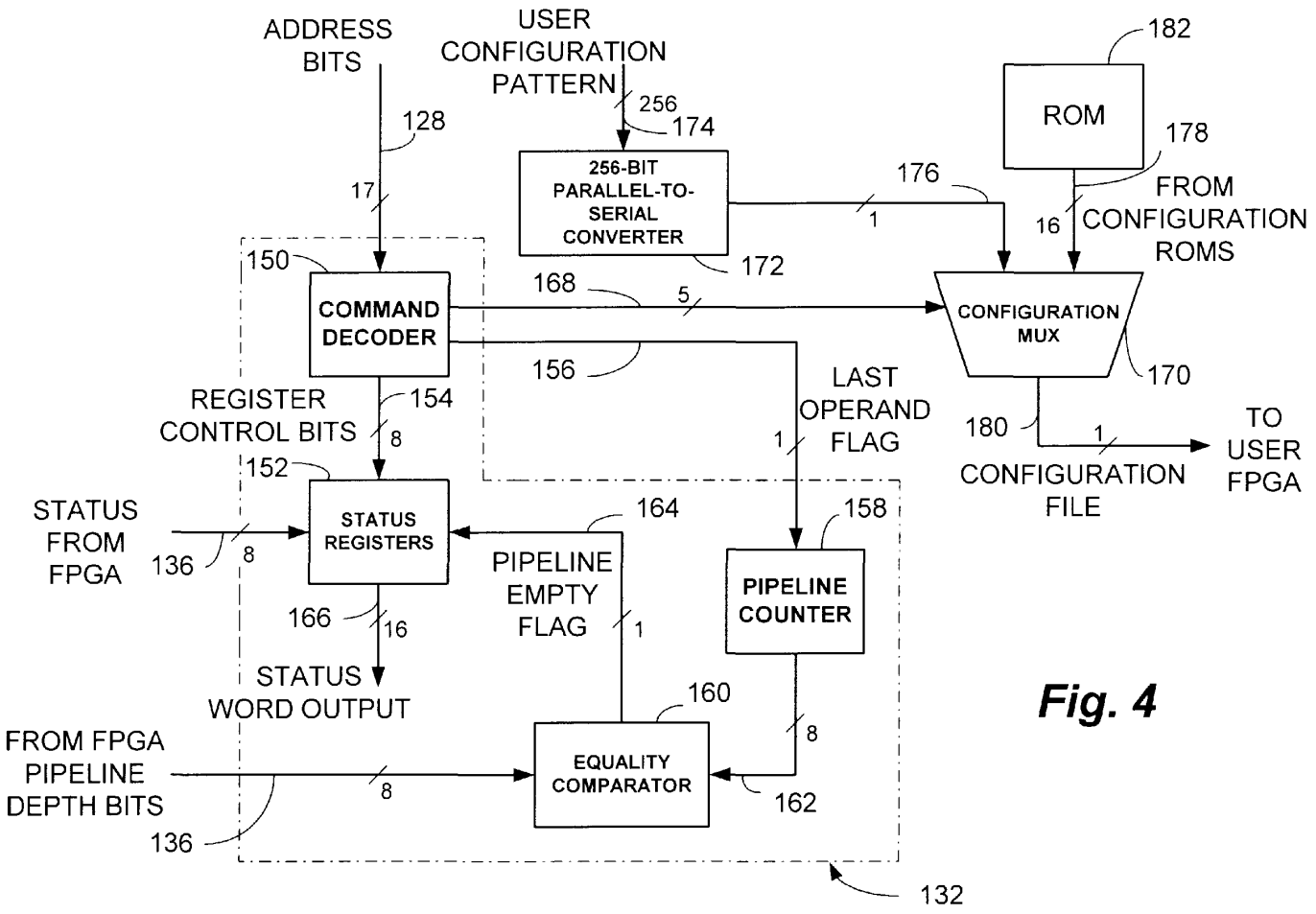

ADDRESS BITS
128
17
USER CONFIGURATION PATTERN
256
174
256-BIT PARALLEL-TO-SERIAL CONVERTER
172
176
1
ROM
182
178
16
FROM CONFIGURATION ROMS
150
COMMAND DECODER
168
5
CONFIGURATION MUX
170
156
LAST OPERAND FLAG
1
REGISTER CONTROL BITS
154
8
152
180
1
TO USER FPGA
CONFIGURATION FILE
STATUS FROM FPGA
136
8
STATUS REGISTERS
164
158
PIPELINE EMPTY FLAG
1
PIPELINE COUNTER
166
16
STATUS WORD OUTPUT
160
8
FROM FPGA PIPELINE DEPTH BITS
8
136
EQUALITY COMPARATOR
162
132

Fig. 4

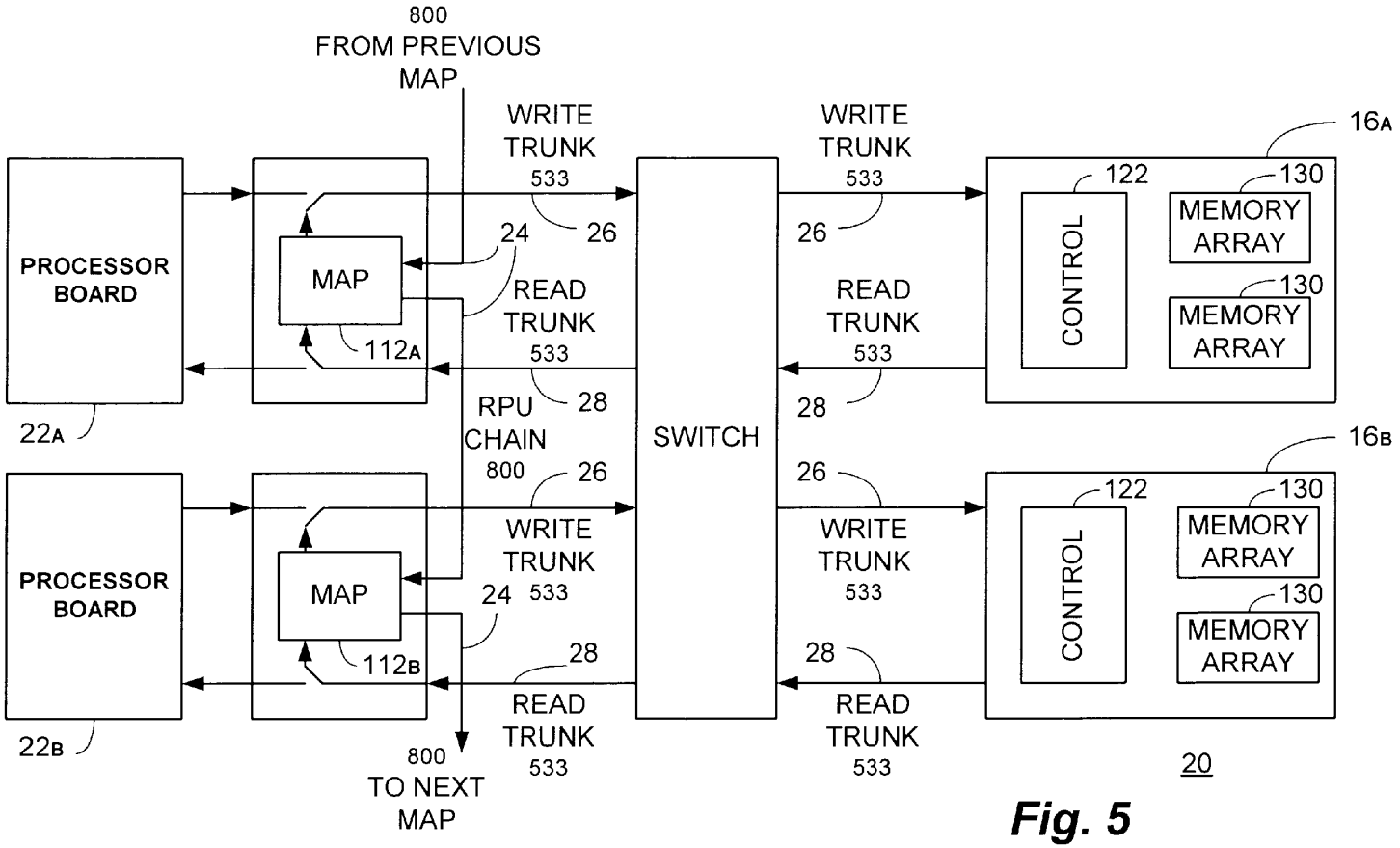

800
FROM PREVIOUS MAP
WRITE TRUNK 533
WRITE TRUNK 533
16A
PROCESSOR BOARD
MAP
24
26
26
122
130
MEMORY ARRAY
CONTROL
130
MEMORY ARRAY
READ TRUNK 533
READ TRUNK 533
112A
RPU CHAIN 800
28
28
22A
SWITCH
16B
26
26
PROCESSOR BOARD
MAP
24
WRITE TRUNK 533
WRITE TRUNK 533
112B
28
28
READ TRUNK 533
READ TRUNK 533
800
TO NEXT MAP
22B
20

Fig. 5

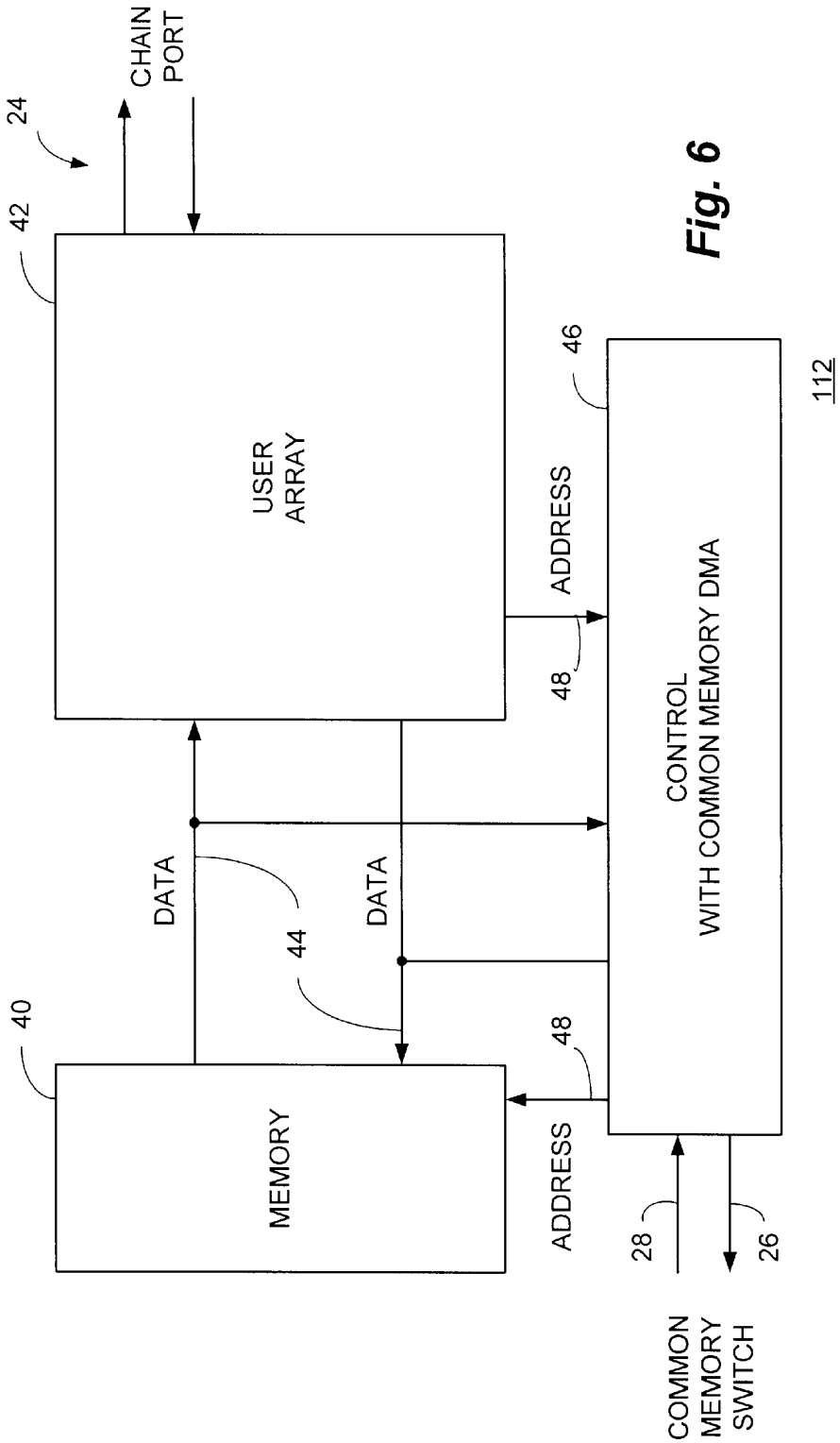

CHAIN PORT
24
42
USER ARRAY
46
112
ADDRESS
48
CONTROL WITH COMMON MEMORY DMA
DATA
DATA
44
40
48
MEMORY
ADDRESS
28
26
COMMON MEMORY SWITCH

Fig. 6

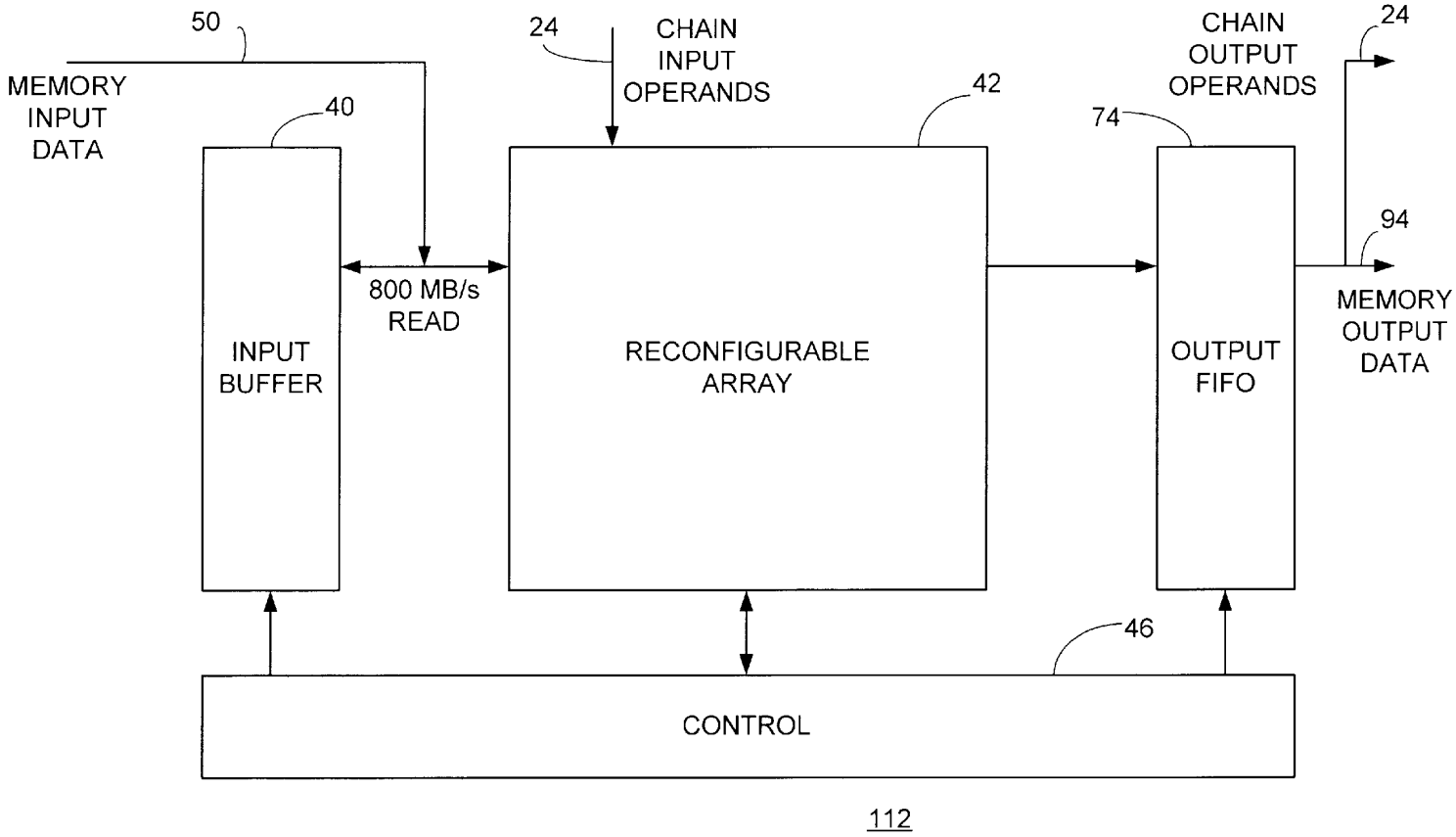

50
MEMORY INPUT DATA
40
24
CHAIN INPUT OPERANDS
42
74
CHAIN OUTPUT OPERANDS
24
94
800 MB/s READ
INPUT BUFFER
RECONFIGURABLE ARRAY
OUTPUT FIFO
MEMORY OUTPUT DATA
46
CONTROL
112

Fig. 7

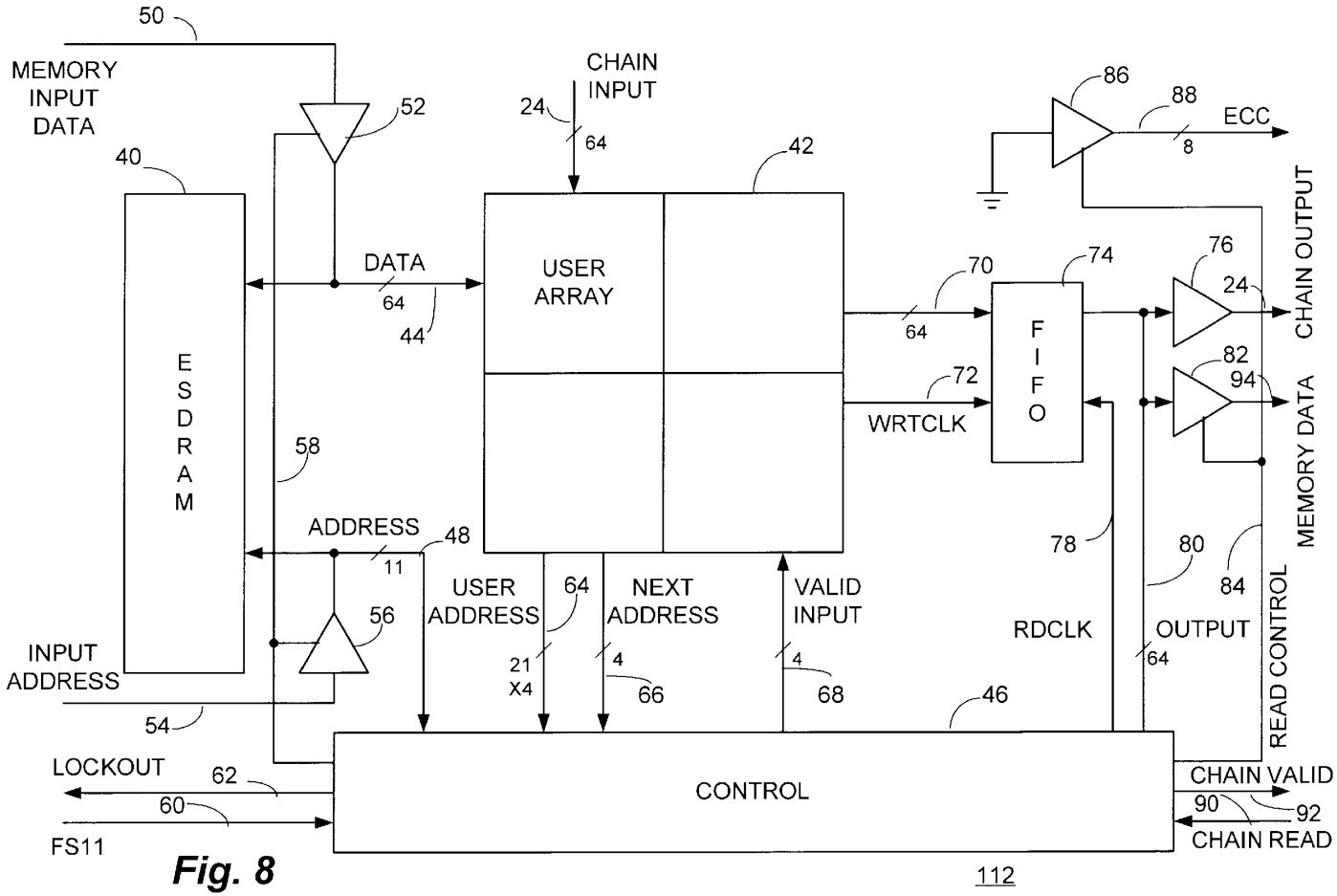

50
MEMORY INPUT DATA
52
CHAIN INPUT
24
64
42
86
88
ECC
8
40
DATA
64
44
USER ARRAY
70
74
76
24
CHAIN OUTPUT
64
F I F O
72
82
94
WRTCLK
MEMORY DATA
E S D R A M
58
ADDRESS
48
11
78
80
84
READ CONTROL
USER ADDRESS
64
NEXT ADDRESS
VALID INPUT
RDCLK
OUTPUT
64
56
INPUT ADDRESS
21 X4
4
66
4
68
46
54
LOCKOUT
62
60
FS11
CONTROL
CHAIN VALID
90
92
CHAIN READ
112

Fig. 8

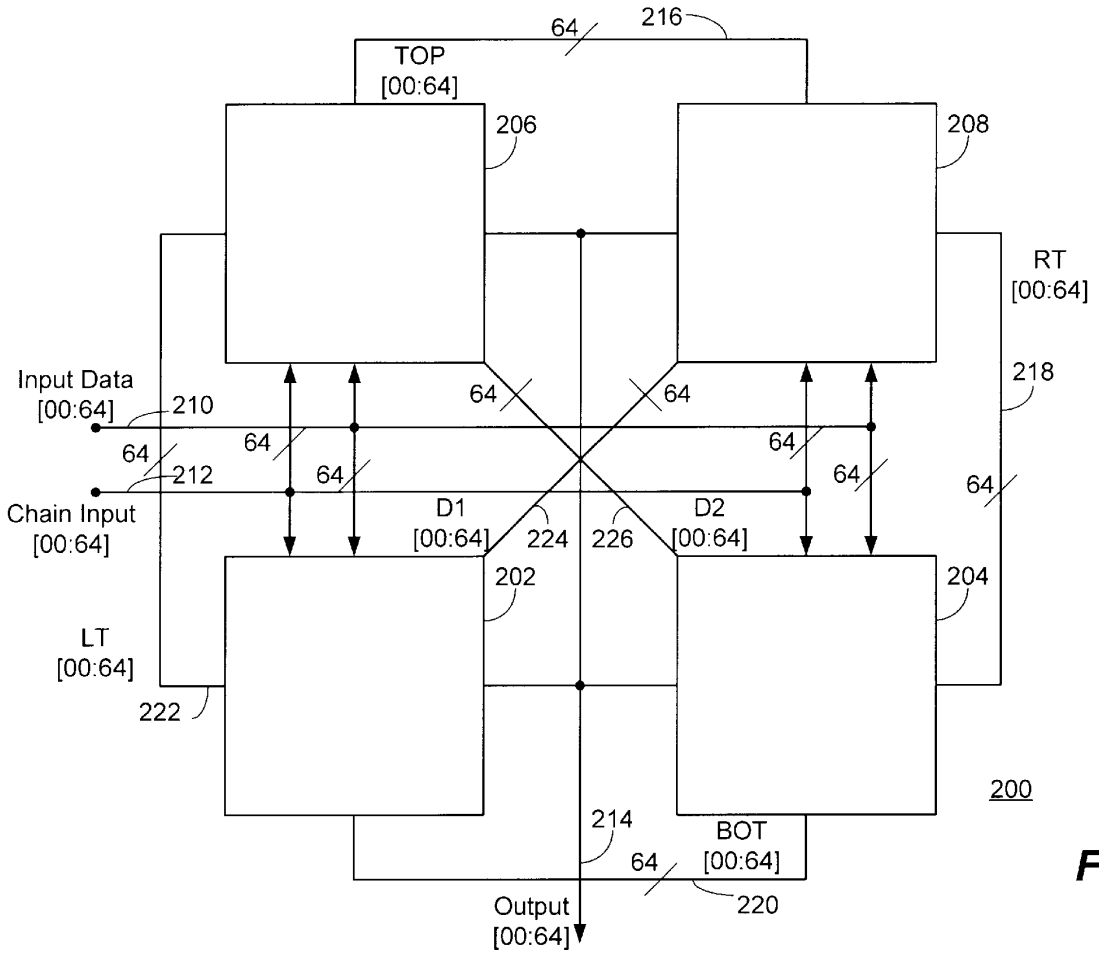

64
216
TOP
[00:64]
206
208
RT
[00:64]
218
Input Data
[00:64]
210
64
64
64
64
64
64
64
64
212
Chain Input
[00:64]
D1
[00:64]
224
226
D2
[00:64]
202
204
LT
[00:64]
222
200
214
BOT
[00:64]
64
220
Output
[00:64]

Fig. 9

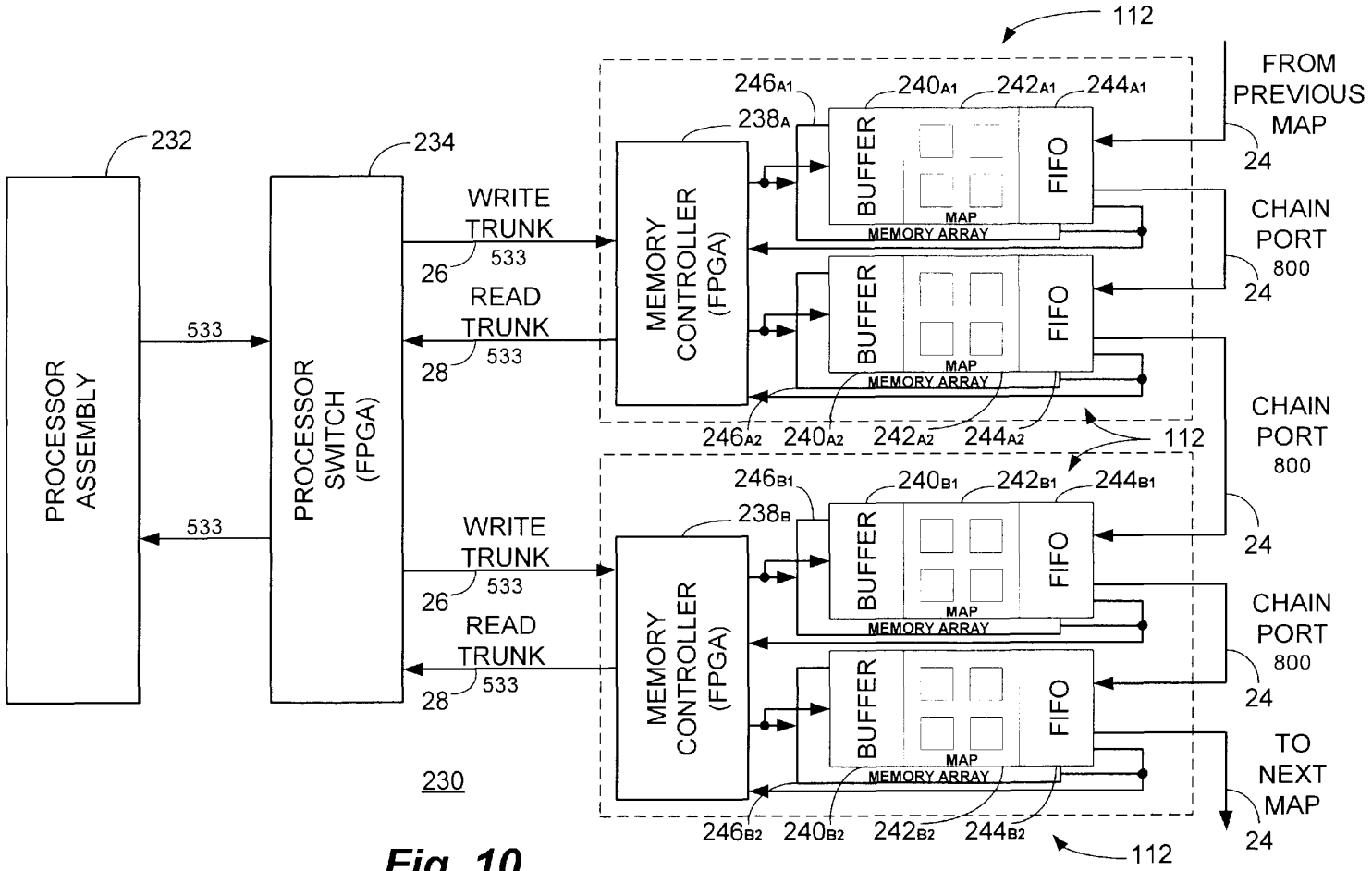

112
232
234
PROCESSOR ASSEMBLY
533
533
PROCESSOR SWITCH (FPGA)
WRITE TRUNK 533
26
READ TRUNK 533
28
MEMORY CONTROLLER (FPGA)
246A1
238A
240A1
242A1
244A1
BUFFER
FIFO
MAP MEMORY ARRAY
246A2
240A2
242A2
244A2
FROM PREVIOUS MAP
24
CHAIN PORT 800
24
CHAIN PORT 800
24
246B1
238B
240B1
242B1
244B1
CHAIN PORT 800
24
TO NEXT MAP
24
246B2
240B2
242B2
244B2
230

Fig. 10

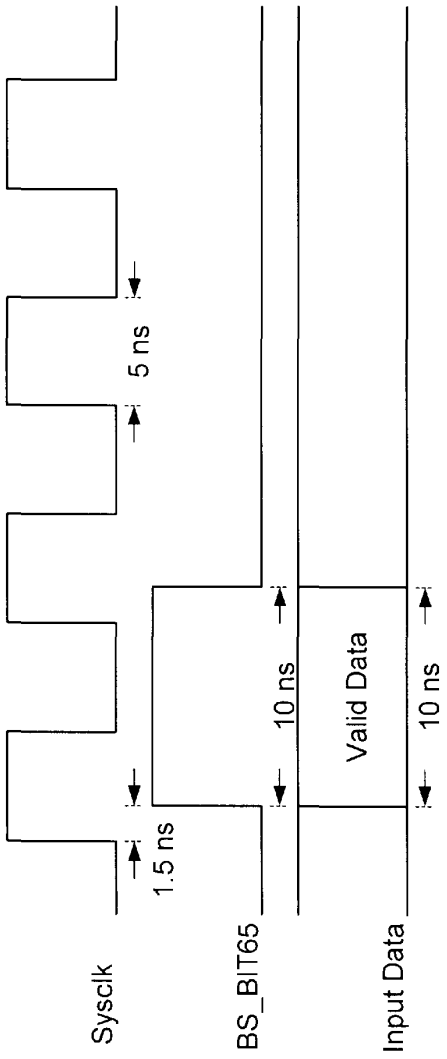


Fig. 11A

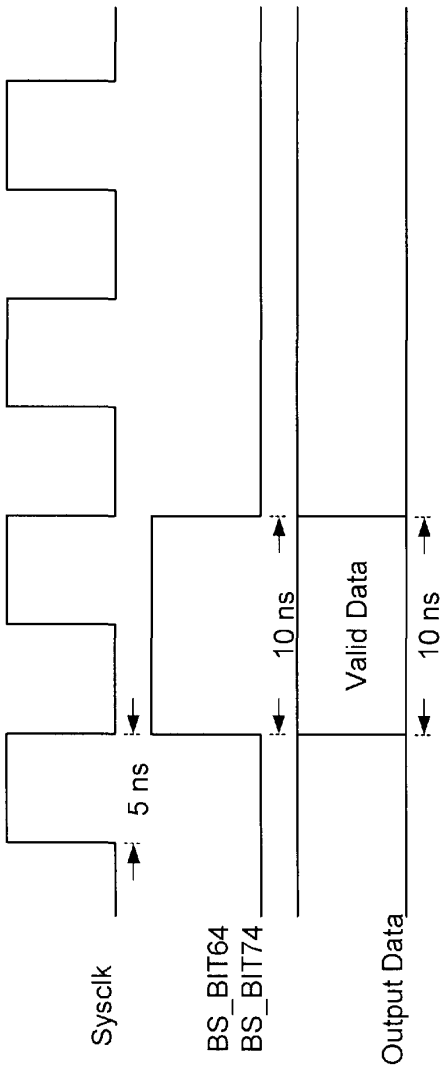


Fig. 11B

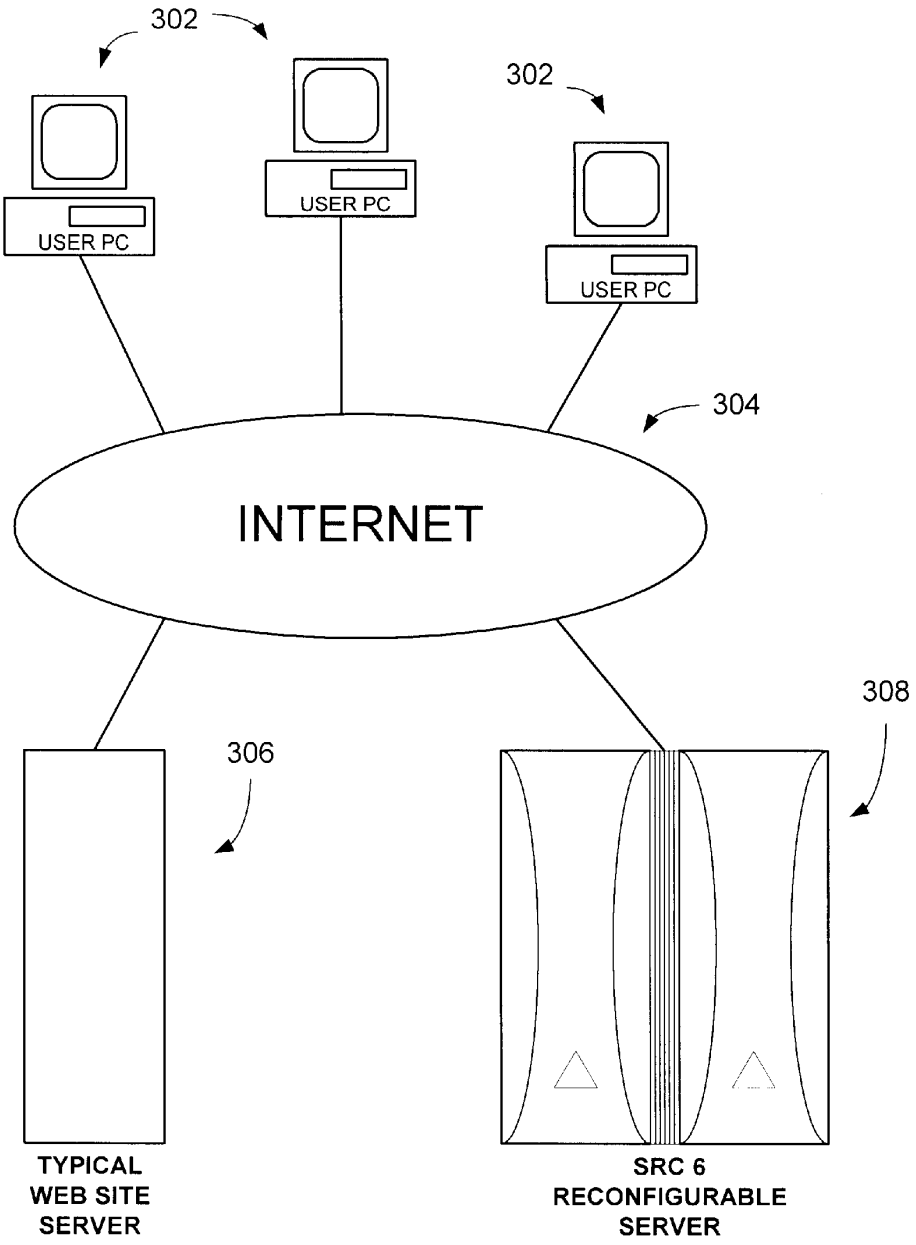

302
302
USER PC
USER PC
USER PC
304
INTERNET
306
308
TYPICAL
WEB SITE
SERVER
SRC 6
RECONFIGURABLE
SERVER
300

Fig. 12

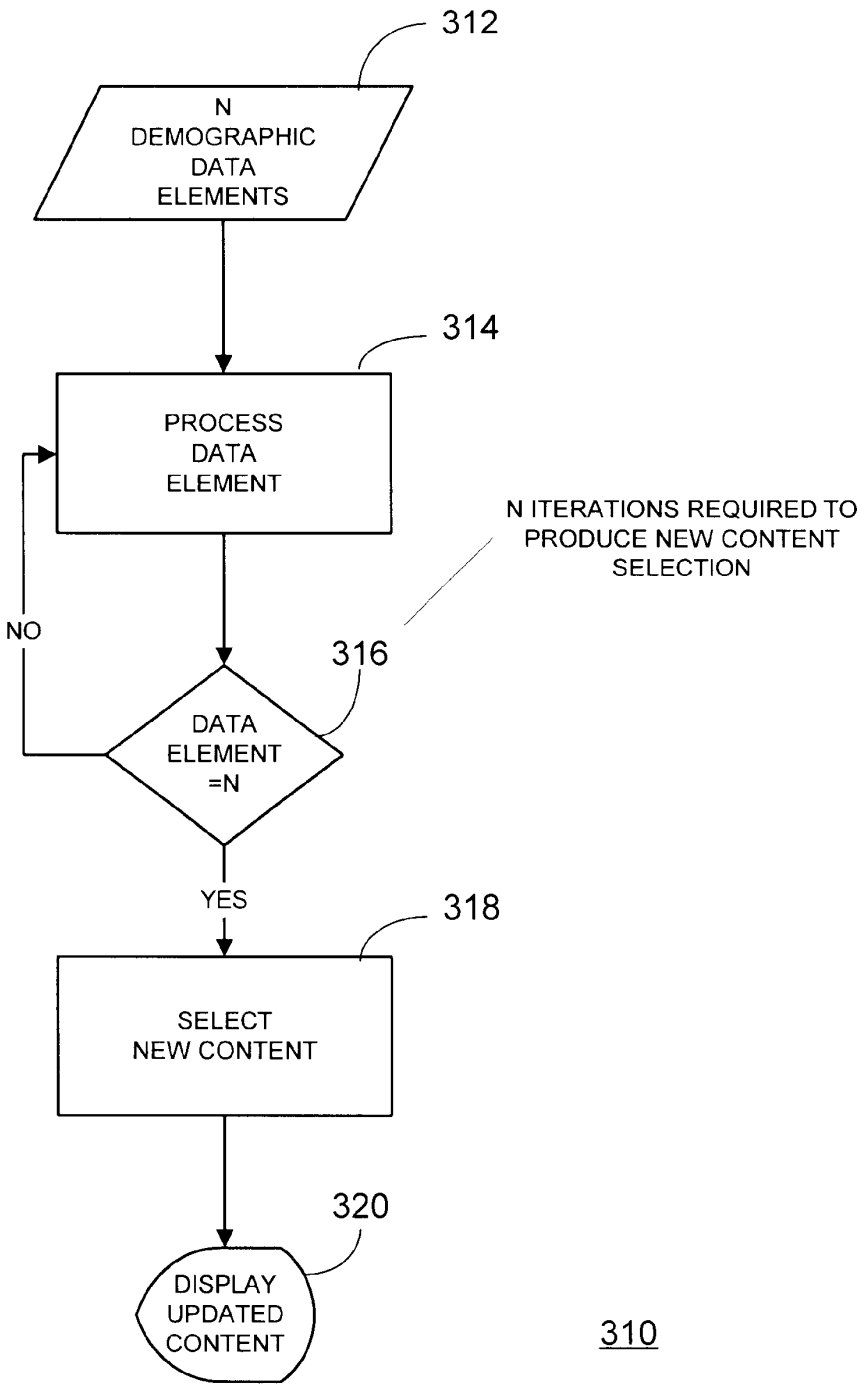


Fig. 13
Prior Art

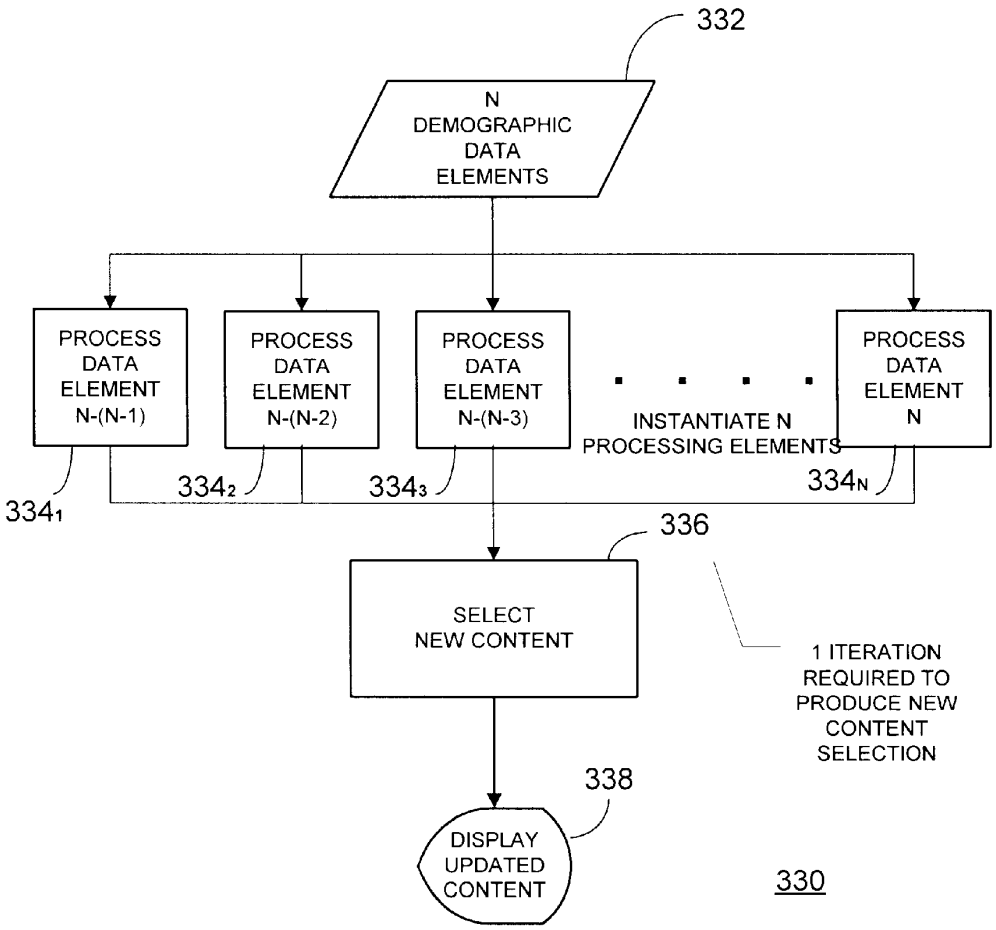

332
N DEMOGRAPHIC DATA ELEMENTS
PROCESS DATA ELEMENT N-(N-1)
PROCESS DATA ELEMENT N-(N-2)
PROCESS DATA ELEMENT N-(N-3)
INSTANTIATE N PROCESSING ELEMENTS
PROCESS DATA ELEMENT N
334_1
334_2
334_3
334_N
336
SELECT NEW CONTENT
1 ITERATION REQUIRED TO PRODUCE NEW CONTENT SELECTION
338
DISPLAY UPDATED CONTENT
330

Fig. 14

SYSTEM AND METHOD FOR ACCELERATING WEB SITE ACCESS AND PROCESSING UTILIZING A COMPUTER SYSTEM INCORPORATING RECONFIGURABLE PROCESSORS OPERATING UNDER A SINGLE OPERATING SYSTEM IMAGE

CROSS REFERENCE TO RELATED PATENT APPLICATIONS

The present invention is a continuation-in-part application of U.S. patent application Ser. No. 09/563,561 filed May 3, 2000, now issued U.S. Pat. No. 6,339,819 B1, which is a continuation-in-part application of U.S. patent application Ser. No. 09/481,902 filed Jan. 12, 2000, now issued U.S. Pat. No. 6,247,110 which is a continuation of U.S. patent application Ser. No. 08/992,763 filed Dec. 17, 1997 for: "Multiprocessor Computer Architecture Incorporating a Plurality of Memory Algorithm Processors in the Memory Subsystem", now issued U.S. Pat. No. 6,076,152 assigned to SRC Computers, Inc., Colorado Springs, Colo., assignee of the present invention, the disclosures of which are herein specifically incorporated by this reference.

BACKGROUND OF THE INVENTION

The present invention relates, in general, to the field of computer architectures incorporating multiple processing elements such as multi-adaptive processors ("MAP™", is a trademark of SRC Computers, Inc., Colorado Springs, Colo.) . More particularly, the present invention relates to systems and methods for accelerating web site access and processing utilizing a computer system incorporating reconfigurable processors operating under a single operating system image.

Presently, many different forms of electronic business and commerce are transacted by means of individual computers coupled to the Internet. By virtue of its computer-based nature, many electronic commerce ("e-commerce") web sites employ various methods to allow their content to be varied based on the demographics of the particular user.

This demographic information may be obtained in a variety of ways, with some sites simply requesting the site visitor respond to one or more questions while others may employ more sophisticated techniques such as "click stream" processing. In this latter instance, the prospective interests of the site visitor are inferred by determination and analysis of, for example, the previous sites he has visited. In either instance however, this data must be processed by the site such that the web page content may be altered in an effort to maximize it appeal to that particular site visitor with a view toward ultimately maximizing site revenue.

Since studies have shown that the average Internet user will wait but a maximum of twenty seconds or so for a web page to be updated, it is vitally important that the updating of the page contents be completed as rapidly as possible. Consequently, a great deal of effort is placed into maximizing the software performance of algorithms that process the user demographic data. Currently, all known web servers that accomplish this processing employ industry standard microprocessor based servers and, as a result, their maximum performance is thereby limited by the limitations inherent in the standard microprocessor "load/store" architecture.

SUMMARY OF THE INVENTION

SRC Computers, Inc., assignee of the present invention, is an industry leader in the design and development of multi-

processor computer systems including those employing industry standard processors together with multi-adaptive processors ("MAP™") utilizing, for example, field programmable gate arrays functioning as the programmable MAP elements.

Particularly disclosed herein is a system and method for accelerating web site access and processing utilizing a multiprocessor computer system incorporating one or more microprocessors and a number of reconfigurable processors operating under a single operating system image. In an exemplary embodiment, a web site may be serviced with a hybrid multiprocessor computer system that contains both industry standard microprocessors and one or more reconfigurable processors that share all the system's resources and operate under a single operating system image, (although, in an alternative embodiment, cluster management software may be used to make a cluster of microprocessors appear to the user as a single copy of the operating system). In such a system, demographic data processing algorithms may be loaded into the reconfigurable processors which may be provided in the form of specially adapted field programmable gate arrays ("FPGAs"). In this manner, the appropriate algorithm may be implemented in hardware gates (as opposed to software) which can process the data up to 1000 times faster than a standard microprocessor based server.

As an exemplary implementation, one particularly efficacious hybrid computing system is the SRC Computers, Inc. SRC-6 incorporating multi-adaptive processors (MAP). In such a system, the algorithms loaded into the MAP elements to process the data can be completely changed in under 100 msec. This allows for the possibility of quickly altering even the processing algorithm without significantly delaying the site visitor. The ability to change the algorithm, coupled with highly accelerated processing times allows for more complex algorithms to be employed leading to even more refined web page content adjustment.

Through the use of such a hybrid system operating under a single operating system image, a standard operating system, such as Solaris™ (trademark of Sun Microsystems, Inc., Palo Alto, Calif.) may be employed and can be easily administered, a feature which is important in such e-commerce based applications. Since the MAP elements are inherently tightly-coupled into the system and are not an attached processor located, for example, on an input/output ("I/O") port, their effectiveness and ease of use is maximized.

Demographic data processing is merely an example of how the unique capabilities of such reconfigurable processing systems can be utilized to accelerate e-commerce, and "secure socket" operation is yet another possible application. In this instance, such operations can often consume as much as 80% of the typical, traditional site server microprocessor cycles. SRC Computers, Inc. has demonstrated that reconfigurable processor based systems, such as the SRC-6, can perform decryption algorithms up to 1000 times faster than a conventional microprocessor thereby also allowing for faster web site access while concomitantly allowing more robust data encryption techniques to be employed. Similarly significant speed advantages could be realized in, for example, implementing database searches wherein the search algorithms can be directly implemented in the hardware of the reconfigurable system providing two to three orders of magnitude execution time improvements over conventional microprocessor based solutions.

In general, the use of hybrid computer systems with a single system image of the operating system for web site

hosting allows the site to employ user selected hardware accelerated versions of software algorithms currently implemented in a wide array of e-commerce related functions. This results in an easy to use system with significantly faster processing capability which translates into shorter site visitor waiting periods.

BRIEF DESCRIPTION OF THE DRAWINGS

The aforementioned and other features and objects of the present invention and the manner of attaining them will become more apparent and the invention itself will be best understood by reference to the following description of a preferred embodiment taken in conjunction with the accompanying drawings, wherein:

FIG. **1** is a simplified, high level, functional block diagram of a multiprocessor computer architecture employing multi-adaptive processors ("MAP™") in accordance with the disclosure of the aforementioned patent applications in an alternative embodiment wherein direct memory access ("DMA") techniques may be utilized to send commands to the MAP elements in addition to data;

FIG. **2** is a simplified logical block diagram of a possible computer application program decomposition sequence for use in conjunction with a multiprocessor computer architecture utilizing a number of MAP elements located, for example, in the computer system memory space, in accordance with a particular embodiment of the present invention;

FIG. **3** is a more detailed functional block diagram of an exemplary individual one of the MAP elements of the preceding figures and illustrating the bank control logic, memory array and MAP assembly thereof;

FIG. **4** is a more detailed functional block diagram of the control block of the MAP assembly of the preceding illustration illustrating its interconnection to the user FPGA thereof in a particular embodiment;

FIG. **5** is a functional block diagram of an alternative embodiment of the present invention wherein individual MAP elements are closely associated with individual processor boards and each of the MAP elements comprises independent chain ports for coupling the MAP elements directly to each other;

FIG. **6** is a functional block diagram of an individual MAP element wherein each comprises on board memory and a control block providing common memory DMA capabilities;

FIG. **7** is an additional functional block diagram of an individual MAP element illustrating the on board memory function as an input buffer and output FIFO portions thereof;

FIG. **8** is a more detailed functional block diagram of an individual MAP element as illustrated in FIGS. **6** and **7**;

FIG. **9** is a user array interconnect diagram illustrating, for example, four user FPGAs interconnected through horizontal, vertical and diagonal buses to allow for expansion in designs that exceed the capacity of a single FPGA;

FIG. **10** is a functional block diagram of another alternative embodiment of the present invention wherein individual MAP elements are closely associated with individual memory arrays and each of the MAP elements comprises independent chain ports for coupling the MAP elements directly to each other;

FIGS. **11**A and **11**B are timing diagrams respectively illustrating input and output timing in relationship to the system clock ("Sysclk") signal

FIG. **12** is a simplified illustration of a representative operating environment for the system and method of the present invention including a typical web site server as would be replaced by an SRC-6 reconfigurable server;

FIG. **13** is a flowchart illustrating a conventional data processing sequence in a conventional application of the typical web site server depicted in the preceding figure. and

FIG. **14** is a corresponding flowchart illustrating the processing of demographic or other data utilizing a reconfigurable server for implementing the system and method of the present invention and which results in significantly improved access and data processing times.

DESCRIPTION OF A PREFERRED EMBODIMENT

With reference now to FIG. **1**, a multiprocessor computer **10** architecture in accordance with one embodiment of the present invention is shown. The multiprocessor computer **10** incorporates N processors $\mathbf{12}_0$ through $\mathbf{12}_N$ which are bi-directionally coupled to a memory interconnect fabric **14**. The memory interconnect fabric **14** is then also coupled to M memory banks comprising memory bank subsystems $\mathbf{16}_0$ (Bank **0**) through **16**M (Bank M). N number of multi-adaptive processors ("MAP™") $\mathbf{112}_0$ through $\mathbf{112}_N$ are also coupled to the memory interconnect fabric **14** as will be more fully described hereinafter.

With reference now to FIG. **2**, a representative application program decomposition for a multiprocessor computer architecture **100** incorporating a plurality of multi-adaptive processors in accordance with the present invention is shown. The computer architecture **100** is operative in response to user instructions and data which, in a coarse grained portion of the decomposition, are selectively directed to one of (for purposes of example only) four parallel regions $\mathbf{102}_1$ through $\mathbf{102}_4$ inclusive. The instructions and data output from each of the parallel regions $\mathbf{102}_1$ through $\mathbf{102}_4$ are respectively input to parallel regions segregated into data areas $\mathbf{104}_1$ through $\mathbf{104}_4$ and instruction areas $\mathbf{106}_1$ through $\mathbf{106}_4$. Data maintained in the data areas $\mathbf{104}_1$ through $\mathbf{104}_4$ and instructions maintained in the instruction areas $\mathbf{106}_1$ through $\mathbf{106}_4$ are then supplied to, for example, corresponding pairs of processors $\mathbf{108}_1$, $\mathbf{108}_2$ (P**1** and P**2**); $\mathbf{108}_3$, $\mathbf{108}_4$ (P**3** and P**4**); $\mathbf{108}_5$, $\mathbf{108}_6$ (P**5** and P**6**); and $\mathbf{108}_7$, $\mathbf{108}_8$ (P**7** and P**8**) as shown. At this point, the medium grained decomposition of the instructions and data has been accomplished.

A fine grained decomposition, or parallelism, is effectuated by a further algorithmic decomposition wherein the output of each of the processors $\mathbf{108}_1$ through $\mathbf{108}_8$, is broken up, for example, into a number of fundamental algorithms $^{110}{}_{1A}$, $\mathbf{110}_{1B}$, $\mathbf{110}_{2A}$, $\mathbf{110}_{2B}$ through $\mathbf{110}_{8B}$ as shown. Each of the algorithms is then supplied to a corresponding one of the MAP elements $\mathbf{112}_{1A}$, $\mathbf{112}_{1B}$, $\mathbf{112}_{2A}$, $\mathbf{112}_{2B}$, through $\mathbf{112}_{8B}$ which may be located in the memory space of the computer architecture **100** for execution therein as will be more fully described hereinafter.

With reference additionally now to FIG. **3**, an exemplary implementation of a memory bank **120** in a MAP system computer architecture **100** of the present invention is shown for a representative one of the MAP elements **112** illustrated in the preceding figure. Each memory bank **120** includes a bank control logic block **122** bi-directionally coupled to the computer system trunk lines, for example, a **72** line bus **124**. The bank control logic block **122** is coupled to a bi-directional data bus **126** (for example 256 lines) and supplies addresses on an address bus **128** (for example 17 lines) for accessing data at specified locations within a memory array **130**.

The data bus **126** and address bus **128** are also coupled to a MAP element **112**. The MAP element **112** comprises a control block **132** coupled to the address bus **128**. The control block **132** is also bi-directionally coupled to a user field programmable gate array ("FPGA") **134** by means of a number of signal lines **136**. The user FPGA **134** is coupled directly to the data bus **126**. In a particular embodiment, the FPGA **134** may be provided as a Lucent Technologies OR3T80 device.

The computer architecture **100** comprises a multiprocessor system employing uniform memory access across common shared memory with one or more MAP elements **112** which may be located in the memory subsystem, or memory space. As previously described, each MAP element **112** contains at least one relatively large FPGA **134** that is used as a reconfigurable functional unit. In addition, a control block **132** and a preprogrammed or dynamically programmable configuration ROM (as will be more fully described hereinafter) contains the information needed by the reconfigurable MAP element **112** to enable it to perform a specific algorithm. It is also possible for the user to directly download a new configuration into the FPGA **134** under program control, although in some instances this may consume a number of memory accesses and might result in an overall decrease in system performance if the algorithm was short-lived.

FPGAs have particular advantages in the application shown for several reasons. First, commercially available FPGAs now contain sufficient internal logic cells to perform meaningful computational functions. Secondly, they can operate at speeds comparable to microprocessors, which eliminates the need for speed matching buffers. Still further, the internal programmable routing resources of FPGAs are now extensive enough that meaningful algorithms can now be programmed without the need to reassign the locations of the input/output ("1/0") pins.

By, for example, placing the MAP element **112** in the memory subsystem or memory space, it can be readily accessed through the use of memory read and write commands, which allows the use of a variety of standard operating systems. In contrast, other conventional implementations-may propose placement of any reconfigurable logic in or near the processor, however these conventional implementations are generally much less effective in a multiprocessor environment because, unlike the system and method of the present invention, only one processor has rapid access to it. Consequently, reconfigurable logic must be placed by every processor in a multiprocessor system, which increases the overall system cost. In addition, MAP element **112** can access the memory array **130** itself, referred to as Direct Memory Access ("DMA"), allowing it to execute tasks independently and asynchronously of the processor. In comparison, were it placed near the processor, it would have to compete with the processors for system routing resources in order to access memory, which deleteriously impacts processor performance. Because MAP element **112** has DMA capability, (allowing it to write to memory), and because it receives its operands via writes to memory, it is possible to allow a MAP element **112** to feed results to another MAP element **112**. This is a very powerful feature that allows for very extensive pipelining and parallelizing of large tasks, which permits them to complete faster.

Many of the algorithms that may be implemented will receive an operand and require many clock cycles to produce a result. One such example may be a multiplication that takes 64 clock cycles. This same multiplication may also

need to be performed on thousands of operands. In this situation, the incoming operands would be presented sequentially so that while the first operand requires 64 clock cycles to produce results at the output, the second operand, arriving one clock cycle later at the input, will show results one clock cycle later at the output. Thus, after an initial delay of 64 clock cycles, new output data will appear on every consecutive clock cycle until the results of the last operand appears. This is called "pipelining".

In a multiprocessor system, it is quite common for the operating system to stop a processor in the middle of a task, reassign it to a higher priority task, and then return it, or another, to complete the initial task. When this is combined with a pipelined algorithm, a problem arises (if the processor stops issuing operands in the middle of a list and stops accepting results) with respect to operands already issued but not yet through the pipeline. To handle this issue, a solution involving the combination of software and hardware is disclosed herein.

To make use of any type of conventional reconfigurable hardware, the programmer could embed the necessary commands in his application program code. The drawback to this approach is that a program would then have to be tailored to be specific to the MAP hardware. The system of the present invention eliminates this problem. Multiprocessor computers often use software called parallelizers. The purpose of this software is to analyze the user's application code and determine how best to split it up among the processors. The present invention provides significant advantages over a conventional parallelizer and enables it to recognize portions of the user code that represent algorithms that exist in MAP elements **112** for that system and to then treat the MAP element **112** as another computing element. The parallelizer then automatically generates the necessary code to utilize the MAP element **112**. This allows the user to write the algorithm directly in his code, allowing it to be more portable and reducing the knowledge of the system hardware that he has to have to utilize the MAP element **112**.

With reference additionally now to FIG. **4**, a block diagram of the MAP control block **132** is shown in greater detail. The control block **132** is coupled to receive a number of command bits (for example, **17**) from the address bus **128** at a command decoder **150**. The command decoder **150** then supplies a number of register-control bits to a group of status registers iS2 on an eight bit bus **154**. The command decoder **150** also supplies a single bit last operand flag on line **156** to a pipeline counter **158**. The pipeline counter **158** supplies an eight bit output to an equality comparitor **160** on bus **162**. The equality comparitor **160** also receives an eight bit signal from the FPGA **134** on bus **136** indicative of the pipeline depth. When the equality comparitor **160** determines that the pipeline is empty, it provides a single bit pipeline empty flag on line **164** for input to the status registers **152**. The status registers **152** are also coupled to receive an eight bit status signal from the FPGA **134** on bus **136**and it produces a sixty four bit status word output on bus **166** in response to the signals on bus **136**, **154** and line **164**.

The command decoder **150** also supplies a five bit control signal on line **168** to a configuration multiplexer ("MUX") **170** as shown. The configuration MUX **170** receives a single bit output of a 256 bit parallel-serial converter **172** on line **176**. The inputs of the 256 bit parallel-to-serial converter **172** are coupled to a 256 bit user configuration pattern bus **174**. The configuration MUX **170** also receives sixteen single bit inputs from the configuration ROMs (illustrated as ROM **182**) on bus **178** and provides a single bit configuration file signal on line **180** to the user FPGA **134** as selected by the control signals from the command decoder **150** on the bus **168**.

In operation, when a processor **108** is halted by the operating system, the operating system will issue a last operand command to the MAP element **112** through the use of command bits embedded in the address field on bus **128**. This command is recognized by the command decoder **150** of the control block **132** and it initiates a hardware pipeline counter **158**. When the algorithm was initially loaded into the FPGA **134**, several output bits connected to the control block **132** were configured to display a binary representation of the number of clock cycles required to get through its pipeline (i.e. pipeline "depth") on bus **136** input to the equality comparitor **160**. After receiving the last operand command, the pipeline counter **158** in the control block **132** counts clock cycles until its count equals the pipeline depth for that particular. algorithm. At that point, the equality comparitor **160** in the control block **132** de-asserts a busy bit on line **164** in an internal group of status registers **152**. After issuing the last operand signal, the processor **108** will repeatedly read the status registers **152** and accept any output data on bus **166**. When the busy flag is de-asserted, the task can be stopped and the MAP element **112** utilized for a different task. It should be noted that it is also possible to leave the MAP element **112** configured, transfer the program to a different processor **108** and restart the task where it left off.

In order to evaluate the effectiveness of the use of the MAP element **112** in a given application, some form of feedback to the use is required. Therefore, the MAP element **112** may be equipped with internal registers in the control block **132** that allow it to monitor efficiency related factors such as the number of input operands versus output data, the number of idle cycles over time and the number of system monitor interrupts received over time. One of the advantages that the MAP element **112** has is that because of its reconfigurable nature, the actual function and type of function that are monitored can also change as the algorithm changes. This provides the user with an almost infinite number of possible monitored factors without having to monitor all factors all of the time.

With reference additionally now to FIG. **5**, a functional block diagram of a portion of an alternative embodiment of a computer system **20** in accordance with the of the present invention is shown. In the computer system **20** illustrated, individual MAP elements 112_A, 112_B etc. are each closely associated with individual processor boards 22_A, 22_B respectively. As depicted, each of the MAP elements **112** comprises independent chain ports **24** for coupling the MAP elements **112** directly to each other.

Individual ones of the MAP elements **112** are coupled between the processor board **22** write trunk **26** and read trunk **28** of each processor board **22** in addition to their coupling to each other by means of the chain ports **24**. A switch couples the write trunk **26** and read trunk **28** of any given processor board to any other memory subsystem bank 16_A, 16_B etc. As generally illustrated, each of the memory subsystem banks **16** includes a control-block **122** and one or more memory arrays **130**.

With reference additionally now to FIG. **6**, a functional block diagram of an individual MAP element **112** is shown wherein each MAP element **112** comprises an on board memory **40** and a control block **46** providing common memory DMA capabilities. Briefly, the write trunk **26** and read trunk **28** are coupled to the control block **46** from the common memory switch which provides addresses to the memory **40** and receives addresses from the user array **42** on address lines **48**. Data supplied on the write trunk **26** is provided by the control block **46** to the memory **40** on data lines **44** and data read out of the memory **40** is provided on these same lines both to the user array **42** as well as the control block **46** for subsequent presentation on the read trunk **28**. As indicated, the chain port **24** is coupled to the user array **42** for communication of read and write data directly with other MAP elements **112**.

With reference additionally now to FIG. **7**, an additional functional block diagram of an individual MAP element **112** is shown particularly illustrating the memory **40** of the preceding figure functioning as an input buffer **40** and output FIFO **74** portions thereof. In this figure, an alternative view of the MAP element **112** of FIG. **6** is shown in which memory input data on line **50** (or the write trunk **26**) is supplied to an input buffer (memory **40**) as well as to a reconfigurable user array **42** coupled to the chain port **24**. The output of the reconfigurable array **42** is supplied to an output FIFO **74** to provide memory output data on line **94** (or the read trunk **28**) as well as to the chain port **24**. The input buffer **40**, reconfigurable array **42** and output FIFO **74** operate under the control of the control block **46**.

With respect to the foregoing figures, each MAP element **112** may consist of a printed circuit board containing input operand storage (i.e. the memory/input buffer **40**), user array **42**, intelligent address generator control block **46**, output result storage FIFO **74** and I/O ports to allow connections to other MAP elements **112** through the chain port **24** as well as the host system memory array.

Input Operand Storage

The input storage consists of memory chips that are initially loaded by memory writes from one of the microprocessors **12** in the host system or by MAP DMA. The buffer **40** may be, in a particular embodiment, **72** bits wide and 2M entries deep. This allows for storage of 64 bit operands and **8** error correction code ("ECC") bits for data correction if needed. Operands or reference data can be read from this buffer **40** by the user array **42**. Data is not corrupted after use allowing for operand reuse by the MAP elements **112**. By reading operands only after the buffer **40** is loaded, operands do not need to arrive at the MAP elements **112** in time order. MAP elements **112** only require that store order be maintained thus allowing for out-of-order arrival of operands prior to storage in the input buffer **40**. This means cache line transfers, which typically can not be performed in a timed order but have four times the bandwidth of un-cached transfers, can be used to load the input buffers **40**.

Intelligent Address Generator

The input buffer **40** contents are accessed by providing address and read enable signals to it from the control block **46**. These addresses may be generated in one of two ways. First the address bits can be provided by the programmable user array **42** to the address generator control block **46** where it is combined with other control signals and issued to the input buffer **40**. This allows for very random access into the buffer **40** such as would be needed to access reference data. Another address mode requires the user to issue a start command which contains a start address, stop address, and stride. The address generator control block **46** will then start accessing the input buffer **40** at the start address and continue accessing it by adding the stride value to the last address sent until the stop address is reached. This is potentially a very useful technique when performing vector processing where like elements are extracted out of an array. Since the stride can be any number less than the delta between the start and stop addresses, it is very easy for the MAP element **112** to perform a data gather function which is highly valuable in the high performance computing market.

User Array

The array **42** performs the actual computational functions of the MAP element **112**. It may comprise one or more high performance field programmable gate arrays ("FPGAs") interconnected to the other elements of the MAP element **112**. A particular implementation of the present invention disclosed in more detail hereinafter, may use four such devices yielding in excess of 500,000 usable gates. These components are configured by user commands that load the contents of selected configuration ROMs into the FPGAs. After configuration, the user array **42** can perform whatever function it was programmed to do. In order to maximize its performance for vector processing, the array **42** should be able to access two streams of operands simultaneously. This is accomplished by connecting one 72 bit wide input port to the input operand storage and a second 72 bit wide port to the chain input connector port **24**. This connector allows the MAP element **112** to use data provided to it by a previous MAP element **112**. The chain port **24** allows functions to be implemented that would far exceed the capability of a single MAP element **112** assembly. In addition, since in the particular implementation shown, only operands are transferred over the chain port **24**, the bandwidth may exceed the main memory bandwidth resulting in superior performance to that of the fixed instruction microprocessor-based processors **12**.

The FPGAs may also contain on board phase locked loops ("PLLs") that allow the user to specify at what multiple or sub-multiple of the system clock frequency the circuit will run. This is important because certain complex functions may require clocks that are slower than the system clock frequency. It may also be that the user desires to synthesize a function resulting in lower performance but faster time to market. By using PLLs, both of these constraints can be accommodated. Another benefit in the potential utilization of a PLL is that future generation FPGAs that can operate faster than the current system clock speeds can be retrofitted into slower systems and use the PLL frequency multiplication feature to allow the MAP element **112** to run faster than the rest of the system. This is turn results in a higher performance MAP element **112**.

Output Result Storage

When the user array **42** produces a result, it may be sent over a 72 bit wide path to an output result storage element (for example, output FIFO **74**) which can then pass the data to either a 72 bit wide read port or a 72 bit wide chain port **24** to the next MAP element **112**. This storage device can made from a number of different memory types. The use of a FIFO **74** storage device will temporarily hold results that cannot be immediately read by a host microprocessor or passed over the output chain port **24** to the next stage. This feature allows for MAP elements **112** in a chain to run at different frequencies. In this case the output FIFO **74** functions like a speed matching buffer. In non-chained operation, the microprocessor that is reading the results may be delayed. In this case the FIFO **74** prevents the MAP element **112** from "stalling" while waiting for results to be read. In a particular embodiment of the present invention, a FIFO **74** that is 72 bits wide and 512K entries deep may be utilized. As disclosed in the aforementioned patent applications, the output storage may also be a true memory device such as those found in common memory. In this case, write addresses must be provided by the user array **42** or address generator and read addresses provided by the entity reading the results from the memory. While this may be somewhat more electrically complicated, it has the advantage that results may be accessed in any order.

DMA Enhancements

In the aforementioned patent applications, the ability of MAP elements **112** to perform DMA to common memory was disclosed. While this capability was discussed primarily with respect to the movement of operands and results, it is also possible to apply the same concept to commands. The microprocessor that would normally write a series of commands directly to the MAP element **112** may also write the same commands into common memory as well. After writing a series of commands, the microprocessor could then send an interrupt to the MAP element **112**. The MAP element **112** would then read the commands from common memory and execute them as contemplated. Since this command list could contain DMA instructions as specified in the previously mentioned patent applications, the MAP element **112** could retrieve all of its input operands and store all of its results without any further processor **12** intervention. At the completion of MAP element **112** processing, the MAP element **112** could then interrupt the microprocessor to signal that results are available in common memory. Operation in this manner reduces the interaction required between the MAP element **112** and the microprocessor.

On Board Library

As originally disclosed, electrically erasable programmable ROMs ("EEPROMs") or similar devices may be utilized to hold a library of functions for the user array **42**. By placing these algorithms in ROMs on the MAP element **112** itself, the user array **42** function can be changed very rapidly. In this manner, the user program can download a new function into one of the on board ROMs thus updating its contents and allowing the MAP element **112** to perform new functions. In a particular implementation, this may be accomplished by reserving one of the library functions to perform the function of an EEPROM programmer. When a command to update a ROM is received, the user array **42** may be configured with this special function and data read from the MAP element **112** input storage (e.g. input buffer **40**) and then loaded into the ROMs to complete the update process.

With reference additionally now to FIG. **8** a more detailed functional block diagram of an individual MAP element **112** is shown as previously illustrated in FIGS. **6** and **7**. In this depiction, the MAP element **112** includes an enhanced synchronous dynamic random access memory (ESDRAM™, a trademark of Enhanced Memory Systems, Inc., Colorado Springs, Colo.) functioning as the memory, or input buffer **40**. ESDRAM memory is a very high speed memory device incorporating a dynamic random access memory ("DRAM") array augmented with an on-chip static random access memory ("SRAM") row register to speed device read operations.

In this figure, like structure to that previously described is like numbered and the foregoing description thereof shall suffice herefor. Memory input data on lines **50** is supplied through transmission gates **52** to the data lines **44** for provision to the memory **40** and user array **42**. In like manner, address input is received on lines **54** for provision through transmission gates **56** to the address lines **48** coupled to the memory **40** and control block **46**. The control block **46** operatively controls the transmission gates **52**, **56** and receives an FS**11** signal on line **60** and provides a LOCKOUT signal on line **62**.

The user array **42** may be coupled, as shown, to the chain port **24** and it provides a user address signal on lines **64** and a next address signal on lines **66** to the control block **46**. The control block **46**, provides an indication of whether or not an input is valid to the user array **42** on lines **68**. Output of the

user array **42** is provided on lines **70** together with a write clock ("WRTCLK") signal on line **72** to the FIFO **74** or other output storage device. The FIFO **74** receives a read clock ("RDCLK") signal on line **78** from the control block **46**. Output from the FIFO **74** or control block **46** may be selectively supplied on lines **80** through transmission gates **76** to the chain port **24** and/or through transmission gates **82** to provide memory data on lines **94**. The control block **46** also receives a chain read signal on lines **90** and returns a chain valid output on lines **92**. The control block **46** operatively controls the transmission gates **76** and **82** in addition to transmission gates **86** which serve to provide error correction code ("ECC") output signals on lines **88**.

As mentioned previously, the MAP elements **112** may comprise one or more circuit boards, utilizing, for example, one Lucent Orca™ OR3T80 FPGA to function as the control block **46** and, four OR3TI25 FPGAs forming the user array **42**. The user can implement algorithms in these FPGAs that alter data that is written to it and provide this altered data when the MAP element **112** is then read. In addition, each MAP element **112** may also comprise eight sets of four configuration ROMs on board. These ROMs are preprogrammed by the user and configure the four user FPGAs of the user array **42** under program control. These ROMs may be reprogrammed either externally or while on the MAP element **112** located in a system.

The MAP elements **112** are accessed through the use of normal memory READ and WRITE commands. In the representative embodiment illustrated and described, the user can provide operands to the MAP elements **112** either by directly writing 128-bit packets (i.e. in the form of two 64-bit words) into the user array **42** chips or by writing 256-bit packets (in the form of our 64-bit words) into a dedicated 16-MB ESDRAM memory input data buffer **40**. A read from a MAP element **112** always returns a 2-word packet and part of this returned packet contains status information as will be more fully described hereinafter. In addition, the incoming addresses are decoded into commands as will also be defined later.

MAP elements **112** also have the ability to be chained via hardware. This allows the output data from one MAP element **112** to move directly to the user array **42** chips of the next MAP element **112** without processor **12** intervention. Chain length is limited by the quantity of MAP elements **112** in the overall system. The total number of MAP elements **112** may also be broken down into several smaller independent chains. In a chained mode of operation, a MAP element **112** can still read from its input buffer **40** to access reference information such as reciprocal approximation tables.

Logic Conventions

In the representative implementation of the computer system of the present invention disclosed herein, the processors **12** may comprise Pentium™ (a trademark of Intel Corporation, Santa Clara, Calif.) processors and these devices utilize an active "low" logic convention which applies to all address bits and data words transmitted to or from the MAP elements **112** including the returned status word.

With reference additionally now to FIG. **9**, a user array interconnect **200** diagram is shown, for example, utilizing four user FPGAs interconnected through horizontal, vertical and diagonal buses to allow for expansion in designs that might exceed the capacity of a single FPGA. In this regard, the interconnect diagram **200** corresponds to the user array **42** of the preceding figures with input data bus **210** corresponding to the data lines **44**, the chain input bus **212** corresponding to the chain port **24** and the output bus **214** corresponding to the lines **70** of FIG. **8**. The four FPGAs **202**, **204**, **206** and **208** comprising the user array **42** are each coupled to the input data bus **210**, chain input bus **212** and output bus **214** as well as to each other by means of top bus **216**, right bus **218**, bottom bus **220**, left bus **222** and diagonal buses **224** and **226**.

User Array Interconnect

As previously described, the four user FPGAs (**202**, **204**, **206** and **208**) are interconnected through a series of horizontal, vertical, and diagonal buses which allow the easiest expansion of the existing symmetric internal chip routing for designs that exceed the capacity of a single FPGA for the user array **42**. In the exemplary illustration shown, bus sizes were chosen to utilize as many pins as possible while maintaining a bus width of at least 64 bits.

Address Structure

Because MAP may be located in the memory array of the system and decodes a portion of the address field, the address generated by the processor **12** must be correctly assembled. The following Table 1 shows the address bit allocation as seen by the processor **12** and the MAP element **112** board. The processor board bridge elements will reallocate the bit positions that are actually transmitted to the MAP element **112** based on system size.

Field Select Bits

The Field Select bits are the two most significant address bits leaving the bridge elements and are used to select which of the four possible mezzanine cards in the memory stack is being accessed. The Field Select bits for all mezzanine cards are determined by the state of P**6** bus bits A[**21**:**20**]. If bit A**21** is set, a MAP element **112** operation is underway and the Field Select bits are set to **11**. The MAP element **112** is always located just above the semaphore registers with the first MAP element **112** in segment **0** bank **0**, the second in segment **1** bank **0** and so on until one MAP element **112** is each segment's bank **0**. They are then placed in segment **0** bank **1** and the same pattern is followed until all are placed. This keeps them in a continuous address block.

Chip Select Bits

The next **3** most significant bits are Chip Select bits. These normally select which one of the eight rows of memory chips on a mezzanine board are activated. For MAP elements **112**, Chip Selects **0** and **1** are used. Chip Select **0** is used to write to the ESDRAM memory input buffer **40** and Chip Select **1** is used to access the control block **46** and user chips of the user array **42**.

Memory Address Bits

The next **19** most significant bits on the P**6** bus are Memory Address bits that normally select the actual location within the memory chip of the cache line in use. Five of these bits are decoded by the MAP element **112** into various commands that are discussed in greater detail hereinafter.

Bank Select Bits

The next 4 most significant bits are the Bank Select bits. These bits are used to select the specific bank within a segment in which the desired memory or MAP element **112** is located.

Trunk Select Bits

The next 4 most significant bits are the Trunk Select bits. The number of these bits range from 0 to 4 depending upon the number of segments in the system. These bits are used to select the segment that contains the desired memory or MAP. Unused bits are set to 0.

TABLE 1

P6 to Packet Bit Translation

Address	P6 Bus	Packet Bit	Bridge Output
0	0		
1	0		
2	0		
3	Cmd 0	13	Cmd 0
4	Cmd 1	14	Cmd 1
5	0	15	Map Sel 4
6	0	19	Map Sel 0
7	0	20	Map Sel 1
8	0	21	Map Sel 2
9	0	22	Map Sel 3
10	Cmd 2	23	Cmd 2
11	Cmd 3	24	Cmd 3
12	Sel 0	25	Sel 0
13	Sel 1	26	Sel 1
14	Sel 2	27	Sel 2
15	0	28	0
16	Map Sel 0	29	0
17	Map Sel 1	30	0
18	Map Sel 2	31	0
19	Map Sel 3	32	0
20	Map Sel 4	33	0
21	1	34	0
22	0	35	0
23	0	36	0
24	0	37	0
25	0	38	0
26	0	39	0
27	0	40	0
28	0	41	0
29	0	42	Chip Sel 0
30	0	43	Chip Sel 1
31	0	44	Chip Sel 2
32	0	45	1
33	0	46	1
34	0		
35	0		

Word Select Bits

The next 2 most significant bits are the Word Select bits. These bits determine the order in which each word of a 4-word cache line is being used. With CS[**1**:**0**] set to 01, these bits are part of the decoded command.

MAP Command Decode

CMD[**3**:**0**] are decoded into the following commands by the MAP control block **46** chip when CS[**1**:**0**] are 01 as shown in the following Table 2. This decode is also dependent upon the transaction being either a READ or WRITE. In addition, SEL[**2**:**0**] are used in conjunction with the RECON and LDROM commands described hereinafter to select which one of the eight ROM'S to be used.

TABLE 2

Address Bit Command Decode

CMD [3:0]						
3	2	1	0	Read/Write	Command	Basic Function
1	1	1	1	Write	Null	MAP operation continues as before this was received.
1	1	1	0	Write	RMB	Resets MAP Board user chips and reconfigures control chips.
1	1	0	1	Write	RUC	Resets User and control chip latches
1	1	0	0	Write	RECON	RECONfigures user circuits. Used with SEL[2:0].
1	0	1	1	Write	LASTOP	LAST OPerand is being written.
1	0	1	0	Write	WRTOP	WRiTe OPerand to user circuit.
1	0	0	1	Write	DONE	Processor is DONE with MAP clears busy flag.
1	0	0	0	Write	LDROM	Loads a new algorithm from input buffer into the ROM selected by SEL[2:0].
0	1	1	1	Write	START	Sends start address. stop address. auto/user. and stride to input control chip starting MAP operation.
0	1	1	0	Write	Future	Reserved.
0	1	0	1	Write	Future	Reserved.
0	1	0	0	Write	Future	Reserved.
0	0	1	1	Write	Future	Reserved.
0	0	1	0	Write	Future	Reserved.
0	0	0	1	Write	Future	Reserved.
0	0	0	0	Write	Future	Reserved.
1	1	1	1	Read	Null	MAP operation continues as before this was received.
1	1	1	0	Read	RDSTAT	Reads status word
1	1	0	1	Read	RDDAT	Reads 2 data words
1	1	0	0	Read	RDDAST	Reads status word and 1 data word
1	0	1	1	Read	Future	Reserved.
1	0	1	0	Read	Future	Reserved.
1	0	0	1	Read	Future	Reserved.
1	0	0	0	Read	Future	Reserved.
0	1	1	1	Read	Future	Reserved.
0	1	1	0	Read	Future	Reserved.
0	1	0	1	Read	Future	Reserved.
0	1	0	0	Read	Future	Reserved.
0	0	1	1	Read	Future	Reserved.
0	0	1	0	Read	Future	Reserved.
0	0	0	1	Read	Future	Reserved.
0	0	0	0	Read	Future	Reserved.

Null Command Description

When a MAP element **112** is not actively receiving a command, all inputs are set to 1 and all internal circuits are held static. Therefore, an incoming command of "1 1 1 1" cannot be decoded as anything and is not used.

RMB

This command, issued during a write transaction, causes the control block **46** chips to generate a global set reset ("GSR") to the user chips of the user array **42** and reprograms the control chips. All internal latches are reset but the configuration of the user chip is not changed. Any data that was waiting to be read will be lost.

RUC

This command, issued during a write transaction, causes the control chips to generate GSR signal to all four user FPGAs of the user array **42**. All internal latches are reset, but the configuration is not changed. Any operands will be lost, but data waiting to be read in the control block **46** chips will not.

RECON

This command, issued during a write transaction, causes the control chips to reconfigure the four user FPGAs of the user array **42** with the ROM selected by SEL[**2**:**0**]. Any operands still in process will be lost, but data waiting to be read in the control chip will not.

LASTOP

This command is issued during a write transaction to inform the MAP element **112** control block **46** chip that no more operands will be sent and the pipeline should be

flushed. The control chips start the pipeline counter and continue to provide read data until the pipeline depth is reached.

WRTOP

This command is issued during a write transaction to inform the MAP element **112** control block **46** chip that it is receiving a valid operand to be forwarded directly to the user circuits.

DONE

This command is issued during a write transaction to inform the MAP element **112** control block **46** chip that the processor **12** is done using the MAP element **112**. The control chips reset the busy bit in the status word and wait for a new user. The configuration currently loaded into the user circuits is not altered.

LDROM

This command is issued during a write transaction to inform the MAP element **112** control block **46** chip that the ROM specified by SEL[**2**:**0**] is to be reloaded with the contents of the input buffer **40** starting at address **0**. This will cause a nonvolatile change to be made to one of the eight on-board algorithms.

START

This command is issued during a write transaction and sends the start address, stop address, auto/user selection and stride to input controller. The input controller then takes control of input buffer **40** and starts transferring operands to the user chips of the user array **42** using these parameters until the stop address is hit. The data word **0** that accompanies this instruction contains the start address in bits **0** through **20**, the stop address in bits **23** through **43**, the stride in bits **46** through **51** and the user/auto bit in bit position **54**. In all cases the least significant bit ("LSB") of each bit group contains the LSB of the value.

RDSTAT

This command is issued during a read transaction to cause a status word to be returned to the processor **12**. This transaction will not increment the pipeline counter if it follows a LASTOP command. Details of the status word are shown in the following Table 4.

RDDAT

This command is issued during a read transaction to cause 2 data words to be returned to the processor **12**. This transaction will increment the pipeline counter if it follows a LASTOP command. Details of the status word are also shown in Table 4.

RDDAST

This command is issued during a read transaction to cause a status word and data word to be returned to the processor **12**.

SEL[**2**:**0**] Decode

The SEL[**2**:**0**] bits are used for two purposes. When used in conjunction with the RECON or LDROM commands, they determine which of the eight on-board ROM sets are to be used for that instruction. This is defined in the following Table 3.

TABLE 3

SEL[2:0] Decode			
2	1	0	ROM Select Function
0	0	0	ROM set 0
0	0	1	ROM set 1
0	1	0	ROM set 2
0	1	1	ROM set 3
1	0	0	ROM set 4
1	0	1	ROM set 5
1	1	0	ROM set 6
1	1	1	ROM set 7

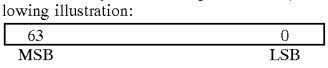
Status Word Structure

Whenever a read transaction occurs, a status word is returned to the processor **12** issuing the read. The structure of this 64-bit word is as follows:

TABLE 4

Status Word Structure	
Bits	Function
0–7	Contains the pipeline depth of the current user algorithm
8	A 1 indicates that the pipeline is empty following a LASTOP command.
9–31	These lines are tied low and are not used at this time.
32–35	Contains the current configuration selection loaded into the user FPGA's.
36–58	These lines are tied low and are not used at this time.
59	A 1 indicates that data was written and has overflowed the input buffers.
60	A 1 indicates that a reconfiguration of the user FPGA's is complete.
61	A 1 indicates that the data word is valid
62	A 1 indicates that at least 128 words are available
63	A 1 indicates that the MAP is busy and cannot be used by another processor.

Note:
Bit 63 is always the most significant bit ("MSB") as indicated in the following illustration:



Single MAP Element Operation

Normal operation of the MAP elements **112** are as follows. After power up, the MAP element **112** control block **46** chip automatically configures and resets itself. No configuration exists in the four user chips of the user array **42**. A processor **12** that wants to use a MAP element **112** first sends an RDSTAT command to the MAP element **112**.

If the MAP element **112** is not currently in use, the status word is returned with bit **63** "0" (not busy) and the busy bit is then set to 1 on the MAP element **112**. Any further RDSTAT or RDDAST commands show MAP element **112** to be busy.

After evaluating the busy bit and observing it to be "low", the processor **12** issues a RECON command along with the appropriate configuration ROM selection bits set. This causes the MAP element **112** to configure the user chips of the user array **42**. While this is happening, status bit **60** is "low". The processor **12** issues an RDSTAT and evaluates bit **60** until it returns "high". At this point, configuration is complete and the user chips of the user array **42** have reset themselves clearing all internal registers. The user then issues an RUC command to ensure that any previous data left in the user array **42** or control block **46** circuits has been cleared.

The user now has two methods available to present data to the MAP element **112**. It can either be directly written two quad words at a time into the user chips of the user array **42** or the input buffer **40** can be loaded.

Writing quad words is useful for providing a small number of reference values to the user array **42** but does

have lower bandwidth than using the input buffers **40** due to the 128-bit per transfer limit on un-cached writes. To use this mode, a WRTOP command is sent that delivers two 64-bit words to the user circuits. Based on previous knowledge of the algorithm, the program should know how many operands can be issued before an RDDAST could be performed. Evaluating status bits **0** through **7** after configuration also indicates the pipeline depth for this calculation.

If a large data set is to be operated on, or if a large quantity of the operands are to be reused, the input data buffer **40** should be used. In a particular embodiment of the present invention, this buffer may comprise 2M quad words of ESDRAM memory storage. This memory is located on the MAP element **112** and is accessed by performing cache line writes. This allows the loading of four 64-bit words per transaction. Once the data set is loaded, a START command is issued.

The control block **46** chip will assert the lockout bit signaling the memory controller not to access the input buffer **40**. It will also evaluate data word "0"of this transaction in accordance with the previously defined fields.

If the Auto/User bit is a "1", the addresses will automatically be generated by the control block **46** chip. The first address will be the start address that was transferred. The address is then incremented by the stride value until the stop address is hit. This address is the last address accessed.

At this point the lockout bit is released and the memory controller can access the input buffer **40**. It should be noted that the input control chip must interleave accesses to the input buffer **40** with refresh signals provided by the memory controller in order to maintain the ESDRAM memory while the lockout bit is set.

If the Auto/User bit was a "0", the operation is the same except the addresses are provided to the input control block **46** chip by the user algorithm.

Once the START command is issued, the processor **12** can start to read the output data. The user must first issue a RDDAST, which will return a status word and a data word. If bit **61** of the status word is a 1, the data word is valid. The user will continue this process until status word bit **62** is a 1. At this point the user knows that the output FIFO **74** on the MAP element **112** contains at least **128** valid data words and the RDDAT command can now be used for the next **64** reads. This command will return two valid data words without any status. After the 64 RDDAT commands the user must again issue a RDDAST command and check bits **61** and **62**. If neither is set, the FIFO **74** has no further data. If only **61** is set the program should continue to issue RDDAST commands to empty the FIFO **74**. If **61** and **62** are set, the program can resume with another set of 64 RDDAT commands and repeat the process until all results are received.

After all data is read and the user has completed his need for a MAP element **112**, a DONE command is issued. This will clear the busy flag and allow other processors **12** to use it. It should be noted that data in the input buffer **40** is not corrupted when used and can therefore be reused until a DONE is issued.

Chained MAP Operation MAP elements **112** have the ability to run in a vectored or VMAP™ mode (VMAP is a trademark of SRC Computers, Inc., assignee of the present invention). This mode allows the output data from one MAP element **112** to be sent directly to the user-chips in the user array **42** of the next MAP element **112** with no processor **12** intervention. In a representative embodiment, this link, or chain port **24**, operates at up to 800 MB/sec and connects all MAP elements **112** in a system in a chain. A chain must consist of a sequential group of at least two MAP elements **112** and up to as many as the system contains. Multiple non-overlapping chains may coexist.

To use this mode, the user simply designs the algorithm to accept input data from the chainin[**00**:**63**] pins. Output data paths are unchanged and always go to both the memory data bus and the chainout[**00**:**63**] pins.

VMAP mode operation is identical to single MAP element **112** operation except the data buffer **40** on the first MAP element **112** in the chain is loaded with data and all results are read from the last MAP element **112**. Chained MAP elements **112** simultaneously read from their input buffer **40** while accepting operands from the chainin port. This allows the buffers **40** used to supply reference during chained operation. To do this the input buffers **40** must first be loaded and then START commands must be sent to all MAP elements in the chain. The first MAP element **112** in the chain must be the last one to receive a START command. All MAP elements **112** other than the first in the chain must receive a START command with the user address mode selected.

LDROM Operation

MAP elements **112** have the capability to allow the contents of an on-board ROM to be externally reloaded while the system is operating, thus changing the algorithm. It should be noted that-the same ROM for all four user chips in the user array **42** will simultaneously be updated.

To accomplish this, the configuration files of the four ROMs of a given set are converted from a serial stream to 16-bit words. The first words of each ROM file are then combined to form a 64-bit word. User chip **0** of the user array **42** files fill bits **0** through **15**, chip **1** is **16** through **31**, chip **2** is **31** through **47**, and chip **3** is **48** through **64**. This process is repeated until all four of the individual files are consumed. This results in a file that is 64-bits wide and 51,935 entries deep.

If the contents of a particular ROM in the set are to be unaltered, its entries must be all **0**. At the top of this file, a header word is added that contains all **1**'s in all bit positions for all ROMs in the set that are to be updated. ROMs that are to be unaltered will contain zeros in this word. This file is then loaded into the MAP element **112** input buffer **40** with the header loaded into address **0**.

Upon receiving an LDROM command, the input controller will load the user chips of the user array **42** with a special algorithm that turns them into ROM programmers. These chips will then start accessing the data in the input buffer **40** and will evaluate word 0.

If this is a 0, no further action will be taken by that chip. If it is a 1, the chip will continue to extract data, serialize it, and load it into the ROM that was selected by the state of the SEL lines during the LDROM command. While this is happening, bit **60** of the status word is 0. When complete, bit **60** will return to a 1.

The user must always issue a RECON command following an LDROM command in order to load a valid user algorithm back into the user array **42** and overwrite the ROM programmer algorithm.

With reference additionally now to FIG. **10**, a functional block diagram of another alternative embodiment **230** of the present invention is shown wherein individual MAP elements **112** are closely associated with individual memory arrays and each of the MAP elements **112** comprises independent chain ports **24** for coupling the MAP elements **112** directly to each other. The system illustrated comprises a processor assembly comprising one or more processors **12** bi-directionally coupled through a processor switch (which may comprise an FPGA) to a write trunks **26** and read trunks **28**.

In the example illustrated, a number of MAP elements **112** are associated with a particular memory array **246** under control of a memory controller **238** (which may also comprise an FPGA). As illustrated, each of the memory controllers $\mathbf{238}_A$ and $\mathbf{238}_B$ are coupled to the processor assembly **232** through the processor switch **234** by means of the write and read trunks **26**, **28**. Each of the memory controllers may be coupled to a plurality of MAP elements **112** and associated memory array **246** and to additional MAP elements **112** by means of a chain port **24** as previously described. In the embodiment illustrated, memory controller $\mathbf{238}_A$ is in operative association with a pair of MAP elements, the first comprising buffer $\mathbf{240}_{A1}$, user array $\mathbf{242}_{A1}$ and FIFO $\mathbf{244}_{A1}$ associated with memory array $\mathbf{246}_{A1}$ and the second comprising buffer $\mathbf{240}_{A2}$, user array $\mathbf{242}_{A2}$ and FIFO $\mathbf{244}_{A2}$ associated with memory array $\mathbf{246}_{A2}$. In like manner, memory controller $\mathbf{238}_B$ is in operative association with a pair of MAP elements, the first comprising buffer $\mathbf{240}_{B1}$, user array $\mathbf{242}_{B1}$ and FIFO $\mathbf{244}_{B1}$ associated with memory array $\mathbf{246}_{B1}$ and the second comprising buffer $\mathbf{240}_{B2}$, user array $\mathbf{242}_{B2}$ and FIFO $\mathbf{244}_{B2}$ associated with memory array $\mathbf{246}_{B2}$.

With reference additionally now to FIG. **11**A and **11**B separate timing diagrams are illustrated respectively depicting input and output timing in relationship to the system clock ("Sysclk") signal.

Interface Timing

The MAP element **112** user array **42** can accept data from the input memory bus, input buffer **40** or the chain port **24**. In the embodiment of the present invention previously described and illustrated, all sixty four bits from any of these sources are sent to all four of the user chips (**202**, **204**, **206** and **208**; FIG. **9**) along with a VALID IN signal on lines **68** (FIG. **8**) sent from the control block **46** that enables the input clock in the user chips of the user array **42**.

This signal stays high for ten, twenty or forty nanoseconds depending on whether one, two or four words are being transferred. This VALID IN signal on lines **68** connects to the clock enable pins of input latches in the user chips of the user array **42**. These latches then feed the user circuit in the MAP element **112**. The timing for the various write operations is shown in with particularity in FIG. **11**A.

Input Timing

After the algorithm operation has completed, output data is formed into 64-bit words-in the user chips of the user array **42** on pins connected to the DOUT[**00**:**63**] nets. These nets, in turn, connect to the output FIFO **74** (FIG. **8**) that ultimately provides the read data to the memory controller or the next MAP element **112** in the chain. After forming the 64-bit result, the user circuitry must ensure that a "FULL" signal is "low". When the signal is "low", the transfer is started by providing a "low" from the user array **42** to the control block **46** and the FIFO#WE input on the FIFO **74**.

At the same time, valid data must appear on the data out ("DOUT") nets. This data must remain valid for 10 nanoseconds and FIFO#WE must remain "low" until the end of this 10-nanosecond period. If multiple words are to be transferred simultaneously, the FIFO#WE input must remain "low" until the end of this 10-nanosecond period as shown with particularity in FIG. **11**B.

Output Timing

Three result words can be transferred out of the user array **42** before a "read" should occur to maximize the "read" bandwidth. The output FIFO **74** (FIG. **8**) is capable of holding 512 k words in the embodiment illustrated. When three words are held in the control block **46**, the word counter in the status word will indicate binary "11".

Pipeline Depth

To aid in system level operation, the user array **42** must also provide the pipeline depth of the algorithm to the control block **46**. In a particular embodiment of the present invention, this will be equal to the number of 100-MHz clock cycles required to accept a data input word, process that data, and start the transfer of the results to the FIFO **74**.

If an algorithm is such that initialization parameters or reference numbers are sent prior to actual operands, the pipeline depth is equal only to the number of clock cycles required to process the operands. This depth is provided as a static 8-bit number on nets DOUT[**64**:**71**] from FPGAs **202** and/or **204** (FIG. **9**). Each of the eight bits are generally output from only of the FPGAs of the user array **42** but the eight bits may be spread across both chips.

In a particular embodiment of the present invention, the ROMs that are used on the MAP elements **112** may be conveniently provided as ATMEL™ AT**17**LVO**1**O in a 20-pin PLCC package. Each ROM contains the configuration information for one of the four user FPGAs of the user array **42**. There may be eight or more ROM sockets allocated to each of the user chips of the user array **42** to allow selection of up to eight or more unique algorithms. In an embodiment utilizing eight ROMs, the first ROM listed for each of the four user chips may be selected by choosing configuration Oh and the last ROM selected by choosing configuration **8**h.

If all four user chips of the user array **42** are not needed for an algorithm, the unused chips do not require that their ROM sockets be populated. However, at least one of the user chips must always contain a correctly programmed ROM even if it is not used in the algorithm because signals related to the configuration timing cycle are monitored by the control block. The user FPGA that directly connects to both the DIN and DOUT signals, should always be used first when locating the algorithm circuit.

With reference additionally now to FIG. **12**, a simplified illustration of a representative operating environment **300** for the system and method of the present invention is shown including a typical web site server **306** as would be replaced by, for example, an SRC-6 reconfigurable server **308** (comprising, for example, the multiprocessor computer **10** or computer system **20** of the preceding figures) or other computer system incorporating one or more industry standard processors together with one or more reconfigurable processors having all of the processor controlled by a single system image of the operating system. In this simplified illustration, a number of personal computers **302** or other computing devices are coupled to either the typical web site server **306** (in a prior art implementation) or the reconfigurable sever **308** (in accordance with the system and method of the present invention) through the Internet **304**.

With reference additionally now to FIG. **13**, a flowchart is shown illustrating a conventional data processing sequence **310** in a conventional application of a typical web site server **306** as depicted in the preceding figure. The sequence **310** begins with the input of a number "N" of demographic data elements for processing by the typical web site server **306**. These N data elements are then serially processed at step **314** until the last of the data elements is determined and processed at decision step **316**. Therefore, N iterations by the microprocessor of the typical web site server **306** are required to complete processing of the input data elements.

Following this protracted data processing period, the typical web site server **306** then can undertake to select the new web page content specifically adapted to the particular web site visitor at step **318**, which updated site content is displayed at step **320**.

With reference additionally now to FIG. **14**, a corresponding flowchart is shown illustrating the processing of demographic or other data utilizing the reconfigurable server **308** of FIG. **12** in a significantly faster data processing sequence **330**. The processing sequence **330** again begins with the input of N demographic data elements or other secure socket, database or other data for processing by the site server at input step **332**. Importantly, the reconfigurable server **308** is now able to process the individual data elements in parallel through the use of a single reconfigurable processor, (such as a MAP element), due to its ability to instantiate more than one processing unit that is tailored to the job as opposed to reusing one or two processing units located within a microprocessor. In the exemplary embodiment shown, all of reconfigurable processors may share all of the system's resources and be controlled by a single system image of the operating system although, in alternative embodiments, cluster management software may be utilized to effectively make a cluster of microprocessors appear to a user to be but a single copy of the operating system. In any event, the completion of steps $\mathbf{334}_1$ through $\mathbf{334}_N$ requires only 1 iteration to prepare the site to select the new content at step **336** and then display it at step **338**.

While there have been described above the principles of the present invention in conjunction with one or more specific embodiments of the present invention and MAP elements, it is to be clearly understood that the foregoing description is made only by way of example and not as a limitation to the scope of the invention. Particularly, it is recognized that the teachings of the foregoing disclosure will suggest other modifications to those persons skilled in the relevant art for use in processing differing types of data at a web site. Such modifications may involve other features which are already known per se and which may be used instead of or in addition to features already described herein. Although claims have been formulated in this application to particular combinations of features, it should be understood that the scope of the disclosure herein also includes any novel feature or any novel combination of features disclosed either explicitly or implicitly or any generalization or modification thereof which would be apparent to persons skilled in the relevant art, whether or not such relates to the same invention as presently claimed in any claim and whether or not it mitigates any or all of the same technical problems as confronted by the present invention. The applicants hereby reserve the right to formulate new claims to such features and/or combinations of such features during the prosecution of the present application or of any further application derived therefrom.

What is claimed is:

1. A method for processing data at an internet site comprising:

providing a reconfigurable server at said site incorporating at least one microprocessor and at least one reconfigurable processing element;

receiving N data elements at said site relative to a remote computer coupled to said site;

instantiating N of said reconfigurable processing elements at said reconfigurable server; and

processing said N data elements with corresponding ones of said N reconfigurable processing elements.

2. The method of claim **1** further comprising:

selecting a content of said site in response to said processed N data elements.

3. The method of claim **2** further comprising:

transmitting said content to said remote computer.

4. The method of claim **3** further comprising:

displaying said content at said remote computer.

5. The method of claim **1** wherein said N data elements comprise demographic data pertaining to said remote computer.

6. The method of claim **1** wherein said N data elements comprise N encrypted data elements transmitted by said remote computer to said site.

7. The method of claim **6** wherein said step of processing said N data elements comprises:

decrypting said N encrypted data elements.

8. The method of claim **1** wherein said N data elements comprise N database query elements.

9. The method of claim **8** wherein said step of processing said N data elements comprises:

processing said N database query elements.

10. The method of claim **1** further comprising:

controlling said at least one microprocessor and at least a portion of said at least one reconfigurable processing element by a single system image of an operating system.

11. An internet processing acceleration service comprising:

a reconfigurable server coupled to said internet, said server comprising at least one microprocessor and at least one reconfigurable processor; and

a single system image of an operating system controlling said at least one microprocessor and at least a portion of said at least one reconfigurable processors;

said service instantiating N of said at least a portion of said at least one reconfigurable processors for substantially parallel processing of N data elements received by said server.

12. The service of claim **11** wherein said N data elements comprise demographic data pertaining to a computer coupled to said server by means of said internet.

13. The service of claim **12** wherein said server selects a content for transmission to said computer by means of said internet based upon said demographic data.

14. The service of claim **11** wherein said N data elements comprise encrypted data elements transmitted to said server by a computer coupled to said internet.

15. The service of claim **14** wherein said server is operative to decrypt said encrypted data elements.

16. The service of claim **11** wherein said N data elements comprise database query elements.

17. The service of claim **16** wherein said server is operative to process said data base query.

18. A process of accelerating access time of a remote computer to an internet site comprising:

providing a reconfigurable server at said site incorporating at least one microprocessor and at least one reconfigurable processor;

transmitting N data elements from said remote computer to said server;

substantially concurrently processing said N data elements with N of said at least one reconfigurable processors;

selecting a content of said internet site in response to said N data elements; and

transmitting said content to said remote computer.

19. The process of claim **18** wherein said N data elements comprise demographic data pertaining to said remote computer.

20. The process of claim **18** wherein said N data elements comprise N encrypted data elements.

21. The process of claim **20** wherein said step of substantially concurrently processing said N data elements comprises:

decrypting said N encrypted data elements.

22. The process of claim **18** wherein said N data elements comprise N database query elements.

23. The process of claim **22** wherein said step of substantially concurrently processing said N data elements comprises:

processing said N database query elements.

24. The process of claim **18** further comprising:

controlling said at least one microprocessor and at least a portion of said at least one reconfigurable processors by a single system image of an operating system.

25. The process of claim **18** further comprising:

displaying said content at said remote computer.

* * * * *

EXHIBIT G

US007620800B2

(12) **United States Patent**
Huppenthal et al.

(10) **Patent No.: US 7,620,800 B2**
(45) **Date of Patent: *Nov. 17, 2009**

(54) **MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS**

(75) Inventors: **Jon M. Huppenthal**, Colorado Springs, CO (US); **David E. Caliga**, Colorado Springs, CO (US)

(73) Assignee: **SRC Computers, Inc.**, Colorado Springs, CO (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 95 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **11/733,064**

(22) Filed: **Apr. 9, 2007**

(65) **Prior Publication Data**

US 2007/0204131 A1 Aug. 30, 2007

Related U.S. Application Data

(63) Continuation of application No. 10/285,318, filed on Oct. 31, 2002, now Pat. No. 7,225,324.

(51) **Int. Cl.**
G06F 15/82 (2006.01)

(52) **U.S. Cl.** **712/226**

(58) **Field of Classification Search** 712/226, 712/15, 19, 215

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,727,503	A	2/1988	McWhirter
4,763,294	A	8/1988	Fong
4,872,133	A	10/1989	Leeland
4,962,381	A	10/1990	Helbig, Sr.
5,020,059	A	5/1991	Gorin et al.
5,072,371	A	12/1991	Benner et al.
5,230,057	A	7/1993	Shido et al.
5,274,832	A	12/1993	Khan
5,471,627	A	11/1995	Means et al.
5,477,221	A	12/1995	Chang et al.

(Continued)

FOREIGN PATENT DOCUMENTS

JP	59-206972	11/1984

(Continued)

OTHER PUBLICATIONS

Gaudiot, Jean-Luc, Data-Driven Multicomputers in Digital Signal Processing,1987, IEEE, Proceedings of the IEEE, vol. 75,No. 9, pp. 1220-1234.*

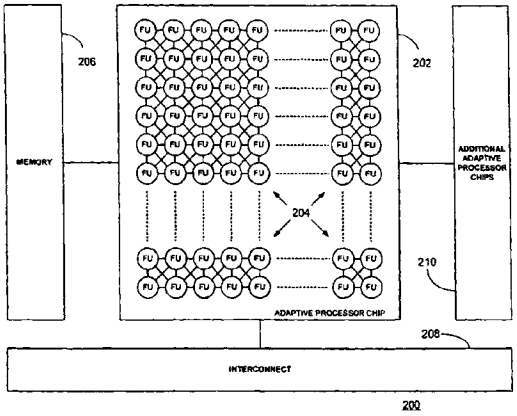
(Continued)

Primary Examiner—Eric Coleman
(74) *Attorney, Agent, or Firm*—Michael C. Martensen; William J. Kubida; Hogan & Hartson LLP

(57) **ABSTRACT**

Multi-adaptive processing systems and techniques for enhancing parallelism and performance of computational functions are disclosed which can be employed in a myriad of applications including multi-dimensional pipeline computations for seismic applications, search algorithms, information security, chemical and biological applications, filtering and the like as well as for systolic wavefront computations for fluid flow and structures analysis, bioinformatics etc. Some applications may also employ both the multi-dimensional pipeline and systolic wavefront methodologies disclosed.

52 Claims, 20 Drawing Sheets



U.S. PATENT DOCUMENTS

5,509,134	A	4/1996	Fandrich et al.
5,570,040	A	10/1996	Lytle et al.
5,640,586	A	6/1997	Pechanek et al.
5,715,453	A	2/1998	Stewart
5,737,766	A	4/1998	Tan
5,784,108	A	7/1998	Skaletzky et al.
5,802,290	A	9/1998	Casselman
5,892,962	A	4/1999	Cloutier
5,903,771	A	5/1999	Sgro et al.
5,915,123	A	6/1999	Mirsky et al.
5,953,502	A	9/1999	Helbig, Sr.
5,956,518	A	9/1999	DeHon et al.
5,966,534	A	10/1999	Cooke et al.
6,023,755	A	2/2000	Casselman
6,052,773	A	4/2000	DeHon et al.
6,061,706	A	5/2000	Gai et al.
6,076,152	A	6/2000	Huppenthal et al.
6,128,663	A	10/2000	Thomas
6,192,439	B1	2/2001	Grunewald et al.
6,215,898	B1	4/2001	Woodfill et al.
6,226,776	B1	5/2001	Panchul et al.
6,289,440	B1	9/2001	Casselman
6,385,757	B1	5/2002	Gupta et al.
6,704,816	B1	3/2004	Burke
6,721,884	B1	4/2004	De Oliveira Kastrup Pereira et al.

FOREIGN PATENT DOCUMENTS

JP	63-086079	4/1988

OTHER PUBLICATIONS

Dennis,J. B., Data Flow Supercomputers, Nov. 1980, IEEE, Computer, pp. 48-56.*

Qunnn M.J., et al., Data-Parallel Programming on Multicomputers, Sep. 1990, IEEE, pp. 69-76.*

Trevleaven, P.C., et al., Data-Driven and Demand-Driven Computer Architecture, 1982, ACM, Computiing Surveys vol. 14, No. 1, pp. 93-143.*

Webster. M., Webster's Ninth New Collegiate Dictionary, 1985, Merriam-Webster pub., p. 627.*

Miyamori, Takashi, "REMARC: Reconfigurable Multimedia Array Coprocessor", IEICE Transactions on Information and Systems, Information & Systems Society, Tokyo, JP, vol. E82-D, No. 2, Feb. 1999, pp. 389-397, XP000821922.

Gross Thomas, et al., "Compilation for a High-performance Systolic Array", Sigplan Notices USA, vol. 21, No. 7, Jul. 1986, pp. 27-38, XP002418625.

Rauchwerger, Lawrence, et al., "The LRPD Test: Speculative Run-Time Parallelization of Loops with Privatization and Reduction Parallelization", IEEE Transactions on Parallel and Distributed Systems, IEEE Service Center, Los Alamitos, CA, vol. 10, No. 2, Feb. 1999, pp. 160-180, XP000908318.

Arnold Jeffrey M. et al., "The Splash 2 Processor and Applications", Computer Design: VLSI in Computers and Processors, 1993, ICCD '93 Proceedings, 1993 IEEE International Conference on Cambridge, MA, Oct. 3-6, 1993, Los Alamitos, CA, IEEE Comput. Soc., Oct. 3, 1993, pp. 482-485, XP010134571.

Hwang, Kai, "Computer Architecture and Parallel Processing", Data Flow Computers and VLSI Computations, 1985, McGraw Hill, Chapter 10, pp. 732-807, XP-002418655.

Hartenstein, Reiner W., et al. "A Synthesis System for Bus-based Wavefront Array Architectures", Proceedings, International Conference on Application-Specific Systems, Architectures and Processors, 1996, pp. 274-283, XP002132819.

Alexander, Thomas, et al. "A Reconfigurable Approach To A Systolic Sorting Architecture", ISCAS 89, May 8, 1989, pp. 1178-1182, XP010084477.

Wu, Youfeng, et al. "Better Exploration of Region-Level Value Locality with Integrated Computation Reuse and Value Prediction", Proceedings of the 28th International Symposium on Computer Architecture, ISCA 2001, Goteborg, Sweden, Jun. 30-Jul. 4, 2001, International Symposium on Computer Architecture, (ISCA), Los Alamitos, CA, IEEE Comp. Soc, US, Jun. 30, 2001, pp. 93-103, XP010552866.

Babb, Jonathan, et al., "Parallelizing applications into silicon", © 1999 IEEE.

Deshpande, Deepali, et al., "Hybrid data/configuration caching for striped FPGAs" © 1999 IEEE.

Purna, Karthikeya, et al., "Temporal partitioning and sched uling data flow graphs for reconfigurable computers", © 1999 IEEE, Publ. No, 0018-9340/99, pp. 579-590.

Gibbs, W. Wayt, "Blitzing bits", © 1999 Scientific American Presents, pp. 57-61.

Gonzalez, Ricardo, "Configurable and extensible processors change system design", Aug. 15-17, 1999, Hot Chips 11 Tutorials, pp. 135-146.

Graham, Paul, et al., "FPGA-based sonar processing", © 1998 ACM 0-89791-978-5/98, pp. 201-208.

Hasebe, A., et al., "Architecture of SIPS, a real time image processing system," © 1988 IEEE, Publ. No. CH2603-9/88/0000/0621, pp. 621-630.

Hammond, Lance, et al., "The Stanford Hydra CMP", Aug. 15-17, 1999 H ot Chips 11 Tutorials, pp. 23-31.

Jean, Jack, et al., "Dynamic reconfiguration to support concurrent applications", © 1999 IEEE, Publ. No. 0018-9340/99, pp. 591-602.

Kastrup, Bernardo, et al., "Concise: a compiler-driven CPLD-based instruction set accelerator", © 1999 IEEE.

Motomura, Masato, et al., "An embedded DRAM-FPGA chip with instantaneous logic reconfiguration", © 1998 IEEE, Publ. No. 0-8186-8900-5/98, pp. 264-266.

McConnell, Ray, "Massively parallel computing on the Fuzion chip", Aug. 15-17, 1999, Hot Chips 11 Tutorials, pp. 83-94.

McShane, Erik, et al., "Functionally integrated systems on a chip: technologies, architectures, CAD tools, and applications", © 1998 IEEE, Publ. No. 8-8186-8424-0/98, pp. 67-75.

Rupp, Charley, et al., "The NAPA adaptive processing architecture", © 1998 the Authors, pp. 1-10.

Saito, Osamu, et al., "A 1M synapse self learnin g digital neural network chip", ©0 1998 IEEE, Publ. No. 0-7803-4344/1/98, pp. 94-95.

Schott, Brian, et al., "Architectures for system-level applications of adaptive computing", © 1999 IEEE.

Mencer, Oskar, et al., "PAM-Blox: High Performance FPGA Design for Adaptive Computing", © 1998 IEEE, Conference Paper, INSPEC Abstract Nos. B9611-1265B-044, C9811-5210-009.

Miyamori, Takashi, et al., "A quantitative analysis of reconfigurable coprocessors for multimedia applications", © 1998 IEEE, Conference Paper, INSPEC Abstract Nos. B9811-1265F-011, C 9811-5310-010.

Agarwal, A., et al., "The Raw Compiler Project", pp. 1-12, http://cag-www.lcs.mit.edu/raw, Proceedings of the Second SUIF Compiler Workshop, Aug. 21-23, 1997.

Albaharna, Osama, et al., "On the viability of FPGA-based integrated coprocessors", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 206-215.

Amerson, Rick, et al., "Teramac—Configurable Custom Computing", © 1995 IEEE, Publ. No. 0-8186-7086-X/95, pp. 32-38.

Barthel, Dominique Aug. 25-26, 1997, "PVP a Parallel Video coProcessor", Hot Chips IX, pp. 203-210.

Bertin, Patrice, et al., "Programmable active memories: a performance assessment", © 1993 Massachusetts Institute of Technology, pp. 88-102.

Bittner, Ray, et al., "Computing kernels implemented with a wormhole RTR CCM", © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 98-105.

Buell, D., et al. "Splash 2: FPGAs in a Custom Computing Machine—Chapter 1—Custom Computing Machines: An Introduction", pp. 1-11, http://www.computer.org/espress/catalog/bp07413/spls-ch1.html (originally believed published in J. of Supercomputing, vol. IX, 1995, pp. 219-230.

Casselman, Steven, "Virtual Computing and The Virtual Computer",© 1993 IEEE, Publ. No. 0-8186-3890-7193, pp. 43-48.

Chan, Pak, et al., "Architectural tradeoffs in field-programmable-device-based computing systems", © 1993 IEEE, Publ, No. 0-8186-3890-7/93, pp. 152-161.
Clark, David, et al., "Supporting FPGA microprocessors through retargetable software tools", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 195-103.
Cuccaro, Steven, et al., "The CM-2X: a hybrid CM-2/Xilink prototype", © 1993 IEEE, Publ. No. 0-8186-3890-7/93, pp. 121-130.
Culbertson, W. Bruce, et al., "Exploring architectures for volume visualization on the Teramac custom computer", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 80-88.
Culbertson, W. Bruce, et al., "Defect tolerance on the Teramac custom computer", © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 116-123.
Dehon, Andre, DPGA-Coupled microprocessors: commodity IC for the early 21st century',© 1994 IEEE, Publ. No. 0-8186-5490-2/94, pp. 31-39.
Dehon, A., et al., "Matrix a Reconfigurable Computing Device with Configurable Instruction Distribution", Hot Chips IX, Aug. 25-26, 1997, Stanford, California, MIT Artificial Intelligence Laboratory.
Dhaussy, Philippe, et al., "Global control synthesis for an MIMD/FPGA machine", © 1994 IEEE, Publ. No. 0-8186-5490-2/94, pp. 72-81.
Elliott, Duncan, et al., "Computational Ram: a memory-SIMD hybrid and its application to DSP",© 1992 IEEE, Publ. No. 0-7803-0246-X/92, pp. 30.6.1-30.6.4.
Fortes, Jose, et al., "Systolic arrays, a survey of seven projects", © 1987 IEEE, Publ. No. 0018-9162/87/0700-0091, pp. 91-103.
Gokhale, M., et al., "Processing in Memory: The Terasys Massively Parallel PIM Array" © Apr. 1995, IEEE, pp. 23-31.
Gunther, Bernard, et al., "Assessing Document Relevance with Run-Time Reconfigurable Machines", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 10-17.
Hagiwara, Hiroshi, et al., "A dynamically microprogrammable computer with low-level parallelism", © 1980 IEEE, Publ. No. 0018-9340/80/07000-0577, pp. 577-594.
Hartenstein, R. W., et al. "A General Approach in System Design Integrating Reconfigurable Accelerators," http://xputers.informatik.uni-kl.de/papers/paper026-1.html, IEEE 1996 Conference, Austin, TX, Oct. 9-11, 1996.
Hartenstein, Reiner, et al., "A reconfigurable data-driven ALU for Xputers", © 1994 IEEE, Publ. No. 0-8186-5490-2/94, pp. 139-146.
Hauser, John, et al.: "GARP: a MIPS processor with a reconfigurable co-processor", © 1997 IEEE, Publ. No. 0-08186-8159-4/97, pp. 12-21.
Hayes, John, et al., "A microprocessor-based hypercube, supercomputer", © 1986 IEEE, Publ. No. 0272-1732/86/1000-0006, pp. 6-17.
Herpel, H. -J., et al., "A Reconfigurable Computer for Embedded Control Applications", © 1993 IEEE, Publ. No. 0-8186-3890-7/93, pp. 111-120.
Hogl, H., et al., "Enable++: A second generation FPGA processor", © 1995 IEEE, Publ. No. 0-8186-7086-X/95, pp. 45-53.
King, William, et al., "Using MORRPH in an industrial machine vision system". © 1996 IEEE, Publ. No. 08186-7548-9/96, pp. 18-26.
Manohar, Swaminathan, et al., "A pragmatic approach to systolic design", © 1988 IEEE, Publ. No. CH2603-9/88/0000/0463, pp. 463-472.
Mauduit, Nicolas, et al., "Lneuro 1.0: a piece of hardware LEGO for building neural network systems," © 1992 IEEE, Publ. No. 1045-9227/92, pp. 414-422.
Mirsky, Ethan A., "Coarse-Grain Reconfigurable Computing", Massachusetts Institute of Technology, Jun. 1996.
Mirsky, Ethan, et al., "Matrix: A Reconfigurable Computing Architecture with Configurable Instruction Distribution and Deployable Resources", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 157-166.
Morley, Robert E., Jr., et al., "A Massively Parallel Systolic Array Processor System",© 1988 IEEE, Publ. No. CH2603-9/88/0000/0217, pp. 217-225.
Patterson, David, et al., "A case for Intelligent DRAM: IRAM", Hot Chips VIII, Aug. 19-20, 1996, pp. 75-94.
Peterson, Janes, et al., "Scheduling and partitloning ANSI-C programs onto multi-FPGA CCM architectures", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 178-187.
Schmit, Herman, "Incremental reconfiguration for pipelined applications," © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 47-55.
Sitkoff, Nathan, et al., "Implementing a Genetic Algorithm on a Parallel Custom Computing Machine", Publ. No. 0-8186-7086-X/95, pp. 180-187.
Stone, Harold, "A logic-in-memory computer", © 1970 IEEE, IEEE Transactions on Computers, pp. 73-78, Jan. 1990.
Tangen, Uwe, et al., "A parallel hardware evolvable computer POLYP extended abstract", © 1997 IEEE, Publ. No. 0-8186-8159/4/97, pp. 238-239.
Thornburg, Mike, et al., "Transformable Computers", © 1994 IEEE, Publ. No. 0-8186-5602-6/94, pp. 674-679.
Tomita, Shinji, et al., "A computer low-level parallelism QA-2", © 1986 IEEE, Publ. No. 0-0384-7495/86/0000/0280, pp. 280-289.
Trimberger, Steve, et al., "A time-multiplexed FPGA", © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 22-28.
Ueda, Hirotada, et al., "A multiprocessor system utilizing enhanced DSP's for Image processing", © 1988 IEEE, Publ. No. CH2603-9/88/0000/0611, pp. 611-620.
Villasenor, John, et al., "Configurable computing", © 1997 Scientific American, Jun. 1997.
Wang, Quiang, et al., "Automated field-programmable compute accelerator design using partial evaluation", © 1997 IEEE, Publ. No. 0-8186-8159-4/97, pp. 145-154.
W.H. Manglone-Smith and B.L. Hutchings. Configurable computing: The Road Ahead. In Proceedings of the Reconfigurable Architectures Workshop (RAW'97), pp. 81-96, 1997.
Wirthlin, Michael, et al., "The Nano processor: a low resource reconfigurable processor", © 1994 IEEE, Publ. No. 0-8186-5490-2/94, pp. 23-30.
Wirthlin, Michael, et al., "A dynamic instruction set computer", © 1995 IEEE, Publ. No. 0-8186-7086-X/95, pp. 99-107.
Wittig, Ralph, et al., "One Chip: An FPGA processor with reconfigurable logic", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 126-135.
Yamauchi, Tsukasa, et al., "SOP: A reconfigurable massively parallel system and its control-data flow based compiling method", © 1996 IEEE, Publ. No. 0-8186-7548-9/96, pp. 148-156.
"Information Brief", PCI Bus Technology, © IBM Personal Computer Company, 1997, pp. 1-3.
Yun, Hyun-Kyu and Silverman, H. F.; "A distributed memory MIMD multi-computer with reconfigurable custom computing capabilities", Brown University, Dec. 10-13, 1997, pp. 7-13.
Hoover, Chris and Hart, David; "San Diego Supercomputer Center, Timelogic and Sun Validate Ultra-Fast Hidden Markov Model Analysis-One DeCypher-accelerated Sun Fire 6800 beats 2,600 CPUs running Linux-", San Diego Supercomputer Center, http://www.sdsc.edu/Press/02/050802_markovmodel.html, May 8, 2002, pp. 1-3.
Caliga, David and Barker, David Peter, "Delivering Acceleration: The Potential for Increased HPC Application Performance Using Reconfigurable Logic", SRC Computers, Inc., Nov. 2001, pp. 20.
Hammes, J.P., Rinker, R. E., McClure, D. M., Böhm, A. P. W., Najjar, W. A, "The SA-C Compiler Dataflow Description", Colorado State University, Jun. 21, 2001, pp. 1-25.
Callahan, Timothy J. and Wawrzynek, John, "Adapting Software Pipelining for Reconfigurable Computing", University of California at Berkeley, Nov. 17-19, 2000, pp. 8.
Ratha, Nalini K., Jain, Anil K. and Rover, Diane T., "An FPGA-based Point Pattern Matching Processor with Application to Fingerprint Matching", Michigan State University, Department of Computer Science, pp. 8.
Dehon, André, "Comparing Computing Machines", University of California at Berkeley, Proceedings of SPIE vol. 3526, Nov. 2-3, 1998, pp. 11.
Vemuri, Ranga R. and Harr, Randolph E., "Configurable Computing: Technology and Applications", University of Cincinnati and Synopsys Inc., IEEE, Apr. 2000, pp. 39-40.
Demon, André, "The Density Advantage of Configurable Computing", California Institute of Technology, IEEE, Apr. 2000. pp. 41-49.

Haynes, Simon D., Stone, John, Cheung, Peter Y.K. and Luk, Wayne, "Video Image Processing with the Sonic Architecture", Sony Broadcast & Professional Europe, Imperial College, University of London, IEEE, Apr. 2000, pp. 50-57.
Platzner , Marco, "Reconfigurable Accelerators for Combinatorial Problems", Swiss Federal Institute of Technology (ETH) Zurich, IEEE, Apr. 2000, pp. 58-60.
Callahan, Timothy J., Hauser, John R. and Wawrzynek, John, "The Garp Architecture and C Compiler", University of California, Berkeley, IEEE, Apr. 2000. pp. 62-69.
Goldstein, Seth Copen, Schmit, Herman, Budiu , Mihai, Cadambi, Srihari, Moe, Matt and Taylor, R. Reed, "PipeRench: A Reconfigurable Architecture and Compiler", Carnegie Mellon University, IEEE, Apr. 2000, pp. 70-76.
Muchnick, Steven S., "Advanced Compiler Design and Implementation", Morgan Kaufmann Publishers, pp. 217.
Hammes, Jeffrey P., Dissertation "Compiling SA-C to Reconfigurable Computing Systems", Colorado State University, Department of Computer Science, Summer 2000, pp. 179.
Automatic Target Recognition, Colorado State University & USAF, http://www.cs.colostate.edu/cameron/applications.html, pp. 1-3.
Chodowiec, Pawel, Khuon, Po, Gaj, Kris, Fast Implementations of Secret-Key Block Ciphers Using Mixed Inner- and Outer-Round Pipelining, George Mason University, Feb. 11-13, 2001, pp. 9.
Hastie, Neil, et al., "The Implementation of Hardware Subroutines on Field Programmable Gate Arrays", XP010005485, Plessey Semiconductors, Tamerton Rd., Plymouth, Devon, England, IEEE, May 13, 1990, Custom Integrated Circuits Conference, pp. 314. 1-4. *the whole document*.
Harbaum, Till, et al., "Design of a Flexible Coprocessor Unit", Institute of Operating Systems and Computer Networks, XP000879556TU Braunschweig, Germany, Proceedings of the Euromicro Conference, Sep. 1999, pp. 335-342. *whole document*.
Mathias P C; Patnaik L M: "Systolic Evaluation of Polynomial Expressions," IEEE Transactions on Computers, vol. 39, No. 5, May 1, 1990, pp. 653-665, XP000116659.

* cited by examiner

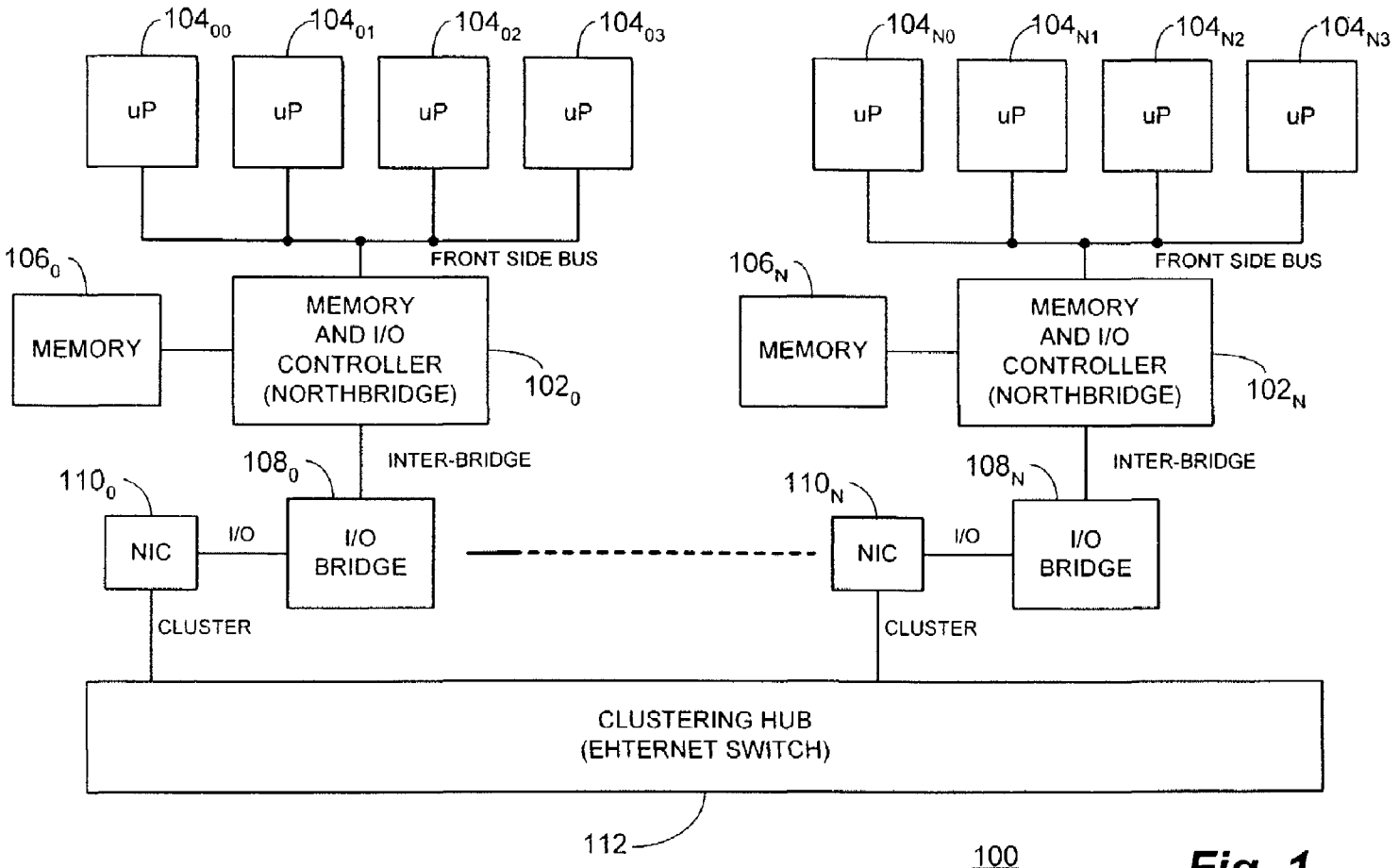


Fig. 1
Prior Art

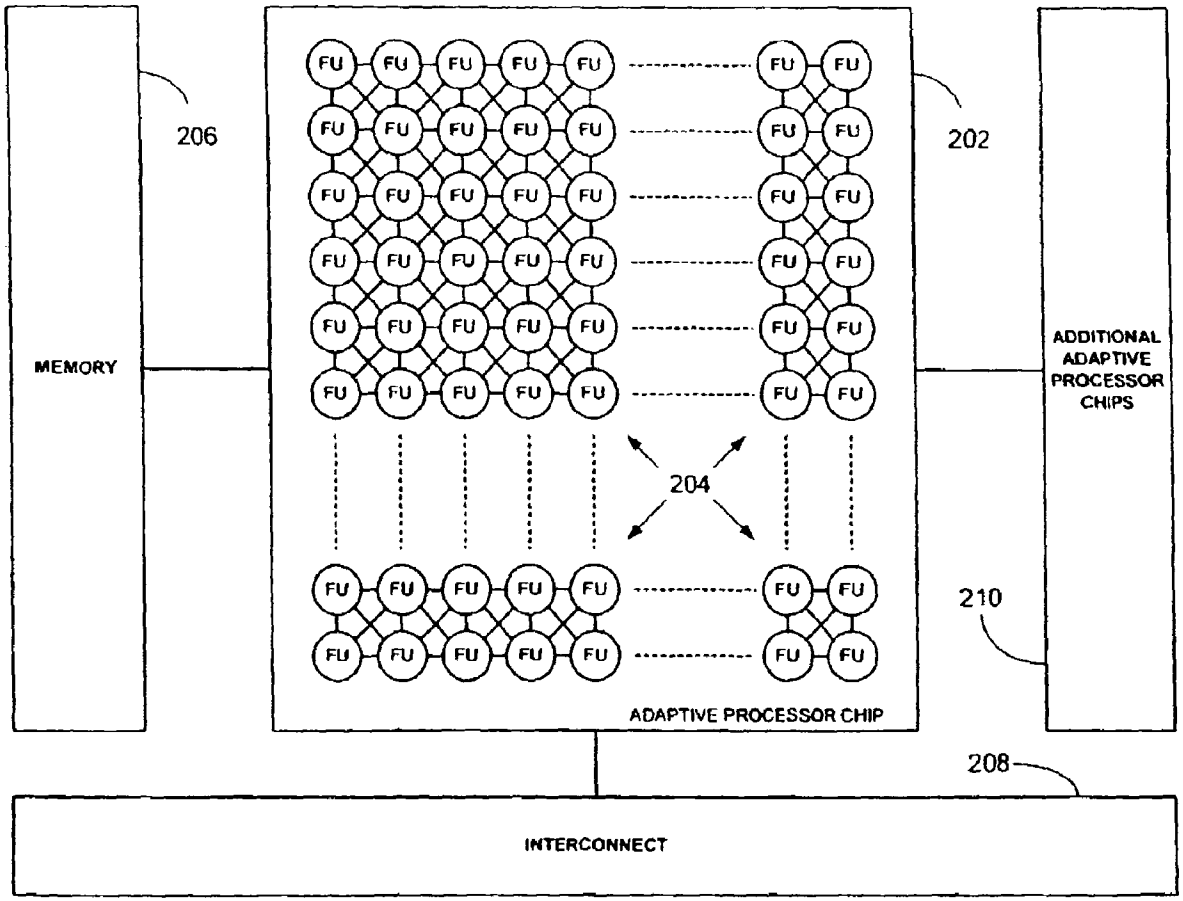

206
202
MEMORY
ADDITIONAL ADAPTIVE PROCESSOR CHIPS
FU
204
210
ADAPTIVE PROCESSOR CHIP
208
INTERCONNECT
200

Fig. 2

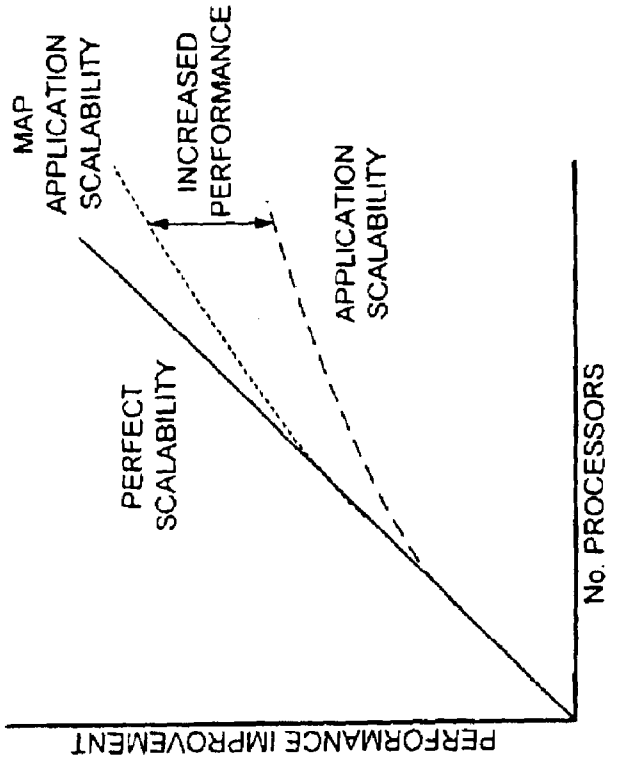


Fig. 3B

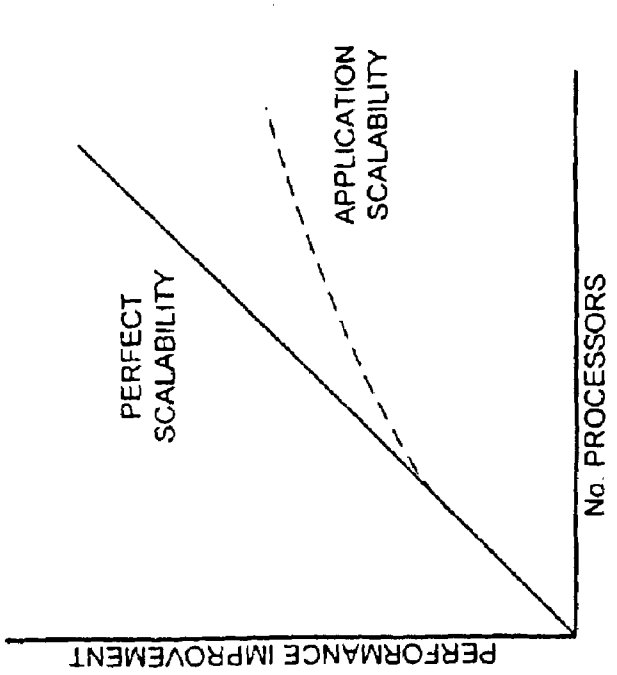
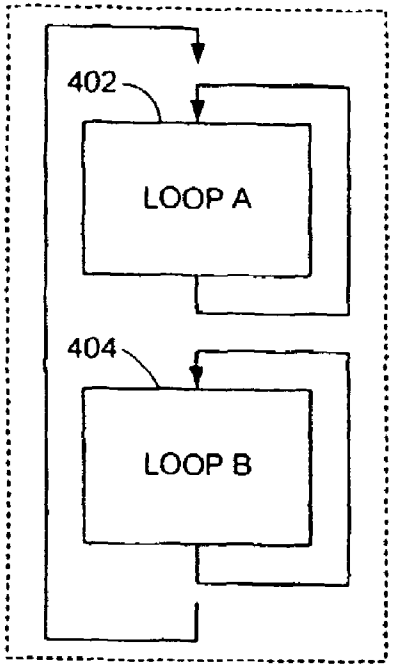


Fig. 3A
PRIOR ART

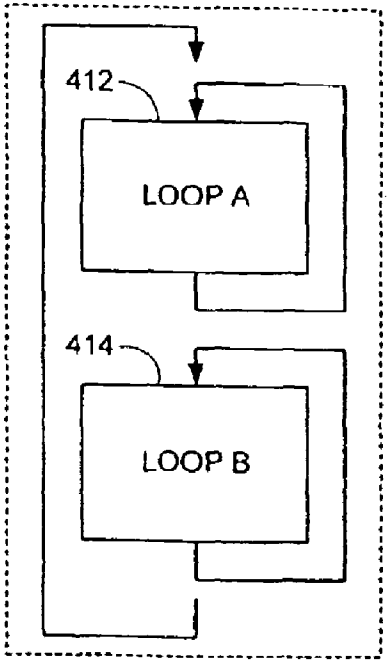


PHASE 1
- LOOP A "ACTIVE"
- LOOP B "INACTIVE"

PHASE 2
- LOOP A "INACTIVE"
- LOOP B "ACTIVE"

400

Fig. 4A
Prior Art



PHASE 1
- LOOP A WORKS ON DIMENSION 1
- LOOP B WORKS ON DIMENSION 0
(may be dummy data)

PHASE 2
- LOOP A WORKS ON DIMENSION 2
- LOOP B WORKS ON DIMENSION 1

410

Fig. 4B

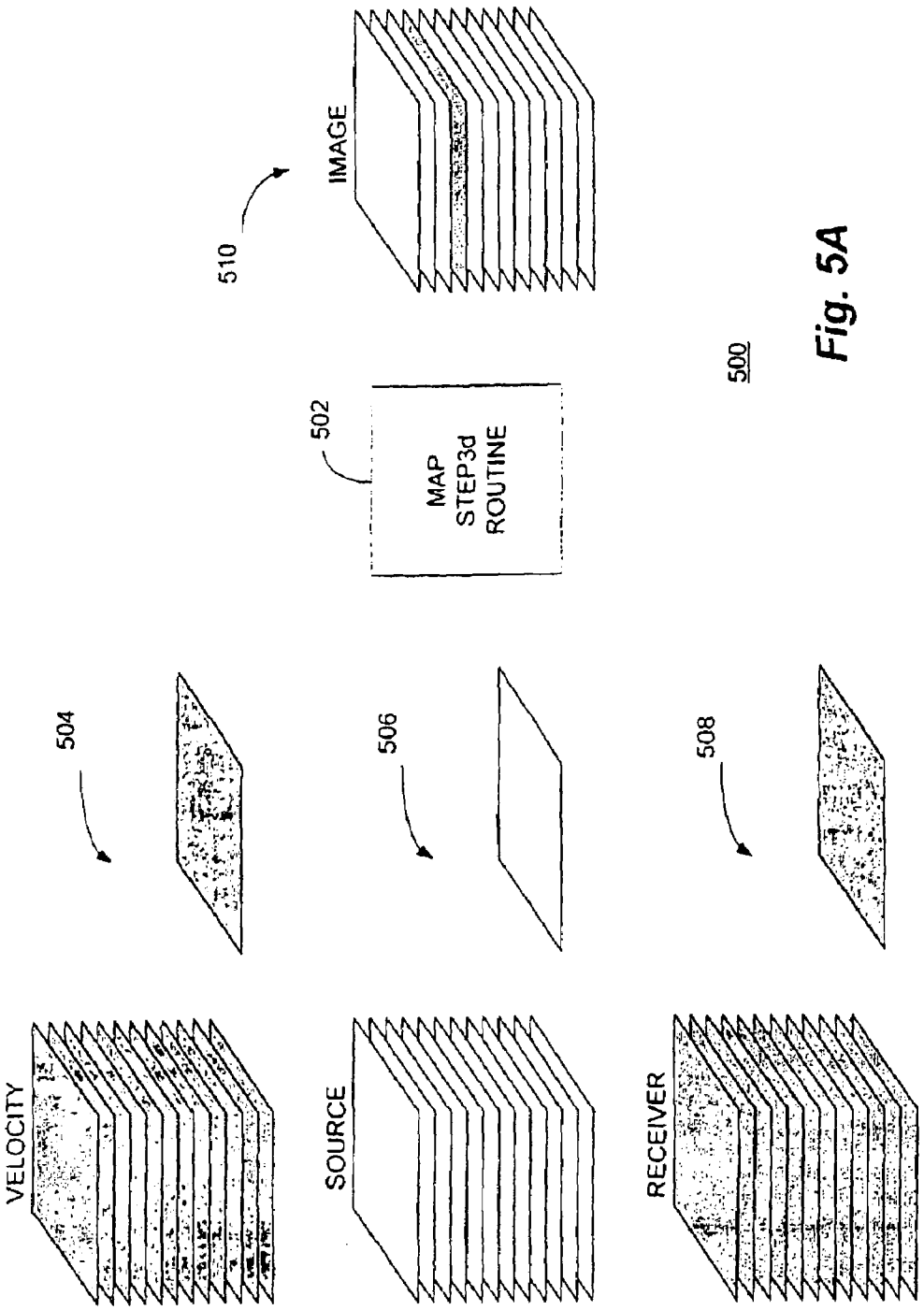

500
510
IMAGE
502
MAP
STEP3d
ROUTINE
504
506
508
VELOCITY
SOURCE
RECEIVER

Fig. 5A

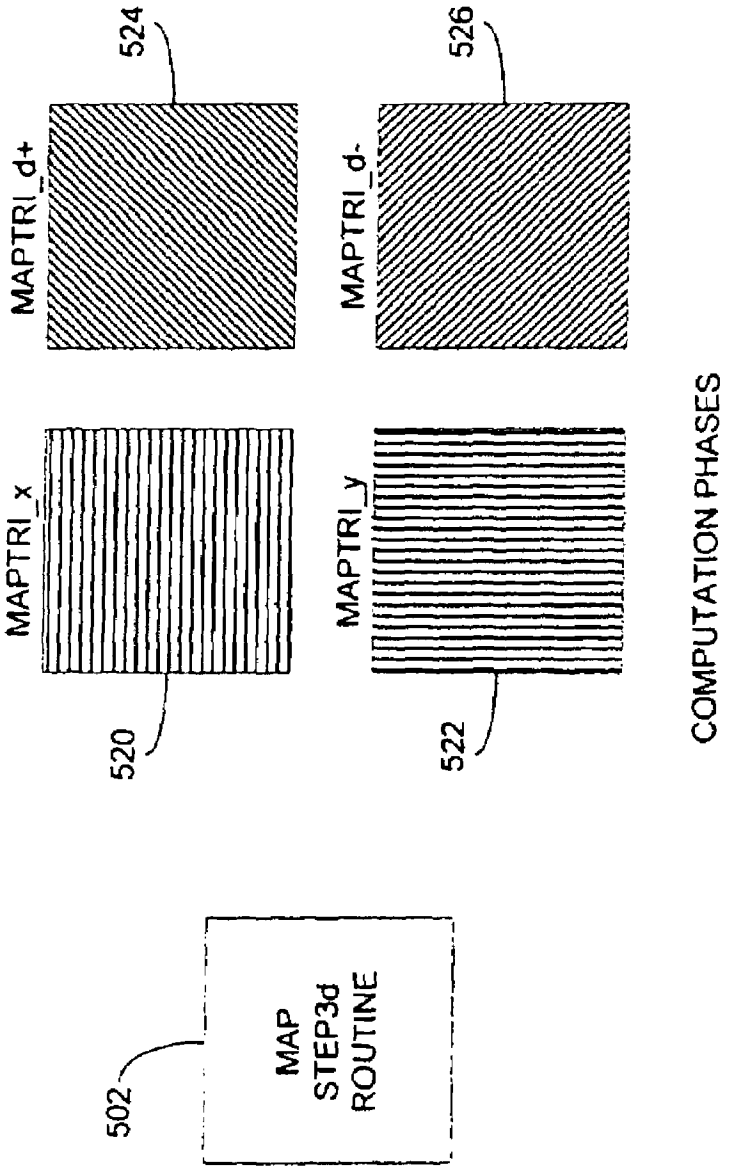

524
526
MAPTRI_d+
MAPTRI_d-
MAPTRI_x
MAPTRI_y
520
522
COMPUTATION PHASES
502
MAP STEP3d ROUTINE

Fig. 5B

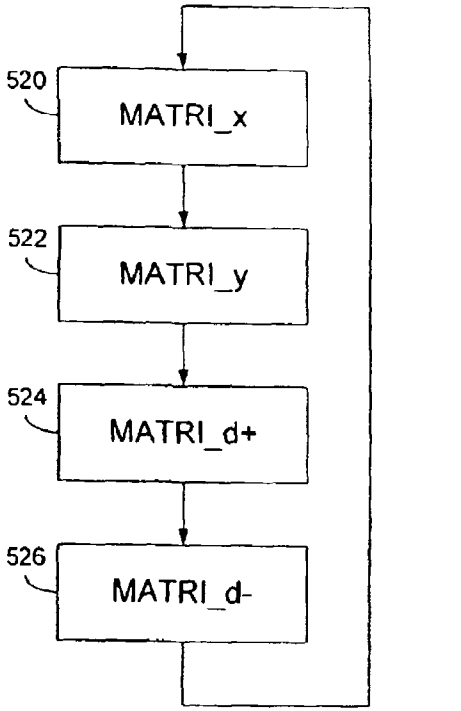

520
MATRI_x
522
MATRI_y
524
MATRI_d+
526
MATRI_d-
600

Fig. 6A

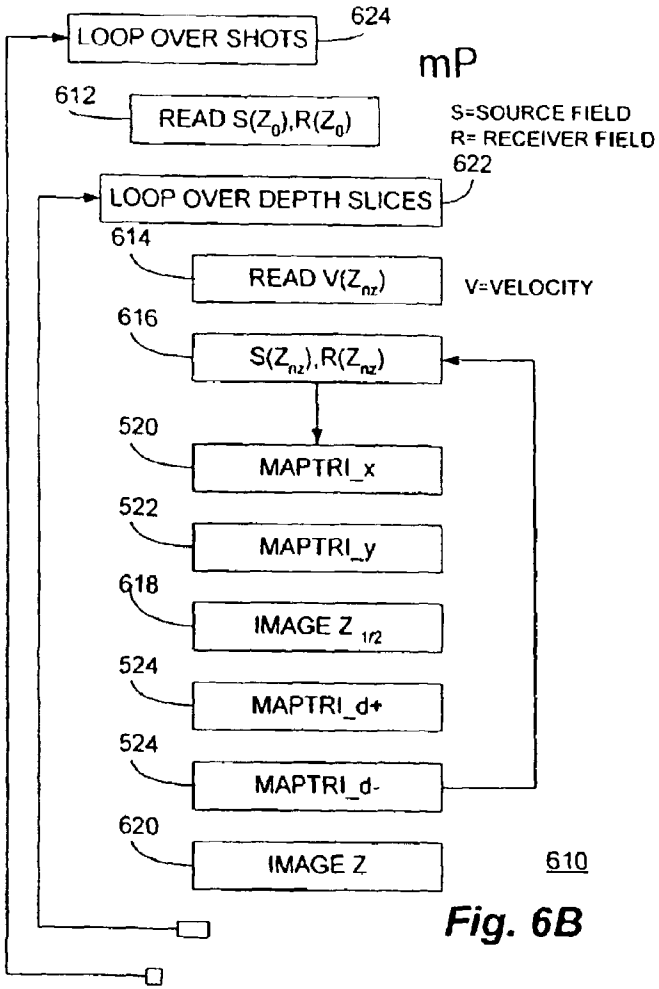

624
LOOP OVER SHOTS
mP
612
READ S(Z0),R(Z0)
S=SOURCE FIELD
R= RECEIVER FIELD
622
LOOP OVER DEPTH SLICES
614
READ V(Znz)
V=VELOCITY
616
S(Znz),R(Znz)
520
MAPTRI_x
522
MAPTRI_y
618
IMAGE Z 1/2
524
MAPTRI_d+
524
MAPTRI_d-
620
IMAGE Z
610

Fig. 6B

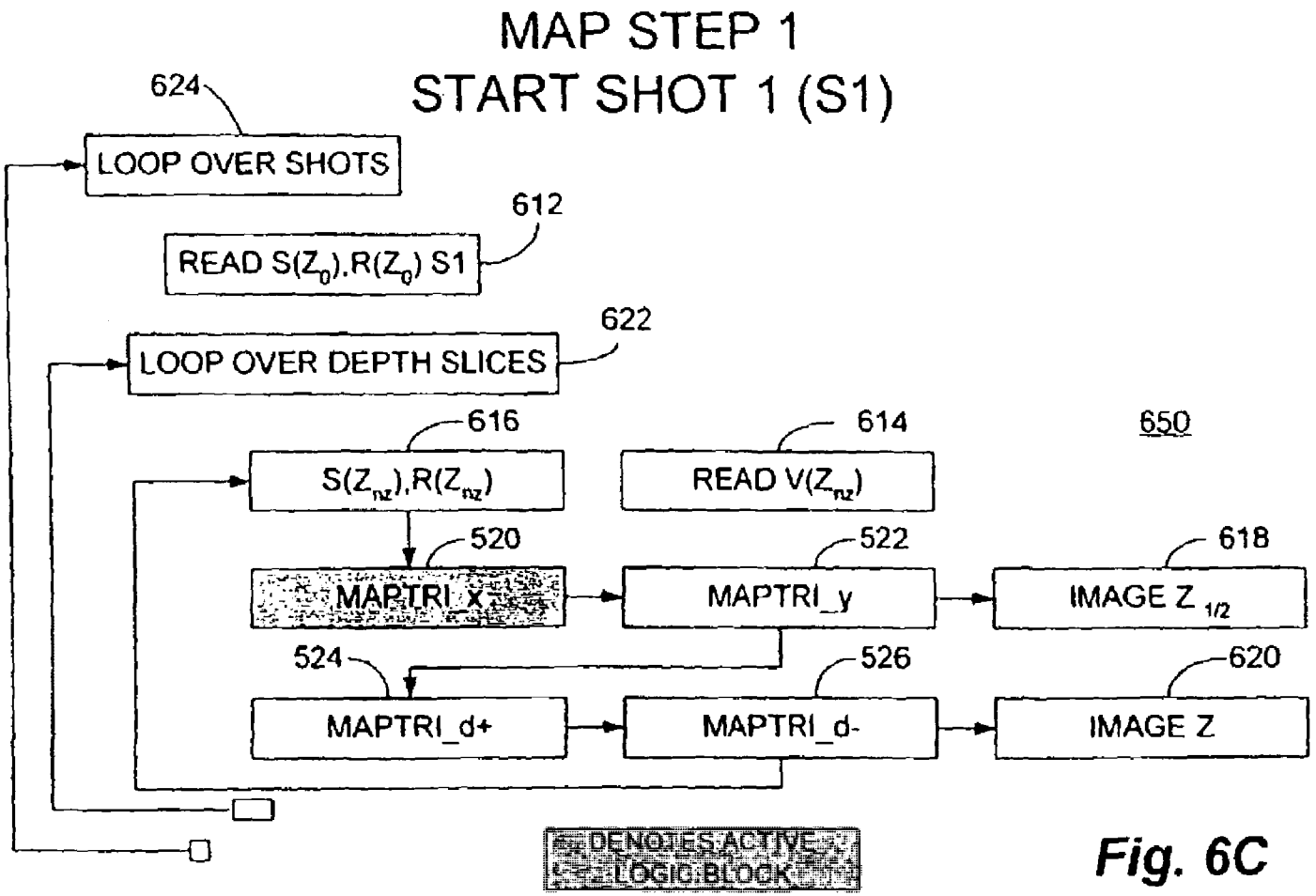

MAP STEP 1
START SHOT 1 (S1)
624
LOOP OVER SHOTS
612
READ S(Z0).R(Z0) S1
622
LOOP OVER DEPTH SLICES
616
S(Znz).R(Znz)
614
READ V(Znz)
650
520
MAPTRI_x
522
MAPTRI_y
618
IMAGE Z 1/2
524
MAPTRI_d+
526
MAPTRI_d-
620
IMAGE Z
DENOTES ACTIVE LOGIC BLOCK

Fig. 6C

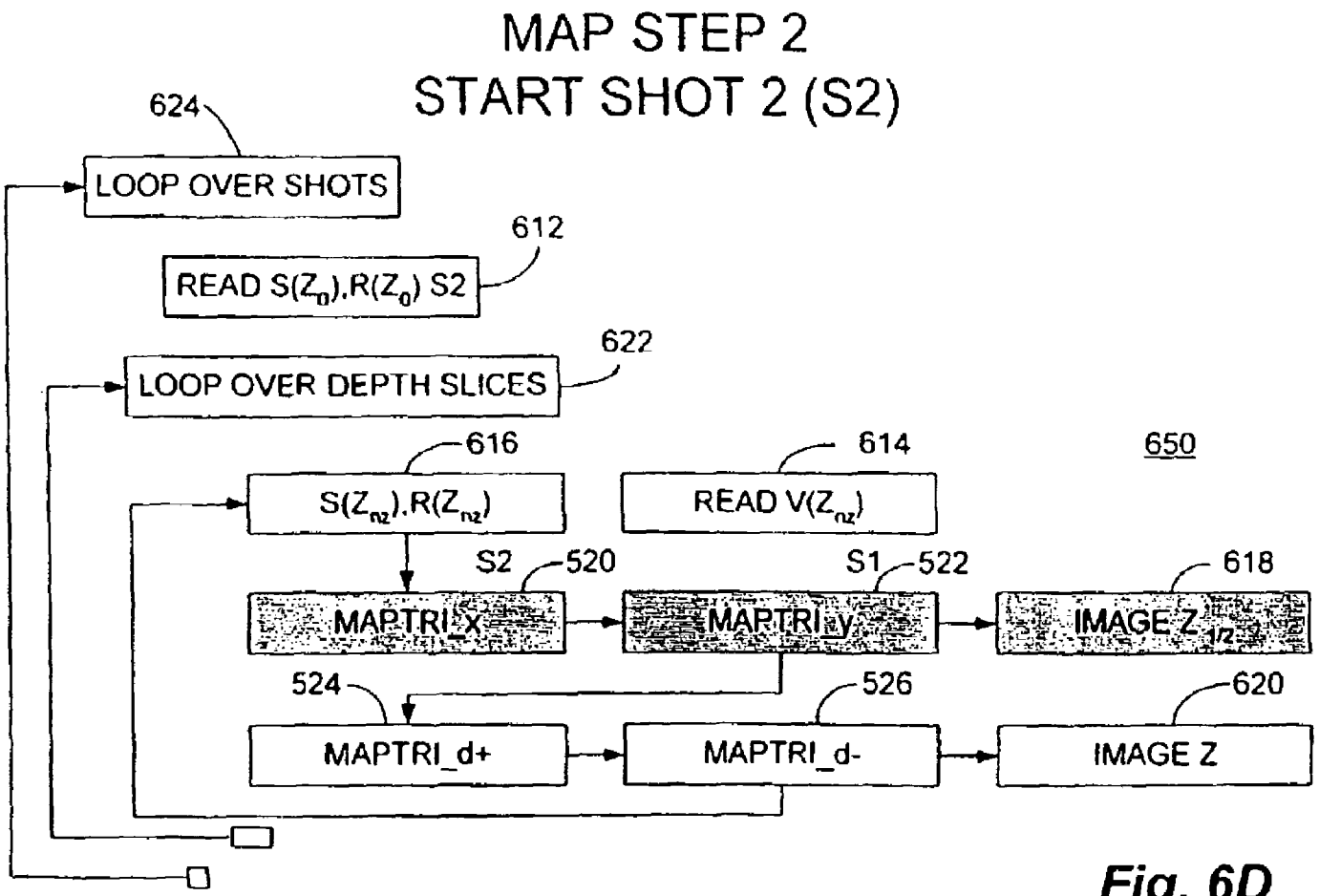

MAP STEP 2
START SHOT 2 (S2)
624
LOOP OVER SHOTS
612
READ S(Z0),R(Z0) S2
622
LOOP OVER DEPTH SLICES
616
S(Zn2),R(Zn2)
614
READ V(Zn2)
650
S2
520
S1
522
618
MAPTRI_x
MAPTRI_y
IMAGE Z12
524
526
620
MAPTRI_d+
MAPTRI_d-
IMAGE Z

Fig. 6D

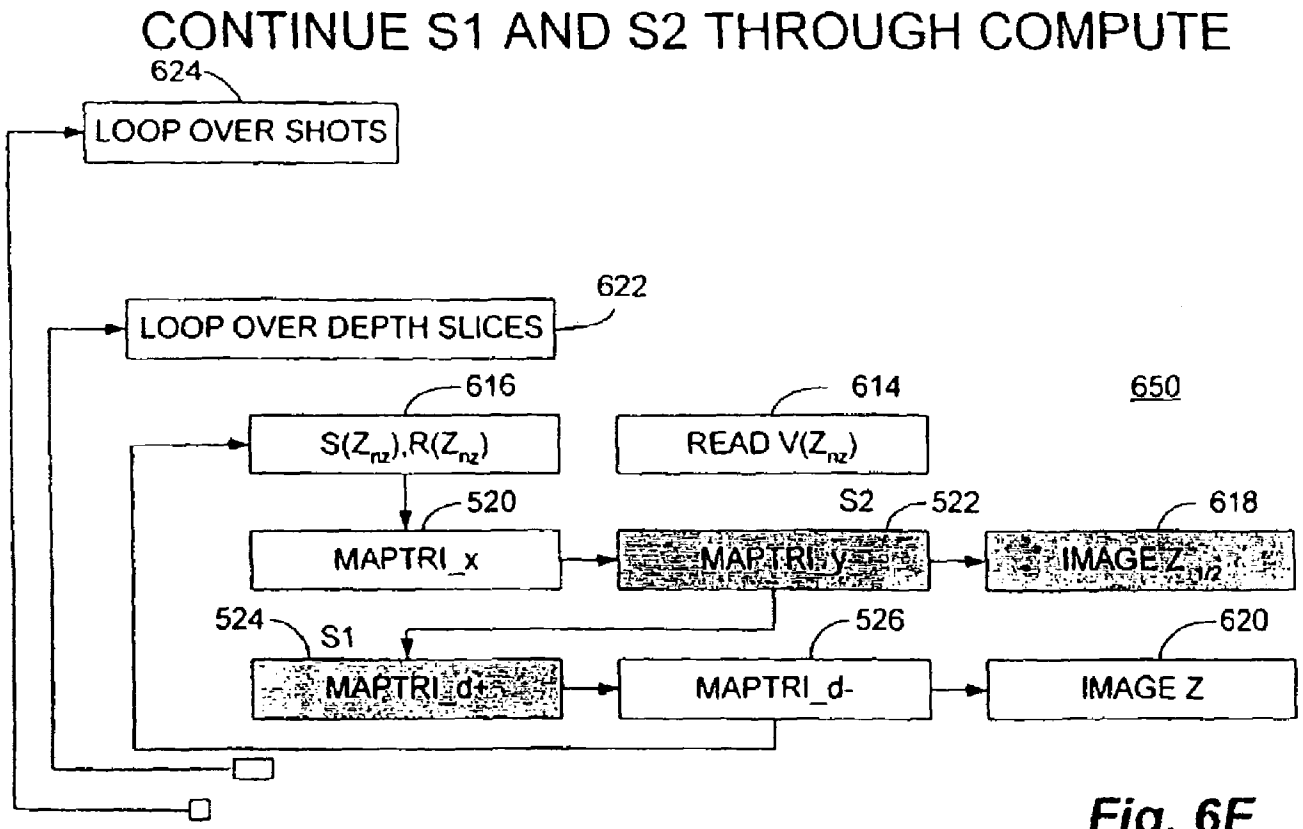

CONTINUE S1 AND S2 THROUGH COMPUTE
624
LOOP OVER SHOTS
622
LOOP OVER DEPTH SLICES
616
S(Z_nz),R(Z_nz)
614
READ V(Z_nz)
650
520
MAPTRI_x
S2
522
MAPTRI_y
618
IMAGE Z_n/2
524
S1
526
620
MAPTRI_d+
MAPTRI_d-
IMAGE Z

Fig. 6E

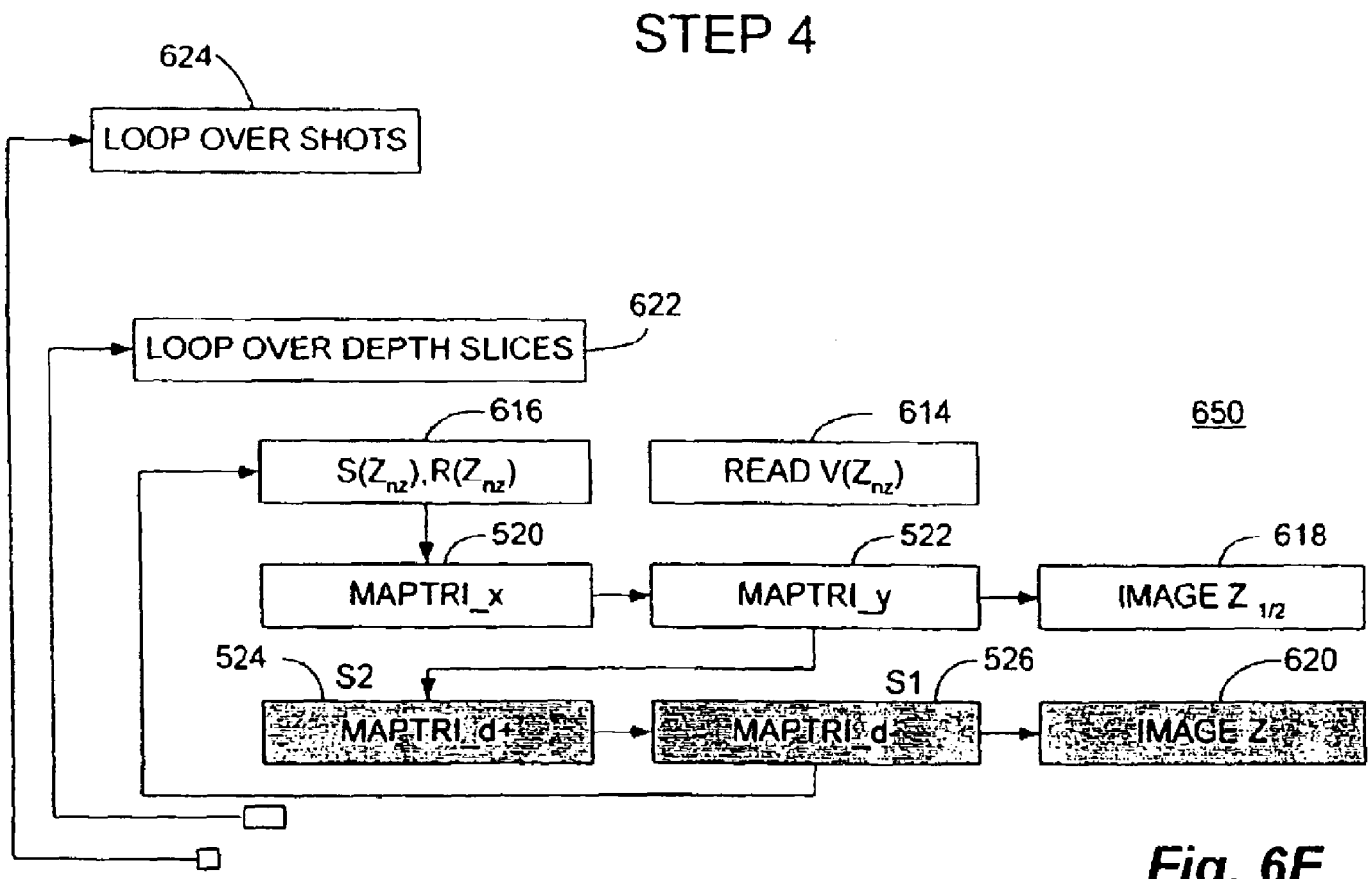

STEP 4
624
LOOP OVER SHOTS
622
LOOP OVER DEPTH SLICES
616
S(Z_nz),R(Z_nz)
614
READ V(Z_nz)
650
520
MAPTRI_x
522
MAPTRI_y
618
IMAGE Z 1/2
524
S2
526
S1
620
MAPTRI_d+1
MAPTRI_d
IMAGE Z

Fig. 6F

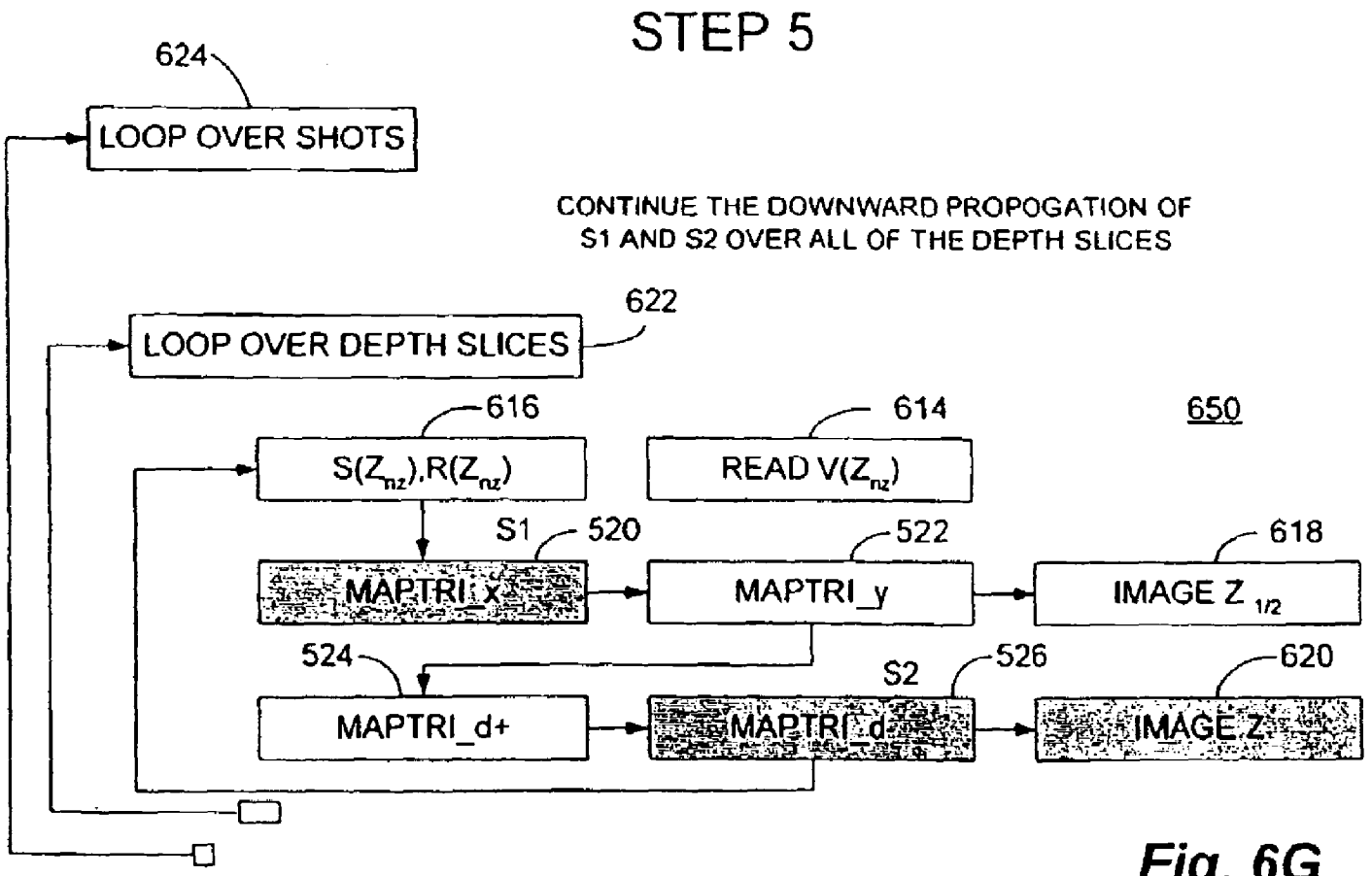

STEP 5
624
LOOP OVER SHOTS
CONTINUE THE DOWNWARD PROPOGATION OF S1 AND S2 OVER ALL OF THE DEPTH SLICES
622
LOOP OVER DEPTH SLICES
616
614
650
S(Z_nz),R(Z_nz)
READ V(Z_nz)
S1
520
522
618
MAPTRI_x
MAPTRI_y
IMAGE Z 1/2
524
S2
526
620
MAPTRI_d+
MAPTRI_d-
IMAGE Z

Fig. 6G

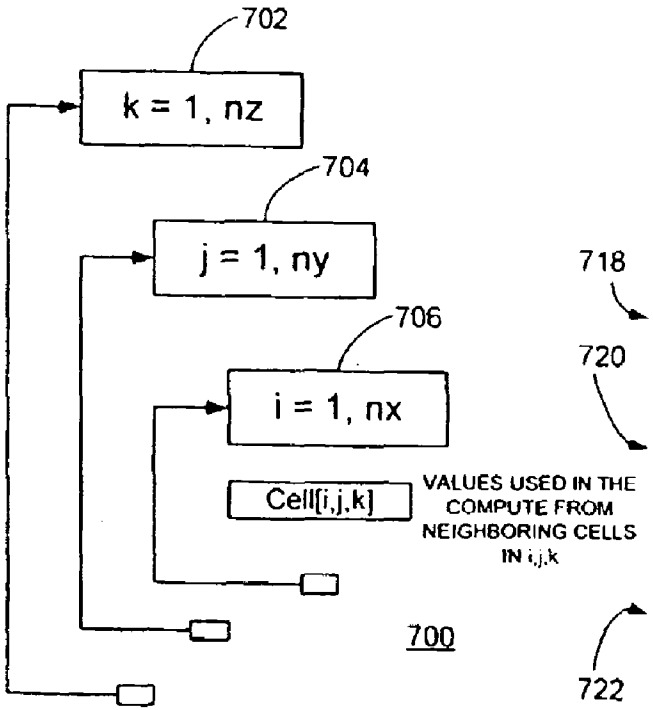

702
k = 1, nz
704
j = 1, ny
706
i = 1, nx
Cell[i,j,k]
VALUES USED IN THE COMPUTE FROM NEIGHBORING CELLS IN i,j,k
700

Fig. 7A

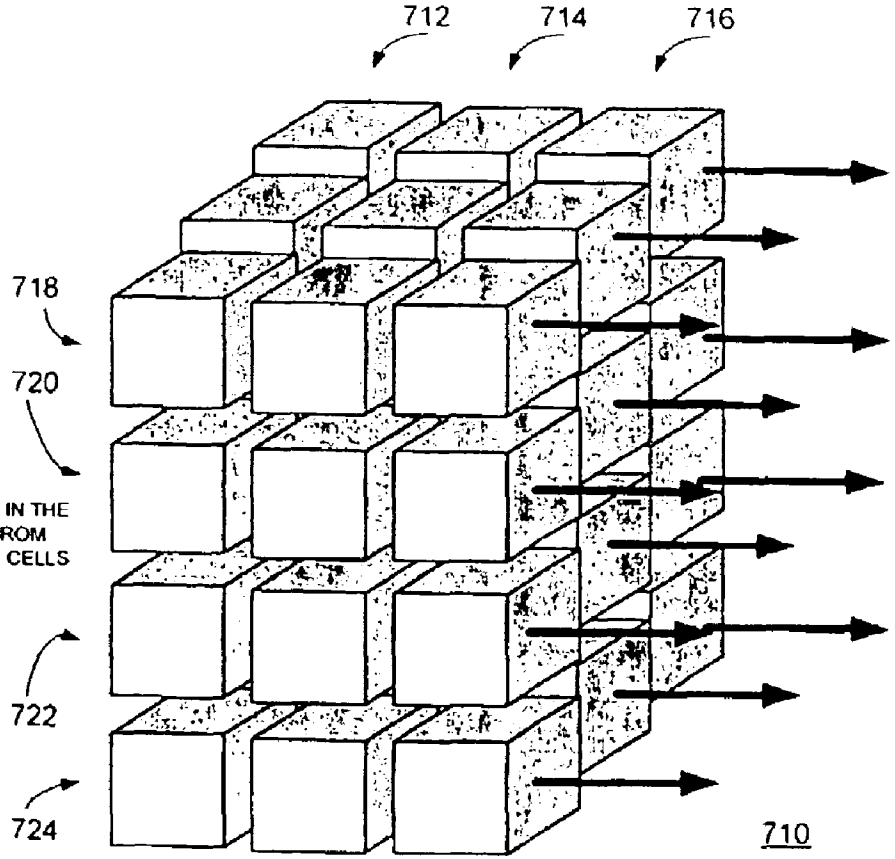
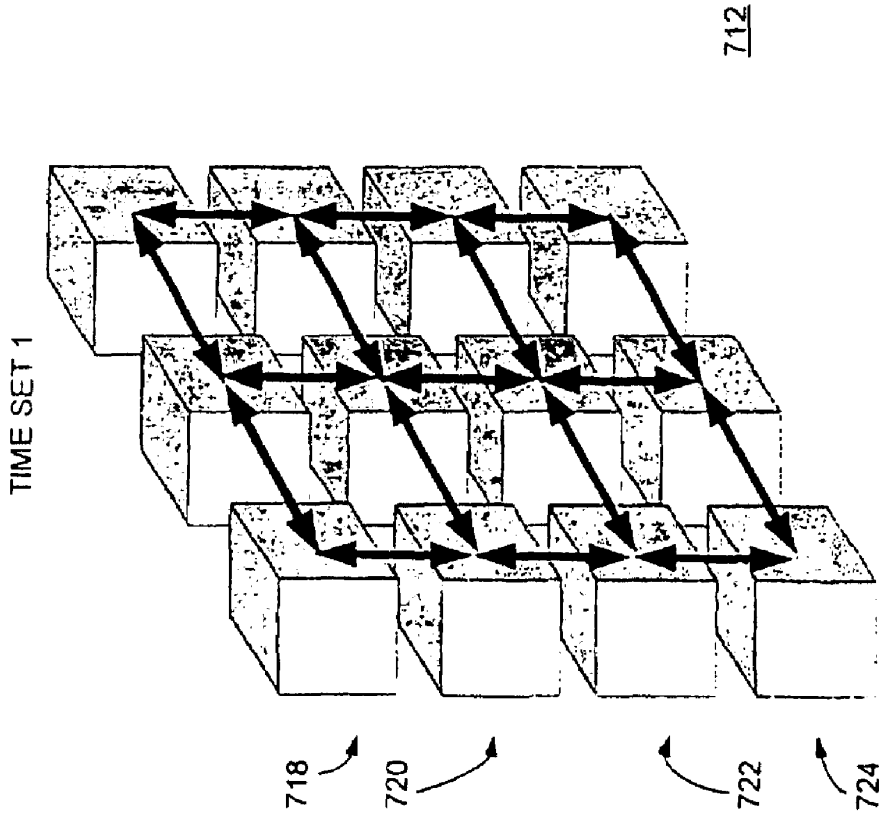

712
714
716
718
720
722
724
710

Fig. 7B

Fig.7C



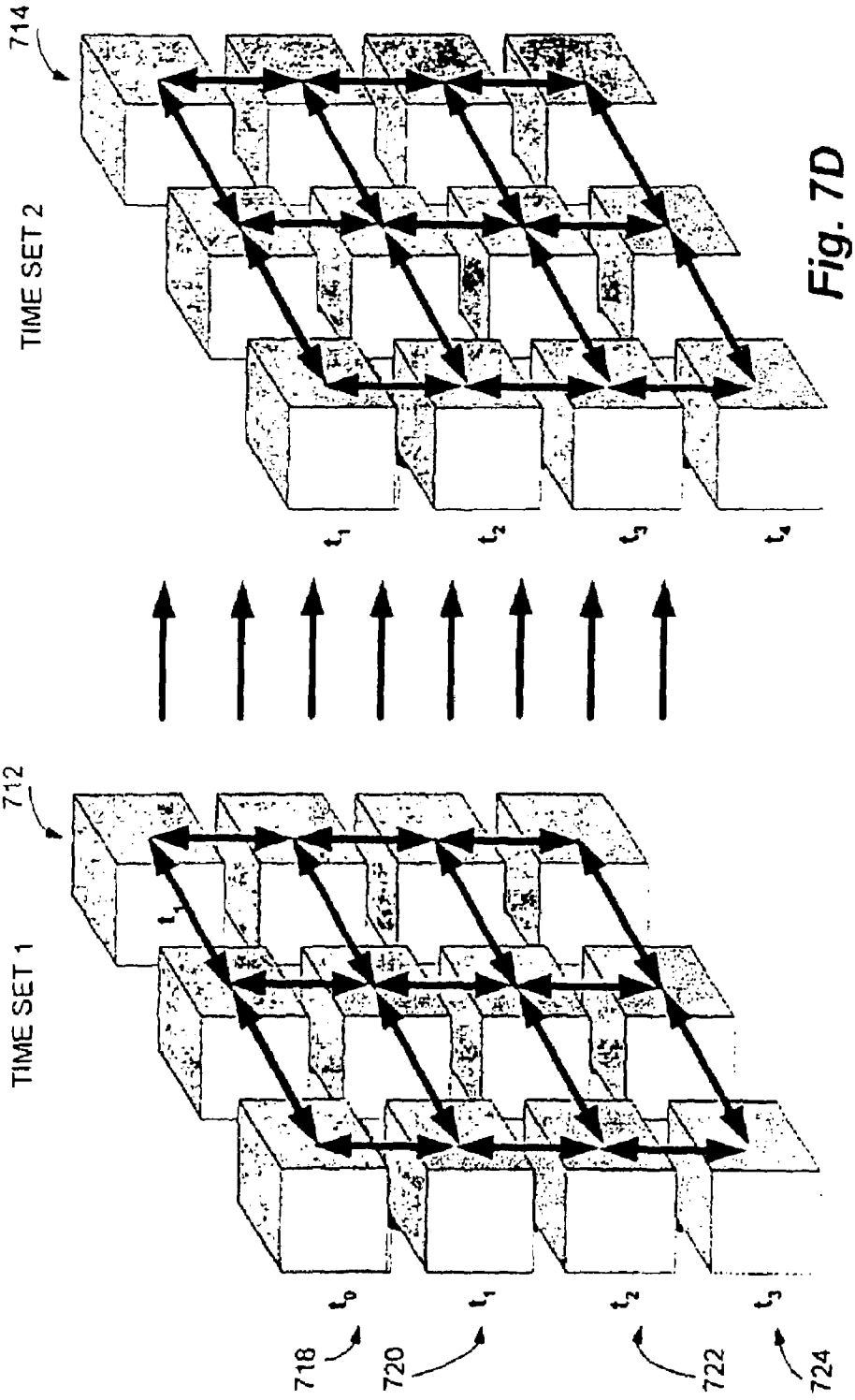


Fig. 7D

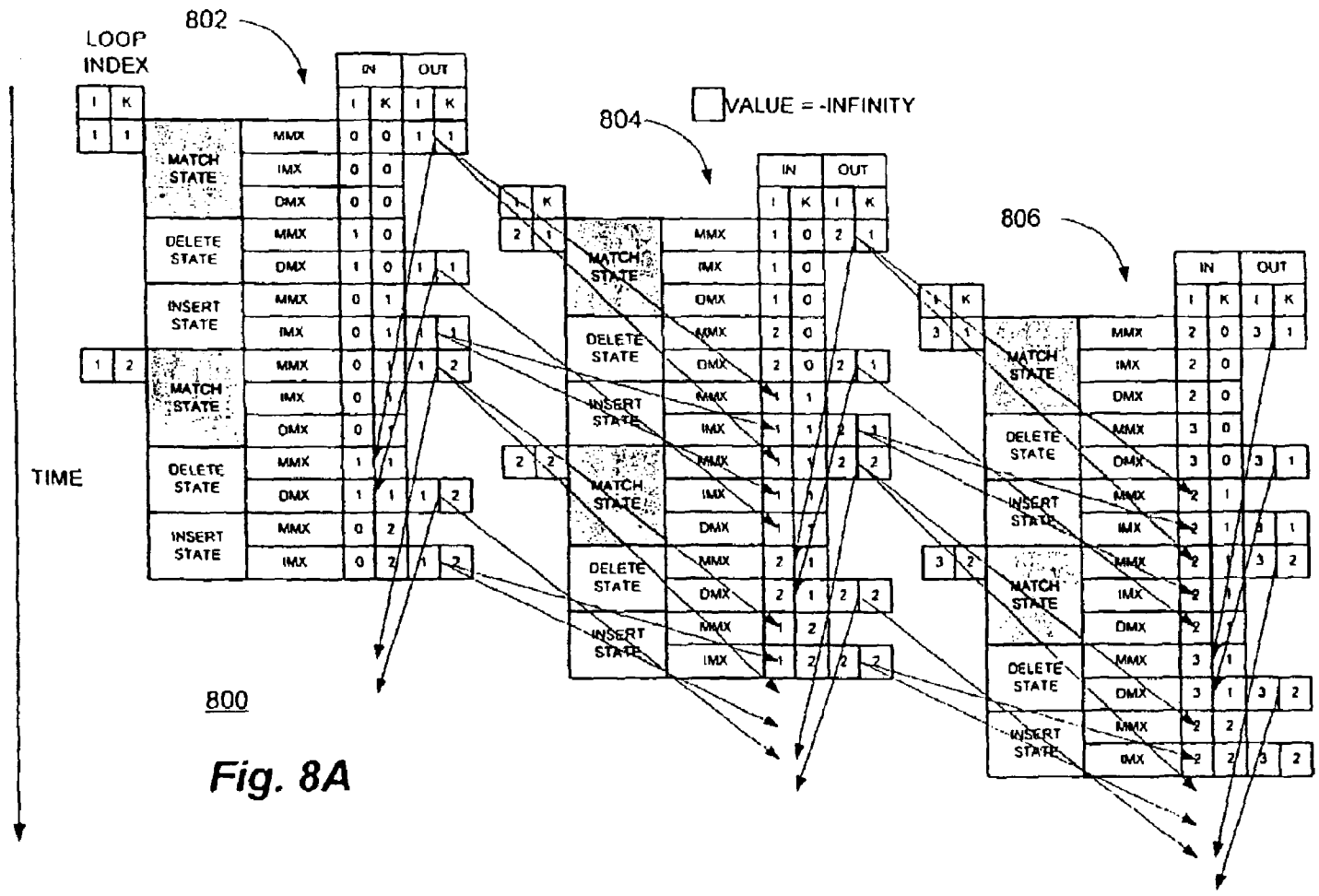

802
LOOP INDEX
IN
OUT
MATCH STATE
DELETE STATE
INSERT STATE
MMX
IMX
DMX
804
VALUE = -INFINITY
806
TIME
800

Fig. 8A

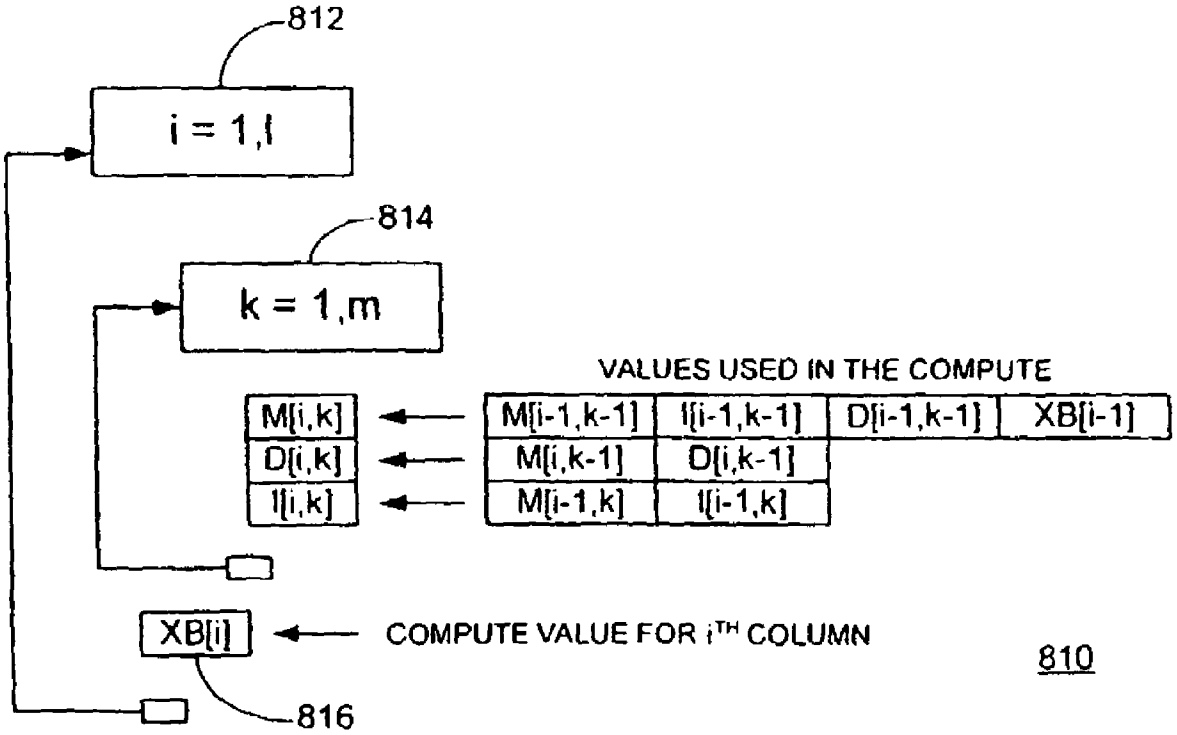
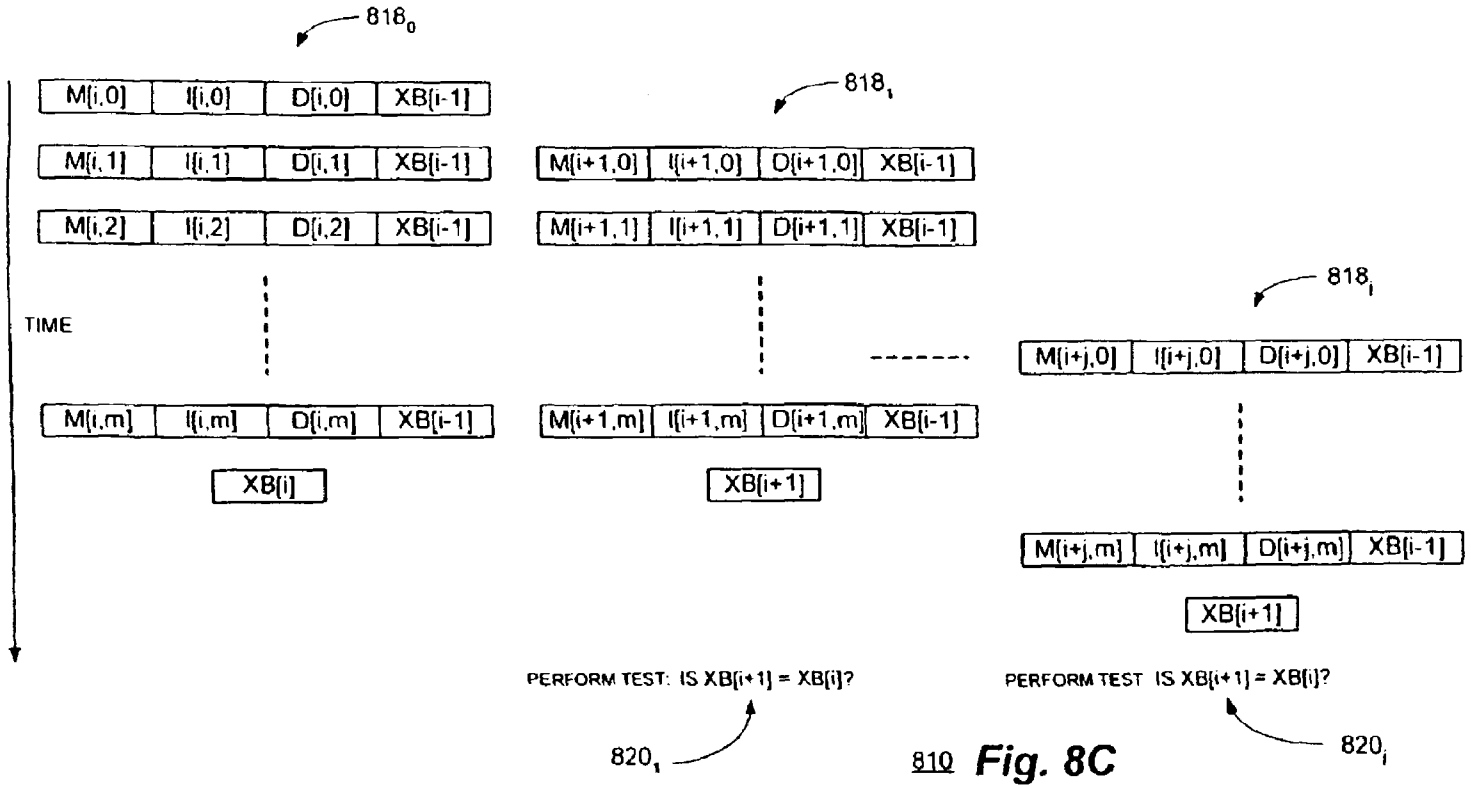
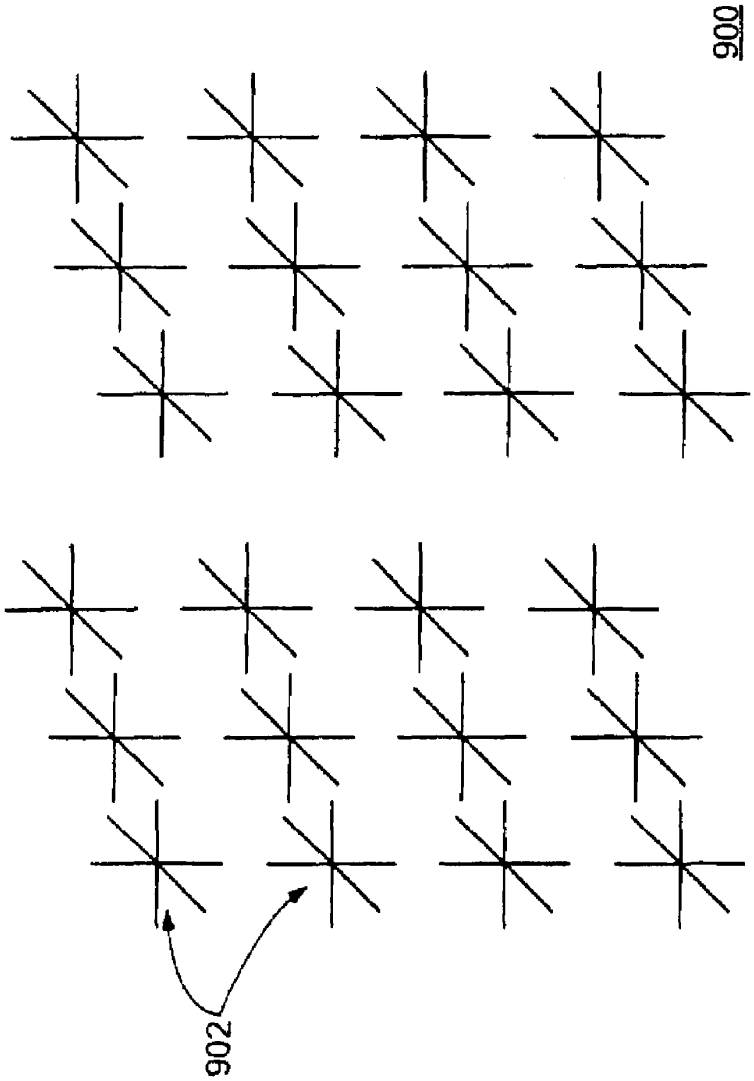

812
i = 1,I
814
k = 1,m
VALUES USED IN THE COMPUTE
M[i,k]
D[i,k]
I[i,k]
M[i-1,k-1]
I[i-1,k-1]
D[i-1,k-1]
XB[i-1]
M[i,k-1]
D[i,k-1]
M[i-1,k]
I[i-1,k]
XB[i]
COMPUTE VALUE FOR iTH COLUMN
816
810

Fig. 8B


818$_0$
M[i,0] I[i,0] D[i,0] XB[i-1]
M[i,1] I[i,1] D[i,1] XB[i-1]
M[i,2] I[i,2] D[i,2] XB[i-1]
TIME
M[i,m] I[i,m] D[i,m] XB[i-1]
XB[i]
818$_1$
M[i+1,0] I[i+1,0] D[i+1,0] XB[i-1]
M[i+1,1] I[i+1,1] D[i+1,1] XB[i-1]
M[i+1,m] I[i+1,m] D[i+1,m] XB[i-1]
XB[i+1]
818$_i$
M[i+j,0] I[i+j,0] D[i+j,0] XB[i-1]
M[i+j,m] I[i+j,m] D[i+j,m] XB[i-1]
XB[i+1]
PERFORM TEST: IS XB[i+1] = XB[i]?
820$_1$
PERFORM TEST IS XB[i+1] = XB[i]?
820$_i$

810 *Fig. 8C*

Fig. 9A



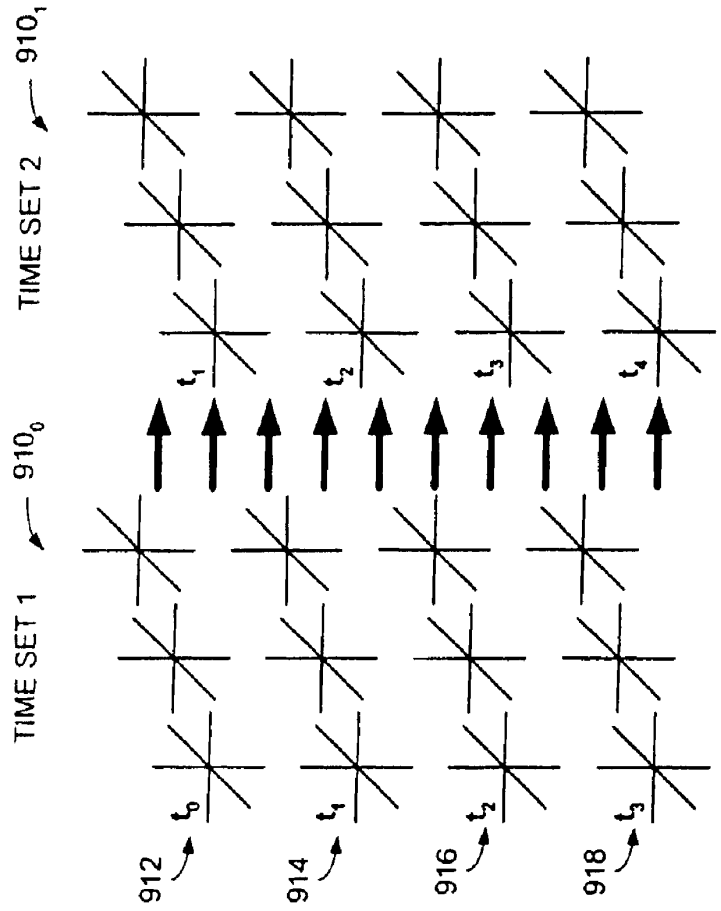

TIME SET 2
910_1
TIME SET 1
910_0
t_1
t_2
t_3
t_4
t_0
t_1
t_2
t_3
912
914
916
918

Fig. 9C

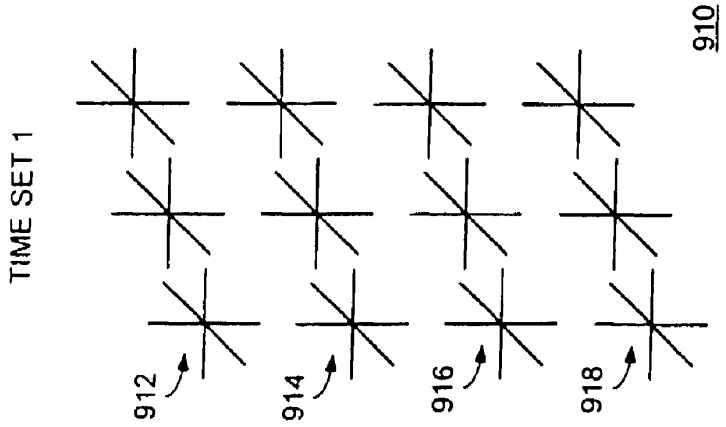

910
TIME SET 1
912
914
916
918

Fig. 9B

MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS

CROSS REFERENCE TO RELATED PATENT APPLICATIONS

The present application is a Continuation of U.S. patent application Ser. No. 10/285,318 filed Oct. 31, 2002 which is related to the subject matter of U.S. patent application Ser. No. 09/755,744 filed Jan. 5, 2001 for: "Multiprocessor Computer Architecture Incorporating a Plurality of Memory Algorithm Processors in the Memory Subsystem" and is further related to the subject matter of U.S. Pat. No. 6,434,687 for: "System and Method for Accelerating Web Site Access and Processing Utilizing a Computer System Incorporating Reconfigurable Processors Operating Under a Single Operating System Image", all of which are assigned to SRC Computers, Inc., Colorado Springs, Colo. and the disclosures of which are herein specifically incorporated in their entirety by this reference.

COPYRIGHT NOTICE/PERMISSION

BACKGROUND OF THE INVENTION

The present invention relates, in general, to the field of computing systems and techniques. More particularly, the present invention relates to multi-adaptive processing systems and techniques for enhancing parallelism and performance of computational functions.

Currently, most large software applications achieve high performance operation through the use of parallel processing. This technique allows multiple processors to work simultaneously on the same problem to achieve a solution in a fraction of the time required for a single processor to accomplish the same result. The processors in use may be performing many copies of the same operation, or may be performing totally different operations, but in either case all processors are working simultaneously.

The use of such parallel processing has led to the proliferation of both multi-processor boards and large scale clustered systems. However, as more and more performance is required, so is more parallelism, resulting in ever larger systems. Clusters exist today that have tens of thousands of processors and can occupy football fields of space. Systems of such a large physical size present many obvious downsides, including, among other factors, facility requirements, power, heat generation and reliability.

SUMMARY OF THE INVENTION

However, if a processor technology could be employed that offers orders of magnitude more parallelism per processor, these systems could be reduced in size by a comparable factor. Such a processor or processing element is possible through the use of a reconfigurable processor. Reconfigurable processors instantiate only the functional units needed to solve a particular application, and as a result, have available space to instantiate as many functional units as may be required to solve the problem up to the total capacity of the integrated circuit chips they employ.

At present, reconfigurable processors, such as multi-adaptive processor elements (MAP™, a trademark of SRC Computers, Inc.) can achieve two to three orders of magnitude more parallelism and performance than state-of-the-art microprocessors. Through the advantageous application of adaptive processing techniques as disclosed herein, this type of reconfigurable processing parallelism may be employed in a variety of applications resulting in significantly higher performance than that which can now be achieved while using significantly smaller and less expensive computer systems.

However, in addition to these benefits, there is an additional much less obvious one that can have even greater impact on certain applications and has only become available with the advent of multi-million gate reconfigurable chips. Performance gains are also realized by reconfigurable processors due to the much tighter coupling of the parallel functional units within each chip than can be accomplished in a microprocessor based computing system.

In a multi-processor, microprocessor-based system, each processor is allocated but a relatively small portion of the total problem called a cell. However, to solve the total problem, results of one processor are often required by many adjacent cells because their cells interact at the boundary and upwards of six or more cells, all having to interact to compute results, would not be uncommon. Consequently, intermediate results must be passed around the system in order to complete the computation of the total problem. This, of necessity, involves numerous other chips and busses that run at much slower speeds than the microprocessor thus resulting in system performance often many orders of magnitude lower than the raw computation time.

On the other hand, in the use of an adaptive processor-based system, since ten to one thousand times more computations can be performed within a single chip, any boundary data that is shared between these functional units need never leave a single integrated circuit chip. Therefore, data moving around the system, and its impact on reducing overall system performance, can also be reduced by two or three orders of magnitude. This will allow both significant improvements in performance in certain applications as well as enabling certain applications to be performed in a practical timeframe that could not previously be accomplished.

Particularly disclosed herein is a method for data processing in a reconfigurable computing system comprising a plurality of functional units. The method comprises: defining a calculation for the reconfigurable computing system; instantiating at least two of the functional units to perform the calculation; utilizing a first of the functional units to operate upon a subsequent data dimension of the calculation and substantially concurrently utilizing a second of the functional units to operate upon a previous data dimension of the calculation.

Further disclosed herein is a method for data processing in a reconfigurable computing system comprising a plurality of functional units. The method comprises: defining a first systolic wall comprising rows of cells forming a subset of the plurality of functional units; computing a value at each of the cells in at least a first row of the first systolic wall; communicating the values between cells in the first row of the cells to produce updated values; communicating the updated values

to a second row of the first systolic wall; and substantially concurrently providing the updated values to a first row of a second systolic wall of rows of cells in the subset of the plurality of functional units.

Also disclosed herein is a method for data processing in a reconfigurable processing system which includes setting up a systolic processing form employing a speculative processing strategy.

BRIEF DESCRIPTION OF THE DRAWINGS

The aforementioned and other features and objects of the present invention and the manner of attaining them will become more apparent and the invention itself will be best understood by reference to the following description of a preferred embodiment taken in conjunction with the accompanying drawings, wherein:

FIG. **1** is a simplified functional block diagram of typical clustered inter-processor communications path in a conventional multi-processor computing system;

FIG. **2** is a functional block diagram of an adaptive processor communications path illustrating the many functional units ("FU") interconnected by reconfigurable routing resources within the adaptive processor chip;

FIG. **3**A is a graph of the actual performance improvement versus the number of processors utilized and illustrating the deviation from perfect scalability of a particular application utilizing a conventional multi-processor computing system such as that illustrated in FIG. **1**;

FIG. **3**B is a corresponding graph of the actual performance improvement versus the number of processors utilized and illustrating the performance improvement over a conventional multi-processor computing system utilizing an adaptive processor-based computing system such as that illustrated in FIG. **2**;

FIG. **4**A is a simplified logic flowchart illustrating a conventional sequential processing operation in which nested Loops A and B are alternately active on different phases of the process;

FIG. **4**B is a comparative, simplified logic flowchart illustrating multi-dimensional processing in accordance with the technique of the present invention wherein multiple dimensions of data are processed by both Loops A and B such that the computing system logic is operative on every clock cycle;

FIG. **5**A is illustrative of a general process for performing a representative multi-dimensional pipeline operation in the form of a seismic migration imaging function utilizing the parallelism available in the utilization of the adaptive processing techniques of the present invention;

FIG. **5**B is a follow-on illustration of the computation phases employed in implementing the exemplary seismic migration imaging function of the preceding figure;

FIG. **6**A is a simplified logic flowchart for a particular seismic migration imaging application illustrative of the parallelism provided in the use of an adaptive processor-based computing system;

FIG. **6**B illustrates the computational process which may be employed by a microprocessor in the execution of the seismic imaging application of the preceding figure;

FIG. **6**C illustrates the first step in the computational process which may be employed by an adaptive processor in the execution of the seismic imaging application of FIG. **6**A in which a first shot (S**1**) is started;

FIG. **6**D illustrates the second step in the same computational process for the execution of the seismic imaging application of FIG. **6**A in which a second shot (S**2**) is started;

FIG. **6**E illustrates the third step in the same computational process for the execution of the seismic imaging application of FIG. **6**A in which the operation on the first and second shots is continued through compute;

FIG. **6**F illustrates the fourth step in the same computational process showing the subsequent operation on shots S**1** and S**2**;

FIG. **6**G illustrates the fifth step in the same computational process followed by the continued downward propagation of shots S**1** and S**2** over all of the depth slices;

FIG. **7**A illustrates a process for performing a representative systolic wavefront operation in the form of a reservoir simulation function also utilizing the parallelism available in the utilization of the adaptive processing techniques of the present invention;

FIG. **7**B illustrates the general computation of fluid flow properties in the reservoir simulation of the preceding figure which are communicated to neighboring cells;

FIG. **7**C illustrates the creation of a systolic wall of computation at Time Set **1** which has been started for a vertical wall of cells and in which communication of values between adjacent rows in the vertical wall can occur without storing values to memory;

FIG. **7**D is a follow on illustration of the creation of a systolic wall of computation at Time Set **1** and Time Set **2** showing how a second vertical wall of cells is started after the computation for cells in the corresponding row of the first wall has been completed;

FIG. **8**A illustrates yet another process for performing a representative systolic wavefront operation in the form of the systolic processing of bioinformatics also utilizing the parallelism available in the utilization of the adaptive processing techniques of the present invention;

FIG. **8**B illustrates a systolic wavefront processing operation which further incorporates a speculative processing strategy based upon an evaluation of the rate of change of XB;

FIG. **8**C is a further illustration of the systolic wavefront processing operation of the preceding figure incorporating speculative processing;

FIG. **9**A illustrates still another process for performing a representative systolic wavefront operation in the form of structure codes calculating polynomials at grid intersections, again utilizing the parallelism available in the utilization of the adaptive processing techniques of the present invention;

FIG. **9**B illustrates the computation start for a vertical wall of grid points at Time Set **1** for a polynomial evaluation performed on grid intersections wherein calculations between rows are done in a stochastic fashion using values from a previous row; and

FIG. **9**C is a further illustration of the polynomial evaluation performed on grid intersections of the preceding figure wherein a second wall is started after the cells in the corresponding row of the first wall have been completed.

DESCRIPTION OF A REPRESENTATIVE EMBODIMENT

This application incorporates by reference the entire disclosure of Caliga, D. et al. "Delivering Acceleration: "The Potential for Increased HPC Application Performance Using Reconfigurable Logic", SC2001, November 2001, ACM 1-58113-293-X/01/0011.

With reference now to FIG. **1**, a simplified functional block diagram of typical clustered inter-processor communications path in a conventional multi-processor computing system **100** is shown. The computer system comprises a number of memory and input/output ("I/O" controller integrated circuits

("ICs") **102**$_0$ through **102**$_N$, (e.g. "North Bridge") **102** such as the P4X333/P4X400 devices available from VIA Technologies, Inc.; the M1647 device available from Acer Labs, Inc. and the 824430X device available from Intel Corporation. The North Bridge IC **102** is coupled by means of a Front Side Bus ("FSB") to one or more microprocessors **104**$_{00}$ though **104**$_{03}$ and **104**$_{N0}$ through **104**$_{N3}$ such as one of the Pentium® series of processors also available from Intel Corporation.

The North Bridge ICs **102**$_0$ through **102**$_N$ are coupled to respective blocks of memory **106**$_0$ through **106**$_N$ as well as to a corresponding I/O bridge element **108**$_0$ through **108**$_N$. A network interface card ("NIC") **110**$_0$ through **210**$_N$ couples the I/O bus of the respective I/O bridge **108**$_0$ through **108**$_N$ to a cluster bus coupled to a common clustering hub (or Ethernet Switch) **112**.

Since typically a maximum of four microprocessors **104**, each with two or four functional units, can reside on a single Front Side Bus, any communication to more than four must pass over the Front Side Bus, inter-bridge bus, input/output ("I/O") bus, cluster interconnect (e.g. an Ethernet clustering hub **112**) and then back again to the receiving processor **104**. The I/O bus is typically an order of magnitude lower in bandwidth than the Front Side Bus, which means that any processing involving more than the four processors **104** will be significantly throttled by the loose coupling caused by the interconnect. All of this is eliminated with a reconfigurable processor having hundreds or thousands of functional units per processor.

With reference additionally now to FIG. **2**, a functional block diagram of an adaptive processor **200** communications path for implementing the technique of the present invention is shown. The adaptive processor **200** includes an adaptive processor chip **202** incorporates a large number of functional units ("FU") **204** interconnected by reconfigurable routing resources. The adaptive processor chip **202** is coupled to a memory element **206** as well as an interconnect **208** and a number of additional adaptive processor chips **210**.

As shown, each adaptive processor chip **202** can contain thousands of functional units **204** dedicated to the particular problem at hand. Interconnect between these functional units is created by reconfigurable routing resources inside each chip **202**. As a result, the functional units **204** can share or exchange data at much higher data rates and lower latencies than a standard microprocessor **104** (FIG. **1**). In addition, the adaptive processor chips **202** can connect directly to the interprocessor interconnect **208** and do not require the data to be passed through multiple chips in a chipset in order to communicate. This is because the adaptive processor can implement whatever kind of interface is needed to accomplish this connection.

With reference additionally now to FIG. **3**A, a graph of the actual performance improvement versus the number of processors utilized in a conventional multi-processor computing system **100** (FIG. **1**) is shown. In this figure, the deviation from perfect scalability of a particular application is illustrated for such a system.

With reference additionally now to FIG. **3**B, a corresponding graph of the actual performance improvement versus the number of processors utilized in an adaptive processor-based computing system **200** (FIG. **2**) is shown. In this figure, the performance improvement provided with an adaptive processor-based computing system **200** over that of a conventional multi-processor computing system **100** is illustrated.

With reference additionally now to FIG. **4**A, a simplified logic flowchart is provided illustrating a conventional sequential processing operation **400** in which nested Loops A (first loop **402**) and B (second loop **404**) are alternately active on different phases of the process.

As shown, the standard implementation of applications that have a set of nested loops **402**,**404** is to complete the processing of the first loop **402** before proceeding to the second loop **404**. The problem inherent in this approach, particularly when utilized in conjunction with field programmable gate arrays ("FPGAs") is that all of the logic that has been instantiated is not being completely utilized.

With reference additionally now to FIG. **4**B, a comparative, simplified logic flowchart is shown illustrating a multi-dimensional process **410** in accordance with the technique of the present invention. The multi-dimensional process **410** is effectuated such that multiple dimensions of data are processed by both Loops A (first loop **412**) and B (second loop **414**) such that the computing system logic is operative on every clock cycle.

In contrast to the sequential processing operation **400** (FIG. **4**A) the solution to the problem of most effectively utilizing available resources is to have an application evaluate a problem in a data flow sense. That is, it will "pass" a subsequent dimension of a given problem through the first loop **412** of logic concurrently with the previous dimension of data being processed through the second loop **414**. In practice, a "dimension" of data can be: multiple vectors of a problem, multiple planes of a problem, multiple time steps in a problem and so forth.

With reference additionally now to FIG. **5**A, a general process for performing a representative multi-dimensional pipeline operation is shown in the form of a seismic migration imaging function **500**. The process **500** can be adapted to utilize the parallelism available in the utilization of the adaptive processing techniques of the present invention in the form of a multi-adaptive processor (MAP™, a trademark of SRC Computers, Inc., assignee of the present invention) STEP**3***d* routine **502**. The MAP STEP**3***d* routine **502** is operation to utilize velocity data **504**, source data **506** and receiver data **508** to produce a resultant image **510** as will be more fully described hereinafter.

With reference additionally now to FIG. **5**B, the MAP STEP**3***d* routine **502** of the preceding figure is shown in the various computational phases of: MAPTRI_x **520**, MAPTRI_y **522**, MAPTRI_d+ **524** and MAPTRI_d− **526**.

With reference additionally now to FIG. **6**A, a simplified logic flowchart for a particular seismic migration imaging application **600** is shown. The seismic migration imaging application **600** is illustrative of the parallelism provided in the use of an adaptive processor-based computing system **200** such as that shown in FIG. **2**. The representative application **600** demonstrates a nested loop parallelism in the tri-diagonal solver and the same logic can be implemented for the multiple tri-diagonal solvers in the x, y, d+ and d− directions. The computational phases of: MAPTRI_x **520**, MAPTRI_y **522**, MAPTRI_d+ **524** and MAPTRI_d− **526** are again illustrated.

With reference additionally now to FIG. **6**B, a computational process **610** is shown which may be employed by a microprocessor ("mP") in the execution of the seismic imaging application **600** of the preceding figure. The process **610** includes the step **612** of reading the source field [S(Z_0)] and receiver field [R(Z_0)] as well as the velocity field [V(Z_0)] at step **614**. At step **616** values are computed for S(Z_{nz}), R(Z_{nz}) which step is followed by the phases MAPTRI_x **520** and MAPTRI_y **522**. At step **618**, the image of $Z_{1/2}$ is computed. This is followed by the phases MAPTRI_d+ **524** and MAPTRI_d− **526** to produce the resultant image Z at step **620**. The

process **610** loops over the depth slices as indicated by reference number **622** and loops over the shots as indicated by reference number **624**.

With reference additionally now to FIG. **6**C, the first step in a computational process **650** in accordance with the technique of the present invention is shown in which a first shot (S**1**) is started. The process **650** may be employed by an adaptive processor (e.g. a MAP™ adaptive processor) as disclosed herein in the execution of the seismic imaging application **600** of FIG. **6**A. As indicated by the shaded block, the phase MAPTRI_x **520** is active.

With reference additionally now to FIG. **6**D, the second step in the computational process **650** is shown at a point at which a second shot (S**2**) is started. Again, as indicated by the shaded blocks, the phase MAPTRI_x **520** is active for S**2**, the phase MAPTRI_y **522** is active for S**1** and image $Z_{1/2}$ has been produced at step **618**. As shown, adaptive processors in accordance with the disclosure of the present invention support computation pipelining in multiple dimensions and the parallelism in Z and shots is shown at step **612**.

With reference additionally now to FIG. **6**E, the third step in the computational process **650** is shown in which the operation on the first and second shots is continued through compute. As indicated by the shaded blocks, the phase MAPTRI_d+ **524** is active for S**1**, the phase MAPTRI_y **522** is active for S**2** and image $Z_{1/2}$ has been produced at step **618**.

With reference additionally now to FIG. **6**F, the fourth step in the computational process **650** is shown illustrating the subsequent operation on shots S**1** and S**2**. The phase MAPTRI_d+ **524** is active for S**2**, the phase MAPTRI_d− **526** is active for S**1** and image Z has been produced at step **620**.

With reference additionally now to FIG. **6**G, the fifth step in the computational process **650** is shown as followed by the continued downward propagation of shots S**1** and S**2** over all of the depth slices. The phase MAPTRI_x **520** is active for S**1**, the phase MAPTRI_d− **526** is active for S**2** and image Z has been produced at step **620**.

With reference additionally now to FIG. **7**A, a process **700** for performing a representative systolic wavefront operation in the form of a reservoir simulation function is shown which utilizes the parallelism available in the adaptive processing techniques of the present invention. The process **700** includes a "k" loop **702**, "j" loop **704** and "i" loop **706** as shown.

With reference additionally now to FIG. **7**B, the general computation of fluid flow properties in the reservoir simulation process **700** of the preceding figure are illustrated as values are communicated between a group of neighboring cells **710**. The group of neighboring cells **710** comprises, in the simplified illustration shown, first, second and third walls of cells **712**, **714** and **716** respectively. Each of the walls of cells includes a corresponding number of first, second, third and fourth rows **718**, **720**, **722** and **724** respectively.

As shown, the computation of fluid flow properties are communicated to neighboring cells **710** and, importantly, this computation can be scheduled to eliminate the need for data storage. In accordance with the technique of the present invention, a set of cells can reside in an adaptive processor and the pipeline of computation can extend across multiple adaptive processors. Communication overhead between multiple adaptive processors may be advantageously minimized through the use of MAP™ adaptive processor chain ports as disclosed in U.S. Pat. No. 6,339,819 issued on Jan. 15, 2002 for: "Multiprocessor With Each Processor Element Accessing Operands in Loaded Input Buffer and Forwarding Results to FIFO Output Buffer", assigned to SRC Computers, Inc., assignee of the present invention, the disclosure of which is herein specifically incorporated by this reference.

With reference additionally now to FIG. **7**C, the creation of a systolic wall **712** of computation at Time Set **1** is shown. The systolic wall **712** has been started for a vertical wall of cells and communication of values between adjacent rows **718** through **724** in the vertical wall can occur without storing values to memory.

With reference additionally now to FIG. **7**D, a follow on illustration of the creation of a systolic wall **712** of computation at Time Set **1** and a second systolic wall **714** at Time Set **2** is shown. In operation, a second vertical wall of cells is started after the computation for cells in the corresponding row of the first wall has been completed. Thus, for example, at time to, the first row **718** of systolic wall **712** is completed and the results passed to the first row **718** of the second systolic wall **714**. At time t_1, the second row **720** of the first systolic wall **712** and the first row **718** of the second systolic wall **714** are computed. Thereafter, at time t_2, the third row **722** of the first systolic wall **712** and the second row **720** of the second systolic wall **714** are computed. The process continues in this manner for all rows and all walls.

With reference additionally now to FIG. **8**A, yet another process **800** for performing a representative systolic wavefront operation is shown. The process **800** is in the form of the systolic processing of bioinformatics and also utilizes the parallelism available in the adaptive processing techniques of the present invention. As shown, systolic processing in the process **800** can pass previously computed data down within a column (e.g. one of columns **802**, **804** and **806**) as to subsequent columns as well (e.g. from column **802** to **804**; from column **804** to **806** etc.) The computational advantage provided is the processing of the second column **804** can begin after only a few clock cycles following the start of the processing of the first column **802** to compute the first "match" state.

With reference additionally now to FIG. **8**B, a systolic wavefront processing operation **810** is shown. The processing operation **810**, comprising "i" loop **812** and "k" loop **814** now further incorporates a speculative processing strategy based upon an evaluation of the rate of change of XB.

A straightforward systolic processing operation could be used for performing the operation **810** but for the problem inherent in the computation of XB as its value XB[i] **816** can not be known until the completion of the entire "k" loop **814**. After evaluating the rate of change of XB, it was determined that a speculative processing strategy could be used for the problem. A normal systolic form is set up and the value of XB is held constant for the set of columns computed in the systolic set. At the bottom of each column, the value of XB[i] **816** is then computed.

With reference additionally now to FIG. **8**C, a further illustration of the systolic wavefront processing operation **810** incorporating speculative processing of the preceding figure is shown. The speculative processing includes "j" columns $\mathbf{818}_0$ through $\mathbf{818}_j$ as shown. Each of the columns **818** assumes that XB[i+j] has a constant value. A test is conducted at the bottom of each of the columns **818** to determine with the XB value changes as indicated at steps $\mathbf{820}_1$ through $\mathbf{820}_j$. If the value of XB changes at the i+n column, the process is then restarted at that column **818**. Since the rate of change of XB is relatively slow, the "cost" of the compute operation can be greatly reduced.

With reference additionally now to FIG. **9**A, another process **900** for performing a representative systolic wavefront operation is shown in the form of structure codes calculating polynomials at grid intersections **902**. The process **900** advantageously utilizes the parallelism available in the adaptive processing techniques of the present invention.

With reference additionally now to FIGS. **9**B and **9**C, the computation start for a vertical wall **910** of grid points at Time Set **1** is shown for a polynomial evaluation performed on grid intersections **902** (FIG. **9**A) wherein calculations between rows **912**, **914**, **016** and **918** are done in a stochastic fashion using values from a previous row. As shown, a polynomial evaluation is performed on the grid intersections **902** such that a second wall $\mathbf{910}_1$ is started after the cells in the corresponding row of the first wall $\mathbf{910}_0$ have been completed.

As can be determined from the foregoing, the multi-adaptive processing systems and techniques for enhancing parallelism and performance of computational functions disclosed herein can be employed in a myriad of applications including multi-dimensional pipeline computations for seismic applications, search algorithms, information security, chemical and biological applications, filtering and the like as well as for systolic wavefront computations for fluid flow and structures analysis, bioinformatics etc. Some applications may also employ both the multi-dimensional pipeline and systolic wavefront methodologies.

Following are representative applications of the techniques for adaptive processor based computation disclosed herein:

Imaging

Seismic: These applications, typically used in the oil and gas exploration industries, process echo data to produce detailed analysis of subsurface features. The applications use data collected at numerous points and consisting of many repeated parameters. Due to this, these programs are ideal candidates to take advantage of parallel computing. In addition, because the results of the computation on one data point are used in the computation of the next, these programs will particularly benefit from the tight parallelism that can be found in the use of adaptive or reconfigurable processors.

Synthetic Aperture Radar ("SAR"): These applications are typically used in geographical imaging. The applications use data collected in swaths. Processing consists of repeated operations on data that has been sectioned in cells. These programs are also ideal candidates to take advantage of parallel computing and in particular to benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

JPEG Image compression: These applications partition an image into numerous blocks. These blocks then have a set of operations performed on them. The operations can be parallelized across numerous blocks. The combination of the set of operations and the parallelism will particularly benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

MPEG Image compression: These applications partition a frame into numerous blocks. These blocks then have a set of operations performed on them. The operations can be parallelized across numerous blocks. In addition, there are numerous operations that are performed on adjacent frames. The combination of the set of operations and the parallelism will particularly benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Fluid Flow

Reservoir Simulation: These applications, also typically used in the oil and gas production industries, process fluid flow data in the oil and gas subsurface reservoirs to produce extraction models. The application will define a three dimensional ("3d") set of cells that contain the oil and gas reservoir. These programs are ideal candidates to take advantage of parallel or adaptive computing because there are repeated operations on each cell. In addition, information computed for each cell is then passed to neighboring cells. These programs will particularly benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Weather prediction: Such an application will partition the forecast area into logical grid cells. The computational algorithms will then perform calculations that have polynomials that have nodes associated with the grid cells. These programs are ideal candidates to take advantage of adaptive or parallel computing because there are repeated operations on each cell associated with the set of times computed in the forecast.

Automotive: These applications investigate the aerodynamics of automobile or other aerodynamic structures. The application generally divides the space surrounding the automobile structure into logical cells that are associated with nodes in computational polynomials. These programs are ideal candidates to take advantage of adaptive or parallel computing because there are repeated operations on each cell associated with the set of wind velocities computed in the forecast. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Aerospace: These applications investigate the aerodynamics of aerospace/airplane structures. The application divides the space surrounding the aerospace/airplane structure into logical cells that are associated with nodes in computational polynomials. These programs are ideal candidates to take advantage of parallel computing because there are repeated operations on each cell associated with the set of wind velocities computed in the forecast. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Plastic Injection Molding: These applications investigate the molding parameters of injecting liquid plastic into molds. The application divides the space inside the mold into logical cells that are also associated with nodes in computational polynomials. These programs are ideal candidates to take advantage of parallel computing because there are repeated operations on each cell associated with the set of injection parameters. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Structures

Crash Analysis: These applications are typically used in the automotive or aviation industry. The application will partition the entire automobile into components. These components are then subdivided into cells. The application will analyze the effect of a collision on the structure of the automobile. These programs are ideal candidates for parallel computing because there are repeated operations on each cell and they receive computed information from their neighboring cells. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Structural Analysis: These applications investigate the properties of structural integrity. The application divides the structure into logical cells that are associated with nodes in computational polynomials. These programs are ideal candidates to take advantage of parallel computing because there are repeated operations on each cell associated with load and stress. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Search Algorithms

Image searches: These applications are typically used in the security industry for fingerprint matching, facial recognition and the like. The application seeks matches in either a collection of subsets of the total image or the total image itself. The process compares pixels of the model to pixels of a record from an image database. These programs are ideal

candidates for parallel computing because of the correlation of comparison results that exist for each pixel in the subsets or entire image. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Data mining: These applications are typically used in commercial market spaces. The application seeks matches in a set of search information (e.g. character strings) in each record in a database. The application then produces a match correlation for all data records. A match correlation is produced from the comparison results for each set of search information with all characters in a database record. These programs are ideal candidates for parallel computing because of the repeated comparison operations that exist all character comparisons of the set of search information with each character in the database record. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Finance

Financial modeling: The application creates numerous strategies for each decision step in the modeling process. The results of a computational step are feed into another set of strategies for subsequence modeling steps. These programs are ideal candidates to take advantage of parallel computing because there are repeated operations on each strategy within a modeling step. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Information Security

Encryption/Decryption: The application applies an algorithm that converts the original data into an encrypted, or "protected", form. The process is applied to each set of N bits in the original data. Decryption reverses the process to deliver the original data. These programs are ideal candidates for parallel computing because there are repeated operations on each N bits of data. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Chemistry/Biology

Genetic pattern matching: These applications are typically used in the bioinformatics industry. The application looks for matches of a particular genetic sequence (or model) to a database of genetic records. The application compares each character in the model to the characters in genetic record. These programs are ideal candidates for parallel computing because of the repeated comparison operations that exist for all character comparisons of the model with each character in the genetic record. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

Protein Folding: These applications are typically used by pharmaceutical companies. The application investigates the dynamics of the deformation of the protein structure. The application uses a set of equations which are recomputed at various "time" intervals to model the protein folding. These programs are ideal candidates for parallel computing because of the repeated computations on a large set of time intervals in the modeling sequence. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors

Organic structure interaction: These applications are typically used by chemical and drug companies. The application investigates the dynamics of organic structures as they are interacting. The application uses a set of equations which are recomputed at various "time" intervals to model how the organic structure interact. These programs are ideal candidates for parallel computing because of the repeated computations on a large set of time intervals in the modeling sequence. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors

Signals

Filtering: Applications often utilize filtering techniques to "clean-up" a recorded data sequence. This technique is utilized in a wide variety of industries. The application generally applies a set of filter coefficients to each data point in the recorded sequence. These programs are ideal candidates for parallel computing because of the repeated computations to all data points in the sequence and all sequences. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors.

While there have been described above the principles of the present invention in conjunction with specific, exemplary applications for the use of adaptive processor-based systems in the implementation of multi-dimensional pipeline and systolic wavefront computations, it is to be clearly understood that the foregoing descriptions are made only by way of example and not as a limitation to the scope of the invention. Particularly, it is recognized that the teachings of the foregoing disclosure will suggest other modifications to those persons skilled in the relevant art. Such modifications may involve other features which are already known per se and which may be used instead of or in addition to features already described herein. Although claims have been formulated in this application to particular combinations of features, it should be understood that the scope of the disclosure herein also includes any novel feature or any novel combination of features disclosed either explicitly or implicitly or any generalization or modification thereof which would be apparent to persons skilled in the relevant art, whether or not such relates to the same invention as presently claimed in any claim and whether or not it mitigates any or all of the same technical problems as confronted by the present invention. The applicants hereby reserve the right to formulate new claims to such features and/or combinations of such features during the prosecution of the present application or of any further application derived therefrom.

What is claimed is:

1. A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor, the reconfigurable processor comprising a plurality of functional units, said method comprising:

transforming an algorithm into a data driven calculation that is implemented by said reconfigurable computing system at the at least one reconfigurable processor;

forming at least two of said functional units at the at least one reconfigurable processor to perform said calculation wherein only functional units needed to solve the calculation are formed and wherein each formed functional unit at the at least one reconfigurable processor interconnects with each other formed functional unit at the at least one reconfigurable processor based on reconfigurable routing resources within the at least one reconfigurable processor as established at formation, and wherein lines of code of said calculation are formed as clusters of functional units within the at least one reconfigurable processor;

utilizing a first of said formed functional units to operate upon a subsequent data dimension of said calculation forming a first computational loop; and

substantially concurrently utilizing a second of said formed functional units to operate upon a previous data dimension of said calculation generating a second computational loop wherein said implementation of said calculation enables said first computational loop and said second computational loop execute concurrently and pass computed data seamlessly between said computational loops.

2. The method of claim **1** wherein said subsequent and previous data dimensions of said calculation comprise multiple vectors in said calculation.

3. The method of claim **1** wherein said subsequent and previous data dimensions of said calculation comprise multiple planes in said calculation.

4. The method of claim **1** wherein said subsequent and previous data dimensions of said calculation comprise multiple time steps in said calculation.

5. The method of claim **1** wherein said subsequent an previous data dimensions of said calculation comprise multiple grid points in said calculation.

6. The method of claim **1** wherein said calculation comprises a seismic imaging calculation.

7. The method of claim **1** wherein said calculation comprises a synthetic aperture radar imaging calculation.

8. The method of claim **1** wherein said calculation comprises a JPEG image compression calculation.

9. The method of claim **1** wherein said calculation comprises an MPEG image compression calculation.

10. The method of claim **1** wherein said calculation comprises a fluid flow calculation for a reservoir simulation.

11. The method of claim **1** wherein said calculation comprises a fluid flow calculation for weather prediction.

12. The method of claim **1** wherein said calculation comprises a fluid flow calculation for automotive applications.

13. The method of claim **1** wherein said calculation comprises a fluid flow calculation for aerospace applications.

14. The method of claim **1** wherein said calculation comprises a fluid flow calculation for an injection molding application.

15. The method of claim **1** wherein instantiating includes establishing a stream communication connection between functional units.

16. The method of claim **1** wherein said calculation is comprises a structures calculation for structural analysis.

17. The method of claim **1** wherein said calculation comprises a search algorithm for an image search.

18. The method of claim **1** wherein said calculation comprises a search algorithm for data mining.

19. The method of claim **1** wherein said calculation comprises a financial modeling application.

20. The method of claim **1** wherein said calculation comprises an encryption algorithm.

21. The method of claim **1** wherein said calculation comprises a genetic pattern matching function.

22. The method of claim **1** wherein said calculation comprises a protein folding function.

23. The method of claim **1** wherein said calculation comprises an organic structure interaction function.

24. The method of claim **1** wherein said calculation comprises a signal filtering application.

25. A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor comprising a plurality of functional units, said method comprising:

transforming an algorithm into a data driven calculation that is implemented by said reconfigurable computing system at the at least one reconfigurable processor wherein linked lines of code of said calculation are fashioned as walls of functional units within the at least one reconfigurable processor;

defining a first wall comprising rows of cells forming a subset of said plurality of functional units;

computing at the at least one reconfigurable processor a value at each of said cells in at least a first row of said first wall substantially concurrently;

communicating said values between cells in said first row of said cells to produce updated values, wherein communicating said values is based on reconfigurable routing resources within the at least one reconfigurable processor;

communicating said updated values substantially concurrently to a second row of said first wall, wherein communicating said updated values is based on reconfigurable routing resources within the at least one reconfigurable processor; and

communicating said updated values substantially concurrently to a first row of a second wall of rows of cells in said subset of said plurality of functional units, wherein communicating said updated values is based on reconfigurable routing resources within the at least one reconfigurable processor and wherein said first wall of rows of cells and said second wall of rows of cells execute substantially concurrently and pass computed data seamlessly between said walls.

26. The method of claim **25** wherein said values correspond to vectors in a computation.

27. The method of claim **25** wherein said values correspond to planes in a computation.

28. The method of claim **25** wherein said values correspond to time steps in a computation.

29. The method of claim **25** wherein said values correspond to grid points in a computation.

30. The method of claim **25** wherein said step of communicating said updated values to a second row of said first wall is carried out without storing said updated values in an extrinsic memory.

31. The method of claim **25** wherein said values correspond to a seismic imaging calculation.

32. The method of claim **25** wherein said values correspond to a synthetic aperture radar imaging calculation.

33. The method of claim **25** wherein said values correspond to a JPEG image compression calculation.

34. The method of claim **25** wherein said values correspond to an MPEG image compression calculation.

35. The method of claim **25** wherein said values correspond to a fluid flow calculation for a reservoir simulation.

36. The method of claim **25** wherein said values correspond to a fluid flow calculation for weather prediction.

37. The method of claim **25** wherein said values correspond to a fluid flow calculation for automotive applications.

38. The method of claim **25** wherein said values correspond to a fluid flow calculation for aerospace applications.

39. The method of claim **25** wherein said values correspond to a fluid flow calculation for an injection molding application.

40. The method of claim **25** wherein defining includes establishing a stream communication connection between functional units and wherein only functional units needed to solve the calculations are instantiated.

41. The method of claim **25** wherein said values correspond to a structures calculation for structural analysis.

42. The method of claim **25** wherein said values correspond to a search algorithm for an image search.

43. The method of claim 25 wherein said values correspond to a search algorithm for data mining.

44. The method of claim 25 wherein said values correspond to a financial modeling application.

45. The method of claim 25 wherein said values correspond to an encryption algorithm.

46. The method of claim 25 wherein said values correspond to a genetic pattern matching function.

47. The method of claim 25 wherein said values correspond to a protein folding function.

48. The method of claim 25 wherein said values correspond to an organic structure interaction function.

49. The method of claim 25 wherein said values correspond to a signal filtering application.

50. The method of claim 25 wherein said reconfigurable computing system comprises at least one microprocessor.

51. A method for data processing in a reconfigurable computing system, the reconfigurable computer system comprising at least one reconfigurable processor comprising a plurality of functional units, said method comprising:

transforming an algorithm into a calculation implemented by said reconfigurable computing system at the at least one reconfigurable processor and driven by data propagation wherein lines of code of said calculation are linked based on said data propagation and fashioned as subsets of said plurality of functional units within the at least one reconfigurable processor forming columns of said calculation;

performing said calculation at the at least one reconfigurable processor by said subsets of said plurality of functional units to produce computed data;

exchanging said computed data between a first column of said calculation and a next column in said calculation, wherein said exchanging is based on reconfigurable routing resources within the at least one reconfigurable processor and wherein execution of said subsets of said plurality of function units occurs concurrently and said computed data is seamlessly passed between said first column of said calculation and said second column of said calculation;

evaluating a rate of change in at least one variable for each of said columns in said calculation;

continuing said calculation when said variable does not change for a particular column of said calculation; and

restarting said calculation at said column of said calculation where said variable does change.

52. The method of claim 51 wherein how many functional units comprise the subset and functional type of each functional unit in said subset is based on the calculation.

* * * * *

EXHIBIT H

US007149867B2

(12) **United States Patent**
Poznanovic et al.

(10) **Patent No.: US 7,149,867 B2**
(45) **Date of Patent: Dec. 12, 2006**

(54) **SYSTEM AND METHOD OF ENHANCING EFFICIENCY AND UTILIZATION OF MEMORY BANDWIDTH IN RECONFIGURABLE HARDWARE**

(75) Inventors: **Daniel Poznanovic**, Colorado Springs, CO (US); **David E. Caliga**, Colorado Springs, CO (US); **Jeffrey Hammes**, Colorado Springs, CO (US)

(73) Assignee: **SRC Computers, Inc.**, Colorado Springs, CO (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **10/869,200**

(22) Filed: **Jun. 16, 2004**

(65) **Prior Publication Data**

US 2004/0260884 A1 Dec. 23, 2004

Related U.S. Application Data

(60) Provisional application No. 60/479,339, filed on Jun. 18, 2003.

(51) **Int. Cl.**
G06F 12/00 (2006.01)

(52) **U.S. Cl.** **711/170**; 711/154

(58) **Field of Classification Search** 711/170–173; 712/15

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,076,152	A	6/2000	Huppenthal et al.	
6,243,791	B1 *	6/2001	Vondran, Jr.	711/120
6,247,110	B1	6/2001	Huppenthal et al.	
6,460,122	B1 *	10/2002	Otterness et al.	711/154
6,507,898	B1 *	1/2003	Gibson et al.	711/168
6,594,736	B1	6/2003	Parks	
6,714,041	B1 *	3/2004	Darling et al.	326/38
2003/0046492	A1 *	3/2003	Gschwind et al.	711/118
2003/0046530	A1 *	3/2003	Poznanovic	713/100
2003/0084244	A1 *	5/2003	Paulraj	711/118
2003/0088737	A1 *	5/2003	Burton	711/118
2003/0208658	A1 *	11/2003	Magoshi	711/122
2005/0044327	A1 *	2/2005	Howard et al.	711/147

OTHER PUBLICATIONS

"Summary: The Cache Read/Write Process," The PC Guide, 2001, www.pcguide.com/ref/mbsys/cache/func.htm.*

Chien et al., "Safe and Protected Execution for the Morph/AMRM Reconfigurable Processor," IEEE, 1999, pp. 1-13.*

IEEE 100: The Authoritative Dictionary of IEEE Standards Terms, Standards Information Network, 2000, pp. 874.*

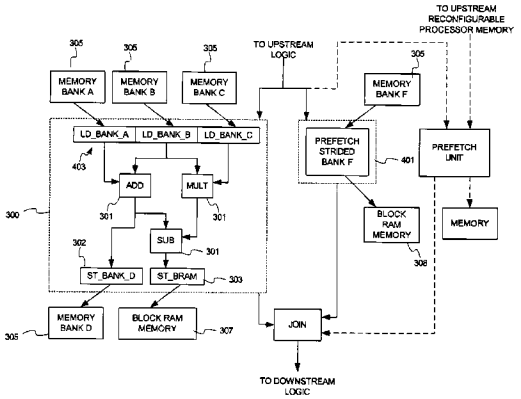
* cited by examiner

Primary Examiner—Gary Portka
Assistant Examiner—Shane M. Thomas
(74) *Attorney, Agent, or Firm*—William J. Kubida; Michael C. Martensen; Hogan & Hartson LLP

(57) **ABSTRACT**

A reconfigurable processor that includes a computational unit and a data prefetch unit coupled to the computational unit, where the data prefetch unit retrieves data from a memory and supplies the data to the computational unit through memory and a data access unit, and where the data prefetch unit, memory, and data access unit is configured by a program. Also, a reconfigurable hardware system that includes a common memory; and one or more reconfigurable processors coupled to the common memory, where at least one of the reconfigurable processors includes a data prefetch unit to read and write data between the unit and the common memory, and where the data prefetch unit is configured by a program executed on the system. In addition, a method of transferring data that includes transferring data between a memory and a data prefetch unit in a reconfigurable processor; and transferring the data between a computational unit and the data prefetch unit.

19 Claims, 12 Drawing Sheets



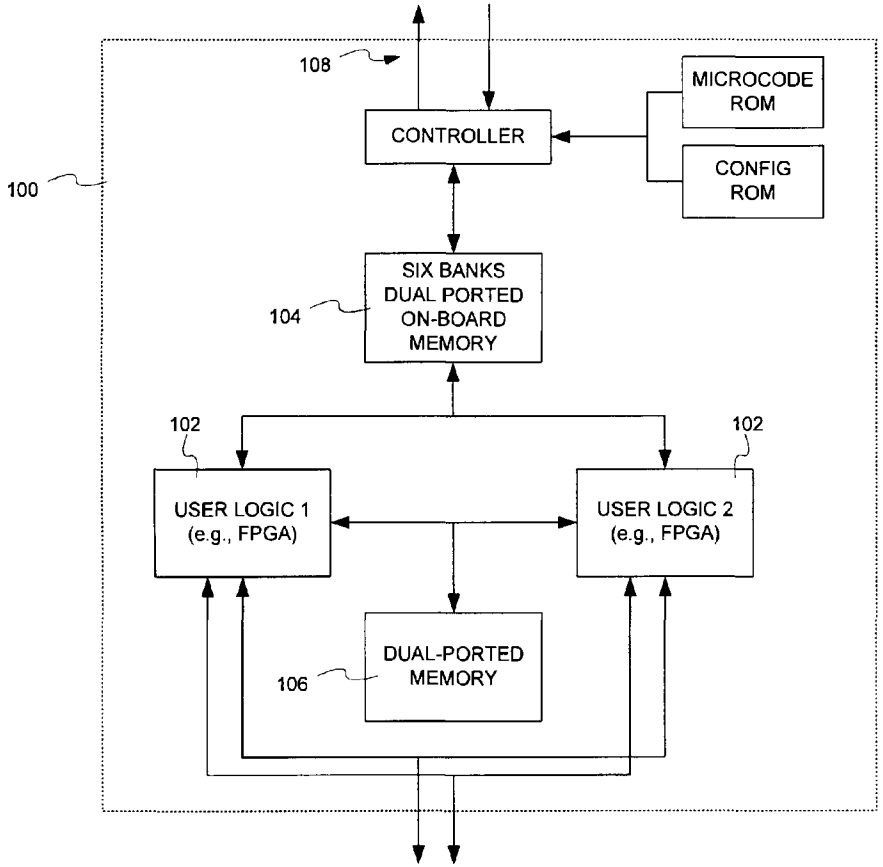

100
108
CONTROLLER
MICROCODE ROM
CONFIG ROM
SIX BANKS DUAL PORTED ON-BOARD MEMORY
104
102
102
USER LOGIC 1 (e.g., FPGA)
USER LOGIC 2 (e.g., FPGA)
DUAL-PORTED MEMORY
106

FIG. 1

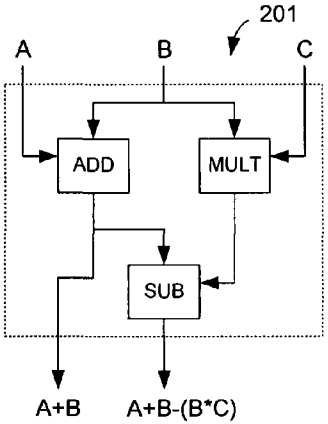

201
A
B
C
ADD
MULT
SUB
A+B
A+B-(B*C)

FIG. 2

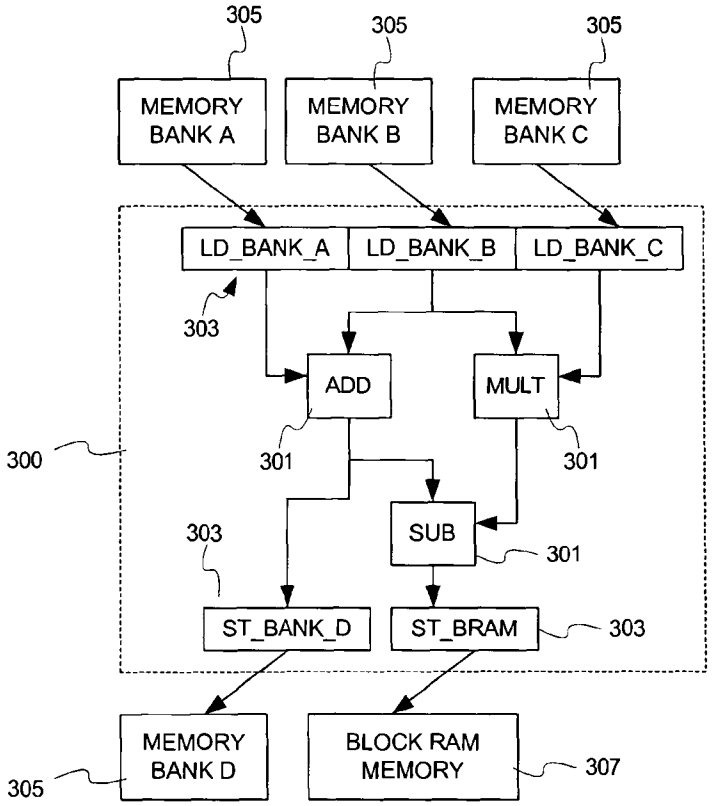

305
305
305
MEMORY BANK A
MEMORY BANK B
MEMORY BANK C
LD_BANK_A
LD_BANK_B
LD_BANK_C
303
ADD
MULT
300
301
301
303
SUB
301
ST_BANK_D
ST_BRAM
303
MEMORY BANK D
BLOCK RAM MEMORY
305
307

FIG. 3

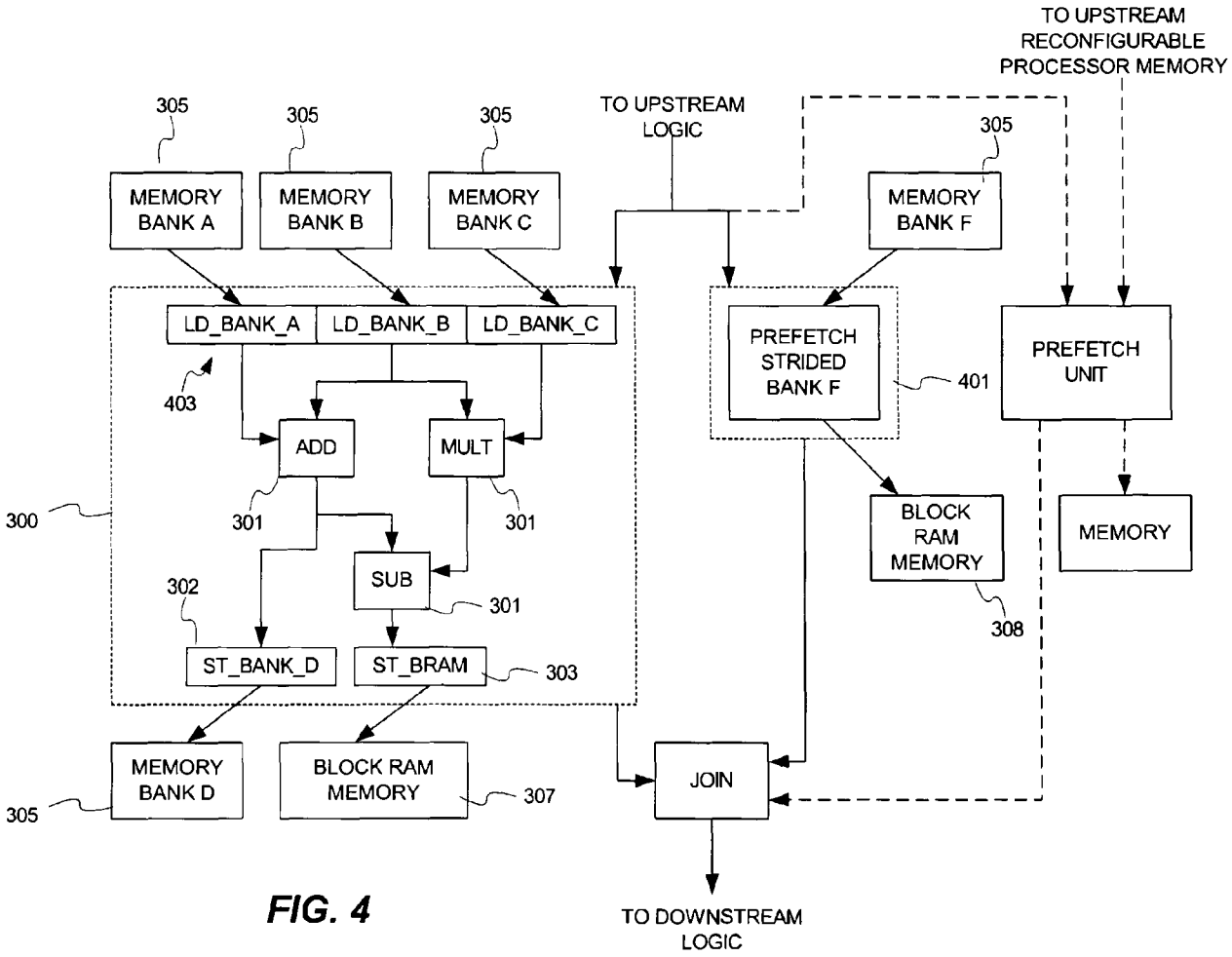

TO UPSTREAM RECONFIGURABLE PROCESSOR MEMORY
TO UPSTREAM LOGIC
305
MEMORY BANK A
MEMORY BANK B
MEMORY BANK C
MEMORY BANK F
LD_BANK_A
LD_BANK_B
LD_BANK_C
403
ADD
MULT
SUB
301
302
300
ST_BANK_D
ST_BRAM
303
PREFETCH STRIDED BANK F
401
PREFETCH UNIT
BLOCK RAM MEMORY
308
MEMORY
MEMORY BANK D
BLOCK RAM MEMORY
307
JOIN
TO DOWNSTREAM LOGIC

FIG. 4

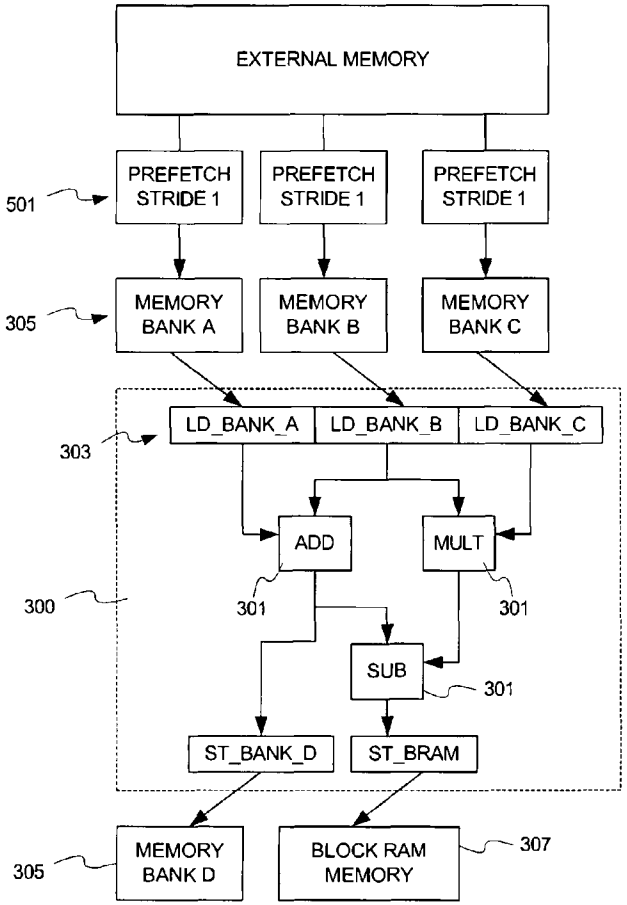

EXTERNAL MEMORY
PREFETCH STRIDE 1
PREFETCH STRIDE 1
PREFETCH STRIDE 1
501
MEMORY BANK A
MEMORY BANK B
MEMORY BANK C
305
LD_BANK_A
LD_BANK_B
LD_BANK_C
303
ADD
MULT
300
301
301
SUB
301
ST_BANK_D
ST_BRAM
305
MEMORY BANK D
BLOCK RAM MEMORY
307

FIG. 5

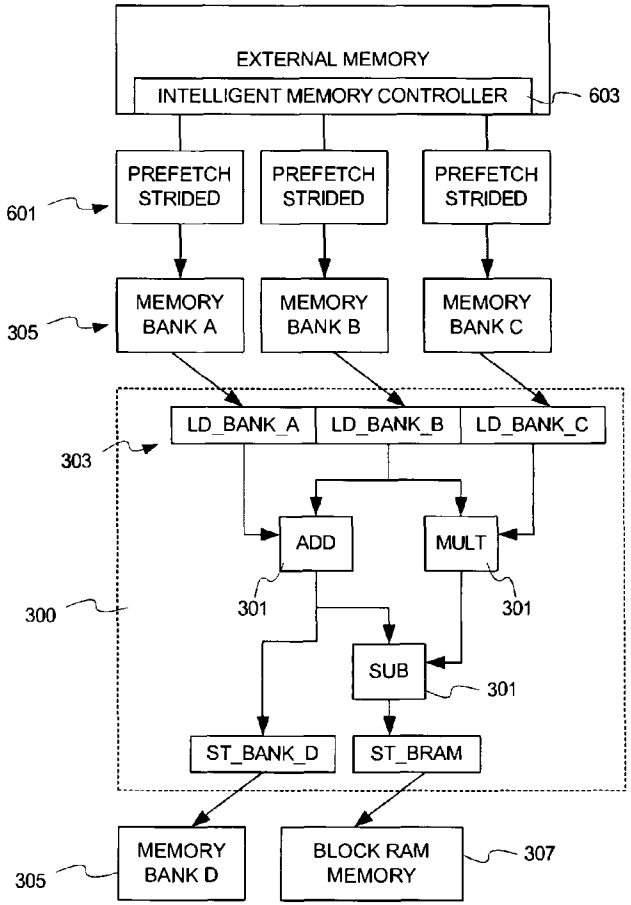

EXTERNAL MEMORY
INTELLIGENT MEMORY CONTROLLER
603
PREFETCH STRIDED
PREFETCH STRIDED
PREFETCH STRIDED
601
MEMORY BANK A
MEMORY BANK B
MEMORY BANK C
305
LD_BANK_A
LD_BANK_B
LD_BANK_C
303
ADD
MULT
300
301
301
SUB
301
ST_BANK_D
ST_BRAM
305
MEMORY BANK D
BLOCK RAM MEMORY
307

FIG. 6

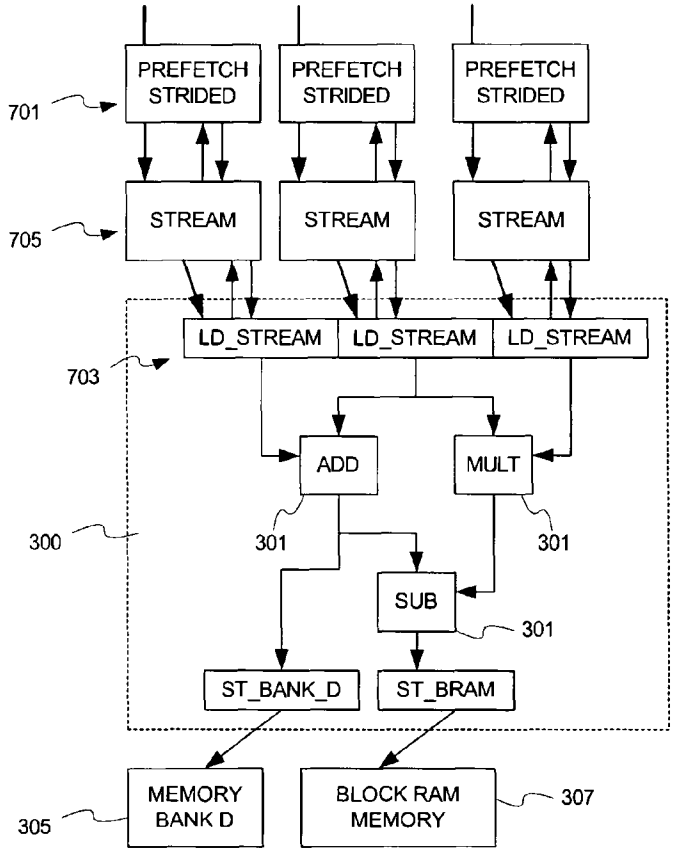

PREFETCH STRIDED
PREFETCH STRIDED
PREFETCH STRIDED
701
STREAM
STREAM
STREAM
705
LD_STREAM
LD_STREAM
LD_STREAM
703
ADD
MULT
300
301
301
SUB
301
ST_BANK_D
ST_BRAM
MEMORY BANK D
BLOCK RAM MEMORY
305
307

FIG. 7

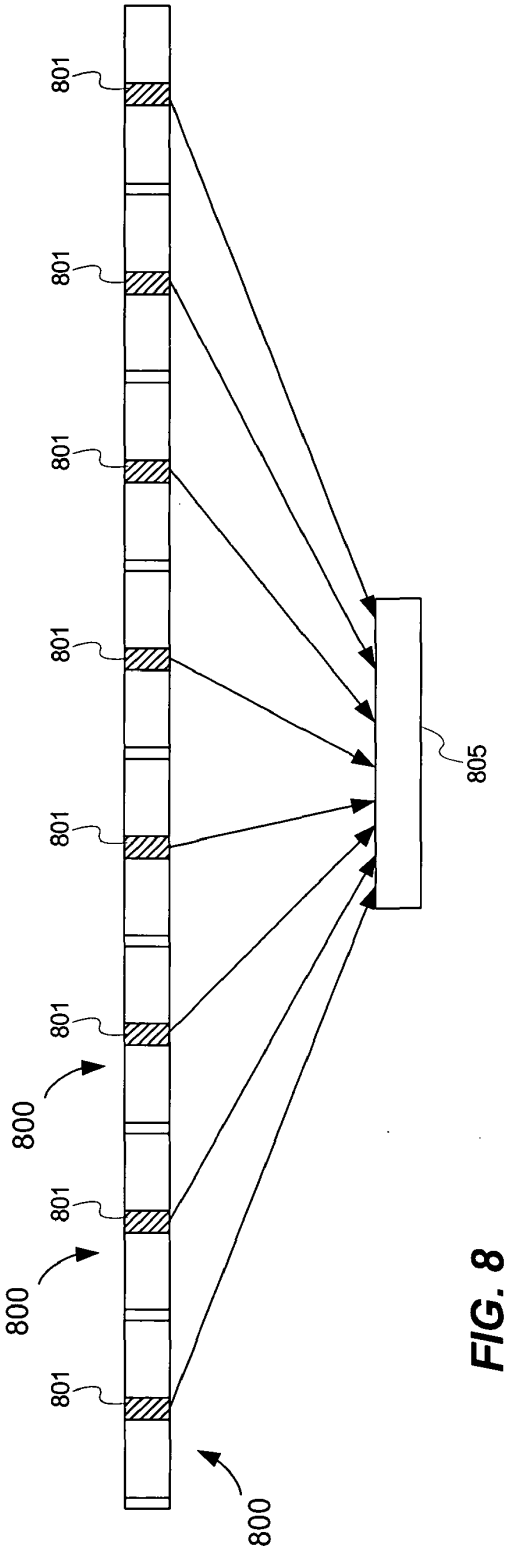


FIG. 8

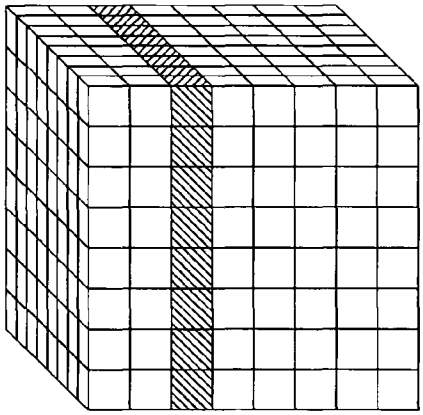


FIG. 9A

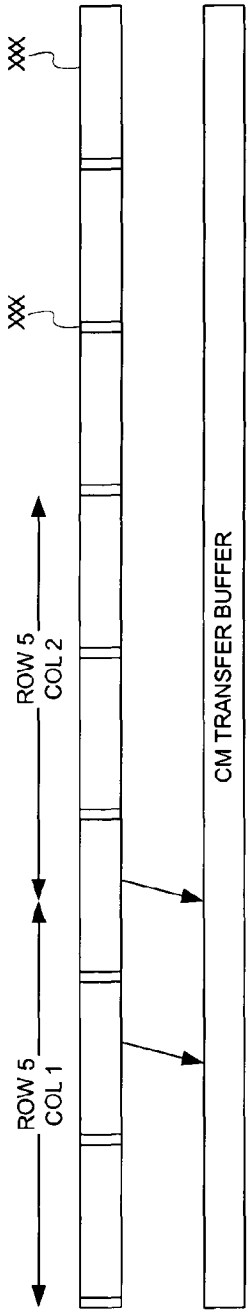


FIG. 9B

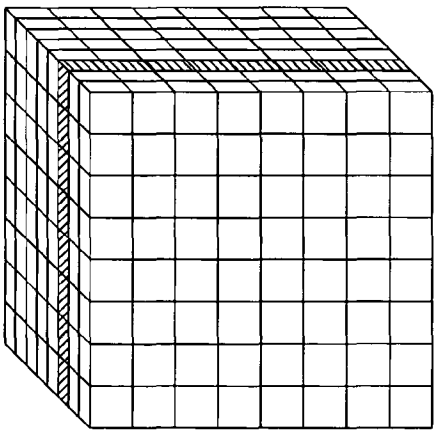


FIG. 10A

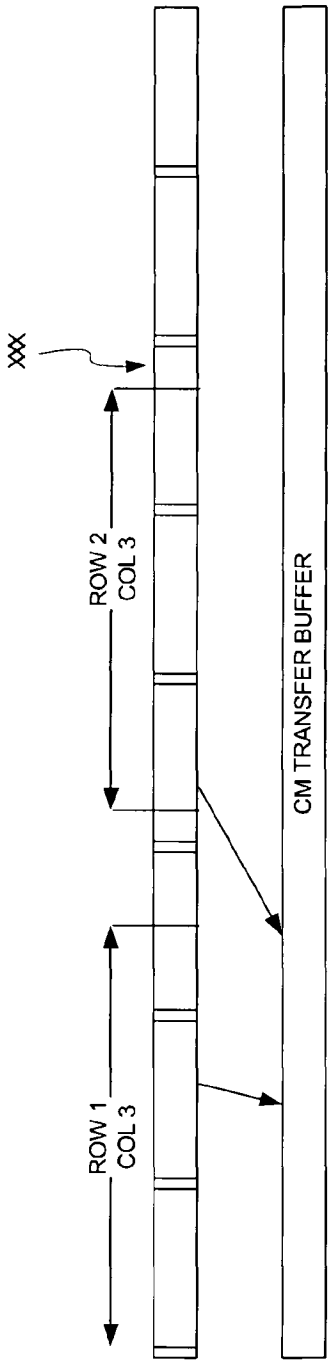


FIG. 10B

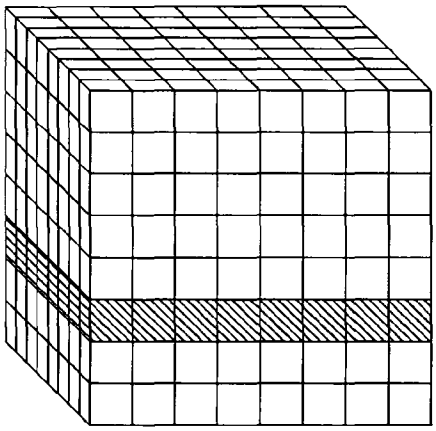


FIG. 11A

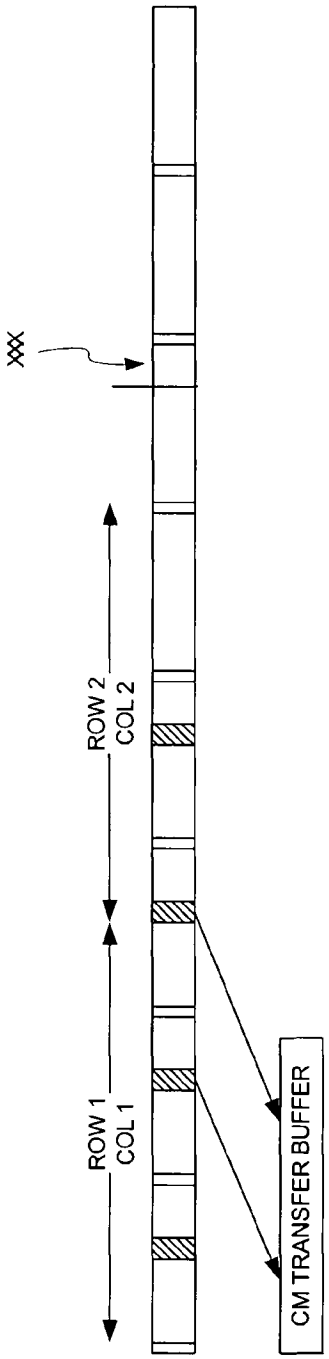


FIG. 11B

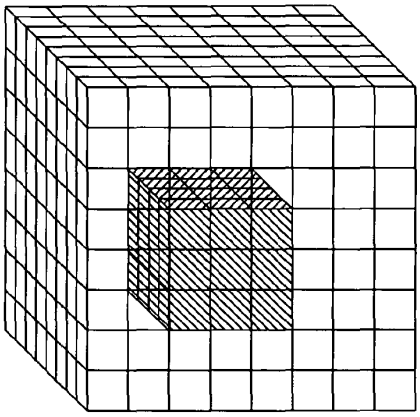


FIG. 12A

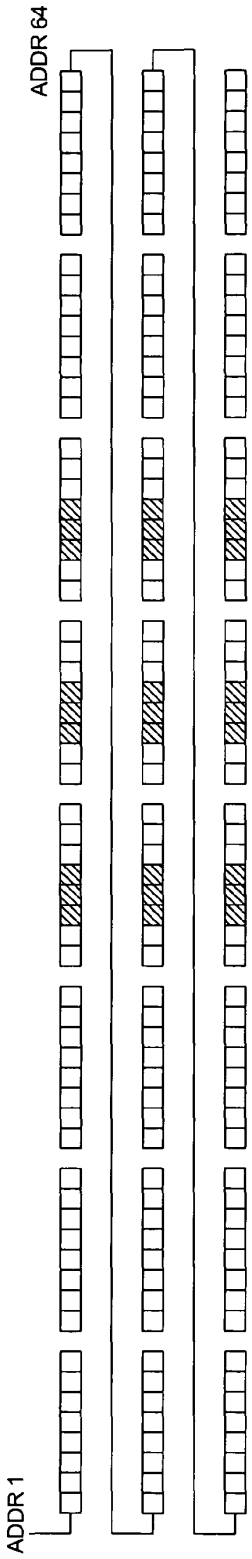


FIG. 12B

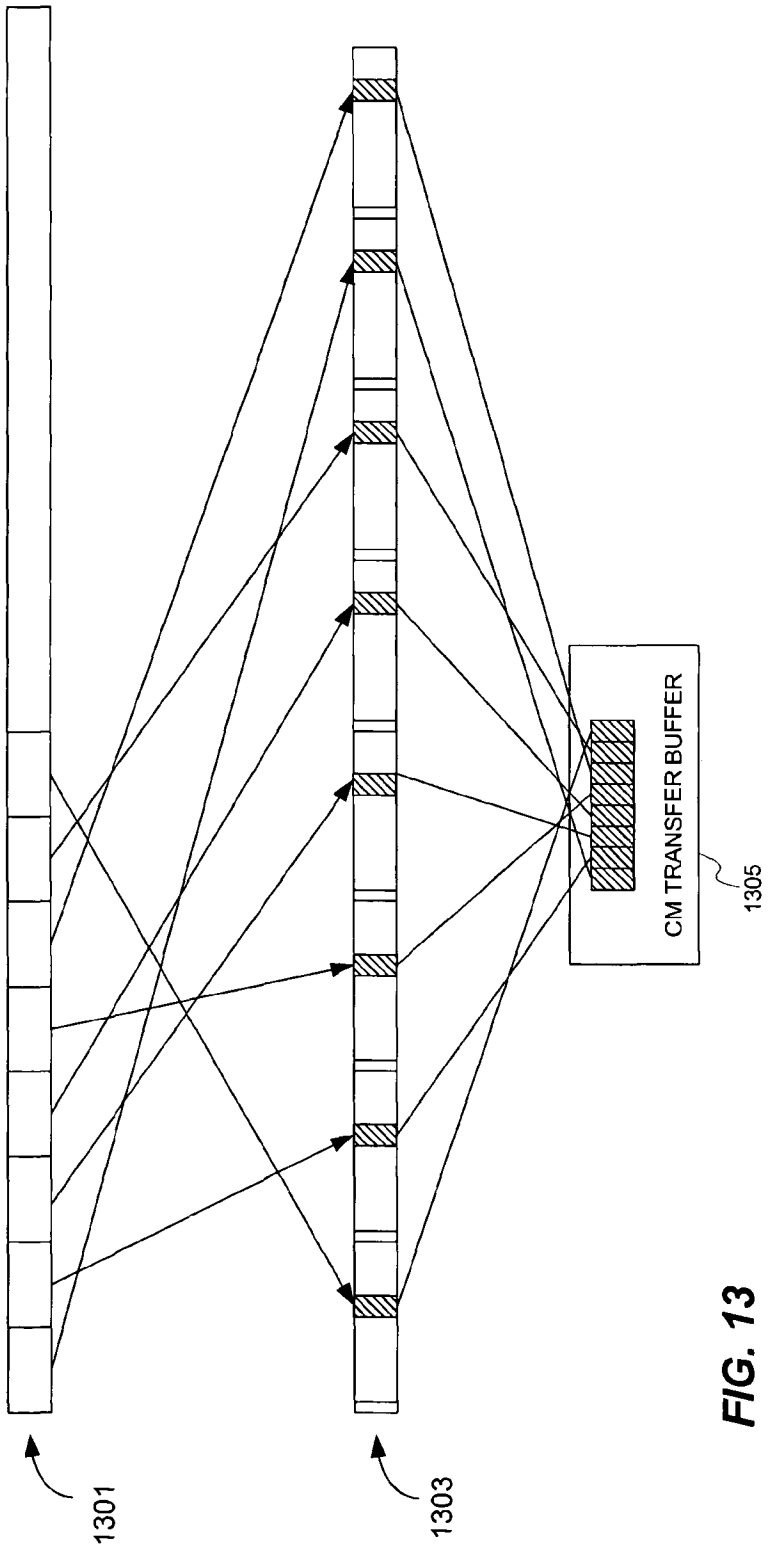


FIG. 13

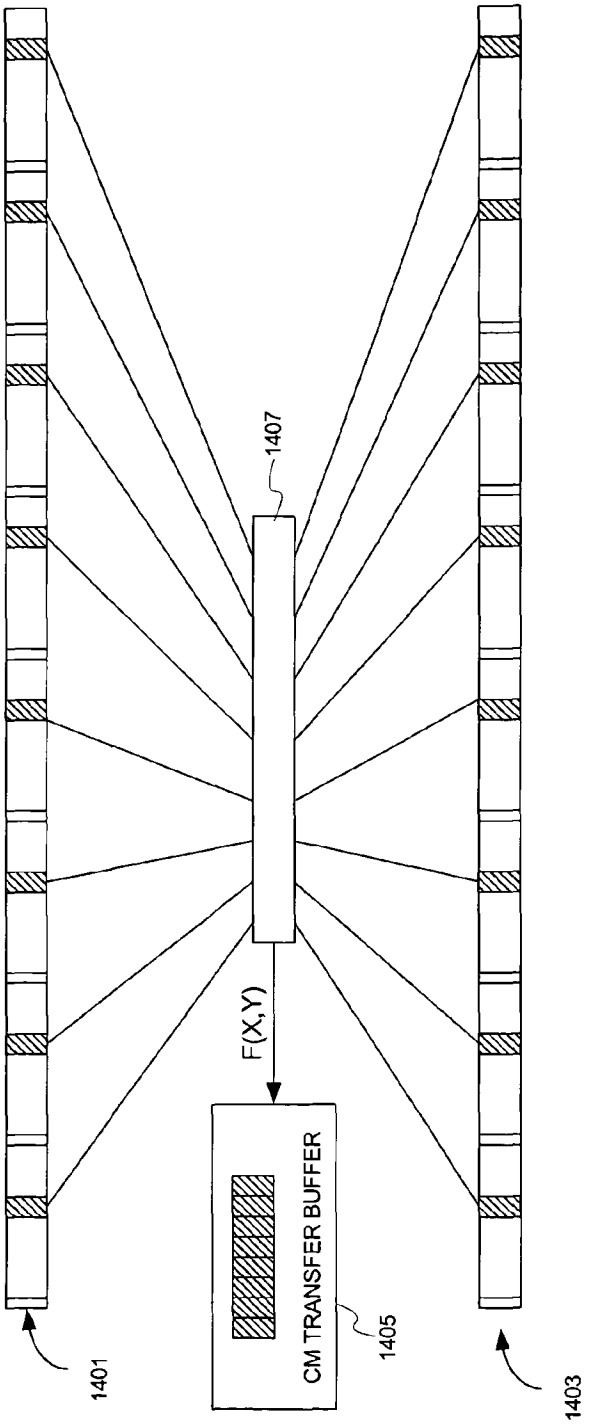


FIG. 14

SYSTEM AND METHOD OF ENHANCING EFFICIENCY AND UTILIZATION OF MEMORY BANDWIDTH IN RECONFIGURABLE HARDWARE

RELATED APPLICATIONS

The present invention claims the benefit of U.S. Provisional Patent application Ser. No. 60/479,339 filed on Jun. 18, 2003, which is incorporated herein by reference in its entirety.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates, in general, to enhancing the efficiency and utilization of memory bandwidth in reconfigurable hardware. More specifically, the invention relates to implementing explicit memory hierarchies in reconfigurable processors that make efficient use of off-board, on-board, on-chip storage and available algorithm locality. These explicit memory hierarchies avoid many of the tradeoffs and complexities found in the traditional memory hierarchies of microprocessors.

2. Relevant Background

Over the past 30 years, microprocessors have enjoyed annual performance gains averaging about 50% per year. Most of the gains can be attributed to higher processor clock speeds, more memory bandwidth and increasing utilization of instruction level parallelism (ILP) at execution time.

As microprocessors and other dense logic devices (DLDs) consume data at ever-increasing rates it becomes more of a challenge to design memory hierarchies that can keep up. Two measures of the gap between the microprocessor and memory hierarchy are bandwidth efficiency and bandwidth utilization. Bandwidth efficiency refers to the ability to exploit available locality in a program or algorithm. In the ideal situation, when there is maximum bandwidth efficiency, all available locality is utilized. Bandwidth utilization refers to the amount of memory bandwidth that is utilized during a calculation. Maximum bandwidth utilization occurs when all available memory bandwidth is utilized.

Potential performance gains from using a faster microprocessor can be reduced or even negated by a corresponding drop in bandwidth efficiency and bandwidth utilization. Thus, there has been significant effort spent on the development of memory hierarchies that can maintain high bandwidth efficiency and utilization with faster microprocessors.

One approach to improving bandwidth efficiency and utilization in memory hierarchies has been to develop ever more powerful processor caches. These caches are high-speed memories (typically SRAM) in close proximity to the microprocessor that try to keep copies of instructions and data the microprocessor may soon need. The microprocessor can store and retrieve data from the cache at a much higher rate than from a slower, more distant main memory.

In designing cache memories, there are a number of considerations to take into account. One consideration is the width of the cache line. Caches are arranged in lines to help hide memory latency and exploit spatial locality. When a load suffers a cache miss, a new cache line is loaded from main memory into the cache. The assumption is that a program being executed by the microprocessor has a high degree of spatial locality, making it likely that other memory locations in the cache line will also be required.

For programs with a high degree of spatial locality (e.g., stride-one access), wide cache lines are more efficient since they reduce the number of times a processor has to suffer the latency of a memory access. However, for programs with lower levels of spatial locality, or random access, narrow lines are best as they reduce the wasted bandwidth from the unused neighbors in the cache line. Caches designed with wide cache lines perform well with programs that have a high degree of spatial locality, but generally have poor gather/scatter performance. Likewise, caches with short cache lines have good gather/scatter performance, but loose efficiency executing programs with high spatial locality because of the additional runs to the main memory.

Another consideration in cache design is cache associativity, which refers to the mapping between locations in main memory and cache sectors. At one extreme of cache associativity is a direct-mapped cache, while at another extreme is a fully associative cache. In a direct mapped-cache, a specific memory location can be mapped to only a single cache line. Direct-mapped caches have the advantage of being fast and easy to construct in logic. The disadvantage is that they suffer the maximum number of cache conflicts. At the other extreme, a fully associative cache allows a specific location in memory to be mapped to any cache line. Fully associative caches tend to be slower and more complex due to the large amount of comparison logic they need, but suffer no cache conflict misses. Oftentimes, caches fall between the extremes of direct-mapped and fully associative caches. A design point between the extremes is a k-set associative cache, where each memory location can map to k cache sectors. These caches generally have less overhead than fully associative caches, and reduce cache conflicts by increasing the value of k.

Another consideration in cache design is how cache lines are replaced due to a capacity or conflict miss. In a direct-mapped cache, there is only one possible cache line that can be replaced due to a miss. However, in caches with higher levels of associativity, cache lines can be replaced in more that one way. The way the cache lines are replaced is referred to as the replacement policy.

Options for the replacement policy include least recently used (LRU), random replacement, and first in-first out (FIFO). LRU is used in the majority of circumstances where the temporal locality set is smaller than the cache size, but it is normally more expensive to build in hardware than a random replacement cache. An LRU policy can also quickly degrade depending on the working set size. For example, consider an iterative application with a matrix size of N bytes running through a LRU cache of size M bytes. If N is less than M, then the policy has the desired behavior of 100% cache hits, however, if N is only slightly larger than M, the LRU policy results in 0% cache hits as lines are removed just as they are needed.

Another consideration is deciding on a write policy for the cache. Write-through caches send data through the cache hierarchy to main memory. This policy reduces cache coherency issues for multiple processor systems and is best suited for data that will not be re-read by the processor in the immediate future. In contrast, write-back caches place a copy of the data in the cache, but does not immediately update main memory. This type of caching works best when a data just written to the cache is quickly requested again by the processor.

In addition to write-through and write-back caches, another kind of write policy is implemented in a write-allocate cache where a cache line is allocated on a write that misses in cache. Write-allocate caches improve performance

when the microprocessor exhibits a lot of write followed by read behavior. However, when writes are not subsequently read, a write-allocate cache has a number of disadvantages: When a cache line is allocated, it is necessary to read the remaining values from main memory to complete the cache line. This adds unnecessary memory read traffic during store operations. Also, when the data is not read again, potentially useful data in the cache is displaced by the unused data.

Another consideration is made between the size and the speed of the cache: small caches are typically much faster than larger caches, but store less data and fewer instructions. Less data means a greater chance the cache will not have data the microprocessor is requesting (i.e., a cache miss) which can slow everything down while the data is being retrieved from the main memory.

Newer cache designs reduce the frequency of cache misses by trying to predict in advance the data that the microprocessor will request. An example of this type of cache is one that supports speculative execution and branch prediction. Speculative execution allows instructions that likely will be executed to start early based on branch prediction. Results are stored in a cache called a reorder buffer and retired if the branch was correctly predicted. Of course, when mis-predictions occur instruction and data bandwidth are wasted.

There are additional considerations and tradeoffs in cache design, but it should be apparent from the considerations described hereinbefore that it is very difficult to design a single cache structure that is optimized for many different programs. This makes cache design particularly challenging for a multipurpose microprocessor that executes a wide variety of programs. Cache designers try to derive the program behavior of "average" program constructed from several actual programs that run on the microprocessor. The cache is optimized for the average program, but no actual program behaves exactly like the average program. As a result, the designed cache ends up being sub-optimal for nearly every program actually executed by the microprocessor. Thus, there is a need for memory hierarchies that have data storage and retrieval characteristics that are optimized for actual programs executed by a processor.

Designers trying to develop ever more efficient caches optimized for a variety of actual programs also face another problem: as caches add additional features, the overhead needed to implement the added features also grows. Caches today have so much overhead that microprocessor performance may be reaching a point of diminishing returns as the overhead starts to cut into performance. In the Intel Pentium III processor for example, more than half of the 10 million transistors are dedicated to instruction cache, branch prediction, out-of-order execution and superscalar logic. The situation has prompted predictions that as microprocessors grow to a billion transistors per chip, performance increases will drop to about 20% per year. Such a prediction, if borne out, could have a significant impact on technology growth and the computer business.

Thus, there is a growing need to develop improved memory hierarchies that limit the overhead of a memory hierarchy without also reducing bandwidth efficiency and utilization.

SUMMARY OF THE INVENTION

Accordingly, an embodiment of the invention includes a reconfigurable processor that includes a computational unit and a data access unit coupled to the computational unit, where the data access unit retrieves data from an on-processor memory and supplies the data to the computational unit, and where the computational unit and the data access unit are configured by a program.

The present invention also involves a reconfigurable processor that includes a first memory of a first type and a data prefetch unit coupled to the memory, where the data prefetch unit retrieves data from a second memory of a second type different from the first type, and the first and second memory types and the data prefetch unit are configured by a program.

Another embodiment of the invention includes a reconfigurable hardware system that includes a common memory, also referred to as external memory, and one or more reconfigurable processors coupled to the common memory, where at least one of the reconfigurable processors includes a data prefetch unit to read and write data between the unit and the common memory, and where the data prefetch unit is configured by a program executed on the system.

Another embodiment of the invention includes a method of transferring data that includes transferring data between a memory and a data prefetch unit in a reconfigurable processor, transferring data between the prefetch unit and a data access unit, and transferring the data between a computational unit and the data access unit, where the computational unit, data access unit and the data prefetch unit are configured by a program.

Additional embodiments of the invention are set forth in part in the description that follows, and in part will become apparent to those skilled in the art upon examination of the following specification, or may be learned by the practice of the invention. The advantages of the invention may be realized and attained by means of the instrumentalities, combinations, compositions, and methods particularly pointed out in the appended claims.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** shows a reconfigurable processor in which the present invention may be implemented;

FIG. **2** shows computational logic as might be loaded into a reconfigurable processor;

FIG. **3** shows a reconfigurable processor as in FIG. **1**, but with the addition of data access units;

FIG. **4** shows a reconfigurable processor as in FIG. **3**, but with the addition of data prefetch units;

FIG. **5** shows reconfigurable processor with the inclusion of external memory;

FIG. **6** shows reconfigurable processors with external memory and with an intelligent memory controller;

FIG. **7** shows a reconfigurable processor having a combination of data prefetch units and data access units feeding computational logic;

FIG. **8** shows the bandwidth efficiency and utilization gains obtained when utilizing a data prefetch unit and an intelligent memory controller to perform strided memory references;

FIG. **9**A and FIG. **9**B show the bandwidth efficiency and utilization gains obtained when utilizing a data prefetch unit and an intelligent memory controller to perform subset memory references in X-Y plane;

FIG. **10**A and FIG. **10**B show the bandwidth efficiency and utilization gains obtained when utilizing a data prefetch unit and an intelligent memory controller to perform subset memory references in X-Z plane;

FIG. **11**A and FIG. **11**B show the bandwidth efficiency and utilization gains obtained when utilizing a data prefetch

unit and an intelligent memory controller to perform subset memory references in Y-Z plane;

FIG. **12**A and FIG. **12**B show the bandwidth efficiency and utilization gains obtained when utilizing a data prefetch unit and an intelligent memory controller to perform subset memory references in a mini-cube;

FIG. **13** shows the bandwidth efficiency and utilization gains obtained when utilizing a data prefetch unit and an intelligent memory controller to perform indirect memory references;

FIG. **14** shows the bandwidth efficiency and utilization gains obtained when utilizing a data prefetch unit and an intelligent memory controller to perform strided memory reference together with computation.

DETAILED DESCRIPTION

1. Definitions:

Direct execution logic (DEL)—is an assemblage of dynamically reconfigurable functional elements that enables a program to establish an optimized interconnection among selected functional units in order to implement a desired computational, data prefetch and/or data access functionality for maximizing the parallelism inherent in the particular code.

Reconfigurable Processor—is a computing device that contains reconfigurable components such as FPGAs and can, through reconfiguration, instantiate an algorithm as hardware.

Reconfigurable Logic—is composed of an interconnection of functional units, control, and storage that implements an algorithm and can be loaded into a Reconfigurable Processor.

Functional Unit—is a set of logic that performs a specific operation. The operation may for example be arithmetic, logical, control, or data movement. Functional units are used as building blocks of reconfigurable logic.

Macro—is another name for a functional unit.

Memory Hierarchy—is a collection of memories

Data prefetch Unit—is a functional unit that moves data between members of a memory hierarchy. The movement may be as simple as a copy, or as complex as an indirect indexed strided copy into a unit stride memory.

Data access Unit—is a functional unit that accesses a component of a memory hierarchy, and delivers data directly to computational logic.

Intelligent Memory Control Unit—is a control unit that has the ability to select data from its storage according to a variety of algorithms that can be selected by a data requester, such as a data prefetch unit.

Bandwidth Efficiency—is defined as the percentage of contributory data transferred between two points. Contributory data is data that actually participates in the recipients processing.

Bandwidth Utilization—is defined as the percentage of maximum bandwidth between two points that is actually used to pass contributory data.

2. Description

A reconfigurable processor (RP) **100** implements direct executable logic (DEL) to perform computation, as well a memory hierarchy for maintaining input data and computational results. DEL is an assemblage of dynamically reconfigurable functional elements that enables a program to establish an optimized interconnection among selected functional units in order to implement a desired computational, data prefetch and/or data access functionality for maximizing the parallelism inherent in the particular code. The term DEL may also be used to refer to the set of constructs such as code, data, configuration variables, and the like that can be loaded into RP **100** to cause RP **100** to implement a particular assemblage of functional elements.

FIG. **1** presents an RP **100**, which may be implemented using field programmable gate arrays (FPGAs) or other reconfigurable logic devices, that can be configured and reconfigured to contain functional units and interconnecting circuits, and a memory hierarchy comprising on-board memory banks **104**, on-chip block RAM **106**, registers wires, and a connection **108** to external memory. On-chip reconfigurable components **102** create memory structures such as registers, FIFOs, wires and arrays using block RAM. Dual-ported memory **106** is shared between on-chip reconfigurable components **102**. The reconfigurable processor **100** also implements user-defined computational logic (e.g., such as DEL **200** shown in FIG. **2**) constructed by programming an FPGA to implement a particular interconnection of computational functional units. In a particular implementation, a number of RPs **100** are implemented within a memory subsystem of a conventional computer, such as on devices that are physically installed in dual inline memory module (DIMM) sockets of a computer. In this manner the RPs **100** can be accessed by memory operations and so coexist well with a more conventional hardware platform. It should be noted that, although the exemplary implementation of the present invention illustrated includes six banks of dual ported memory **104** and two reconfigurable components **102**, any number of memory banks and/or reconfigurable components may be used depending upon the particular implementation or application.

Any computer program, including complex graphics processing programs, word processing programs, database programs and the like, is a collection of algorithms that interact to implement desired functionality. In the common case in which static computing hardware resources are used (e.g., a conventional microprocessor), the computer program is compiled into a set of executable code (i.e., object code) units that are linked together to implement the computer program on the particular hardware resources. The executable code is generated specifically for a particular hardware platform. In this manner, the computer program is adapted to conform to the limitations of the static hardware platform. However, the compilation process makes many compromises based on the limitations of the static hardware platform.

Alternatively, an algorithm can be defined in a high level language then compiled into DEL. DEL can be produced via a compiler from high level programming languages such as C or FORTRAN or may be designed using a hardware definition language such as Verilog, VHDL or a schematic capture tool. Computation is performed by reconfiguring a reconfigurable processor with the DEL and flowing data through the computation. In this manner, the hardware resources are essentially adapted to conform to the program rather than the program being adapted to conform to the hardware resources.

For purposes of this description a single reconfigurable processor will be presented first. A sample of computational logic **201** is shown in FIG. **2**. This simple assemblage of functional units performs computation of two results ("A+B" and "A+B−(B*C)) from three input variables or operands "A", "B" and "C". In practice, computational units **201** can be implemented to perform very simple or arbitrarily complex computations. The input variables (operands) and output or result variables may be of any size necessary for a particular application. Theoretically, any number of oper-

ands and result variables may be used/generated by a particular DEL. Great complexity of computation can be supported by adding additional reconfigurable chips and processors.

For greatest performance the DEL **200** is constructed as parallel pipelined logic blocks composed of computational functional units capable of taking data and producing results with each clock pulse. The highest possible performance that can be achieved is computation of a set of results with each clock pulse. To achieve this, data should be available at the same rate the computation can consume the data. The rate at which data can be supplied to DEL **200** is determined, at least in significant part, by the memory bandwidth utilization and efficiency. Maximal computational performance can be achieved with parallel and pipelined DEL together with maximizing the memory bandwidth utilization and efficiency. Unlike conventional static hardware platforms, however, the memory hierarchy provided in a RP **100** is reconfigurable. In accordance with the present invention, through the use of data access units and associated memory hierarchy components, computational demands and memory bandwidth can be matched.

High memory bandwidth efficiency is achieved when only data required for computation is moved within the memory hierarchy. FIG. **3** shows a simple logic block **300** comprising computational functional units **301**, control (not shown), and data access functional units **303**. The data access unit **303** presents data directly to the computational logic **301**. In this manner, data is moved from a memory device **305** to the computational logic and from the computational logic back into a memory device **305** or block RAM memory **307** within an RP **100**.

FIG. **4** illustrates the logic block **300** with an addition of a data prefetch unit **401**. The data prefetch unit **401** moves data from one member of the memory hierarchy **305** to another **308**. Data prefetch unit **401** operates independently of other functional units **301**, **302** and **303** and can therefore operate prior to, in parallel with, or after computational logic. This independence of operation permits hiding the latency associated with obtaining data for use in computation. The data prefetch unit deposits data into the memory hierarchy within RP **100**, where computational logic **301**, **302** and **303** can access it through data access units. In the example of FIG. **4**, prefetch unit **401** is configured to deposit data into block RAM memory **308**. Hence, the prefetch units **401** may be operated independently of logic block **300** that uses prefetched data.

An important feature of the present invention is that many types of data prefetch units can be defined so that the prefetch hardware can be configured to conform to the needs of the algorithms currently implemented by the computational logic. The specific characteristics of the prefetch can be matched with the needs of the computational logic and the format and location of data in the memory hierarchy. For example, FIG. **9**A and FIG. **9**B show an external memory that is organized in a 128 byte (16 word) block structure. This organization is optimized for stride **1** access of cache based computers. A stride **128** access can result in a very inefficient use of bandwidth from the memory, since an extra 120 bytes of data is moved for every 8 bytes of requested data yielding a 6.25% bandwidth efficiency.

FIG. **5** shows an example of data prefetch in which there are no bandwidth gains since all data fetched from external memory blocks is also transferred and used in computational units **301** through memory bank access units **303**. However, bandwidth utilization is increased due to the ability of the data prefetch units **501** to initiate a data transfer in advance of the requirement for data by computational logic.

In accordance with an embodiment of the present invention, data prefetch units **601** are configured to communicate with an intelligent memory controller **603** in FIG. **6** and can extract only the desired 8 bytes of data, discard the remainder of the memory block, and transmit to the data prefetch unit only the requested portion of the stride **128** data. The prefetch units **601** then delivers that data to the appropriate memory components within the memory hierarchy of the logic block **300**.

FIG. **6** shows the prefetch units **601** delivering data to the RP's onboard memory banks **305**. An onboard memory bank data access unit **303** then delivers the data to computational logic **301** when required. The data prefetch units **501** couple with an intelligent memory controller **601** in the implementation of FIG. **6** that supports a strided reference pattern, which yields a 100% bandwidth efficiency in contrast to the 6.25% efficiency. Although illustrated as a single block of external memory, multiple numbers of external memories may be employed as well.

In FIG. **7**, the combination of data prefetch units **701** and data access units **703** feeding computational logic **301** such that bandwidth efficiency and utilization are maximized is shown in FIG. **7**. In this example strided data prefetch units **701** fetch only the required data words from external memory. FIG. **8** demonstrates the efficiency gains enabled by this combination. Prefetch units **701** deliver the data into stream memory components **705** that is accessed by stream data access units **703**. The stream data access units **703** fetch data from the stream based on valid data bits that are provided to the stream by the data prefetch units **701** as data is presented to the stream. Use of the stream data access unit allows computational logic to be activated upon initiation of the data prefetch operation. This, in turn, allows computation to start with the arrival of the first data item, signaled by valid data bits. Computational logic **301** does not have to await arrival of a complete buffer of data in order to proceed. This elimination of latency increases the bandwidth utilization, by allowing data transfer to continue uninterrupted and in parallel with computation.

FIG. **8** illustrates the efficiency gains enabled by the configuration of FIG. **7**. FIG. **8** shows a plurality of memory blocks **800** in which only one memory element **801** exists in each memory block **800**. The configuration of FIG. **7** allows the desired portions **801** of each memory block **800** to be compacted into a transfer buffer **805**. The desired data elements **801** are compacted in order. Since only the contents of the transfer buffer **805** need be transferred to the computational logic, a significant increase in transfer efficiency can be realized.

FIGS. **9**A/**9**B, **10**A/**10**B, **11**A/**11**B and **12**A/**12**B show bandwidth efficiency gains that are achieved in various situations when a subset of stored data is required for computation. Applications store data in a specific order in memory. However it is often the case that the actual reference pattern required during computation is different from the ordering of data in memory. FIGS. **9**A/**9**B, **10**A/**10**B, **11**A/**11**B and **12**A/**12**B show an example of a X,Y,Z coordinate oriented data which is stored such that striding though the X axis is the most efficient for retrieving blocked data.

Coupling data prefetch units in the RP **100** with an intelligent memory controller **601** in the external memory yields a significant improvement in bandwidth efficiency and utilization. Four examples are presented in the FIGS. **9**A/**9**B, **10**A/**10**B, **11**A/**11**B and **12**A/**12**B in which the shaded memory locations indicate desired data. The Figures

illustrate an intelligent memory controller's response to each of four different data prefetch unit's requests for data. Again, an important feature of the present invention is the ability to implement various kinds or styles of prefetch units to meet the needs of a particular algorithm being implemented by computational elements **301**. For ease of illustration, each example shows the same set of computational logic, however, in most cases the function being implemented by components **301** would change and therefore alter the decision as to which prefetch strategy is most appropriate. In accordance with the present invention, the prefetch units are implemented in a manner that is optimized for the implemented computational logic.

FIGS. **9**A/**9**B shows response to a request from an XY-slice data prefetch unit. FIGS. **10**A/**10**B shows response to a XZ-slice data prefetch unit request. FIGS. **11**A/**11**B shows response to a YZ-slice data prefetch unit request. FIGS. **12**A/**12**D shows the response to a SubCube data prefetch unit request. In each of these examples the data prefetch units are configured to pass information to the intelligent memory controller **601** to identify the type of request that is being made, as well as a data address and parameters, in this case, defining the slice size or sub-cube size.

One of the largest bandwidth efficiency and utilization gains can be seen in the case of a Gather data prefetch unit working in cooperation with an intelligent memory controller **601**. FIG. **13** illustrates the activity in the external memory controller **601**. In this example an index array **1301** and a data array **1303** reside in memory. A gather data prefetch unit in an RP **100** requests a gather by specifying the access type as "gather", and providing a pointer to index array **1301**, and another pointer to the data array **1303**. The memory controller uses the index array **1301** to select desired data elements, indicated by shading, and then delivers an in order stream of data to the prefetch unit. Gains are made by delivering only requested data from transfer buffer **1305** (not the remainder of a data block as in cache line oriented systems) by eliminating the need to transfer an index array either to the processor or to the memory controller, and by eliminating the start/stop time required when the data is not streamed to the requestor.

A further bandwidth efficiency and utilization gain is made when coupling a data prefetch unit with memory controller capable of computation. FIG. **14** illustrates activity in a cooperating memory controller having a computational component **1407** in response to a data prefetch unit. Here the prefetch units requests a "strided compute", providing parameters for an operator, and addresses, and strides for data to be operated upon. In FIG. **14**, the data to be operated on comprises "X" data **1401** and "Y" data **1403**. The data **1401** and **1403** are processed by computational component **1407** to generate a resultant value that is a specified function of X and Y as indicated by F(X,Y) in FIG. **14**. The resultant values are then passed to the requesting prefetch unit via transfer buffer **1405**. In this case only computed results are passed and no operand data need to transferred. Accordingly, where the desired data, indicated by shading in FIG. **14**, resides across multiple blocks, efficiency is achieved not only by avoiding transfer of the undesired data surrounding the desired data, but also because only the result is transferred, not the original data **1401**/**1403**.

EXAMPLES

Some programming examples utilizing the memory hierarchy of the present invention will now be illustrated. The first example illustrates how a computational intensive matrix multiplication problem may be handled by the explicitly parallel and addressable storage of the present invention.

Example 1

Explicit Parallel and Addressable Storage

Consider the matrix multiplication C=A×B, where:
A is a matrix of size M rows by 64 columns;
B is a matrix of size 64 rows by N columns; and
C is a matrix of size M rows by N columns.

The size and shape of this problem typically arises in the context of LU decomposition in linear algebra libraries (e.g., LAPACK). The operation count for this problem would be 2*M*N*64, and the total data necessary to transport would be (M*64+N*64+M*N), making the problem quite computationally intensive.

The dot-product formulation of the matrix multiplication may be represented as the following a triple-nested loop:

```
for (i = 0; I<m; I++) {
        for (j = 0; j< n; j++) {
            sum = 0;
for (k = 0; k < 64; k++) {
            sum += A[k*m*I] * B[j*64+k];
    }
    C1[i+j+mn] = sum;
  }
}
```

On a conventional microprocessor with static execution resources, these loops would be arranged to give stride-one data access where possible and also block or tile these uses to facilitate data cache hits on the B and A matrices, which are read many times. With the configurable memory hierarchy of the present invention, matrix B may be stored in on-board BRAM memory **307** and rows of matrix A in registers.

The rows of matrix B may be stored in independently, locally declared BRAM arrays (B**0**, B**1**, . . . B**63**). The rows are stored as independent memory structures, and may be accessed in parallel. Rows of matrix A may be stored in 64 registers described with scalar variables. With these explicit data structures, the following pseudo code can describe the matrix multiplication:

```
Load B into BRAM;
for (i=0; i<m; i++) {
Load ith Row of A into registers A00 to A63;
For (j=0; j<n; j++) {
C[i+j+m]+=
A00*b0[j]+
A01*b1[j]+
A02*b2[j]+//inner loop produces
A03*b3[j]+//128 results per
A04*b4[j]+//clock cycle. 64 rows
A05*b5[j]+//of B are read in
A06*b6[j]+//parallel
. . .
A63*b63[j];
```

The code is designed to minimize the amount of data motion. The A and B matrices are read once and the C matrix is written just once at it is produced. When computational resources permit, the i loop could also be unrolled to process multiple rows of matrix A against matrix B in the inner loop.

Processing two rows of A, for example, would produce 256 computational results per clock cycle.

Example 2

Irregular Memory Access

Benchmarks have been developed for measuring the ability of a computer system to perform indirect updates. An indirect update, written in the C programming language, looks like:

```
for (I = 0; I < N; I++) {
A[Index[I]]) = A[Index[I]] + B[I];
}
```

Typically, A is a large array, and Index has an unpredictable distribution. The benchmark generally forces memory references to miss in cache, and for entire cache lines to be brought in for single-word updates. The problem gets worse as memories get further away from processors and cache lines become wider.

In this example, the arrays have 64-bit data. To complete one iteration of this loop, 24 bytes of information is required from memory and 8 bytes are written back for a total of 32 bytes of memory motion per iteration. On an implicit architecture with cache-lines of width W bytes, each iteration results in the following memory bus traffic:

1. Index[I]: 8 bytes per iteration due to stride-1 nature;
2. B[I]: 8 bytes per iteration due to stride-1 nature; and
3. A[Index[I]]: W bytes read and written per iteration.

The total amount of bus traffic is 2*W+16 bytes per iteration. On an average microprocessor today, W=128 so an iteration of this loop results in 272 bytes of memory traffic when only 32 bytes is algorithmically required, making only 12% of the data moved as being useful for the problem.

In addition, because microprocessors rely on wide cache lines and hardware pre-fetching strategies to amortize the long latency to main memory, only a small number of outstanding cache-line misses are typically tolerated. Because of the irregular nature of this example, hardware pre-fetching provides little benefit, making it difficult to keep the memory bus saturated, even with the large amount of wasted memory traffic. Bus utilization on the microprocessor processing only consumes about 700 MB/sec of the 3.2 GB/sec available, or 22%. Combining the poor bus utilization with the relatively small amount of data that is useful results in the microprocessor executing at about 2.5% of peak.

The memory hierarchy of the present invention does not require that memory traffic be organized in a cache-line structure, permitting loop iteration to be accomplished with the minimum number of bytes (in this case 32 bytes of memory traffic). In addition, data pre-fetch functional units may be fully pipelined, allowing full use of available memory bus bandwidth. Data storing may be handled in a similar pipelined fashion. An example of the pseudo code that performs the random update in the memory hierarchy looks like:

```
for (i=0; I < N-Gather_size; I=I+Gather_size) {
        gather ( A, Index, I, A_local, Gather_size)
        for (j=); j < Gather_size; j++) {
            A_local[j] = A_local[j] + B[j];
        }
        scatter (A_local, Index, &A[I], Gather_size);
    }
```

This loop will pipeline safely as described by the pseudo code provided that the index vector has no repeated values within each Gather_size segment. If repeats are present, then logic within the gather unit can preprocess the Index vector and B vector into safe sub-lists that can be safely pipelined with little or no overhead.

CONCLUSION

It should be apparent that the scaleable, programmable memory mechanisms enabled by the present invention are available to the exploit available algorithm locality and thereby achieve up to 100% bandwidth efficiency. In addition, the scaleable computational resources can be leveraged to attain 100% bandwidth utilization. As a result, the present invention provides a programmable computational system that delivers the maximum possible performance for any memory bus speed. This combination of efficiency and utilization yields orders of magnitude performance benefit compared with implicit models when using an equivalent memory bus.

Although the invention has been described and illustrated with a certain degree of particularity, it is understood that the present disclosure has been made only by way of example, and that numerous changes in the combination and arrangement of parts can be resorted to by those skilled in the art without departing from the spirit and scope of the invention, as hereinafter claimed.

We claim:

1. A reconfigurable processor that instantiates an algorithm as hardware comprising:

a first memory having a first characteristic memory bandwidth and/or memory utilization; and

a data prefetch unit coupled to the memory, wherein the data prefetch unit retrieves only computational data required by the algorithm from a second memory of second characteristic memory bandwidth and/or memory utilization and places the retrieved computational data in the first memory wherein the data prefetch unit operates independent of and in parallel with logic blocks using the computional data, and wherein at least the first memory and data prefetch unit are configured to conform to needs of the algorithm, and the data prefetch unit is configured to match format and location of data in the second memory.

2. The reconfigurable processor of claim **1**, wherein the data prefetch unit is coupled to a memory controller that controls the transfer of the data between the memory and the data prefetch unit and transmits only portions of data desired by the data prefetch unit and discards other portions of data prior to transmission of the data to the data prefetch unit.

3. The reconfigurable processor of claim **1**, wherein the data prefetch unit receives processed data from on-processor memory and writes the processed data to an external off-processor memory.

4. The reconfigurable processor of claim **1**, wherein the data prefetch unit comprises at least one register from the reconfigurable processor.

5. The reconfigurable processor of claim 1, wherein the data prefetch unit is disassembled when another program is executed on the reconfigurable processor.

6. The reconfigurable processor of claim 1 wherein said second memory comprises a processor memory and said data prefetch unit is operative to retrieve data from a processor memory.

7. The reconfigurable processor of claim 6 wherein said processor memory is a microprocessor memory.

8. The reconfigurable processor of claim 6 wherein said processor memory is a reconfigurable processor memory.

9. A reconfigurable hardware system, comprising:

a common memory; and

one or more reconfigurable processors that can instantiate an algorithm as hardware coupled to the common memory, wherein at least one of the reconfigurable processors includes a data prefetch unit to read and write only data required for computations by the algorithm between the data prefetch unit and the common memory wherein the data prefetch unit operates independent of and in parallel with logic blocks using the computational data, and wherein the data prefetch unit is configured to conform to needs of the algorithm and match format and location of data in the common memory.

10. The reconfigurable hardware system of claim 9, comprising a memory controller coupled to the common memory and the data prefetch unit that transmits to the prefetch unit only data desired by the data prefetch unit as required by the algorithm.

11. The reconfigurable hardware system of claim 9, wherein the at least of the reconfigurable processors also includes a computational unit coupled to the data access unit.

12. The reconfigurable hardware system of claim 11, wherein the computational unit is supplied the data by the data access unit.

13. A method of transferring data comprising:

transferring data between a memory and a data prefetch unit in a reconfigurable processor; and

transferring the data between a computational unit and the data access unit, wherein the computational unit and the data access unit, and the data prefetch unit are configured to conform to needs of an algorithm implemented on the computational unit and transfer only data necessary for computations by the computational unit, and wherein the prefetch unit operates independent of and in parallel with the computational unit.

14. The method of claim 13, wherein the data is written to the memory, said method comprising:

transferring the data from the computational unit to the data access unit; and

writing the data to the memory from the data prefetch unit.

15. The method of claim 13, wherein the data is read from the memory, said method comprising:

transferring only the data desired by the data prefetch unit as required by the computational unit from the memory to the data prefetch unit; and

reading the data directly from the data prefetch unit to the computational unit through a data access unit.

16. The method of claim 15, wherein all the data transferred from the memory to the data prefetch unit is processed by the computational unit.

17. The method of claim 15, wherein the data is selected by the data prefetch unit based on an explicit request from the computational unit.

18. The method of claim 13, wherein the data transferred between the memory and the data prefetch unit is not a complete cache line.

19. The method of claim 13, wherein a memory controller coupled to the memory and the data prefetch unit, controls the transfer of the data between the memory and the data prefetch unit.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 7,149,867 B2
APPLICATION NO. : 10/869200
DATED : December 12, 2006
INVENTOR(S) : Daniel Poznanovic, David E. Caliga and Jeffrey Hammes

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 12, line 43, insert --first-- after "coupled to the"

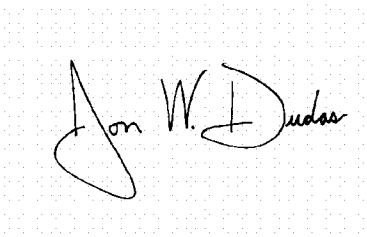
Column 12, line 57, insert --second-- after "between the"

Column 13, line 6, "a" should be --the--

Column 14, line 4, the second occurrence of "the" should be --a--.

Signed and Sealed this

Twenty-fourth Day of April, 2007



JON W. DUDAS
Director of the United States Patent and Trademark Office

EXHIBIT I

UNITED STATES DEPARTMENT OF COMMERCE
Patent and Trademark Office

MF

NOTICE OF ALLOWANCE AND ISSUE FEE DUE

RECEIVED
FEB - 9 2001
HOGAN & HARTSON, LLP

TM02/0205

WILIAM J KUBIDA
HOGAN & HARTSON LLP
1200 17TH STREET
SUITE 1500
DENVER CO 80202

APPLICATION NO.	FILING DATE	TOTAL CLAIMS	EXAMINER AND GROUP ART UNIT	DATE MAILED
09/481,902	01/12/00	022	FOLLANSBEE, J 2154	02/05/01

First Named Applicant: HUPPENTHAL, 35 USC 154(b) term ext. = 0 Days.

TITLE OF INVENTION: MULTIPROCESSOR COMPUTER ARCHITECTURE INCORPORATING A PLURALITY OF MEMORY ALGORITHM PROCESSORS IN THE MEMORY SUBSYSTEM

ATTY'S DOCKET NO.	CLASS-SUBCLASS	BATCH NO.	APPLN. TYPE	SMALL ENTITY	FEE DUE	DATE DUE
0 SRC-001-CON	712-015.000	859	UTILITY	YES	$620.00	05/07/01

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED.

THE ISSUE FEE MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED.

HOW TO RESPOND TO THIS NOTICE:

I. Review the SMALL ENTITY status shown above.
If the SMALL ENTITY is shown as YES, verify your current SMALL ENTITY status:

A. If the status is changed, pay twice the amount of the FEE DUE shown above and notify the Patent and Trademark Office of the change in status, or
B. If the status is the same, pay the FEE DUE shown above.

If the SMALL ENTITY is shown as NO:

A. Pay FEE DUE shown above, or

B. File verified statement of Small Entity Status before, or with, payment of 1/2 the FEE DUE shown above.

II. Part B-Issue Fee Transmittal should be completed and returned to the Patent and Trademark Office (PTO) with your ISSUE FEE. Even if the ISSUE FEE has already been paid by charge to deposit account, Part B Issue Fee Transmittal should be completed and returned. If you are charging the ISSUE FEE to your deposit account, section "4b" of Part B-Issue Fee Transmittal should be completed and an extra copy of the form should be submitted.

III. All communications regarding this application must give application number and batch number. Please direct all communications prior to issuance to Box ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

YOUR COPY

PTOL-85 (REV. 10-96) Approved for use through 06/30/99. (0651-0033)

*U.S. GPO: 1999-454-457/24601

Notice of Allowability	Application No.	Applicant(s)	
	09/481,902	HUPPENTHAL ET AL.	
	Examiner	Art Unit	
	John Follansbee	2154	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance and Issue Fee Due or other appropriate communication will be mailed in due course.

1. ☒ This communication is responsive to *1/03/01*.
2. ☒ The allowed claim(s) is/are *25-46 (now 1-22)*.
3. ☐ The drawings filed on _____ are acceptable.
4. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d).
   a) ☐ All b) ☐ Some* c) ☐ None of the CERTIFIED copies of the priority documents have been
      1. ☐ received.
      2. ☐ received in Application No. (Series Code / Serial Number). _____ .
      3. ☐ received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.
5. ☐ Acknowledgement is made of a claim for domestic priority under 35 U.S.C. & 119(e).

A SHORTENED STATUTORY PERIOD FOR REPLY to comply with the requirements noted below is set to EXPIRE **THREE MONTHS** FROM THE "DATE MAILED" of this Office Action. Failure to timely comply will result in ABANDONMENT of this application. Extensions of time may be available under the provisions of 37 CFR 1.136(a).

6. ☐ Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient. A SUBSTITUTE OATH OR DECLARATION IS REQUIRED.

7. ☐ Applicant MUST submit NEW FORMAL DRAWINGS
   (a) ☐ because the originally filed drawings were declared by applicant to be informal.
   (b) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review(PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No. ____.
   (c) ☐ including changes required by the proposed drawing correction filed _____, which has been approved by the examiner.
   (d) ☐ including changes required by the attached Examiner's Amendment / Comment.

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the reverse side of the drawings. The drawings should be filed as a separate paper with a transmittal letter addressed to the Official Draftsperson.

8. ☐ Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Any reply to this letter should include, in the upper right hand corner, the APPLICATION NUMBER (SERIES CODE / SERIAL NUMBER). If applicant has received a Notice of Allowance and Issue Fee Due, the ISSUE BATCH NUMBER and DATE of the NOTICE OF ALLOWANCE should also be included.

Attachment(s)

1☐ Notice of References Cited (PTO-892)
3☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
5☐ Information Disclosure Statements (PTO-1449), Paper No. ____.
7☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
2☐ Notice of Informal Patent Application (PTO-152)
4☐ Interview Summary (PTO-413), Paper No.____ .
6☒ Examiner's Amendment/Comment
8☐ Examiner's Statement of Reasons for Allowance
9☐ Other

JOHN A. FOLLANSBEE
PRIMARY EXAMINER

Application Number: 09/481,902
Art Unit: 2154

Examiner's Amendment

1. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

2. The application has been amended as follows:

In the claims:

I. **Cancel claims 47-65.**

Any inquiry concerning this communication or earlier communications from the examiner should be directed to John Follansbee whose telephone number is (703) 305-8498.

John Follansbee

February 2, 2001.

JOHN A. FOLLANSBEE
PRIMARY EXAMINER

Client Matter No. 80404.0004.001
Attorney Docket No. SRC001CON
Express Mail No.: EL700671795US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: Jon M. Huppenthal and Paul A. Leskar Serial No. 09/481,902 Filed: January 12, 2000 For: MULTIPROCESSOR COMPUTER ARCHITECTURE INCORPORATING A PLURALITY OF MEMORY ALGORITHM PROCESSORS IN THE MEMORY SUBSYSTEM	Examiner: Follansbee, J. Art Unit: 2154

AMENDMENT AND RESPONSE PURSUANT TO OFFICE ACTION DATED November 21, 2000

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

In response to the office communication mailed November 21, 2000, please enter the enclosed Terminal Disclaimer and Restriction Requirement.

REMARKS

In the first Office Action mailed November 21, 2000, claims 25-65 were made subject to a restriction requirement. A Response to Restriction Requirement is enclosed herewith.

In the same Office Action mailed November 21, 2000, pending claims 25-46 were rejected under the judicially created doctrine of obviousness-type double patenting over U.S. Patent No. 6,076,152 to the assignee hereof. The Examiner indicated that a timely filed terminal disclaimer could be used to overcome this rejection. A Terminal Disclaimer and check for $110.00

representing the required fee, are enclosed herewith.

IN THE CLAIMS:

Please amend the following claims:

25. (Amend) In a computer system having at least one data processor for executing an application program by operating on user data in accordance with application program instructions, said computer system having at least one memory bank with a data bus and an address bus connected to said at least one data processor, the improvement comprising:

a plurality of reconfigurable memory algorithm processors within individually addressable portions of said memory bank[.],

means connecting said plurality of memory algorithm processors to said data bus and to said address bus such that said plurality of memory algorithm processors are individually memory addressable by said at least one data processor at said one data processor executes said application program; and

said plurality of memory algorithm processors being configured as individual data processing elements to perform data processing related to said application program in accordance with an identified algorithm, said data processing being performed on at least one operand that is received directly from said at least one data processor.

35. (Amend) A multiprocessor computer system comprising:

[A]a plurality of data processors for executing at least one application program by operating on user data in accordance with program instructions;

a memory bank having a data bus and an address bus connected to said plurality of data processors;

a plurality of reconfigurable memory algorithm processors within said memory bank at plurality of individual memory addressable memory locations;

means coupling said plurality of individual memory algorithm processors to said data bus and to said address bus;

said plurality of [individual] reconfigurable memory algorithm processors being individually memory addressable by all of said plurality of data processors; and

said plurality of memory algorithm processors being individually configurable to perform an identified algorithm on an operand that is received from a write operation by one of said plurality of data processors to said memory bank as said at least one of said plurality of data processors executes said at least one application program.

No additional fees are believed due for this filing. However any fee deficiency associated herewith may be charged to Deposit Account No. 50-1123.

02 January, 2000

Respectfully submitted,

William J. Kubida, No. 29,664
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5909 Tel
(303) 899-7333 Fax

Attorney Docket No. SRC001CON
Client/Matter No.: 80404.0004.001
Express Mail Number: EL700671795US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

Jon M. Huppenthal and Paul A. Leskar

Serial No. 09/481,902

Filed: January 12, 2000

For: MULTIPROCESSOR COMPUTER ARCHITECTURE INCORPORATING A PLURALITY OF MEMORY ALGORITHM PROCESSORS IN THE MEMORY SUBSYSTEM

Group Art Unit: 2154

Examiner: Follansbee, J.

RESPONSE TO RESTRICTION REQUIREMENT

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

In the Office Action dated November 21, 2000, the Examiner entered a restriction requirement requiring election between continued prosecution of claims 25-46 and claims 47-65.

The Applicant elects claims 25-46 for continued prosecution. Accordingly, please withdraw claims 47-65 without prejudice from further prosecution in this application.

Dated: 02 January 2001

Respectfully submitted,

William J. Kubida, Reg. No. 29,664
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5909 Tel
(303) 899-7333 Fax

\\\CS - 80404/4 - #41138 v1

Office Action Summary	Application No. 09/481,902	Applicant(s) Huppenthal et al.
	Examiner John Follansbee	Group Art Unit 2154

☒ Responsive to communication(s) filed on *Jun 30, 2000* .

☐ This action is **FINAL.**

☐ Since this application is in condition for allowance except for formal matters, **prosecution as to the merits is closed** in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11; 453 O.G. 213.

A shortened statutory period for response to this action is set to expire *three* month(s), or thirty days, whichever is longer, from the mailing date of this communication. Failure to respond within the period for response will cause the application to become abandoned. (35 U.S.C. § 133). Extensions of time may be obtained under the provisions of 37 CFR 1.136(a).

Disposition of Claims

☒ Claim(s) *25-65* is/are pending in the application.

Of the above, claim(s) *47-65* is/are withdrawn from consideration.

☐ Claim(s) _____ is/are allowed.

☒ Claim(s) *25-46* is/are rejected.

☐ Claim(s) _____ is/are objected to.

☒ Claims *25-65* are subject to restriction or election requirement.

Application Papers

☐ See the attached Notice of Draftsperson's Patent Drawing Review, PTO-948.

☐ The drawing(s) filed on _____ is/are objected to by the Examiner.

☐ The proposed drawing correction, filed on _____ is ☐approved ☐disapproved.

☐ The specification is objected to by the Examiner.

☐ The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. § 119

☐ Acknowledgement is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d).

☐ All ☐ Some* ☐ None of the CERTIFIED copies of the priority documents have been

☐ received.

☐ received in Application No. (Series Code/Serial Number) _____ .

☐ received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

*Certified copies not received: _____ .

☐ Acknowledgement is made of a claim for domestic priority under 35 U.S.C. § 119(e).

Attachment(s)

☒ Notice of References Cited, PTO-892

☒ Information Disclosure Statement(s), PTO-1449, Paper No(s). *7-9*

☐ Interview Summary, PTO-413

☐ Notice of Draftsperson's Patent Drawing Review, PTO-948

☐ Notice of Informal Patent Application, PTO-152

--- SEE OFFICE ACTION ON THE FOLLOWING PAGES ---

1. Claims 25-65 are pending in the application. Claims 1-24 have been canceled.

2. Restriction to one of the following inventions is required under 35 U.S.C. 121:

I. Claims 25-46, drawn to processor architecture (i.e., reconfigurable memory algorithm processors within individually portions of the memory bank), classified in class 712, subclass 37.

II. Claims 47-65, drawn to the implementation of application specific devices, classified in class 712, subclass 221.

3. The inventions are distinct, each from the other because of the following reasons: Inventions I and II are related as subcombinations disclosed as usable together in a single combination. The subcombinations are distinct from each other if they are shown to be separately usable. In the instant case, invention II has separate utility such as in a system requiring the reconfigurable memory algorithm processors within individually portions of the memory bank. See MPEP § 806.05(d).

4. Because these inventions are distinct for the reasons given above and have acquired a separate status in the art as shown by their different classification, restriction for examination purposes as indicated is proper.

5. Because these inventions are distinct for the reasons given above and the search required for Group I is not required for Group II, restriction for examination purposes as indicated is proper.

6. During a telephone conversation with William Kubida on 11/15/00 a provisional election was made with traverse to prosecute the invention of Group I, claims 25-46. Affirmation of this election must be made by applicant in replying to this Office action. Claims 47-65 are withdrawn from further consideration by the examiner, 37 CFR 1.142(b), as being drawn to a non-elected invention.

7. Applicant is reminded that upon the cancellation of claims to a non-elected invention, the inventorship must be amended in compliance with 37 CFR 1.48(b) if one or more of the currently named inventors is no longer an inventor of at least one claim remaining in the application. Any amendment of inventorship must be accompanied by a petition under 37 CFR 1.48(b) and by the fee required under 37 CFR 1.17(I).

8. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970);and, *In re Thorington,* 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321© may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

9. Claims 25-46 are rejected under the judicially created doctrine of double patenting over claims 1-22 of U. S. Patent No. 6,076,152 since the claims, if allowed, would improperly extend the "right to exclude" already granted in the patent.

The subject matter claimed in the instant application is fully disclosed in the patent and is covered by the patent since the patent and the application are claiming common subject matter.

Furthermore, there is no apparent reason why applicant was prevented from presenting claims corresponding to those of the instant application during prosecution of the application which matured into a patent. See *In re Schneller*, 397 F.2d 350, 158 USPQ 210 (CCPA 1968). See also MPEP § 804.

10. Claim 25 is objected to because of the following informalities: line 7 "." should be --,-- and line 17, there is no period. Also, claim 35, line 2, "A" should be --a--. Appropriate correction is required.

11. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

12. Claim 35 recites the limitation "said plurality of individual memory algorithm processors" in line 10. There is insufficient antecedent basis for this limitation in the claim.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to John Follansbee whose telephone number is (703) 305-8498.

Any inquiry of a general nature or relating to the status of this application should be directed to the Group receptionist whose telephone number is (703) 305-3900.

John Follansbee

November 14, 2000.

JOHN A. FOLLANSBEE
PRIMARY EXAMINER

UNITED STATES DEPARTMENT OF COMMERCE
Patent and Trademark Office
ASSISTANT SECRETARY AND COMMISSIONER
OF PATENTS AND TRADEMARKS
Washington, D.C. 20231

OCTOBER 17, 2000

PTAS

101446840A

HOGAN & HARTSON LLP
CAROL W. BURTON, ESQ.
1200 17TH STREET, SUITE 1500
DENVER, CO 80202

UNITED STATES PATENT AND TRADEMARK OFFICE
NOTICE OF RECORDATION OF ASSIGNMENT DOCUMENT

THE ENCLOSED DOCUMENT HAS BEEN RECORDED BY THE ASSIGNMENT DIVISION OF THE U.S. PATENT AND TRADEMARK OFFICE. A COMPLETE MICROFILM COPY IS AVAILABLE AT THE ASSIGNMENT SEARCH ROOM ON THE REEL AND FRAME NUMBER REFERENCED BELOW.

PLEASE REVIEW ALL INFORMATION CONTAINED ON THIS NOTICE. THE INFORMATION CONTAINED ON THIS RECORDATION NOTICE REFLECTS THE DATA PRESENT IN THE PATENT AND TRADEMARK ASSIGNMENT SYSTEM. IF YOU SHOULD FIND ANY ERRORS OR HAVE QUESTIONS CONCERNING THIS NOTICE, YOU MAY CONTACT THE EMPLOYEE WHOSE NAME APPEARS ON THIS NOTICE AT 703-308-9723. PLEASE SEND REQUEST FOR CORRECTION TO: U.S. PATENT AND TRADEMARK OFFICE, ASSIGNMENT DIVISION, BOX ASSIGNMENTS, CG-4, 1213 JEFFERSON DAVIS HWY, SUITE 320, WASHINGTON, D.C. 20231.

RECORDATION DATE: 08/30/2000

REEL/FRAME: 011024/0701
NUMBER OF PAGES: 4

BRIEF: ASSIGNMENT OF ASSIGNOR'S INTEREST (SEE DOCUMENT FOR DETAILS).

ASSIGNOR:
HUPPENTHAL, JON M.

DOC DATE: 07/25/2000

ASSIGNOR:
LESKAR, PAUL A.

DOC DATE: 07/21/2000

ASSIGNEE:
SRC COMPUTERS, INC.
4240 N. NEVADA AVENUE
COLORADO SPRINGS, COLORADO 80907

SERIAL NUMBER: 09481902
PATENT NUMBER:

FILING DATE: 01/12/2000
ISSUE DATE:

RECEIVED
OCT 23 2000
HOGAN & HARTSON, LLP

MARCUS KIRK, EXAMINER
ASSIGNMENT DIVISION
OFFICE OF PUBLIC RECORDS

U.S. Department of Commerce
Patent and Trademark Office
PATENT

Correspondent Name and Address Area Code and Telephone Number **(303) 454-2454**

Name (line 1) **Carol W. Burton, Esq.**

Address (line 1) **Hogan & Hartson LLP**

Address (line 2) **1200 17th Street, Suite 1500**

Address (line 3) **Denver, CO 80202**

Address (line 4)

Pages Enter the total number of pages of the attached conveyance document including any attachments # **2**

Application Number(s) or Patent Number(s) ☐ Mark if additional numbers attached

Enter either the Patent Application Number or the Patent Number (DO NOT ENTER BOTH numbers for the same property).

Patent Application Number(s)			Patent Number(s)		
09/481,902					

If this document is being filed together with a new Patent Application, enter the date the patent application was signed by the first named executing inventor. Month Day Year

Patent Cooperation Treaty (PCT)

Enter PCT application number only if a U.S. Application Number has not been assigned

PCT ____ PCT ____ PCT ____

PCT ____ PCT ____ PCT ____

Number of Properties Enter the total number of properties involved #

Fee Amount Fee Amount for Properties Listed (37 CFR 3,41): $ **40.00**

Method of payment: Deposit Account Enclosed ☒ Deposit Account ☐

(Enter for payment by deposit account or if additional fees can be charged to the account.)

Deposit Account Number: # **50-1223**

Authorization to charge additional fees: Yes ☒ No ☐

Statement and Signature

To the best of my knowledge and belief, the foregoing information is true and correct and any attached copy is a true copy of the original document. Charges to deposit account are authorized, as indicated herein.

Carol W. Burton		**July 28, 2000**
Name of Person Signing	Signature	Date

\\\DE - 80409/4 - #78554 v1

Page 2

U.S. Department of Commerce
Patent and Trademark Office
PATENT

Correspondent Name and Address — Area Code and Telephone Number: (303) 454-2454

Name (line 1): Carol W. Burton, Esq.

Address (line 1): Hogan & Hartson LLP

Address (line 2): 1200 17th Street, Suite 1500

Address (line 3): Denver, CO 80202

Address (line 4):

Pages — Enter the total number of pages of the attached conveyance document including any attachments: # 2

Application Number(s) or Patent Number(s) — ☐ Mark if additional numbers attached

Enter either the Patent Application Number or the Patent Number (DO NOT ENTER BOTH numbers for the same property).

Patent Application Number(s)			Patent Number(s)		
09/481,902					

If this document is being filed together with a new Patent Application, enter the date the patent application was signed by the first named executing inventor. Month Day Year:

Patent Cooperation Treaty (PCT)

Enter PCT application number only if a U.S. Application Number has not been assigned

PCT ____ PCT ____ PCT ____
PCT ____ PCT ____ PCT ____

Number of Properties — Enter the total number of properties involved: #

Fee Amount — Fee Amount for Properties Listed (37 CFR 3,41): $ 40.00

Method of payment: Deposit Account — Enclosed ☒ Deposit Account ☐

(Enter for payment by deposit account or if additional fees can be charged to the account.)

Deposit Account Number: # 50-1223

Authorization to charge additional fees: Yes ☒ No ☐

Statement and Signature

To the best of my knowledge and belief, the foregoing information is true and correct and any attached copy is a true copy of the original document. Charges to deposit account are authorized, as indicated herein.

Carol W. Burton		July 28, 2000
Name of Person Signing	Signature	Date

\\\DE - 80409/4 - #78554 v1

Attorney Docket No. SRC001 CON
Client/Matter No. 80404.0004.001
Serial No. 09/481,902

ASSIGNMENT

WHEREAS, we, Jon M. Huppenthal, having a residence address of 10015 Burgess Road, Colorado Springs, CO 80908, and Paul A. Leskar, having a residence address of 755 Nebula Court, Colorado Springs, Colorado 80906, respectively, have made a certain new and useful invention relating to a MULTIPROCESSOR COMPUTER ARCHITECTURE INCORPORATING A PLURALITY OF MEMORY ALGORITHM PROCESSORS IN THE MEMORY SUBSYSTEM, for which we have made application for Letters Patent of the United States, said application being identified by Serial No. 09/481,902, Docket No. SRC001 CON and Client/Matter No. 80404.0004.001, in the law firm of HOGAN & HARTSON LLP, One Tabor Center, 1200 17th Street, Suite 1500, Denver, Colorado 80202;

WHEREAS, we now own the entire right, title and interest therein; and

WHEREAS, SRC Computers, Inc., hereinafter ASSIGNEE, a Colorado corporation, whose post office address is 4240 N. Nevada Ave. Colorado Springs, CO 80907, is desirous of acquiring the entire interest in and to said invention, said application, and the Letters Patent to be obtained therefor;

NOW THEREFORE, upon consideration of our present and/or previous employment under which the invention was made, and other good and valuable consideration, the adequacy of which is hereby acknowledged, we do hereby sell and assign unto ASSIGNEE, and ASSIGNEE's legal representatives, successors, and assigns, the entire right, title, and interest in and to said invention, said application, and the Letters Patent, both of the United States and of other countries, that may or shall issue thereon; and we do hereby authorize and request the Commissioner of Patents and Trademarks to issue said Letters Patent to ASSIGNEE, consistent with the terms of this Assignment.

UPON SAID CONSIDERATION, we do hereby covenant and agree with ASSIGNEE that we will not execute any writing or do any act whatsoever conflicting with this assignment, and that we will, at any time upon request,

without further or additional consideration, but at the expense of ASSIGNEE, execute such additional assignments and other writings and do such additional acts as ASSIGNEE may deem necessary or desirable to perfect the ASSIGNEE's enjoyment of this grant, and render all necessary assistance in applying for and obtaining original, continuation, continuation-in-part, divisional, reexamined, renewal, reissued or extended Letters Patent of the United States, or of other countries, or inventor's certificates, on said invention, and in enforcing any rights or causes in action accruing as a result of such applications, certificates, or patents, by giving testimony in any proceedings or transactions involving such applications, certificates, or patents, and by executing preliminary statements and other affidavits, it being understood that the foregoing covenant and agreement shall bind, and inure to the benefit of, the assigns and legal representatives of both parties.

IN WITNESS WHEREOF, we have hereunto set our hands on the date hereinafter set forth.

Date: 7/25/00

Jon M. Huppenthal

Date: 7/21/00

Paul A. Leskar

\\\DE - 80404/4 - #92665 v1

PATENT
Atty. Docket No. SRC001CON
Client Matter No. 80404.0004
Express Mail Label No. EL384120184US

In re Continuation Application of:

Jon H. Huppenthal and Paul A. Leskar

Serial No. 09/481,902

Filed: January 12, 2000

For: MULTIPROCESSOR COMPUTER ARCHITECTURE INCORPORATING A PLURALITY OF MEMORY ALGORITHM PROCESSORS IN THE MEMORY SUBSYSTEM

Examiner:

Art Unit: 2783

PRELIMINARY AMENDMENT

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

Please amend the above-identified patent application as follows:

IN THE CLAIMS:

Please add the following newly presented claims 47 through 65:

47. A method of processing data using reconfigurable processors, said method comprising:

configuring a first reconfigurable processor to perform a first function;

configuring a second reconfigurable processor to perform a second function;

writing a first data value to a first memory address location;

when said first memory address is accessed, performing said first function, using said first data value, and generating a second data value;

writing said second data value to a second memory address location; and

when said second memory address is accessed, performing said second function, using said second data value.

48. The method of claim 47 wherein performing said second function includes generating a third data value.

49. The method of claim 47 wherein performing said first function includes

multiplying.

50. The method of claim 49 wherein performing said multiplying executes in 64 clock cycles.

51. The method of claim 48 wherein configuring said first reconfigurable processor includes a fixed instruction set processor selecting confirguration bits corresponding to said first function.

52. The method of claim 51 further comprising:

said fixed instruction set processor performing a math function.

53. The method of claim 52 wherein said math function is a 64-bit floating point math function.

54. The method of claim 51 further comprising:

signaling said fixed instruction set processor when said third data value is available.

55. The method of claim 54 wherein said signaling includes writing a status value to a status register.

56. The method of claim 47 wherein writing said second data value includes operatively passing said second data value from said first reconfigurable function unit to said second reconfigurable function unit.

57. A system for processing data using reconfigurable processors, the system comprising:

a memory having an address space to allow access to a set of addressable memory locations;

a first reconfigurable processor coupled to a first address in said address space, responsive to a first data value being written at said first address, said first reconfigurable processor performs a first configured function and generates a second data value;

a second reconfigurable processor coupled to a second address in said address space, responsive to said second data value being written at said second address, said second reconfigurable processor performs a second configured function.

58. The system of claim 57 wherein said second reconfigurable processor generates a third data value.

59. The system of claim 57 wherein said first function is a multiplication function.

60. The system of claim 59 wherein performing said multiplication function

executes in 64 clock cycles.

61. The system of claim 58 further comprising:

a fixed instruction set processor coupled to said memory, programmed to select configuration bits that configure said first function in said first reconfigurable processor.

62. The system of claim 61 wherein said fixed instruction set processor is further programmed to perform a math function.

63. The system of claim 62 wherein said math function is a 64-bit floating point math function.

64. The system of claim 61 wherein said second reconfigurable processor signals said fixed instruction set processor when said third data value is available.

65. The system of claim 64 wherein said signal includes a status value written to a status register.

REMARKS

Entry of and consideration for allowance of claims 47 - 65 of the above-identified continuation application is respectfully requested.

Any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

23 March, 2000

William J. Kubida, Atty. Reg. No. 29,664
HOGAN & HARTSON LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5909 Tel

\\\CS - 80404/4 - #35240 v1

Attorney Docket No. SRC 001 CON
Client Matter No. 80404.0004.001

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Continuation Application of:	
Jon M. HUPPENTHAL, et al.	Art Unit:
Serial No.	Examiner:
Filed: January 12, 2000	
For: MULTIPROCESSOR COMPUTER ARCHITECTURE INCORPORATING A PLURALITY OF MEMORY ALGORITHM PROCESSORS IN THE MEMORY SUBSYSTEM	

PRELIMINARY AMENDMENT
ACCOMPANYING
CONTINUATION APPLICATION TRANSMITTAL

Box Patent Application
Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

Enclosed herewith is a true copy of co-pending U.S. Patent Application Serial No. 08/992,763 filed December 17, 1997 in support of the continuation Patent Application Transmittal filed herewith. Please amend the specification and claims as follows:

IN THE SPECIFICATION:

Page 1, line 5, in the line before "BACK GROUND OF THE INVENTION" insert

--RELATED APPLICATION

The present application is a continuation of co-pending U.S. Patent Application Serial No. 08/992,763 filed December 17, 1997, incorporated herein by referenced, which is assigned to the assignee of the present application.--

IN THE CLAIMS:

Please cancel claims 1-24.

Please add the following newly presented claims 25-46:

25. In a computer system having at least one data processor for executing an application program by operating on user data in accordance with application program instructions, said computer system having at least one memory bank with a data bus and an address bus connected to said at least one data processor, the improvement comprising:

a plurality of reconfigurable memory algorithm processors within individually addressable portions of said memory bank.

means connecting said plurality of memory algorithm processors to said data bus and to said address bus such that said plurality of memory algorithm processors are individually memory addressable by said at least one data processor at said one data processor executes said application program; and

said plurality of memory algorithm processors being configured as individual data processing elements to perform data processing related to said application program in accordance with an identified algorithm, said data processing being performed on at least one operand that is received directly from said at least one data processor

26. The improvement of claim 25 wherein each of said plurality of memory algorithm processors comprises a field programmable gate array.

27. The improvement of claim 25 wherein each of said plurality of memory algorithm processors is operative to memory address said memory bank independent of said at least one data processor.

28. The improvement of claim 25 wherein an identified algorithm is preprogrammed into each of said plurality of memory algorithm processors.

29. The improvement of claim 28 wherein a plurality of identified algorithms are preprogrammed into a memory device that is associated with said plurality of memory algorithm processors.

30. The improvement of claim 29 wherein said memory device comprises at least one read only memory device.

31. The improvement of claim 25 wherein any given one of said plurality of memory algorithm processors is operative to pass a data processing result of an operand that has been processed by an identified algorithm to another of said plurality of memory algorithm processors.

32. The improvement of claim 25 wherein said plurality of memory algorithm processors comprise a memory algorithm processor assembly, said memory algorithm processor assembly including:

a control block having a command decoder coupled to said address bus and having a pipeline counter coupled to said command decoder;

said command decoder for providing a last operand flag to said pipeline counter in response to a last operand command from an operating system of said at least one data processor.

33. The improvement of claim 32 wherein said control block further includes:

at least one status register; and

an equality comparator coupled to receive a pipeline depth signal and an output of said pipeline counter, said equality comparator for providing a pipeline empty flag to said at least one status register.

34. The improvement of claim 33 wherein said at least one status register is coupled to said command decoder to receive a register control signal and is coupled to said plurality of memory algorithm processors to receive a status signal, said at least one status register providing a status word output signal.

35. A multiprocessor computer system comprising:

A plurality of data processors for executing at least one application program by operating on user data in accordance with program instructions;

a memory bank having a data bus and an address bus connected to said plurality of data processors;

a plurality of reconfigurable memory algorithm processors within said memory bank at plurality of individual memory addressable memory locations;

means coupling said plurality of individual memory algorithm processors to said data bus and to said address bus;

said plurality of individual memory algorithm processors being individually memory addressable by all of said plurality of data processors; and

said plurality of memory algorithm processors being individually configurable to perform an identified algorithm on an operand that is received from a write operation by one of said plurality of data processors to said memory bank as said at least one of said plurality of data processors executes said at least one application program.

36. The multiprocessor computer system of claim 35 wherein all of said plurality of memory algorithm processors are memory addressable by all of said plurality of data processors.

37. The multiprocessor computer system of claim 36 wherein all of said plurality of memory algorithm processors are mutually memory addressable.

38. The multiprocessor computer system of claim 37 wherein said plurality of memory algorithm processors collectively comprises a memory algorithm processor assembly, said memory algorithm processor assembly including:

a control block operative to provide a last operand flag in response to a last operand having been processed by said memory algorithm processor assembly.

39. The multiprocessor computer system of claim 35 including:

at least one memory device associated with said plurality of memory algorithm processors for storing a plurality of pre loaded identified algorithms.

40. The multiprocessor computer system of claim 39 wherein said at least one memory device is responsive to a predetermined command from a data processor and operates in response thereto to selected one of said plurality of pre-loaded identified algorithms to be implemented by an addressed one of said plurality of memory algorithm processors.

41. The multiprocessor computer system of claim 40 wherein said at least one memory device comprises at least one read only memory device.

42. The multiprocessor computer system of claim 35 wherein each of said plurality of memory algorithm processors comprises a field programmable gate array.

43. The multiprocessor computer system of claim 35 wherein each of said plurality of memory algorithm processors is memory accessible through normal memory access protocol.

44. The multiprocessor computer system of claim 35 wherein each of said plurality of memory algorithm processors has direct memory access capability to said memory bank.

45. The multiprocessor computer system of claim 35 wherein each of said plurality of memory algorithm processors is operative to pass a result of a processed operand to another memory algorithm processor.

46. The multiprocessor computer system of claim 35 operative to detect at least one parallel region of said at least one application program, wherein at least one of said plurality of memory algorithm processors is configured as a function of said detected at least one parallel region of said at least one application program.

REMARKS

Please enter the following amendments in the above-referenced continuation application filed today prior to calculation of the filing fee due therefor. After cancellation of claims 1-24 and addition of claims 25-46, 22 total claims are pending of which 2 are independent claims.

Accordingly the small entity filing fee of $345.00 plus the fee for extra claims of $18.00 is calculated as totaling $363.00, for which a check in the amount of $363.00 is enclosed. Please charge any fee deficiency associated with this new application transmittal to Deposit Account No. 50-1123.

The correspondence address for the application is to be:

William J. Kubida, Esq.
Hogan & Hartson LLP
1200 17th Street, Suite 1500
Denver, Colorado 80202
Telephone (719) 448-5909
Facsimile (303) 899-7333

Respectfully submitted,

January 12, 2000

Carol W. Burton, Reg. No. 35,465
Hogan & Hartson LLP
1200 17th Street, Suite 1500
Denver, Colorado 80202
Telephone (303) 454-2454
Facsimile (303) 899-7333

EXHIBIT J

4a 9-11-99

PATENT

Attorney Docket No. SRC 001
Client/Matter No. 40055.830001.000
Express Mail No.EL415727189US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of)
JON M. HUPPENTHAL et al.)
Serial No. 08/992,763)
Filed: December 17, 1997)
For: MULTIPROCESSOR COMPUTER ARCHITECTURE INCORPORATING A PLURALITY OF MEMORY ALGORITHM PROCESSORS IN THE MEMORY SUBSYSTEM)

Group Art Unit: 2783

Examiner: J. Follansbee

AMENDMENT

Asst. Commissioner for Patents
Washington, D.C. 20231

Sir:

In response to the office communication mailed on June 3, 1999, by the Examiner in charge of the above-identified patent application, please amend the application as follows.

IN THE CLAIMS:

Please cancel claim 1-24 and substitute the following new claims 25-46.

--25. In a computer system having at least one data processor for executing an application program by operating on user data in accordance with application program instructions, said computer system having at least one memory bank with a data bus and an address bus

connected to said at least one data processor, the improvement comprising:

a plurality of memory algorithm processors within individually addressable portions of said memory bank;

means connecting said plurality of memory algorithm processors to said data bus and to said address bus such that said plurality of memory algorithm processors are individually memory addressable by said at least one data processor as said at least one data processor executes said application program; and

said plurality of memory algorithm processors being configured as individual data processing machines that can be memory addressed to perform data processing related to said application program in accordance with an identified algorithm, said data processing being performed on at least one operand that is received as a result of a write operation to said memory bank by said at least one data processor.

2 ~~26~~ The improvement of claim 1 ~~25~~ wherein each of said plurality of memory algorithm processors comprises a field programmable gate array.

3 ~~27~~. The improvement of claim 1 ~~25~~ wherein each of said plurality of memory algorithm processors is operative to memory address said memory bank independent of said at least one data processor.

4 ~~28~~. The improvement of claim 1 ~~25~~ wherein an identified algorithm is preprogrammed into each of said plurality of memory algorithm processors.

16

5 4
~~29~~. The improvement of claim ~~28~~ wherein a plurality of identified algorithms are preprogrammed into a memory device that is associated with said plurality of memory algorithm processors.

6 5
~~30~~. The improvement of claim ~~29~~ wherein said memory device comprises at least one read only memory device.

7 1
~~31~~. The improvement of claim ~~25~~ wherein any given one of said plurality of memory algorithm processors is operative to pass a data processing result of an operand that has been processed by an identified algorithm to another of said plurality of memory algorithm processors.

8 1
~~32~~. The improvement of claim ~~25~~ wherein said plurality of memory algorithm processors comprise a memory algorithm processor assembly, said memory algorithm processor assembly including:

a control block having a command decoder coupled to said address bus and having a pipeline counter coupled to said command decoder;

said command decoder for providing a last operand flag to said pipeline counter in response to a last operand command from an operating system of said at least one data processor.

9 8
~~33~~. The improvement of claim ~~32~~ wherein said control block further includes:

at least one status register; and

17

an equality comparator coupled to receive a pipeline depth signal and an output of said pipeline counter, said equality comparator for providing a pipeline empty flag to said at least one status register.

10 9
~~34~~. The improvement of claim ~~33~~ wherein said at least one status register is coupled to said command decoder to receive a register control signal and is coupled to said plurality of memory algorithm processors to receive a status signal, said at least one status register providing a status word output signal.

11
~~35~~. A multiprocessor computer system comprising:

a plurality of data processors for executing at least one application program by operating on user data in accordance with program instructions;

a memory bank having a data bus and an address bus connected to said plurality of data processors;

a plurality of memory algorithm processors within said memory bank at a plurality of individual memory addressable memory locations;

means coupling said plurality of individual memory algorithm processors to said data bus and to said address bus;

said plurality of individual memory algorithm processors being individually memory addressable by all of said plurality of data processors; and

said plurality of memory algorithm processors being individually configurable to perform an identified algorithm on an operand that is received from a write operation by said at least one data processor to

16

said memory bank as said at least one data processor executes said at least one application program.

12 ~~36~~. The multiprocessor computer system of claim 11 ~~35~~ wherein all of said plurality of memory algorithm processors are memory addressable by all of said plurality of data processors.

13 ~~37~~. The multiprocessor computer system of claim 12 ~~36~~ wherein all of said plurality of memory algorithm processors are mutually memory addressable.

14 ~~38~~. The multiprocessor computer system of claim 13 ~~37~~ wherein said plurality of memory algorithm processors collectively comprises a memory algorithm processor assembly, said memory algorithm processor assembly including:

a control block operative to provide a last operand flag in response to a last operand having been processed by said memory algorithm processor assembly.

15 ~~39~~. The multiprocessor computer system of claim 11 ~~35~~ including:

at least one memory device associated with said plurality of memory algorithm processors for storing a plurality of pre loaded identified algorithms.

16 ~~40~~. The multiprocessor computer system of claim 15 ~~39~~ wherein said at least one memory device is responsive to a predetermined command from a data processor and operates in response thereto to

19

selected one of said plurality of pre-loaded identified algorithms to be implemented by an addressed one of said plurality of memory algorithm processors.

17 ~~41~~. The multiprocessor computer system of claim 16 ~~40~~ wherein said at least one memory device comprises at least one read only memory device.

18 ~~42~~. The multiprocessor computer system of claim 11 ~~35~~ wherein each of said plurality of memory algorithm processors comprises a field programmable gate array.

19 ~~43~~. The multiprocessor computer system of claim 11 ~~35~~ wherein each of said plurality of memory algorithm processors is memory accessible through normal memory access protocol.

20 ~~44~~. The multiprocessor computer system of claim 11 ~~35~~ wherein each of said plurality of memory algorithm processors has direct memory access capability to said memory bank.

21 ~~45~~. The multiprocessor computer system of claim 11 ~~35~~ wherein each of said plurality of memory algorithm processors is operative to pass a result of a processed operand to another memory algorithm processor.

22 ~~46~~. The multiprocessor computer system of claim 11 ~~35~~ operative to detect at least one parallel region of said at least one application program, wherein at least one of said plurality of memory algorithm processors is configured as a function of said detected at least one parallel region of said at least one application program. --

REMARKS

The claims remaining in this application are claims 25-46.

Claims 1-24 have been canceled.

No claims are allowed.

Notice is taken of the NOTICE OF DRAFTSPERSON'S PATENT DRAWING REVIEW, and of the fact that the Examiner will require new, corrected drawings when necessary.

While it is believed that originally-submitted claims 1-24 patentably define over the prior art, the present Amendment provides claims that even more clearly distinguish over the Examiner's citations. Rather than extensively amending claims 1-24, new claims 25-46 are submitted for the convenience of the Examiner.

Summary of the Examiner's Rejection:

Paragraph 3 of the Examiner's communication rejects claims 1-6 and 9-11 (filed 17 December 1997) as unpatentable (35 USC 103a) over;

(1) U.S. Patent 5,892,962 to Coutier (filed 12 November 1996 and issued 6 April 1999), and

(2) OFFICIAL NOTICE that providing for data and address buses are well known and expected.

Paragraph 9 of the Examiner's communication rejects claims 7, 8 and 12-24 as unpatentable over the above mentioned Cloutier patent in view of admitted prior art contained at FIG. 1 and pages 1 and 2 of the present specification.

COMMENT: Notice is hereby given that all rights under 37 CFR 1.131 are reserved.

Summary of U.S. Patent 5,892,962 to Cloutier:

The Cloutier patent provides a multiprocessor 100 having a multi-dimensional torroidal array 102 of Field Programmable Gate Arrays (FPGAs) 104 (i.e., each FPGA 104 is connected to its four neighboring FPGAs 104).

As shown in FIG. 1, multiprocessor 100 includes a 2-dimensional array 102 of FPGAs 104, each FPGA 104 having its own local memory (SRAM)120.

Each FPGA 104 is either one-time programmable or reprogrammable. Each FPGA 104 receives program control data from an external process controller 108 via a program control bus 118, and receives other data from an external I/O controller 106 via a global bus 114. Thus, multiprocessor 100 can be programmed to perform a wide variety of applications (col. 2, lines 55-67).

FPGAs 104 are programmed to function as one or more Processing Elements (PEs), each FPGA 104 has its own local memory 120, and each FPGA 104 can be programmed to use the same local memory addresses as the other FPGAs 104 that are in array 102 (col. 2, lines 9-16).

Multiprocessor 100 can be programmed such that FPGAs 104 are used exclusively for computing operations, while process controller 108 is programmed exclusively for program control operations by decoding, wherein process controller 108 executes a program that is stored in instruction memory 112 and reads/writes to data memory 110.

Once a program instruction is decoded, process controller 108 sends data and control signals FPGAs 104 through global bus 114 and program control bus 118 (col. 3, lines 29-44).

The FIG. 1 multiprocessor 100 can be configured with one or more such multiprocessors 100 to form a large parallel processing architecture (col. 4, lines 12-14).

Multiprocessor 100 can be programmed on-line at the logic level with an optical character recognition algorithm, or other algorithms that include matrix multiplication, matrix addition, vector multiplication, vector addition, convolution, neural network forward pass, and neural network learning (col. 4, lines 26-33).

COMMENT: All that Cloutier provides is a control processor 108 that controls a number of FPGAs 104, which FPGAs 104 collectively operate as a data processor.

Note that Cloutier does not provide a computer system wherein;

(1) a data processor is addressable associated with a memory bank,

(2) wherein the data processor's memory bank contains memory algorithm processors that are memory addressable by the data processor --- that is, the data processor views the memory algorithm processors merely as part of the data processor's memory, and

(3) when multiple data processors are provided, the memory algorithm processors are shared by all of the data processors.

Summary of the Present Invention:

As shown in FIG. 2, the present invention provides multiprocessor computer architecture 100 that includes a plurality of PROCESSORS 108 and a MEMORY SPACE that is addressable by all of the PROCESSORS 108.

In FIG. 3 this MEMORY SPACE is shown as a MEMORY BANK 120 that can be accessed by all of the PROCESSORS 108 by way of SYSTEM TRUNK LINES 124, a data bus 126, and an address bus 128.

In accordance with the invention;

(1) The FIG. 2 MEMORY SPACE includes a plurality of Memory Algorithm Processors (MAPs) 112 (collectively shown as a MAP ASSEMBLY 112 in FIG. 3);

(2) Each of the MAPs 112 is viewed as an individually-addressable portion of the MEMORY SPACE by PROCESSORS 108 (i.e., MAPs 112 can be individually memory addressed by PROCESSORS 108); and

(3) Each of the MAPs 112 comprises a user-configurable hardware element that is globally accessible by all of the PROCESSORS 108.

By placing MAPs 112 in the MEMORY SPACE that is associated with the plurality of PROCESSORS 108, any given MAP 112 can be readily accessed by a any given PROCESSORS 108 merely through the use of a memory read operation or a memory write operation, thus allowing the use of a variety of operating systems within multiprocessor computer architecture 100.

Optionally, because all MAPs 112 can write to the MEMORY SPACE (i.e., MEMORY ARRAY 130 in FIG. 3) using direct memory access mechanisms, and because all MAPs 112 receive operands via writes to the MEMORY SPACE (i.e., MEMORY ARRAY 130 in FIG. 3), any given MAP 112 can feed the results of its algorithm computation to any other MAP 112.

FIG. 3 shows FIG. 2 MEMORY SPACE designated as MEMORY BANK 120. MEMORY BANK 120 is connected to FIG. 2's PROCESSORS 108 by way of SYSTEM TRUNK LINES 124.

In FIG. 3, the plurality of MAPs 112 shown in FIG. 2 are illustrated as a single MAP ASSEMBLY 112. In accordance with a preferred embodiment

of the invention, each individual one of the FIG. 2 MAPs 112 includes a USER FPGA (Field Programmable Gate Array) 134, wherein each USER FPGA 134 operates as a reconfigurable functional unit; i.e., operates as a reconfiguable hardware machine.

The plurality of USER FPGAs 134 (i.e., one for each of FIG. 2's MAPs 112) perform a plurality of user-defined algorithms with, and tightly coupled to, PROCESSORS 108. As stated above, all of the USER FPGAs 134 are globally addressable or accessible by all of the system PROCESSORS 108 for the selective execution of user-defined algorithms for which the USER FPGAs 134 are configured.

One or more ROMs 182 (see FIG. 4) are coupled to the plurality of USER FPGAs 134 within MAP ASSEMBLY 112, to allow a user program that is being executed by one or more of the PROCESSORS 108 to use a memory address operation (see ADDRESS BITS 128 of FIG. 4) to generate a command (see control signal 168 of FIG. 4) that operates to select one of a number of algorithms that are pre-loaded in ROM(s) 182.

Selection of a given algorithm that is within ROM(s) 182 by a given PROCESSOR 108 causes that given algorithm to be loaded into a given USER FPGA 134 that has been memory addressed by the given PROCESSOR 108. In this way, that given USER FPGA 134 is configured into a hardware machine that implements the given algorithm, thus reconfiguring the given USER FPGA 134 to operate in accordance with the given algorithm.

As shown in FIG. 3, MEMORY BANK 120 (i.e., FIG. 2 MEMORY SPACE) includes;

(1) MAP ASSEMBLY 112 corresponding to the plurality of MAPs 112 shown in FIG. 2;

wherein, MAP ASSEMBLY 112 includes,

(a) CONTROL 132, and

(b) a plurality of USER FPGAs 134, one USER FPGA 134 for each of the FIG. 2 MAPs 112;

(2) BANK CONTROL LOGIC 122 that is connected to PROCESSORS 108 by way of SYSTEM TRUNK LINES 124;

(3) MEMORY ARRAY 130;

(4) DATA bus 126 that interconnects BANK CONTROL LOGIC 122, MEMORY ARRAY 130, and CONTROL 132 that is within MAP ASSEMBLY 112; and

(5) ADDRESS bus 128 that interconnects BANK CONTROL LOGIC 122, MEMORY ARRAY 130, and CONTROL 132 that is within MAP ASSEMBLY 112.

MEMORY ARRAY 130 contains information that is needed by USER FPGAs 134 to enable USER FPGAs 134 to perform the algorithms for which USER FPGAs 134 are configured.

As shown in FIG. 4, CONTROL 132 within MAP ASSEMBLE 112 includes a COMMAND DECODER 150 that responds to ADDRESS BITS 128, and that operates to generate a control signal 168 that enables a given algorithm within ROM(s) 182 to be applied to a selected USER FPGA 134, thus causing the selected USER FPGA 134 to be reconfigured to execute the given algorithm, using an operand that is provided from a memory write operation to MEMORY BANK 120 by a PROCESSOR 108.

Argument for the Patentability of Claims 25-46:

Critical to the whole of the presently claimed invention are the features:

(1) Computer system 100 includes a plurality of data processors 108, one or more of which perform data processing operations on data 104 and instructions 106 in accordance with an application program that is being executed by one or more of data processors 108;

(2) Computer system 100 also includes a memory bank 120;

(3) Memory bank 120 includes a plurality of memory-resident memory algorithm processors 112, each of which includes a USER FPGA 134 --- that is memory algorithm processors 112 are a part of memory bank 120;

(4) Each of the processors 108 "views" each individual one of the plurality of USER FPGAs 134 merely as an individual memory addressable locations within memory bank 120;

(5) Each of the USER FPGAs 134 is configured to act as a data processing machine when it is memory addressed by a data processor 108;

(6) In this way, processors 108 can memory address any given USER FPGA 134, whereupon the addressed USER FPGA 134 operates as a secondary data processor to assist that processor 108 in the execution of the application program.

Using independent claim 25 as an example, the whole of this claim requires:

(1) At least one data processor 108 that executes an application program by operating on user data 104 in accordance with application program instructions 106;

(2) At least one memory bank 120 with a data bus 126 and an address bus 128 connected to the data processors 108;

(3) A plurality of memory algorithm processors 112 within individually addressable portions of memory bank 120, such that the memory algorithm processors 112 are individually memory addressable by data processors 108 as the application program is executed by data processors 108; and

(4) Memory algorithm processors 112 are configured as individual data processing machines that are memory addressable to perform data processing related to the application program in accordance with an identified algorithm, this data processing being performed on at least one operand that is received as a result of a write operation to memory bank 120 by a data processor 108.

With reference to the Cloutier patent, FPGAs 104 do not comprise portions of memory wherein a data processor utilizes FPGAs 104 merely by addressing a portion of memory.

Rather, in Cloutier, FPGAs 104 are programmed to function as processing elements, and **all** data processing occurs within the array 102 of FPGAs 104. That is, in Cloutier, array 102 comprises the only data processor that is programmed to execute a variety of application programs. Note that Cloutier teaches that FPGAs are used exclusively for computing operations, while process controller 108 is programmed exclusively for program control operations.

Clearly, Cloutier does not teach the presently claimed computing system wherein one or more main data processors 108 use a memory bank 120 to execute an application program, and wherein this same memory bank must include secondary data processing machines (in the form of MAPs 112 and FPGAs 134) that can be memory addressed by a main data processor 108 in

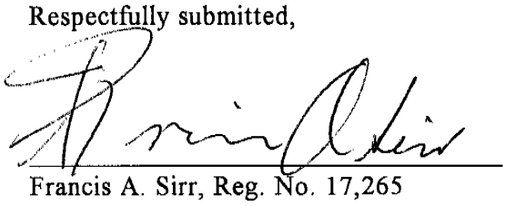
order to use the data processing power of these secondary data processing machines in the execution of the application program.

Request:

It is respectfully submitted that newly submitted claims 25-46 clearly define over the Examiner's citations, and a notice of allowance is respectfully requested.

Signed at Boulder, Colorado this 1 day of September, 1999.

Respectfully submitted,



Francis A. Sirr, Reg. No. 17,265
ATTORNEY FOR APPLICANT
HOLLAND & HART LLP
Post Office Box 8749
555 17th Street, suite 3200
Denver, Colorado 80201-8749
Telephone: (303) 473-2709

EXHIBIT K

PTO/SB/05 (08-08)
Approved for use through 06/30/2010. OMB 0651-0032

UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 CFR 1.53(b))

Attorney Docket No.	SRC036
First Inventor	Timothy J. Tewalt
Title	SYSTEM AND METHOD FOR RETAINING DRAM DATA WHEN REPROGRAMMING RECONFIGURABLE DEVICES WITH DRAM MEMORY CONTROLLERS
	EFS-Web

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPLICATION ELEMENTS

1. ☒ Fee Transmittal Form (e.g., PTO/SB/17) **(submit an original and a duplicate for fee processing)**
2. ☐ Applicant claims small entity status. See 37 CFR 1.27
3. ☒ Specification [total pages 20] Both the claims and abstract must start on a new page (For information on the preferred arrangement, see MPEP 608.01(a))
4. ☒ Drawing(s) [total sheets 2]
5. ☒ Oath or Declaration [total pages 1]
   a. ☒ Newly executed (original or copy)
   b. ☐ Copy from prior appl. (37 C.F.R. § 1.63(d)) (for continuation/divisional with Box 18 completed)
      i. ☐ DELETION OF INVENTOR(S) Signed statement attached deleting inventor(s) name in the prior application, see 37 C.F.R. 1.63(d)(2) and1.33(b).
6. ☒ Application Data Sheet. (See 37 CFR 1.76)
7. ☐ CD-ROM or CD-R in duplicate, large table or Computer Program (Appendix)
   ☐ Landscape Table on CD
8. Nucleotide and/or Amino Acid Sequence Submission (if *applicable, all necessary*)
   a. ☐ Computer Readable Form (CRF)
   b. ☐ Specification Sequence Listing on:
      i. ☐ CD-ROM or CD-R (2 copies); or
      ii. ☐ paper
   c. ☐ Statements verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

9. ☒ Assignment Papers (coversheet (PTO-1595) & document(s))
   Name of Assignee: **SRC Computers, LLC.**
10. ☐ 37 CFR. 3.73(b) Statement ☒ Power of Attorney (when there is an assignee)
11. ☐ English Translation Document (if applicable)
12. ☒ Information Disclosure Statement (PTO/SB/08 or PTO-1449)
    ☒ Copies of foreign patent documents, publications and other information
13. ☐ Preliminary Amendment
14. ☐ Return Receipt Postcard (MPEP 503) (Should be specifically itemized)
15. ☐ Certified Copy of Priority Document(s) (if foreign priority is claimed)
16. ☐ Nonpublication Request Under 35 U.S.C. 122(b)(2)(B)(i).Applicant must attach form PTO/SB/35
17. ☐ Other: Certificate of Mailing by Express Mail

18. **If a CONTINUING APPLICATION, check appropriate box, and supply the requisite information below and in the first sentence of the specification following the title, or in an Application Data Sheet under 37 CFR 1.76:**

☐ Continuation ☐ Divisional ☐ Continuation-in-part (CIP) of prior application No.: __/_____

Prior application information: *Examiner:* _____ *Art Unit:* _____

19. CORRESPONDENCE ADDRESS

☒ The address associated with Customer Number **25235** or ☐ Correspondence address below

Name					
Address					
City		State		ZIP	
Country		Telephone		Email	

Name (Print/Type)	**Peter Meza**	Registration No.	**32,920**
(Signature)	Peter Meza	Date	5/27/2014

SCS - 080404/000045 - 206353 v1

PATENT
Attorney Docket No. SRC036
Client No. 80404.0045
EFS-Web

SYSTEM AND METHOD FOR RETAINING DRAM DATA WHEN REPROGRAMMING RECONFIGURABLE DEVICES WITH DRAM MEMORY CONTROLLERS

BACKGROUND OF THE INVENTION

The present invention relates, in general, to the field of reconfigurable computing systems. More particularly, the present invention relates to a system and method for retaining dynamic random access memory (DRAM) data when reprogramming reconfigurable devices with DRAM memory controllers.

The majority of today's programmable logic designs include a DRAM based memory solution at the heart of their memory subsystem. Today's DRAM devices are significantly faster than previous generation's, albeit at the cost of requiring increasingly complex and resource intensive memory controllers. One example is in double data rate 3 and 4 (DDR3 and DDR4) controllers which require read and write calibration logic. This added logic was not necessary when using previous versions of DRAM (e.g. DDR and DDR2. As a result, companies are forced to absorb substantial design costs and increased project completion times when designing proprietary DRAM controllers utilizing modern DRAM technology.

In order to mitigate design engineering costs and verification time, it is very common for field programmable gate array (FPGA) designers to implement vendor provided memory controller intellectual property (IP) when including DRAM based memory solutions in their designs. See, for example, Allan, Graham; "DDR IP

Integration: How to Avoid Landmines in this Quickly Changing Landscape"; Chip Design, June/July 2007; pp 20-22 and Wilson, Ron; "DRAM Controllers for System Designers"; Altera Corporation Articles, 2012.

FPGA designers tend to choose device manufacturer IP designs because they are proven, tested and have the incredible benefit of significantly reduced design costs and project completion times. Many times there is the added benefit of exploiting specialized circuitry within the programmable device to increase controller performance, which is not always readily apparent when designing a controller from scratch.

The downside to using factory supplied IP memory controllers is that there is little flexibility when trying to modify operating characteristics. A significant problem arises in reconfigurable computing when the FPGA is reprogrammed during a live application and the memory controller tri-states all inputs and outputs (I/O) between the FPGA device and the DRAM. The result is corrupted data in the memory subsystem. Therefore, dynamically reconfigurable processors are excluded as viable computing options, especially in regard to database applications or context switch processing. The reason for this is that the time it takes to copy the entire contents of DRAM data and preserve it in another part of the system, reconfigure the processor, then finally retrieve the data and restore it in DRAM is just too excessive.

SUMMARY OF THE INVENTION

Disclosed herein is a system and method for preserving DRAM memory contents when a reconfigurable

device, for example an FPGA having a DRAM memory controller, is reconfigured, reprogrammed or otherwise powered down. When an FPGA is reprogrammed, the DRAM inputs are tri-stated including self-refresh command signals. Indeterminate states on the reset or clock enable inputs results in DRAM data corruption.

In accordance with the system and method of the present invention, an FPGA based DRAM controller is utilized in concert with an internally or externally located data maintenance block. In operation, the FPGA drives the majority of the DRAM input/output (I/O) and the data maintenance block drives the self-refresh command inputs. Even though the FPGA reconfigures and the majority of the DRAM inputs are tri-stated, the data maintenance block provides stable input levels on the self-refresh command inputs.

Functionally, the data maintenance block does not contain the memory controller and therefore has no point of reference for when and how to initiate the self-refresh commands, particularly the DRAM self-refresh mode. As also disclosed herein, a communication port is implemented between the FPGA and the data maintenance block that allows the memory controller in the FPGA to direct the self-refresh commands to the DRAM via the data maintenance block. Specifically, this entails when to put the DRAM into self-refresh mode and preserve the data in memory.

At this point, the DRAM data has been preserved throughout the FPGA reconfiguration via the self-refresh mode initiated by the data maintenance block, but the DRAM controller must now re-establish write/read timing

windows and will corrupt specific address contents with guaranteed write and read data required during the calibration/leveling process. Consequently, using the self-refresh capability of DRAM alone is not adequate for maintaining data integrity during reconfiguration. (It should be noted that the memory addresses used during calibration/leveling are known and typically detailed in the controller IP specification).

In order to effectuate this, the system transmits a "reconfiguration request" to the DRAM controller. Once received, glue logic surrounding the FPGA vendor provided memory controller IP issues read requests to the controller specifying address locations used during the calibration/leveling process. As data is retrieved from the DRAM, it is transmitted via the communication port from the FPGA device to a block of storage space residing within the data maintenance block itself or another location in the system.

Once the process is complete, the data maintenance block sends a self-refresh command to the DRAM and transmits an acknowledge signal back to the FPGA. The data maintenance block recognizes this as an FPGA reconfiguration condition versus an FPGA initial power up condition and retains this state for later use.

Once the FPGA has been reprogrammed, the DRAM controller has re-established calibration settings and several specific addresses in the DRAM have been corrupted with guaranteed write/read data patterns. At this point, glue logic surrounding the vendor memory controller IP is advised by the data maintenance block (through the communication port) that it has awakened

from either an initial power up condition or a reconfiguration condition. If a reconfiguration condition is detected, and before processing incoming DMA requests, the controller retrieves stored DRAM data from the data maintenance block (again through the communication port) and writes it back to the specific address locations corrupted during the calibration/leveling process. Once complete, the DRAM controller in the FPGA is free to begin servicing system memory requests in the traditional fashion.

Among the benefits provided in conjunction with the system and method of the present invention is that since the data maintenance block functions to hold the DRAM in self-refresh mode, the FPGA is free to be reprogrammed to perform a very application-specific computing job that may not require DRAM. This means all the device resources previously reserved for creating a DRAM controller are now free to be used for different functions.

Further, the overall computer system benefits from the present invention because data previously stored in DRAM has now been preserved and is available for use by the next application that needs it. This leads to the fact that computing solutions requiring a series of specific data manipulation tasks now have the ability to be implemented in a small reconfigurable processor. Each application performs its intended function and data is passed from application to application between reconfiguration periods via the DRAM.

Importantly, it should also be noted that the DRAM data contents are retained even if the reconfigurable

device is powered down. This is especially critical, for example, when the system and method of the present invention is implemented in mobile devices.

Particularly disclosed herein is a system and method for preserving DRAM data contents when reconfiguring a device containing one or more DRAM controllers. Also particularly disclosed herein is a system and method for preserving DRAM data contents in a reconfigurable computing environment when the programmable device is reconfigured with a new design that does not include a DRAM controller. Further disclosed herein is a system and method for passing DRAM data between sequential computing tasks in a reconfigurable computing environment as well as system and method for preserving DRAM contents when the reconfigurable device is powered down.

Also particularly disclosed herein is a computer system which comprises a DRAM memory, a reconfigurable logic device having a memory controller coupled to selected inputs and outputs of the DRAM memory and a data maintenance block coupled to the reconfigurable logic device and self-refresh command inputs of the DRAM memory. The data maintenance block is operative to provide stable input levels on the self-refresh command inputs while the reconfigurable logic device is reconfigured.

Still further particularly disclosed herein is a method for preserving the contents of a DRAM memory associated with a reconfigurable device having a memory controller. The method comprises providing a data maintenance block coupled to the reconfigurable device,

coupling the data maintenance block to self-refresh command inputs of the DRAM memory, storing data received from the reconfigurable device at the data maintenance block and maintaining stable input levels on the self-refresh command inputs while the reconfigurable logic device is reconfigured.

BRIEF DESCRIPTION OF THE DRAWINGS

The aforementioned and other features and objects of the present invention and the manner of attaining them will become more apparent and the invention itself will be best understood by reference to the following description of a preferred embodiment taken in conjunction with the accompanying drawings, wherein:

Fig. 1 is a functional block diagram of a computer subsystem comprising a reconfigurable logic device having a reconfigurable DRAM controller with associated DRAM memory and illustrating the data maintenance block of the present invention for retaining DRAM data when the logic device is reconfigured; and

Fig. 2 is a block diagram of a reconfigurable computer system, such as that available from SRC Computers, LLC, assignee of the present invention, incorporating a pair of data maintenance blocks and DRAM memory in accordance with the system and method of the present invention in association with reconfigurable application logic.

DESCRIPTION OF A REPRESENTATIVE EMBODIMENT

With reference now to Fig. 1, a functional block diagram of a computer subsystem 100 comprising a DRAM memory 102 and reconfigurable logic device 104 is shown.

In a representative embodiment of the present invention, the reconfigurable logic device 104 may comprise a field programmable gate array (FPGA). However, it should be noted that the reconfigurable logic device 104 may comprise any and all forms of reconfigurable logic devices including hybrid devices, such as a reconfigurable logic device with partial reconfiguration capabilities or an application specific integrated circuit (ASIC) device with reprogrammable regions contained within the chip.

Also illustrated is a data maintenance block 106 in accordance with the present invention for retaining DRAM memory 102 data when the logic device 104 is reconfigured during operation of the computer subsystem 100. In a representative embodiment of the present invention, the data maintenance block 106 may be conveniently provided as a complex programmable logic device (CPLD) or other separate integrated circuit device or, in alternative embodiments, may be provided as a portion of an FPGA comprising the reconfigurable logic device 104.

As illustrated, the reconfigurable logic device 104 comprises a primary system logic block 108 which issues a reconfigure request command to a reconfigure controller 110 and receives a reconfigure request acknowledgement (Ack) signal in return. The reconfigure controller 110, in turn, issues a command to the command decode block 112 of the data maintenance block 106 and receives an acknowledgement (Ack) signal in return. A block RAM portion 114 of the data maintenance block 106 exchanges data with the reconfigure controller 110.

The reconfigure controller 110 receives an input from a refresh timer 116 which is coupled to receive row address select (RAS#), column address select (CAS#) and write enable (WE#) signals from a memory controller and physical interface block 118. The memory controller and physical interface block 118 also provides the RAS#, CAS# and WE# signals to the DRAM memory 102 as well as clock (CK, CK#), chip select (CS#), address (A), bank address (BA), data mask (DM) and on-die termination (ODT) input signals. Bididrectional data (DQ) input/output (I/O) and differential data strobe signals (DQS/DQS#) are exchanged between the DRAM memory 102 and the memory controller and physical interface block 118 as shown. The data maintenance block 106 is coupled to the DRAM memory 102 to supply reset (RESET#) and clock enable (CKE#) signals thereto.

The memory controller and physical interface block 118 responds to a request from the controller interface 120 to provide data read from the DRAM memory 102 (Rd Data) and to receive data to be written to the DRAM memory 102 (Wr Data) as shown. A source logic block 122 is coupled to the controller interface 120 as well as the reconfigure controller 110 as also illustrated. The source logic block 122 receives a data request from the primary system logic block 108 and supplies data read from the DRAM memory 102 while receiving data to be written thereto.

As indicated by the operation at numeral 1, a reconfiguration request is received at the reconfigure controller 110 from the primary system logic block 108 of the reconfigurable logic device 104. The reconfigure

controller 110 initiates direct memory access (DMA) read requests to memory addresses used in a calibration/leveling sequence after the reconfigurable logic device 104 is reconfigured. Returned data is stored in a small section of block RAM (not shown) in the reconfigure controller 110.

As indicated by the operation at numeral 2, the reconfigure Controller 110 stores its block RAM contents in another small section of block RAM 114 located in the data maintenance block 106. When complete, the data maintenance block 106 asserts an acknowledge signal from its command decode block 112. At the operation indicated by numeral 3, the reconfigure controller 110 detects a refresh command from the refresh timer 116, waits a refresh cycle time (t_{RFC}) and instructs the data maintenance block 106 to de-assert CKE to the DRAM memory 102.

The reconfigure controller 110 asserts the Reconfigure Request Ack signal at the operation indicated by numeral 4 and the reconfigurable logic device 104 is reconfigured. As indicated by the operation at numeral 5, the reconfigure controller 110 recognizes a post-reconfigure condition (Ack = High), holds the memory controller and physical interface 118 in reset and instructs the data maintenance block 106 to assert CKE to the DRAM memory 102. The memory controller and physical interface 118 is then released from reset and initializes the DRAM memory 102.

At the operation indicated by numeral 6, the reconfigure controller 110 retrieves the data maintenance block 106 block RAM 114 contents and stores

it in a small section of block RAM (not shown) in the reconfigure controller 110. The reconfigure controller 110 detects that the memory controller and physical interface 118 and DRAM memory 102 initialization is complete at the operation indicated by numeral 7 and initiates DMA write requests to restore the memory contents corrupted during the calibration/leveling sequence with the data values read prior to reconfiguration. At the operation indicated by numeral 8, the memory controller and physical interface 118 glue logic (comprising reconfigure controller 110, refresh timer 116, controller interface 120 and source logic block 122) resumes DMA activity with the primary system logic 108 in a conventional fashion.

It should be noted certain of the aforementioned operational steps may, in fact, operate substantially concurrently. Further, while functionally accurate, some of the operational steps enumerated have been listed out of order to provide logical continuity to the overall operation and to facilitate comprehensibility of the process. In a particular implementation of the system and method of the present invention, one or more of the operational steps disclosed may be conveniently re-ordered to increase overall hardware efficiency. Moreover, steps which can serve to facilitate relatively seamless integration in an active application can be provided in addition to those described as may be desired.

With reference additionally now to Fig. 2, a block diagram of a reconfigurable computer system 200 is illustrated incorporating a pair of data maintenance

blocks 106 and DRAM memory 102 in accordance with the system and method of the present invention in association with reconfigurable application logic 202. In this representative embodiment of a reconfigurable computer system 200, the DRAM memory 102 is illustrated in the form of 32GB error correction code (ECC) synchronous dynamic random access memory (SDRAM).

The reconfigurable application logic 202 is coupled to the data maintenance blocks 106 and DRAM memory 102 as depicted and described previously with respect to the preceding figure and is also illustrated as being coupled to a number of 8GB ECC static random access memory (SRAM) memory modules 204. The reconfigurable application logic 202 is also coupled to an SRC Computers, LLC SNAP™ and network processors block 206 having a number of serial gigabit media independent interface (SGMII) links as shown. It should be noted that the DRAM memory 102 controller in the reconfigurable application block 202 may be omitted upon subsequent reconfigurations as the DRAM memory 102 data contents will be maintained in the data maintenance blocks 106.

The SNAP and network processors block 206 shares equal read/write access to a 1GB peer SDRAM system memory 208 along with a microprocessor subsystem 210. The microprocessor subsystem 210, as illustrated, also comprises an SGMII link as well as a pair of serial advanced technology attachment (SATA) interfaces.

For continuity and clarity of the description herein, the term "FPGA" has been used in conjunction with the representative embodiment of the system and

method of the present invention and refers to just one type of reconfigurable logic device. However, it should be noted that the concept disclosed herein is applicable to any and all forms of reconfigurable logic devices including hybrid devices, inclusive of reconfigurable logic devices with partial reconfiguration capabilities or an ASIC device with reprogrammable regions contained within the chip.

Representative embodiments of dynamically reconfigurable computing systems incorporating the DRAM memory 102, reconfigurable logic device 104, associated microprocessors and programming techniques are disclosed in one or more of the following United States Patents and United States Patent Publications to SRC Computers LLC, assignee of the present invention, the disclosures of which are herein specifically incorporated by this reference in their entirety: U.S. Pat. No. 6,026,459; U.S. Pat. No. 6,076,152; U.S. Pat. No. 6,247,110; U.S. Pat. No. 6,295,598; U.S. Pat. No. 6,339,819; U.S. Pat. No. 6,356,983; U.S. Pat. No. 6,434,687; U.S. Pat. No. 6,594,736; U.S. Pat. No. 6,836,823; U.S. Pat. No. 6,941,539; U.S. Pat. No. 6,961,841; U.S. Pat. No. 6,964,029; U.S. Pat. No. 6,983,456; U.S. Pat. No. 6,996,656; U.S. Pat. No. 7,003,593; U.S. Pat. No. 7,124,211; U.S. Pat. No. 7,134,120; U.S. Pat. No. 7,149,867; U.S. Pat. No. 7,155,602; U.S. Pat. No. 7,155,708; U.S. Pat. No. 7,167,976; U.S. Pat. No. 7,197,575; U.S. Pat. No. 7,225,324; U.S. Pat. No. 7,237,091; U.S. Pat. No. 7,299,458; U.S. Pat. No. 7,373,440; U.S. Pat. No. 7,406,573; U.S. Pat. No. 7,421,524; U.S. Pat. No. 7,424,552; U.S. Pat. No.

7,565,461; U.S. Pat. No. 7,620,800; U.S. Pat. No. 7,680,968; U.S. Pat. No. 7,703,085; U.S. Pat. No. 7,890,686; U.S. Pat. No. 8,589,666; U.S. Pat. Pub. No. 2012/0117318; U.S. Pat. Pub. No. 2012/0117535; and U.S. Pat. Pub. No. 2013/0157639.

While there have been described above the principles of the present invention in conjunction with specific apparatus and methods, it is to be clearly understood that the foregoing description is made only by way of example and not as a limitation to the scope of the invention. Particularly, it is recognized that the teachings of the foregoing disclosure will suggest other modifications to those persons skilled in the relevant art. Such modifications may involve other features which are already known per se and which may be used instead of or in addition to features already described herein. Although claims have been formulated in this application to particular combinations of features, it should be understood that the scope of the disclosure herein also includes any novel feature or any novel combination of features disclosed either explicitly or implicitly or any generalization or modification thereof which would be apparent to persons skilled in the relevant art, whether or not such relates to the same invention as presently claimed in any claim and whether or not it mitigates any or all of the same technical problems as confronted by the present invention. The applicants hereby reserve the right to formulate new claims to such features and/or combinations of such features during the prosecution of

the present application or of any further application derived therefrom.

As used herein, the terms "comprises", "comprising", or any other variation thereof, are intended to cover a non-exclusive inclusion, such that a process, method, article, or apparatus that comprises a recitation of certain elements does not necessarily include only those elements but may include other elements not expressly recited or inherent to such process, method, article or apparatus. None of the description in the present application should be read as implying that any particular element, step, or function is an essential element which must be included in the claim scope and THE SCOPE OF THE PATENTED SUBJECT MATTER IS DEFINED ONLY BY THE CLAIMS AS ALLOWED. Moreover, none of the appended claims are intended to invoke paragraph six of 35 U.S.C. Sect. 112 unless the exact phrase "means for" is employed and is followed by a participle.

What is claimed is:

CLAIMS:

1. A computer system comprising:

a DRAM memory;

a reconfigurable logic device having a memory controller coupled to selected inputs and outputs of said DRAM memory; and

a data maintenance block coupled to said reconfigurable logic device and self-refresh command inputs of said DRAM memory, said data maintenance block operative to provide stable input levels on said self-refresh command inputs while said reconfigurable logic device is reconfigured.

2. The computer system of claim 1 wherein said DRAM memory comprises DDR3 compliant memory devices.

3. The computer system of claim 1 wherein said reconfigurable logic device comprises an FPGA.

4. The computer system of claim 1 wherein said data maintenance block comprises a command decode portion coupled to a reconfigure controller of said reconfigurable logic device.

5. The computer system of claim 4 wherein said command decode portion of said data maintenance block is operative in response to a command from said reconfigure controller and provides an acknowledgement signal in response.

6. The computer system of claim 1 wherein said data maintenance block comprises a memory block coupled to a

reconfigure controller of said reconfigurable logic device.

7. The computer system of claim 6 wherein said memory block is operative to retain data received from said reconfigure controller of said reconfigurable logic device.

8. The computer system of claim 1 wherein said data maintenance block comprises a CPLD.

9. The computer system of claim 1 wherein said reconfigurable logic device comprises said data maintenance block.

10. The computer system of claim 1 wherein said data maintenance block is operable to hold said DRAM memory in self-refresh mode while said reconfigurable logic device is reconfigured.

11. A method for preserving contents of a DRAM memory associated with a reconfigurable device having a memory controller comprising:

providing a data maintenance block coupled to said reconfigurable device;

coupling said data maintenance block to self-refresh command inputs of said DRAM memory;

storing data received from said reconfigurable device at said data maintenance block; and

maintaining stable input levels on said self-refresh command inputs while said reconfigurable logic device is reconfigured.

12. The method of claim 11 wherein said step of providing comprises:

providing a command decode portion of said data maintenance block coupled to receive commands from said reconfigurable device and return acknowledgment signals in response thereto.

13. The method of claim 11 wherein said step of storing comprises:

providing a memory block in said data maintenance block for storing said data received from said reconfigurable device and returning said data to said reconfigurable device upon completion of a reconfiguration function.

14. The method of claim 11 wherein said step of storing comprises:

providing a memory block in said data maintenance block for storing said data received directly from said DRAM memory and returning said data directly to said DRAM memory upon completion of a reconfiguration function.

15. The method of claim 11 wherein said step of providing said data maintenance block comprises:

providing a portion of said reconfigurable device as said data maintenance block.

16. The method of claim 11 wherein said step of providing said data maintenance block comprises:

providing a CPLD as said data maintenance block.

17. The method of claim 11 wherein said step of providing said data maintenance block comprises:

providing a block RAM for storing said data received from said reconfigurable device; and

providing a command decode portion responsive to said reconfigurable device and coupled to said reset and lock enable inputs of said DRAM memory.

18. The method of claim 11 further comprising:

passing said data between sequential computing tasks in a reconfigurable computing environment.

19. The method of claim 11 further comprising:

preserving said data at said data maintenance block while said reconfigurable logic device is powered down.

ABSTRACT OF THE DISCLOSURE

A system and method for retaining dynamic random access memory (DRAM) data when reprogramming reconfigurable devices with DRAM memory controllers such as field programmable gate arrays (FPGAs). The DRAM memory controller is utilized in concert with an internally or externally located data maintenance block wherein the FPGA drives the majority of the DRAM input/output (I/O) and the data maintenance block drives the self-refresh command inputs. Even though the FPGA reconfigures and the majority of the DRAM inputs are tri-stated, the data maintenance block provides stable input levels on the self-refresh command inputs.

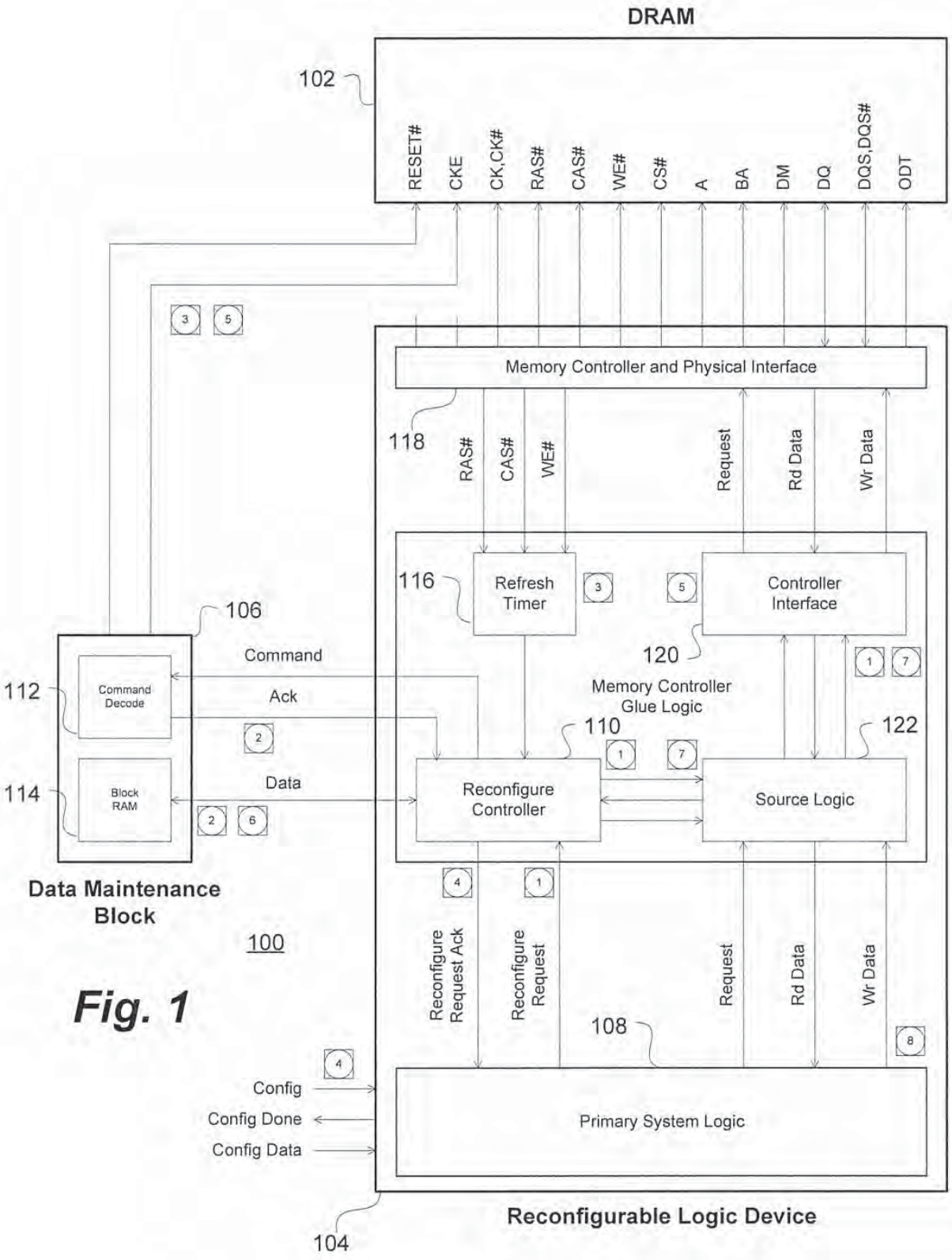

DRAM
102
RESET#
CKE
CK,CK#
RAS#
CAS#
WE#
CS#
A
BA
DM
DQ
DQS,DQS#
ODT
Memory Controller and Physical Interface
118
RAS#
CAS#
WE#
Request
Rd Data
Wr Data
116
Refresh Timer
Controller Interface
120
Memory Controller Glue Logic
110
122
Reconfigure Controller
Source Logic
106
Command
Ack
Data
112
Command Decode
114
Block RAM
Data Maintenance Block
100
Reconfigure Request Ack
Reconfigure Request
Request
Rd Data
Wr Data
108
Config
Config Done
Config Data
Primary System Logic
Reconfigurable Logic Device
104

Fig. 1

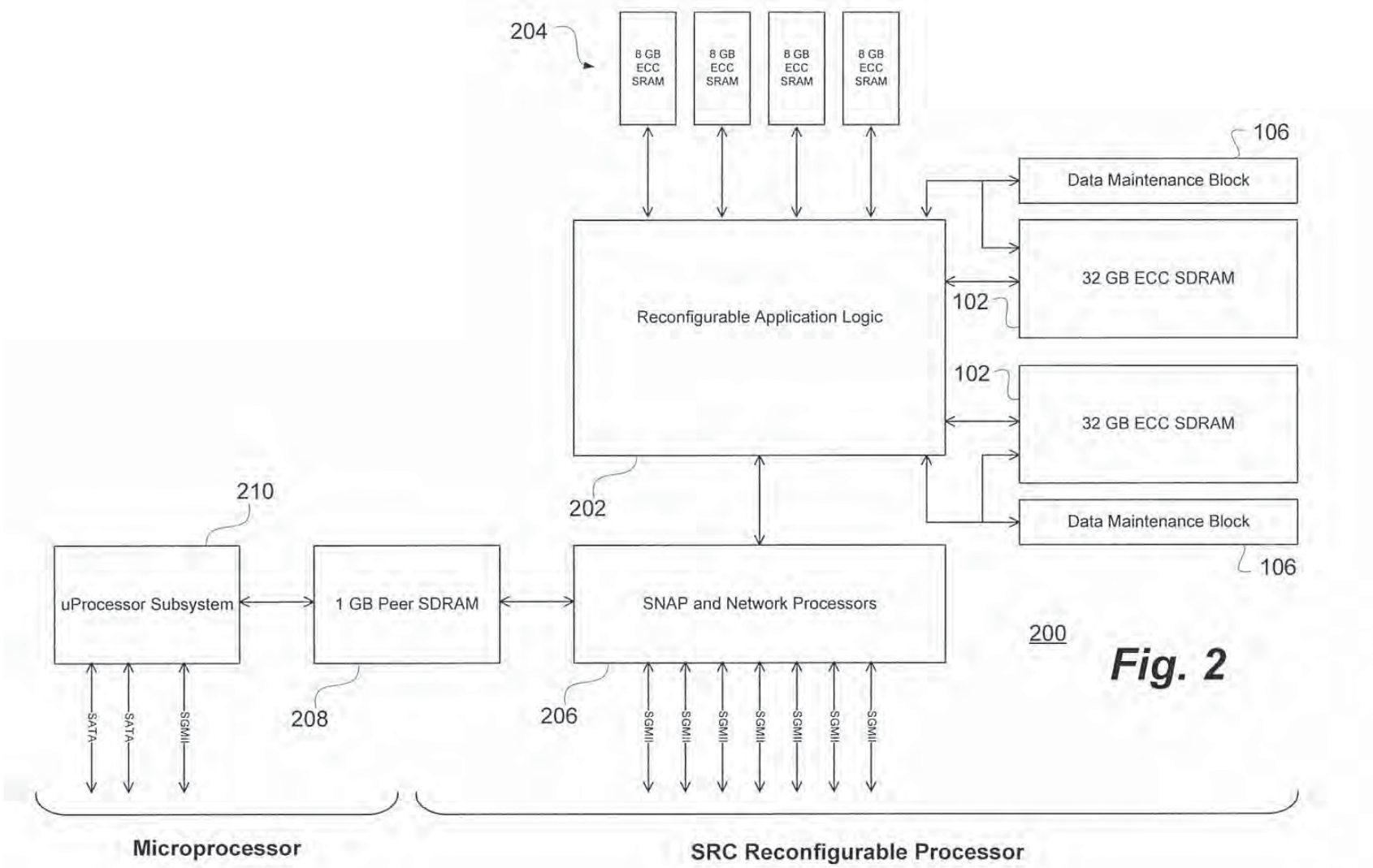


Fig. 2

SRC00001792

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

NOTICE OF ALLOWANCE AND FEE(S) DUE

25235 7590 06/09/2015

HOGAN LOVELLS US LLP
TWO NORTH CASCADE AVENUE
SUITE 1300
COLORADO SPRINGS, CO 80903

EXAMINER
HO, HOAI V

ART UNIT	PAPER NUMBER
2827	

DATE MAILED: 06/09/2015

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
14/288,094	05/27/2014	Timothy J. Tewalt	SRC036	9362

TITLE OF INVENTION: SYSTEM AND METHOD FOR RETAINING DRAM DATA WHEN REPROGRAMMING RECONFIGURABLE DEVICES WITH DRAM MEMORY CONTROLLERS

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	$480	$0	$0	$480	09/09/2015

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE DOES NOT REFLECT A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE IN THIS APPLICATION. IF AN ISSUE FEE HAS PREVIOUSLY BEEN PAID IN THIS APPLICATION (AS SHOWN ABOVE), THE RETURN OF PART B OF THIS FORM WILL BE CONSIDERED A REQUEST TO REAPPLY THE PREVIOUSLY PAID ISSUE FEE TOWARD THE ISSUE FEE NOW DUE.

HOW TO REPLY TO THIS NOTICE:

I. Review the ENTITY STATUS shown above. If the ENTITY STATUS is shown as SMALL or MICRO, verify whether entitlement to that entity status still applies.

If the ENTITY STATUS is the same as shown above, pay the TOTAL FEE(S) DUE shown above.

If the ENTITY STATUS is changed from that shown above, on PART B - FEE(S) TRANSMITTAL, complete section number 5 titled "Change in Entity Status (from status indicated above)".

For purposes of this notice, small entity fees are 1/2 the amount of undiscounted fees, and micro entity fees are 1/2 the amount of small entity fees.

II. PART B - FEE(S) TRANSMITTAL, or its equivalent, must be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed and an extra copy of the form should be submitted. If an equivalent of Part B is filed, a request to reapply a previously paid issue fee must be clearly made, and delays in processing may occur due to the difficulty in recognizing the paper as an equivalent of Part B.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Mail Stop ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

PTOL-85 (Rev. 02/11)

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), to: Mail **Mail Stop ISSUE FEE**
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450
or Fax (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

25235 7590 06/09/2015
HOGAN LOVELLS US LLP
TWO NORTH CASCADE AVENUE
SUITE 1300
COLORADO SPRINGS, CO 80903

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission
I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

(Depositor's name)
(Signature)
(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
14/288,094	05/27/2014	Timothy J. Tewalt	SRC036	9362

TITLE OF INVENTION: SYSTEM AND METHOD FOR RETAINING DRAM DATA WHEN REPROGRAMMING RECONFIGURABLE DEVICES WITH DRAM MEMORY CONTROLLERS

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	$480	$0	$0	$480	09/09/2015

EXAMINER	ART UNIT	CLASS-SUBCLASS
HO, HOAI V	2827	365-222000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).
- ☐ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.
- ☐ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list
(1) The names of up to 3 registered patent attorneys or agents OR, alternatively.
(2) The name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.
1 _____
2 _____
3 _____

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE (B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent) : ☐ Individual ☐ Corporation or other private group entity ☐ Government

4a. The following fee(s) are submitted:
- ☐ Issue Fee
- ☐ Publication Fee (No small entity discount permitted)
- ☐ Advance Order - # of Copies _____

4b. Payment of Fee(s): **(Please first reapply any previously paid issue fee shown above)**
- ☐ A check is enclosed.
- ☐ Payment by credit card. Form PTO-2038 is attached.
- ☐ The director is hereby authorized to charge the required fee(s), any deficiency, or credits any overpayment, to Deposit Account Number _____ (enclose an extra copy of this form).

5. **Change in Entity Status** (from status indicated above)
- ☐ Applicant certifying micro entity status. See 37 CFR 1.29
- ☐ Applicant asserting small entity status. See 37 CFR 1.27
- ☐ Applicant changing to regular undiscounted fee status.

NOTE: Absent a valid certification of Micro Entity Status (see forms PTO/SB/15A and 15B), issue fee payment in the micro entity amount will not be accepted at the risk of application abandonment.

NOTE: If the application was previously under micro entity status, checking this box will be taken to be a notification of loss of entitlement to micro entity status.

NOTE: Checking this box will be taken to be a notification of loss of entitlement to small or micro entity status, as applicable.

NOTE: This form must be signed in accordance with 37 CFR 1.31 and 1.33. See 37 CFR 1.4 for signature requirements and certifications.

Authorized Signature _____ Date _____

Typed or printed name _____ Registration No. _____

PTOL-85 Part B (10-13) Approved for use through 10/31/2013. OMB 0651-0033 U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
14/288,094	05/27/2014	Timothy J. Tewalt	SRC036	9362

25235 7590 06/09/2015

HOGAN LOVELLS US LLP
TWO NORTH CASCADE AVENUE
SUITE 1300
COLORADO SPRINGS, CO 80903

EXAMINER	
HO, HOAI V	

ART UNIT	PAPER NUMBER
2827	

DATE MAILED: 06/09/2015

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(Applications filed on or after May 29, 2000)

The Office has discontinued providing a Patent Term Adjustment (PTA) calculation with the Notice of Allowance.

Section 1(h)(2) of the AIA Technical Corrections Act amended 35 U.S.C. 154(b)(3)(B)(i) to eliminate the requirement that the Office provide a patent term adjustment determination with the notice of allowance. See Revisions to Patent Term Adjustment, 78 Fed. Reg. 19416, 19417 (Apr. 1, 2013). Therefore, the Office is no longer providing an initial patent term adjustment determination with the notice of allowance. The Office will continue to provide a patent term adjustment determination with the Issue Notification Letter that is mailed to applicant approximately three weeks prior to the issue date of the patent, and will include the patent term adjustment on the patent. Any request for reconsideration of the patent term adjustment determination (or reinstatement of patent term adjustment) should follow the process outlined in 37 CFR 1.705.

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at 1-(888)-786-0101 or (571)-272-4200.

OMB Clearance and PRA Burden Statement for PTOL-85 Part B

The Paperwork Reduction Act (PRA) of 1995 requires Federal agencies to obtain Office of Management and Budget approval before requesting most types of information from the public. When OMB approves an agency request to collect information from the public, OMB (i) provides a valid OMB Control Number and expiration date for the agency to display on the instrument that will be used to collect the information and (ii) requires the agency to inform the public about the OMB Control Number's legal significance in accordance with 5 CFR 1320.5(b).

The information collected by PTOL-85 Part B is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450. Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Notice of Allowability	Application No.	Applicant(s)	
	14/288,094	TEWALT, TIMOTHY J.	
	Examiner	Art Unit	
	HOAI V. HO	2827	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *5/27/2014*.

2. ☐ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.

3. ☒ The allowed claim(s) is/are *1-19*. As a result of the allowed claim(s), you may be eligible to benefit from the **Patent Prosecution Highway** program at a participating intellectual property office for the corresponding application. For more information, please see http://www.uspto.gov/patents/init_events/pph/index.jsp or send an inquiry to PPHfeedback@uspto.gov.

4. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☒ Notice of References Cited (PTO-892)
2. ☒ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date 5/14
3. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
4. ☐ Interview Summary (PTO-413), Paper No./Mail Date _____ .
5. ☒ Examiner's Amendment/Comment
6. ☐ Examiner's Statement of Reasons for Allowance
7. ☐ Other _____.

The present application, filed on or after March 16, 2013, is being examined under the first inventor to file provisions of the AIA.

1. This office acknowledges receipt of the following item from the Applicant:

Information Disclosure Statement (IDS) was considered.

2. Claims 1-19 are presented for examination and allowed.

Allowable Subject matter

3. The following is a statement of reasons for the indication of allowable subject matter:

Claims include allowable subject matter since the prior art made of record and considered pertinent to the applicant's disclosure, taken individually or in combination, does not teach or suggest the claimed limitations having a data maintenance block coupled to said reconfigurable logic device and self-refresh command inputs of said DRAM memory, said data maintenance block operative to provide stable input levels on said self- refresh command inputs while said reconfigurable logic device is reconfigured and a combination of the other limitations thereof as recited in claims 1 and 11.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

4. Any inquiry concerning this communication or earlier communications from the examiner should be directed to HOAI V. HO whose telephone number is (571)272-1777. The examiner can normally be reached on 7:00 AM -- 5:30 PM from Monday through Thursday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Amir Zarabian can be reached on (571) 272-1852. The fax phone number for the organization where this application or proceeding is assigned is (571)-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

/Hoai V. Ho/
Primary Examiner, Art Unit 2827

Notice of References Cited	Application/Control No. 14/288,094	Applicant(s)/Patent Under Reexamination TEWALT, TIMOTHY J.	
	Examiner HOAI V. HO	Art Unit 2827	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	Classification
*	A	US-2014/0211579	07-2014	Lovelace, John V.	365/200
	B	US-			
	C	US-			
	D	US-			
	E	US-			
	F	US-			
	G	US-			
	H	US-			
	I	US-			
	J	US-			
	K	US-			
	L	US-			
	M	US-			

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	ISSUE DATE	PATENT NO.	ATTORNEY DOCKET NO.	CONFIRMATION NO.
14/288,094	10/06/2015	9153311	SRC036	9362

25235 7590 09/16/2015

HOGAN LOVELLS US LLP
TWO NORTH CASCADE AVENUE
SUITE 1300
COLORADO SPRINGS, CO 80903

ISSUE NOTIFICATION

The projected patent number and issue date are specified above.

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment is 0 day(s). Any patent to issue from the above-identified application will include an indication of the adjustment on the front page.

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Application Assistance Unit (AAU) of the Office of Data Management (ODM) at (571)-272-4200.

APPLICANT(s) (Please see PAIR WEB site http://pair.uspto.gov for additional applicants):

Timothy J. Tewalt, Larkspur, CO;
SRC Computers, LLC., Colorado Springs, CO;

EXHIBIT L

Office Action Summary	Application No.	Applicant(s)	
	10/285,318	HUPPENTHAL ET AL.	
	Examiner	Art Unit	
	Eric Coleman	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 1 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-55 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-55 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1-5,26-31,52,53 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta (US patent No. 6,385,757) in view of Khan US Patent No. 5,274,832).

3. Gupta taught the invention substantially as claimed including a data processing ("DP") system comprising: defining a calculation for a reconfigurable computing system instantiating the performance of at least two array functional units (FU00-FU10)(e.g., see col. 17, lines 28-52 and col. 21, lines 22-29) to perform the calculation.

4. Gupta did not expressly detail utilizing the array functional units to operate on a subsequent data dimension of the calculation and substantially concurrently using the second of the array units to operate on a previous data dimension of the calculation. Khan however taught operating on three dimensions using plural two dimensional arrays that operate concurrently on respective dimensions and are coupled to together to produce the three dimensional array (e.g., see col. 4, lines 35-62 and col. 12, lines 15-55).

5. It would have been obvious to one of ordinary skill in the DP art to combine the

teachings of Gupta and Khan. One of ordinary skill would have been motivated to incorporate the three dimensional array operation of the Khan reference into the Gupta system to allow the combined system to be able to perform calculations on more complicated (three dimensional) problems.

6. As to the further limitations of claim 26, Khan taught (e.g., see fig. 8) a three dimensional systolic array with connections between processors in three dimensions.

7. As to claim 2-5,27-30 Khan taught the calculation comprising plurality of planes, and grid points and plural time-steps and vectors(e.g., see fig. 8 and col. 12, lines 15-55). As per claim 31, the system taught by Khan shows direct connection between the processing elements in the array and therefore the storing of data to an extrinsic memory (i.e., outside the array) would have been unnecessary when the transfer of data between columns was performed (e.g., see fig. 8).

8. As to the limitations of claims 52 and 53 the reconfigurable systolic processor would have been able to adapt to the application an therefore would have been an adaptive processor. As to the processor comprising a microprocessor one of ordinary skill would have been motivated to implement the systolic processor as described above as an microprocessor at least to take advantage of the reduced cost and reduced system size as was well known in the art at the time of the claimed invention.

9. Claims19,45 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta and Khan as applied to claims 1-2,26 above, and further in view of Leeland (US patent No. 4,872,133).

10. Leeland taught calculation comprised a financial application modeling using a spreadsheet application (e.g., see col. 5, lines 3-32).

11. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Leeland and Gupta. One of ordinary skill would have been motivated to incorporate the Leeland teaching of financial spreadsheet application for an array processor in order to provide an additional use for the combined system.

12. Claim 10-16 and 36-42,54 rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta and Khan as applied to claims 1-2,26 above, and further in view of Benner (US Patent No. 5,072,371).

13. Benner taught the calculation comprising fluid flow calculation and structural analysis (e.g., see col. 22, lines 35-52).

14. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Benner and Gupta. One of ordinary skill would have been motivated to incorporate the Benner teaching of fluid flow and structural analysis applications for an array processor in order to provide an additional uses for the combined system.

15. As to the limitation in claim 54 of performing a calculation unit a variable changed is value in a system processing an restarting at that value The Benner system taught systolically performing calculations on fluid flow. Since in such a problem one of ordinary skill would at times be interested when a change in the data occurred and adjust the calculation to pin point the calculation around that certain point then one of ordinary skill would have been motivated to operate the Benner and Gupta and Khan

system to process systolically until a change in data occurred and then restart the calculation at the point of the change to better determine the magnitude of the change in data.

16. Claim 6-9,25,32-35,51 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta and Khan as applied to claims 1-2,26 above, and further in view of Helbig (US patent No. 4,962,381).

17. Helbig taught the application of a systolic processor for radar, medical ultrasound and other imaging applications (e.g., see col. 1, lines 1-5) Clearly this would have also comprised images processed by standard MPEG and JPEG standards.

18. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Helbig and Gupta. One of ordinary skill would have been motivated to incorporate the Helbig teaching of radar, medical ultrasound and other imaging applications for an systolic processor in order to provide an additional uses for the combined system.

19. As to the limitation of claims 25 and 51, since signal filtering would have been associated with the applications taught by Helbig such as radar then one of ordinary skill would have been motivated to use the Helbig systolic processor in signal filtering applications.

20. Claim 17,18,22-24,43,44,48-50 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta and Khan as applied to claims 1-2,26 above, and further in view of Skaletsky (US patent No. 5,784,108).

21. Skaletsky taught using an systolic processor for processing search algorithm for image search such as when a best match was to be found and clearly this would have been applicable to data mining as these are similar applications (e.g., see col. 3, line 13-col. 4, line 57).

22. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Skaletsky and Gupta. One of ordinary skill would have been motivated to incorporate the Skaletsky teaching of search algorithm applications for an systolic processor in order to provide an additional uses for the combined system.

23. As to the limitations of claims 22-24,48-50 in light of the search algorithm teaching especially for finding a best match for data then the use of systolic processors for similar applications such as the genetic pattern matching, protein folding and organic structure interaction would have been an obvious uses for systolic processors (such as taught by Skaletsky) to one of ordinary skill in the DP art.

24. Claim 20,21,46,47 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta and Khan as applied to claims 1-2,26 above, and further in view of Gai (US patent No. 6,061,706).

25. Gai taught use of systolic processors in encryption/decryption applications to speed the encryption/decryption of public keys (e.g. see col. 1, lines 25-41.

26. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Gai and Gupta. One of ordinary skill would have been motivated to

incorporate the Gai teaching of encryption and decryption applications for an systolic processor in order to provide an additional uses for the combined system.

27. Claims 55 is rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta (US patent No. 6,385,757)

28. Gupta taught the invention substantially as claimed including data processing ("DP") system comprising a reconfigurable processor that provides indication of whether it performs speculative and systolic processing (e.g., see col. 15, lines 6-66). Consequently, one ordinary skill would have been motivated to perform systolic and speculative processing at least in order to utilize the parameters indicated by Gupta for use in systolic and speculative processing (e.g., see col. 15, lines 56-63).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eric Coleman whose telephone number is (571) 272-4163. The examiner can normally be reached on Monday-Thursday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

EC

ERIC COLEMAN
PRIMARY EXAMINER

RECEIVED
CENTRAL FAX CENTER
DEC 16 2005

Client Matter No. 80404.0018
Via Facsimile

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Serial No. 10/285,318 Application of: Jon M. Huppenthal and David E. Caliga Filed: October 31, 2002 Art Unit: 2183 Examiner: Coleman, Eric Attorney Docket No. SRC015 For: MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS	Confirmation No.: 1420 Customer No.: **25235**

AMENDMENT

MAIL STOP AMENDMENT
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the office communication mailed October 7, 2005, please amend the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims which begins on page 2 of this paper.

Remarks/Arguments begin on page 9 of this paper.

PAGE 2/16 * RCVD AT 12/16/2005 11:29:54 AM [Eastern Standard Time] * SVR:USPTO-EFXRF-6/26 * DNIS:2738300 * CSID:+ * DURATION (mm-ss):03-34

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1. (currently amended) A method for data processing in a reconfigurable computing system comprising a plurality of functional units, said method comprising:

defining a calculation for said reconfigurable computing system;

instantiating at least two of said functional units to perform said calculation wherein how many functional units and functional type of each functional unit is based on the calculation;

utilizing a first of said functional units to operate upon a subsequent data dimension of said calculation; and

substantially concurrently utilizing a second of said functional units to operate upon a previous data dimension of said calculation.

2. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple vectors in said calculation.

3. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple planes in said calculation.

4. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple time steps in said calculation.

5. (original) The method of claim 1 wherein said subsequent an previous data dimensions of said calculation comprise multiple grid points in said calculation.

6. (original) The method of claim 1 wherein said calculation comprises a seismic imaging calculation.

7. (original) The method of claim 1 wherein said calculation comprises a synthetic aperture radar imaging calculation.

8. (original) The method of claim 1 wherein said calculation comprises a JPEG image compression calculation.

9. (original) The method of claim 1 wherein said calculation comprises an MPEG image compression calculation.

10. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for a reservoir simulation.

11. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for weather prediction.

12. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for automotive applications.

13. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for aerospace applications.

14. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for an injection molding application.

15. (currently amended) The method of claim 1 wherein ~~said calculation comprises a structures calculation for crash analysis~~ instantiating includes establishing a stream communication connection between functional units.

16. (original) The method of claim 1 wherein said calculation is comprises a structures calculation for structural analysis.

17. (original) The method of claim 1 wherein said calculation comprises a search algorithm for an image search.

18. (original) The method of claim 1 wherein said calculation comprises a search algorithm for data mining.

19. (original) The method of claim 1 wherein said calculation comprises a financial modeling application.

20. (original) The method of claim 1 wherein said calculation comprises an encryption algorithm.

21. (currently amended) The method of claim 1 wherein said ~~calculation comprises an decryption algorithm~~ reconfigurable computing system communicates between functional units independent of external communication protocols.

22. (original) The method of claim 1 wherein said calculation comprises a genetic pattern matching function.

23. (original) The method of claim 1 wherein said calculation comprises a protein folding function.

24. (original) The method of claim 1 wherein said calculation comprises an organic structure interaction function.

25. (original) The method of claim 1 wherein said calculation comprises a signal filtering application.

26. (original) A method for data processing in a reconfigurable computing system comprising a plurality of functional units, said method comprising:

defining a first systolic wall comprising rows of cells forming a subset of said plurality of functional units;

computing a value at each of said cells in at least a first row of said first systolic wall;

communicating said values between cells in said first row of said cells to produce updated values;

communicating said updated values to a second row of said first systolic wall; and

substantially concurrently providing said updated values to a first row of a second systolic wall of rows of cells in said subset of said plurality of functional units.

27. (original) The method of claim 26 wherein said values correspond to vectors in a computation.

28. (original) The method of claim 26 wherein said values correspond to planes in a computation.

29. (original) The method of claim 26 wherein said values correspond to time steps in a computation.

30. (original) The method of claim 26 wherein said values correspond to grid points in a computation.

31. (original) The method of claim 26 wherein said step of communicating said updated values to a second row of said first systolic wall is carried out without storing said updated values in an extrinsic memory.

32. (original) The method of claim 26 wherein said values correspond to a seismic imaging calculation.

33. (original) The method of claim 26 wherein said values correspond to a synthetic aperture radar imaging calculation.

34. (original) The method of claim 26 wherein said values correspond to a JPEG image compression calculation.

35. (original) The method of claim 26 wherein said values correspond to an MPEG image compression calculation.

36. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for a reservoir simulation.

37. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for weather prediction.

38. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for automotive applications.

39. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for aerospace applications.

40. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for an injection molding application.

41. (currently amended) The method of claim 26 wherein ~~said values correspond to a structures calculation for crash analysis~~ defining includes establishing a stream communication connection between functional units and

wherein how many functional units and functional type of each functional unit is based on a computing algorithm within the reconfigurable computing system.

42. (original) The method of claim 26 wherein said values correspond to a structures calculation for structural analysis.

43. (original) The method of claim 26 wherein said values correspond to a search algorithm for an image search.

44. (original) The method of claim 26 wherein said values correspond to a search algorithm for data mining.

45. (original) The method of claim 26 wherein said values correspond to a financial modeling application.

46. (original) The method of claim 26 wherein said values correspond to an encryption algorithm.

47. (currently amended) The method of claim 26 wherein ~~said values correspond to an decryption algorithm~~ reconfigurable computing system communicates between functional units independent of external communication protocols.

48. (original) The method of claim 26 wherein said values correspond to a genetic pattern matching function.

49. (original) The method of claim 26 wherein said values correspond to a protein folding function.

50. (original) The method of claim 26 wherein said values correspond to an organic structure interaction function.

51. (original) The method of claim 26 wherein said values correspond to a signal filtering application.

52. (original) The method of claim 26 wherein said reconfigurable computing system comprises at least one adaptive processor.

53. (original) The method of claim 52 wherein said reconfigurable computing system further comprises at least one microprocessor.

54. (currently amended) A method for data processing in a reconfigurable computing system comprising a plurality of functional units, said method comprising:

performing a calculation by a subset of said plurality of functional units to produce computed data;

passing said computed data from a first column of said calculation to a next column in said calculation;

evaluating a rate of change in at least one variable for each of said columns in said calculation;

continuing said calculation [[if]] when said variable does not change for a particular column of said calculation; and

restarting said calculation at said column of said calculation where said variable does change.

55. (Canceled)

56. (New) The method of claim 54 wherein how many functional units comprise the subset and functional type of each functional unit in said subset is based on the calculation and wherein the passing step is external communication protocol independent.

REMARKS/ARGUMENTS

Claims 1-55 were presented for examination and are pending in this application. In an Official Office Action dated October 7, 2005, claims 1-55 were rejected. The Applicants thank the Examiner for his consideration and address the Examiner's comments concerning the claims pending in this application below.

Applicants herein amend claims 1, 15, 21, 41, 47 and 54 and respectfully traverse the Examiners rejections. Claim 55 is presently canceled without prejudice and new claim 56 is presently added. Claims 1-54 and 56 are now pending in this application. These changes are believed not to introduce new matter, and their entry is respectfully requested. Support of the amendments can be generally found on page 11 and page 16 of the specification. The claims have been amended to expedite the prosecution and issuance of the application. In making this amendment, Applicants have not and are not narrowing the scope of the protection to which the Applicants consider the claimed invention to be entitled and do not concede, directly or by implication, that the subject matter of such claims was in fact disclosed or taught by the cited prior art. Rather, Applicants reserve the right to pursue such protection at a later point in time and merely seek to pursue protection for the subject matter presented in this submission.

35 U.S.C. §103(a) Obviousness Rejection of Claims

Claims 1-5, 26-31, 52 and 53 were rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 6,385,757 ("Gupta") in view of U.S. Patent No. 5,274,832 ("Khan"). Applicants respectfully traverse these rejections in light of the aforementioned remarks and respectfully requests reconsideration.

MPEP §2143 provides:

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation,

> either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teaching. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations.

The cited references fail to teach or suggest all of the limitations recited in the claims as currently amended. For example, independent claim 1 recites, "...wherein how many functional units and functional type of each functional unit is based on the calculation ..." and "...substantially concurrently utilizing a second of said functional units to operate upon a previous data dimension of said calculation." Neither Gupta nor Khan teach or suggest a substantially concurrent use of functional units of a reconfigurable computing system to concurrently operate upon data dimensions of a calculation. In contrast both Gupta and Khan follow the traditional parallel processing format of sequential processing data since the result of one processor, functional unit, or cell may be required by an adjacent processor, functional unit, or cell.

Typically, in a multi-processor, microprocessor-based system, each processor is allocated but a relatively small portion of the total problem called a cell. However, to solve the total problem, results of one processor are often required by many adjacent cells because their cells interact at the boundary. Consequently, intermediate results must be passed around the system in order to complete the computation of the total problem. This, by necessity, involves numerous other chips and busses that run at much slower speeds than the microprocessor thus resulting in system performance often many orders of magnitude lower than the raw computation time.

In the use of an adaptive or reconfigurable processor-based system as is claimed in the Applicants' invention, ten to one thousand times more computations

can be performed within a single chip and any boundary data that is shared between these functional units need never leave a single integrated circuit chip eliminating the need for external communication protocols and simplifying internal communications. For example a complier associated with the reconfigurable computing system can establish stream connections between functional units that rely on general communication protocols. Therefore, data moving around the system, and its impact on reducing overall system performance, can also be reduced by two or three orders of magnitude. This will allow both significant improvements in performance in certain applications as well as enabling certain applications to be performed in a practical timeframe that could not previously be accomplished. Such an adaptive processor-based system is distinct from that taught by Khan.

In addition, the Applicants' invention build functional units of the reconfigurable processor-based system based on the algorithms being used in the calculations. The type of each functional unit and the total number of functional units created is unique for each assigned task. This is distinct from Gupta. Gupta teaches a system using a Very Long Instruction Word ("VLIW") processor. VLIW processors do have the ability to use multiple arithmetic functional units one at a time but the set of functional units are limited and fixed within the VLIW processor. The flexible nature of the Applicants' invention allows for computational flow in one or more dimensions of the problem. The system disclosed by Gupta and Kahn does not offer such an approach.

Gupta also appears to teach a system to generate an instruction format that is used to control a processor control path in what is called parallel instruction computing. This instruction-level parallelism issues several operations per instruction to multiple functional units to control a processors data path. As the Examiner admits, Gupta fails to teach a substantially concurrent use of data dimensions during a calculation. The Applicants disagree with the Examiner's

conclusion that Khan teaches this noted insufficiency of Gupta. Khan appears to teach a serialized or sequential approach to multi-processor parallelism using systolic arrays. As stated in Khan, "...the corresponding matrix and vector signals are inputted into their respective processing elements sequentially, multiplied and accumulated therein." See Khan Col. 12, lines 35-37. Thus, Khan teaches a systolic sequential parallel approach to processing that moves in one direction in a one by one fashion.

Systolic sequential parallelism utilizes an array of processing elements (typically multiplier-accumulator chips) in a pipeline structure. The "systolic," coined by H. T. Kung of Carnegie-Mellon, refers to the rhythmic transfer of data through the pipeline, like blood flowing through the vascular system. Such an approach inherently accomplishes calculations by using a serialized approach. As recited in Gupta, "...the algorithm selects a set of FUs [Functional Units] to be instantiated in the data path, one by one, by looking at the requirement of the operation group cliques provided." Gupta, Col. 21, lines 23-24. (emphasis added)

In contrast to the sequential processing operation of Khan and Gupta, the Applicants' invention utilizes available resources to have an application evaluate a problem in a concurrent data flow sense. That is, it will "pass" a subsequent dimension of a given problem through a first loop of logic concurrently with the previous dimension of data being processed through a second loop. This type of concurrent operation is not taught or suggested by Gupta or Kahn. In practice, a "dimension" of data can be: multiple vectors of a problem, multiple planes of a problem, multiple time steps in a problem and so forth.

In addition, and as recited in claim 26, the Applicants' method "substantially concurrently provides updated values to a first row of a second systolic wall of rows of cells...." (emphasis added) The combination of defining a calculation for a reconfigurable computing system and concurrently operating on data and communicating values between cells is distinct from Gupta in view of Khan. This

and other features of claim 26 were recognized as having novelty, an inventive step, and industrial applicability by the International Preliminary Examining Authority of the Patent Cooperation Treaty. A recently received International Preliminary Examination Report received November 16, 2005 found that the combination of limitations found in claim 26 and 54 (designated as claims 1 and 7 of the PCT application) in full consideration of Gupta and Khan, possessed novelty, inventive step, and industrial applicability.

The Applicants also assert that Gupta in view of Khan are improperly combined. To establish a prima facie case of obviousness there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teaching. Assuming for arguments sake that the elements of the Applicants' invention are found in a combination of Gupta and Khan, (an assumption to which the Applicants do not agree) there is nothing in either Gupta or Khan to suggest or motivate such a combination or modification. The long felt need of the Applicants' invention given problems associated with parallel processing as well as the commercial success of products derived from the Applicants' invention are evidence that one of ordinary skill in the art would not and have not been motivated to combine these references. The Applicants thus traverse the Examiner's rejections of independent claims 1 and 26 for the aforementioned reasons.

Claims 2-5, 27-31, 52 and 53 depend from claims 1 and 26 respectively and are, for at least the same aforementioned reasons, patentable over Gupta in view of Khan. The Applicants respectfully request the rejections be withdrawn and the claims reconsidered.

Claims 19 and 45 were rejected under 35 U.S.C § 103(a) as being unpatentable over Gupta in view of Khan and in further view of U.S. Patent No. 4,872,133 ("Leeland"). Leeland fails to rectify the aforementioned deficiencies of

Gupta and Khan with respect to independent claims 1 and 26 and therefore, as claims 19 and 45 depend from claims 1 and 26 respectively, the Applicants submit claims 19 and 45 are patentable over Gupta in view of Khan in further view of Leeland.

The Examiner also rejects dependent claims 10-16, 36-42 and independent claim 54 under 35 U.S.C. § 103(a) as being unpatentable over Gupta in view of Khan in further view of U.S. Patent No. 5,072,371 ("Benner"). The Applicants traverse these rejections. For at least the aforementioned reasons, the Applicants submit that Brenner fails to resolve the deficiencies noted in Gupta and Kahn. Claims 10-16 and 16-42, which depend from claims 1 and 21 respectively, are therefore patentable over Gupta in view of Khan in further view of Benner.

With regard to independent claim 54, the Examiner asserts that Benner discloses continuing calculations of variables that do not change in a column and restarting calculations of variables once a change occurs. The Applicants disagree. The text cited by the Examiner (Benner Col. 22, lines 35-52) does not teach or suggest systolic calculations as recited in claim 54 and the Examiner's conclusion that the words "wave mechanics, fluid dynamics, and beam strain analysis" imply the data processing in a reconfigurable computing system claimed by the Applicants is unjustified. The Applicants respectfully request either the rejection be withdrawn or specific reference to portions of Gupta, Kahn, and Benner that teach and suggest each and every limitation of claim 54 be identified.

Claims 6-9, 17-18, 20-25, 32-35, 43-44 and 46-51 are rejected by the Examiner under 35 U.S.C. § 103(a) as being unpatentable over Gupta and Kahn as applied to claims 1-2 and 26 and in further view of U.S. Patent No. 4,962,381 ("Helbig"), U.S. Patent No. 5,784,108 ("Skaletsky") and U.S. Patent No. 6, 061,706 ("Gai"), respectively. As all of these claims depend from either independent claim 1 or 26, the Applicants submit, for at least the aforementioned reasons, each is patentable over Gupta and Kahn. Reconsideration is respectfully requested.

Based on the above amendment and the following remarks, Applicants respectfully request that the Examiner reconsider all outstanding rejections and withdraw them.

In view of all of the above, the claims are now believed to be allowable and the case in condition for allowance which action is respectfully requested. Should the Examiner be of the opinion that a telephone conference would expedite the prosecution of this case, the Examiner is requested to contact Applicants' attorney at the telephone number listed below.

No fee is believed due for this submittal. However, any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

16 December, 2005

Michael C. Martensen, Reg. No. 46,901
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5906 Tel
(303) 899-7333 Fax

Office Action Summary	Application No.	Applicant(s)	
	10/285,318	HUPPENTHAL ET AL.	
	Examiner	Art Unit	
	Eric Coleman	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-54,56 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-54,56 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1-5,15,21,26-31,41,47,52,53,56 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta (US patent No. 6,385,757) in view of Khan US Patent No. 5,274,832).

3. Gupta taught the invention substantially as claimed including a data processing ("DP") system comprising: defining a calculation for a reconfigurable computing system instantiating the performance of at least two array functional units (FU00-FU10)(e.g., see col. 17, lines 28-52 and col. 21, lines 22-29) to perform the calculation.

4. Gupta did not expressly detail utilizing the array functional units to operate on a subsequent data dimension of the calculation and substantially concurrently using the second of the array units to operate on a previous data dimension of the calculation. Khan however taught operating on three dimensions using plural two dimensional arrays that operate concurrently on respective dimensions and are coupled to together to produce the three dimensional array (e.g., see col. 4, lines 35-62 and col. 12, lines 15-55).

5. It would have been obvious to one of ordinary skill in the DP art to combine the

teachings of Gupta and Khan. One of ordinary skill would have been motivated to incorporate the three dimensional array operation of the Khan reference into the Gupta system to allow the combined system to be able to perform calculations on more complicated (three dimensional) problems.

6. Claims 1,41,56 have the limitation of wherein how many functional units and functional type of each functional unit is based on the calculation (which comprises the algorithm). As to this limitation Khan taught specific selection of the number of processing elements (which correspond to claimed functional units) are different depending on whether the calculation was a two dimensional or three dimensional calculation (e.g., see col. 5, lines 17-30). As to the type of functional unit being based on the calculation the system is a special purpose system that uses a specific type of functional unit namely processing elements that perform systolic array calculations readily and where connections and transfer of data for performing the calculation is readily done. Therefore in the implementation of the Khan teachings the type of functional unit is based on the type of calculations and the algorithm that were to be performed by the system (e.g., two dimensional algorithm or three dimensional algorithm) (e.g., see col. 5, lines 32-49).

7. As to instantiating including establishing a stream communication connection between functional units (claims 15,41) Khan taught minimizes interconnections of processing elements and the matrix and vector signal subsets are specifically formed so that they need to be inputted to only one row and one columns and yet still be properly processing systolically along all dimensions within the array (e.g., see col. 5, lines 2-48).

Consequently the stream of communication between functional units is established as the interconnections are made and data is transferred systolically in at least one stream between processors.

8. As to the further limitations of claim 26, Khan taught (e.g., see fig. 8) a three dimensional systolic array with connections between processors in three dimensions and the selection is done to minimize global interconnections.

9. As to claim 2-5,27-30 Khan taught the calculation comprising plurality of planes, and grid points and plural time-steps and vectors (e.g., see fig. 8 and col. 12, lines 15-55). As per claim 31, the system taught by Khan shows direct connection between the processing elements in the array and therefore the storing of data to an extrinsic memory (i.e., outside the array) would have been unnecessary when the transfer of data between columns was performed (e.g., see fig. 8).

10. As to the limitations of claims 52 and 53 the reconfigurable systolic processor would have been able to adapt to the application a therefore would have been an adaptive processor. As to the processor comprising a microprocessor one of ordinary skill would have been motivated to implement the systolic processor as described above as an microprocessor at least to take advantage of the reduced cost and reduced system size as was well known in the art at the time of the claimed invention.

Claim 21,47,56 comprises the limitation of reconfigurable computing system communicates between functional units independent of external communication protocols. Since the Khan and Gupta system taught systems that did not use external

protocols to communicate between the processors it is anticipated that in the implementation of a system using the Khan and Gupta teachings that the communication between processors would have been protocol independent.

11. Claims 19, 45 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta and Khan as applied to claims 1-2,26 above, and further in view of Leeland (US patent No. 4,872,133).

12. Leeland taught calculation comprised a financial application modeling using a spreadsheet application (e.g., see col. 5, lines 3-32).

13. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Leeland and Gupta. One of ordinary skill would have been motivated to incorporate the Leeland teaching of financial spreadsheet application for an array processor in order to provide an additional use for the combined system.

14. Claim 10-14,16 and 36-40,42,54 rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta and Khan as applied to claims 1-2,15,26 above, and further in view of Benner (US Patent No. 5,072,371).

15. Benner taught the calculation comprising fluid flow calculation and structural analysis (e.g., see col. 22, lines 35-52).

16. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Benner and Gupta. One of ordinary skill would have been motivated to incorporate the Benner teaching of fluid flow and structural analysis applications for an array processor in order to provide an additional uses for the combined system.

17. As to the limitation in claim 54 of performing a calculation unit a variable changed is value in a system processing a restarting at that value The Benner system taught systolically performing calculations on fluid flow. Since in such a problem one of ordinary skill would at times be interested when a change in the data occurred and adjust the calculation to pin point the calculation around that certain point then one of ordinary skill would have been motivated to operate the Benner and Gupta and Khan system to process systolically until a change in data occurred and then restart the calculation at the point of the change to better determine the magnitude of the change in data.

18. Claim 6-9,25,32-35,51 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta and Khan as applied to claims 1-2,26 above, and further in view of Helbig (US patent No. 4,962,381).

19. Helbig taught the application of a systolic processor for radar, medical ultrasound and other imaging applications (e.g., see col. 1, lines 1-5) Clearly this would have also comprised images processed by standard MPEG and JPEG standards.

20. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Helbig and Gupta. One of ordinary skill would have been motivated to incorporate the Helbig teaching of radar, medical ultrasound and other imaging applications for an systolic processor in order to provide an additional uses for the combined system.

21. As to the limitation of claims 25 and 51, since signal filtering would have been associated with the applications taught by Helbig such as radar then one of ordinary

skill would have been motivated to use the Helbig systolic processor in signal filtering applications.

22. Claim 17,18,22-24,43,44,48-50 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta and Khan as applied to claims 1-2,26 above, and further in view of Skaletsky (US patent No. 5,784,108).

23. Skaletsky taught using an systolic processor for processing search algorithm for image search such as when a best match was to be found and clearly this would have been applicable to data mining as these are similar applications (e.g., see col. 3, line 13-col. 4, line 57).

24. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Skaletsky and Gupta. One of ordinary skill would have been motivated to incorporate the Skaletsky teaching of search algorithm applications for an systolic processor in order to provide an additional uses for the combined system.

25. As to the limitations of claims 22-24,48-50 in light of the search algorithm teaching especially for finding a best match for data then the use of systolic processors for similar applications such as the genetic pattern matching, protein folding and organic structure interaction would have been an obvious uses for systolic processors (such as taught by Skaletsky) to one of ordinary skill in the DP art.

26. Claim 20,46, are rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta and Khan as applied to claims 1-2,26 above, and further in view of Gai (US patent No. 6,061,706).

27. Gai taught use of systolic processors in encryption/decryption applications to speed the encryption/decryption of public keys (e.g. see col. 1, lines 25-41.

28. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Gai and Gupta. One of ordinary skill would have been motivated to incorporate the Gai teaching of encryption and decryption applications for an systolic processor in order to provide an additional uses for the combined system.

The change in scope of the amended claims has necessitated a new search.

Response to Arguments

Applicant's arguments with respect to claims 1-25,41,47,54,56 have been considered but are moot in view of the new ground(s) of rejection.

Applicant's arguments filed 12/16/05 have been fully considered but they are not persuasive.

The applicant argues in substance the following:

Gupta and Khan did not teach concurrently operating on plural dimensions of a calculation The examiner contends that this is taught by Khan as detailed in the outstanding rejection above (e.g., see col. 4, lines 35-62 and col. 12, lines 15-55 of Khan). Note that during a pipelined operation plural functional units in an array operate concurrently and when the pipelines are in plural directions or dimensions then the concurrency extends to plural dimensions.

Applicant alleges that the combination is improper, The Examiner contends that the reasoning for the combination provided in the outstanding rejection above is proper.

Applicant alleges that Benner does not teach continuing calculations of variables that do not change in a column and restarting calculations of variables one a change occurs. The Examiner contends as expressed in outstanding rejection above that the applications utilized for the system of Benner such as fluid flow would have motivated one of ordinary skill to use the data processing in a reconfigurable computing system.

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

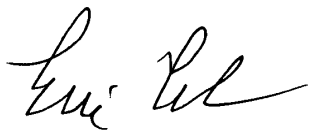
A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eric Coleman whose telephone number is (571) 272-4163. The examiner can normally be reached on Monday-Thursday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

EC



ERIC COLEMAN
PRIMARY EXAMINER

Client Matter No. 80404.0018
EFS-Web

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Serial No. 10/285,318 Application of: Jon M. Huppenthal and David E. Caliga Filed: October 31, 2002 Art Unit: 2183 Examiner: Coleman, Eric Attorney Docket No. SRC015 For: MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS	Confirmation No.: 1420 Customer No.: **25235**

AMENDMENT AFTER FINAL

MAIL STOP AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the office communication mailed March 6, 2006, please amend the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims which begins on page 2 of this paper.

Remarks/Arguments begin on page 9 of this paper.

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1. (currently amended) A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor, the reconfigurable processor comprising a plurality of functional units, said method comprising:

defining a calculation at the at least one reconfigurable processor for said reconfigurable computing system;

instantiating at least two of said functional units at the at least one reconfigurable processor to perform said calculation wherein how many functional units and functional type of each functional unit is based on the calculation and wherein communications between said functional units is external communication protocol independent and internal communication protocol independent;

utilizing a first of said functional units to operate upon a subsequent data dimension of said calculation; and

substantially concurrently utilizing a second of said functional units to operate upon a previous data dimension of said calculation.

2. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple vectors in said calculation.

3. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple planes in said calculation.

4. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple time steps in said calculation.

5. (original) The method of claim 1 wherein said subsequent an previous data dimensions of said calculation comprise multiple grid points in said calculation.

6. (original) The method of claim 1 wherein said calculation comprises a seismic imaging calculation.

7. (original) The method of claim 1 wherein said calculation comprises a synthetic aperture radar imaging calculation.

8. (original) The method of claim 1 wherein said calculation comprises a JPEG image compression calculation.

9. (original) The method of claim 1 wherein said calculation comprises an MPEG image compression calculation.

10. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for a reservoir simulation.

11. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for weather prediction.

12. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for automotive applications.

13. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for aerospace applications.

14. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for an injection molding application.

15. (previously presented) The method of claim 1 wherein instantiating includes establishing a stream communication connection between functional units.

16. (original) The method of claim 1 wherein said calculation is comprises a structures calculation for structural analysis.

17. (original) The method of claim 1 wherein said calculation comprises a search algorithm for an image search.

18. (original) The method of claim 1 wherein said calculation comprises a search algorithm for data mining.

19. (original) The method of claim 1 wherein said calculation comprises a financial modeling application.

20. (original) The method of claim 1 wherein said calculation comprises an encryption algorithm.

21. (Canceled)

22. (original) The method of claim 1 wherein said calculation comprises a genetic pattern matching function.

23. (original) The method of claim 1 wherein said calculation comprises a protein folding function.

24. (original) The method of claim 1 wherein said calculation comprises an organic structure interaction function.

25. (original) The method of claim 1 wherein said calculation comprises a signal filtering application.

26. (currently amended) A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one

reconfigurable processor comprising a plurality of functional units, said method comprising:

defining a first systolic wall comprising rows of cells forming a subset of said plurality of functional units;

computing at the at least one reconfigurable processor a value at each of said cells in at least a first row of said first systolic wall;

communicating said values between cells in said first row of said cells to produce updated values, wherein communicating said values is both internal and external communication protocol independent;

communicating said updated values to a second row of said first systolic wall, wherein communicating said updated values is both internal and external communication protocol independent; and

substantially concurrently providing said updated values to a first row of a second systolic wall of rows of cells in said subset of said plurality of functional units.

27. (original) The method of claim 26 wherein said values correspond to vectors in a computation.

28. (original) The method of claim 26 wherein said values correspond to planes in a computation.

29. (original) The method of claim 26 wherein said values correspond to time steps in a computation.

30. (original) The method of claim 26 wherein said values correspond to grid points in a computation.

31. (original) The method of claim 26 wherein said step of communicating said updated values to a second row of said first systolic wall is carried out without storing said updated values in an extrinsic memory.

32. (original) The method of claim 26 wherein said values correspond to a seismic imaging calculation.

33. (original) The method of claim 26 wherein said values correspond to a synthetic aperture radar imaging calculation.

34. (original) The method of claim 26 wherein said values correspond to a JPEG image compression calculation.

35. (original) The method of claim 26 wherein said values correspond to an MPEG image compression calculation.

36. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for a reservoir simulation.

37. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for weather prediction.

38. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for automotive applications.

39. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for aerospace applications.

40. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for an injection molding application.

41. (previously presented) The method of claim 26 wherein defining includes establishing a stream communication connection between functional units and wherein how many functional units and functional type of each functional unit is based on a computing algorithm within the reconfigurable computing system.

42. (original) The method of claim 26 wherein said values correspond to a structures calculation for structural analysis.

43. (original) The method of claim 26 wherein said values correspond to a search algorithm for an image search.

44. (original) The method of claim 26 wherein said values correspond to a search algorithm for data mining.

45. (original) The method of claim 26 wherein said values correspond to a financial modeling application.

46. (original) The method of claim 26 wherein said values correspond to an encryption algorithm.

47. (canceled)

48. (original) The method of claim 26 wherein said values correspond to a genetic pattern matching function.

49. (original) The method of claim 26 wherein said values correspond to a protein folding function.

50. (original) The method of claim 26 wherein said values correspond to an organic structure interaction function.

51. (original) The method of claim 26 wherein said values correspond to a signal filtering application.

52. (canceled)

53. (currently amended) The method of claim [[52]]26 wherein said reconfigurable computing system ~~further~~ comprises at least one microprocessor.

54. (currently amended) A method for data processing in a reconfigurable computing system, the reconfigurable computer system comprising at least one reconfigurable processor comprising a plurality of functional units, said method comprising:

performing a calculation at the at least one reconfigurable processor by a subset of said plurality of functional units to produce computed data;

passing said computed data from a first column of said calculation to a next column in said calculation, wherein said passing is both internal and external communication protocol independent;

evaluating a rate of change in at least one variable for each of said columns in said calculation;

continuing said calculation when said variable does not change for a particular column of said calculation; and

restarting said calculation at said column of said calculation where said variable does change.

55. (Canceled)

56. (Previously Presented) The method of claim 54 wherein how many functional units comprise the subset and functional type of each functional unit in said subset is based on the calculation and wherein the passing step is external communication protocol independent.

REMARKS/ARGUMENTS

Claims 1-54 and 56 were presented for examination and are pending in this application. In an Official Final Office Action dated March 6, 2006, claims 1-54, and 56 were rejected. The Applicants thank the Examiner for his consideration and address the Examiner's comments concerning the claims pending in this application below.

Applicants herein amend claims 1, 26, 53 and 54 and respectfully traverse the Examiner's rejections. Claims 21, 47, and 52 are presently canceled without prejudice. Claims 1-20, 22-46, 48-51 and 53, 54 and 56 are now pending in this application. The additional limitations brought into the independent claims place the claims in better condition for consideration on appeal and because they appear in dependent claims as filed, these amendments do not raise any new issues that would require further research by the Examiner. These changes are believed not to introduce new matter, and their entry is respectfully requested. The claims have been amended to expedite the prosecution and issuance of the application. In making these amendments, Applicants have not and are not narrowing the scope of the protection to which the Applicants consider the claimed invention to be entitled and do not concede, directly or by implication, that the subject matter of such claims was in fact disclosed or taught by the cited prior art. Rather, Applicants reserve the right to pursue such protection at a later point in time and merely seek to pursue protection for the subject matter presented in this submission.

35 U.S.C. §103(a) Obviousness Rejection of Claims

Claims 1-5, 15, 21, 26-31, 41, 47, 52, 53 and 56 were rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 6,385,757 ("Gupta") in view of U.S. Patent No. 5,274,832 ("Khan"). Applicants respectfully traverse these rejections in light of the aforementioned remarks and respectfully request reconsideration.

Gupta in view of Khan fails to teach or suggest computing data flows between functional units of a single reconfigurable processor. Gupta and Khan teach traditional parallel processing with sequential processing of data between one processor, functional unit, or cell and an adjacent processor, functional unit, or cell.

The Applicants' invention calculates the number of required functional units and the type of units entirely within a single reconfigurable processor to accomplish the processing task. Computations performed by these functional units are shared within the processor and thus never need leave the single reconfigurable processor environment. This consolidation of computations eliminates the need for external and internal communication protocols. Such an adaptive (reconfigurable) processor-based system is distinct from that taught by Khan or Gupta. Khan and Gupta do not teach performing these calculations in a single processor. Rather multiple processors are disclosed which would require consideration for both internal and external communication protocols.

The Applicants reject the sweeping and unsupported conclusion by the Examiner with respect to claims 21, 47, and 56 (now incorporated into independent claims 1, 26 and 54). The Examiner states that it is "anticipated that in the implementation of a system using Khan and Gupta teachings that the communication between processors would have been protocol independent." There is no basis for this conclusion. The invention as claimed states that communication between functional units, and not the processors, is communication protocol independent. Furthermore, the Examiner's rejection of the claims fail to meet the criteria established by the MPEP for rejection under 35 U.S.C. §103(a). The Applicants submit that this limitation is not taught or suggested in Khan or Gupta and as incorporated into claims 1, 26 and 54, place these claims in condition for allowance.

The Applicants reiterate that the Applicants' invention builds functional units of a reconfigurable processor-based system based on the algorithms being used in the calculations in a single reconfigurable processor. The type of each functional unit and the total number of functional units created is unique for each assigned task. This is distinct from the teachings of Gupta of a system using a Very Long Instruction Word ("VLIW") processor. VLIW processors do have the ability to use multiple arithmetic functional units one at a time but the set of functional units are limited and fixed within the VLIW processor.

Gupta teaches a system to generate an instruction format that is used to control a processor control path in what is called parallel instruction computing. This instruction-level parallelism issues several operations per instruction to multiple functional units to control a processors data path. As the Examiner admits in a previous Office Action, Gupta fails to teach a substantially concurrent use of data dimensions during a calculation. The Applicants reassert their disagreement with the Examiner's conclusion that Khan teaches this noted insufficiency of Gupta. The Examiner states that during a "pipelined" operation, plural functional units in an array operate concurrently and when the pipelines are in plural directions, then the concurrency extends to plural dimensions. [See Final Office Action dated March 6, 2006]. Khan teaches a systolic sequential parallel approach to processing that moves in one direction in a one by one fashion. The Applicants concur that plural pipelines moving in different directions extend the processing described in Khan in plural dimensions, but disagree that such an extension is the equivalent to concurrent operations as claimed by the Applicants. Each pipeline is by definition a serialized operation. While multiple pipelines may be operating concurrently each pipeline individually still operates and communicates in a serial fashion.

The Applicants' invention utilizes available resources to have an application evaluate a problem in a concurrent data flow sense and not in a pipeline sense. That is, it will "pass" a subsequent dimension of a given problem through a first loop

of logic concurrently with the previous dimension of data being processed through a second loop. This type of concurrent operation cannot occur in the pipeline operation described in Khan. Accordingly the Applicants submit that independent claims 1, 26 and 54 are patentable over Gupta in view of Khan.

Claims 2-5, 15, 21, 27-31, 41, 47, 53 and 56 depend from claims 1, 26 or 54 and are, for at least the same aforementioned reasons, patentable over Gupta in view of Khan. The Applicants respectfully request the rejections be withdrawn and the claims reconsidered.

Claims 19 and 45 were rejected under 35 U.S.C § 103(a) as being unpatentable over Gupta in view of Khan and in further view of U.S. Patent No. 4,872,133 ("Leeland"). Leeland fails to rectify the aforementioned deficiencies of Gupta and Khan with respect to independent claims 1 and 26 and therefore, as claims 19 and 45 depend from claims 1 and 26 respectively, the Applicants submit claims 19 and 45 are patentable over Gupta in view of Khan in further view of Leeland.

The Examiner also rejects dependent claims 10-14, 16, 36-40, 42 and independent claim 54 under 35 U.S.C. § 103(a) as being unpatentable over Gupta in view of Khan in further view of U.S. Patent No. 5,072,371 ("Benner"). The Applicants traverse these rejections. For at least the aforementioned reasons, the Applicants submit that Brenner fails to resolve the deficiencies noted in Gupta and Kahn. Claims 10-14, 16 and 36-40, 42, which depend from claims 1 and 26 respectively, are therefore patentable over Gupta in view of Khan in further view of Benner.

The Examiner's rejection of claim 54 as being unpatentable over Gupta in view of Khan and in further view of Benner is traversed in light of the present amendments and the aforementioned remarks.

Claims 6-9, 17-18, 20-25, 32-35, 43-44 and 46-51 are rejected by the Examiner under 35 U.S.C. § 103(a) as being unpatentable over Gupta and Kahn as

applied to claims 1-2 and 26 and in further view of U.S. Patent No. 4,962,381 ("Helbig"), or in further view of U.S. Patent No. 5,784,108 ("Skaletsky"), or in further view of U.S. Patent No. 6, 061,706 ("Gai"), respectively. As all of these claims depend from either independent claim 1 or 26, the Applicants submit, for at least the aforementioned reasons, each is patentable over Gupta and Kahn. Reconsideration is respectfully requested.

Based on the above amendment and the following remarks, Applicants respectfully request that the Examiner reconsider all outstanding rejections and withdraw them.

In view of all of the above, the claims are now believed to be allowable and the case in condition for allowance which action is respectfully requested. Should the Examiner be of the opinion that a telephone conference would expedite the prosecution of this case, the Examiner is requested to contact Applicants' attorney at the telephone number listed below.

No fee is believed due for this submittal. However, any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

4 May, 2006

Michael C. Martensen, Reg. No. 46,901
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5910 Tel
(303) 899-7333 Fax

Advisory Action Before the Filing of an Appeal Brief	Application No. 10/285,318	Applicant(s) HUPPENTHAL ET AL.	
	Examiner Eric Coleman	Art Unit 2183	

--The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

THE REPLY FILED 04 May 2006 FAILS TO PLACE THIS APPLICATION IN CONDITION FOR ALLOWANCE.

1. ☒ The reply was filed after a final rejection, but prior to or on the same day as filing a Notice of Appeal. To avoid abandonment of this application, applicant must timely file one of the following replies: (1) an amendment, affidavit, or other evidence, which places the application in condition for allowance; (2) a Notice of Appeal (with appeal fee) in compliance with 37 CFR 41.31; or (3) a Request for Continued Examination (RCE) in compliance with 37 CFR 1.114. The reply must be filed within one of the following time periods:
   a) ☐ The period for reply expires _____months from the mailing date of the final rejection.
   b) ☒ The period for reply expires on: (1) the mailing date of this Advisory Action, or (2) the date set forth in the final rejection, whichever is later. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of the final rejection.
   Examiner Note: If box 1 is checked, check either box (a) or (b). ONLY CHECK BOX (b) WHEN THE FIRST REPLY WAS FILED WITHIN TWO MONTHS OF THE FINAL REJECTION. See MPEP 706.07(f).

Extensions of time may be obtained under 37 CFR 1.136(a). The date on which the petition under 37 CFR 1.136(a) and the appropriate extension fee have been filed is the date for purposes of determining the period of extension and the corresponding amount of the fee. The appropriate extension fee under 37 CFR 1.17(a) is calculated from: (1) the expiration date of the shortened statutory period for reply originally set in the final Office action; or (2) as set forth in (b) above, if checked. Any reply received by the Office later than three months after the mailing date of the final rejection, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

NOTICE OF APPEAL

2. ☐ The Notice of Appeal was filed on _____. A brief in compliance with 37 CFR 41.37 must be filed within two months of the date of filing the Notice of Appeal (37 CFR 41.37(a)), or any extension thereof (37 CFR 41.37(e)), to avoid dismissal of the appeal. Since a Notice of Appeal has been filed, any reply must be filed within the time period set forth in 37 CFR 41.37(a).

AMENDMENTS

3. ☒ The proposed amendment(s) filed after a final rejection, but prior to the date of filing a brief, will not be entered because
   (a) ☒ They raise new issues that would require further consideration and/or search (see NOTE below);
   (b) ☐ They raise the issue of new matter (see NOTE below);
   (c) ☒ They are not deemed to place the application in better form for appeal by materially reducing or simplifying the issues for appeal; and/or
   (d) ☐ They present additional claims without canceling a corresponding number of finally rejected claims.
   NOTE: *See Continuation Sheet*. (See 37 CFR 1.116 and 41.33(a)).
4. ☐ The amendments are not in compliance with 37 CFR 1.121. See attached Notice of Non-Compliant Amendment (PTOL-324).
5. ☐ Applicant's reply has overcome the following rejection(s): _____.
6. ☐ Newly proposed or amended claim(s) _____ would be allowable if submitted in a separate, timely filed amendment canceling the non-allowable claim(s).
7. ☒ For purposes of appeal, the proposed amendment(s): a) ☒ will not be entered, or b) ☐ will be entered and an explanation of how the new or amended claims would be rejected is provided below or appended.
   The status of the claim(s) is (or will be) as follows:
   Claim(s) allowed: _____.
   Claim(s) objected to: _____.
   Claim(s) rejected: *1-54 and 56*.
   Claim(s) withdrawn from consideration: _____.

AFFIDAVIT OR OTHER EVIDENCE

8. ☐ The affidavit or other evidence filed after a final action, but before or on the date of filing a Notice of Appeal will not be entered because applicant failed to provide a showing of good and sufficient reasons why the affidavit or other evidence is necessary and was not earlier presented. See 37 CFR 1.116(e).
9. ☐ The affidavit or other evidence filed after the date of filing a Notice of Appeal, but prior to the date of filing a brief, will not be entered because the affidavit or other evidence failed to overcome all rejections under appeal and/or appellant fails to provide a showing a good and sufficient reasons why it is necessary and was not earlier presented. See 37 CFR 41.33(d)(1).
10. ☐ The affidavit or other evidence is entered. An explanation of the status of the claims after entry is below or attached.

REQUEST FOR RECONSIDERATION/OTHER

11. ☒ The request for reconsideration has been considered but does NOT place the application in condition for allowance because: See Continuation Sheet.
12. ☐ Note the attached Information Disclosure Statement(s). (PTO/SB/08 or PTO-1449) Paper No(s). _____
13. ☐ Other: _____.

Eric Coleman
Primary Examiner

Continuation Sheet (PTO-303) **Application No. 10/285,318**

Continuation of 3. NOTE: The proposed change in scope of the claims (e.g., addition of defining a calculation "at the at least one reconfigurable processor" and "wherein communications between said functional units is external communication protocol independent and internal communication independent") would necessitate a new search.

Continuation of 11. does NOT place the application in condition for allowance because: of the reasons stated in the final rejection. Also, the proposed added wherein clauses are not required by, or are not a consequence of, any element or step in the claims consequently it is merely intended use. The implementation by Khan using an array of processing elements does not require any change in protocol for communcation to perform any process with the array also There is no requirement that there would be the only one particular protocol that would allow implementation of the invention in Gupta. Also processing in Gupta and Khan was performed at at least one reconfigurable processor (e.g., see col. 17, lines 28-52 and col. 21, lines 22-29 of Gupta; and col. 7, line 7-col. 8, line 65 of Khan).

Client Matter No. 80404.0018
EFS-Web

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Serial No. 10/285,318 Application of: Jon M. Huppenthal and David E. Caliga Filed: October 31, 2002 Art Unit: 2183 Examiner: Coleman, Eric Attorney Docket No. SRC015 For: MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS	Confirmation No.: 1420 Customer No.: **25235**

SUPPLEMENTAL AMENDMENT AFTER FINAL

MAIL STOP AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the office communication mailed March 6, 2006, please amend the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims which begins on page 2 of this paper.

Remarks/Arguments begin on page 10 of this paper.

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1. (currently amended) A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor, the reconfigurable processor comprising a plurality of functional units, said method comprising:

defining a calculation at the at least one reconfigurable processor for said reconfigurable computing system;

instantiating at least two of said functional units at the at least one reconfigurable processor to perform said calculation wherein how many functional units and functional type of each functional unit is based on the calculation and wherein each functional unit at the at least one reconfigurable processor communications with each other functional unit at the at least one reconfigurable processor independent of external and internal communication protocols;

utilizing a first of said functional units to operate upon a subsequent data dimension of said calculation; and

substantially concurrently utilizing a second of said functional units to operate upon a previous data dimension of said calculation.

2. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple vectors in said calculation.

3. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple planes in said calculation.

4. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple time steps in said calculation.

5. (original) The method of claim 1 wherein said subsequent an previous data dimensions of said calculation comprise multiple grid points in said calculation.

6. (original) The method of claim 1 wherein said calculation comprises a seismic imaging calculation.

7. (original) The method of claim 1 wherein said calculation comprises a synthetic aperture radar imaging calculation.

8. (original) The method of claim 1 wherein said calculation comprises a JPEG image compression calculation.

9. (original) The method of claim 1 wherein said calculation comprises an MPEG image compression calculation.

10. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for a reservoir simulation.

11. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for weather prediction.

12. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for automotive applications.

13. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for aerospace applications.

14. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for an injection molding application.

15. (previously presented) The method of claim 1 wherein instantiating includes establishing a stream communication connection between functional units.

16. (original) The method of claim 1 wherein said calculation is comprises a structures calculation for structural analysis.

17. (original) The method of claim 1 wherein said calculation comprises a search algorithm for an image search.

18. (original) The method of claim 1 wherein said calculation comprises a search algorithm for data mining.

19. (original) The method of claim 1 wherein said calculation comprises a financial modeling application.

20. (original) The method of claim 1 wherein said calculation comprises an encryption algorithm.

21. (Canceled)

22. (original) The method of claim 1 wherein said calculation comprises a genetic pattern matching function.

23. (original) The method of claim 1 wherein said calculation comprises a protein folding function.

24. (original) The method of claim 1 wherein said calculation comprises an organic structure interaction function.

25. (original) The method of claim 1 wherein said calculation comprises a signal filtering application.

26. (currently amended) A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor comprising a plurality of functional units, said method comprising:

defining a first systolic wall comprising rows of cells forming a subset of said plurality of functional units;

computing at the at least one reconfigurable processor a value at each of said cells in at least a first row of said first systolic wall;

communicating said values between cells in said first row of said cells to produce updated values, wherein communicating said values is both internal and external communication protocol independent;

communicating said updated values to a second row of said first systolic wall, wherein communicating said updated values is both internal and external communication protocol independent; and

substantially concurrently providing said updated values to a first row of a second systolic wall of rows of cells in said subset of said plurality of functional units.

27. (original) The method of claim 26 wherein said values correspond to vectors in a computation.

28. (original) The method of claim 26 wherein said values correspond to planes in a computation.

29. (original) The method of claim 26 wherein said values correspond to time steps in a computation.

30. (original) The method of claim 26 wherein said values correspond to grid points in a computation.

31. (original) The method of claim 26 wherein said step of communicating said updated values to a second row of said first systolic wall is carried out without storing said updated values in an extrinsic memory.

32. (original) The method of claim 26 wherein said values correspond to a seismic imaging calculation.

33. (original) The method of claim 26 wherein said values correspond to a synthetic aperture radar imaging calculation.

34. (original) The method of claim 26 wherein said values correspond to a JPEG image compression calculation.

35. (original) The method of claim 26 wherein said values correspond to an MPEG image compression calculation.

36. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for a reservoir simulation.

37. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for weather prediction.

38. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for automotive applications.

39. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for aerospace applications.

40. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for an injection molding application.

41. (previously presented) The method of claim 26 wherein defining includes establishing a stream communication connection between functional units and wherein how many functional units and functional type of each functional unit is based on a computing algorithm within the reconfigurable computing system.

42. (original) The method of claim 26 wherein said values correspond to a structures calculation for structural analysis.

43. (original) The method of claim 26 wherein said values correspond to a search algorithm for an image search.

44. (original) The method of claim 26 wherein said values correspond to a search algorithm for data mining.

45. (original) The method of claim 26 wherein said values correspond to a financial modeling application.

46. (original) The method of claim 26 wherein said values correspond to an encryption algorithm.

47. (canceled)

48. (original) The method of claim 26 wherein said values correspond to a genetic pattern matching function.

49. (original) The method of claim 26 wherein said values correspond to a protein folding function.

50. (original) The method of claim 26 wherein said values correspond to an organic structure interaction function.

51. (original) The method of claim 26 wherein said values correspond to a signal filtering application.

52. (canceled)

53. (currently amended) The method of claim [[52]]26 wherein said reconfigurable computing system ~~further~~ comprises at least one microprocessor.

54. (currently amended) A method for data processing in a reconfigurable computing system, the reconfigurable computer system comprising at least one reconfigurable processor comprising a plurality of functional units, said method comprising:

performing a calculation at the at least one reconfigurable processor by a subset of said plurality of functional units to produce computed data;

passing said computed data from a first column of said calculation to a next column in said calculation, wherein said passing is both internal and external communication protocol independent;

evaluating a rate of change in at least one variable for each of said columns in said calculation;

continuing said calculation when said variable does not change for a particular column of said calculation; and

restarting said calculation at said column of said calculation where said variable does change.

55. (Canceled)

56. (Previously Presented) The method of claim 54 wherein how many functional units comprise the subset and functional type of each functional unit in said subset is based on the calculation and wherein the passing step is external communication protocol independent.

REMARKS/ARGUMENTS

Claims 1-54 and 56 were presented for examination and are pending in this application. In an Official Final Office Action dated March 6, 2006, claims 1-54, and 56 were rejected. The Applicants thank the Examiner for his consideration and address the Examiner's comments concerning the claims pending in this application below.

Applicants herein amend claims 1, 26, 53 and 54 and respectfully traverse the Examiner's rejections. Claims 21, 47, and 52 are presently canceled without prejudice. Claims 1-20, 22-46, 48-51 and 53, 54 and 56 are now pending in this application. The additional limitations brought into the independent claims place the claims in better condition for consideration on appeal and because they appear in dependent claims as filed, these amendments do not raise any new issues that would require further research by the Examiner. These changes are believed not to introduce new matter, and their entry is respectfully requested. The claims have been amended to expedite the prosecution and issuance of the application. In making these amendments, Applicants have not and are not narrowing the scope of the protection to which the Applicants consider the claimed invention to be entitled and do not concede, directly or by implication, that the subject matter of such claims was in fact disclosed or taught by the cited prior art. Rather, Applicants reserve the right to pursue such protection at a later point in time and merely seek to pursue protection for the subject matter presented in this submission.

35 U.S.C. §103(a) Obviousness Rejection of Claims

Claims 1-5, 15, 21, 26-31, 41, 47, 52, 53 and 56 were rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 6,385,757 ("Gupta") in view of U.S. Patent No. 5,274,832 ("Khan"). Applicants respectfully traverse these rejections in light of the aforementioned remarks and respectfully request reconsideration.

Section 103(a) of title 35 of the United States Code states that a patent may not be obtained if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which the subject matter pertains. See 35 U.S.C. §103(a). To form a *prima facie* case of obviousness under 35 U.S.C. §103 and in accord with section 2143 of the MPEP, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify a reference or to combine reference teachings. Second, there must be a reasonable expectation that the art suggested in the references cited by the Examiner will succeed in creating the claimed invention. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. These three criteria have not been met by the Examiner.

A. The Examiner provides neither explicit nor implicit reasons why one skilled in the art at the time of the Applicants' invention would modify Gupta with the teachings of Khan.

The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. In re Mills, 916 F.2d 680, 16 U.S.P.Q.2d 1430 (Fed. Cir., 1990). According to the Federal Circuit, this motivation may be found implicitly or explicitly: 1) in the prior art references themselves; 2) in the knowledge of those of ordinary skill in the art that certain references, or disclosures in those references, are of special interest or importance in the field; 3) or from the nature of the problem to be solved leading inventors to look to reference relating to possible solutions to that problem. See Ruiz v. A.B. Chance Co., 234 F.3d 654, 57 U.S.P.Q.2d 1161 (Fed. Cir. (Mo.), 2000). To prevent the use of hindsight based on the Applicants' invention to defeat the

patentability of the Applicants' invention, the Examiner must show a motivation to combine the references that creates the case of obviousness. "In other words, the examiner must show reasons that the skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would select the elements from the cited prior art references for combination in the manner claimed." In re Rouffet, 149 F.3d 1350, 47 U.S.P.Q.2d 1453 (Fed. Cir., 1998). Thus absent some teaching, suggestion or incentive supporting the proposed combination of art, obviousness cannot be established.

The Examiner asserts that it would be obvious for one of ordinary skill in the art at the time of the Applicants' invention to combine the teachings of Gupta and Khan. The Examiner attempts to support his assertion by stating that one skilled in the DP art would have been motivated to incorporate the three dimensional array operations of Khan reference into the Gupta system to allow the combined system to be able to perform calculations on more complicated (three dimensional) problems. A careful word search of Gupta and Khan reveals no such motivating statement thus the Applicants assume the Examiner finds this motivation inherent. The Court in Rouffet stated that to "prevent the use of hindsight based on the invention to defeat patentability of the invention, this court requires the examiner to show a motivation to combine the references that create the case of obviousness." Id. As in the present invention, the examiner in Roufett relied on the high level of skill in the art to provide the necessary motivation. Finding such motivation absent, the Rouffet Court stated that "if such rote invocation could suffice to supply a motivation to combine, the more sophisticated scientific fields would rarely, if ever, experience a patentable technical advance." Id.

The Federal Circuit has repeatedly warned against the use of the Applicants' invention as a blueprint by which to build a case of obviousness.

The Examiner offers no explanation of the specific understanding or principle within the knowledge of one skilled in the DP art that would motivate one with no knowledge of the Applicants' invention to combine the teachings of Gupta and Khan to create the Applicants' invention. A system to be able to perform calculations on more complicated problems applies to any system. There is no foundation to motivate one skilled in the relevant art at the time of the Applicants' invention to combine the teachings of Gupta with reference to Very Long Instruction Word processors with that of a systolic array for multidimensional matrix computations as taught by Khan other than the Applicants' invention. The Examiner's use of hindsight is improper and can not be used to support a rejection under 35 U.S.C. §103.

B. Modifying Gupta by the teachings of Khan fails to provide a reasonable expectation of success to produce the Applicants' claimed invention as a whole because neither Gupta nor Khan address boundary interactions and communications between functional units conducting parallel computations in a reconfigurable processor based system.

A proper analysis under 35 U.S.C. §103 includes the determination of "whether the prior art would also have revealed that in so making or carrying out, those of ordinary would have a reasonable expectation of success." Noelle v. Lederman, 355 F.3d 1343, 69 U.S.P.Q.2d 1508 (Fed. Cir., 2004). While an absolute expectation of success is not necessary, the combined art must provide a reasonable expectation that one skilled in the art will succeed in making the claimed subject matter as a whole. "To have a reasonable expectation of success, one must be motivated to do more than merely to vary all parameters or try each of numerous possible choices until one possibly arrived at a successful result, where the prior art gave either no indication of which parameters were critical or no direction as to which of many possible

choices is likely to be successful. Medichem, S.A. v. Rolabo, S.L. 437 F.3d 1157, *1165 (C.A.Fed. (N.Y.),2006). Foster in view of O'Sullivan provides no indication, no direction and no such expectation of success.

The Examiner fails to offer any suggestion that one skilled in the art would reasonably expect a modification of Gupta based on the teachings of Khan would succeed in creating the Applicants' claimed invention. The prior art and surrounding circumstances must provide a reasonable reason to do, not a reasonable reason to try to do. Based on what has been accomplished in the art up to the time of the invention, and specifically what is suggested and taught in Gupta and Khan, there is no suggestion of a direction on how to proceed to produce the Applicants' invention. Countless objective pieces of evidence exist supporting the computing industries goal to increase computing speed, efficiency, and bandwidth. Parallel processor computing in one such advance. However, a reasonable combination of Gupta and Khan would teach a multiple processor system wherein each processor would be allocated a small portion of the problem in one or more cells. The results of each cell must interact to pass along intermediary results leading to the final computation via internal and external communication protocols. As taught by Kahn and Gupta, this would necessitate numerous chips, busses, and other I/O operations that would operate of much lower computational speeds than that offered by the Applicants' invention. Advancements in computing hardware have progressed to a point where the bandwidth and speed of computing is soon to be limited by the speed of light. In such an environment, even a 10% increase in computing capability is heralded as a major achievement. Significantly, the Applicants' invention utilizes (and claims) reconfigurable processors that are independent of such limiting communication protocols. The Applicants' invention, which the Examiner asserts is obvious by the teachings of Gupta and Khan, advances processing speed, through the use of reconfigurable processors as currently

claimed, by more than three fold of that taught by Gupta or Khan. One skilled in the art at the time of the Applicants' invention would not reasonably expect that combining the teachings of Gupta with the teachings of Khan would produce such a result.

C. The Examiner fails to consider the claimed subject matter as a whole in making his obviousness rejection.

One of the hallmarks of 35 U.S.C. §103(a) is that for an invention to be unpatentable, the differences between the prior art and the claimed subject matter taken as a whole must be obvious to one skilled in the art. The Examiner fails to consider the invention as a whole and rather dissects and attacks each element individually. As has been repeatedly voiced by the Federal Circuit, "In determining obviousness, the invention must be considered as a whole without the benefit of hindsight, and the claims, must be considered in their entirety." Rockwell Intern. Corp. v. U.S., 147 F.3d 1358, 47 U.S.P.Q.2d 1027 (Fed. Cir., 1998). "In making the assessment of differences, section 103 specifically requires consideration of the claimed invention 'as a whole'". Ruiz v. A.B. Chance Co., 357 F.3d 1270, 69 U.S.P.Q.2d 1686 (Fed. Cir. (Mo), 2004).

The Examiner argues Khan suggests utilizing an array of functional units to operate on a subsequent data dimension of the calculation and substantially concurrently using a second of said array of functional units to operate on a previous data dimension of the same calculation. To support his argument, the Examiner turns to Khan Col.4, lines 35-62 and Col 12 lines 15-55. In this section, according to the Examiner, Khan teaches operating on three dimensions using plural two dimensional arrays that operate concurrently on respective dimensions and are coupled together to produce the three dimensional array. The Examiner suggests that during a pipelined operation, plural functional units in an array operate concurrently and when the pipelines are in plural directions or dimensions then the concurrency extends to plural

dimensions. The Examiner, however, fails to step back and take the Applicants' invention as a whole. In the Applicants' invention, concurrent to the first functional units operating on a subsequent set of data, a second set of function units is operating on a previous set of data. The concurrent pipelined operations described by the Examiner and as taught by Khan is a serialized concurrent approach which does not permit a second set of functional units operate on a previous set of data.

Section 103 requires that the differences in the prior art as compared to the claimed subject matter as a whole must be so slight as to make the claimed subject matter obvious. Taking the claimed subject matter as a whole, the differences between what is suggested by Gupta in view of Khan are enormous.

D. Each and every element of the claimed invention is not disclosed in the combined references, namely Gupta and Khan.

Gupta in view of Khan also fails to teach or suggest computing data flows between functional units of a single reconfigurable processor. Gupta and Khan teach traditional parallel processing with sequential processing of data between one processor, functional unit, or cell and an adjacent processor, functional unit, or cell.

The Applicants' invention calculates the number of required functional units and the type of units entirely within a single reconfigurable processor to accomplish the processing task. Computations performed by these functional units are shared within the processor and thus never need leave the single reconfigurable processor environment. This consolidation of computations eliminates the need for external and internal communication protocols. Such an adaptive (reconfigurable) processor-based system is distinct from that taught by Khan or Gupta. Khan and Gupta do not teach performing these calculations in a single processor. Rather, multiple processors are taught that would require consideration for both internal and external communication protocols. The Examiner states in the Advisory Action that

there is no requirement in Gupta or Khan that would require only one particular protocol that would allow implementation. Khan teaches combining multiple processors to achieve pipelining processing. See Khan Col. 5, lines 3-16. While Khan teaches a minimization of global interactions, the interactions nonetheless exist. As one skilled in the art at the time of the Applicants' invention would recognize, these interactions necessitate a common communication protocol. The Applicants' invention operates independent of these protocols.

The Applicants, therefore, reject the sweeping and unsupported conclusion by the Examiner with respect to claims 21, 47 and 56 (now incorporated into independent claims 1, 26 and 54). The Examiner states in his final rejection that it is "anticipated that in the implementation of a system using Khan and Gupta teachings that the communication between processors would have been protocol independent." (emphasis added) There is no basis for this conclusion. Again the Examiner uses hindsight provided by the Applicants to reach his conclusion of obviousness. The invention as claimed states that communication between functional units is communication protocol independent. This is a function of the inherent nature of a reconfigurable processor. Similarly, the inherent nature of coupling multiple processors as taught by Khan is communications protocol dependent. The Applicants submit that this limitation is not taught or suggested in Khan or Gupta and as incorporated into claims 1, 26 and 54, place these claims in condition for allowance.

The Applicants reiterate that the Applicants' invention builds functional units of a reconfigurable processor-based system based on the algorithms being used in the calculations in a single reconfigurable processor. The type of each functional unit and the total number of functional units created is unique for each assigned task. This is distinct from the teachings of Gupta of a system using a Very Long Instruction Word ("VLIW") processor. VLIW processors do have the ability to use

multiple arithmetic functional units one at a time but the set of functional units are limited and fixed within the VLIW processor.

Gupta teaches a system to generate an instruction format that is used to control a processor control path in what is called parallel instruction computing. This instruction-level parallelism issues several operations per instruction to multiple functional units to control a processors data path. As the Examiner admits in a previous Office Action, Gupta fails to teach a substantially concurrent use of data dimensions during a calculation. The Applicants reassert their disagreement with the Examiner's conclusion that Khan teaches this noted insufficiency of Gupta. Khan teaches a systolic sequential parallel approach to processing using multiple processors. The Applicants concur that plural pipelines moving in different directions extend the processing described in Khan in plural dimensions, but disagree that such an extension is the equivalent to concurrent operations as claimed by the Applicants. Each pipeline is, by definition, a serialized operation. While multiple pipelines may be operating concurrently, each pipeline individually still operates and communicates in a serial fashion.

The Applicants' invention utilizes available resources to have an application evaluate a problem in a concurrent data flow sense and not in a pipeline sense. That is, it will "pass" a subsequent dimension of a given problem through a first loop of logic concurrently with the previous dimension of data of the given problem being processed through a second loop. This type of concurrent operation cannot occur in the serialized pipeline operation described in Khan nor is it suggested or taught in Gupta. Accordingly, the Applicants submit that independent claims 1, 26 and 54 are patentable over Gupta in view of Khan.

The Examiner also suggests that Khan and Gupta teach the use of at least one reconfigurable processor. As discussed earlier, functional units in a VLIW processor are limited and fixed thus the claimed limitation that the number and type of functional units instantiated in the reconfigurable processor based on the

calculation is not possible in a VLIW as taught by Gupta. The section of text of Khan cited by the Examiner, Col. 7 line 7 – Col. 8 line 65 mentions the reconfiguration of a single dimensional vector signal set to a two dimensional vector signal set. There is no apparent suggestion or teaching of the use of a reconfigurable processor or the instantiation of functional units in said reconfigurable processor.

E. Conclusion

For at least the aforementioned reasons, the Applicants submit that claims 1, 26 and 54, are patentable over Gupta in view of Khan. Reconsideration and withdrawal of the rejection of the claims under 35 U.S.C. § 103(a) is respectfully requested.

Claims 2-5, 15, 21, 27-31, 41, 47, 53 and 56 depend from claims 1, 26 or 54 and are, for at least the same aforementioned reasons, patentable over Gupta in view of Khan. The Applicants respectfully request the rejections be withdrawn and the claims reconsidered.

Claims 19 and 45 were rejected under 35 U.S.C § 103(a) as being unpatentable over Gupta in view of Khan and in further view of U.S. Patent No. 4,872,133 ("Leeland"). Leeland fails to rectify the aforementioned deficiencies of Gupta and Khan with respect to independent claims 1 and 26 and therefore, as claims 19 and 45 depend from claims 1 and 26 respectively, the Applicants submit claims 19 and 45 are patentable over Gupta in view of Khan in further view of Leeland.

The Examiner also rejects dependent claims 10-14, 16, 36-40, 42 and independent claim 54 under 35 U.S.C. § 103(a) as being unpatentable over Gupta in view of Khan in further view of U.S. Patent No. 5,072,371 ("Benner"). The Applicants traverse these rejections. For at least the aforementioned reasons, the Applicants submit that Brenner fails to resolve the deficiencies noted in Gupta and Kahn. Claims 10-14, 16 and 36-40, 42, which depend from claims 1 and 26

respectively, are therefore patentable over Gupta in view of Khan in further view of Benner.

The Examiner's rejection of claim 54 as being unpatentable over Gupta in view of Khan and in further view of Benner is traversed in light of the present amendments and the aforementioned remarks.

Claims 6-9, 17-18, 20-25, 32-35, 43-44 and 46-51 are rejected by the Examiner under 35 U.S.C. § 103(a) as being unpatentable over Gupta and Kahn as applied to claims 1-2 and 26 and in further view of U.S. Patent No. 4,962,381 ("Helbig"), or in further view of U.S. Patent No. 5,784,108 ("Skaletsky"), or in further view of U.S. Patent No. 6,061,706 ("Gai"), respectively. As all of these claims depend from either independent claim 1 or 26, the Applicants submit, for at least the aforementioned reasons, each is patentable over Gupta and Kahn. Reconsideration is respectfully requested.

Based on the above amendment and the following remarks, Applicants respectfully request that the Examiner reconsider all outstanding rejections and withdraw them.

In view of all of the above, the claims are now believed to be allowable and the case in condition for allowance which action is respectfully requested. Should the Examiner be of the opinion that a telephone conference would expedite the prosecution of this case, the Examiner is requested to contact Applicants' attorney at the telephone number listed below.

No fee beyond that associated with the RCE is believed due for this submittal. However, any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

2 June, 2006

Michael C. Martensen, Reg. No. 46,901
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5910 Tel
(303) 899-7333 Fax

Office Action Summary	Application No.	Applicant(s)	
	10/285,318	HUPPENTHAL ET AL.	
	Examiner	Art Unit	
	Eric Coleman	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *02 June 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-20,22-46,48-51,53,54 and 56* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-20,22-46,48-51,53,54 and 56* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 112

The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

Claims 1-20,22-46,48-51,53,54,56 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contain subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention. Claim 1 (in lines 10-12) contains the language "wherein the functional unit at the at least one reconfigurable processor communications with each other functional unit at the at least one reconfigurable processor independent of external and internal communication protocols" Claim 26 (in lines 10-11) contains the language " wherein communicating said values is both internal and external communication protocol independent" and (in lines 13-14) wherein communicating said updated values is both internal and external communication protocol independent" and claim 54 (in lines 8-9) contains the language "wherein said passing is both internal and external communication protocol independent". The dependent claims 2-20,22-25,27-46,48-51,53 and 56 respectively contain the language in the corresponding independent claim above.

The language detailed above provides for a functioning of the reconfigurable computing system that is independent of the internal and external communication

protocols. The Dictionary, of Computers Information Processing & Telecommunications 2nd Edition, on page 496, defines protocol as (1) a specification for the format and relative timing of information exchanged between communicating parties; (2) the set of rules governing the operation of functional units of a communication system that must be followed if communication is to be achieved". The originally filed application does not provide for the operation of the system that would operate in a properly timed way where the data would have properly been communicated between functional units in a format providing proper operation of the system without use of internal and external protocols that the system would require for operation. This is especially true since the system is reconfigurable. The communications requirements would not always be the same so without some type of use of an internal or external protocol there would have had to have been some other means to provided for proper communications. This was not disclosed in the originally filed application. Therefore the written description requirement has not been met.

Also claim 1 contains the language (in line 9) wherein how many functional type of each functional unit is based on the calculation; claim 41 (lines 3-4) contains "functional type of each functional unit is based on a computing algorithm"; claim 56 (lines 2-3) contains "functional type of each functional unit in said subset is based on the calculation". These features were not described in the originally filed application. Therefore additionally for these reasons the written description has not been met.

Claims 1-20,22-46,48-51,53,54,56 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention.

Claims 1-20,22-46,48-51,53,54,56 rejected under 35 U.S.C. 112, first paragraph, because the specification, while being enabling for a reconfigurable processor that communicates using internal and external protocols, The specification does not enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to , make and use the invention commensurate in scope with these claims.

Claim 1 (in lines 10-12) contains the language "wherein the functional unit at the at least one reconfigurable processor communications with each other functional unit at the at least one reconfigurable processor independent of external and internal communication protocols" Claim 26 (in lines 10-11) contains the language " wherein communicating said values is both internal and external communication protocol independent" and (in lines 13-14) wherein communicating said updated values is both internal and external communication protocol independent" and claim 54 (in lines 8-9) contains the language "wherein said passing is both internal and external communication protocol independent". The dependent claims 2-20,22-25,27-46,48-51,53 and 56 respectively contain the language in the corresponding independent claim above.

The language detailed above provides for a functioning of the reconfigurable computing system that is independent of the internal and external communication protocols. The Dictionary, of Computers Information Processing & Telecommunications 2nd Edition, on page 496, defines protocol as (1) a specification for the format and relative timing of information exchanged between communicating parties; (2) the set of rules governing the operation of functional units of a communication system that must be followed if communication is to be achieved". The originally filed application does not provide for the operation of the system that would operate in a properly timed way where the data would have properly been communicated between functional units in a format providing proper operation of the system without use of internal and external protocols that the system would require for operation. This is especially true since the system is reconfigurable. The communications requirements would not always be the same so without some type of use of an internal or external protocol there would have had to have been some other means to provided for proper communications. This was not disclosed in the originally filed application. The providing of some means to implement the originally disclosed invention with communications independent of the internal and external protocols would have required undue experimentation. Therefore as claimed the invention is not enabling.

Also, claim 1 contains the language (in line 9) wherein how many functional type of each functional unit is based on the calculation; claim 41 (lines 3-4) contains "functional type of each functional unit is based on a computing algorithm"; claim 56 (lines 2-3) contains "functional type of each functional unit in said subset is based on the

calculation". The originally disclosed invention contains a single type of functional unit (e.g., see fig. 2) arranged in a reconfigurable array. This would required the system to comprises plural types of functional units. Consequently it would have required undue experimentation for one of ordinary skill to incorporate the feature functional type of each functional unit is based on a computing algorithm. Therefore additionally for these reasons the invention as claimed is not enabling.

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1-5,15,21,26-31,41,47,52,53,56 are rejected under 35 U.S.C. 103(a) as being unpatentable over DeHon (Patent No. 5,956,518) in view of Khan (US Patent No. 5,274,832).

3. DeHon taught the invention substantially as claimed including a data processing ("DP") system comprising: defining a calculation for a reconfigurable computing system instantiating the performance of at least two array functional units (101,102)(e.g., see fig. 5 and (e.g., see col. 5,lines 3-56) to perform the calculation.

4. DeHon taught systolic pipelined operation of the system (e.g., see col. 16, lines 1-64) but did not expressly detail utilizing the array functional units to operate on a subsequent data dimension of the calculation and substantially concurrently using the

second of the array units to operate on a previous data dimension of the calculation. Khan however taught operating on three dimensions using plural two dimensional arrays that operate concurrently on respective dimensions and are coupled to together to produce the three dimensional array (e.g., see col. 4, lines 35-62 and col. 12, lines 15-55).

5. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of DeHon and Khan. One of ordinary skill would have been motivated to incorporate the three dimensional array operation of the Khan reference into the DeHon system to allow the combined system to be able to perform calculations on more complicated (three dimensional) problems (e.g., see col. 5, lines 3-11 of DeHon, and col. 4, lines 54-63 of Khan).

6. Claim 1,26,54,56 comprises the limitation of reconfigurable computing system communicates between functional units independent of internal and external communication protocols. Since the Khan and DeHon system taught systems that did not use external protocols to communicate between the processors on chip it is would have been obvious to one of ordinary skill that in the implementation of a system using the DeHon and Khan teachings that the communication between processors would have been protocol independent. The limitation of independent of the internal communication protocols as understood with respect to the originally filed application provides for the direct communication between functional units. DeHon taught the direct communication between functional units (e.g., see figs. 1,2,4,5,7) and therefore the DeHon and Khan system meet the claimed limitation.

7. Claims 1,41,56 have the limitation of wherein how many functional units and functional type of each functional unit is based on the calculation (which comprises the algorithm). As to this limitation Khan taught specific selection of the number of processing elements (which correspond to claimed functional units) are different depending on whether the calculation was a two dimensional or three dimensional calculation (e.g., see col. 5, lines 17-30). As to the type of functional unit being based on the calculation the system is a special purpose system that uses a specific type of functional unit namely processing elements that perform systolic array calculations readily and where connections and transfer of data for performing the calculation is readily done. Therefore in the implementation of the Khan teachings the type of functional unit is based on the type of calculations and the algorithm that were to be performed by the system (e.g., two dimensional algorithm or three dimensional algorithm) (e.g., see col. 5, lines 32-49).

8. As to instantiating including establishing a stream communication connection between functional units (claims 15,41) Khan taught minimizes interconnections of processing elements and the matrix and vector signal subsets are specifically formed so that they need to be inputted to only one row and one columns and yet still be properly processing systolically along all dimensions within the array (e.g., see col. 5, lines 2-48). Consequently the stream of communication between functional units is established as the interconnections are made and data is transferred systolically in at least one stream between processors.

9. As to the further limitations of claim 26, Khan taught (e.g., see fig. 8) a three dimensional systolic array with connections between processors in three dimensions and the selection is done to minimize global interconnections.

10. As to claim 2-5,27-30 Khan taught the calculation comprising plurality of planes, and grid points and plural time-steps and vectors (e.g., see fig. 8 and col. 12, lines 15-55). As per claim 31, the system taught by Khan shows direct connection between the processing elements in the array and therefore the storing of data to an extrinsic memory (i.e., outside the array) would have been unnecessary when the transfer of data between columns was performed (e.g., see fig. 8).

11. As to the limitations of claims 52 and 53 the reconfigurable systolic processor would have been able to adapt to the application a therefore would have been an adaptive processor. As to the processor comprising a microprocessor one of ordinary skill would have been motivated to implement the systolic processor as described above as an microprocessor at least to take advantage of the reduced cost and reduced system size as was well known in the art at the time of the claimed invention.

12. Claims 19, 45 are rejected under 35 U.S.C. 103(a) as being unpatentable over DeHon and Khan as applied to claims 1-2,26 above, and further in view of Leeland (US patent No. 4,872,133).

13. Leeland taught calculation comprised a financial application modeling using a spreadsheet application (e.g., see col. 5, lines 3-32).

14. It would have been obvious to one of ordinary skill in the DP art to combine the

teachings of Leeland and DeHon. Speadsheet applications were well known applications to be conventionally used for financial processing financial data. One of ordinary skill would have been motivated to incorporate the Leeland teaching of financial spreadsheet application for an array processor in order to provide an additional use for the combined system.

15. Claim 10-14,16 and 36-40,42,54 rejected under 35 U.S.C. 103(a) as being unpatentable over DeHon and Khan as applied to claims 1-2,15,26 above, and further in view of Benner (US Patent No. 5,072,371).

16. Benner taught the calculation comprising fluid flow calculation and structural analysis (e.g., see col. 22, lines 35-52).

17. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Benner and DeHon. DeHon taught the configuring the system as a systolic pipeline (e.g., see col. 16, lines 10-46). One of ordinary skill would have been motivated to incorporate the Benner teaching of fluid flow and structural analysis applications for an array processor in order to provide an additional uses for the combined system.

18. As to the limitation in claim 54 of performing a calculation unit a variable changed is value in a system processing a restarting at that value The Benner system taught systolically performing calculations on fluid flow. Since in such a problem one of ordinary skill would at times be interested when a change in the data occurred and adjust the calculation to pin point the calculation around that certain point then one of

ordinary skill would have been motivated to operate the Benner and DeHon and Khan system to process systolically until a change in data occurred and then restart the calculation at the point of the change to better determine the magnitude of the change in data.

19. Claim 6-9,25,32-35,51 are rejected under 35 U.S.C. 103(a) as being unpatentable over DeHon and Khan as applied to claims 1-2,26 above, and further in view of Helbig (US patent No. 4,962,381).

20. Helbig taught the application of a systolic processor for radar, medical ultrasound and other imaging applications (e.g., see col. 1, lines 1-5) Clearly this would have also comprised images processed by standard MPEG and JPEG standards.

21. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Helbig and DeHon. DeHon taught the configuring the system as a systolic pipeline (e.g., see col. 16, lines 10-46). Therefore one of ordinary skill would have been motivated to incorporate the Helbig teaching of radar, medical ultrasound and other imaging applications for an systolic processor in order to provide an additional uses for the combined system.

22. As to the limitation of claims 25 and 51, since signal filtering would have been associated with the applications taught by Helbig such as radar then one of ordinary skill would have been motivated to use the Helbig systolic processor in signal filtering applications.

23. Claim 17,18,22-24,43,44,48-50 are rejected under 35 U.S.C. 103(a) as being unpatentable over DeHon and Khan as applied to claims 1-2,26 above, and further in view of Skaletsky (US patent No. 5,784,108).

24. Skaletsky taught using an systolic processor for processing search algorithm for image search such as when a best match was to be found and clearly this would have been applicable to data mining as these are similar applications (e.g., see col. 3, line 13-col. 4, line 57).

25. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Skaletsky and DeHon. . DeHon taught the configuring the system as a systolic pipeline (e.g., see col. 16, lines 10-46). Therefore one of ordinary skill would have been motivated to incorporate the Skaletsky teaching of search algorithm applications for a systolic processor in order to provide an additional uses for the combined system.

26. As to the limitations of claims 22-24,48-50 in light of the search algorithm teaching especially for finding a best match for data then the use of systolic processors for similar applications such as the genetic pattern matching, protein folding and organic structure interaction would have been an obvious uses for systolic processors (such as taught by Skaletsky and DeHon) to one of ordinary skill in the DP art.

27. Claim 20,46, are rejected under 35 U.S.C. 103(a) as being unpatentable over DeHon and Khan as applied to claims 1-2,26 above, and further in view of Gai (US patent No. 6,061,706).

28. Gai taught use of systolic processors in encryption/decryption applications to speed the encryption/decryption of public keys (e.g. see col. 1, lines 25-41.

29. It would have been obvious to one of ordinary skill in the DP art to combine the teachings of Gai and DeHon. One of ordinary skill would have been motivated to incorporate the Gai teaching of encryption and decryption applications for an systolic processor in order to provide an additional uses for the combined system.

Response to Arguments

Applicant's arguments with respect to claims 1-20,22-46,48-51,53,54,56,56 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Casselman (patent No. 6,289,440) disclosed a virtual computer of plural FPG's successively reconfigured in response to a succession of inputs (e.g., see abstract).

Mirsky (patent No. 5,915,123) disclosed a system for controlling configuration memory contexts (e.g., see abstract).

Pechanek (patent No. 5,640,586) disclosed a scalable parallel group parititioned diagonal fold tree computing apparatus (e.g., see abstract).

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eric Coleman whose telephone number is (571) 272-4163. The examiner can normally be reached on Monday-Thursday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

EC

ERIC COLEMAN
PRIMARY EXAMINER

Client Matter No. 80404.0018
EFS-Web

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Serial No. 10/285,318 Application of: Jon M. Huppenthal and David E. Caliga Filed: October 31, 2002 Attorney Docket No. SRC015 For: MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS	Confirmation No.: 1420 Art Unit: 2183 Examiner: Coleman, Eric Customer No.: **25235**

AMENDMENT

MAIL STOP AMENDMENT
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the office communication mailed August 17, 2006, please amend the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims which begin on page 2 of this paper.

Remarks/Arguments begin on page 11 of this paper.

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1. (currently amended) A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor, the reconfigurable processor comprising a plurality of functional units, said method comprising:

~~defining a calculation at the at least one reconfigurable processor for said reconfigurable computing system;~~

transforming an algorithm into a calculation that is systolically implemented by said reconfigurable computing system at the at least one reconfigurable processor;

instantiating at least two of said functional units at the at least one reconfigurable processor to perform said calculation wherein ~~how many~~ only functional units needed to solve ~~and functional type of each functional unit is based on~~ the calculation are instantiated and wherein each instantiated functional unit at the at least one reconfigurable processor ~~communications~~ interconnects with each other instantiated functional unit at the at least one reconfigurable processor ~~independent of external and internal communication protocols~~ based on reconfigurable routing resources within the at least one reconfigurable processor as established at instantiation, and wherein systolically linked lines of code of said calculation are instantiated as clusters of functional units within the at least one reconfigurable processor;

utilizing a first of said instantiated functional units to operate upon a subsequent data dimension of said calculation forming a first computational loop; and

substantially concurrently utilizing a second of said instantiated functional units to operate upon a previous data dimension of said calculation forming a second computational loop wherein said systolic implementation of said calculation enables said first computational loop and said second computational loop execute concurrently and pass computed data seamlessly between said computational loops.

2. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple vectors in said calculation.

3. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple planes in said calculation.

4. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple time steps in said calculation.

5. (original) The method of claim 1 wherein said subsequent an previous data dimensions of said calculation comprise multiple grid points in said calculation.

6. (original) The method of claim 1 wherein said calculation comprises a seismic imaging calculation.

7. (original) The method of claim 1 wherein said calculation comprises a synthetic aperture radar imaging calculation.

8. (original) The method of claim 1 wherein said calculation comprises a JPEG image compression calculation.

9. (original) The method of claim 1 wherein said calculation comprises an MPEG image compression calculation.

10. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for a reservoir simulation.

11. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for weather prediction.

12. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for automotive applications.

13. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for aerospace applications.

14. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for an injection molding application.

15. (previously presented) The method of claim 1 wherein instantiating includes establishing a stream communication connection between functional units.

16. (original) The method of claim 1 wherein said calculation is comprises a structures calculation for structural analysis.

17. (original) The method of claim 1 wherein said calculation comprises a search algorithm for an image search.

18. (original) The method of claim 1 wherein said calculation comprises a search algorithm for data mining.

19. (original) The method of claim 1 wherein said calculation comprises a financial modeling application.

20. (original) The method of claim 1 wherein said calculation comprises an encryption algorithm.

21. (canceled)

22. (original) The method of claim 1 wherein said calculation comprises a genetic pattern matching function.

23. (original) The method of claim 1 wherein said calculation comprises a protein folding function.

24. (original) The method of claim 1 wherein said calculation comprises an organic structure interaction function.

25. (original) The method of claim 1 wherein said calculation comprises a signal filtering application.

26. (currently amended) A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor comprising a plurality of functional units, said method comprising:

transforming an algorithm into a calculation that is systolically implemented by said reconfigurable computing system at the at least one reconfigurable processor wherein systolically linked lines of code of said calculation are instantiated as walls of functional units within the at least one reconfigurable processor;

defining a first systolic wall comprising rows of cells forming a subset of said plurality of functional units;

computing at the at least one reconfigurable processor a value at each of said cells in at least a first row of said first systolic wall substantially concurrently;

communicating said values between cells in said first row of said cells to

produce updated values, wherein communicating said values is based on reconfigurable routing resources within the at least one reconfigurable processor ~~both internal and external communication protocol independent~~;

communicating said updated values substantially concurrently to a second row of said first systolic wall, wherein communicating said updated values is ~~both internal and external communication protocol independent~~ based on reconfigurable routing resources within the at least one reconfigurable processor; and

~~substantially concurrently providing~~ communicating said updated values substantially concurrently to a first row of a second systolic wall of rows of cells in said subset of said plurality of functional units, wherein communicating said updated values is based on reconfigurable routing resources within the at least one reconfigurable processor and wherein said first systolic wall of rows of cells and said second wall of rows of systolic cells execute substantially concurrently and pass computed data seamlessly between said systolic walls.

27. (original) The method of claim 26 wherein said values correspond to vectors in a computation.

28. (original) The method of claim 26 wherein said values correspond to planes in a computation.

29. (original) The method of claim 26 wherein said values correspond to time steps in a computation.

30. (original) The method of claim 26 wherein said values correspond to grid points in a computation.

31. (original) The method of claim 26 wherein said step of communicating said updated values to a second row of said first systolic wall is carried out without storing said updated values in an extrinsic memory.

32. (original) The method of claim 26 wherein said values correspond to a seismic imaging calculation.

33. (original) The method of claim 26 wherein said values correspond to a synthetic aperture radar imaging calculation.

34. (original) The method of claim 26 wherein said values correspond to a JPEG image compression calculation.

35. (original) The method of claim 26 wherein said values correspond to an MPEG image compression calculation.

36. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for a reservoir simulation.

37. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for weather prediction.

38. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for automotive applications.

39. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for aerospace applications.

40. (original) The method of claim 26 wherein said values correspond to a fluid flow calculation for an injection molding application.

41. (currently amended) The method of claim 26 wherein defining includes establishing a stream communication connection between functional units and wherein ~~how many~~ only functional units ~~and functional type of each functional unit is~~

~~based on a computing algorithm within the reconfigurable computing system~~ needed to solve the calculations are instantiated.

42. (original) The method of claim 26 wherein said values correspond to a structures calculation for structural analysis.

43. (original) The method of claim 26 wherein said values correspond to a search algorithm for an image search.

44. (original) The method of claim 26 wherein said values correspond to a search algorithm for data mining.

45. (original) The method of claim 26 wherein said values correspond to a financial modeling application.

46. (original) The method of claim 26 wherein said values correspond to an encryption algorithm.

47. (canceled)

48. (original) The method of claim 26 wherein said values correspond to a genetic pattern matching function.

49. (original) The method of claim 26 wherein said values correspond to a protein folding function.

50. (original) The method of claim 26 wherein said values correspond to an organic structure interaction function.

51. (original) The method of claim 26 wherein said values correspond to a signal filtering application.

52. (canceled)

53. (previously presented) The method of claim 26 wherein said reconfigurable computing system comprises at least one microprocessor.

54. (currently amended) A method for data processing in a reconfigurable computing system, the reconfigurable computer system comprising at least one reconfigurable processor comprising a plurality of functional units, said method comprising:

transforming an algorithm into a calculation that is systolically implemented by said reconfigurable computing system at the at least one reconfigurable processor wherein systolically linked lines of code of said calculation are instantiated as subsets of said plurality of functional units within the at least one reconfigurable processor forming columns of said calculation;

performing said calculation at the at least one reconfigurable processor by said subsets ~~a subset~~ of said plurality of functional units to produce computed data;

~~passing~~ exchanging said computed data between [[from]] a first column of said calculation [[to]]and a next column in said calculation, wherein said exchanging ~~passing~~ is based on reconfigurable routing resources within the at least one reconfigurable processor and wherein execution of said subsets of said plurality of function units occurs concurrently and said computed data is seamlessly passed between said first column of said calculation and said second column of said calculation ~~both internal and external communication protocol independent~~;

evaluating a rate of change in at least one variable for each of said columns in said calculation;

continuing said calculation when said variable does not change for a particular column of said calculation; and

restarting said calculation at said column of said calculation where said variable does change.

55. (canceled)

56. (currently amended) The method of claim 54 wherein how many functional units comprise the subset and functional type of each functional unit in said subset is based on the calculation ~~and wherein the passing step is external communication protocol independent~~.

REMARKS/ARGUMENTS

Claims 1-20, 22-46, 48-51, 53, 54 and 56 were presented for examination and are pending in this application. In an Official Office Action dated August 17, 2006, claims 1-20, 22-46, 48-51, 53, 54 and 56 were rejected. The Applicant thanks the Examiner for his consideration and addresses the Examiner's comments concerning the claims pending in this application below.

Applicant herein amends claims 1, 26, 41, 54 and 56 and respectfully traverses the Examiner's prior rejections. No claims are presently canceled and no new claims are presently added. These changes are believed not to introduce new matter, and their entry is respectfully requested. The claims have been amended to expedite the prosecution and issuance of the application. In making this amendment, Applicant has not and is not narrowing the scope of the protection to which the Applicant considers the claimed invention to be entitled and does not concede, directly or by implication, that the subject matter of such claims were in fact disclosed or taught by the cited prior art. Rather, Applicant reserves the right to pursue such protection at a later point in time and merely seeks to pursue protection for the subject matter presented in this submission.

Based on the above amendment and the following remarks, Applicant respectfully requests that the Examiner reconsider all outstanding rejections and withdraw them.

Summary of Interview with the Examiner

A telephonic interview was conducted between the Examiner and the Applicant's attorney on October 12, 2006 and on October 24, 2006. During these interviews distinctions between the Applicant's invention and the prior art were

discussed. Specifically, the Applicant discussed the transformation of a serial based algorithm to one that can be systolically implemented in a reconfigurable processor. Additionally, the claims concurrent execution of computation units in such a systolic fashion in comparison to the non concurrent systolic nature of the cited references were discussed. Discussion also centered on the use of the words "protocol independent" to impart the ability of functional units to seamlessly pass computed data between computational loops comprised of functional units. Proposed amendments where discussed although no specific language was agreed upon. The Examiner requested that the Applicant further define the term instantiated and systolic in subsequent communications.

Rejection of the claims under 35 U.S.C. §112

Claims 1-20, 22-46, 48-51, 53, 54, and 56 were rejected under 35 U.S.C. §112 first paragraph as failing to comply with the written description requirement. The Examiner asserts that the application fails to comply with the written description and enablement requirement with respect to the language of the independent claims stating "wherein each functional unit at the at least one reconfigurable processor communicates with each other functional unit at the at least one reconfigurable processor independent of external and internal communication protocols." The Applicant respectfully disagrees.

The present invention describes and claims methods in a reconfigurable processing system comprising at least one reconfigurable processor. As described in the specification at least in the paragraphs beginning on line 26 of page 10 and on line 3 of page 11, each reconfigurable processor can possess a plurality of functional units. The instantiation of the at least one reconfigurable processor with at least two functional units enables each functional unit to communicate with each other. Certainly communication between other reconfigurable processors within the

system would require communication protocol but communication between functional units within an individual reconfigurable processor is free of such a requirement. To alleviate any confusion, the reference to the term "protocol" has been replaced with an "interconnection" between functional units that is established by reconfigurable routing resources inside each chip.

The Examiner also rejects the aforementioned claims on the grounds that the number of functional units needed to solve a particular problem is not described in the specification. The Applicant refers the Examiner to lines 2-8 of page 3 of the specification. While the Applicant believes the concept of the "type" and "number" of functional unit is implied in the aforementioned portion of the specification, the wording of the claim has been amended to align with the specification, namely that only the functional units needed to solve a particular application are instantiated.

Instantiation is a term well known to one of ordinary skill in the art of reconfigurable processing. A reconfigurable processor is essentially a blank processor that must be configured (instantiated) to conduct a particular task. To instantiate means to create such an instance or configuration by, for example, defining one particular variation of the processor's structure. This involves allocation of a structure with the types specified by a template and the initialization of instance variables with either default values or those provided by a constructor function. In reconfigurable computing a hard macro library file is typically inserted into a design file. A design may include multiple instances of the same library file with each possessing a unique name. Thus in the Applicant's invention the reconfigurable processor is instantiated and designed to perform the defined calculation. Each instantiation for each calculation is unique.

Similarly the term systolic computation is derived from continual and pulsating pumping of the human heart. In computer architecture a systolic array is an

arrangement of data processing units similar to a central processing unit but without a program counter or clock that drives the movement of data. That is because the operation of the systolic array is transport triggered, i.e. by the arrival of a data object. Data flows across the array between functional units, usually with different data flowing in different directions. David J. Evans in his work, Systolic algorithms. Systolic algorithms, number 3 in Topics in Computer Mathematics, Gordon and Breach, 1991 define a Systolic system as a "network of processors which rhythmically compute an pass data through the system" Thus in the Applicant's invention Systolic implementation will connect computational loops such that data from one compute loop will be passed as input data to a concurrently executing compute loop. In the Applicant's invention data computed by computation units or groups of functional units flows seamlessly and concurrently with data being computed by other groups of functional units. Thus, the process claimed by the Applicant therefore significantly increases the computing processes taking place in a reconfigurable processor.

35 U.S.C. §103(a) Obviousness Rejection of Claims

Claims 1-5, 15, 21, 26-31, 41, 47, 52, 53 and 56 were rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,956,518 ("DeHon") in view of U.S. Patent No. 5,274,832 ("Khan"). Dependent claims 2-14, 16-20, 22-25, 32-40, 42-46, 48-51, and 54 were rejected as being unpatentable over DeHon in view of Khan as applied to the applicable independent claim in further view of various citations of prior art. Applicant respectfully traverses these rejections in view of the following remarks and addresses the rejection of all dependent claims as being based on the rejection of the underlying independent claim.

MPEP §2143 provides:

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation,

> either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teaching. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations.

The cited references fail to teach or suggest all of the limitations recited in the claims as currently amended. For example, independent claim 1 (and claims 26 and 54 in varying language) recites, among other things, "transforming an algorithm into a calculation that is systolically implemented by said reconfigurable computing system at the at least one reconfigurable processor ... wherein systolically linked lines of code of said calculation are instantiated as clusters of functional units within the at least one reconfigurable processor ... [and] wherein said systolic implementation of said calculation enables said first computational loop and said second computational loop execute concurrently and pass computed data seamlessly between said computational loops."

DeHon and Khan fail to teach or suggest transforming an algorithm that is serial in nature to one that is systolic in nature for implementation on a reconfigurable processor. Converting a serial based algorithm into a calculation that can be implemented on a reconfigurable processor in a systolic fashion is described in detail in the examples provided in the specification. Specifically, Figures 8A-8C and the text beginning on line 8 of page 17 continuing through line 19 of page 20 describe a systolic wave front operation relating to, in this example, bioinformatics.

The Applicant's invention further forms computation units that are executed concurrently in a systolic fashion such that data computed within each computational unit is seamlessly and concurrently passed between the computational units. This transformation of the serial algorithm to one that is

systolic in nature and its implementation on a reconfigurable processor such that the computation units can operate concurrently and share data seamlessly provides considerable computational advantages and resulting efficiencies.

Several examples of this transformation and its implementation are illustrated in the specification and accompanying figures. As mentioned, Figures 8A-8C and the accompanying text beginning on page 17, line 8 convey the transformation of a bioinformatics process into a systolic wave-front operation. This and the other examples described in the specification detail the advantage and utility of the Applicant's invention which is distinguishable from both DeHon and Khan.

In view of all of the above, the claims are now believed to be allowable and the case in condition for allowance which action is respectfully requested. Should the Examiner be of the opinion that a telephone conference would expedite the prosecution of this case, the Examiner is requested to contact Applicant's attorney at the telephone number listed below.

No fee is believed due for this submittal. However, any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

13 Nov, 2006

Michael C. Martensen, No. 46,901
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5910 Tel
(303) 899-7333 Fax

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

NOTICE OF ALLOWANCE AND FEE(S) DUE

25235 7590 01/10/2007

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

EXAMINER	
COLEMAN, ERIC	
ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 01/10/2007

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/285,318	10/31/2002	Jon M. Huppenthal	SRC015	1420

TITLE OF INVENTION: MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	$1400	$300	$0	$1700	04/10/2007

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE DOES NOT REFLECT A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE IN THIS APPLICATION. IF AN ISSUE FEE HAS PREVIOUSLY BEEN PAID IN THIS APPLICATION (AS SHOWN ABOVE), THE RETURN OF PART B OF THIS FORM WILL BE CONSIDERED A REQUEST TO REAPPLY THE PREVIOUSLY PAID ISSUE FEE TOWARD THE ISSUE FEE NOW DUE.

HOW TO REPLY TO THIS NOTICE:

I. Review the SMALL ENTITY status shown above.

If the SMALL ENTITY is shown as YES, verify your current SMALL ENTITY status:

A. If the status is the same, pay the TOTAL FEE(S) DUE shown above.

B. If the status above is to be removed, check box 5b on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and twice the amount of the ISSUE FEE shown above, or

If the SMALL ENTITY is shown as NO:

A. Pay TOTAL FEE(S) DUE shown above, or

B. If applicant claimed SMALL ENTITY status before, or is now claiming SMALL ENTITY status, check box 5a on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and 1/2 the ISSUE FEE shown above.

II. PART B - FEE(S) TRANSMITTAL, or its equivalent, must be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed and an extra copy of the form should be submitted. If an equivalent of Part B is filed, a request to reapply a previously paid issue fee must be clearly made, and delays in processing may occur due to the difficulty in recognizing the paper as an equivalent of Part B.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Mail Stop ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

PTOL-85 (Rev. 07/06) Approved for use through 04/30/2007.

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), to: Mail **Mail Stop ISSUE FEE**
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450
or Fax (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

25235 7590 01/10/2007

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission
I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

(Depositor's name)
(Signature)
(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/285,318	10/31/2002	Jon M. Huppenthal	SRC015	1420

TITLE OF INVENTION: MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	$1400	$300	$0	$1700	04/10/2007

EXAMINER	ART UNIT	CLASS-SUBCLASS
COLEMAN, ERIC	2183	712-226000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).
☐ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.
☐ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list
(1) the names of up to 3 registered patent attorneys or agents OR, alternatively,
(2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.
1 _____
2 _____
3 _____

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE (B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent): ☐ Individual ☐ Corporation or other private group entity ☐ Government

4a. The following fee(s) are submitted:
☐ Issue Fee
☐ Publication Fee (No small entity discount permitted)
☐ Advance Order - # of Copies _____

4b. Payment of Fee(s): **(Please first reapply any previously paid issue fee shown above)**
☐ A check is enclosed.
☐ Payment by credit card. Form PTO-2038 is attached.
☐ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment, to Deposit Account Number _____ (enclose an extra copy of this form).

5. **Change in Entity Status** (from status indicated above)
☐ a. Applicant claims SMALL ENTITY status. See 37 CFR 1.27.
☐ b. Applicant is no longer claiming SMALL ENTITY status. See 37 CFR 1.27(g)(2).

NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Authorized Signature _____ Date _____

Typed or printed name _____ Registration No. _____

This collection of information is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/285,318	10/31/2002	Jon M. Huppenthal	SRC015	1420

25235 7590 01/10/2007

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

EXAMINER	
COLEMAN, ERIC	
ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 01/10/2007

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment to date is 625 day(s). If the issue fee is paid on the date that is three months after the mailing date of this notice and the patent issues on the Tuesday before the date that is 28 weeks (six and a half months) after the mailing date of this notice, the Patent Term Adjustment will be 625 day(s).

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at 1-(888)-786-0101 or (571)-272-4200.

PTOL-85 (Rev. 07/06) Approved for use through 04/30/2007.

Notice of Allowability	Application No. 10/285,318	Applicant(s) HUPPENTHAL ET AL.
	Examiner Eric Coleman	Art Unit 2183

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *amendment filed 11/13/06*.

2. ☒ The allowed claim(s) is/are *1-20,22-46,48-51,53,54 and 56*.

3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☒ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date _____
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☐ Interview Summary (PTO-413), Paper No./Mail Date _____ .
7. ☐ Examiner's Amendment/Comment
8. ☐ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

Eric Coleman
Primary Examiner

Client Matter No. 80404.0018
EFS-Web

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Serial No. 10/285,318 Application of: Jon M. Huppenthal and David E. Caliga Filed: October 31, 2002 Attorney Docket No. SRC015 For: MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS	Confirmation No.: 1420 Art Unit: 2183 Examiner: Coleman, Eric Customer No.: **25235**

AMENDMENT UNDER 37 C.F.R. § 1.312

MAIL STOP ISSUE FEE
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

A Notice of Allowance was mailed in the above application on January 10, 2007.

In response thereto, please enter the following amendments:

Amendments to the Specification begin on page 2 of this paper.

Remarks/Arguments begin on page 3 of this paper.

\\\CS - 080404/000018 - 88940 v1

Amendments to the Specification:

Please replace the paragraph beginning at page 1, line 2, with the following amended paragraph:

The present invention is related to the subject matter of United States Patent Application Ser. No. 09/755,744 filed January 5, 2001 for: "Multiprocessor Computer Architecture Incorporating a Plurality of Memory Algorithm Processors in the Memory Subsystem" and is further related to the subject matter of United States Patent No. ~~6,454,687~~ 6,434,687 for: "System and Method for Accelerating Web Site Access and Processing Utilizing a Computer System Incorporating Reconfigurable Processors Operating Under a Single Operating System Image", all of which are assigned to SRC Computers, Inc., Colorado Springs, Colorado and the disclosures of which are herein specifically incorporated in their entirety by this reference.

REMARKS/ARGUMENTS

The amendment presented herein corrects a typographical error in the patent number of the related application. No new matter is added by this amendment. No fee is believed due for this submittal. However, any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

15 March, 2007

William J. Kubida, No. 29,664
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5909 Tel
(303) 899-7333 Fax

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	ISSUE DATE	PATENT NO.	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/285,318	05/29/2007	7225324	SRC015	1420

25235 7590 05/09/2007

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

ISSUE NOTIFICATION

The projected patent number and issue date are specified above.

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment is 550 day(s). Any patent to issue from the above-identified application will include an indication of the adjustment on the front page.

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at (571)-272-4200.

APPLICANT(s) (Please see PAIR WEB site http://pair.uspto.gov for additional applicants):

Jon M. Huppenthal, Colorado Springs, CO;
David E. Caliga, Colorado Springs, CO;

IR103 (Rev. 11/05)

EXHIBIT M

Attorney Docket No.: SRC012 DIV
Client Matter No. 80404.0015.002
Express Mail No.: EV331756098US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Divisional Application of: Jon M. Huppenthal, Thomas R. Seeman, Lee A. Burton Serial No. NEW Filed: Herewith	For: SWITCH/NETWORK ADAPTER PORT FOR CLUSTERED COMPUTERS EMPLOYING A CHAIN OF MULTI-ADAPTIVE PROCESSORS IN A DUAL IN-LINE MEMORY MODULE FORMAT

PRELIMINARY AMENDMENT ACCOMPANYING DIVISIONAL APPLICATION TRANSMITTAL

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Enclosed herewith is a true copy of U.S. Patent Application Serial No. 09/932,330 filed August 17, 2001, in support of the divisional Patent Application Transmittal filed herewith. Please amend the specification and claims as follows:

\\\CS - 80404/0015 - 71242 v1

Amendments to the Specification:

Please replace the paragraph beginning at page 1, line 2, with the following amended paragraph:

The present invention is a divisional patent application of United States Patent Application Serial No. 09/932,330 filed August 17, 2001, which is a continuation-in-part patent application of United States Patent Application Serial No. 09/755,744 filed January 5, 2001 which is a divisional patent application of United States Patent Application Serial No. 09/481,902 filed January 12, 2000, now United States Patent No. 6,247,110, which is a continuation of United States Patent Application Serial No. 08/992,763 filed December 17, 1997 for: "Multiprocessor Computer Architecture Incorporating a Plurality of Memory Algorithm Processors in the Memory Subsystem", now United States Patent No. 6,076,152, assigned to SRC Computers, Inc., Colorado Springs, Colorado, assignee of the present invention, the disclosures of which are herein specifically incorporated by this reference.

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

Please cancel claims 1-36.

37. (original) A processor element for a memory module bus of a computer system, said processor element comprising:

a field programmable gate array configurable to perform an identified algorithm on an operand provided thereto and operative to alter data provided thereto on said memory module bus; and

a data connection coupled to said field programmable gate array for providing said altered data to an external device coupled thereto.

38. (original) The processor element of claim 37 further comprising:

a control connection coupled to said processor element for indicating to a processor of said computer system an arrival of data on said data connection from said external device.

39. (original) The processor element of claim 38 wherein said control connection indicates said arrival of data to said processor by means of a peripheral bus.

40. (original) The processor element of claim 39 wherein said peripheral bus comprises a PCI bus.

41. (original) The processor element of claim 38 wherein said control connection indicates said arrival of data to said processor by means of a graphics bus.

42. (original) The processor element of claim 41 wherein said graphics bus comprises an AGP bus.

43. (original) The processor element of claim 38 wherein said control connection indicates said arrival of data to said processor by means of a system maintenance bus.

44. (original) The processor element of claim 43 wherein said graphics bus comprises an SM bus.

45. (original) The processor element of claim 37 wherein said memory module bus comprises a DIMM bus.

46. (original) The processor element of claim 45 wherein said processor element comprises a DIMM physical format.

47. (original) The processor element of claim 37 wherein said memory module bus comprises a RIMM bus.

48. (original) The processor element of claim 47 wherein said processor element comprises a RIMM physical format.

49. (original) The processor element of claim 37 wherein said external device comprises one of another computer system, switch or network.

50. (original) The processor element of claim 37 wherein said processor of said computer system comprises a plurality of processors.

51. (original) The processor element of claim 37 wherein said field programmable gate array is further operative to alter data provided thereto from said external device on said data connection and providing said altered data on said memory module bus.

REMARKS/ARGUMENTS

Claims 37-51 are pending in this application. These claims presented for examination are the unelected claims 37-51 of the invention of Group II in the parent case.

This Preliminary Amendment accompanies a divisional patent application transmittal and the fee associated therewith. Although no additional fees are believed due for this Preliminary Amendment, any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

27 November, 2004

William J. Kubida, No. 29,664
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5909 Tel
(303) 899-7333 Fax

Office Action Summary	Application No.	Applicant(s)	
	10/996,016	HUPPENTHAL ET AL.	
	Examiner	Art Unit	
	Eron J. Sorrell	2182	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *prelim. amendment filed 11/23/04*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *37-51* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *37-51* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *23 November 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *11/23/04;5/5/06*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. The preliminary amendment filed 11/23/04 is acknowledged. Claims 37-51 are currently pending.

Information Disclosure Statement

The information disclosure statement (IDS) submitted on 11/23/04 is in compliance with the provisions of 37 CFR 1.97. Accordingly, the information disclosure statement is being considered by the examiner. However, much of the information is of little to no relevance to the claimed subject matter, and has been given only a passing consideration due to the extensive content submitted.

Applicant is reminded that it is desirable to avoid the submission of long list of documents if it can be avoided. Eliminate clearly irrelevant and marginally pertinent cumulative information. If a long list is submitted, highlight those documents which have been specifically brought to applicant's attention and/or are known to be of most significance. See Penn Yan Boats, Inc. v. Sea Lark Boats, Inc., 359 F. Supp. 948, 175 USPQ 260 (S.D. Fla. 1972), aff 'd, 479 F.2d 1338, 178 USPQ 577 (5th Cir. 1973), cert. denied, 414 U.S. 874 (1974). But cf. Molins PLC v. Textron Inc., 48 F.3d 1172, 33 USPQ2d 1823 (Fed. Cir. 1995).

Claim Objections

2. Claim 44 is objected to because of the following informalities: line 1 of the claim recites, "said graphics bus," however it appears as if the claim should recite, "said system maintenance bus". Appropriate correction is required.

Claim Rejections - 35 USC § 112

3. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

4. Claims 47 and 48 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claims 47 and 48 contain the trademark/trade name "RIMM". Where a trademark or trade name is used in a claim as a limitation to identify or describe a particular material or product, the claim does not comply with the requirements of 35 U.S.C. 112, second paragraph. See *Ex parte Simpson*, 218 USPQ 1020 (Bd. App. 1982). The claim scope is uncertain since the trademark or trade name cannot be used properly to identify any particular material or product. A trademark or trade name is used to identify a source of goods, and not the goods themselves. Thus, a trademark or trade name does not identify or describe the goods associated with the trademark or trade name. In the present case, the trademark/trade name is used to identify/describe a memory module and, accordingly, the identification/description is indefinite.

Claim Rejections - 35 USC § 102

5. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless -

> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

6. Claims 37-39,41,42,49, and 51 are rejected under 35 U.S.C. 102(e) as being anticipated by Klingelhofer (U.S. Patent No. 5,673,204).

7. Referring to claim 37, Klingelhofer teaches a processor element (item 120 in figure 1) for a memory module bus (item 60 in figure 1) of a computer system, said processor element comprising:

a field programmable gate array (see item labeled DX in figure 1) configurable to perform an identified algorithm on an operand provided thereto and operative to alter data provided thereto on said memory module bus (see lines 7-18 of column 4 and lines 20-27 of column 5); and

a data connection coupled to said field programmable gate array for providing said altered data to an external device coupled thereto (see connection to bus 140 in figure 1).

8. Referring to claims 38 and 39, Klingelhofer teaches a control connection coupled to said processor element for indicating to a processor of said computer system an arrival of data on said data connection from said external device wherein said control connection indicates said arrival of data to said processor by means of a peripheral bus (see paragraph bridging columns 10 and 11).

9. Referring to claim 41 and 42, Klingelhofer teaches the control connection indicates said arrival of data to said processor by means of a an AGP bus (see M bus and lines 1-5 of column 11).

10. Referring to claim 49, Klingelhofer teaches the external device comprises one of another computer system, switch or network (see lines 46-52 of column 4, note the external device can be another video processing system).

11. Referring to claim 51, Klingelhofer teaches the field programmable gate array is further operative to alter data provided thereto from said external device on said data connection and providing said altered data on said memory module bus (see lines 41-52 of column 5).

Claim Rejections - 35 USC § 103

12. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

13. Claims 40 and 50 are rejected under 35 U.S.C. 103(a) as being unpatentable over Klingelhofer in view of Acton et al. (U.S. Patent No. 6,094,532 hereinafter "Acton").

14. Referring to claim 40, Klingelhofer teaches the apparatus of claim 39 as shown above, however fails to teach the peripheral bus comprises a PCI bus.

Acton teaches in an analogous system, that PCI busses are notoriously well known in the art (see lines 29-45 of column 1).

It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the apparatus of Klingelhofer with the above teachings of Acton because PCI buses are old and well known and often used because of the higher data transfer rate and compatibility with existing peripheral devices.

15. Referring to claim 50, Acton teaches that is old and well known for the processor of a computer system to comprise a plurality of processors (see lines 13-26 of column 1).

It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the apparatus of Klingelhofer such that the processor comprises multiple processors for the advantage of providing increased processing power.

16. Claims 43 and 44 are rejected under 35 U.S.C. 103(a) as being unpatentable over Klingelhofer in view of Whittaker et al. (U.S. Patent No. 5,889,959 hereinafter "Whittaker").

17. Referring to claims 43 and 44, Klingelhofer teaches the apparatus of claim 38 as shown above, however fails to teach the control connection indicates said arrival of data to said processor by means of a SM bus.

Whittaker teaches the above limitation (see lines 42-49 of column 4).

It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the teachings of Klingelhofer with the above teachings of Whittaker.

One of ordinary skill in the art would have been motivated to make such modification in order to provide diagnostic functions to all of the modules in the system as suggested by Whittaker (see lines 30-43 of column 1).

18. Claims 45-48 are rejected under 35 U.S.C. 103(a) as being unpatentable over Klingelhofer in view of Applicant's Admitted prior Art (AAPA).

19. Referring to claim 45-48, Klingelhofer teaches the processor element of claim 37 as shown above, however Klingelhofer fails to teach the memory module bus comprises a DIMM bus or a RIMM bus and that the processor element has a DIMM or RIMM form factor.

At paragraph 7 of the instant specification, the Applicant admits that the DIMM format is one of the most commonly used memory formats in PCs today and that the RIMM format is similar to the DIMM format.

It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the teachings of Klingelhofer with the above teachings from AAPA. One of ordinary skill in the art would have been motivated to make such modification to provide compatibility and increased flexibility with existing systems.

Conclusion

20. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. The following reference is cited to show the state of art as it pertains to the applicant's invention:

U.S. Patent No. 5,978,862 to Kou et al. teaches a processor element with a data interface and an FPGA.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eron J. Sorrell whose telephone number is 571 272-4160. The examiner can normally be reached on Monday-Friday 8:00AM - 4:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kim Huynh can be reached on 571-272-4147. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

EJS
February 26, 2007

KIM HUYNH
SUPERVISORY PATENT EXAMINER
2/28/07

Client Matter No. 80404.0015.002
EFS-Web

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Serial No. 10/996,016 Application of: Jon M. Huppenthal, Thomas R. Seeman, Lee A. Burton Filed: November 23, 2004 Attorney Docket No. SRC012 DIV For: SWITCH/NETWORK ADAPTER PORT FOR CLUSTERED COMPUTERS EMPLOYING A CHAIN OF MULTI-ADAPTIVE PROCESSORS IN A DUAL IN-LINE MEMORY MODULE FORMAT	Confirmation No.: 4730 Art Unit: 2182 Examiner: Sorrell, Eron J. Customer No.: **25235**

AMENDMENT

MAIL STOP AMENDMENT
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the office communication mailed March 6, 2007, please amend the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims which begin on page 2 of this paper.

Remarks/Arguments begin on page 5 of this paper.

\\\CS - 080404/000015 - 89096 v1

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1-36. (cancelled)

37. (currently amended) A processor element for a memory module bus of a computer system, said processor element comprising:

a field programmable gate array configurable to perform an identified algorithm on an operand provided thereto and operative to alter data provided directly thereto on said memory module bus; and

a direct data connection coupled to said field programmable gate array for providing said altered data from said memory module bus directly to an external device coupled thereto.

38. (original) The processor element of claim 37 further comprising:

a control connection coupled to said processor element for indicating to a processor of said computer system an arrival of data on said data connection from said external device.

39. (original) The processor element of claim 38 wherein said control connection indicates said arrival of data to said processor by means of a peripheral bus.

40. (original) The processor element of claim 39 wherein said peripheral bus comprises a PCI bus.

41. (original) The processor element of claim 38 wherein said control connection indicates said arrival of data to said processor by means of a graphics bus.

42. (original) The processor element of claim 41 wherein said graphics bus comprises an AGP bus.

43. (original) The processor element of claim 38 wherein said control connection indicates said arrival of data to said processor by means of a system maintenance bus.

44. (currently amended) The processor element of claim 43 wherein said ~~graphics bus~~ system maintenance bus comprises an SM bus.

45. (original) The processor element of claim 37 wherein said memory module bus comprises a DIMM bus.

46. (original) The processor element of claim 45 wherein said processor element comprises a DIMM physical format.

47. (currently amended) The processor element of claim 37 wherein said memory module bus comprises a [[RIMM]] in-line memory module serial interface bus.

48. (currently amended) The processor element of claim 47 wherein said processor element comprises a [[RIMM]] in-line memory module serial interface physical format.

49. (original) The processor element of claim 37 wherein said external device comprises one of another computer system, switch or network.

50. (original) The processor element of claim 37 wherein said processor of said computer system comprises a plurality of processors.

51. (original) The processor element of claim 37 wherein said field programmable gate array is further operative to alter data provided thereto from said external device on said data connection and providing said altered data on said memory module bus.

REMARKS/ARGUMENTS

Claims 37-51 were presented for examination and are pending in this application. In an Official Office Action dated March 6, 2007, claims 37-51 were rejected. The Applicant thanks the Examiner for his consideration and addresses the Examiner's comments concerning the claims pending in this application below.

Applicant herein amends claims 37, 44, 47 and 48 and respectfully traverses the Examiner's prior rejections. No claims are canceled and no new claims are presently added. These changes are believed not to introduce new matter, and their entry is respectfully requested. The claims have been amended to expedite the prosecution and issuance of the application. In making this amendment, the Applicant has not and is not narrowing the scope of the protection to which the Applicant considers the claimed invention to be entitled and does not concede, directly or by implication, that the subject matter of such claims was in fact disclosed or taught by the cited prior art. Rather, the Applicant reserves the right to pursue such protection at a later point in time and merely seeks to pursue protection for the subject matter presented in this submission.

Based on the above amendment and the following remarks, Applicant respectfully requests that the Examiner reconsider all outstanding rejections and withdraw them.

I. Claim objections

Claim 44 was objected to for minor informalities. Claim 44 has been amended to correctly state "said system maintenance bus". Reconsideration is requested.

II. 35 U.S.C. § 112 Rejection of Claims

Claims 47 and 48 were rejected under 35 U.S.C. § 112 second paragraph as being indefinite for failing to particularly point out and distinctly claim the subject matter that the Applicant regards as the invention. Specifically, the aforementioned claims contain the trademark/trade name RIMM or Rambus™ In-Line Memory Module. Claims 47and 48 have been amended to remove the trademark Rambus™ leaving the limitations generic as to an in-line memory module serial interface. The Applicant requests the rejection be withdrawn.

III. Rejection of the Claims

Claims 37-39, 41, 42, 49 and 51 were rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent No. 5,673,204 by Klingelhofer ("Klingelhofer"). Applicant respectfully traverses these rejections in light of the following remarks.

MPEP §2131 provides:

> "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegall Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 U.S.P.Q.2d 1051, 1053 (Fed. Cir.1987). "The identical invention must be shown in as complete detail as contained in the claim." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 U.S.P.Q.2d 1913, 1920 (Fed. Cir. 1989).

The claims as currently amended recite features lacking in the applied references. For example, independent claim 37 recites, among other things, "a direct data connection coupled to said field programmable gate array for providing said altered data from said memory module bus directly to an external device coupled thereto." (emphasis added)

The present invention provides a processor element for a memory module bus that enables a direct data connection between an external device and the memory module bus. Klingelhofer fails to provide such a direct data connection. The SIMMBUS of Klingelhofer is simply a memory bus and memory bus by itself fails to provide any mechanism or link by which to signal the memory controller that data on the bus is available. As stated in Klingelhofer, data must be in the VRAM to be operated upon. See Klingelhofer Col. 5, lines 27-33. This fact is especially significant with respect to claims 38 and 39 and other claims that refer to a control connection.

In addition, the IOSIMM of Figure 1 of Klingelhofer shows a data accelerator coupled to VRAM which is in turn coupled to a SIMMBUS. According to Klingelhofer, VRAM is a necessary component for communication to the video adapter (external device). The present invention removes this limitation.

As stated in Col. 3, lines 45-53 of Klingelhofer, "while the IOSIMM 120 VRAM is directly coupled to data accelerator (DX) and then to lead 140, DRAM in DSIMM 70, 70' must be coupled via SIMMBUS 60 through the IOSIMM and then into lead 140, for communication with the video adapter 10." Klingelhofer discloses a component (IOSIMM) possessing a processor (DX) coupled to VRAM that is used as a communications port to convert data for submission to a video adapter. The IOSIMM of Klingelhofer does not provide for the direct transfer of data from the memory module bus to the external device. Reconsideration is respectfully requested.

Claims 40 and 50 and claims 43 and 44 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Klingelhofer in view of U.S. Patent No. 6,094,532 and U.S. Patent No. 5,889,959, respectively. As claims 40, 43, 44 and 50 depend from claims 37 and neither of these two additional references resolve the

deficiencies of Klingelhofer, claims 40, 43, 44 and 50 are also deemed to be patentable for at least the same reasons as claim 37.

In view of all of the above, the claims are now believed to be allowable and the case in condition for allowance which action is respectfully requested. Should the Examiner be of the opinion that a telephone conference would expedite the prosecution of this case, the Examiner is requested to contact Applicant's attorney at the telephone number listed below.

No fee is believed due for this submittal. However, any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

1 June, 2007

Michael C. Martensen, No. 46,901
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5910 Tel
(303) 899-7333 Fax

Office Action Summary	Application No.	Applicant(s)	
	10/996,016	HUPPENTHAL ET AL.	
	Examiner	Art Unit	
	Eron J. Sorrell	2182	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *06 June 2007*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *37-51* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *37-51* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *6/6/07*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Examiner's Remarks

1. Applicant's amendment to the claims is sufficient to overcome the claim objections and rejection under 35 USC 112-2nd paragraph.

Response to Arguments

2. Applicant's argument pertaining to the data being directly transferred to the FPGA from the memory module bus is moot in light of the new grounds of rejection.

3. Applicant's arguments filed 6/6/07 have been fully considered but they are not persuasive. The applicant argues:

1) Klingelhofer fails to teach a direct data connection coupled to the FPGA for providing data from the memory module bus directly to the external device (see last paragraph of page 6 of applicant's remarks);

2) The SIMMBUS of Klingelhofer is simply a memory bus and a memory bus itself fails to provide a mechanism or link by which signal the memory controller that data on the bus is available (see first paragraph of page 7).

3) The IOSIMM of Klingelhofer does not provide for the direct transfer of data from the memory module bus to the external device.

4. **As per argument 1, the Examiner disagrees.** Klingelhofer teaches a direct data connection between the FPGA and the video adapter (external device) (see item 140, figure 1). ***The data comes from the memory module bus***, is temporarily stored in the VRAM, operated on by the FPGA, and ***directly transferred to the video adapter*** (see lines 53-61 of column 4).

5. **As per argument 2, the Examiner disagrees.** The applicant only claims a memory module bus, and not any circuitry or functionality that provides a mechanism or link by which signals the memory controller that data on the bus is available.

6. **As per argument 3**, the applicant again is arguing limitations that are not required by the claim language. The claim recites **a direct data connection coupled to the FPGA for providing data from the memory module bus directly to an internal device**, and **not providing for the direct transfer of data from the memory module bus to the external device** (emphasis added). There is an important distinction in what is claimed compared to what is argued by the applicant. The claim only requires that the data be directly provided to the external device and that the data come from the memory module bus. The data does come from the memory module bus, and is temporarily stored in the VRAM before being directly transferred to the external device. The claim does not require a direct transfer from the memory module bus to the external device.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8. Claims 37-39,41,42,49, and 51 are rejected under 35 U.S.C. 103(a) as being unpatentable over Klingelhofer (U.S. Patent No. 5,673,204) in view of Garnett (U.S. Patent No. 5,911,778).

9. Referring to claim 37, Klingelhofer teaches a processor element (item 120 in figure 1) for a memory module bus (item 60 in figure 1) of a computer system, said processor element comprising:

a field programmable gate array (see item labeled DX in figure 1) configurable to perform an identified algorithm on an operand provided thereto and operative to alter data provided thereto on said memory module bus (see lines 7-18 of column 4 and lines 20-27 of column 5); and

a direct data connection coupled to said field programmable gate array for providing said altered data from the memory module bus directly to an external device coupled thereto (see connection to bus 140 in figure 1).

Klingelhofer fails to teach the FPGA alters data provided directly thereto on the memory module bus.

Garnett teaches a processing system comprising and FPGA directly coupled to a memory module bus (see lines 45-63 of column 2).

It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the system of Klingelhofer such that is comprises a direct connection from the FPGA to the memory bus. One of ordinary skill in the art would have been motivated to make such modification in order to be able to reprogram the FPGA in a secure manner to perform new operations as suggested by Garnett (see lines 45-63 of column 2).

10. Referring to claims 38 and 39, Klingelhofer teaches a control connection coupled to said processor element for indicating to a processor of said computer system an arrival of data on said data connection from said external device wherein said control connection indicates said arrival of data to said processor by means of a peripheral bus (see paragraph bridging columns 10 and 11).

11. Referring to claim 41 and 42, Klingelhofer teaches the control connection indicates said arrival of data to said processor by means of a an AGP bus (see M bus and lines 1-5 of column 11).

12. Referring to claim 49, Klingelhofer teaches the external device comprises one of another computer system, switch or network (see lines 46-52 of column 4, note the external device can be another video processing system).

13. Referring to claim 51, Klingelhofer teaches the field programmable gate array is further operative to alter data provided thereto from said external device on said data connection and providing said altered data on said memory module bus (see lines 41-52 of column 5).

14. Claims 40 and 50 are rejected under 35 U.S.C. 103(a) as being unpatentable over Klingelhofer in view Garnett as applied to claim 37 above, and further in view of Acton et al. (U.S. Patent No. 6,094,532 hereinafter "Acton").

15. Referring to claim 40, Klingelhofer teaches the apparatus of claim 39 as shown above, however fails to teach the peripheral bus comprises a PCI bus.

Acton teaches in an analogous system, that PCI busses are notoriously well known in the art (see lines 29-45 of column 1).

It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the apparatus of Klingelhofer with the above teachings of Acton because PCI buses are old and well known and often used because of the higher data transfer rate and compatibility with existing peripheral devices.

16. Referring to claim 50, Acton teaches that is old and well known for the processor of a computer system to comprise a plurality of processors (see lines 13-26 of column 1).

It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the apparatus of Klingelhofer such that the processor comprises multiple processors for the advantage of providing increased processing power.

17. Claims 43 and 44 are rejected under 35 U.S.C. 103(a) as being unpatentable over Klingelhofer in view Garnett as applied to claim 37 above, and further in view of Whittaker et al. (U.S. Patent No. 5,889,959 hereinafter "Whittaker").

18. Referring to claims 43 and 44, Klingelhofer teaches the apparatus of claim 38 as shown above, however fails to teach the control connection indicates said arrival of data to said processor by means of a SM bus.

Whittaker teaches the above limitation (see lines 42-49 of column 4).

It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the teachings of Klingelhofer with the above teachings of Whittaker.

One of ordinary skill in the art would have been motivated to make such modification in order to provide diagnostic functions to all of the modules in the system as suggested by Whittaker (see lines 30-43 of column 1).

19. Claims 45-48 are rejected under 35 U.S.C. 103(a) as being unpatentable over Klingelhofer in view of Applicant's Admitted prior Art (AAPA).

20. Referring to claim 45-48, Klingelhofer teaches the processor element of claim 37 as shown above, however Klingelhofer fails to teach the memory module bus comprises a DIMM bus or an in-line memory module serial interface bus and that the processor element has a DIMM or RIMM form factor.

At paragraph 7 of the instant specification, the Applicant admits that the DIMM format is one of the most commonly used memory formats in PCs today and that the RIMM format is similar to the DIMM format.

It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the teachings of Klingelhofer with the above teachings from AAPA. One of ordinary skill in the art would have been motivated to make such modification to provide compatibility and increased flexibility with existing systems.

Conclusion

21. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL.** See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action

is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eron J. Sorrell whose telephone number is 571 272-4160. The examiner can normally be reached on Monday-Friday 8:00AM - 4:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kim Huynh can be reached on 571-272-4147. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

EJS
August 13, 2007

TANH Q NGUYEN
PRIMARY EXAMINER
TECHNOLOGY CENTER 2100

August 16, 2007

Client Matter No. 80404.0015.002
EFS-Web

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Serial No. 10/996,016 Application of: Jon M. Huppenthal, Thomas R. Seeman, Lee A. Burton Filed: November 23, 2004 Attorney Docket No. SRC012 DIV For: SWITCH/NETWORK ADAPTER PORT FOR CLUSTERED COMPUTERS EMPLOYING A CHAIN OF MULTI-ADAPTIVE PROCESSORS IN A DUAL IN-LINE MEMORY MODULE FORMAT	Confirmation No.: 4730 Art Unit: 2182 Examiner: Sorrell, Eron J. Customer No.: **25235**

AMENDMENT AND RESPONSE PURSUANT TO FINAL OFFICE ACTION DATED AUGUST 27, 2007

MAIL STOP AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the final office communication mailed August 27, 2007, please amend the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims which begins on page 2 of this paper.

Remarks/Arguments begin on page 5 of this paper.

A **Supplemental Information Disclosure Statement with fee** is attached following page 8 of this paper.

\\\CS - 080404/000015 - 95546 v1

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1-36. (cancelled)

37. (currently amended) A processor element for a memory module bus of a computer system, said processor element comprising:

a field programmable gate array configurable to perform an identified algorithm on an operand provided thereto and operative to alter data provided directly thereto on said memory module bus; and

a direct data connection coupled to said field programmable gate array for providing said altered data directly from said memory module bus ~~directly~~ to an external device coupled thereto.

38. (original) The processor element of claim 37 further comprising:

a control connection coupled to said processor element for indicating to a processor of said computer system an arrival of data on said data connection from said external device.

39. (original) The processor element of claim 38 wherein said control connection indicates said arrival of data to said processor by means of a peripheral bus.

40. (original) The processor element of claim 39 wherein said peripheral bus comprises a PCI bus.

41. (original) The processor element of claim 38 wherein said control connection indicates said arrival of data to said processor by means of a graphics bus.

42. (original) The processor element of claim 41 wherein said graphics bus comprises an AGP bus.

43. (original) The processor element of claim 38 wherein said control connection indicates said arrival of data to said processor by means of a system maintenance bus.

44. (previously presented) The processor element of claim 43 wherein said system maintenance bus comprises an SM bus.

45. (original) The processor element of claim 37 wherein said memory module bus comprises a DIMM bus.

46. (original) The processor element of claim 45 wherein said processor element comprises a DIMM physical format.

47. (previously presented) The processor element of claim 37 wherein said memory module bus comprises a in-line memory module serial interface bus.

48. (previously presented) The processor element of claim 47 wherein said processor element comprises a in-line memory module serial interface physical format.

49. (original) The processor element of claim 37 wherein said external device comprises one of another computer system, switch or network.

50. (original) The processor element of claim 37 wherein said processor of said computer system comprises a plurality of processors.

51. (original) The processor element of claim 37 wherein said field programmable gate array is further operative to alter data provided thereto from said external device on said data connection and providing said altered data on said memory module bus.

REMARKS/ARGUMENTS

Claims 37-51 were presented for examination and are pending in this application. In an Official Office Action dated August 27, 2007, claims 37-51 were rejected. The Applicant thanks the Examiner for his consideration and addresses the Examiner's comments concerning the claims pending in this application below.

Applicant herein amends claim 37 consistent with the results of an Examiner Interview conducted on October 9, 2007 and respectfully traverses the Examiner's prior rejections. No claims are cancelled and no new claims are added. These changes are believed not to introduce new matter but rather to better clarify the novelty of the present invention. Their entry is respectfully requested. The claims have been amended to expedite the prosecution and issuance of the application. In making this amendment, the Applicant has not and is not narrowing the scope of the protection to which the Applicant considers the claimed invention to be entitled and does not concede, directly or by implication, that the subject matter of such claims was in fact disclosed or taught by the cited prior art. Rather, the Applicant reserves the right to pursue such protection at a later point in time and merely seeks to pursue protection for the subject matter presented in this submission.

Interview Summary

An Examiner Interview for the present application was conducted on October 9, 2007 at approximately 8:00 am Mountain Time zone. Participants included the Examiner, Eron Sorrell; Inventor Lee Burton; and Michael Martensen of Hogan & Hartson, LLP. The Examiner's response to the third argument presented in the response of June 6, 2007, was discussed during the interview. Namely, that the second limitation of claim 37 does not require

a direct transfer of data from the memory module bus to the external device. The Examiner suggested that rewording the limitation to state, "providing said altered data directly from said memory module bus to an external device coupled thereto" would be a positive reciting of this limitation. The impact of this amendment was discussed with respect to Klingelhoffer and the Examiner confirmed that this wording would overcome Klingelhoffer since Klingelhoffer includes a buffer (VRAM) interposed between the memory module bus and the external device.

The Examiner also confirmed that he had examined the implications of a direct data connection between the memory module bus and an external device and that this amendment would not raise a new issue warranting further search.

The interview concluded at 8:15 am Mountain time with an agreement that an amendment as indicated by the Examiner would move the prosecution forward.

35 U.S.C. §103(a) Obviousness Rejection of Claims

Claims 37-51 were rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,673,204 by Klingelhofer ("Klingelhofer") in view of various combinations of U.S. Patent No. 5,,911,778 by Garnett ("Garnett"), U.S. Patent No. 6,094,532 by Acton ("Acton"), U.S. Patent No. 5,889,959 by Whittaker ("Whittaker"), and Applicant's Admitted Prior Art ("AAPA"). Applicant respectfully traverses these rejections in light of the aforementioned remarks and respectfully requests reconsideration.

Claim 37 has been amended to recite "a direct data connection coupled to said field programmable gate array for providing said altered data directly from said memory module bus to an external device coupled thereto."

(emphasis added). The Applicant deems that this minor modification clarifies the position presented in the previous communications with the USPTO and brings claim 37 and those claims that depend from claim 37 into condition for allowance, which is respectfully requested.

Should the Examiner be of the opinion that a telephone conference would expedite the prosecution of this case, the Examiner is requested to contact Applicants' attorney at the telephone number listed below.

Please charge deposit account $180 to cover the Supplemental Information Disclosure Statement fee. No additional fees are believed due however, any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

10/26, 2007

Peter J. Meza, No. 32,920
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5906 Tel
(303) 899-7333 Fax

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/996,016	11/23/2004	Jon M. Huppenthal	SRC012 DIV	4730

25235 7590 03/21/2008

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

EXAMINER	
SORRELL, ERON J	
ART UNIT	PAPER NUMBER
2182	

DATE MAILED: 03/21/2008

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment to date is 407 day(s). If the issue fee is paid on the date that is three months after the mailing date of this notice and the patent issues on the Tuesday before the date that is 28 weeks (six and a half months) after the mailing date of this notice, the Patent Term Adjustment will be 407 day(s).

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at 1-(888)-786-0101 or (571)-272-4200.

PTOL-85 (Rev. 08/07) Approved for use through 08/31/2010.

Notice of Allowability	Application No. 10/996,016	Applicant(s) HUPPENTHAL ET AL.	
	Examiner ERON J. SORRELL	Art Unit 2182	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *RCE filed 11/27/07*.

2. ☒ The allowed claim(s) is/are *37-51*.

3. ☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☒ All b) ☐ Some* c) ☐ None of the:
      1. ☒ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☒ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date _____
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☐ Interview Summary (PTO-413), Paper No./Mail Date _____ .
7. ☐ Examiner's Amendment/Comment
8. ☒ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

Allowable Subject Matter

1. Claims 37-51 are allowed.

2. The following is an examiner's statement of reasons for allowance: The prior art of record fails to teach alone or in combination, a processor element for a memory module bus comprising an FPGA that receives and operates on data directly from a memory module bus, producing altered data and providing that altered data directly to an external device via a direct data connection, in combination with the other recited claim elements.

U.S. Patent No. to 5,911,778 to Garnett shows a FPGA directly coupled to a memory module bus that operates on data but the operated on data is not provided directly to an external device. The device in Garnett is an internal device.

U.S. Patent No. 5,857,109 to Taylor teaches a similar arrangement to that of Garnett with the FPGA providing data from a memory to an internal device.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to ERON J. SORRELL whose telephone number is (571)272-4160. The examiner can normally be reached on Monday-Friday 8:00AM - 4:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Alford Kindred can be reached on 571-272-4037. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/E. J. S./
Examiner, Art Unit 2182

/Alford W. Kindred/
Supervisory Patent Examiner, Art Unit 2163

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

NOTICE OF ALLOWANCE AND FEE(S) DUE

25235 7590 03/21/2008

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

EXAMINER	
SORRELL, ERON J	
ART UNIT	PAPER NUMBER
2182	

DATE MAILED: 03/21/2008

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/996,016	11/23/2004	Jon M. Huppenthal	SRC012 DIV	4730

TITLE OF INVENTION: SWITCH/NETWORK ADAPTER PORT FOR CLUSTERED COMPUTERS EMPLOYING A CHAIN OF MULTI-ADAPTIVE PROCESSORS IN A DUAL IN-LINE MEMORY MODULE FORMAT

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	$1440	$300	$0	$1740	06/23/2008

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE DOES NOT REFLECT A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE IN THIS APPLICATION. IF AN ISSUE FEE HAS PREVIOUSLY BEEN PAID IN THIS APPLICATION (AS SHOWN ABOVE), THE RETURN OF PART B OF THIS FORM WILL BE CONSIDERED A REQUEST TO REAPPLY THE PREVIOUSLY PAID ISSUE FEE TOWARD THE ISSUE FEE NOW DUE.

HOW TO REPLY TO THIS NOTICE:

I. Review the SMALL ENTITY status shown above.

If the SMALL ENTITY is shown as YES, verify your current SMALL ENTITY status:

A. If the status is the same, pay the TOTAL FEE(S) DUE shown above.

B. If the status above is to be removed, check box 5b on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and twice the amount of the ISSUE FEE shown above, or

If the SMALL ENTITY is shown as NO:

A. Pay TOTAL FEE(S) DUE shown above, or

B. If applicant claimed SMALL ENTITY status before, or is now claiming SMALL ENTITY status, check box 5a on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and 1/2 the ISSUE FEE shown above.

II. PART B - FEE(S) TRANSMITTAL, or its equivalent, must be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed and an extra copy of the form should be submitted. If an equivalent of Part B is filed, a request to reapply a previously paid issue fee must be clearly made, and delays in processing may occur due to the difficulty in recognizing the paper as an equivalent of Part B.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Mail Stop ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

PTOL-85 (Rev. 08/07) Approved for use through 08/31/2010.

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), to: Mail **Mail Stop ISSUE FEE**
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450
or Fax (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

25235 7590 03/21/2008

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission

I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

(Depositor's name)

(Signature)

(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/996,016	11/23/2004	Jon M. Huppenthal	SRC012 DIV	4730

TITLE OF INVENTION: SWITCH/NETWORK ADAPTER PORT FOR CLUSTERED COMPUTERS EMPLOYING A CHAIN OF MULTI-ADAPTIVE PROCESSORS IN A DUAL IN-LINE MEMORY MODULE FORMAT

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	$1440	$300	$0	$1740	06/23/2008

EXAMINER	ART UNIT	CLASS-SUBCLASS
SORRELL, ERON J	2182	710-072000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).

❑ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.

❑ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list

(1) the names of up to 3 registered patent attorneys or agents OR, alternatively,

(2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1 _____

2 _____

3 _____

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE (B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent) : ❑ Individual ❑ Corporation or other private group entity ❑ Government

4a. The following fee(s) are submitted:

❑ Issue Fee

❑ Publication Fee (No small entity discount permitted)

❑ Advance Order - # of Copies _____

4b. Payment of Fee(s): **(Please first reapply any previously paid issue fee shown above)**

❑ A check is enclosed.

❑ Payment by credit card. Form PTO-2038 is attached.

❑ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment, to Deposit Account Number _____ (enclose an extra copy of this form).

5. **Change in Entity Status** (from status indicated above)

❑ a. Applicant claims SMALL ENTITY status. See 37 CFR 1.27.

❑ b. Applicant is no longer claiming SMALL ENTITY status. See 37 CFR 1.27(g)(2).

NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Authorized Signature _____ Date _____

Typed or printed name _____ Registration No. _____

This collection of information is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

Mailed January 8, 2008

NOTICE OF GROUNDS OF REJECTION

Patent Application No.	539441/2000
Drafting Date	December 26, 2007
Patent Office Examiner	Masanori KUBO (9642 5B00)
Attorney	Mr. Hisao Fukami (et al.)
Applied Provision	Paragraph 2 of Article 29, Article 36

The present application is recognized as rejected on the following ground. It is required that any remarks be submitted within three months from the date on which the present NOTICE was mailed.

GROUNDS

1. It is recognized that, because the invention described in Claim(s) of SCOPE OF CLAIMS FOR PATENT of the present application could have been invented readily by a person having ordinary knowledge in the field of the art to which the present invention pertains prior to the filing of the present application based on the invention as described in the following publication(s) distributed or the invention as made available to the public through electric telecommunication lines in Japan and/or foreign countries prior to the filing of the present application, a patent cannot be granted thereto under the provision of Paragraph 2 of Article 29 of the Patent Law.

2. It is recognized that the present application does not satisfy the conditions prescribed in Paragraph 6 (ii) of Article 36 of the Patent Law because of the defectiveness of the description in SCOPE OF CLAIMS FOR PATENT on the following point.

REMARKS (See the list of the cited references.)

With regard to Ground 1

- Claims 1-81
- Cited References 1-2
- Note

Cited references 1 and 2 each describe that the processing instruction is given from a main control unit to a plurality of processors through a memory. Cited reference 1 also describes that the processing result is transferred to a different processor. There is no particular difficulty in constructing each processor with a well-known reconfigurable circuit and providing a plurality of main control units.

With regard to Ground 2

The description of the "improvement" in claims 1 to 7 is unclear since it cannot be specified whether it is the "product invention" or the "method invention".

Therefore, the invention according to claims 1 to 7 is unclear.

If any grounds of rejection are newly found, the grounds of rejection will be noticed.

LIST OF CITED REFERENCES

(1) Japanese Patent Laying-Open No. 63-086079

(2) Japanese Patent Laying-Open No. 59-206972

Record of Search for Prior Art Documents

* Searched Technical Field IPC G06F15/80

G06F15/16-15/177

* Prior Art Documents

Japanese Patent Laying-Open No. 11-015773

This record of search for prior art documents does not form any grounds of rejection.

EXHIBIT N

41 61-09/02 #21

Atty. Docket No. SRC010
Client Matter No. 80404.0013
Express Mail Label No. EL922181851US

In re Application of:

Jon M. Huppenthal

Serial No. 09/888,276

Filed: June 22, 2001

For: SYSTEM AND METHOD FOR ACCELERATING WEB SITE ACCESS AND PROCESSING UTILIZING A COMPUTER SYSTEM INCORPORATING RECONFIGURABLE PROCESSORS OPERATING UNDER A SINGLE OPERATING SYSTEM IMAGE

Examiner:

Art Unit: 2183

PRELIMINARY AMENDMENT

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

Please amend the above-identified patent application as follows:

IN THE SPECIFICATION:

Please substitute the following paragraph for the paragraph beginning on page 1, line 2:

The present invention is a continuation-in-part application of United States Patent Application Serial No. 09/563,561 filed May 3, 2000 (now issued U.S. Patent No. 6,339,819 B1) which is a continuation-in-part application of United States Patent Application Serial No. 09/481,902 filed January 12, 2000 (now issued U.S. Patent No. ~~6,339,819~~ 6,247,110) which is a continuation of United States Patent Application Serial No. 08/992,763 filed December 17, 1997 for: "Multiprocessor Computer Architecture Incorporating a Plurality of Memory ~~Algorithm~~ Processors in the Memory Subsystem" (now issued U.S. Patent No. 6,076,152), assigned to SRC Computers, Inc., Colorado Springs, Colorado, assignee of the present invention, the disclosures of which are herein specifically incorporated by this reference.

ICSK 2-11-02

\\\CS - 80404/13 - #47095 v1

REMARKS

Any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

19 September, 2001

William J. Kubida, Atty. Reg. No. 29,664
HOGAN & HARTSON LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5909 Tel

\\\CS - 80404/13 - #47095 v1

Version With Marking to Show Changes

In the Specification:

The present invention is a continuation-in-part application of United States Patent Application Serial No. 09/563,561 filed May 3, 2000 which is a continuation-in-part application of United States Patent Application Serial No. 09/481,902 filed January [12]13, 2000 which is a continuation of United States Patent Application Serial No. 08/992,763 filed December 17, 1997 for: "Multiprocessor Computer Architecture Incorporating a Plurality of Memory Algorithm Processors in the Memory Subsystem", assigned to SRC Computers, Inc., Colorado Springs, Colorado, assignee of the present invention, the disclosures of which are herein specifically incorporated by this reference.

#7

Attorney Docket No. SRC010
Client/Matter No. 80404.0013
Express Mail No. EL910770379US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

Jon M. Huppenthal

Serial No. 09/888,276

Filed: June 22, 2001

For: SYSTEM AND METHOD FOR ACCELERATING WEB SITE ACCESS AND PROCESSING UTILIZING A COMPUTER SYSTEM INCORPORATING RECONFIGURABLE PROCESSORS OPERATING UNDER A SINGLE OPERATING SYSTEM IMAGE

Group Art Unit: 2183

Examiner:

RESPONSE TO NOTICE OF INCOMPLETE REPLY

BOX MISSING PARTS
Assistant Commissioner for Patents
Washington, D.C. 20231

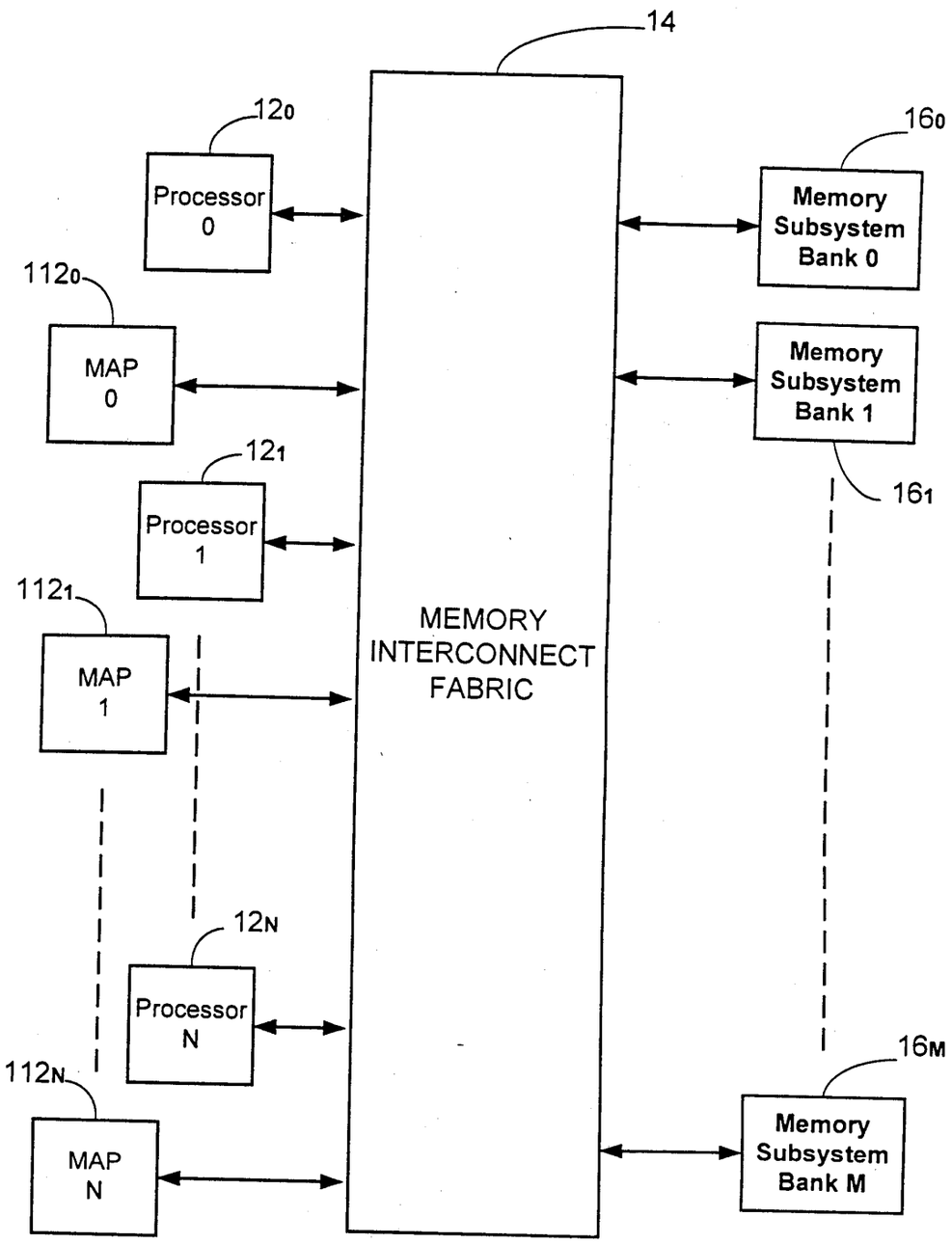
Sir:

In response to the Notice of Incomplete Reply, Filing Date Granted, mailed October 23, 2001, submitted herewith is a set of substitute drawings in compliance with 37 CFR 1.84; a Petition for a Two Month Extension of Time and check in the amount of $400; and a copy of the PTO Notice form. Any fee deficiency associated with this communication may be charged to Deposit Account No. 50-1123.

Dated: 16 Nov., 2001

William J. Kubida, Registration No. 29,664
HOGAN & HARTSON LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5909 Tel
(303) 899-7333 Fax

\\\CS - 80404/13 - #48995 v1



10

Fig. 1

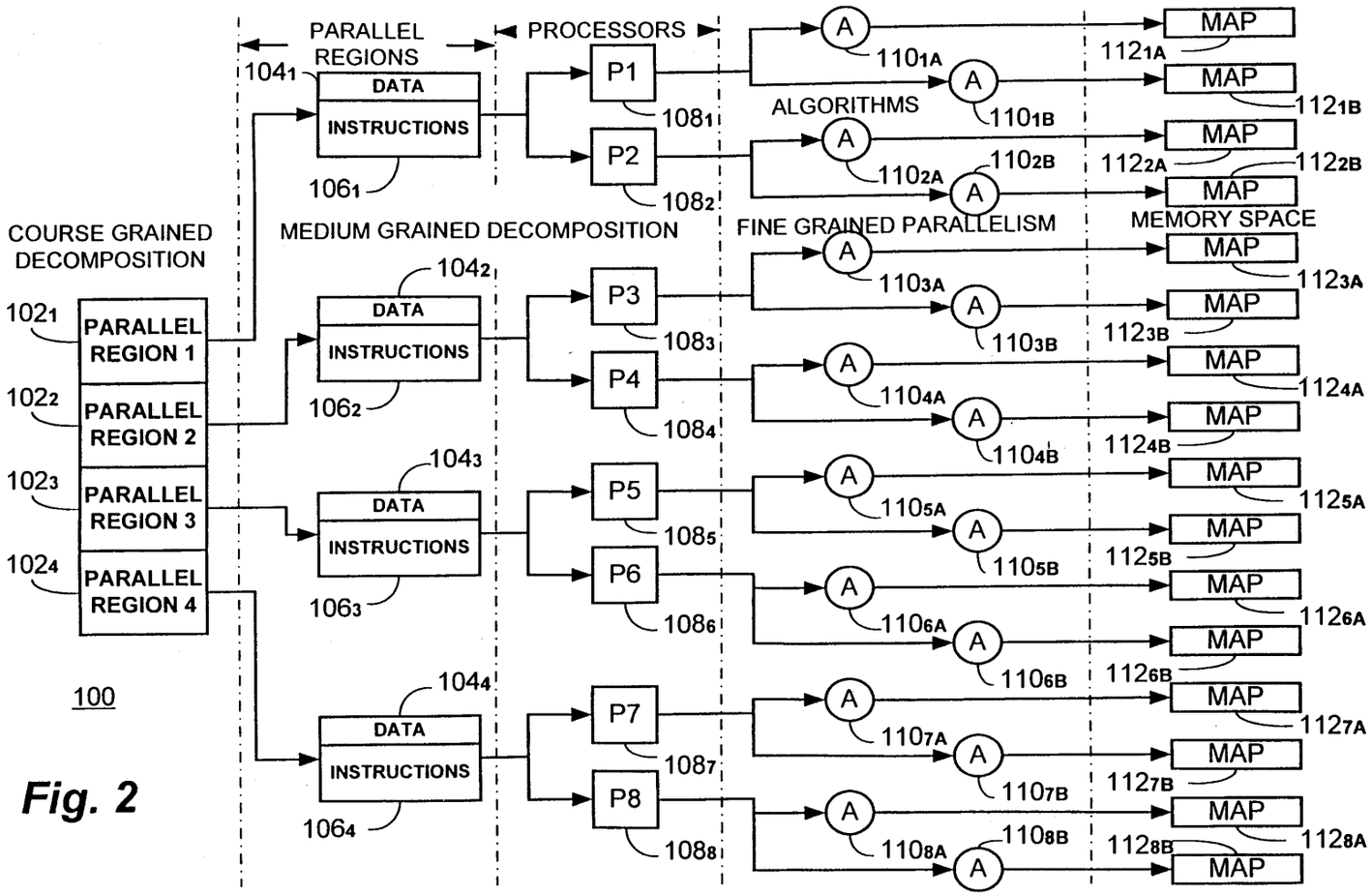


Fig. 2

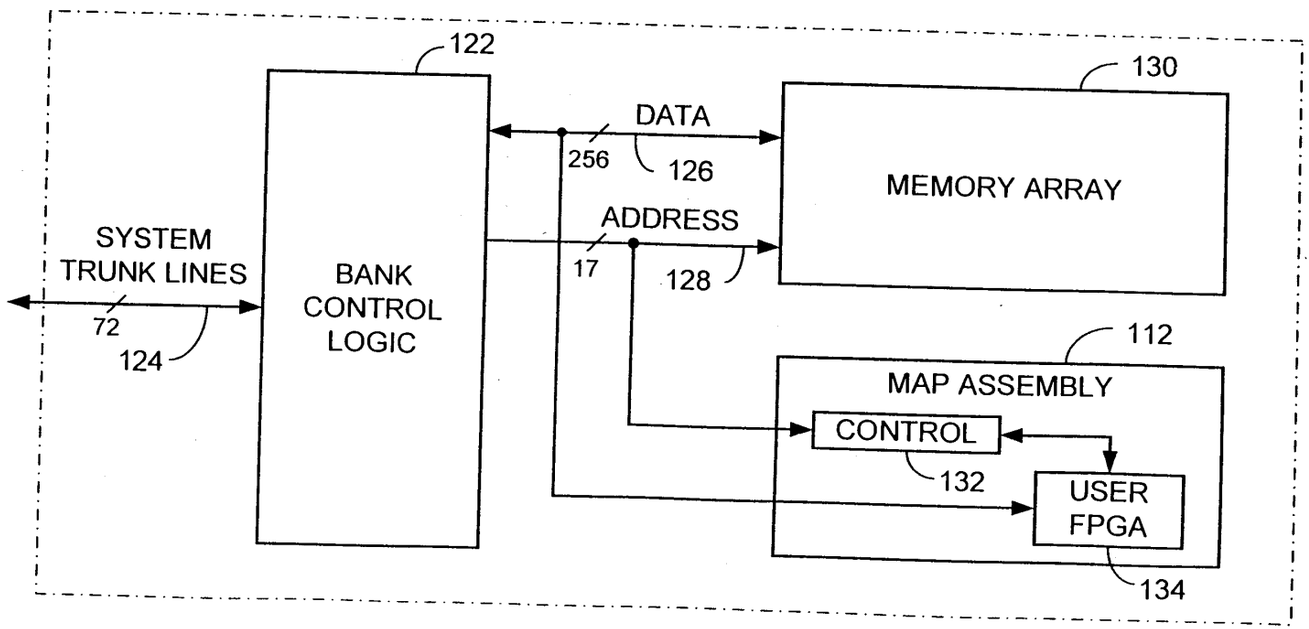


Fig. 3

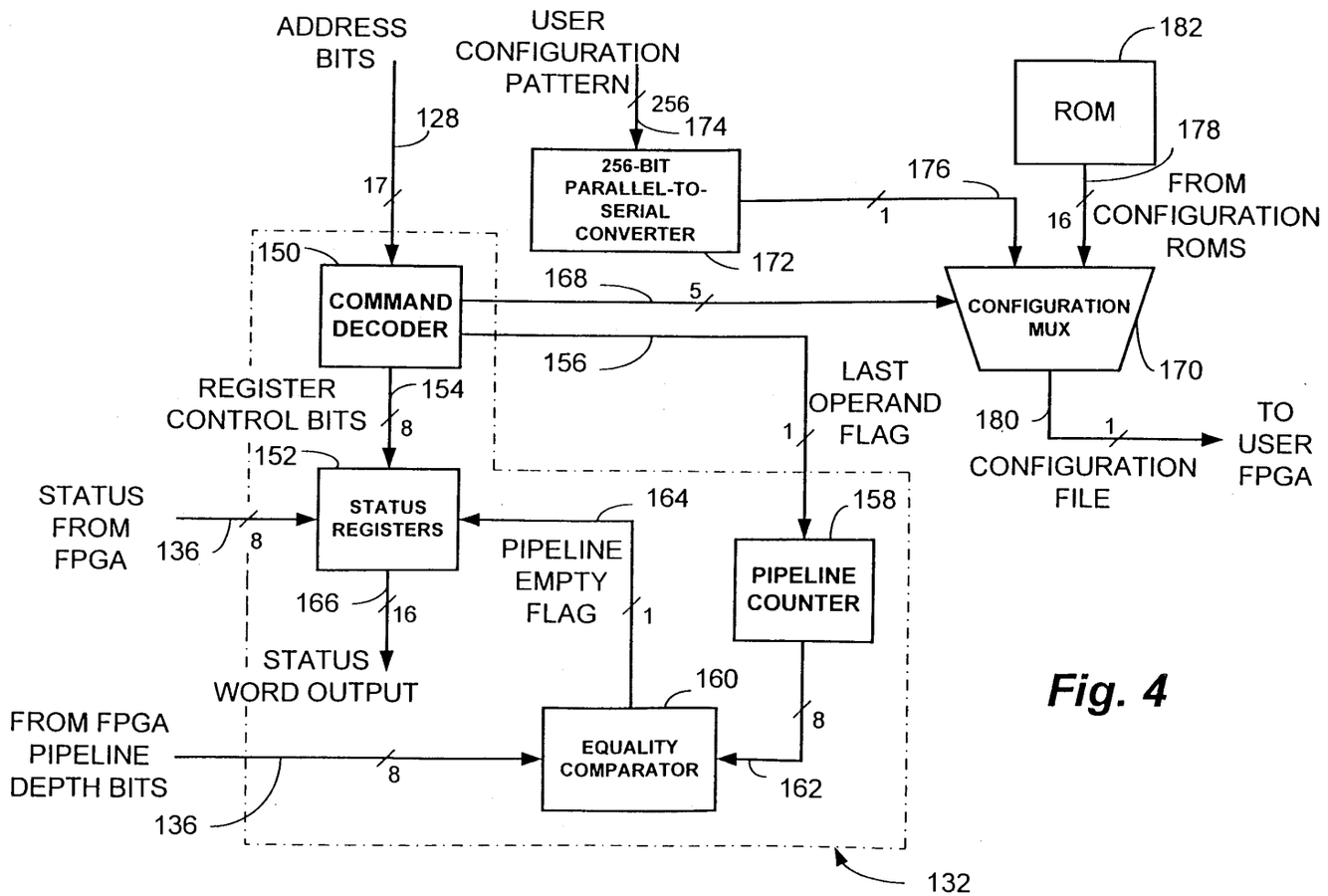


Fig. 4

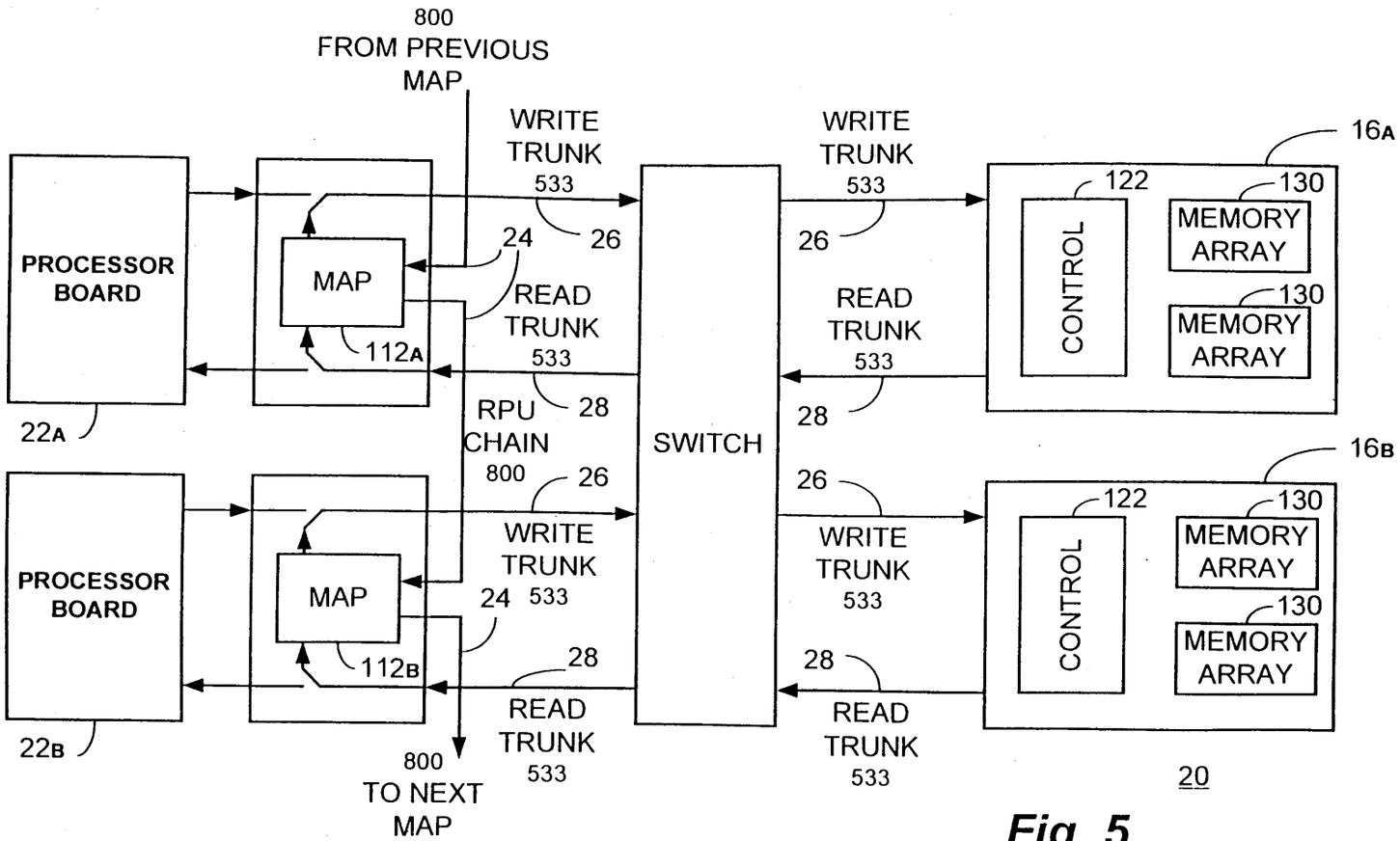


Fig. 5

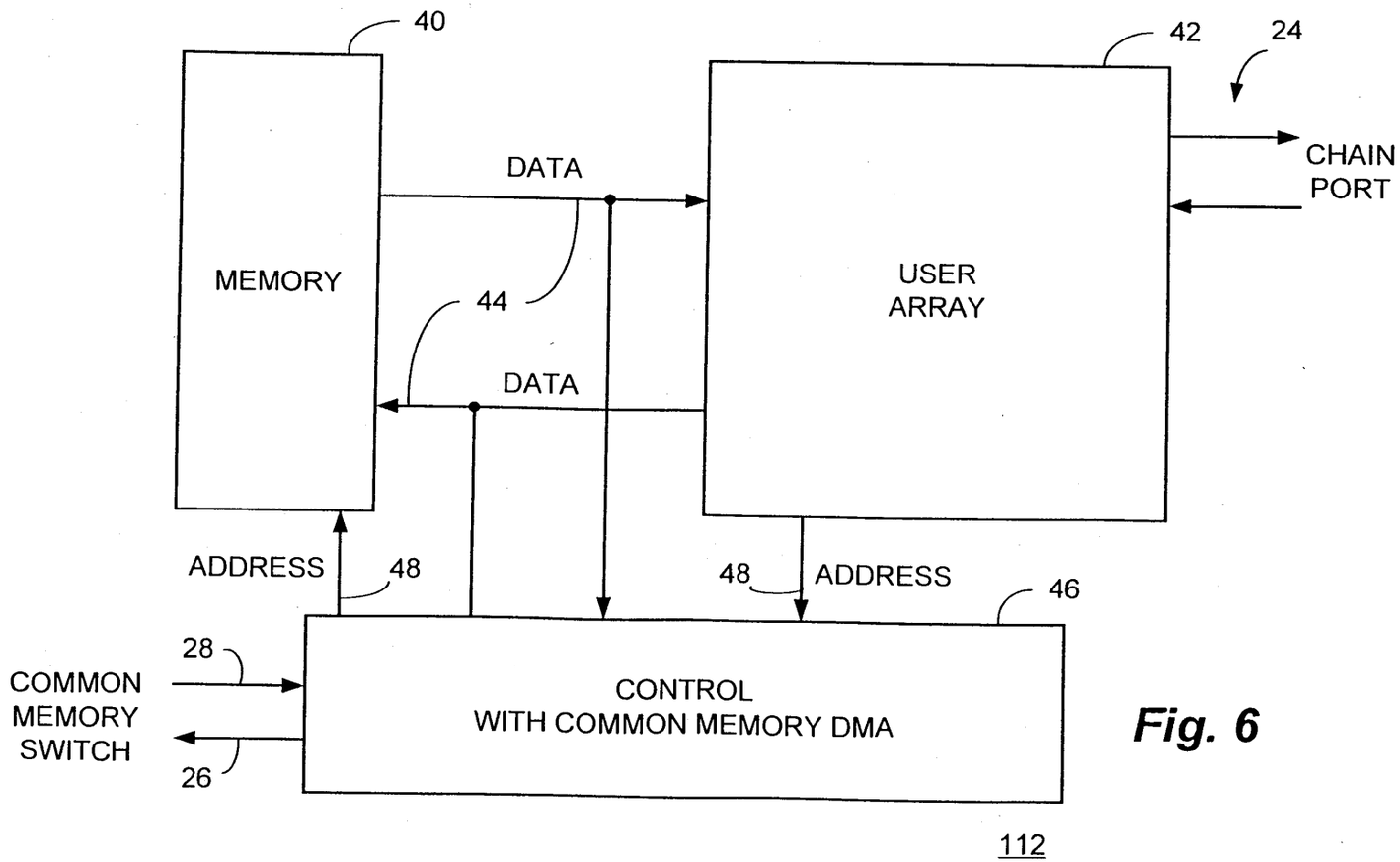


Fig. 6

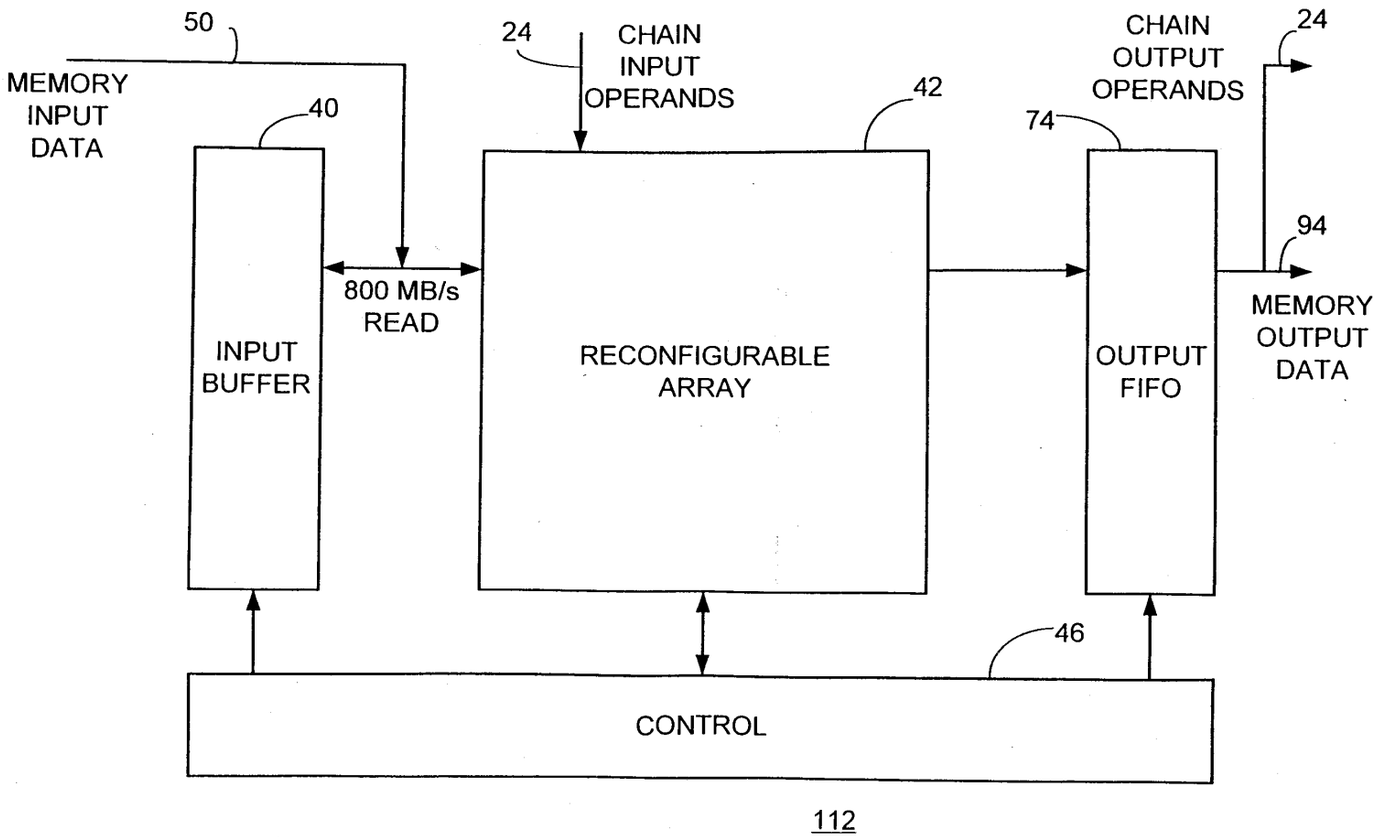


Fig. 7

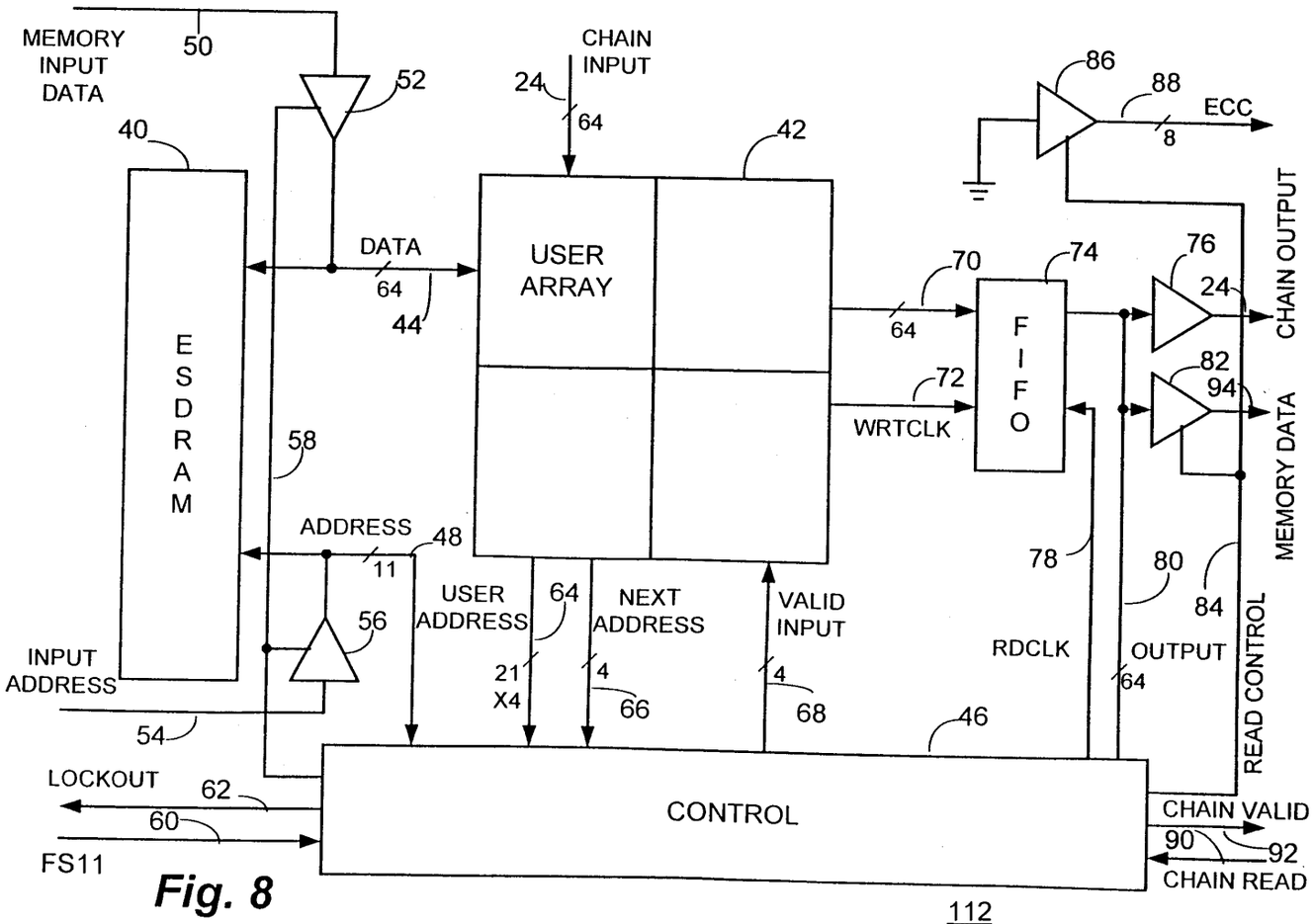

MEMORY INPUT DATA
50
52
CHAIN INPUT
24
64
86
88
ECC
8
40
42
DATA
64
44
USER ARRAY
70
74
76
24
CHAIN OUTPUT
64
F I F O
72
82
94
WRTCLK
MEMORY DATA
E S D R A M
58
ADDRESS
48
11
78
80
84
56
USER ADDRESS
64
NEXT ADDRESS
VALID INPUT
RDCLK
OUTPUT
64
READ CONTROL
INPUT ADDRESS
21
X4
4
66
4
68
46
54
LOCKOUT
62
60
FS11
CONTROL
CHAIN VALID
90
92
CHAIN READ
112

Fig. 8

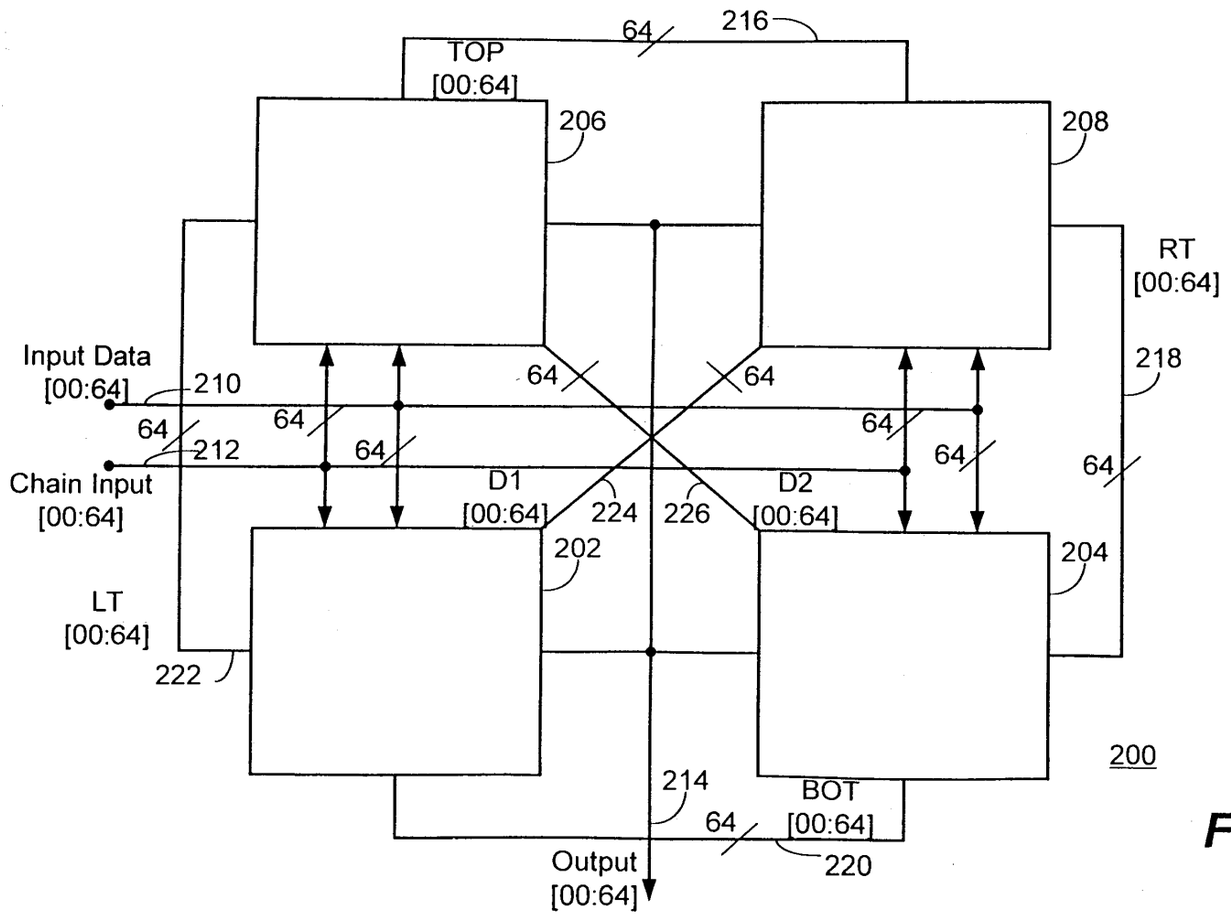


Fig. 9

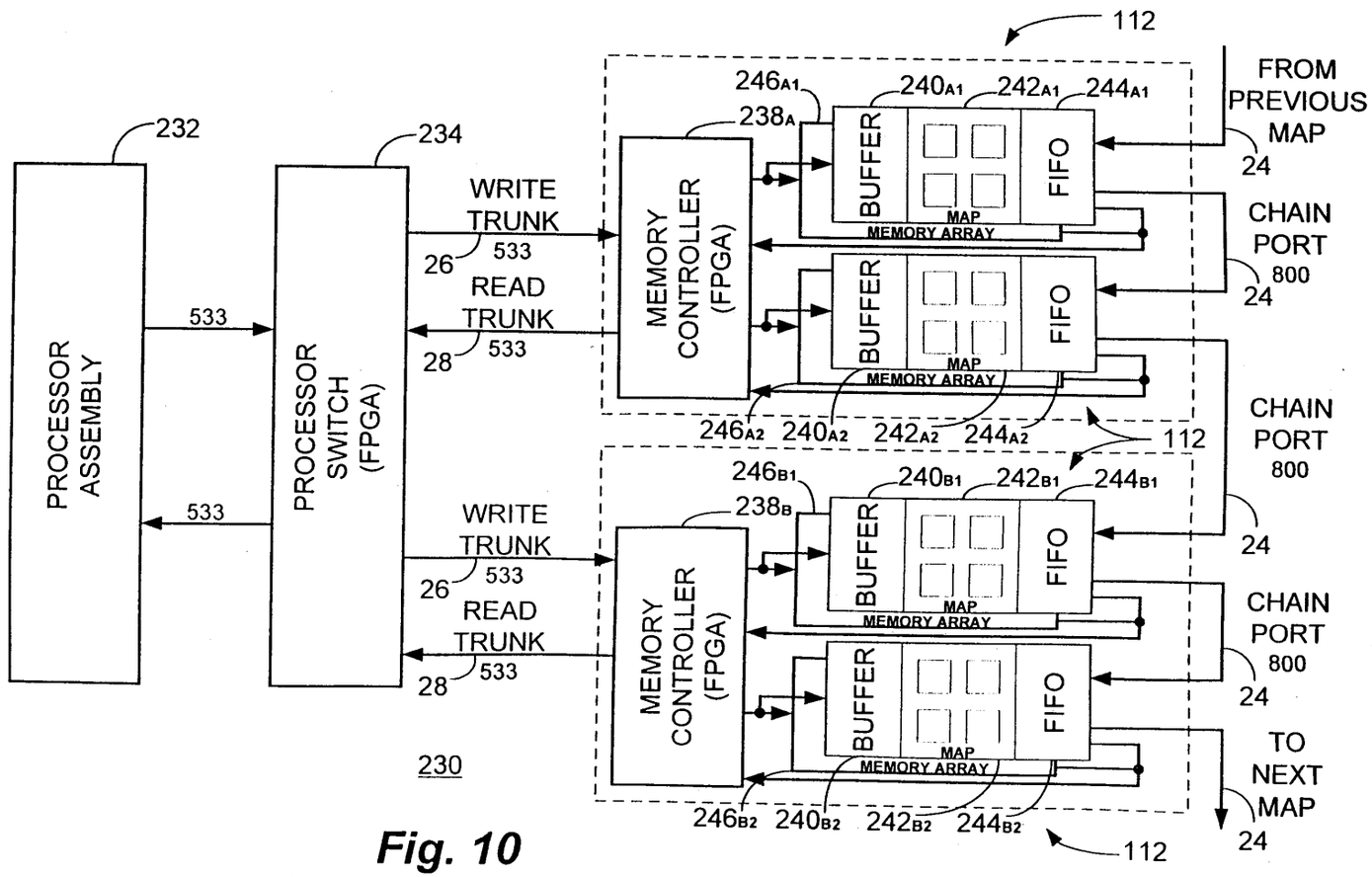

112
FROM PREVIOUS MAP
24
246A1
240A1
242A1
244A1
238A
BUFFER
FIFO
MAP MEMORY ARRAY
CHAIN PORT 800
24
232
234
PROCESSOR ASSEMBLY
PROCESSOR SWITCH (FPGA)
WRITE TRUNK 533
26
READ TRUNK 533
28
MEMORY CONTROLLER (FPGA)
533
533
246A2
240A2
242A2
244A2
112
246B1
240B1
242B1
244B1
238B
CHAIN PORT 800
24
TO NEXT MAP
24
246B2
240B2
242B2
244B2
112
230

Fig. 10

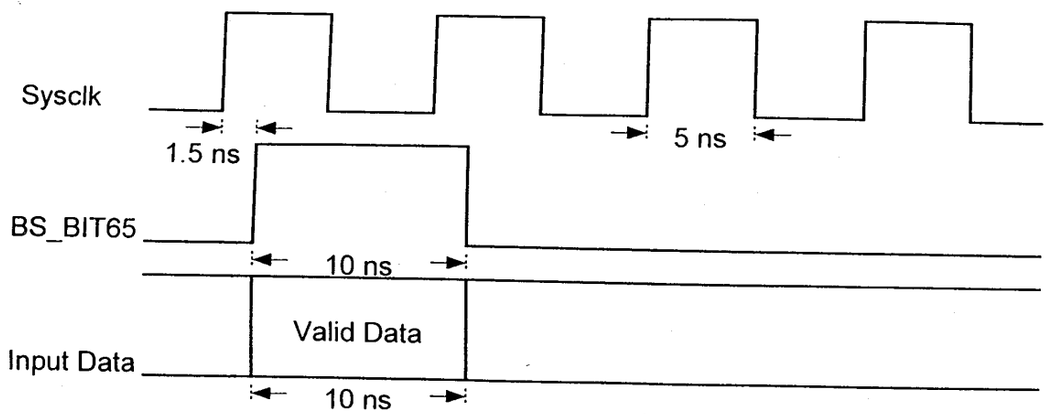


Fig. 11A

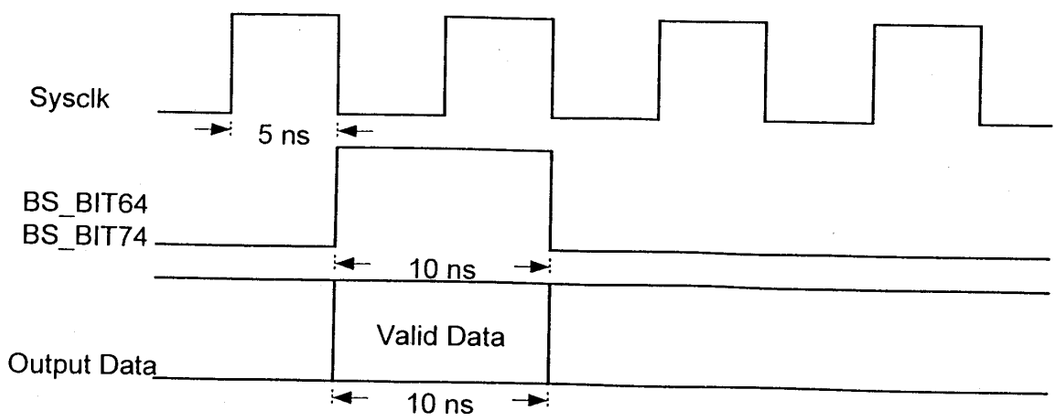
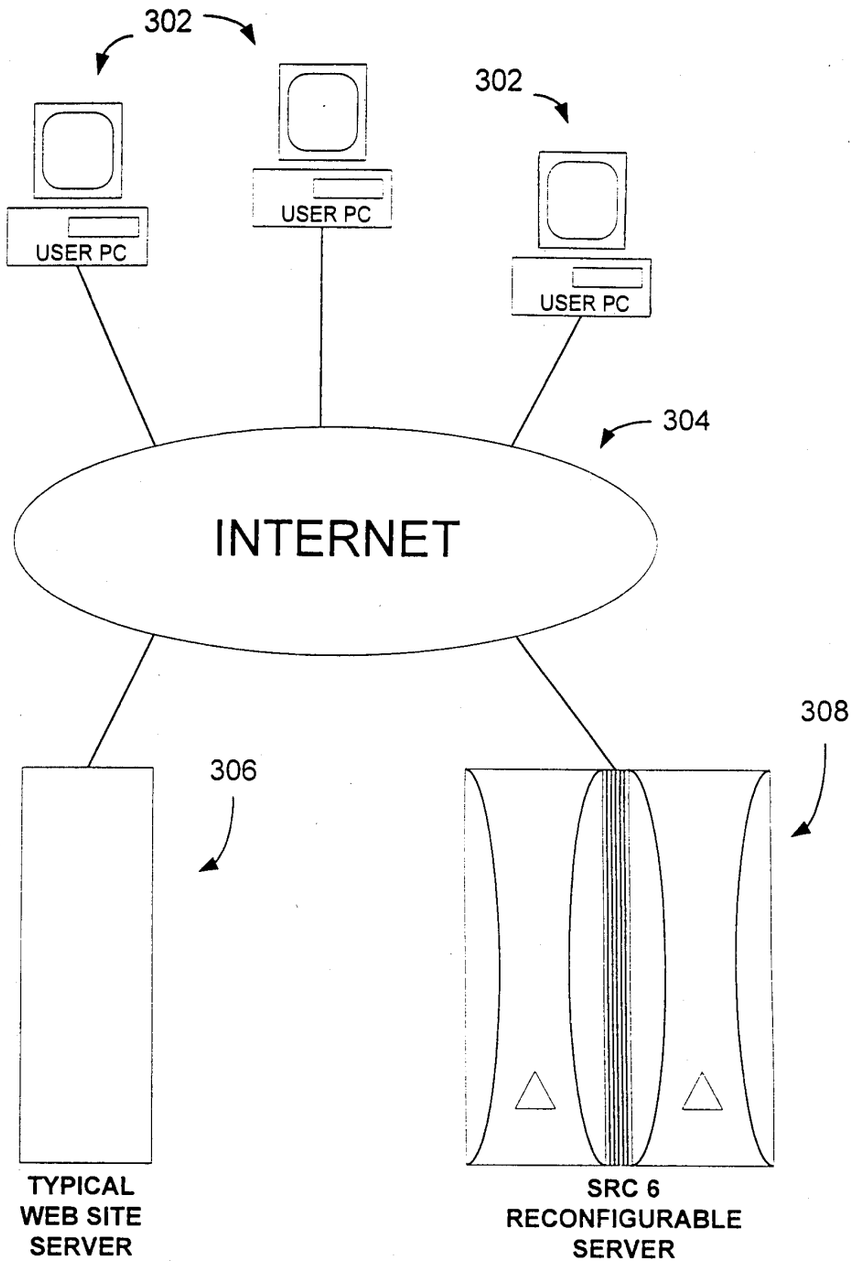


Fig. 11B


302
302
USER PC
USER PC
USER PC
304
INTERNET
306
308
TYPICAL
WEB SITE
SERVER
SRC 6
RECONFIGURABLE
SERVER

300

Fig. 12

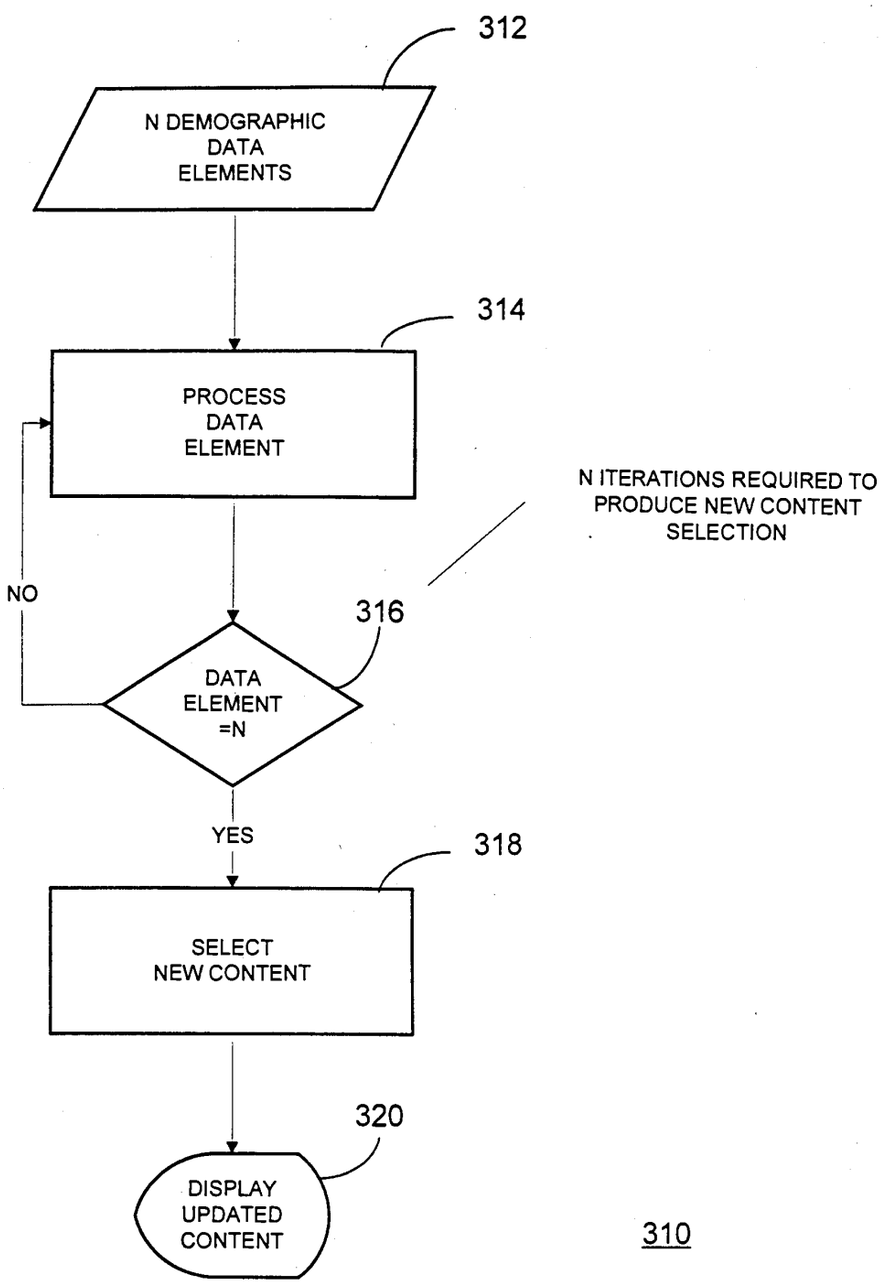


Fig. 13
Prior Art

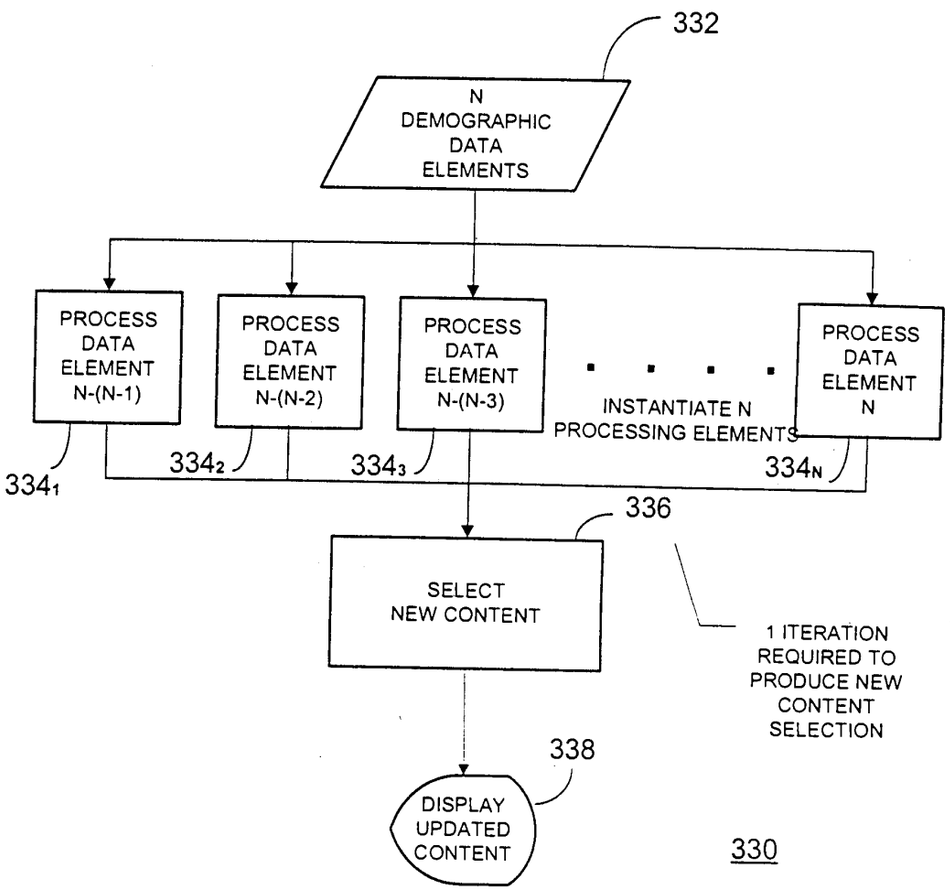

332
N
DEMOGRAPHIC
DATA
ELEMENTS
PROCESS
DATA
ELEMENT
N-(N-1)
PROCESS
DATA
ELEMENT
N-(N-2)
PROCESS
DATA
ELEMENT
N-(N-3)
INSTANTIATE N
PROCESSING ELEMENTS
PROCESS
DATA
ELEMENT
N
334_1
334_2
334_3
334_N
336
SELECT
NEW CONTENT
1 ITERATION
REQUIRED TO
PRODUCE NEW
CONTENT
SELECTION
338
DISPLAY
UPDATED
CONTENT
330

Fig. 14

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER OF PATENTS AND TRADEMARKS
Washington, D.C. 20231
www.uspto.gov

NOTICE OF ALLOWANCE AND FEE(S) DUE

7590 02/21/2002

Hogan & Hartson, LLP
Suite 1500
1200 17th Street
Denver, CO 80202

EXAMINER	
KIM, KENNETH S	
ART UNIT	**CLASS-SUBCLASS**
2183	712-032000

DATE MAILED: 02/21/2002

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/888,276	06/22/2001	Jon M. Huppenthal	SRC010	9011

TITLE OF INVENTION: SYSTEM AND METHOD FOR ACCELERATING WEB SITE ACCESS AND PROCESSING UTILIZING A COMPUTER SYSTEM INCORPORATING RECONFIGURABLE PROCESSORS OPERATING UNDER A SINGLE OPERATING SYSTEM IMAGE

TOTAL CLAIMS	APPLN. TYPE	SMALL ENTITY	ISSUE FEE	PUBLICATION FEE	TOTAL FEE(S) DUE	DATE DUE
25	nonprovisional	NO	$1280	$300	$1580	05/21/2002

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE REFLECTS A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE APPLIED IN THIS APPLICATION. THE PTOL-85B (OR AN EQUIVALENT) MUST BE RETURNED WITHIN THIS PERIOD EVEN IF NO FEE IS DUE OR THE APPLICATION WILL BE REGARDED AS ABANDONED.

HOW TO REPLY TO THIS NOTICE:

I. Review the SMALL ENTITY status shown above. If the SMALL ENTITY is shown as YES, verify your current SMALL ENTITY status:

A. If the status is changed, pay the PUBLICATION FEE (if required) and twice the amount of the ISSUE FEE shown above and notify the United States Patent and Trademark Office of the change in status, or

B. If the status is the same, pay the TOTAL FEE(S) DUE shown above.

If the SMALL ENTITY is shown as NO:

A. Pay TOTAL FEE(S) DUE shown above, or

B. If applicant claimed SMALL ENTITY status before, or is now claiming SMALL ENTITY status, check the box below and enclose the PUBLICATION FEE and 1/2 the ISSUE FEE shown above.

☐ Applicant claims SMALL ENTITY status. See 37 CFR 1.27.

II. PART B - FEE(S) TRANSMITTAL should be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). Even if the fee(s) have already been paid, Part B - Fee(s) Transmittal should be completed and returned. If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed and an extra copy of the form should be submitted.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Box ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

PTOL-85 (REV. 07-01) Approved for use through 01/31/2004.

NM

PART B - FEE(S) TRANSMITTAL

Complete and mail this form, together with applicable fee(s), to:

Box ISSUE FEE
Assistant Commissioner for Patents
Washington, D.C. 20231

MAILING INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 4 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Legibly mark-up with any corrections or use Block 1)

7590 02/21/2002

Hogan & Hartson, LLP
Suite 1500
1200 17th Street
Denver, CO 80202

Note: The certificate of mailing below can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing.

Certificate of Mailing

I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Box Issue Fee address above on the date indicated below.

(Depositor's name)
(Signature)
(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/888,276	06/22/2001	Jon M. Huppenthal	SRC010	9011

TITLE OF INVENTION: SYSTEM AND METHOD FOR ACCELERATING WEB SITE ACCESS AND PROCESSING UTILIZING A COMPUTER SYSTEM INCORPORATING RECONFIGURABLE PROCESSORS OPERATING UNDER A SINGLE OPERATING SYSTEM IMAGE

TOTAL CLAIMS	APPLN. TYPE	SMALL ENTITY	ISSUE FEE	PUBLICATION FEE	TOTAL FEE(S) DUE	DATE DUE
25	nonprovisional	NO	$1280	$300	$1580	05/21/2002

EXAMINER	ART UNIT	CLASS-SUBCLASS
KIM, KENNETH S	2183	712-032000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363). Use of PTO form(s) and Customer Number are recommended, but not required.

❑ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.

❑ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47) attached.

2. For printing on the patent front page, list (1) the names of up to 3 registered patent attorneys or agents OR, alternatively, (2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1 _____

2 _____

3 _____

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. Inclusion of assignee data is only appropriate when an assignment has been previously submitted to the USPTO or is being submitted under separate cover. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE (B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent) ❑ individual ❑ corporation or other private group entity ❑ government

4a. The following fee(s) are enclosed:

❑ Issue Fee

❑ Publication Fee

❑ Advance Order - # of Copies _____

4b. Payment of Fee(s):

❑ A check in the amount of the fee(s) is enclosed.

❑ Payment by credit card. Form PTO-2038 is attached.

❑ The Commissioner is hereby authorized by charge the required fee(s), or credit any overpayment, to Deposit Account Number _____(enclose an extra copy of this form).

The COMMISSIONER OF PATENTS AND TRADEMARKS is requested to apply the Issue Fee and Publication Fee (if any) or to re-apply any previously paid issue fee to the application identified above.

(Authorized Signature) (Date)

NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending on the needs of the individual case. Any comments on the amount of time required to complete this form should be sent to the Chief Information Officer, United States Patent and Trademark Office, Washington, D.C. 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND FEES AND THIS FORM TO: Box Issue Fee, Assistant Commissioner for Patents, Washington, D.C. 20231

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMIT THIS FORM WITH FEE(S)

PTOL-85 (REV. 07-01) Approved for use through 01/31/2004. OMB 0651-0033 U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

5

Notice of Allowability	Application No. 09/888,276	Applicant(s) HUPPENTHAL, JON M.
	Examiner Kenneth S KIM	Art Unit 2183

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *the application filed June 22, 2001*.
2. ☒ The allowed claim(s) is/are *1-25*.
3. ☐ The drawings filed on _____ are accepted by the Examiner.
4. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____.
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.
5. ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).
   (a) ☐ The translation of the foreign language provisional application has been received.
6. ☒ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application. **THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.**

7. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

8. ☒ CORRECTED DRAWINGS must be submitted.
   (a) ☒ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☒ hereto or 2) ☐ to Paper No. ____.
   (b) ☐ including changes required by the proposed drawing correction filed _____, which has been approved by the Examiner.
   (c) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No. _____.

Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the top margin (not the back) of each sheet. The drawings should be filed as a separate paper with a transmittal letter addressed to the Official Draftsperson.

9. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1☒ Notice of References Cited (PTO-892)
2☐ Notice of Informal Patent Application (PTO-152)
3☒ Notice of Draftperson's Patent Drawing Review (PTO-948)
4☐ Interview Summary (PTO-413), Paper No.____.
5☒ Information Disclosure Statements (PTO-1449), Paper No. *8*.
6☐ Examiner's Amendment/Comment
7☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
8☐ Examiner's Statement of Reasons for Allowance
9☐ Other

Match and Return

KENNETH S. KIM
PRIMARY EXAMINER

U.S. Patent and Trademark Office
PTO-37 (Rev. 04-01) **Notice of Allowability** **Part of Paper No. 9 .**

05-23-02

$B

PART B - FEE(S) TRANSMITTAL

Complete and mail this form, together with applicable fee(s), to:

Box ISSUE FEE
Assistant Commissioner for Patents
Washington, D.C. 20231

MAILING INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 4 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Legibly mark-up with any corrections or use Block 1)

7590 02/21/2002

Hogan & Hartson, LLP
Suite 1500
1200 17th Street
Denver, CO 80202

OIPE JC144 MAY 21 2002 PATENT & TRADEMARK OFFICE

Note: The certificate of mailing below can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing.

Certificate of Mailing

I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for ~~first class mail~~ in an envelope addressed to the Box Issue Fee address above on the date indicated below. Express Mail No. EV035492075US

Julie Lange (Depositor's name)

(Signature)

21 May 2002 (Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/888,276	06/22/2001	Jon M. Huppenthal	SRC010	9011

TITLE OF INVENTION: SYSTEM AND METHOD FOR ACCELERATING WEB SITE ACCESS AND PROCESSING UTILIZING A COMPUTER SYSTEM INCORPORATING RECONFIGURABLE PROCESSORS OPERATING UNDER A SINGLE OPERATING SYSTEM IMAGE

TOTAL CLAIMS	APPLN. TYPE	SMALL ENTITY	ISSUE FEE	PUBLICATION FEE	TOTAL FEE(S) DUE	DATE DUE
25	nonprovisional	NO	$1280	$300	$1580	05/21/2002

EXAMINER	ART UNIT	CLASS-SUBCLASS
KIM, KENNETH S	2183	712-032000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363). Use of PTO form(s) and Customer Number are recommended, but not required.

☐ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.

☐ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47) attached.

2. For printing on the patent front page, list (1) the names of up to 3 registered patent attorneys or agents OR, alternatively, (2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1 William J. Kubida
2 Hogan & Hartson LLP
3

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. Inclusion of assignee data is only appropriate when an assignment has been previously submitted to the USPTO or is being submitted under separate cover. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE: SRC Computers, Inc.

(B) RESIDENCE: (CITY and STATE OR COUNTRY): Colorado Springs, Colorado

Please check the appropriate assignee category or categories (will not be printed on the patent) ☐ individual ☒ corporation or other private group entity ☐ government

4a. The following fee(s) are enclosed:

☒ Issue Fee
☒ Publication Fee
☒ Advance Order - # of Copies 10

4b. Payment of Fee(s):

☒ A check in the amount of the fee(s) is enclosed.
☐ Payment by credit card. Form PTO-2038 is attached.
☒ The Commissioner is hereby authorized to charge any deficiencies ~~the required fee(s)~~, or credit any overpayment, to Deposit Account Number 50-1123 (enclose an extra copy of this form).

The COMMISSIONER OF PATENTS AND TRADEMARKS is requested to apply the Issue Fee and Publication Fee (if any) or to re-apply any previously paid issue fee to the application identified above.

(Authorized Signature) (Date) 20 May 2002

NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

05/24/2002 BMSUYEN2 00000116 09888276

01 FC:195 300.00 OP
02 FC:142 1280.00 OP
03 FC:561 30.00 OP

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending on the needs of the individual case. Any comments on the amount of time required to complete this form should be sent to the Chief Information Officer, United States Patent and Trademark Office, Washington, D.C. 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND FEES AND THIS FORM TO: Box Issue Fee, Assistant Commissioner for Patents, Washington, D.C. 20231

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMIT THIS FORM WITH FEE(S)

EXHIBIT O

Mailed January 8, 2008

NOTICE OF GROUNDS OF REJECTION

Patent Application No.	539441/2000
Drafting Date	December 26, 2007
Patent Office Examiner	Masanori KUBO (9642 5B00)
Attorney	Mr. Hisao Fukami (et al.)
Applied Provision	Paragraph 2 of Article 29, Article 36

The present application is recognized as rejected on the following ground. It is required that any remarks be submitted within three months from the date on which the present NOTICE was mailed.

GROUNDS

1. It is recognized that, because the invention described in Claim(s) of SCOPE OF CLAIMS FOR PATENT of the present application could have been invented readily by a person having ordinary knowledge in the field of the art to which the present invention pertains prior to the filing of the present application based on the invention as described in the following publication(s) distributed or the invention as made available to the public through electric telecommunication lines in Japan and/or foreign countries prior to the filing of the present application, a patent cannot be granted thereto under the provision of Paragraph 2 of Article 29 of the Patent Law.

2. It is recognized that the present application does not satisfy the conditions prescribed in Paragraph 6 (ii) of Article 36 of the Patent Law because of the defectiveness of the description in SCOPE OF CLAIMS FOR PATENT on the following point.

REMARKS (See the list of the cited references.)

With regard to Ground 1

- Claims 1-81
- Cited References 1-2
- Note

Cited references 1 and 2 each describe that the processing instruction is given from a main control unit to a plurality of processors through a memory. Cited reference 1 also describes that the processing result is transferred to a different processor. There is no particular difficulty in constructing each processor with a well-known reconfigurable circuit and providing a plurality of main control units.

With regard to Ground 2

The description of the "improvement" in claims 1 to 7 is unclear since it cannot be specified whether it is the "product invention" or the "method invention".

Therefore, the invention according to claims 1 to 7 is unclear.

If any grounds of rejection are newly found, the grounds of rejection will be noticed.

LIST OF CITED REFERENCES

(1) Japanese Patent Laying-Open No. 63-086079

(2) Japanese Patent Laying-Open No. 59-206972

Record of Search for Prior Art Documents

* Searched Technical Field IPC G06F15/80

G06F15/16-15/177

* Prior Art Documents

Japanese Patent Laying-Open No. 11-015773

This record of search for prior art documents does not form any grounds of rejection.

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 59-206972
(43)Date of publication of application : 22.11.1984

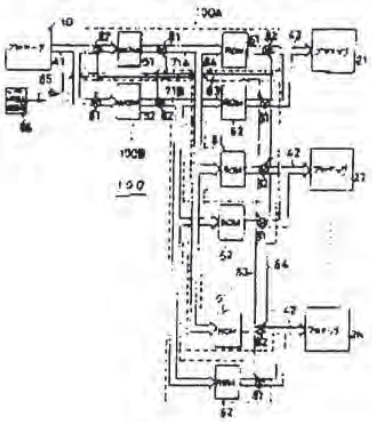
(51)Int.Cl. G06F 15/16

(21)Application number : 58-081318
(22)Date of filing : 10.05.1983

(71)Applicant : TOSHIBA CORP
(72)Inventor : FUJII MAKOTO

(54) SHARED MEMORY

(57)Abstract:
PURPOSE: To eliminate interruption of processors at the time of data transfer between processors by providing plural write-only memories in the input port of a public memory and plural read-only memories in the output port.
CONSTITUTION: Write-only memories 51, 52 that write data from a processor 10 are provided in the input port of a shared memory 100, and read-only memories 61, 62 that read data to processors 21W2N are provided in output ports. Gates 81, 82 that determine transfer mode of data are provided in an A port 100A and a B port 100B. The gate 81 is connected to a change-over signal generating circuit 86, and the gate 82 is connected to a mode changing signal generating circuit 86 through a controlling line 84 and an invertor 85 for inverting signals. By this way, transfer mode of the A port 100A and B port 100B become reverse.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's

decision of rejection]
[Date of extinction of right]

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 63-086079
(43)Date of publication of application : 16.04.1988

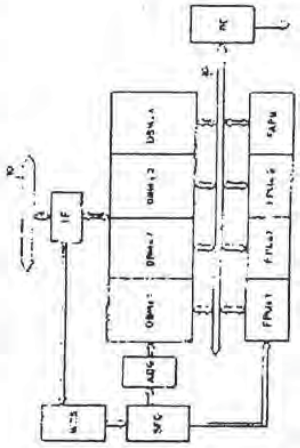
(51)Int.Cl. G06F 15/60
G06F 15/16
G06F 15/347

(21)Application number : 61-232436
(22)Date of filing : 30.09.1986

(71)Applicant : NIPPON TELEGR & TELEPH CORP <NTT>
(72)Inventor : TAMAMURA YOSHIAKI
MITSUYA EIJI
AKIMOTO TAKAAKI

(54) THREE-DIMENSIONAL SHADOW IMAGE FORMING PROCESSING DEVICE

(57)Abstract:
PURPOSE: To attain a highly speedy three-dimensional shadow image forming processing by executing a three-dimensional vector operation and a matrix operation with 3W4 floating point arithmetic units in parallel and in a pipeline way.
CONSTITUTION: Object shape data and a processing parameter used for image forming processing are stored into data memories DBM#1W#4. By floating point arithmetic units FPU#1W#3 and an arithmetic unit FAPU to combine a floating point computing element and an arithmetic and logic computing element in parallel, the three-dimensional vector operation and the matrix operation are executed in parallel and in a pipeline way. The prepared image data are written through a data collector DC to a display memory.



LEGAL STATUS
[Date of request for examination]
[Date of sending the examiner's decision of rejection]
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
[Date of final disposal for application]
[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/733,064	04/09/2007	Jon M. Huppenthal	SRC015 CON	7527

25235 7590 01/12/2009
HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

EXAMINER	
COLEMAN, ERIC	
ART UNIT	PAPER NUMBER
2183	
MAIL DATE	DELIVERY MODE
01/12/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

PTOL-90A (Rev. 04/07)

Office Action Summary	Application No.	Applicant(s)	
	11/733,064	HUPPENTHAL ET AL.	
	Examiner	Art Unit	
	Eric Coleman	2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☐ Responsive to communication(s) filed on _____.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-52 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-52 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date 4/9/07,3/19/08.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Claim 30 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Claim 20 recites the limitation "said first systolic wall" in line 3. There is insufficient antecedent basis for this limitation in the claim.

Double Patenting

The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. A nonstatutory obviousness-type double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claim(s) because the examined application claim is either anticipated by, or would have been obvious over, the reference claim(s). See, e.g., *In re Berg*, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent either is shown to be commonly owned with this application, or claims an invention made as a result of activities undertaken within the scope of a joint research agreement.

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

Claims 1-52 are rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claims 1-52 of U.S. Patent No. 7,225,324 in view of Gaudiot IEEE article entitled Data Driven Multicomputers in Digital Signal processing. The side by showing of the corresponding independent claims 1,25,51 show that the corresponding independent claims are substantially similar.

Instant application	Patent No. 7,225, 324
1. A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor, the reconfigurable processor comprising a plurality of functional units, said method comprising: transforming an algorithm into a **data driven** calculation that is implemented by said reconfigurable computing system at the at least one reconfigurable processor; **forming** at least two of said functional units at the at least one reconfigurable processor to perform said calculation wherein only functional units needed to solve the calculation are **formed** and wherein each **formed** functional unit at the at least one reconfigurable processor interconnects with each other **formed** functional unit at the at least one reconfigurable processor based	1. A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor, the reconfigurable processor comprising a plurality of functional units, said method comprising: transforming an algorithm into a calculation that is systolically implemented by said reconfigurable computing system at the at least one reconfigurable processor; instantiating at least two of said functional units at the at least one reconfigurable processor to perform said calculation wherein only functional units needed to solve the calculation are instantiated and wherein each instantiated functional unit at the at least one reconfigurable processor interconnects with each other instantiated functional unit at the at least one reconfigurable processor based

on reconfigurable routing resources within the at least one reconfigurable processor as established at **formation**, and wherein lines of code of said calculation are **formed** as clusters of functional units within the at least one reconfigurable processor; utilizing a first of said instantiated functional units to operate upon a subsequent data dimension of said calculation forming a first computational loop; and substantially concurrently utilizing a second of said formed functional units to operate upon a previous data dimension of said calculation **generating** a second computational loop wherein said implementation of said calculation enables said first computational loop and said second computational loop execute concurrently and pass computed data seamlessly between said computational loops.	on reconfigurable routing resources within the at least one reconfigurable processor as established at instantiation, and wherein **systolically linked** lines of code of said calculation are instantiated as clusters of functional units within the at least one reconfigurable processor; utilizing a first of said instantiated functional units to operate upon a subsequent data dimension of said calculation forming a first computational loop; and substantially concurrently utilizing a second of said instantiated functional units to operate upon a previous data dimension of said calculation forming a second computational loop wherein said **systolic** implementation of said calculation enables said first computational loop and said second computational loop execute concurrently and pass computed data seamlessly between said computational loops.
25. A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one	25. A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one

reconfigurable processor comprising a plurality of functional units, said method comprising: transforming an algorithm into a **data driven** calculation that is implemented by said reconfigurable computing system at the at least one reconfigurable processor wherein linked lines of code of said calculation are **fashioned** as walls of functional units within the at least one reconfigurable processor; defining a first wall comprising rows of cells forming a subset of said plurality of functional units; computing at the at least one reconfigurable processor a value at each of said cells in at least a first row of said first substantially concurrently; communicating said values between cells in said first row of said cells to produce updated values, wherein communicating said values is based on reconfigurable routing resources within the at least one reconfigurable processor; communicating said updated values substantially concurrently to a second row of said first wall, wherein communicating said updated values is based on reconfigurable routing	reconfigurable processor comprising a plurality of functional units, said method comprising: transforming an algorithm into a calculation that is **systolically** implemented by said reconfigurable computing system at the at least one reconfigurable processor wherein **systolically** linked lines of code of said calculation are instantiated as walls of functional units within the at least one reconfigurable processor; defining a first **systolic** wall comprising rows of cells forming a subset of said plurality of functional units; computing at the at least one reconfigurable processor a value at each of said cells in at least a first row of said first **systolic wall** substantially concurrently; communicating said values between cells in said first row of said cells to produce updated values, wherein communicating said values is based on reconfigurable routing resources within the at least one reconfigurable processor; communicating said updated values substantially concurrently to a second row of said first **systolic** wall, wherein communicating said updated values is based on reconfigurable routing

resources within the at least one reconfigurable processor; and communicating said updated values substantially concurrently to a first row of a second wall of rows of cells in said subset of said plurality of functional units, wherein communicating said updated values is based on reconfigurable routing resources within the at least one reconfigurable processor and wherein said first wall of rows of cells and said second wall of rows of cells execute substantially concurrently and pass computed data seamlessly between said walls.	resources within the at least one reconfigurable processor; and communicating said updated values substantially concurrently to a first row of a second systolic wall of rows of cells in said subset of said plurality of functional units, wherein communicating said updated values is based on reconfigurable routing resources within the at least one reconfigurable processor and wherein said first **systolic** wall of rows of cells and said second wall of rows of **systolic** cells execute substantially concurrently and pass computed data seamlessly between said **systolic** walls.
51. A method for data processing in a reconfigurable computing system, the reconfigurable computer system comprising at least one reconfigurable processor comprising a plurality of functional units, said method comprising: transforming an algorithm into a calculation that is implemented by said reconfigurable computing system at the at least one reconfigurable processor **and driven by data propagation** wherein lines of code of said calculation are	51. A method for data processing in a reconfigurable computing system, the reconfigurable computer system comprising at least one reconfigurable processor comprising a plurality of functional units, said method comprising: transforming an algorithm into a calculation that is **systolically** implemented by said reconfigurable computing system at the at least one reconfigurable processor wherein **systolically** linked lines of code of said calculation are

linked **based on said data propagation and fashioned** as subsets of said plurality of functional units within the at least one reconfigurable processor forming columns of said calculation; performing said calculation at the at least one reconfigurable processor by said subsets of said plurality of functional units to produce computed data; exchanging said computed data between a first column of said calculation and a next column in said calculation, wherein said exchanging is based on reconfigurable routing resources within the at least one reconfigurable processor and wherein execution of said subsets of said plurality of function units occurs concurrently and said computed data is seamlessly passed between said first column of said calculation and said second column of said calculation; evaluating a rate of change in at least one variable for each of said columns in said calculation; continuing said calculation when said variable does not change for a particular column of said calculation; and restarting said calculation at said column of said calculation where said variable does change.	**instantiated** as subsets of said plurality of functional units within the at least one reconfigurable processor forming columns of said calculation; performing said calculation at the at least one reconfigurable processor by said subsets of said plurality of functional units to produce computed data; exchanging said computed data between a first column of said calculation and a next column in said calculation, wherein said exchanging is based on reconfigurable routing resources within the at least one reconfigurable processor and wherein execution of said subsets of said plurality of function units occurs concurrently and said computed data is seamlessly passed between said first column of said calculation and said second column of said calculation; evaluating a rate of change in at least one variable for each of said columns in said calculation; continuing said calculation when said variable does not change for a particular column of said calculation; and restarting said calculation at said column of said calculation where said variable does change.

The claims in the 7,722,324 patent did not expressly detail that the method included *data driven* calculation or propagation of data. However Gaudiot taught that data driven processing and calculations includes systolic processing (e.g., see page 1224 (section IV) and page 1230 (section H). Therefore since patent 7,722,324 claims systolic calculation and propagation and linking of data and the instant application claims data driven calculation and data propagation it would have been obvious to one of ordinary skill in the DP system that the claimed calculation and data propagation are of the same type. Gaudiot taught that the use of one of the several types of data driven processing or calculation would have been more advantageous depending on the features of the calculation however each is considered data driven (e.g., see page 1230 (section H)). The patent 722,324 used forms of the word "instantiate" versus forms of the words "form" or "fashioned" in the instant claims . The claimed instantiating or representing by a concrete instance is not different from the claimed forming because the fashioning or forming (claimed in the instant application) is done by representing data or functional unit by concrete references to data or functional units or other system elements. Claims 2-24, 25-50 and 52 provide the same corresponding limitations in the Patent 7,224,324 and the instant application and therefore are also rejected. It would have been obvious to one of ordinary skill in the DP art to combine the claims of patent No. 7,722,324 and Gaudiot. Both references were directed toward processing data parallel processing of data in a DP system. The addition of the Gaudiot teaching

would have provided ways for optimizing the processing of calculations depending on the attributes of the calculation that was performed by systolic means considering that the systolic processing is a subset of data driven processing. Also the addition of the Gaudiot teachings would have yielded predictable results.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Dennis,J.B. (IEEE Computer chapter on DataFlow Supercomputers).

Quinn, M et al., (IEEE article entitled Data-Parallel Programming on Multicomputers).

Treleaven,P.C et al., Computing Surveys article entitled Data-Driven and Demand-Driven Computer Architecture.

Webster's Ninth New Collegiate Dictionary, (definition of instantiate),p. 627.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eric Coleman whose telephone number is (571) 272-4163. The examiner can normally be reached on Monday-Thursday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

EC
/Eric Coleman/
Primary Examiner, Art Unit 2183

Client Matter No. 80404.0018.001
EFS-Web

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Serial No. 11/733,064 Application of: Jon M. Huppenthal and David E. Caliga Filed: April 9, 2007 Attorney Docket No. SRC015 CON For: MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS	Confirmation No.: 7527 Art Unit: 2183 Examiner: Coleman, Eric Customer No.: **25235**

AMENDMENT

MAIL STOP AMENDMENT
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the office communication mailed January 12, 2009, please amend the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims which begins on page 2 of this paper.

Remarks/Arguments begin on page 10 of this paper.

A **Terminal Disclaimer** is attached following page 11 of this paper.

\\\CS - 080404/000018 - 110334 v1

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1. (original) A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor, the reconfigurable processor comprising a plurality of functional units, said method comprising:

transforming an algorithm into a data driven calculation that is implemented by said reconfigurable computing system at the at least one reconfigurable processor;

forming at least two of said functional units at the at least one reconfigurable processor to perform said calculation wherein only functional units needed to solve the calculation are formed and wherein each formed functional unit at the at least one reconfigurable processor interconnects with each other formed functional unit at the at least one reconfigurable processor based on reconfigurable routing resources within the at least one reconfigurable processor as established at formation, and wherein lines of code of said calculation are formed as clusters of functional units within the at least one reconfigurable processor;

utilizing a first of said formed functional units to operate upon a subsequent data dimension of said calculation forming a first computational loop; and

substantially concurrently utilizing a second of said formed functional units to operate upon a previous data dimension of said calculation generating a

second computational loop wherein said implementation of said calculation enables said first computational loop and said second computational loop execute concurrently and pass computed data seamlessly between said computational loops.

2. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple vectors in said calculation.

3. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple planes in said calculation.

4. (original) The method of claim 1 wherein said subsequent and previous data dimensions of said calculation comprise multiple time steps in said calculation.

5. (original) The method of claim 1 wherein said subsequent an previous data dimensions of said calculation comprise multiple grid points in said calculation.

6. (original) The method of claim 1 wherein said calculation comprises a seismic imaging calculation.

7. (original) The method of claim 1 wherein said calculation comprises a synthetic aperture radar imaging calculation.

8. (original) The method of claim 1 wherein said calculation comprises a JPEG image compression calculation.

9. (original) The method of claim 1 wherein said calculation comprises an MPEG image compression calculation.

10. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for a reservoir simulation.

11. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for weather prediction.

12. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for automotive applications.

13. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for aerospace applications.

14. (original) The method of claim 1 wherein said calculation comprises a fluid flow calculation for an injection molding application.

15. (original) The method of claim 1 wherein instantiating includes establishing a stream communication connection between functional units.

16. (original) The method of claim 1 wherein said calculation is comprises a structures calculation for structural analysis.

17. (original) The method of claim 1 wherein said calculation comprises a search algorithm for an image search.

18. (original) The method of claim 1 wherein said calculation comprises a search algorithm for data mining.

19. (original) The method of claim 1 wherein said calculation comprises a financial modeling application.

20. (original) The method of claim 1 wherein said calculation comprises an encryption algorithm.

21. (original) The method of claim 1 wherein said calculation comprises a genetic pattern matching function.

22. (original) The method of claim 1 wherein said calculation comprises a protein folding function.

23. (original) The method of claim 1 wherein said calculation comprises an organic structure interaction function.

24. (original) The method of claim 1 wherein said calculation comprises a signal filtering application.

25. (original) A method for data processing in a reconfigurable computing system, the reconfigurable computing system comprising at least one reconfigurable processor comprising a plurality of functional units, said method comprising:

transforming an algorithm into a data driven calculation that is implemented by said reconfigurable computing system at the at least one reconfigurable processor wherein linked lines of code of said calculation are fashioned as walls of functional units within the at least one reconfigurable processor;

defining a first wall comprising rows of cells forming a subset of said plurality of functional units;

computing at the at least one reconfigurable processor a value at each of said cells in at least a first row of said first wall substantially concurrently;

communicating said values between cells in said first row of said cells to produce updated values, wherein communicating said values is based on reconfigurable routing resources within the at least one reconfigurable processor;

communicating said updated values substantially concurrently to a second row of said first wall, wherein communicating said updated values is based on reconfigurable routing resources within the at least one reconfigurable processor; and

communicating said updated values substantially concurrently to a first row of a second wall of rows of cells in said subset of said plurality of functional units, wherein communicating said updated values is based on reconfigurable routing resources within the at least one reconfigurable processor and wherein said first wall of rows of cells and said second wall of rows of cells execute substantially concurrently and pass computed data seamlessly between said walls.

26. (original) The method of claim 25 wherein said values correspond to vectors in a computation.

27. (original) The method of claim 25 wherein said values correspond to planes in a computation.

28. (original) The method of claim 25 wherein said values correspond to time steps in a computation.

29. (original) The method of claim 25 wherein said values correspond to grid points in a computation.

30. (currently amended) The method of claim 25 wherein said step of communicating said updated values to a second row of said first ~~systolic~~ wall is carried out without storing said updated values in an extrinsic memory.

31. (original) The method of claim 25 wherein said values correspond to a seismic imaging calculation.

32. (original) The method of claim 25 wherein said values correspond to a synthetic aperture radar imaging calculation.

33. (original) The method of claim 25 wherein said values correspond to a JPEG image compression calculation.

34. (original) The method of claim 25 wherein said values correspond to an MPEG image compression calculation.

35. (original) The method of claim 25 wherein said values correspond to a fluid flow calculation for a reservoir simulation.

36. (original) The method of claim 25 wherein said values correspond to a fluid flow calculation for weather prediction.

37. (original) The method of claim 25 wherein said values correspond to a fluid flow calculation for automotive applications.

38. (original) The method of claim 25 wherein said values correspond to a fluid flow calculation for aerospace applications.

39. (original) The method of claim 25 wherein said values correspond to a fluid flow calculation for an injection molding application.

40. (original) The method of claim 25 wherein defining includes establishing a stream communication connection between functional units and wherein only functional units needed to solve the calculations are instantiated.

41. (original) The method of claim 25 wherein said values correspond to a structures calculation for structural analysis.

42. (original) The method of claim 25 wherein said values correspond to a search algorithm for an image search.

43. (original) The method of claim 25 wherein said values correspond to a search algorithm for data mining.

44. (original) The method of claim 25 wherein said values correspond to a financial modeling application.

45. (original) The method of claim 25 wherein said values correspond to an encryption algorithm.

46. (original) The method of claim 25 wherein said values correspond to a genetic pattern matching function.

47. (original) The method of claim 25 wherein said values correspond to a protein folding function.

48. (original) The method of claim 25 wherein said values correspond to an organic structure interaction function.

49. (original) The method of claim 25 wherein said values correspond to a signal filtering application.

50. (original) The method of claim 25 wherein said reconfigurable computing system comprises at least one microprocessor.

51. (original) A method for data processing in a reconfigurable computing system, the reconfigurable computer system comprising at least one reconfigurable processor comprising a plurality of functional units, said method comprising:

transforming an algorithm into a calculation implemented by said reconfigurable computing system at the at least one reconfigurable processor and driven by data propagation wherein lines of code of said calculation are linked based on said data propagation and fashioned as subsets of said plurality of functional units within the at least one reconfigurable processor forming columns of said calculation;

performing said calculation at the at least one reconfigurable processor by said subsets of said plurality of functional units to produce computed data;

exchanging said computed data between a first column of said calculation and a next column in said calculation, wherein said exchanging is based on reconfigurable routing resources within the at least one reconfigurable processor and wherein execution of said subsets of said plurality of function units occurs concurrently and said computed data is seamlessly passed between said first column of said calculation and said second column of said calculation;

evaluating a rate of change in at least one variable for each of said columns in said calculation;

continuing said calculation when said variable does not change for a particular column of said calculation; and

restarting said calculation at said column of said calculation where said variable does change.

52. (original) The method of claim 51 wherein how many functional units comprise the subset and functional type of each functional unit in said subset is based on the calculation.

REMARKS/ARGUMENTS

Claims 1-52 were presented for examination and are pending in this application. In an Official Office Action dated January 12, 2009, claims 1-52 were rejected. The Applicant thanks the Examiner for his consideration and addresses the Examiner's comments concerning the claims pending in this application below.

Applicant herein amends claim 30 and respectfully traverses the Examiner's rejections. No claims are presently cancelled and no new claims are added. These changes are believed not to introduce new matter, and their entry is respectfully requested. The claims have been amended to expedite the prosecution and issuance of the application. In making this amendment, the Applicant has not and is not narrowing the scope of the protection to which the Applicant considers the claimed invention to be entitled and does not concede, directly or by implication, that the subject matter of such claims was in fact disclosed or taught by the cited prior art. Rather, the Applicant reserves the right to pursue such protection at a later point in time and merely seeks to pursue protection for the subject matter presented in this submission.

Based on the above amendment and the following remarks, Applicant respectfully requests that the Examiner reconsider all outstanding rejections and withdraw them.

35 U.S.C. §112 Rejection of Claims

Claim 30 was rejected under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which the Applicant regards as the invention. Specifically claim 30 refers to "said first systolic wall" which lacks proper antecedent basis. Claim 30 is herein amended deleting the reference to "systolic." The Applicant contends that

claim 30 now possesses proper antecedent basis for all claim elements and meets the requirements set forth in 35 U.S.C. § 112 second paragraph.

Double Patenting.

Claims 1-52 were rejected under the judicially created doctrine of obviousness-type double patenting over commonly owned U.S. Patent No. 7,225,324.

Although the claims as presented are believed to be distinct with respect to U.S. Patent 7,225,324, a Terminal Disclaimer is herein supplied together with the required fee to expedite the allowance of patentable subject matter.

Conclusion

In view of all of the above, the claims are now believed to be allowable and the case in condition for allowance which action is respectfully requested. Should the Examiner be of the opinion that a telephone conference would expedite the prosecution of this case, the Examiner is requested to contact Applicant's attorney at the telephone number listed below.

Other than the Terminal Disclaimer fee, no fee is believed due for this submittal. However, any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

13 April, 2009

Michael C. Martensen, No. 46901
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5910 Tel
(303) 899-7333 Fax

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

NOTICE OF ALLOWANCE AND FEE(S) DUE

25235 7590 06/30/2009

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

EXAMINER	
COLEMAN, ERIC	
ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 06/30/2009

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/733,064	04/09/2007	Jon M. Huppenthal	SRC015 CON	7527

TITLE OF INVENTION: MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	$1510	$300	$0	$1810	09/30/2009

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE DOES NOT REFLECT A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE IN THIS APPLICATION. IF AN ISSUE FEE HAS PREVIOUSLY BEEN PAID IN THIS APPLICATION (AS SHOWN ABOVE), THE RETURN OF PART B OF THIS FORM WILL BE CONSIDERED A REQUEST TO REAPPLY THE PREVIOUSLY PAID ISSUE FEE TOWARD THE ISSUE FEE NOW DUE.

HOW TO REPLY TO THIS NOTICE:

I. Review the SMALL ENTITY status shown above.

If the SMALL ENTITY is shown as YES, verify your current SMALL ENTITY status:

A. If the status is the same, pay the TOTAL FEE(S) DUE shown above.

B. If the status above is to be removed, check box 5b on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and twice the amount of the ISSUE FEE shown above, or

If the SMALL ENTITY is shown as NO:

A. Pay TOTAL FEE(S) DUE shown above, or

B. If applicant claimed SMALL ENTITY status before, or is now claiming SMALL ENTITY status, check box 5a on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and 1/2 the ISSUE FEE shown above.

II. PART B - FEE(S) TRANSMITTAL, or its equivalent, must be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed and an extra copy of the form should be submitted. If an equivalent of Part B is filed, a request to reapply a previously paid issue fee must be clearly made, and delays in processing may occur due to the difficulty in recognizing the paper as an equivalent of Part B.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Mail Stop ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

PTOL-85 (Rev. 08/07) Approved for use through 08/31/2010.

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), to: Mail Mail Stop ISSUE FEE
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450
or Fax (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

25235 7590 06/30/2009

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission

I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

(Depositor's name)

(Signature)

(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/733,064	04/09/2007	Jon M. Huppenthal	SRC015 CON	7527

TITLE OF INVENTION: MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	$1510	$300	$0	$1810	09/30/2009

EXAMINER	ART UNIT	CLASS-SUBCLASS
COLEMAN, ERIC	2183	712-226000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).

☐ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.

☐ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list

(1) the names of up to 3 registered patent attorneys or agents OR, alternatively,

(2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1 _____

2 _____

3 _____

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE

(B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent) : ☐ Individual ☐ Corporation or other private group entity ☐ Government

4a. The following fee(s) are submitted:

☐ Issue Fee

☐ Publication Fee (No small entity discount permitted)

☐ Advance Order - # of Copies _____

4b. Payment of Fee(s): **(Please first reapply any previously paid issue fee shown above)**

☐ A check is enclosed.

☐ Payment by credit card. Form PTO-2038 is attached.

☐ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment, to Deposit Account Number _____ (enclose an extra copy of this form).

5. **Change in Entity Status** (from status indicated above)

☐ a. Applicant claims SMALL ENTITY status. See 37 CFR 1.27.

☐ b. Applicant is no longer claiming SMALL ENTITY status. See 37 CFR 1.27(g)(2).

NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Authorized Signature _____ Date _____

Typed or printed name _____ Registration No. _____

This collection of information is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

Notice of Allowability	Application No.	Applicant(s)
	11/733,064	HUPPENTHAL ET AL.
	Examiner	Art Unit
	Eric Coleman	2183

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *terminal disclaimer filed 4/13/09*.

2. ☒ The allowed claim(s) is/are *1-52*.

3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____.
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☐ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☒ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date 4/17/09
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☒ Interview Summary (PTO-413), Paper No./Mail Date _____.
7. ☒ Examiner's Amendment/Comment
8. ☐ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

/Eric Coleman/
Primary Examiner, Art Unit 2183

EXAMINER'S AMENDMENT

An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

The following changes to the drawings have been approved by the examiner and agreed upon by applicant: Correction of the alignment of the text of literal legends in figure 1 is required. In order to avoid abandonment of the application, applicant must make these above agreed upon drawing changes.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Eric Coleman whose telephone number is (571) 272-4163. The examiner can normally be reached on Monday-Thursday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

EC

/Eric Coleman/
Primary Examiner, Art Unit 2183

Client/Matter No. 80404.0018.001
EFS-Web

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Serial No. 11/733,064 Application of: Jon M. Huppenthal and David E. Caliga Filed: April 9, 2007 Attorney Docket No. SRC015 CON For: MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS	Confirmation No.: 7527 Art Unit: 2183 Examiner: Coleman, Eric Customer No.: **25235**

RE-SUBMISSION OF DRAWING FIG. 1

MAIL STOP ISSUE FEE
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

The attached Fig. 1 drawing sheet should replace the previously filed Fig. 1as required by the Notice of Allowance. An annotated drawing is not included as no new matter has been added.

Attachment: Replacement Sheet

24 July 2009
Date

Michael C. Martensen, Reg. No. 46,901
HOGAN & HARTSON LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5910 Tel
(303) 899-7333 Fax

\\\CS - 080404/000018 - 116236 v1

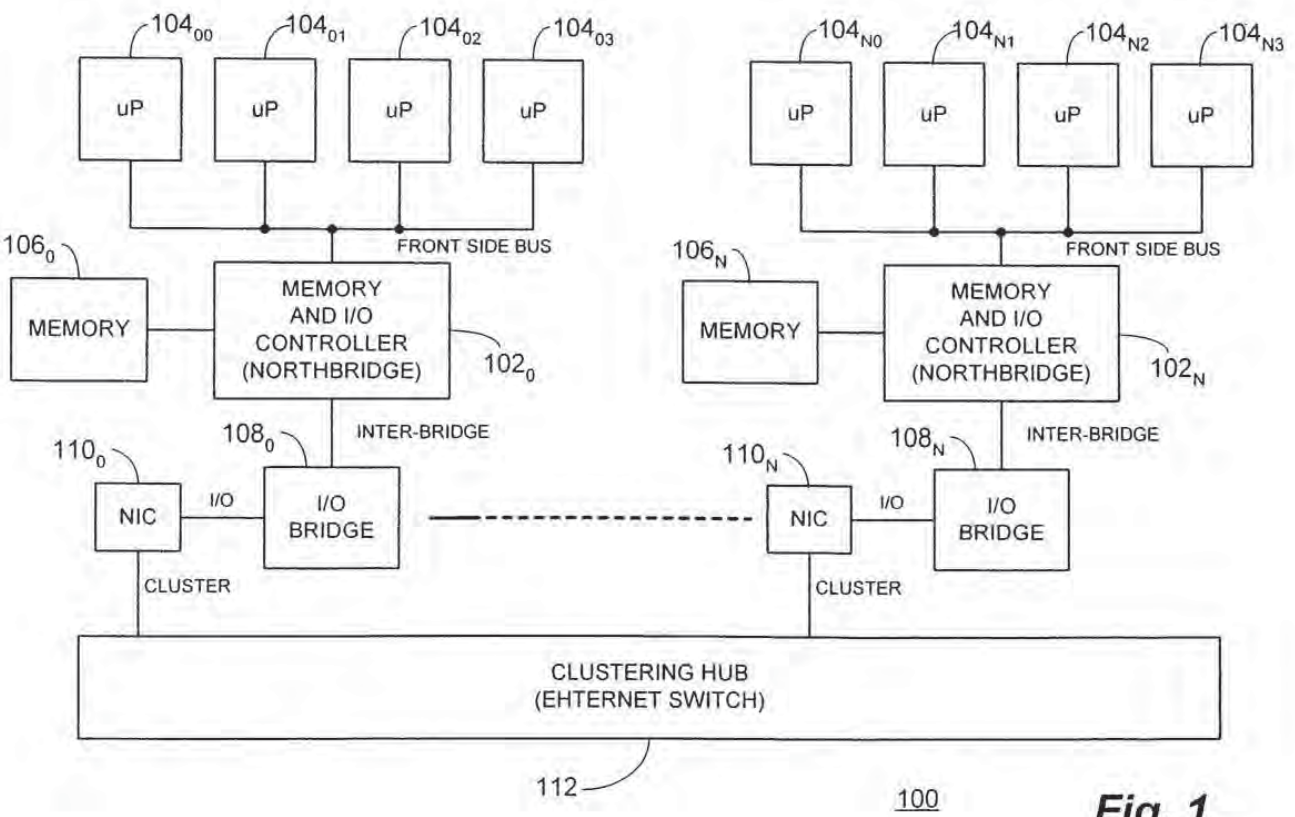


Fig. 1
Prior Art

S/N: 11/733,064
Docket No.: SRC015 CON
Title: MULTI-ADAPTIVE PROCESSING SYSTEMS AND TECHNIQUES FOR ENHANCING PARALLELISM AND PERFORMANCE OF COMPUTATIONAL FUNCTIONS
Inv: Jon M. Huppenthal and David E. Caliga
Replacement Sheet

SRC0001705

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	ISSUE DATE	PATENT NO.	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/733,064	11/17/2009	7620800	SRC015 CON	7527

25235 7590 10/28/2009

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

ISSUE NOTIFICATION

The projected patent number and issue date are specified above.

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment is 95 day(s). Any patent to issue from the above-identified application will include an indication of the adjustment on the front page.

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Application Assistance Unit (AAU) of the Office of Data Management (ODM) at (571)-272-4200.

APPLICANT(s) (Please see PAIR WEB site http://pair.uspto.gov for additional applicants):

Jon M. Huppenthal, Colorado Springs, CO;
David E. Caliga, Colorado Springs, CO;

IR103 (Rev. 10/09)

EXHIBIT P

Office Action Summary	Application No.	Applicant(s)	
	10/869,200	POZNANOVIC ET AL.	
	Examiner	Art Unit	
	Shane M Thomas	2186	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *16 June 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-24* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-24* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *16 June 2004* is/are: a)☐ accepted or b)☒ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

This Office action is responsive to the application filed 6/16/2004. Claims 1-24 are presented for examination.

The examiner requests, in response to this Office action, any reference(s) known to qualify as prior art under 35 U.S.C. sections 102 or 103 with respect to the invention as defined by the independent and dependent claims. That is, any prior art (including any products for sale) similar to the claimed invention that could reasonably be used in a 102 or 103 rejection. This **request** does not require applicant to perform a search. This request is not intended to interfere with or go beyond that **required** under 37 C.F.R. 1.56 or 1.105.

The request may be fulfilled by asking the attorney(s) of record handling prosecution and the inventor(s)/assignee for references qualifying as prior art. A simple statement that the query has been made and no prior art found is sufficient to fulfill the request. Otherwise, the fee and certification requirements of 37 CFR section 1.97 are waived for those documents submitted in reply to this request. This waiver extends only to those documents within the scope of this request that are included in the application's first complete communication responding to this requirement. Any supplemental replies subsequent to the first communication responding to this request and any information disclosures beyond the scope of this are subject to the fee and certification requirements of 37 CFR section 1.97.

In the event prior art documentation is submitted, a discussion of relevant passages, figs. etc. with respect to the claims is requested. The examiner is looking for specific references to 102/103 prior art that identify independent and dependent claim limitations. Since applicant is

most knowledgeable of the present invention and submitted art, his/her discussion of the reference(s) with respect to the instant claims is essential. **A response to this inquiry is greatly appreciated.**

The examiner also requests, in response to this Office action, that support be shown for language added to any original claims on amendment and any new claims. That is, indicate support for newly added claim language by specifically pointing to page(s) and line no(s). in the specification and/or drawing figure(s). This will assist the examiner in prosecuting the application.

Drawings

The element --computation logic 201-- of paragraph 53 should be corrected to 200 as per figure 2.

Claim Objections

Claims 1-23 are objected to because of the following informalities:

As per claim 1, the term -- the memory-- of line 3 should be amended to read --the *first* memory-- since --*the* memory-- has not been previously defined. Appropriate correction is required.

As per claim 2, the term --the processor-- should be amended to --the *reconfigurable* processor since the term --*the* processor-- has not been previously defined in the claims.

As per claim 5, line 3, the term --memory-- has been mistakenly duplicated.

As per claim 8, the term --prefetch unit-- should be amended to --*data* prefetch unit-- since the term --prefetch unit-- has not been previously defined in the claims.

As per claim 11, the term --the unit-- should be amended to --the data prefetch unit-- since the term --the unit-- has not been previously defined in the claim.

As per claim 15, the term --at least of the-- of line 2 should be corrected to read --at least *one* of--.

As per claim 17, the term --the data access unit-- of lines 4-5 should be amended to --*a* data access unit-- since the term --*the* data access unit-- has not been previously defined in the claim.

Claims 3,4,6,7,9,10, 12-14,16, and 18-23, are objected to as being dependent on objected claims.

Claim Rejections - 35 USC § 112

The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

Claims 1-10, 13, and 14, are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

As per claim 1, the terms --first characteristic type-- and --second characteristic type-- are not clearly defined in the Applicant's specification. Applicant is reminded of **37 C.F.R. 1.75 (d)(1)** which states that the claim or claims must conform to the invention as set forth in the remainder of the specification and the terms and phrases used in the claims must find clear support or antecedent basis in the description so that the meaning of the terms in the claims may be ascertainable by reference to the description. (See 1.58(a).) The phrases --first characteristic type-- and --second characteristic type-- are not terms of art; nonetheless, for the purposes of examination, the Examiner shall regard the terms as meaning any type of memory (e.g. a SRAM, Flash Rom, DRAM, hard disk, etc.).

As per claims 2 and 13, the Applicant's disclosure does not explicitly mention that the reconfigurable processors cannot have a cache. The disclosure mentions in the Background section, and specifically in paragraphs 16-17, the drawbacks of having a hard-wired cache in a system; however, the Detailed Description does not explicitly state that the reconfigurable processor as taught by the Applicant *cannot* contain a cache. It appears to the Examiner that no specific (hard-wired) cache memory is included in the reconfigurable processor as taught in the disclosure; rather an on-board memory and user-logic can be configured based on a program (paragraph 52). Therefore, for the purposes of examination, the Examiner shall interpret the claim such that the reconfigurable processor of claim 1 does not contain a *hard-wired* (specific) cache.

As per claims 3 and 14, it follows from the rejection for claims 2 and 13, that since Applicant's disclosure does not explicitly state that a reconfigurable processor *cannot* have a cache, the disclose further does not explicitly teach that the reconfigurable processor cannot have

a cache line-sized unit of contiguous data. For the purposes of examination and based on the discussion of claim 2 above, the Examiner shall interpret the limitation of claim 3 such that the reconfigurable processor of claim 1 does not have a *hard-wired* (specific) cache line-sized unit of contiguous data being retrieved from the [second] memory.

As per claim 4, it is not clear to which memory the term –the memory—refers as –the memory lacks antecedent basis--. For the purposes of examination, the Examiner shall interpret the term –the memory—to indicate the –second memory—of claim 1.

As per claim 7, the term --disassembled-- is not known to be a term of art, and further, not specifically defined in the Applicant's specification. Nonetheless, for the purposes of examination, the Examiner shall regard the term --disassembled-- with the broadest reasonable interpretation. Refer to **37 C.F.R. 1.75 (d)(1)**.

Claims 5, 6, and 8-10, are rejected as being dependent on rejected base claim 1.

The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Claims 2-4,8-10, and 15-23 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

As per claim 2, it is not clear which memory (first or second memory) the term --the memory-- is referring to since --the memory-- lacks antecedent basis. The Examiner recommends amending the term --the memory-- to overcome this rejection. Nonetheless, for the purposes of examination, the Examiner shall interpret the claim as --the first memory--.

As per claim 3, it is not clear which memory (first or second memory) the term --the memory-- is referring to since --the memory-- lacks antecedent basis. The Examiner recommends amending the term --the memory-- to overcome this rejection. Nonetheless, for the purposes of examination, the Examiner shall interpret the claim as --the second memory--.

As per claim 4, it is not clear which memory (first or second memory) the term --the memory-- is referring to since --the memory-- lacks antecedent basis. The Examiner recommends amending the term --the memory-- to overcome this rejection. Nonetheless for the purposes of examination, the Examiner shall interpret that claim as --the second memory--.

As per claim 8, it is not clear whether the processor memory is the same as the second memory or if the processor memory is a separate (third) memory since the data prefetch unit is claimed as retrieving data from both a second memory and a processor memory. The Examiner shall interpret the second memory as being a processor memory.

As per claims 15 and 17, it is not clear if the term --the data access unit-- is referring to --the data prefetch unit-- or is a new entity being defined by the claim since the term --*the* data access unit-- lacks antecedent basis. Nonetheless, for the purposes of examination, the Examiner shall regard the term --the data access unit-- to be a separate entity based in part from the Applicant descriptions of the drawings on page 8 showing that the data prefetch unit and data access unit are distinct entities.

As per claim 19, it is not clear whether the term --a data access unit-- is the same data access unit that has been defined in claim 17 or the --a data access unit-- is a different data access unit that performs the limitation of claim 19 and does not perform the limitation of the data

access unit of claim 17. For the purposes of examination, the Examiner shall interpret the --a data access unit-- as --*the* data access unit-- [of claim 17].

As per claims 9-10 and 16-23, the claims are rejected as being dependent on rejected claims.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Claims 1-24 are rejected under 35 U.S.C. 102(e) as being anticipated by Paulraj (U.S. Patent Application Publication No. 2003/0084244).

As per claim 1, Paulraj shows a reconfigurable processor in figure 6 and a first memory (L1) having a first characteristc memory type (line size, blocking factor, associativity, etc.) and a second memory (L2) having a second characteristic memory type (line size, blocking factor, associativity, etc.). Refer to paragraph 23. Paulraj further teaches a functional unit 102 that executes applications using the memories L1 and L2 (paragraph 9). As is known in the art, a cache memory controller is often used to access and move data between a memory hierarchy. The Examiner is considering a data prefetch unit to be the logic assocatied with the moving, and only the moving, of data between the first and second memories (L1 and L2) since Paulraj shows

a connection between the levels of cache in figure 6. This logic as well as the first and second memory types (L1 and L2) are configued by a program – refer to paragraphs 23-24. The data prefetch unit as defined by the Examiner must be configued as well by the program when moving data since the cache line size and blocking factor can change, so different amounts of data can be exchanged for the same access when different programs run.

As per claims 2 and 13, as taught in paragraphs 23 and 29 of Paulraj, no specific cache is present in the system of Paulraj. Rather, an FPGA is utilized as representing a caching hierarchy and is optimized based on the memory needs of a specific program running on the reconfigurable processor.

As per claims 3 and 14, Paulraj teaches in paragraph 23 that a specific cache line size of contiguous data is not retrieved since the data line size is optimized based on the memory needs of the program when executing on the reconfigurable processor. Refer also to paragraph 29.

As per claim 4, Paulraj teaches that a load/store unit is used to access the caches (L1-L3) in order to determine if cache data is present in the cache hierarchy (paragraph 6). Since the functional unit 102 (figure 6) is responsible for accessing the programmable memory unit 104, the Examiner is therefore considering the load/store unit logic of the programmable memory unit that is responsible for for accessing the L1 and L2 caches (first and second memory types) to be a memory controller. It can be seen that the memory controller, as defined by the Examiner, controls the transfer of data between the memory (assuming second memory L2) and the data prefetch unit, since the memory controller (load/store unit logic) is responsible for retrieving the data from the cache if a hit occurs (paragraph 4).

As per claim 5, as taught in paragraph 1, an external memory (element 18, figure 1) is generaly coupled to a microprocessor and holds data to be used by the microcontroller during program execution. The Examiner is considering the process of writing data back to the external memory from the FPGA memory 104 containing the caches (on-board memory), such as during a write-back scheme as known in the art, to be performed by the data prefetch unit portion of the functional logic as defined above by the Examiner. The data prefetch logic, as defined above, is responsible for all of the transfer of data into, out of, and between the FPGA memory 104.

As per claim 6, the Examiner is regarding a --register-- in its broadest reasonable sense and it thus considering it be to be a unit of logic. Therefore, the portion of the function logic that is responsible for the movement of data (as defined above to be the data prefetch unit) is being considered by the Examiner as containing a --register-- portion of the reconfigurable processor since, for instance, the blocking factor and line size of the programmable memory 112 can change, a --register-- or portion of the reconfigurable processor must be set in order to indicate the currnet line size and blocking factor when a given application is being run on the reconfigurable processor at a given point in time. Refer to paragraph 23.

As per claim 7, the Examiner is considering the process of --disassembling the data prefetch unit-- as modifying the data prefetch unit logic of the fucntion logic 102 every time the program being executed by the reconfigurable processor changes. It can be seen that the data prefetch unit changes during these intervals since the cache line size, blocking factor, and associativity of the FPGA changes when optimal for the next program to be executed (refer to paragraph 23). Thus it can be seen that the data prefetch unit logic is --disassembled-- when another program is executed by the reconfigurable processor of Paulraj.

As per claim 8, as can be seen that the FPGA memory 112, that comprises the first and second memories (L1 and L2) and which is accessed by the data prefetch unit of the functional unit 102 as discussed above, is a --processor memory-- (part of cpu 110). Therefore, since the data pretech unit can access the L2 cache as discussed above in the rejection of claim 1, the data prefetch unit can retrive data from the L2 portion of --processor memory--112.

As per claim 9, as shown in figure 1 and taught in paragraph 1 of Paulraj, the system 10 is actually a microprocessor, which contains a memory controller 14. The main difference between the prior art of figure 1 and the invention of Paulraj in figure 6 is that the memroy hierarchy is configurable and accessed by a fucntional unit in lieu of a separate memory controller logic (paragraph 9). Therefore, since the memory controller logic for accessing the cache hierarchy is still contained within cpu 110 of figure 6, it can be seen that the cpu 110 is actually a microprocessor. It follows that the --processor memory-- 112 is therefore a --microprocessor memory--.

As per claim 10, since the cpu 110 of figure 6 is a reconfigurable processer (able to reconfigure its memory heirarchy to match the needs of the application it is currently running), it can be seen that the cpu memory 112 is a reconfigurable processor memory.

As per claim 11, Paulraj depicts a reconfigurable hardware system in figure 6. Paulraj further teaches in paragraph 26 that when a particular application is to be run by the reconfigrable processor 110, a configuration vector is retrieved to program the programmable memory 112 (figure 6). As shown in figure 6, the step of accesing the configuration vector is executed outside of the reconfigurable processor 110. Therefore, the Examiner is considering the memory that contains the configuration vectors to be a--common memory-- and a data

prefetch unit (reconfiguration unit 106 executing on the reconfigurable processor 110) accessing the common memory in order to determine how to program the memory 112 (paragraph 29). The data prefetch unit 106 is --configured-- by an application to be excuted on the sysem 110 since when a new application is to be executed, the data prefetch unit is called upon (or configured) to access the configuration vector for the particular application.

As per claim 12, the Examiner is considering a --memory controller-- to be the system portion utilized when creating a new configuration vector for an application. Such a process occurs in figure 5 and taught in paragraghs 23-25 of Paulraj. When a new configuration vector is created by analizing performance information that has been collected for the application. The Examiner is thereby considering the --memory controller-- to be the element of the reconfigurable hardware system that is associated with storing the new configuration vector into the common memory so that the vector can be accessed later when the same application is run again.

As per claim 15, the Examiner is considering the reconfiguration module 106 of the reconfigurable processsor 110, as comprising two distinct elements: a --computational unit-- and a --data access unit--. The data access unit is the element that is responsible for accessing the configuration vector as taught in paragraph 29 of Paulraj; or in other words, the Examiner is considering the --data access unit-- to be the same as the --memory controler-- defined in the rejection of claim 12. The Examiner is further considering the --computational unit-- of the rconfiguration module 106 to be the element that sets up the programmable memory module 104 using the configuration vector that was accessed by the --data access unit-- (paragraph 29).

As per claim 16, as taught by Paulraj in paragraph 29, the --data access unit-- supplies the configuration vector to the --computational unit-- in order to set up the programmable memory 104 as required by the application to be run on the reconfurable processor 110.

As per claim 17, the Examiner is considering a --data prefetch unit-- to be the reconfiguration unit 106 of reconfigurable processor 110 (figure 6). As taught in paragraph 26 and 29 of Paulraj, the --data prefetch unit-- accesses a memory in order to determine if a configuration vector is known for a given application, and if so, the vector is retrieved (from the memory). If this --data-- (configuration vector) is not known then a simulation is performed with the application in order to collect performance information. The Examiner is considering the element that executes and collects the performance data as being a --computational unit-- and the element of Paulraj that stores the configuration vector, once determined, to be a --data access unit-- since it stores the vector into the --memory-- from which it can be later retrieved (step 212 of figure 5). The --computational unit--, --data access unit--, and the --data prefetch unit-- are all --configured-- by a program (application) since (1) a new application configures the computational unit portion of the reconfiguration unit to perform a simulation in order to determine the optimal memory hierarchy organization; (2) the new application configures the --data access unit-- to store and retrieve (step 212) the configuration vector for that particular application; and (3) the --data prefetch unit-- is configured by the application to determine if a configuration file exists for the application and if so, the data prefetch unit is configured by the program the programmable memory 112 in order to optimize the programmable memory for that particular application.

As per claim 18, the --data-- (configuration vector) is transferred from the

--computational unit-- to the --data access unit-- when the configuration unit has created a configuration vector (step 208 of figure 5). The --data-- is written to the memory --from-- the --data prefetch unit-- since the data prefetch unit (reconfiguration unit 106) is the element that executed the beginning of the configuration vector creation process (step 200 of figure 5). Refer to paragraph 26. Thus the Examiner is considering the data as being written --from-- the data prefetch unit.

As per claim 19, as taught in paragraph 26, if the configuration vector is known, the vector is retrieved from the memory to the data prefetch unit (reconfiguration unit 106). The data is read directly from the data prefetch unit when a request to create a configuration vector is made for a new application as shown in figure 6 since the data prefetch unit is responsible for being the vector creation process. The data is directed from the data prefetch unit (reconfigure logic) to be read from the memory by the data access unit to the computational unit where it is processed to produce a configuration vector.

As per claim 20, as stated above, the configuration vector (--data--) is created by the computational unit via acquired simulation data. The configuration vector is the resultant product that is transferred from the memory to the data prefect unit when it is determined that the configuration vector for the application is available (paragraph 26). Thus --all-- of the data that is transferred is processed by the computational unit (albeit before the transfer occurs) since the data prefetch unit required the entire configuration vector in order to set up the programmable memory 112.

As per claim 21, Paulraj shows in paragraph 26 that an explicit request for the configuration vector for the current application results in the data (if it exists) selected for the optimal configuration of the programmable memory 112 for that application.

As per claim 22, the Examiner is not considering the data (configuration vector) to be the size of a complete cache line since the data is used to create a cache hierarchy. In other words, the caches (L1-L3) of the programmable memory 112 are not programmed when the data is transferred from the memory to the data prefetch unit; therefore, the data cannot be a complete cache line.

As per claim 23, since the Examiner defined the portion of the reconfiguration unit that accesses the configuration file (data) from the memory, the Examiner is defining the logic that controls the actual transfer of that data to the data prefetch unit (portion of the reconfiguration unit that executes the fetch of the configuration vector and then programs the programmable memory 112) to be a --memory controller--. Thus the data access unit determines whether a configuration vector exists for an application and if so, the memory controller sends that data to the data prefetch unit.

As per claim 24, The Examiner is considering the element that executes and collects the performance data as being a --computational unit-- and the element of Paulraj that stores and retrieves the configuration vector, once determined, to be a --data access unit-- since it stores the vector into the --memory-- from which it can be later retrieved (step 212 of figure 5). The --computational unit-- and --data access unit -- are --configured-- by a program (application) since (1) a new application causes in the configuration of the computational unit portion of the reconfiguration unit to perform a simulation in order to determine the optimal memory hierarchy

organization for the application and (2) the new application causes the configuration of the --data access unit-- to store and retrieve (step 212) the configuration vector for that particular application. Refer to paragraphs 25-27.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Poznanovic (U.S. Patent Application Publication No. 2003/0046530) teaches a reconfigurable processor (figure 2) which can be reprogrammed based on a program.

Vondran (U.S. Patent No. 6,243,791) illustrates an example of the operation of a cache controller in a cache hierarchy (column 1, lines 54-67).

Otterness (U.S. Patent No. 6,460,122) further teaches common operation of a cache controller in column 21, lines 1-16.

Darling (U.S. Patent No. 6,714,041) teaches a reconfigurable system (figure 5) that is able to be reprogrammed based on a program.

Burton (U.S. Patent Application Publication No. 2003/0088737) teaches uncached device operations in a reconfigurable processor system.

Gschwind et al. (U.S. Patent Application Publication No. 2003/0046492) teaches a reconfigurable memory array which can be operated as a cache or a non-cache memory.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Shane M Thomas whose telephone number is (703) 605-0725.

Please note: the aforementioned number will change to (571) 272-4188 effective October 19, 2004. The examiner can normally be reached M-F 8:30 - 5:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matt M Kim can be reached on (703) 305-3821, which will change to (571) 272-4182 effective October 19, 2004. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Shane M. Thomas

MATTHEW ANDERSON
PRIMARY EXAMINER
GROUP 2100

Client Matter No. 80404.0033.001
Express Mail No.: EV330612115US

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Serial No. 10/869,200 Application of: POZNANOVIC Filed: June 16, 2004 Art Unit: 2186 Examiner: THOMAS, Shane M Attorney Docket No. SRC028 For: SYSTEM AND METHOD OF ENHANCING EFFICIENCY AND UTILIZATION OF MEMORY BANDWIDTH IN RECONFIGURABLE HARDWARE	Confirmation No.: 5929 Customer No.: **25235**

AMENDMENT AND RESPONSE PURSUANT TO OFFICE ACTION DATED JANUARY 14, 2005

MAIL STOP AMENDMENT
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the office communication mailed January 14, 2005 please amend the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims which begins on page 3 of this paper.

Amendments to the Drawings begin on page 7 of this paper and include both an attached replacement sheet and an annotated sheet showing changes.

Remarks/Arguments begin on page 8 of this paper.

\\\BO - 80404/0033 - 175820 v1

An **Appendix** including 1 sheet of amended drawing figures is attached following page 8 of this paper.

A. Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently Amended) A reconfigurable processor comprising:

a first memory having a first characteristic memory bandwidth and/or memory utilization ~~type~~; and

a data prefetch unit coupled to the first memory, wherein the data prefetch unit retrieves data from a second memory of second characteristic memory bandwidth and/or memory utilization and place the retrieved data in the first memory ~~type~~ and wherein at least the first ~~the~~ memory ~~types~~ and data prefetch unit are configured by a program.

2. (Currently Amended) The reconfigurable processor of claim 1, wherein the reconfigurable processor does not have a cache to store data from the first memory.

3. (Currently Amended) The reconfigurable processor of claim 1, wherein the second memory has a characteristic line size and the data retrieved from the second memory is not a ~~cache~~ line-sized unit of contiguous data.

4. (Currently Amended) The reconfigurable processor of claim 1, wherein the data prefetch unit is coupled to a memory controller that controls the transfer of the data between the second memory and the data prefetch unit.

5. (Currently Amended) The reconfigurable processor of claim 1, wherein the data prefetch unit receives processed data from on-processor memory and writes the processed data to an external off-processor ~~memory~~ memory.

6. (Original) The reconfigurable processor of claim 1, wherein the data prefetch unit comprises at least one register from the reconfigurable processor.

7. (Original) The reconfigurable processor of claim 1, wherein the data prefetch unit is disassembled when another program is executed on the reconfigurable processor.

8. (Currently Amended) The reconfigurable processor of claim 1 wherein said second memory comprises a processor memory and said data prefetch unit is operative to retrieve data from [[a]] the processor memory.

9. (Original) The reconfigurable processor of claim 8 wherein said processor memory is a microprocessor memory.

10. (Original) The reconfigurable processor of claim 8 wherein said processor memory is a reconfigurable processor memory.

11. (Currently Amended) A reconfigurable hardware system, comprising:

a common memory; and

one or more reconfigurable processors coupled to the common memory, wherein at least one of the reconfigurable processors includes a data prefetch unit to read and write data between the data prefetch unit and the common memory, and wherein the data prefetch unit is configured by a program executed on the system.

12. (Original) The reconfigurable hardware system of claim 11, comprising a memory controller coupled to the common memory and the data prefetch unit.

13. (Currently Amended) The reconfigurable hardware system of claim 11, wherein the one or more reconfigurable processors are [[is]] not coupled to a cache.

14. (Currently Amended) The reconfigurable hardware system of claim 11, wherein the common memory has a characteristic line size and the data written and read between the data prefetch unit and the common memory is not a ~~cache~~ line-sized unit of contiguous data.

15. (Currently Amended) The reconfigurable hardware system of claim 11, wherein the at least one of the reconfigurable processors also includes a computational unit coupled to ~~the~~ a data access unit.

16. (Original) The reconfigurable hardware system of claim 15, wherein the computational unit is supplied the data by the data access unit.

17. (Currently Amended) A method of transferring data comprising:

transferring data between a memory and a data prefetch unit in a reconfigurable processor; and

transferring the data between a computational unit and ~~the~~ a data access unit, wherein the computational unit and the data access unit, and the data prefetch unit are configured by a program.

18. (Original) The method of claim 17, wherein the data is written to the memory, said method comprising:

transferring the data from the computational unit to the data access unit; and

writing the data to the memory from the data prefetch unit.

19. (Currently Amended) The method of claim 17, wherein the data is read from the memory, said method comprising:

transferring the data from the memory to the data prefetch unit; and

reading the data directly from the data prefetch unit to the computational unit through [[a]] the data access unit.

20. (Original) The method of claim 19, wherein all the data transferred from the memory to the data prefetch unit is processed by the computational unit.

21. (Original) The method of claim 19, wherein the data is selected by the data prefetch unit based on an explicit request from the computational unit.

22. (Original) The method of claim 17, wherein the data transferred between the memory and the data prefetch unit is not a complete cache line.

23. (Original) The method of claim 17, wherein a memory controller coupled to the memory and the data prefetch unit, controls the transfer of the data between the memory and the data prefetch unit.

24. (Original) A reconfigurable processor comprising:

a computational unit; and

a data access unit coupled to the computational unit, wherein the data access unit retrieves data from memory and supplies the data to the computational unit, and wherein the computational unit and the data access unit are configured by a program.

REMARKS/ARGUMENTS

Claims 1-24 remain in the application. Claims 1, 2, 5, 8, 11, 15 and 17 are amended to address informalities noted in the Office action. No new matter is added by these amendments.

A. Drawings.

The correction made to Fig. 2 is believed to overcome the objection to the drawings.

B. Claim Objections

Claims 1, 2, 5, 8, 11, 15 and 17 are amended to overcome the objections stated in the office action. It is respectfully requested that the objections to claims 1-23 be withdrawn.

C. Rejections under 35 U.S.C. 112.

Claims 1-10, 13 and 14 were rejected under 35 U.S.C. 112. This rejection is respectfully traversed.

Specifically, the Office action questions the reference to a first characteristic memory type and a second characteristic memory type in claim 1. This is illustrated, for example, in Fig. 3 in which a logic block 300 moves data from a first memory 305 having a first characteristic memory type to a second memory 307 having a second characteristic memory type. As set out in the paragraphs [0007]-[0016] of the specification, for example, the memory characteristics may include one or more of the following characteristics: line size, associativity, replacement policy, write policy, and cache size, all of which provide varying memory bandwidth efficiency and/or memory bandwidth utilization. The amendment to claim 1 is believed to clarify this feature of the invention and overcome the objections raised in the Office action.

With respect to claims 2 and 13, the examiner's interpretation that claims 2 and 13 do not require a hard-wired cache is accurate. It is noted that these limitations appear in claims 2 and 13, not claim 1.

The amendments to claims 3, 4 and 14 are believed to clarify the questions raised in the Office action.

Claims 2-4, 8-10 and 15-23 were rejected under 35 U.S.C. 112 as indefinite. The amendments to claims 2, 3, 4, 8, 15 and 17 are believed to overcome the rejections.

D. Rejections under 35 U.S.C. 102.

Claims 1-24 were rejected under 35 U.S.C. 102 based upon Paulraj. This rejection is respectfully traversed.

Independent claim 1 calls for a reconfigurable processor. As set out in Applicant's specification at paragraph [0039], a reconfigurable processor is a computing device that instantiates an algorithm as hardware. Although the reference show a reconfigurable cache, Paulraj does not show or suggest a reconfigurable processor that instantiates an algorithm as hardware. Moreover, nothing in Paulraj would suggest the rather significant changes required to replace the CPU with a reconfigurable processor. For at least these reasons claim 1 is not anticipated nor made obvious by Paulraj.

Claims 2-10 that depend from claim 1 are allowable over Paulraj for at least the same reasons as claim 1 as well as the limitations that are presented in those claims.

Claim 11 calls for a reconfigurable hardware system comprising one or more reconfigurable processors. As noted above with respect to claim 1, Paulraj does not show or suggest even one reconfigurable processor. For at least these reasons claim 11 and claims 12-16 that depend from claim 11 are believed to be allowable over Paulraj.

Independent claim 17 calls for, among other things, transferring data between a memory and a data prefetch unit in a reconfigurable processor. As noted above, Paulraj does not show or suggest a reconfigurable processor, nor transferring data between a memory and a data prefetch unit in a reconfigurable

processor. For at least these reasons claim 17 and claims 18-23 that depend from claim 17 are allowable over Paulraj.

Claim 24 calls for a reconfigurable processor having a computational unit and a data access unit that are configured by a program. Paulraj does not show a reconfigurable processor. Moreover, the element of Paulraj that stores and retrieves the configuration vector is not configurable by a program. Similarly, the element that executes and collects performance data is not configurable by a program. Paulraj does not suggest making these elements configurable.

E. Conclusion.

The references that were cited but not relied upon are no more relevant than the references that were relied upon. In view of all of the above, the claims are now believed to be allowable and the case in condition for allowance which action is respectfully requested. Should the Examiner be of the opinion that a telephone conference would expedite the prosecution of this case, the Examiner is requested to contact Applicants' attorney at the telephone number listed below.

Any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

April 11, 2005

Stuart T. Langley, Reg. No. 33,940
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(720) 406-5335 Tel
(303) 899-7333 Fax

B. Amendments to the Drawings:

The attached sheet of drawings includes changes to Fig. 2. This sheet which includes Figs. 1-2 replaces the original sheet including Fig. 1-2. In Figure 2, element 201 is correctly identified.

Attachment: Replacement Sheet
Annotated Sheet Showing Changes

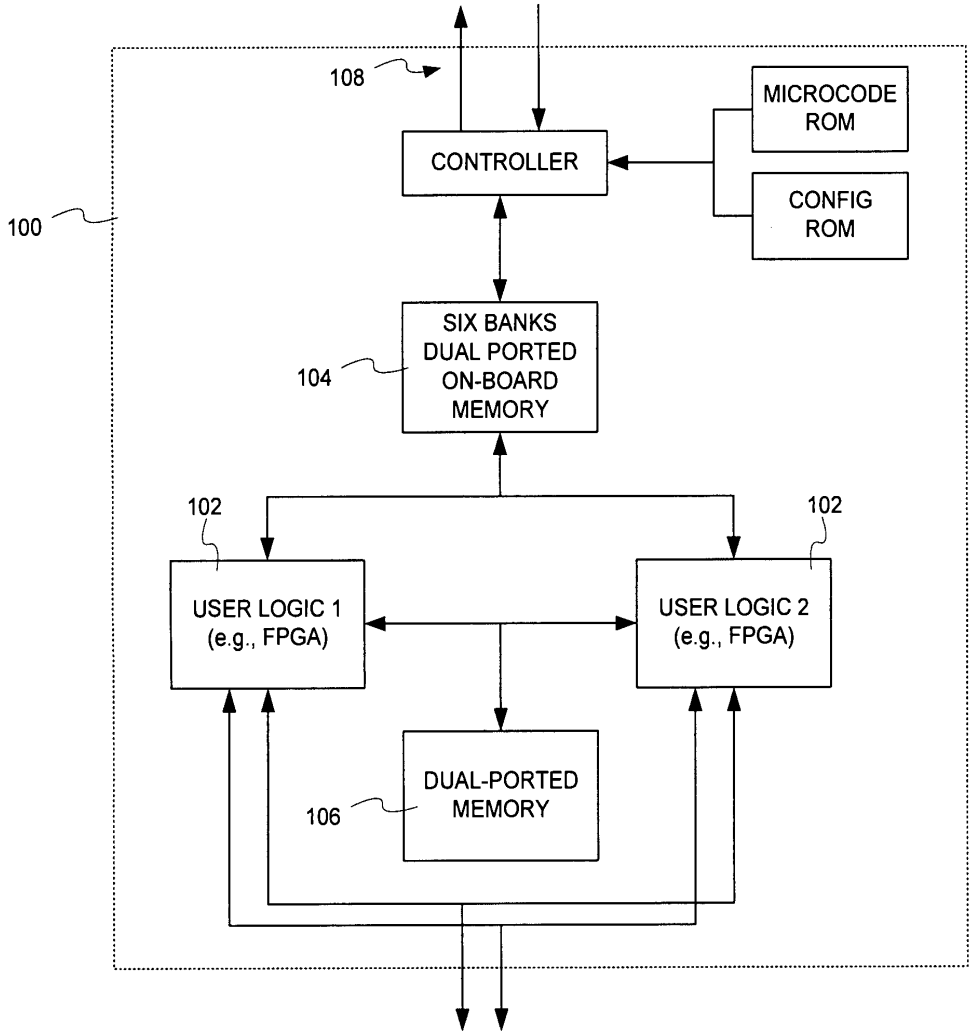


FIG. 1

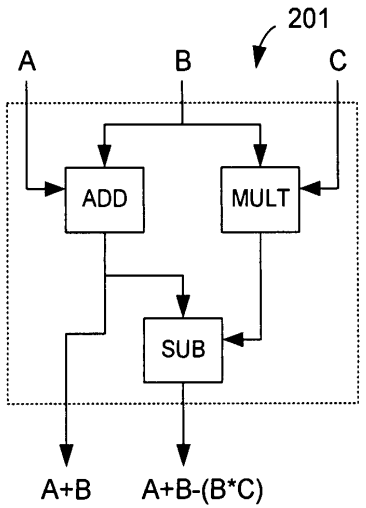


FIG. 2

REPLACEMENT SHEET

1/12

SRC0001016

Office Action Summary	Application No.	Applicant(s)	
	10/869,200	POZNANOVIC ET AL.	
	Examiner	Art Unit	
	Shane M. Thomas	2186	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *11 April 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-24* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-24* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *11 April 2005* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *4/11/05 & 6/6/05*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

This Office action is responsive to the amendment filed 4/11/2005.

All previously outstanding objections and rejections to the Applicant's disclosure and claims not contained in this Action have been respectfully withdrawn by the Examiner hereto.

Information Disclosure Statement

The information disclosure statement (IDS) submitted on 4/11/2005 has NOT been considered by the Examiner as the Application Number field on the Form 1449 reflects application number 10/809,200.

The information disclosure statement (IDS) submitted on 6/6/2005 was filed after the mailing date of the non-final Office action on 1/14/2005. The submission is in compliance with the provisions of 37 CFR 1.97. Accordingly, the information disclosure statement is being considered by the examiner.

Response to Amendment

The rejections of claims 1,3,8, and 14 have been modified to reflect the amendments to the respective claims.

Response to Arguments

Applicant's arguments filed 4/11/2005 have been fully considered but they are not persuasive.

Claims 2,3,13, and 14 remain rejected under 35 U.S.C. 112, first paragraph. While the Applicant's response on page 8 has verified the Examiner's assumption regarding the claim limitations of claims 2,3,13, and 14, no correction or amendment has been executed by the Applicant to overcome the rejection. The Applicant's specification does not disclose that a reconfigurable processor cannot have a cache nor a cache line-sized unit of contiguous data. As such the Examiner has maintained the rejections.

As per the Applicant's arguments regarding claim 1, the Applicant states on page 9 of the Response that Paulraj shows a reconfigurable cache but not a reconfigurable processor. The Examiner disagrees. While the caching system 112 (figure 6) of Paulraj is configurable (step 214, figure 5), it is also shown as being an element of CPU 110. Therefore since, the cache 112 is reconfigurable, it is justified that the processor 110, itself, is also reconfigurable as the reconfiguration of the FPGA module 112 occurs *within* the processor. As such, the CPU 110 can be construed as a --reconfigurable-- processor.

As per the Applicant's arguments regarding claim 11, the Examiner has shown in above in the discussion of claim 1 that Paulraj teaches a reconfigurable processor as claimed by the Applicant.

As per the Applicant's arguments regarding claim 17, the Examiner has shown above in the discussion of claim 1 that Paulraj teaches a reconfigurable processor as claimed by the Applicant. Further, the data prefetch unit, as defined in the rejection by the Examiner, is the

portion of the reconfiguration unit that accesses the memory; the memory stores a vector corresponding to an optimal configuration for a particular application program (¶26). Data is transferred between the memory and the data prefetch unit in a reconfigurable processor since the reconfiguration unit 106 can be part of a reconfigurable processor 100 as shown in figure 4 (¶22).

As per the Applicant's arguments regarding claim 24, the Examiner has modified the rejection to better explain how the prior art reference of Paulraj teaches the limitations of claim 24.

Claim Rejections - 35 USC § 112

The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

Claims 2,3,13, and 14 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claims contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

As per claims 2 and 13, the Applicant's disclosure does not explicitly mention that the reconfigurable processors cannot have a cache. The disclosure mentions in the Background section, and specifically in paragraphs 16-17, the drawbacks of having a hard-wired cache in a system; however, the Detailed Description does not explicitly state that the reconfigurable

processor as taught by the Applicant *cannot* contain a cache. It appears to the Examiner that no specific (hard-wired) cache memory is included in the reconfigurable processor as taught in the disclosure; rather an on-board memory and user-logic can be configured based on a program (paragraph 52). Therefore, for the purposes of examination, the Examiner shall interpret the claim such that the reconfigurable processor of claim 1 does not contain a *hard-wired* (specific) cache.

As per claims 3 and 14, it follows from the rejection for claims 2 and 13, that since Applicant's disclosure does not explicitly state that a reconfigurable processor *cannot* have a cache, the disclose further does not explicitly teach that the reconfigurable processor cannot have a cache line-sized unit of contiguous data. For the purposes of examination and based on the discussion of claim 2 above, the Examiner shall interpret the limitation of claim 3 such that the reconfigurable processor of claim 1 does not have a *hard-wired* (specific) cache line-sized unit of contiguous data being retrieved from the second memory.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Claims 1-24 are rejected under 35 U.S.C. 102(e) as being anticipated by Paulraj (U.S. Patent Application Publication No. 2003/0084244).

As per claim 1, Paulraj shows a reconfigurable processor in figure 6 and a first memory (L1) having a first characteristc memory utilization and a second memory (L2) having a second characteristic memory utilization. It is well known in the art that L1 caches have a higher utilziation rate than a lower-level cache such as L2. Paulraj teaches in ¶1 that upon a command from a processor, a search for the requested data is begines with the highest level cache (L1) and [if a miss occurs] continues next to the next level cache (L2). Thus it is inherent that the memory utilziation characteristc of the L1 cache of the reconfigurable processor 110 in figure 6 is greater than the memory utilziation characteristic of the L2 cache (and likewise for the L3 cache) as the L2 cache would only be utilzied when a miss to the L1 cache occurred. In other words, the reconfigurable processor *always* utilizes the L1 cache for a memory access and the *only* utilzies the L2 cache for requested data when the data is not in the L1 cache. Therefore, the cache utilziation characteristics of the --first memory-- and the --second memory-- are different.

Paulraj further teaches a functional unit 102 that executes applications using the memories L1 and L2 (paragraph 9). As is known in the art, a cache memory controller is often used to access and move data between a memory hierarchy. The Examiner is considering a data prefetch unit to be the logic assocatied with the moving, and only the moving, of data between the first and second memories (L1 and L2) since Paulraj shows a connection between the levels of cache in figure 6. This logic as well as the first and second memory types (L1 and L2) are configued by a program – refer to paragraphs 23-24. The data prefetch unit as defined by the Examiner must be configued as well by the program when moving data since the cache line size and blocking factor can change, so different amounts of data can be exchanged for the same access when different programs run.

As per claims 2 and 13, as taught in paragraphs 23 and 29 of Paulraj, no specific cache is present in the system of Paulraj. Rather, an FPGA is utilized as representing a caching hierarchy and is optimized based on the memory needs of a specific program running on the reconfigurable processor.

As per claims 3 and 14, Paulraj teaches in paragraph 23 that a specific [cache] line size of contiguous data is not retrieved since the data line size is optimized based on the memory needs of the program when executing on the reconfigurable processor. Refer also to paragraph 29. Further, it is therefore inherent that the second memory have a charactersitic line size since Paulraj teaches in ¶¶22-23 that a best line size for the memory arrangement for a particular program is determined and utilzied when that program is run. For example, a line-size characteristic would be ultized when transferring data from the L2 cache to the L1 cache.

As per claim 4, Paulraj teaches that a load/store unit is used to access the caches (L1-L3) in order to determine if cache data is present in the cache hierarchy (paragraph 6). Since the functional unit 102 (figure 6) is responsible for accessing the programmable memory unit 104, the Examiner is therefore considering the load/store unit logic of the programmable memory unit that is responsible for for accessing the L1 and L2 caches (first and second memory types) to be a memory controller. It can be seen that the memory controller, as defined by the Examiner, controls the transfer of data between the memory (assuming second memory L2) and the data prefetch unit, since the memory controller (load/store unit logic) is responsible for retrieving the data from the cache if a hit occurs (paragraph 4).

As per claim 5, as taught in paragraph 1, an external memory (element 18, figure 1) is generaly coupled to a microprocessor and holds data to be used by the microcontroller during program execution. The Examiner is considering the process of writing data back to the external memory from the FPGA memory 104 containing the caches (on-board memory), such as during a write-back scheme as known in the art, to be performed by the data prefetch unit portion of the functional logic as defined above by the Examiner. The data prefetch logic, as defined above, is responsible for all of the transfer of data into, out of, and between the FPGA memory 104.

As per claim 6, the Examiner is regarding a --register-- in its broadest reasonable sense and it thus considering it be to be a unit of logic. Therefore, the portion of the function logic that is responsible for the movement of data (as defined above to be the data prefetch unit) is being considered by the Examiner as containing a --register-- portion of the reconfigurable processor since, for instance, the blocking factor and line size of the programmable memory 112 can change, a --register-- or portion of the reconfigurable processor must be set in order to indicate

the currnet line size and blocking factor when a given application is being run on the reconfigurable processor at a given point in time. Refer to paragraph 23.

As per claim 7, the Examiner is considering the process of --disassembling the data prefetch unit-- as modifying the data prefetch unit logic of the fucntion logic 102 every time the program being executed by the reconfigurable processor changes. It can be seen that the data prefetch unit changes during these intervals since the cache line size, blocking factor, and associativity of the FPGA changes when optimal for the next program to be executed (refer to paragraph 23). Thus it can be seen that the data prefetch unit logic is --disassembled-- when another program is executed by the reconfigurable processor of Paulraj.

As per claim 8, as can be seen that the FPGA memory 112, that comprises the first and second memories (L1 and L2) and which is accessed by the data prefetch unit of the functional unit 102 as discussed above, is a --processor memory-- (part of cpu 110). It can also be seen that the --second memory-- (L2) is also a --processor memory-- since it is contained within reconfigurable processor 110. Therefore, since the data pretech unit can access the L2 cache as discussed above in the rejection of claim 1, the data prefetch unit can retrive data from the L2 portion of --processor memory--112.

As per claim 9, as shown in figure 1 and taught in paragraph 1 of Paulraj, the system 10 is actually a microprocessor, which contains a memory controller 14. The main difference between the prior art of figure 1 and the invention of Paulraj in figure 6 is that the memroy hierarchy is configurable and accessed by a fucntional unit in lieu of a separate memory controller logic (paragraph 9). Therefore, since the memory controller logic for accessing the

cache hierarchy is still contained within cpu 110 of figure 6, it can be seen that the cpu 110 is actually a microprocessor. It follows that the --processor memory-- 112 is therefore a --microprocessor memory--.

As per claim 10, since the cpu 110 of figure 6 is a reconfigurable processer (able to reconfigure its memory heirarchy to match the needs of the application it is currently running), it can be seen that the cpu memory 112 is a reconfigurable processor memory.

As per claim 11, Paulraj depicts a reconfigurable hardware system in figure 6. Paulraj further teaches in paragraph 26 that when a particular application is to be run by the reconfigrable processor 110, a configuration vector is retrieved to program the programmable memory 112 (figure 6). As shown in figure 6, the step of accesing the configuration vector is executed outside of the reconfigurable processor 110. Therefore, the Examiner is considering the memory that contains the configuration vectors to be a--common memory-- and a data prefetch unit (reconfiguration unit 106 executing on the reconfigurable processor 110) accessing the common memory in order to determine how to program the memory 112 (paragraph 29). The data prefetch unit 106 is --configured-- by an application to be excuted on the sysem 110 since when a new application is to be executed, the data prefetch unit is called upon (or configured) to access the configuration vector for the particular application.

As per claim 12, the Examiner is considering a --memory controller-- to be the system portion utilized when creating a new configuration vector for an application. Such a process occurs in figure 5 and taught in paragraghs 23-25 of Paulraj. When a new configuration vector is created by analizing performance information that has been collected for the application. The Examiner is thereby considering the --memory controller-- to be the element of the

reconfigurable hardware system that is associated with storing the new configuration vector into the common memory so that the vector can be accessed later when the same application is run again.

As per claim 15, the Examiner is considering the reconfiguration module 106 of the reconfigurable processsor 110, as comprising two distinct elements: a --computational unit-- and a --data access unit--. The data access unit is the element that is responsible for accessing the configuration vector as taught in paragraph 29 of Paulraj; or in other words, the Examiner is considering the --data access unit-- to be the same as the --memory controler-- defined in the rejection of claim 12. The Examiner is further considering the --computational unit-- of the rconfiguration module 106 to be the element that sets up the programmable memory module 104 using the configuration vector that was accessed by the --data access unit-- (paragraph 29).

As per claim 16, as taught by Paulraj in paragraph 29, the --data access unit-- supplies the configuration vector to the --computational unit-- in order to set up the programmable memory 104 as required by the application to be run on the reconfurable processor 110.

As per claim 17, the Examiner is considering a --data prefetch unit-- to be the reconfiguration unit 106 of reconfigurable processor 110 (figure 6). As taught in paragraph 26 and 29 of Paulraj, the --data prefetch unit-- accesses a memory in order to determine if a configuration vector is known for a given application, and if so, the vector is retrieved (from the memory). If this --data-- (configuration vector) is not known then a simulation is performed with the application in order to collect performance information. The Examiner is considering the element that executes and collects the performance data as being a --computational unit-- and the element of Paulraj that stores the configuration vector, once determined, to be a --data access

unit-- since it stores the vector into the --memory-- from which it can be later retrieved (step 212 of figure 5). The --computational unit--, --data access unit--, and the --data prefetch unit-- are all --configured-- by a program (application) since (1) a new application configures the computational unit portion of the reconfiguration unit to perform a simulation in order to determine the optimal memory hierarchy organization; (2) the new application configures the --data access unit-- to store and retrieve (step 212) the configuration vector for that particular application; and (3) the --data prefetch unit-- is configured by the application to determine if a configuration file exists for the application and if so, the data prefetch unit is configured by the program the programmable memory 112 in order to optimize the programmable memory for that particular application.

As per claim 18, the --data-- (configuration vector) is transferred from the --computational unit-- to the --data access unit-- when the configuration unit has created a configuration vector (step 208 of figure 5). The --data-- is written to the memory --from-- the --data prefetch unit-- since the data prefetch unit (reconfiguration unit 106) is the element that executed the beginning of the configuration vector creation process (step 200 of figure 5). Refer to paragraph 26. Thus the Examiner is considering the data as being written --from-- the data prefetch unit.

As per claim 19, as taught in paragraph 26, if the configuration vector is known, the vector is retrieved from the memory to the data prefetch unit (reconfiguration unit 106). The data is read directly from the data prefetch unit when a request to create a configuration vector is made for a new application as shown in figure 6 since the data prefetch unit is responsible for being the vector creation process. The data is directed from the data prefetch unit (reconfigure

logic) to be read from the memory by the data access unit to the computational unit where it is processed to produce a configuration vector.

As per claim 20, as stated above, the configuration vector (--data--) is created by the computational unit via acquired simulation data. The configuration vector is the resultant product that is transferred from the memory to the data prefect unit when it is determined that the configuration vector for the application is available (paragraph 26). Thus --all-- of the data that is transferred is processed by the computational unit (albeit before the transfer occurs) since the data prefetch unit required the entire configuration vector in order to set up the programmable memory 112.

As per claim 21, Paulraj shows in paragraph 26 that an explicit request for the configuration vector for the current application results in the data (if it exists) selected for the optimal configuration of the programmable memory 112 for that application.

As per claim 22, the Examiner is not considering the data (configuration vector) to be the size of a complete cache line since the data is used to create a cache hierarchy. In other words, the caches (L1-L3) of the programmable memory 112 are not programmed when the data is transferred from the memory to the data prefetch unit; therefore, the data cannot be a complete cache line.

As per claim 23, since the Examiner defined the portion of the reconfiguration unit that accesses the configuration file (data) from the memory, the Examiner is defining the logic that controls the actual transfer of that data to the data prefetch unit (portion of the reconfiguration unit that executes the fetch of the configuration vector and then programs the programmable memory 112) to be a --memory controller--. Thus the data access unit determines whether a

configuration vector exists for an application and if so, the memory controller sends that data to the data prefetch unit.

As per claim 24, Paulraj shows a reconfigurable processor in figure 6 that comprises a computation unit 110 and a data access unit (elements 120 and 114, which comprise the reconfiguration unit 106 of figure 4 - ¶28). In figure 6, the data access unit can be seen as being coupled to the computational unit. The data access unit retrieves data (configuration vector) from a memory internal to the data access unit (i.e. reconfiguration unit) and supplies the data to the computation unit in the form of modifications to the cache FPGA module 112. Refer to ¶23.

The computation unit is configured by the program (application) that is to be executed on it by the run-time profile that is created and stored by the reconfiguration unit (¶22), thereby creating the optimal configuration of the different caches. The data access unit (specifically the memory portion used to store configuration profiles for the different application programs) is configured by the program that is to run on the reconfigurable processor. When a new program is to be run, [as a result] the program configures the reconfiguration unit to collect statistics regarding the memory usages (caches L1, L2, and L3) of the program and a configuration vector is associated with the respective program and stored in the reconfiguration unit. Refer to ¶¶23-24. When a program is known, the program [as a result] configures the data access unit (reconfiguration unit) to retrieve the associated configuration vector and apply it to the FPGA memory of the reconfigurable processor (¶29).

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Magoshi (U.S. Patent Application Publication No. 2003/0208658) teaches the memory utilization characteristics of an L1 and an L2 cache in figure 2. As shown, the L1 cache is always accessed (high memory utilization) upon an access request from a processor and the L2 cache is only accessed (lower memory utilization) when a miss occurs with respect to the L1 cache.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Shane M. Thomas whose telephone number is (571) 272-4188. The examiner can normally be reached on M-F 8:30 - 5:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matt M. Kim can be reached on (571) 272-4182. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Shane M. Thomas

HONG CHONG KIM
PRIMARY EXAMINER

Client Matter No. 80404.0033.001
Express Mail No.: Via Facsimile

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Serial No. 10/869,200 Application of: Daniel Poznanovic, et al. Filed: June 16, 2004 Art Unit: 2186 Examiner: Thomas, Shane M. Attorney Docket No. SRC028 For: SYSTEM AND METHOD OF ENHANCING EFFICIENCY AND UTILIZATION OF MEMORY BANDWIDTH IN RECONFIGURABLE HARDWARE	Confirmation No.: 5929 Customer No.: **25235** **EXPEDITED PROCEDURE UNDER 37 C.F.R. 1.116**

AMENDMENT AND RESPONSE PURSUANT TO OFFICE ACTION DATED JULY 12, 2005

MAIL STOP AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the office communication mailed July 12, 2005 please amend the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims which begins on page 2 of this paper.

Remarks/Arguments begin on page 6 of this paper.

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently Amended) A reconfigurable processor that instantiates an algorithm as hardware comprising:

a first memory having a first characteristic memory bandwidth and/or memory utilization; and

a data prefetch unit coupled to the first memory, wherein the data prefetch unit retrieves data from a second memory of second characteristic memory bandwidth and/or memory utilization and place the retrieved data in the first memory and wherein at least the first memory and data prefetch unit are configured by a program.

2. (Cancelled)

3. (Cancelled)

4. (Previously Presented) The reconfigurable processor of claim 1, wherein the data prefetch unit is coupled to a memory controller that controls the transfer of the data between the second memory and the data prefetch unit.

5. (Previously Presented) The reconfigurable processor of claim 1, wherein the data prefetch unit receives processed data from on-processor memory and writes the processed data to an external off-processor memory.

6. (Original) The reconfigurable processor of claim 1, wherein the data prefetch unit comprises at least one register from the reconfigurable processor.

7. (Original) The reconfigurable processor of claim 1, wherein the data prefetch unit is disassembled when another program is executed on the reconfigurable processor.

8. (Previously Presented) The reconfigurable processor of claim 1 wherein said second memory comprises a processor memory and said data prefetch unit is operative to retrieve data from the processor memory.

9. (Original) The reconfigurable processor of claim 8 wherein said processor memory is a microprocessor memory.

10. (Original) The reconfigurable processor of claim 8 wherein said processor memory is a reconfigurable processor memory.

11. (Currently Amended) A reconfigurable hardware system, comprising:

a common memory; and

one or more reconfigurable processors that can instantiate an algorithm as hardware coupled to the common memory, wherein at least one of the reconfigurable processors includes a data prefetch unit to read and write data between the data prefetch unit and the common memory, and wherein the data prefetch unit is configured by a program executed on the system.

12. (Original) The reconfigurable hardware system of claim 11, comprising a memory controller coupled to the common memory and the data prefetch unit.

13. (Cancelled)

14. (Cancelled)

15. (Previously Presented) The reconfigurable hardware system of claim 11, wherein the at least one of the reconfigurable processors also includes a computational unit coupled to a data access unit.

16. (Original) The reconfigurable hardware system of claim 15, wherein the computational unit is supplied the data by the data access unit.

17. (Previously Presented) A method of transferring data comprising:

transferring data between a memory and a data prefetch unit in a reconfigurable processor; and

transferring the data between a computational unit and a data access unit, wherein the computational unit and the data access unit, and the data prefetch unit are configured by a program.

18. (Original) The method of claim 17, wherein the data is written to the memory, said method comprising:

transferring the data from the computational unit to the data access unit; and

writing the data to the memory from the data prefetch unit.

19. (Previously Presented) The method of claim 17, wherein the data is read from the memory, said method comprising:

transferring the data from the memory to the data prefetch unit; and

reading the data directly from the data prefetch unit to the computational unit through the data access unit.

20. (Original) The method of claim 19, wherein all the data transferred from the memory to the data prefetch unit is processed by the computational unit.

21. (Original) The method of claim 19, wherein the data is selected by the data prefetch unit based on an explicit request from the computational unit.

22. (Original) The method of claim 17, wherein the data transferred between the memory and the data prefetch unit is not a complete cache line.

23. (Original) The method of claim 17, wherein a memory controller coupled to the memory and the data prefetch unit, controls the transfer of the data between the memory and the data prefetch unit.

24. (Currently Amended) A reconfigurable processor comprising:

a computational unit; and

a data access unit coupled to the computational unit, wherein the data access unit retrieves data from memory and supplies the data to the computational unit, and wherein the computational unit and the data access unit are configured by a program to instantiate an algorithm as hardware.

REMARKS/ARGUMENTS

Claims 1, 4-12 and 15-24 remain in the application. Claims 2, 3, 13 and 14 are cancelled. Claims 1, 11 and 24 are amended to more distinctly describe the subject matter of the invention.

A. Rejections under 35 U.S.C. 112.

The cancellation of claims 2, 3, 13, 14 renders the rejection under 35 U.S.C. 112 moot. However, the concept of a configurable processor that does not have a cache is believed to be supported by the claims themselves, and the subject matter of these claims is not waived.

B. Rejections under 35 U.S.C. 102.

Claims 1-24 were rejected under 35 U.S.C. 102 based upon Paulraj. This rejection is respectfully traversed.

Claim 1 is amended to adopt language from the definition of "reconfigurable processor" appearing in paragraph 39 of the specification as filed. This amendment is not believed to raise any new issues nor require further search because this meaning of reconfigurable processor is consistent with the application as filed and consistent with the definition of that term asserted in prior remarks submitted on April 11, 2005.

As amended, independent claim 1 calls for a reconfigurable processor that instantiates an algorithm as hardware. Although the reference show a reconfigurable cache, Paulraj does not show or suggest a reconfigurable processor that instantiates an algorithm as hardware. Moreover, nothing in Paulraj would suggest the rather significant changes required to replace the CPU with a reconfigurable processor that can instantiate an algorithm as hardware. For at least these reasons claim 1 is not anticipated nor made obvious by Paulraj.

Claims 2-10 that depend from claim 1 are allowable over Paulraj for at least the same reasons as claim 1 as well as the limitations that are presented in those claims.

Claim 11 calls for a reconfigurable hardware system comprising one or more reconfigurable processors that can instantiate an algorithm as hardware. As noted above with respect to claim 1, Paulraj does not show or suggest even one reconfigurable processor that can instantiate an algorithm as hardware. For at least these reasons claim 11 and claims 12-16 that depend from claim 11 are believed to be allowable over Paulraj.

Independent claim 17 calls for, among other things, transferring data between a memory and a data prefetch unit in a reconfigurable processor. Paulraj does not show or suggest a data prefetch unit, nor does Paulraj suggest transferring data between a memory and a data prefetch unit in a reconfigurable processor. The cited portions of Paulraj deal with retrieving a configuration vector but do not use the work "data prefetch unit" or or describe any functional unit that operates in the same way as a data prefetch unit. Moreover, even if the broad construction set out in the Office action is applied, Paulraj does not suggest configuring the computational unit, data access unit and the data prefetch unit by a program. Paulraj simply cannot suggest this configurability because the computational unit in Paulraj is not configurable. For at least these reasons claim 17 and claims 18-23 that depend from claim 17 are allowable over Paulraj.

Claim 24 as amended is believed to clarify that the term "configured" as used in the claims refers to configuration that allows the configured device to instantiate an algorithm as hardware. Loading a software program into a general purpose computational device such as shown in Paulraj does not result in the instantiation of an algorithm as hardware. Accordingly, claim 24 is believed to be allowable over the relied on reference.

C. Conclusion.

In view of all of the above, the claims are now believed to be allowable and the case in condition for allowance which action is respectfully requested. Should the Examiner be of the opinion that a telephone conference would

expedite the prosecution of this case, the Examiner is requested to contact Applicants' attorney at the telephone number listed below.

Any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

August 26, 2005

Stuart T. Langley, Reg. No. 33,940
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(720) 406-5335 Tel
(303) 899-7333 Fax

Advisory Action Before the Filing of an Appeal Brief	**Application No.** 10/869,200	**Applicant(s)** POZNANOVIC ET AL.	
	Examiner Shane M. Thomas	**Art Unit** 2186	

--The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

THE REPLY FILED 26 August 2005 FAILS TO PLACE THIS APPLICATION IN CONDITION FOR ALLOWANCE.

1. ☒ The reply was filed after a final rejection, but prior to or on the same day as filing a Notice of Appeal. To avoid abandonment of this application, applicant must timely file one of the following replies: (1) an amendment, affidavit, or other evidence, which places the application in condition for allowance; (2) a Notice of Appeal (with appeal fee) in compliance with 37 CFR 41.31; or (3) a Request for Continued Examination (RCE) in compliance with 37 CFR 1.114. The reply must be filed within one of the following time periods:
   a) ☒ The period for reply expires 3 months from the mailing date of the final rejection.
   b) ☐ The period for reply expires on: (1) the mailing date of this Advisory Action, or (2) the date set forth in the final rejection, whichever is later. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of the final rejection.

   Examiner Note: If box 1 is checked, check either box (a) or (b). ONLY CHECK BOX (b) WHEN THE FIRST REPLY WAS FILED WITHIN TWO MONTHS OF THE FINAL REJECTION. See MPEP 706.07(f).

Extensions of time may be obtained under 37 CFR 1.136(a). The date on which the petition under 37 CFR 1.136(a) and the appropriate extension fee have been filed is the date for purposes of determining the period of extension and the corresponding amount of the fee. The appropriate extension fee under 37 CFR 1.17(a) is calculated from: (1) the expiration date of the shortened statutory period for reply originally set in the final Office action; or (2) as set forth in (b) above, if checked. Any reply received by the Office later than three months after the mailing date of the final rejection, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

NOTICE OF APPEAL

2. ☐ The Notice of Appeal was filed on _____. A brief in compliance with 37 CFR 41.37 must be filed within two months of the date of filing the Notice of Appeal (37 CFR 41.37(a)), or any extension thereof (37 CFR 41.37(e)), to avoid dismissal of the appeal. Since a Notice of Appeal has been filed, any reply must be filed within the time period set forth in 37 CFR 41.37(a).

AMENDMENTS

3. ☒ The proposed amendment(s) filed after a final rejection, but prior to the date of filing a brief, will not be entered because
   (a) ☒ They raise new issues that would require further consideration and/or search (see NOTE below);
   (b) ☐ They raise the issue of new matter (see NOTE below);
   (c) ☐ They are not deemed to place the application in better form for appeal by materially reducing or simplifying the issues for appeal; and/or
   (d) ☐ They present additional claims without canceling a corresponding number of finally rejected claims.

   NOTE: *See Continuation Sheet*. (See 37 CFR 1.116 and 41.33(a)).
4. ☐ The amendments are not in compliance with 37 CFR 1.121. See attached Notice of Non-Compliant Amendment (PTOL-324).
5. ☐ Applicant's reply has overcome the following rejection(s): _____.
6. ☐ Newly proposed or amended claim(s) _____ would be allowable if submitted in a separate, timely filed amendment canceling the non-allowable claim(s).
7. ☒ For purposes of appeal, the proposed amendment(s): a) ☒ will not be entered, or b) ☐ will be entered and an explanation of how the new or amended claims would be rejected is provided below or appended.
   The status of the claim(s) is (or will be) as follows:
   Claim(s) allowed: _____.
   Claim(s) objected to: _____.
   Claim(s) rejected: *1-24*.
   Claim(s) withdrawn from consideration: _____.

AFFIDAVIT OR OTHER EVIDENCE

8. ☐ The affidavit or other evidence filed after a final action, but before or on the date of filing a Notice of Appeal will not be entered because applicant failed to provide a showing of good and sufficient reasons why the affidavit or other evidence is necessary and was not earlier presented. See 37 CFR 1.116(e).
9. ☐ The affidavit or other evidence filed after the date of filing a Notice of Appeal, but prior to the date of filing a brief, will not be entered because the affidavit or other evidence failed to overcome all rejections under appeal and/or appellant fails to provide a showing a good and sufficient reasons why it is necessary and was not earlier presented. See 37 CFR 41.33(d)(1).
10. ☐ The affidavit or other evidence is entered. An explanation of the status of the claims after entry is below or attached.

REQUEST FOR RECONSIDERATION/OTHER

11. ☐ The request for reconsideration has been considered but does NOT place the application in condition for allowance because: _____.
12. ☐ Note the attached Information Disclosure Statement(s). (PTO/SB/08 or PTO-1449) Paper No(s). _____
13. ☐ Other: _____.

Continuation Sheet (PTOL-303) **Application No.**

Continuation of 3. NOTE: The amendment to the claims has changed the scope of independent claims 1,11, and 24, and as such, further search and consideration are required.

HONG CHONG KIM
PRIMARY EXAMINER

Office Action Summary	Application No.	Applicant(s)	
	10/869,200	POZNANOVIC ET AL.	
	Examiner	Art Unit	
	Shane M. Thomas	2186	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *12 September 2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,4-12 and 15-24* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,4-12 and 15-24* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

This Office action is responsive to the amendment filed 8/26/2005. Claims 1,11, and 24. have been amended; claims 2,3,13, and 14 have been canceled. Claims 1,4-12, and 15-24 are pending.

Continued Examination Under 37 CFR 1.1 1 4

A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection on 9/12/2005. Since this application is eligible for continued examination under 37 CFR 1.1 14, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 8/26/2005 has been entered.

All previously outstanding objections and rejections to the Applicant's disclosure and claims not contained in this Action have been respectfully withdrawn by the Examiner hereto.

Response to Amendment

The rejections of claims 1,11,17, and 24 have been modified to reflect the amendments and/or Applicant's arguments to the respective claims.

Response to Arguments

Applicant's arguments filed 8/26/2005 have been fully considered but they are not persuasive for the following reasons.

Applicant argues on page 6 of the response that the prior art reference of Paulraj "does not show or suggest a reconfigurable processor that instantiates an algorithm as hardware." The Examiner respectfully traverses. Paulraj teaches in the abstract for one, that the system described determines "an optimal configuration of memory for a particular *application*." The Applicant teaches in ¶55 of the originally filed disclosure that "any computer program [i.e. application] is a collection of algorithms." Therefore it can be seen that since the processor 100 of Paulraj can reconfigure the memory 104 based on the application (or computer program) that is to execute on the processor, that so to can the reconfigurable processor system of Paulraj "instantiate an algorithm (i.e. an application) as hardware (i.e. the FPGA module 104 that is used as a cache memory)."

As per the Applicant's arguments regarding claim 11, the Examiner has shown in above in the discussion of claim 1 that Paulraj teaches a reconfigurable processor 100, as claimed by the Applicant, that instantiates an algorithm as hardware.

As per the Applicant's arguments regarding claim 17 on page 7, the Applicant argues that the prior art reference of Paulraj "does not show or suggest a data prefetch unit, nor suggest transferring data between a memory and a data prefetch unit in a reconfigurable processor. As explained in the Examiner's previous rejection of claim 17, the Examiner is considering the reconfiguration unit 106 of Paulraj to be a --data prefetch unit-- since Paulraj teaches that the unit 106 *prefetches* a configuration vector (i.e. retrieves data from an inherent and non-shown

memory) and sets up a programmable memory module 104 (i.e. cache) *before* executing the application relating to the configuration vector (refer to ¶24 and ¶29). Figure 4 of Paulraj clearly shows the --data prefetch unit-- 106 being in a reconfigurable processor 100. Although the cited reference does not explicitly use the phrase "data prefetch unit," and may or may not perform all of the functionality of a "data prefetch unit," as discussed in the Applicants disclosure, the reconfiguration unit 106 performs the ***claimed*** *functionality* of the "data prefetch unit" as discussed above (i.e. merely transferring data between a memory in a reconfigurable processor).

Further, the Applicant argues regarding claim 17 that "Paulraj does not suggest configuring the computational unit, data access unit, and the data prefetch unit by a program. Paulraj simply cannot suggest this configurability because the computational unit in Paulraj is not configurable." The Examiner respectfully traverses. All of the computational, data access, and data prefetch units are configured by a program, as immediately discussed. As defined by the Examiner, the "computational unit" of Paulraj is being considered to be the element of the system of Paulraj that executes and collects the performance data regarding how a specific application utilizes memory in order to determine an optimal memory configuration as discusses in ¶27. Figure 5 of Paulraj shows a method for creating a configuration vector by using the --computational unit-- in steps 204-206. The Examiner is considering the inherent *program* that is being executed in order to perform the steps of figure 5 to be the *program* that configures the computational unit. Therefore, it can be seen that Paulraj *does* suggest configuring the computational unit by a program. The *program* of figure 5 *configures* the computational unit to collect data for a specific application's memory usage statistics in order to create a configuration vector that allows the system of Paulraj to optimally reconfigure the programmable memory

module 104. Thus the computational unit can be configured to collect memory usage statistics for a plurality of applications that are to be executed by the reconfigurable processor 100 of Paulraj (¶23).

The same reasoning applies to the data access and data prefetch units. The *program* that is executing the steps of figure 5 (i.e. running on the system of Paulraj that implements the method) *configures* the data access unit to retrieve/store a configuration vector (step 212) based on if a new configuration vector had to be created and further *configures* the data prefetch unit to search for a configuration vector and retrieve that vector if found (steps 200 and 212).

As per the Applicant's arguments regarding claim 24 "that loading a software program into a general purpose computational device such as shown in Paulraj does not result in the instantiation of an algorithm as hardware." The Examiner respectfully traverses. Once the software program has been loaded into the computational unit, a variety of simulations are performed and memory usage statistics are gathered by the computational unit in order to create a configuration vector as taught in ¶¶23-24. This vector allows the programmable memory module 104 of Paulraj to be reconfigured to the most optimal memory configuration for that specific software program (¶26). As discussed supra, a software program or application is a collection of "algorithms"; therefore, the configuration vector for a particular software program allows the system of Paulraj to instantiate a software program as hardware since the configuration vector represents optimal configuration of the hardware (programmable memory module 104 - element 112 of figure 6).

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Claims 1-24 are rejected under 35 U.S.C. 102(e) as being anticipated by Paulraj (U.S. Patent Application Publication No. 2003/0084244).

As per claim 1, Paulraj shows a reconfigurable processor in figure 6 and a first memory (L1) having a first characteristc memory utilization and a second memory (L2) having a second characteristic memory utilization. It is well known in the art that L1 caches have a higher utilziation rate than a lower-level cache such as L2. Paulraj teaches in ¶1 that upon a command from a processor, a search for the requested data is begines with the highest level cache (L1) and [if a miss occurs] continues next to the next level cache (L2). Thus it is inherent that the memory utilziation characteristc of the L1 cache of the reconfigurable processor 110 in figure 6 is greater than the memory utilziation characteristic of the L2 cache (and likewise for the L3 cache) as the L2 cache would only be utilzied when a miss to the L1 cache occurred. In other words, the reconfigurable processor *always* utilizes the L1 cache for a memory access and the *only* utilzies the L2 cache for requested data when the data is not in the L1 cache. Therefore, the cache utilziation characteristics of the --first memory-- and the --second memory-- are different.

Paulraj further teaches a functional unit 102 that executes applications using the memories L1 and L2 (paragraph 9). As is known in the art, a cache memory controller is often used to access and move data between a memory hierarchy. The Examiner is considering a data prefetch unit to be the logic assocatied with the moving, and only the moving, of data between the first and second memories (L1 and L2) since Paulraj shows a connection between the levels of cache in figure 6. This logic as well as the first and second memory types (L1 and L2) are configued by a program – refer to paragraphs 23-24. The data prefetch unit as defined by the Examiner must be configued as well by the program when moving data since the cache line size and blocking factor can change, so different amounts of data can be exchanged for the same access when different programs run.

The reconfigurable processor of Paulraj has the ability to collect memory usage statistics for a particular application and based on those statistics, create a configuration vector as taught in ¶¶23-24. This vector allows the programmable memory module 104 of Paulraj to be reconfigured to the most optimal memory configuration for that specific software program (¶26). As defined by the Applicant in ¶55 of the originally filed specification, a software program or application is a collection of "algorithms"; therefore, the configuration vector for a particular software program allows the system of Paulraj to instantiate a software program as hardware since the configuration vector represents optimal configuration of the hardware (programmable memory module 104 - element 112 of figure 6).

As per claims 2 and 13, as taught in paragraphs 23 and 29 of Paulraj, no specific cache is present in the system of Paulraj. Rather, an FPGA is utilized as representing a caching hierarchy

and is optimized based on the memory needs of a specific program running on the reconfigurable processor.

As per claims 3 and 14, Paulraj teaches in paragraph 23 that a specific [cache] line size of contiguous data is not retrieved since the data line size is optimized based on the memory needs of the program when executing on the reconfigurable processor. Refer also to paragraph 29. Further, it is therefore inherent that the second memory have a charactersitic line size since Paulraj teaches in ¶¶22-23 that a best line size for the memory arrangement for a particular program is determined and utilzied when that program is run. For example, a line-size characteristic would be ultized when transferring data from the L2 cache to the L1 cache.

As per claim 4, Paulraj teaches that a load/store unit is used to access the caches (L1-L3) in order to determine if cache data is present in the cache hierarchy (paragraph 6). Since the functional unit 102 (figure 6) is responsible for accessing the programmable memory unit 104, the Examiner is therefore considering the load/store unit logic of the programmable memory unit that is responsible for for accessing the L1 and L2 caches (first and second memory types) to be a memory controller. It can be seen that the memory controller, as defined by the Examiner, controls the transfer of data between the memory (assuming second memory L2) and the data prefetch unit, since the memory controller (load/store unit logic) is responsible for retrieving the data from the cache if a hit occurs (paragraph 4).

As per claim 5, as taught in paragraph 1, an external memory (element 18, figure 1) is generaly coupled to a microprocessor and holds data to be used by the microcontroller during program execution. The Examiner is considering the process of writing data back to the external memory from the FPGA memory 104 containing the caches (on-board memory), such as during

a write-back scheme as known in the art, to be performed by the data prefetch unit portion of the functional logic as defined above by the Examiner. The data prefetch logic, as defined above, is responsible for all of the transfer of data into, out of, and between the FPGA memory 104.

As per claim 6, the Examiner is regarding a --register-- in its broadest reasonable sense and it thus considering it be to be a unit of logic. Therefore, the portion of the function logic that is responsible for the movement of data (as defined above to be the data prefetch unit) is being considered by the Examiner as containing a --register-- portion of the reconfigurable processor since, for instance, the blocking factor and line size of the programmable memory 112 can change, a --register-- or portion of the reconfigurable processor must be set in order to indicate the currnet line size and blocking factor when a given application is being run on the reconfigurable processor at a given point in time. Refer to paragraph 23.

As per claim 7, the Examiner is considering the process of --disassembling the data prefetch unit-- as modifying the data prefetch unit logic of the fucntion logic 102 every time the program being executed by the reconfigurable processor changes. It can be seen that the data prefetch unit changes during these intervals since the cache line size, blocking factor, and associativity of the FPGA changes when optimal for the next program to be executed (refer to paragraph 23). Thus it can be seen that the data prefetch unit logic is --disassembled-- when another program is executed by the reconfigurable processor of Paulraj.

As per claim 8, as can be seen that the FPGA memory 112, that comprises the first and second memories (L1 and L2) and which is accessed by the data prefetch unit of the functional unit 102 as discussed above, is a --processor memory-- (part of cpu 110). It can also be seen that the --second memory-- (L2) is also a --processor memory-- since it is contained within

reconfigurable processor 110. Therefore, since the data pretech unit can access the L2 cache as discussed above in the rejection of claim 1, the data prefetch unit can retrive data from the L2 portion of --processor memory--112.

As per claim 9, as shown in figure 1 and taught in paragraph 1 of Paulraj, the system 10 is actually a microprocessor, which contains a memory controller 14. The main difference between the prior art of figure 1 and the invention of Paulraj in figure 6 is that the memroy hierarchy is configurable and accessed by a fucntional unit in lieu of a separate memory controller logic (paragraph 9). Therefore, since the memory controller logic for accessing the cache hierarchy is still contained within cpu 110 of figure 6, it can be seen that the cpu 110 is actually a microprocessor. It follows that the --processor memory-- 112 is therefore a --microprocessor memory--.

As per claim 10, since the cpu 110 of figure 6 is a reconfigurable processer (able to reconfigure its memory heirarchy to match the needs of the application it is currently running), it can be seen that the cpu memory 112 is a reconfigurable processor memory.

As per claim 11, Paulraj depicts a reconfigurable hardware system in figure 6. Paulraj further teaches in paragraph 26 that when a particular application is to be run by the reconfigrable processor 110, a configuration vector is retrieved to program the programmable memory 112 (figure 6). As shown in figure 6, the step of accesing the configuration vector is executed outside of the reconfigurable processor 110. Therefore, the Examiner is considering the memory that contains the configuration vectors to be a--common memory-- and a data prefetch unit (reconfiguration unit 106 executing on the reconfigurable processor 110) accessing the common memory in order to determine how to program the memory 112 (paragraph 29).

The data prefetch unit 106 is --configured-- by an application to be excuted on the sysem 110 since when a new application is to be executed, the data prefetch unit is called upon (or configured) to access the configuration vector for the particular application.

The reconfigurable processor of Paulraj has the ability to collect memory usage statistics for a particular application and based on those statistics, create a configuration vector as taught in ¶¶23-24. This vector allows the programmable memory module 104 of Paulraj to be reconfigured to the most optimal memory configuration for that specific software program (¶26). As defined by the Applicant in ¶55 of the originally filed specification, a software program or application is a collection of "algorithms"; therefore, the configuration vector for a particular software program allows the system of Paulraj to instantiate a software program as hardware since the configuration vector represents optimal configuration of the hardware (programmable memory module 104 - element 112 of figure 6).

As per claim 12, the Examiner is considering a --memory controller-- to be the system portion utilized when creating a new configuration vector for an application. Such a process occurs in figure 5 and taught in paragraghs 23-25 of Paulraj. When a new configuration vector is created by analizing performance information that has been collected for the application. The Examiner is thereby considering the --memory controller-- to be the element of the reconfigurable hardware system that is associated with storing the new configuration vector into the common memory so that the vector can be accessed later when the same application is run again.

As per claim 15, the Examiner is considering the reconfiguration module 106 of the reconfigurable processsor 110, as comprising two distinct elements: a --computational unit-- and

a --data access unit--. The data access unit is the element that is responsible for accessing the configuration vector as taught in paragraph 29 of Paulraj; or in other words, the Examiner is considering the --data access unit-- to be the same as the --memory controler-- defined in the rejection of claim 12. The Examiner is further considering the --computational unit-- of the rconfiguration module 106 to be the element that sets up the programmable memory module 104 using the configuration vector that was accessed by the --data access unit-- (paragraph 29).

As per claim 16, as taught by Paulraj in paragraph 29, the --data access unit-- supplies the configuration vector to the --computational unit-- in order to set up the programmable memory 104 as required by the application to be run on the reconfurable processor 110.

As per claim 17, the Examiner is considering a --data prefetch unit-- to be the reconfiguration unit 106 of reconfigurable processor 110 (figure 6). As taught in paragraph 26 and 29 of Paulraj, the --data prefetch unit-- accesses a memory in order to determine if a configuration vector is known for a given application, and if so, the vector is retrieved (from the memory). If this --data-- (configuration vector) is not known then a simulation is performed with the application in order to collect performance information. The Examiner is considering the element that executes and collects the performance data as being a --computational unit-- and the element of Paulraj that stores the configuration vector, once determined, to be a --data access unit-- since it stores the vector into the --memory-- from which it can be later retrieved (step 212 of figure 5).

All of the computational, data access, and data prefetch units are configured by a program, as immediately discussed. As defined by the Examiner, the "computational unit" of Paulraj is being considered to be the element of the system of Paulraj that executes and collects

the performance data regarding how a specific application utilizes memory in order to determine an optimal memory configuration as discusses in ¶27. Figure 5 of Paulraj shows a method for creating a configuration vector by using the --computational unit-- in steps 204-206. The Examiner is considering the inherent *program* that is being executed in order to perform the steps of figure 5 to be the *program* that configures the computational unit. Therefore, it can be seen that Paulraj suggests configuring the computational unit by a program. The *program* of figure 5 *configures* the computational unit to collect data for a specific application's memory usage statistics in order to create a configuration vector that allows the system of Paulraj to optimally reconfigure the programmable memory module 104. Thus the computational unit can be configured to collect memory usage statistics for a plurality of applications that are to be executed by the reconfigurable processor 100 of Paulraj (¶23).

The same reasoning applies to the data access and data prefetch units. The *program* that is executing the steps of figure 5 (i.e. running on the system of Paulraj that implements the method) *configures* the data access unit to retrieve/store a configuration vector (step 212) based on if a new configuration vector had to be created and further *configures* the data prefetch unit to search for a configuration vector and retrieve that vector if found (steps 200 and 212).

As per claim 18, the --data-- (configuration vector) is transferred from the --computational unit-- to the --data access unit-- when the configuration unit has created a configuration vector (step 208 of figure 5). The --data-- is written to the memory --from-- the --data prefetch unit-- since the data prefetch unit (reconfiguration unit 106) is the element that executed the beginning of the configuration vector creation process (step 200 of figure 5). Refer

to paragraph 26. Thus the Examiner is considering the data as being written --from-- the data prefetch unit.

As per claim 19, as taught in paragraph 26, if the configuration vector is known, the vector is retrieved from the memory to the data prefetch unit (reconfiguration unit 106). The data is read directly from the data prefetch unit when a request to create a configuration vector is made for a new application as shown in figure 6 since the data prefetch unit is responsible for being the vector creation process. The data is directed from the data prefetch unit (reconfigure logic) to be read from the memory by the data access unit to the computational unit where it is processed to produce a configuration vector.

As per claim 20, as stated above, the configuration vector (--data--) is created by the computational unit via acquired simulation data. The configuration vector is the resultant product that is transferred from the memory to the data prefect unit when it is determined that the configuration vector for the application is available (paragraph 26). Thus --all-- of the data that is transferred is processed by the computational unit (albeit before the transfer occurs) since the data prefetch unit required the entire configuration vector in order to set up the programmable memory 112.

As per claim 21, Paulraj shows in paragraph 26 that an explicit request for the configuration vector for the current application results in the data (if it exists) selected for the optimal configuration of the programmable memory 112 for that application.

As per claim 22, the Examiner is not considering the data (configuration vector) to be the size of a complete cache line since the data is used to create a cache hierarchy. In other words, the caches (L1-L3) of the programmable memory 112 are not programmed when the data is

transferred from the memory to the data prefetch unit; therefore, the data cannot be a complete cache line.

As per claim 23, since the Examiner defined the portion of the reconfiguration unit that accesses the configuration file (data) from the memory, the Examiner is defining the logic that controls the actual transfer of that data to the data prefetch unit (portion of the reconfiguration unit that executes the fetch of the configuration vector and then programs the programmable memory 112) to be a --memory controller--. Thus the data access unit determines whether a configuration vector exists for an application and if so, the memory controller sends that data to the data prefetch unit.

As per claim 24, Paulraj shows a reconfigurable processor in figure 6 that comprises a computation unit 110 and a data access unit (elements 120 and 114, which comprise the reconfiguration unit 106 of figure 4 - ¶28). In figure 6, the data access unit can be seen as being coupled to the computational unit. The data access unit retrieves data (configuration vector) from a memory internal to the data access unit (i.e. reconfiguration unit) and supplies the data to the computation unit in the form of modifications to the cache FPGA module 112. Refer to ¶23.

The Examiner is considering the inherent *program* that is being executed in order to perform the steps of figure 5 to be the *program* that configures the computational unit. Therefore, it can be seen that Paulraj suggests configuring the computational unit by a program. The *program* of figure 5 *configures* the computational unit to collect data for a specific application's memory usage statistics in order to create a configuration vector that allows the system of Paulraj to optimally reconfigure the programmable memory module 104. Thus the

computational unit can be configured to collect memory usage statistics for a plurality of applications that are to be executed by the reconfigurable processor 100 of Paulraj (¶23).

The data access unit (specifically the memory portion used to store configuration profiles for the different application programs) is configured by the *program* that is responsible for running the method of figure 5 of Paulraj as discussed supra. When a new application is to be run, [as a result] the *program* performs the steps 204-206 to configure the reconfiguration unit to collect statistics regarding the memory usages (caches L1, L2, and L3) of the application and a configuration vector is associated with the respective application and stored in the reconfiguration unit. Refer to ¶¶23-24. When an application is known, the program executing the method of figure 5 [as a result] configures the data access unit (reconfiguration unit) to retrieve the associated configuration vector and apply it to the FPGA memory of the reconfigurable processor (¶29).

In other words, once the software program has been loaded into the computational unit, a variety of simulations are performed and memory usage statistics are gathered by the computational unit in order to create a configuration vector as taught in ¶¶23-24. This vector allows the programmable memory module 104 of Paulraj to be reconfigured to the most optimal memory configuration for that specific software program (¶26). As discussed supra, a software program or application is a collection of "algorithms"; therefore, the configuration vector for a particular software program allows the system of Paulraj to instantiate a software program as hardware since the configuration vector represents optimal configuration of the hardware (programmable memory module 104 - element 112 of figure 6).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Shane M. Thomas whose telephone number is (571) 272-4188. The examiner can normally be reached on M-F 8:30 - 5:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matt M. Kim can be reached on (571) 272-4182. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Shane M. Thomas

HONG CHONG KIM
PRIMARY EXAMINER

Client Matter No. 80404.0033.001
Via Facsimile

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Serial No. 10/869,200 Application of: Daniel Poznanovic, David E. Caliga, and Jeffrey Hammes Filed: June 16, 2004 Art Unit: 2186 Examiner: Thomas, Shane M. Attorney Docket No. SRC028 For: SYSTEM AND METHOD OF ENHANCING EFFICIENCY AND UTILIZATION OF MEMORY BANDWIDTH IN RECONFIGURABLE HARDWARE	Confirmation No.: 5929 Customer No.: **25235**

AMENDMENT

MAIL STOP AMENDMENT
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the office communication mailed October 19, 2005, please amend the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims which begins on page 2 of this paper.

Remarks/Arguments begin on page 6 of this paper.

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1. (Previously Presented) A reconfigurable processor that instantiates an algorithm as hardware comprising:

a first memory having a first characteristic memory bandwidth and/or memory utilization; and

a data prefetch unit coupled to the first memory, wherein the data prefetch unit retrieves data from a second memory of second characteristic memory bandwidth and/or memory utilization and place the retrieved data in the first memory and wherein at least the first memory and data prefetch unit are configured by a program.

2. (Cancelled)

3. (Cancelled)

4. (Previously Presented) The reconfigurable processor of claim 1, wherein the data prefetch unit is coupled to a memory controller that controls the transfer of the data between the second memory and the data prefetch unit.

5. (Previously Presented) The reconfigurable processor of claim 1, wherein the data prefetch unit receives processed data from on-processor memory and writes the processed data to an external off-processor memory.

6. (Original) The reconfigurable processor of claim 1, wherein the data prefetch unit comprises at least one register from the reconfigurable processor.

7. (Original) The reconfigurable processor of claim 1, wherein the data prefetch unit is disassembled when another program is executed on the reconfigurable processor.

8. (Previously Presented) The reconfigurable processor of claim 1 wherein said second memory comprises a processor memory and said data prefetch unit is operative to retrieve data from the processor memory.

9. (Original) The reconfigurable processor of claim 8 wherein said processor memory is a microprocessor memory.

10. (Original) The reconfigurable processor of claim 8 wherein said processor memory is a reconfigurable processor memory.

11. (Previously Presented) A reconfigurable hardware system, comprising:

a common memory; and

one or more reconfigurable processors that can instantiate an algorithm as hardware coupled to the common memory, wherein at least one of the reconfigurable processors includes a data prefetch unit to read and write data between the data prefetch unit and the common memory, and wherein the data prefetch unit is configured by a program executed on the system.

12. (Original) The reconfigurable hardware system of claim 11, comprising a memory controller coupled to the common memory and the data prefetch unit.

13. (Cancelled)

14. (Cancelled)

15. (Previously Presented) The reconfigurable hardware system of claim 11, wherein the at least one of the reconfigurable processors also includes a computational unit coupled to a data access unit.

16. (Original) The reconfigurable hardware system of claim 15, wherein the computational unit is supplied the data by the data access unit.

17. (Previously Presented) A method of transferring data comprising:

transferring data between a memory and a data prefetch unit in a reconfigurable processor; and

transferring the data between a computational unit and a data access unit, wherein the computational unit and the data access unit, and the data prefetch unit are configured by a program.

18. (Original) The method of claim 17, wherein the data is written to the memory, said method comprising:

transferring the data from the computational unit to the data access unit; and

writing the data to the memory from the data prefetch unit.

19. (Previously Presented) The method of claim 17, wherein the data is read from the memory, said method comprising:

transferring the data from the memory to the data prefetch unit; and

reading the data directly from the data prefetch unit to the computational unit through the data access unit.

20. (Original) The method of claim 19, wherein all the data transferred from the memory to the data prefetch unit is processed by the computational unit.

21. (Original) The method of claim 19, wherein the data is selected by the data prefetch unit based on an explicit request from the computational unit.

22. (Original) The method of claim 17, wherein the data transferred between the memory and the data prefetch unit is not a complete cache line.

23. (Original) The method of claim 17, wherein a memory controller coupled to the memory and the data prefetch unit, controls the transfer of the data between the memory and the data prefetch unit.

24. (Cancelled)

REMARKS/ARGUMENTS

Claims 1, 4-12, and 15-24 were presented for examination and are pending in this application. In an Official Office Action dated October 19, 2005, claims 1, 4-12, and 15-24 were rejected. Claim 24 is canceled without prejudice and no new claims are presently added. Claims 1, 4-12, and 15-23 remain pending. The Applicants thank the Examiner for his consideration and address the Examiner's comments concerning the claims pending in this application below.

Rejection of the Claims under 35 U.S.C. §102(e)

Claims 1, 3, 4, 7-10, and 12-18 were rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent Application Publication No. 2003/0084244 ("Paulraj"). Applicants respectfully traverse these rejections in light of the following remarks.

MPEP §2131 provides:

> "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegall Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 U.S.P.Q.2d 1051, 1053 (Fed. Cir.1987). "The identical invention must be shown in as complete detail as contained in the claim." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 U.S.P.Q.2d 1913, 1920 (Fed. Cir. 1989).

Paulraj fails to disclose each and every limitation recited in the claims. The Examiner reasons that Paulraj discloses a system having a program that reconfigures computational units, data access units, and pre-fetch units. The Applicants disagree.

The Examiner's logic in making the above assertion is faulty. Assume for argument sake (as does the Examiner) that the computational unit is the element of the Paulraj system that executes and collects performance data regarding an

application to determine an optimal memory configuration. The program operating on the Paulraj system depicted in Figure 5 of Paulraj "configures" the collection process so as to ascertain information about a specific application. In this sense the Examiner uses the term configure to state that the program executed by the Paulraj system modifies, directs, and/or controls the collection means (the computational unit) to properly assess the target application so that the memory can be optimally configured.

The Examiner then extends this argument to the data access units and pre-fetch units. While such an extension is perhaps conceivable today given the present invention, it is not, nonetheless, disclosed by Paulraj. Nor is it reasonable to conclude that such an extension would be apparent to one skilled in the art at the time of the Applicants' invention.

As the Examiner points out, Paulraj discloses creating a "configuration vector containing data relating to the optimal configuration to the necessary instruction for programming the programmable memory module." Paulraj [0024]. Paulraj also discloses a reconfiguration module that uses the vector to configure the programmable memory module. Once the Paulraj system collects information about the target application and creates the configuration vector for optimal memory module configuration, "the configuration vector is then retrieved (step 212), used to program the FPGA module (step 214), and the application is executed with the optimal memory configuration for that application (step 216)." Paulraj [0026].

The "program" that the Examiner considers to configure the computational unit does not, according to Paulraj, "configure" the data access unit nor the pre-fetch unit. The Examiner restates that he considers the reconfiguration unit of Paulraj to be a data pre-fetch unit. The Examiner also correctly states that Paulraj discloses that the reconfiguration unit retrieves the configuration vector and sets up a programmable memory module. It is conceivable to argue that the "program" of Figure 5 of Paulraj configures the configuration vector to configure the

programmable memory module but once the vector is configured Paulraj discloses that the vector is simply retrieved and used by the reconfiguration unit to program the FPGA module. No configuration by the "program" of the reconfiguration module is even implied let alone disclosed. The Examiner expands Paulraj beyond the four corners of the document and what is literally presented so as to craft an argument for anticipation. Such a creation is not contemplated nor allowable under 35 U.S.C. § 102(e). As the rules governing anticipation are clear, the Applicants submit that Paulraj does not disclose a pre-fetch unit and a memory unit that is configured by a program as is recited in claim 1.

For at least the same aforementioned reasons, claims 11 and 17 are not anticipated by Paulraj. As Claims 4-10, 12, 15, 16, and 18-23 depend from claims 1, 11, or 17 and carry with them the limitations recited in those independent claims, claims 4-10, 12, 15, 16, and 18-23 are also not anticipated by Paulraj. The Applicants respectfully request withdrawal of the rejections and reconsideration of the claims.

In view of all of the above, the claims are now believed to be allowable and the case in condition for allowance which action is respectfully requested. Should the Examiner be of the opinion that a telephone conference would expedite the prosecution of this case, the Examiner is requested to contact Applicants' attorney at the telephone number listed below.

No fee is believed due for this submittal. However, any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

January 5, 2006

Respectfully submitted,

Michael C. Martensen, No. 46,901
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5910 Tel
(303) 899-7333 Fax

Office Action Summary	Application No.	Applicant(s)	
	10/869,200	POZNANOVIC ET AL.	
	Examiner	Art Unit	
	Shane M. Thomas	2186	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *05 January 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,4-12 and 15-23* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,4-12 and 15-23* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

Client Matter No. 80404.0033.001
EFS-Web

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Serial No. 10/869,200 Application of: Daniel Poznanovic, David E. Caliga, and Jeffrey Hammes Filed: June 16, 2004 Attorney Docket No. SRC028 For: SYSTEM AND METHOD OF ENHANCING EFFICIENCY AND UTILIZATION OF MEMORY BANDWIDTH IN RECONFIGURABLE HARDWARE	Confirmation No.: 5929 Art Unit: 2186 Examiner: THOMAS, Shane M. Customer No.: **25235**

AMENDMENT AND RESPONSE PURSUANT TO FINAL OFFICE ACTION DATED MARCH 23, 2006

MAIL STOP AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the final office communication mailed March 23, 2006, please amend the above-identified application as follows:

Amendments to the Claims are reflected in the listing of claims which begins on page 2 of this paper.

Remarks/Arguments begin on page 6 of this paper.

\\\CS - 80404/0033 - 80526 v1

Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims:

1. (Currently Amended) A reconfigurable processor that instantiates an algorithm as hardware comprising:

a first memory having a first characteristic memory bandwidth and/or memory utilization; and

a data prefetch unit coupled to the first memory, wherein the data prefetch unit retrieves only computational data required by the algorithm from a second memory of second characteristic memory bandwidth and/or memory utilization and places the retrieved computational data in the first memory wherein the data prefetch unit operates independent of and in parallel with logic blocks using the computational data, and wherein at least the first memory and data prefetch unit are configured ~~by a program~~ to conform to needs of the algorithm, and the data prefetch unit is configured to match format and location of data in the second memory.

2. (Cancelled)

3. (Cancelled)

4. (Currently Amended) The reconfigurable processor of claim 1, wherein the data prefetch unit is coupled to a memory controller that controls the transfer of the data between the second memory and the data prefetch unit and transmits only portions of data desired by the data prefetch unit and discards other portions of data prior to transmission of the data to the data prefetch unit.

5. (Previously Presented) The reconfigurable processor of claim 1, wherein the data prefetch unit receives processed data from on-processor memory and writes the processed data to an external off-processor memory.

6. (Original) The reconfigurable processor of claim 1, wherein the data prefetch unit comprises at least one register from the reconfigurable processor.

7. (Original) The reconfigurable processor of claim 1, wherein the data prefetch unit is disassembled when another program is executed on the reconfigurable processor.

8. (Previously Presented) The reconfigurable processor of claim 1 wherein said second memory comprises a processor memory and said data prefetch unit is operative to retrieve data from the processor memory.

9. (Original) The reconfigurable processor of claim 8 wherein said processor memory is a microprocessor memory.

10. (Original) The reconfigurable processor of claim 8 wherein said processor memory is a reconfigurable processor memory.

11. (Currently Amended) A reconfigurable hardware system, comprising:

a common memory; and

one or more reconfigurable processors that can instantiate an algorithm as hardware coupled to the common memory, wherein at least one of the reconfigurable processors includes a data prefetch unit to read and write only data required for computations by the algorithm between the data prefetch unit and the common memory wherein the data prefetch unit operates independent

of and in parallel with logic blocks using the computational data., and wherein the data prefetch unit is configured ~~by a program executed on the system~~ to conform to needs of the algorithm and match format and location of data in the common memory.

12. (Currently Amended) The reconfigurable hardware system of claim 11, comprising a memory controller coupled to the common memory and the data prefetch unit that transmits to the prefetch unit only data desired by the data prefetch unit as required by the algorithm.

13. (Cancelled)

14. (Cancelled)

15. (Previously Presented) The reconfigurable hardware system of claim 11, wherein the at least one of the reconfigurable processors also includes a computational unit coupled to a data access unit.

16. (Original) The reconfigurable hardware system of claim 15, wherein the computational unit is supplied the data by the data access unit.

17. (Currently Amended) A method of transferring data comprising:

transferring data between a memory and a data prefetch unit in a reconfigurable processor; and

transferring the data between a computational unit and a data access unit, wherein the computational unit and the data access unit, and the data prefetch unit are configured ~~by a program~~ to conform to needs of an algorithm implemented on the computational unit and transfer only data necessary for computations by the computational unit, and wherein the prefetch unit operates independent of and in parallel with the computational unit.

18. (Original) The method of claim 17, wherein the data is written to the memory, said method comprising:

transferring the data from the computational unit to the data access unit; and

writing the data to the memory from the data prefetch unit.

19. (Currently Amended) The method of claim 17, wherein the data is read from the memory, said method comprising:

transferring only the data desired by the data prefetch unit as required by the computational unit from the memory to the data prefetch unit; and

reading the data directly from the data prefetch unit to the computational unit through the data access unit.

20. (Original) The method of claim 19, wherein all the data transferred from the memory to the data prefetch unit is processed by the computational unit.

21. (Original) The method of claim 19, wherein the data is selected by the data prefetch unit based on an explicit request from the computational unit.

22. (Original) The method of claim 17, wherein the data transferred between the memory and the data prefetch unit is not a complete cache line.

23. (Original) The method of claim 17, wherein a memory controller coupled to the memory and the data prefetch unit, controls the transfer of the data between the memory and the data prefetch unit.

24. (Cancelled)

REMARKS/ARGUMENTS

Claims 1, 4-12 and 15-23 were presented for examination and are pending in this application. In an Official Final Office Action dated March 23, 2006, claims 1, 4-12 and 15-23 were rejected. The Applicants thank the Examiner for his consideration and address the Examiner's comments concerning the claims pending in this application below.

Rejection of the Claims under 35 U.S.C. §102(e)

Claims 1, 4-12 and 15-23 were rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent Application Publication No. 2003/0084244 ("Paulraj"). In light of the aforementioned amendments, the Applicants traverse these rejections and request reconsideration. Independent claims 1, 11 and 17 have been amended to further describe the nature of the data retrieved by the prefetch unit. Support for the amendments can be found in the specification beginning generally at paragraph [0055] and continuing to paragraph [0064]. Paulraj discloses a system for cache optimization that configures a computational unit for a particular application. The Applicants' invention claims a system having a prefetch unit that retrieves computational data needed to complete the algorithm instantiated on the reconfigurable processor during processing. The retrieval of this data is done such that only data necessary for computations by the computational unit is accomplished in a manner so that the prefetch unit operates independent of and in parallel with the computational unit.

In view of all of the above, the claims are now believed to be allowable and the case in condition for allowance which action is respectfully requested. Should the Examiner be of the opinion that a telephone conference would expedite the prosecution of this case, the

Examiner is requested to contact Applicants' attorney at the telephone number listed below.

No fee is believed due for this submittal. However, any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

Respectfully submitted,

May 16, 2006

Michael C. Martensen, No. 46,901
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5910 Tel
(303) 899-7333 Fax

Advisory Action Before the Filing of an Appeal Brief	Application No. 10/869,200	Applicant(s) POZNANOVIC ET AL.	
	Examiner Shane M. Thomas	Art Unit 2186	

--The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

THE REPLY FILED 16 May 2006 FAILS TO PLACE THIS APPLICATION IN CONDITION FOR ALLOWANCE.

1. ☒ The reply was filed after a final rejection, but prior to or on the same day as filing a Notice of Appeal. To avoid abandonment of this application, applicant must timely file one of the following replies: (1) an amendment, affidavit, or other evidence, which places the application in condition for allowance; (2) a Notice of Appeal (with appeal fee) in compliance with 37 CFR 41.31; or (3) a Request for Continued Examination (RCE) in compliance with 37 CFR 1.114. The reply must be filed within one of the following time periods:
   a) ☒ The period for reply expires 3 months from the mailing date of the final rejection.
   b) ☐ The period for reply expires on: (1) the mailing date of this Advisory Action, or (2) the date set forth in the final rejection, whichever is later. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of the final rejection.

   Examiner Note: If box 1 is checked, check either box (a) or (b). ONLY CHECK BOX (b) WHEN THE FIRST REPLY WAS FILED WITHIN TWO MONTHS OF THE FINAL REJECTION. See MPEP 706.07(f).

Extensions of time may be obtained under 37 CFR 1.136(a). The date on which the petition under 37 CFR 1.136(a) and the appropriate extension fee have been filed is the date for purposes of determining the period of extension and the corresponding amount of the fee. The appropriate extension fee under 37 CFR 1.17(a) is calculated from: (1) the expiration date of the shortened statutory period for reply originally set in the final Office action; or (2) as set forth in (b) above, if checked. Any reply received by the Office later than three months after the mailing date of the final rejection, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

NOTICE OF APPEAL

2. ☐ The Notice of Appeal was filed on _____. A brief in compliance with 37 CFR 41.37 must be filed within two months of the date of filing the Notice of Appeal (37 CFR 41.37(a)), or any extension thereof (37 CFR 41.37(e)), to avoid dismissal of the appeal. Since a Notice of Appeal has been filed, any reply must be filed within the time period set forth in 37 CFR 41.37(a).

AMENDMENTS

3. ☒ The proposed amendment(s) filed after a final rejection, but prior to the date of filing a brief, will not be entered because
   (a) ☒ They raise new issues that would require further consideration and/or search (see NOTE below);
   (b) ☐ They raise the issue of new matter (see NOTE below);
   (c) ☐ They are not deemed to place the application in better form for appeal by materially reducing or simplifying the issues for appeal; and/or
   (d) ☐ They present additional claims without canceling a corresponding number of finally rejected claims.

   NOTE: *See Continuation Sheet*. (See 37 CFR 1.116 and 41.33(a)).
4. ☐ The amendments are not in compliance with 37 CFR 1.121. See attached Notice of Non-Compliant Amendment (PTOL-324).
5. ☐ Applicant's reply has overcome the following rejection(s): _____.
6. ☐ Newly proposed or amended claim(s) _____ would be allowable if submitted in a separate, timely filed amendment canceling the non-allowable claim(s).
7. ☒ For purposes of appeal, the proposed amendment(s): a) ☒ will not be entered, or b) ☐ will be entered and an explanation of how the new or amended claims would be rejected is provided below or appended.
   The status of the claim(s) is (or will be) as follows:
   Claim(s) allowed: _____.
   Claim(s) objected to: _____.
   Claim(s) rejected: *1,4-12 and 15-25*.
   Claim(s) withdrawn from consideration: _____.

AFFIDAVIT OR OTHER EVIDENCE

8. ☐ The affidavit or other evidence filed after a final action, but before or on the date of filing a Notice of Appeal will not be entered because applicant failed to provide a showing of good and sufficient reasons why the affidavit or other evidence is necessary and was not earlier presented. See 37 CFR 1.116(e).
9. ☐ The affidavit or other evidence filed after the date of filing a Notice of Appeal, but prior to the date of filing a brief, will not be entered because the affidavit or other evidence failed to overcome all rejections under appeal and/or appellant fails to provide a showing a good and sufficient reasons why it is necessary and was not earlier presented. See 37 CFR 41.33(d)(1).
10. ☐ The affidavit or other evidence is entered. An explanation of the status of the claims after entry is below or attached.

REQUEST FOR RECONSIDERATION/OTHER

11. ☐ The request for reconsideration has been considered but does NOT place the application in condition for allowance because: _____.
12. ☐ Note the attached Information Disclosure Statement(s). (PTO/SB/08 or PTO-1449) Paper No(s). _____
13. ☐ Other: _____.

Continuation Sheet (PTO-303) **Application No. 10/869,200**

Continuation of 3. NOTE: Independent claims 1,11, and 17, all contain new limitations that were not previously considered by the Examiner; thus, a further search and additional consideration is required..

HONG CHONG KIM
PRIMARY EXAMINER

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

NOTICE OF ALLOWANCE AND FEE(S) DUE

25235 7590 07/26/2006

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

EXAMINER	
THOMAS, SHANE M	
ART UNIT	PAPER NUMBER
2186	

DATE MAILED: 07/26/2006

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/869,200	06/16/2004	Daniel Poznanovic	SRC028	5929

TITLE OF INVENTION: SYSTEM AND METHOD OF ENHANCING EFFICIENCY AND UTILIZATION OF MEMORY BANDWIDTH IN RECONFIGURABLE HARDWARE

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	$1400	$300	$0	$1700	10/26/2006

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE DOES NOT REFLECT A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE IN THIS APPLICATION. IF AN ISSUE FEE HAS PREVIOUSLY BEEN PAID IN THIS APPLICATION (AS SHOWN ABOVE), THE RETURN OF PART B OF THIS FORM WILL BE CONSIDERED A REQUEST TO REAPPLY THE PREVIOUSLY PAID ISSUE FEE TOWARD THE ISSUE FEE NOW DUE.

HOW TO REPLY TO THIS NOTICE:

I. Review the SMALL ENTITY status shown above.

If the SMALL ENTITY is shown as YES, verify your current SMALL ENTITY status:

A. If the status is the same, pay the TOTAL FEE(S) DUE shown above.

B. If the status above is to be removed, check box 5b on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and twice the amount of the ISSUE FEE shown above, or

If the SMALL ENTITY is shown as NO:

A. Pay TOTAL FEE(S) DUE shown above, or

B. If applicant claimed SMALL ENTITY status before, or is now claiming SMALL ENTITY status, check box 5a on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and 1/2 the ISSUE FEE shown above.

II. PART B - FEE(S) TRANSMITTAL, or its equivalent, must be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed and an extra copy of the form should be submitted. If an equivalent of Part B is filed, a request to reapply a previously paid issue fee must be clearly made, and delays in processing may occur due to the difficulty in recognizing the paper as an equivalent of Part B.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Mail Stop ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

PTOL-85 (Rev. 07/06) Approved for use through 04/30/2007.

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), to: Mail **Mail Stop ISSUE FEE**
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450
or Fax (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

25235 7590 07/26/2006

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission
I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

(Depositor's name)
(Signature)
(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/869,200	06/16/2004	Daniel Poznanovic	SRC028	5929

TITLE OF INVENTION: SYSTEM AND METHOD OF ENHANCING EFFICIENCY AND UTILIZATION OF MEMORY BANDWIDTH IN RECONFIGURABLE HARDWARE

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	NO	$1400	$300	$0	$1700	10/26/2006

EXAMINER	ART UNIT	CLASS-SUBCLASS
THOMAS, SHANE M	2186	711-137000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).
☐ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.
☐ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list
(1) the names of up to 3 registered patent attorneys or agents OR, alternatively,
(2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.
1 _____
2 _____
3 _____

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE (B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent) : ☐ Individual ☐ Corporation or other private group entity ☐ Government

4a. The following fee(s) are submitted:
☐ Issue Fee
☐ Publication Fee (No small entity discount permitted)
☐ Advance Order - # of Copies _____

4b. Payment of Fee(s): **(Please first reapply any previously paid issue fee shown above)**
☐ A check is enclosed.
☐ Payment by credit card. Form PTO-2038 is attached.
☐ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment, to Deposit Account Number _____ (enclose an extra copy of this form).

5. **Change in Entity Status** (from status indicated above)
☐ a. Applicant claims SMALL ENTITY status. See 37 CFR 1.27.
☐ b. Applicant is no longer claiming SMALL ENTITY status. See 37 CFR 1.27(g)(2).

NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Authorized Signature _____ Date _____

Typed or printed name _____ Registration No. _____

This collection of information is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/869,200	06/16/2004	Daniel Poznanovic	SRC028	5929

25235 7590 07/26/2006

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

EXAMINER	
THOMAS, SHANE M	
ART UNIT	PAPER NUMBER
2186	

DATE MAILED: 07/26/2006

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment to date is 0 day(s). If the issue fee is paid on the date that is three months after the mailing date of this notice and the patent issues on the Tuesday before the date that is 28 weeks (six and a half months) after the mailing date of this notice, the Patent Term Adjustment will be 0 day(s).

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at 1-(888)-786-0101 or (571)-272-4200.

PTOL-85 (Rev. 07/06) Approved for use through 04/30/2007.

Notice of Allowability	Application No.	Applicant(s)	
	10/869,200	POZNANOVIC ET AL.	
	Examiner	Art Unit	
	Shane M. Thomas	2186	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *RCE / Amendment filed 6/15/2006*.

2. ☒ The allowed claim(s) is/are *1,4-12,15-23 (renumbered 1-19)*.

3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☒ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO-1449 or PTO/SB/08), Paper No./Mail Date _____
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application (PTO-152)
6. ☐ Interview Summary (PTO-413), Paper No./Mail Date _____ .
7. ☐ Examiner's Amendment/Comment
8. ☒ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

REASONS FOR ALLOWANCE

Claims 1,4-12, and 15-23 are allowable over the prior art of record.

The following is an examiner's statement of reasons for allowance:

As per independent claims 1,11, and 17, the prior art of record does not teach or suggest, either alone or in combination, the every limitation of each claim. Specifically the prior art of record does not teach in combination a reconfigurable processor with a data prefetch unit only fetching computational data required by an algorithm in addition to a first memory and the prefetch unit being configurable to conform to the requirements (needs) of a particular algorithm where the data prefetch unit is configured to match format and location of the in the second memory (claim 1). Further regarding claims 11 and 17, the prior art of record does not teach the prefetch unit operating independent and in parallel with the logic blocks that are using computational data with the data prefetch unit only transferring data necessary for computations. Further regarding claim 17, the prior art of record does not specifically teach a computation unit, prefetch unit, and data access unit all being configurable in order to conform to the needs of an algorithm implemented on the computational unit.

Gibson et al. (U.S. Patent No. 6,507,898) teaches a reconfigurable cache controller but does not teach each limitation of the independent claims of Applicant.

Howard et al. (U.S. Patent Application Publication No. 2005/0044327) teaches a reconfigurable processor that may be reconfigured based on the algorithm being run (¶52 and ¶90).

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Shane M. Thomas whose telephone number is (571) 272-4188. The examiner can normally be reached on M-F 8:30 - 5:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matt M. Kim can be reached on (571) 272-4182. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Shane M. Thomas

MATTHEW KIM
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

Supplemental Notice of Allowability	Application No. 10/869,200	Applicant(s) POZNANOVIC ET AL.	
	Examiner Shane M. Thomas	Art Unit 2186	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *IDS filed 10/6/2006, after Notice of Allowance*.

2. ☒ The allowed claim(s) is/are *1,4-12 and 15-23 (renumbered 1-19)*.

3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☐ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☒ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date 10/06/2006
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☐ Interview Summary (PTO-413), Paper No./Mail Date _____ .
7. ☐ Examiner's Amendment/Comment
8. ☐ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

PIERRE BATAILLE
PRIMARY EXAMINER 10/12/06

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	ISSUE DATE	PATENT NO.	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/869,200	12/12/2006	7149867	SRC028	5929

25235 7590 11/22/2006

HOGAN & HARTSON LLP
ONE TABOR CENTER, SUITE 1500
1200 SEVENTEENTH ST
DENVER, CO 80202

ISSUE NOTIFICATION

The projected patent number and issue date are specified above.

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment is 0 day(s). Any patent to issue from the above-identified application will include an indication of the adjustment on the front page.

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at (571)-272-4200.

APPLICANT(s) (Please see PAIR WEB site http://pair.uspto.gov for additional applicants):

Daniel Poznanovic, Colorado Springs, CO;
David E. Caliga, Colorado Springs, CO;
Jeffrey Hammes, Colorado Springs, CO;

IR103 (Rev. 11/05)

Attorney Docket No.: SRC028
Client/Matter No: 80404.0033.001
EFS-Web

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Name of Patentee:

Daniel Poznanovic, David E. Caliga, and Jeffrey Hammes

Patent No.: 7,149,867

Issued: Dec. 12, 2006

Title: SYSTEM AND METHOD OF ENHANCING EFFICIENCY AND UTILIZATION OF MEMORY BANDWIDTH IN RECONFIGURABLE HARDWARE

ATTENTION: Certificate of Corrections Branch
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

REQUEST FOR CERTIFICATE OF CORRECTION OF PATENT FOR PTO Mistake (37 C.F.R. 1.322(a))

DEAR SIR:

An error appears in this patent. The error is a formatting mistake by the PTO. The error occurred in good faith. Correction thereof does not involve such changes in the patent as would constitute new matter or would require reexamination.

Attached hereto in duplicate is form PTO-1050, with at least one copy being suitable for printing.

Please send the certificate to:

William J. Kubida
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, CO 80202

Although no fee is believed due, any fee deficiency associated with this transmittal may be charged to Deposit Account 50-1123.

Respectfully submitted,

Date: 16 March 2007

BY: _____

William J. Kubida, Reg. No. 29,664
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5909 Tel
(303) 899-7333 Fax

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 7,149,867 B2
APPLICATION NO. : 10/869200
DATED : December 12, 2006
INVENTOR(S) : Daniel Poznanovic, David E. Caliga and Jeffrey Hammes

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 12, line 43, insert --first-- after "coupled to the"

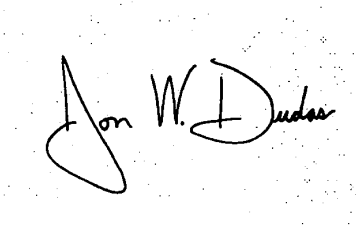
Column 12, line 57, insert --second-- after "between the"

Column 13, line 6, "a" should be --the--

Column 14, line 4, the second occurrence of "the" should be --a--.

Signed and Sealed this

Twenty-fourth Day of April, 2007



JON W. DUDAS
Director of the United States Patent and Trademark Office

SRC Labs, LLC v. Microsoft Corp.
Joint Claim Construction

EXHIBIT Q - JOINT CLAIM CHART

Claim Language (Disputed Terms in **Bold**) `110 Patent `152 Patent	Plaintiff's Proposed Construction and Evidence in Support	Defendant's Proposed Construction and Evidence in Support
"**memory bank**" Found in claim numbers: '152 Patent: 1, 3, 11 '110 Patent: 1, 3, 11	A physical module that is part of the memory subsystem having a range of memory addresses assigned to it **Intrinsic Evidence:**[1] '152 patent, Fig. 1; Fig. 3, col. 3:10-17, col. 3:47-67, col. 4:1-6 ("The computer architecture 100 comprises a multiprocessor system employing uniform memory access across common shared memory with one or more MAPs 112 located in the memory subsystem, or memory space. As previously described, each MAP 112 contains at least one relatively large FPGA 134 that is used as a reconfigurable functional unit."). '152 patent, prosecution history at MS_SRC-SRMT_0471095 ("wherein the data processor's memory bank contains memory algorithm processors"); MS_SRC-SRMT_0471099 ("that is memory algorithm processors 112 are a part of memory bank 120"). **Extrinsic Evidence**:	a group of devices which are part of the memory subsystem and connected together for use as a memory for a data processor, and which may also be used for other purposes. **Intrinsic Evidence:** '152 patent, Fig. 1, Fig. 3, 3:46-4:7 ("[A] preferred implementation of a memory bank 120 in a MAP system computer architecture 100 of the present invention is shown for a representative one of the MAPs 112 illustrated in the preceding figure."), 4:29-51 ("By placing the MAP 112 in the memory subsystem or memory space, it can be readily accessed through the use of memory read and write commands, which allows the use of a variety of standard operating systems. In contrast, other conventional implementations propose placement of any reconfigurable logic in or near the processor. This is much less effective in a multiprocessor environment because only one processor has rapid access to it. Consequently, reconfigurable logic must be placed by every processor in a multiprocessor system, which increases the

[1] The '152 and '110 patent specifications contain identical disclosures so to save space Plaintiffs' citations are to the '152 patent specification only.

	U.S. Patent No. 5,367,494 col. 1:61-2:4 ("The multiple concurrent accesses are accomplished by providing a plurality of memory banks wherein each of the memory banks is independently and distinctly addressed and controlled."). Microsoft Press Computer Dictionary 3rd Ed. 1997 ("Memory Bank: the physical location on a motherboard where a memory module can be inserted.") IEEE Standard Glossary of Computer Hardware Terminology, 1995 ("bank: (3) A contiguous section of addressable memory."). Declaration of Dr. El-Ghazawi at ¶ 25. Deposition of Dr. El-Ghazawi at 38-42 Deposition of Dr. Houh at 47-48, 53-67. Declarations of Dr. Houh. Deposition of Derek Chiou at 265-266.	overall system cost. In addition, MAP 112 can access the memory array 130 itself, referred to as Direct Memory Access ("DMA"), allowing it to execute tasks independently and asynchronously of the processor. In comparison, were it [] placed near the processor, it would have to compete with the processors for system routing resources in order to access memory, which deleteriously impacts processor performance. Because MAP 112 has DMA capability, (allowing it to write to memory), and because it receives its operands via writes to memory, it is possible to allow a MAP 112 to feed results to another MAP 112. This is a very powerful feature that allows for very extensive pipelining and parallelizing of large tasks, which permits them to complete faster."); *see also* 3:10-17. '110 patent, Fig. 1, Fig. 3, 3:17-23, 3:54-4:14. '152 patent file history at MS_SRC-SRMT_0471095-1101 (September 1, 1999 Response to June 3, 1999 Office Action mailed at 9-15) (discussing the prior art Cloutier patent that was cited by the Examiner). **Extrinsic Evidence:** The New IEEE Standard Dictionary of Electrical and Electronics Terms (1993) ("IEEE Dictionary") (bank: An aggregation of similar devices (for example, transformers, lamps, etc.) connected together and used in cooperation.)

		Microsoft Press Computer Dictionary (3rd ed. 1997) ("Microsoft Dictionary") (bank: 1. Any group of similar electrical devices connected together for use as a single device. For example, transistors may be connected in a row/column array inside a chip to form memory, or several memory chips may be connected together to form a memory module such as a SIMM. See the illustration. *See also* SIMM. 2. A section of memory, usually of a size convenient for a CPU to address. For example, an 8-bit processor can address 65,536 bytes of memory; therefore, a 64-kilobyte (64-KB) memory bank is the largest that the processor can address at once. To address another 64-KB bank of memory requires circuitry that fools the CPU into looking at a separate block of memory. *See also* bank switching, page (definition 2). U.S. Patent No. 6,052,773 to DeHon U.S. Patent No. 5,677,864 to Chung Tarek El-Ghazawi Deposition Tr. (10.18.18) at 32:2-33:17, 35:9-37:2, 37:21-38:12, 38:18-41:9, 42:7-43:10. Henry Houh Deposition Tr. (10.17.18) at 45:23-16, 53:22-54:16 Declaration of Henry Houh ¶¶ 126-135.

"**memory algorithm processor**" / "**reconfigurable memory algorithm processor**" Found in claim numbers: '152 Patent: 1-5, 7, 11, 12, 15, 21 '110 Patent: 1-5, 7, 11, 12, 15, 21	Memory-resident reconfigurable logic configurable to perform an identified algorithm **Intrinsic Evidence:** '152 patent: Abstract; Figs. 2-3; col. 1:30-59 ("Recently, several groups have begun to experiment with creating a processor out of circuits that are electrically reconfigurable."); col. 1:66-2:5 ("Particularly disclosed herein is the utilization of one or more FPGAs to perform user defined algorithms in conjunction with, and tightly coupled to, a microprocessor. More particularly, in a multiprocessor computer system, the FPGAs are globally accessible by all of the system processors for the purpose of executing user definable algorithms."); col. 2:40-50 ("The memory algorithm processor is configurable to perform at least one identified algorithm on an operand received from a write operation to the associated one of the second plurality of memory arrays."); col. 7:7-9 ("2. The improvement of claim 1 wherein each of said plurality of memory algorithm processors comprises a field programmable gate array."); col. 8:42-45 ("18. The multiprocessor computer system of claim 11 wherein each of said plurality of memory algorithm processors comprises a field programmable gate array."). '152 patent, prosecution history at MS_SRC-SRMT_0471096 ("Each of the MAPs 112 comprises a user-configurable hardware element), MS_SRC-SRMT0471097 ("each individual one	A memory-resident reconfigurable processing device that is viewed by each data processor of the system merely as part of the data processor's memory, that can be used by a data processor of the system merely by addressing a portion of that memory, and that is not in or near a data processor of the system. **Intrinsic Evidence:** '152 patent, Fig. 1, 3:46-4:7 ("[A] preferred implementation of a memory bank 120 in a MAP system computer architecture 100 of the present invention is shown for a representative one of the MAPs 112 illustrated in the preceding figure."); *see also* 3:10-17. '152 patent file history at MS_SRC-SRMT_0471095-1101 (September 1, 1999 Response to June 3, 1999 Office Action at 9-15) (discussing the prior art Cloutier patent that was cited by the Examiner). **Extrinsic Evidence:** Tarek El-Ghazawi Deposition Tr. (10.18.18) at 36:2-37:2, 102:24-103:12. Henry Houh Deposition Tr. (10.17.18) at 76:8-77:23 Declaration of Henry Houh ¶¶ 136-143.

	of the FIG. 2 MAPs 112 includes a USER FPGA (Field Programmable Gate Array) 134, wherein each USER FPGA 134 operates as a reconfigurable functional unit; i.e., operates as a reconfigurable hardware machine. The plurality of USER FPGAs 134 (i.e., one for each of FIG. 2's MAPs 112) perform a plurality of user-defined algorithms with, and tightly coupled to, PROCESSORS 108."). **Extrinsic Evidence:** U.S. Patent No. 7,149,867. Deposition of Dr. Houh at 74-77.	
"**means connecting** said plurality of memory algorithm processors to said data bus and to said address bus such that said plurality of memory algorithm processors are individually memory addressable by said at least one data processor as said at least one data processor executes said application program" **(152 Patent)** Found in claim numbers: '152 Patent: 1 '110 Patent: 1	Not a means plus function. In the alternative, if the court disagrees: **Function**: allow the plurality of memory algorithm processors to be accessible using normal memory access protocols by the data processor **Structure**: Placing the plurality of memory algorithm processors in the memory subsystem. **Intrinsic Evidence:** '152 patent: Figs. 1-4; col. 2:16-50 ("The computer comprises a memory algorithm processor associated with at least one of the second plurality of memory arrays and coupled to the data and address bus thereof. The memory algorithm processor is configurable to perform at least one identified algorithm on an operand	Governed by 35 U.S.C. § 112(6). **Function:** "connecting said plurality of memory algorithm processors to said data bus and to said address bus such that said plurality of memory algorithm processors are individually memory addressable by said at least one data processor as said at least one data processor executes said application program" **Structure:** No disclosed corresponding structure. Indefinite **Intrinsic Evidence:** None. **Extrinsic Evidence:**

	received from a write operation to the associated one of the second plurality of memory arrays."); col. 4:1-7; col. 4:29-23; 5:27-6:26 ("With reference additionally now to FIG. 4, a block diagram of the MAP control block 132 is shown in greater detail. The control block 132 is coupled to receive a number of command bits (for example, 17) from the address bus 128 at a command decoder 150."); col. 6:52-55. **Extrinsic Evidence:** Declaration and rebuttal declaration of Dr. El-Ghazawi Rebuttal Declaration of Dr. El-Ghazawi Declaration of Dr. Homayoun. Deposition of Dr. El-Ghazawi at 36-38, 46-59, 65-66. Declarations of Henry Houh Deposition of Henry Houh at 42-44, 77-85, 90-101, 105-108, 112-116. Deposition of Derek Chiou at 267-268.	Tarek El-Ghazawi Deposition Tr. (10.18.18) at 44:12-24, 45:22 –46:11, 50:3-51:19, 52:14-53:22, 58:18-59:13, 65:11-66:14 Declaration of Henry Houh ¶¶ 144-151.
Governed by 35 U.S.C. § 112(6): "**means coupling** said plurality of individual memory algorithm	Not a means plus function. In the alternative, if the court disagrees: **Function**: allows the memory algorithm processors to be individually memory addressable	Governed by 35 U.S.C. § 112(6). **Function:** "coupling said plurality of individual memory algorithm processors to said data bus and to said address bus"

processors to said data bus and to said address bus" **(110 Patent)** Found in claim numbers: '152 Patent: 11 '110 Patent: 11	**Structure**: connecting the plurality of memory algorithm processors to the data bus and address bus. **Intrinsic Evidence:** '152 patent: Figs. 1-4; col. 2:16-50 ("The computer comprises a memory algorithm processor associated with at least one of the second plurality of memory arrays and coupled to the data and address bus thereof. The memory algorithm processor is configurable to perform at least one identified algorithm on an operand received from a write operation to the associated one of the second plurality of memory arrays."); col. 4:1-7; col. 4:29-23; 5:27-6:26 ("With reference additionally now to FIG. 4, a block diagram of the MAP control block 132 is shown in greater detail. The control block 132 is coupled to receive a number of command bits (for example, 17) from the address bus 128 at a command decoder 150."); col. 6:52-55. **Extrinsic Evidence:** Declaration and rebuttal declaration of Dr. El-Ghazawi Rebuttal Declaration of Dr. El-Ghazawi Declaration of Dr. Homayoun. Deposition of Dr. El-Ghazawi at 36-38, 46-59, 65-66.	**Structure:** No disclosed corresponding structure. Indefinite **Intrinsic Evidence:** None. **Extrinsic Evidence:** Tarek El-Ghazawi Deposition Tr. (10.18.18) at 44:12-24, 45:22 –46:11, 50:3-51:19, 52:14-53:22, 58:18-59:13, 65:11-66:14 Declaration of Henry Houh ¶¶ 152-158.

	Declarations of Henry Houh Deposition of Henry Houh at 42-44, 77-85, 90-101, 105-108, 112-116. Deposition of Derek Chiou at 267-268.	

Claim Language (Disputed Terms in **Bold**) `324 Patent `800 Patent	Plaintiff's Proposed Construction and Evidence in Support	Defendant's Proposed Construction and Evidence in Support
"**systolic**" / "**systolically**" **(324 Patent)** Found in claim numbers: '324 Patent: 1	This term has its plain and ordinary meaning and need not be construed. In the alternative, this term may be construed as: rhythmically computing and passing data in a transport triggered manner **Intrinsic Evidence:** '324 patent: Fig. 2; Fig. 4; Figs. 7A-7D; Fig. 8A; col. 6:1-30 ("With reference additionally now to FIG. 4A, a simplified logic flowchart is provided illustrating a conventional sequential processing operation 400 in which nested Loops A (first loop 402) and B (second loop 404) are alternately active on different phases of the process…In contrast to the sequential processing operation 400 (FIG. 4A) the solution to the problem of most effectively utilizing available resources is to have an application evaluate a problem in a data flow	The characteristic of rhythmically computing and passing data directly between processing elements "without a program counter or clock that drives the movement of data" and operating in a manner that is "transport triggered, *i.e.*, by the arrival of a data object" **Intrinsic Evidence:** '324 patent, Fig. 8A, 8B, 8:27-45 ("systolic wavefront operation[s]"), ("systolic processing in the process 800 can pass previously computed data down within a column (e.g. one of columns 802, 804 and 806) as to subsequent columns as well (e.g. from column 802 to 804; from column 804 to 806 etc.)"); *see also* 8:37-39. '324 patent file history at MS_SRC-SRMT_0473086-3087, 3089-3090, 3093-3094, 3098-3100 (November 13, 2006 Response to August 17, 2006 Office Action at 2-3, 5-6, 9-10, 14-16)

	sense. That is, it will 'pass' a subsequent dimension of a given problem through the first loop 412 of logic concurrently with the previous dimension of data being processed through the second loop 414. In practice, a 'dimension' of data can be: multiple vectors of a problem, multiple planes of a problem, multiple time steps in a problem and so forth."), col. 8:27-40. '324 patent, prosecution history at MS_SRC-SRMT0473096-99 ("Similarly the term systolic computation is derived from continual and pulsating pumping of the human heart. In computer architecture a systolic array is an arrangement of data processing units similar to a central processing unit but without a program counter or clock that drives the movement of data. That is because the operation of the systolic array is transport triggered, i.e. by the arrival of a data object. … define a Systolic system as a 'network of processors which rhythmically compute and pass data through the system.'"). David Caliga and David Peter Barker, "Delivering Acceleration: The Potential for Increased HPC Application Performance Using Reconfigurable Logic," Nov. 2001, ACM/IEEE SC 2001 Conference, pp. 20. **Extrinsic Evidence:**	("Similarly the term systolic computation is derived from continual and pulsating pumping of the human heart. In computer architecture a systolic array is an arrangement of data processing units similar to a central processing unit but without a program counter or clock that drives the movement of data. That is because the operation of the systolic array is transport triggered, i.e., by the arrival of a data object. Data flows across the array between functional units, usually with different data flowing in different directions. David J. Evans in his work, Systolic algorithms. Systolic algorithms, number 3 in Topics in Computer Mathematics, Gordon and Breach, 1991 define a Systolic system as a 'network of processors which rhythmically compute an [sic] pass data through the system' Thus in the Applicant's invention Systolic implementation will connect computational loops such that data from one compute loop will be passed as input data to a concurrently executing compute loop."), ("Data flows across the array between functional units, usually with different data flowing in different directions…Thus in the Applicant's invention Systolic implementation will connect computational loops such that data from one compute loop will be passed as input data to a concurrently executing compute loop. In the Applicant's invention data computed by computation units or groups of functional units flows seamlessly and concurrently with data being computed by other groups of functional units.").

	Jean-Luc Gaudiot, "Data-Driven Multicomputers in Digital Signal Processing," 1987, IEEE, Proceedings of the IEEE, Vol. 75, No. 9, pp. 1220-1234. A Dictionary of Computer Science, Oxford University Press (7th ed. 2016) (Dataflow machine: a computer in which the primitive operations are triggered by the availability of inputs or operands... in a dataflow machine there is a flow of data values from operations that produce those values to operations that 'consume' those values as operands."). A Dictionary of Computer Science, Oxford University Press (7th ed. 2016) (systolic array: Data is pulsed from the memory into processor(s) at an array boundary and then pulses through the array…). IEEE std 610.10-994, IEEE Standard Glossary of Computer Hardware Terminology, 1995, pg. 7 ("3.106 asynchronous computer. A computer in which each event or operation is performed upon receipt of a signal generated by the completion of a previous event or operation …."). Kung, H.T., "Why Systolic Architectures?", *IEEE Computer*, Jan. 1982, 37-46 Kung, et al., "Systolic arrays for VLSI," SIAM, Sparse Matrix Proceedings, 1978 1-29.	**Extrinsic Evidence:** Kung, et al., "Systolic arrays for VLSI," SIAM, Sparse Matrix Proceedings, 1978 1-29. Gaudiot, Jean-Luc, "Data-Driven Multicomputers in Digital Signal Processing," IEEE, Proceedings of the IEEE, vol. 75, no. 9, 1987, 1220-1234. Kung, H.T., "Why Systolic Architectures?", *IEEE Computer*, Jan. 1982, 37-46. U.S. Patent No. 5,956,518 to DeHon et al. U.S. Patent No. 5,274,832 to Khan Declaration of Henry Houh ¶¶ 71-75; 205-214.

	Declaration of Dr. El-Ghazawi at ¶¶ 14 and 16. Deposition of Henry Houh at 163:5-14.	
"**pass computed data seamlessly**" Found in claim numbers: '324 Patent: 1 '800 Patent: 1	communicating the computed data over the reconfigurable routing resources **Intrinsic Evidence:** '324 patent: Fig. 4; col. 6:1-30 ("With reference additionally now to FIG. 4A, a simplified logic flowchart is provided illustrating a conventional sequential processing operation 400 in which nested Loops A (first loop 402) and B (second loop 404) are alternately active on different phases of the process…In contrast to the sequential processing operation 400 (FIG. 4A) the solution to the problem of most effectively utilizing available resources is to have an application evaluate a problem in a data flow sense. That is, it will 'pass' a subsequent dimension of a given problem through the first loop 412 of logic concurrently with the previous dimension of data being processed through the second loop 414. In practice, a 'dimension' of data can be: multiple vectors of a problem, multiple planes of a problem, multiple time steps in a problem and so forth."). '800 patent: Fig. 4; col. 5:65-6:28 ("With reference additionally now to FIG. 4A, a simplified logic flowchart is provided	To communicate computed data directly **Intrinsic Evidence:** '324 patent, 7:42-8:6 ("With reference additionally now to FIG. 7A, a process 700 for performing a representative systolic wavefront operation in the form of a reservoir simulation function is shown which utilizes the parallelism available in the adaptive processing techniques of the present invention. The process 700 includes a "k" loop 702, "j" loop 704 and "i" loop 706 as shown. With reference additionally now to FIG. 7B, the general computation of fluid flow properties in the reservoir simulation process 700 of the preceding figure are illustrated as values are communicated between a group of neighboring cells 710."), 8:27-45 ("systolic wavefront operation[s]"), ("systolic processing in the process 800 can pass previously computed data down within a column (e.g. one of columns 802, 804 and 806) as to subsequent columns as well (e.g. from column 802 to 804; from column 804 to 806 etc.)"). '324 patent file history at MS_SRC-SRMT_0473098 (November 13, 2006 Response to August 17, 2006 Office Action at 14) ("Data flows across the array between functional units, usually with different data flowing in different directions . . . Thus in the Applicant's invention Systolic

	illustrating a conventional sequential processing operation 400 in which nested Loops. A (first loop 402) and B (second loop 404) are alternately active on different phases of the process… In contrast to the sequential processing operation 400 (FIG. 4A) the solution to the problem of most effectively utilizing available resources is to have an application evaluate a problem in a data flow sense. That is, it will "pass a subsequent dimension of a given problem through the first loop 412 of logic concurrently with the previous dimension of data being processed through the second loop 414. In practice, a "dimension of data can be: multiple vectors of a problem, multiple planes of a problem, multiple time steps in a problem and so forth."). '324 patent, prosecution history at MS_SRC-SRMT0473096 ("the use of the words 'protocol independent' to impart the ability of the functional units to seamlessly pass computed data between computational loops comprised of functional unit."); MS_SRC-SRMT0473097 ("'protocol' has been replaced with an 'interconnection' between functional units that is established by reconfigurable routing resources inside each chip.""); MS_SRC-SRMT0473086-87 ("communication between other reconfigurable processors within the system would require communication protocol but	implementation will connect computational loops such that data from one compute loop will be passed as input data to a concurrently executing compute loop. In the Applicant's invention data computed by computation units or groups of functional units flows seamlessly and concurrently with data being computed by other groups of functional units."). **Extrinsic Evidence:** Declaration of Henry Houh ¶¶ 215-220.

	communication between functional units within an individual reconfigurable processor is free of such a requirement."); MS_SRC-SRMT0473098 ("Data flows across the array between functional units, usually with different data flowing in different directions. David J. Evans in his work, Systolic algorithms. Systolic algorithms, number 3 in Topics in Computer Mathematics, Gordon and Breach, 1991 define a Systolic system as a 'network of processors which rhythmically compute an pass data through the system' Thus in the Applicant's invention Systolic implementation will connect computational loops such that data from one compute loop will be passed as input data to a concurrently executing compute loop. In the Applicant's invention data computed by computation units or groups of functional units flows seamlessly and concurrently with data being computed by other groups of functional units."). David Caliga and David Peter Barker, "Delivering Acceleration: The Potential for Increased HPC Application Performance Using Reconfigurable Logic," Nov. 2001, ACM/IEEE SC 2001 Conference, pp. 7-20. **Extrinsic Evidence:** A Dictionary of Computer Science, Oxford University Press (7th ed. 2016) (Dataflow machine: a computer in which the primitive	

	operations are triggered by the availability of inputs or operands... in a dataflow machine there is a flow of data values from operations that produce those values to operations that 'consume' those values as operands."). IEEE std 610.10-994, IEEE Standard Glossary of Computer Hardware Terminology, 1995, pg. 7 ("3.106 asynchronous computer. A computer in which each event or operation is performed upon receipt of a signal generated by the completion of a previous event or operation …."). Jack B. Dennis, "Data Flow Supercomputers" 1980, Computer, 10.1109/MC.1980.1653418, pp. 52.	
"**a data driven calculation**" Found in claim numbers: '800 Patent: 1	The term "a data driven calculation" should be construed to mean: Computation triggered by the availability of input data *Note: Amazon has agreed to this construction.* **Intrinsic Evidence:** '800 patent: Fig. 2; Fig. 4; Figs. 7A-9C; col. 2:50-59; col. 4:45-49; 5:65-6:28 ("In contrast to the sequential processing operation 400 (FIG. 4A) the solution to the problem of most effectively utilizing available resources is to have an application evaluate a problem in a data flow sense. That is, it will "pass a subsequent dimension of a given problem through the first loop 412 of logic	The term "data driven" should be construed to mean: The scheduling of operations upon the availability of their operands. **Intrinsic Evidence:** None. **Extrinsic Evidence:** Hartenstein, et al., "A reconfigurable data-driven ALU for Xputers," Proceedings of the 1994 IEEE Workshop on FPGAs for CCMs, April 10-13, 1994, 139-146. Declaration of Henry Houh ¶¶ 221-223.

	concurrently with the previous dimension of data being processed through the second loop 414.”). ’800 patent, prosecution history at SRC00001634-42 (“data driven processing and calculations includes systolic processing”). ’324 patent: Fig. 2; Fig. 4; Figs. 7A-7D; Fig. 8A; col. 6:1-30 (“That is, it will ‘pass’ a subsequent dimension of a given problem through the first loop 412 of logic concurrently with the previous dimension of data being processed through the second loop 414. In practice, a ‘dimension’ of data can be: multiple vectors of a problem, multiple planes of a problem, multiple time steps in a problem and so forth.”), col. 8:27-40. ’324 patent, prosecution history at MS_SRC-SRMT0473096-99 (“Similarly the term systolic computation is derived from continual and pulsating pumping of the human heart. In computer architecture a systolic array is an arrangement of data processing units similar to a central processing unit but without a program counter or clock that drives the movement of data. That is because the operation of the systolic array is transport triggered, i.e. by the arrival of a data object. … define a Systolic system as a ‘network of processors which rhythmically compute and pass data through the system.’”).	

	David Caliga and David Peter Barker, "Delivering Acceleration: The Potential for Increased HPC Application Performance Using Reconfigurable Logic," Nov. 2001, ACM/IEEE SC 2001 Conference, pp. 7-20. Jean-Luc Gaudiot, "Data-Driven Multicomputers in Digital Signal Processing," 1987, IEEE, Proceedings of the IEEE, Vol. 75, No. 9, pp. 1220-1234. Philip C. Treleaven , David R. Brownbridge , Richard P. Hopkins, "Data-Driven and Demand-Driven Computer Architecture," 1982, ACM Computing Surveys, Vol.14, No. 1, 93-143. **Extrinsic Evidence:** Hartenstein, et al., "A Reconfigurable Data-Driven ALU for Xputers," Proc. of the 1994 IEEE Workshop on FPGAs for CCMs, April 10-13, 1994, 139-146 ("The term data sequencing derives from the fact that the sequence of data triggers the operations in the rALU, instead of von-Neumann instruction sequence." A Dictionary of Computer Science, Oxford University Press (7th ed. 2016) (Dataflow machine: a computer in which the primitive operations are triggered by the availability of inputs or operands... in a dataflow machine	

	there is a flow of data values from operations that produce those values to operations that 'consume' those values as operands."). IEEE std 610.10-994, IEEE Standard Glossary of Computer Hardware Terminology, 1995, pg. 7 ("3.106 asynchronous computer. A computer in which each event or operation is performed upon receipt of a signal generated by the completion of a previous event or operation …."). Jack B. Dennis, "Data Flow Supercomputers" 1980, Computer, 10.1109/MC.1980.1653418, pp. 52. Declaration of Dr. El-Ghazawi (Microsoft) at ¶ 14.	
"**instantiating**"/ "**instantiated**"/ "**instantiation**" Found in claim numbers: '324 Patent: 1	This term has its plain and ordinary meaning and need not be construed. In the alternative, this term may be construed as: Configuring/configured **Intrinsic Evidence:** '324 patent: Fig. 2; Fig. 4; Figs. 7A-9C; col. 2:1-5 ("Reconfigurable processors instantiate only the functional units needed to solve a particular application, and as a result, have available space to instantiate as many functional units as may be required to solve the problem up to the total capacity of the integrated circuit chips they employ."); col.	Configuring/configured/configuration, such that each configuration for each calculation is unique. **Intrinsic Evidence:** '324 patent, 2:1-5 ("Reconfigurable processors instantiate only the functional units needed to solve a particular application, and as a result, have available space to instantiate as many functional units as may be required to solve the problem up to the total capacity of the integrated circuit chips they employ.") '324 patent file history at MS_SRC-SRMT_0473097 (November 13, 2006 Response to August 17, 2006 Office Action at 13) ("A

	2:49-58 ("instantiating at least two of the functional units to perform the calculation"); '800 patent: Fig. 2; Fig. 4; Figs. 7A-9C; col. 2:1-7; col. 2:53-55. '324 patent, prosecution history at MS_SRC-SRMT047309-97 ("Instantiation is a term well known to one of ordinary skill in the art of reconfigurable processing. A reconfigurable processor is essentially a blank processor that must be configured (instantiated) to conduct a particular task. To instantiate means to create such an instance or configuration by, for example, defining one particular variation of the processor's structure."). David Caliga and David Peter Barker, "Delivering Acceleration: The Potential for Increased HPC Application Performance Using Reconfigurable Logic," Nov. 2001, ACM/IEEE SC 2001 Conference, pp. 20. **Extrinsic Evidence:** U.S. Patent No. 6,434,687 John Villasenor and Brad Hutchings, "The flexibility of configurable computing," 1997, IEEE Signal Processing Magazine, pp 67-84. Katherine Compton and Scott Hauck, "Reconfigurable Computing: A Survey of	reconfigurable processor is essentially a blank processor that must be configured (instantiated) to conduct a particular task. To instantiate means to create such an instance or configuration by, for example, defining one particular variation of the processor's structure. This involves allocation of a structure with the types specified by a template and the initialization of instance variables with either default values or those provided by a constructor function. In reconfigurable computing a hard macro library file is typically inserted into a design file. A design may include multiple instances of the same library file with each possessing a unique name. Thus in the Applicant's invention the reconfigurable processor is instantiated and designed to perform the defined calculation. Each instantiation for each calculation is unique."). **Extrinsic Evidence:** Declaration of Henry Houh ¶¶ 224-227. Derek Chiou Deposition Tr. (10.12.18) at 33:17-37:2; 42:12-25.

	Systems and Software," June 2002, ACM Computing Surveys, Vol. 34, No. 2, pp. 171–210. IPR2018-01601, Ex. 1003 – Declaration of Dr. Stone at ¶¶ 86-88 IPR2018-01601, Paper 1 at 16-17. Declaration of Henry Houh at ¶¶ 199-200.**Error! Hyperlink reference not valid.** Deposition of Dr. El-Ghazawi (Microsoft) at 67:13-25. Deposition of Dr. Houh at 162:22-163:4. Deposition of Derek Chiou at 34:2-35:22.**Error! Hyperlink reference not valid.**	

Claim Language (Disputed Terms in **Bold**) `687 Patent	Plaintiff's Proposed Construction and Evidence in Support	Defendant's Proposed Construction and Evidence in Support
Order of steps of claim 1 to be performed	No specific order is required **Intrinsic Evidence:** '687 patent: Abstract, Fig. 14; col. 2:18-25, col. 2:47-65, col. 21:1-23/ **Extrinsic Evidence:**	Steps to be performed in order. **Intrinsic Evidence:** '687 patent, Claim 1, Fig. 14 **Extrinsic Evidence:** Declaration of Henry Houh ¶¶ 181-189.

	IPR2018-01601, Ex. 1003 – Declaration of Dr. Stone at ¶¶ 45-54 (no order requirement noted). Rebutal Declaration of Dr. El-Ghazawi at ¶¶ 16-18. Declaration of Dr. Homayoun at ¶¶ 31-34. Deposition of Dr. El-Ghazawi at 66-72 Deposition of Dr. Houh at 12:6-30:23, 32:25-38:22, 40:8-16, 157:11-18. Declaration of Dr. Houh at ¶ 183-188.	Rebuttal Declaration of Tarek El-Ghazawi ¶¶ 16-18. Rebuttal Declaration of Houman Homayoun ¶¶ 31-34. Tarek El-Ghazawi Deposition Tr. (10.18.18) at 66:21-72:5.
"**internet site**" / "**at an internet site**" Found in claim numbers: `687 Patent: 1, 18	one or more internet connected web site servers **Intrinsic Evidence:** '687 patent: Figs. 12-14; col. 20:36-21:24 ("With reference additionally now to FIG. 12, a simplified illustration of a representative operating environment 300 for the system and method of the present invention is shown including a typical web site server 306 as would be replaced by, for example, an SRC-6 reconfigurable server 308 (comprising, for example, the multiprocessor computer 10 or computer system 20 of the preceding figures"). **Extrinsic Evidence:** U.S. Patent No. 5,715,453: Figs 1-9; col.	A location publicly accessible on the Internet. **Intrinsic Evidence:** '687 patent, Fig. 12, 1:37-52 ("By virtue of its computer-based nature, many electronic commerce ("e-commerce") web sites employ various methods to allow their content to be varied based on the demographics of the particular user. . . . In either instance however, this data must be processed by the site such that the web page content may be altered in an effort to maximize it [sic] appeal to that particular site visitor with a view toward ultimately maximizing site revenue.") **Extrinsic Evidence:** U.S. Patent No. 5,838,910 to Domenikos

	1:23-63 ("One significant computer network that has recently become very popular is the Internet. The Internet grew out of this proliferation of computers and networks, and has evolved into a sophisticated worldwide network of computer systems. A user at an individual PC (i.e., workstation) that wishes to access the Internet typically does so using a software application known as a web browser… These web servers typically have hard-coded transaction processors that detect a specific type of query for dynamic data within the HTML page data, and that perform the necessary accesses to a dedicated data source to retrieve the dynamic data. The retrieved data is then inserted into the HTML page, thereby allowing the dynamic data to be displayed to the user within the HTML page. If more than one data source is present, or if more than one type of data source is used, the web server must be manually reprogrammed to accommodate the specific number and types of data sources to be accessed.), 2:62-3:52 A Dictionary of Computer Science, Oxford University Press (7th ed. 2016) (web service: a software component that publishes its *interface on and responds to requests from a network, particularly the internet… The ambition of web services is to use the Internet to implement large-scale *distributed processing in an open way.).	U.S. Patent No. 6,167,383 to Henson Microsoft Press Computer Dictionary (4th ed., 1999) (web site: A group of related HTML documents and associated files, scripts, and databases that is served up by an HTTP server on the World Wide Web. The HTML documents in a Web site generally cover one or more related topics and are interconnected through hyperlinks. Most Web sites have a home page as their starting point, which frequently functions as a table of contents for the site. Many large organizations, such as corporations, will have one or more HTTP servers dedicated to a single Web site. However, an HTTP server can also serve several small Web sites, such as those owned by individuals. Users need a Web browser and an Internet connection to access a Web site. *See also* home page, HTML, HTTP server (definition 1), Web browser.) Declaration of Henry Houh ¶¶ 190-193.

"**demographic data**" Found in claim numbers: `687 Patent: 5, 12, 13	information pertaining to the user **Intrinsic Evidence:** '687 patent: Figs. 13-14; col. 1:35-64 ("By virtue of its computer-based nature, many electronic commerce ('e-commerce') web sites employ various methods to allow their content to be varied based on the demographics of the particular user. This demographic information may be obtained in a variety of ways, with some sites simply requesting the site visitor respond to one or more questions while others may employ more sophisticated techniques such as "click stream" processing. In this latter instance, the prospective interests of the site visitor are inferred by determination and analysis of, for example, the previous sites he has visited. In either instance however, this data must be processed by the site such that the web page content may be altered in an effort to maximize it appeal to that particular site visitor with a view toward ultimately maximizing site revenue."); 2:48-65 ("Demographic data processing is merely an example of how the unique capabilities of such reconfigurable processing systems can be utilized to accelerate e-commerce, and 'secure socket' operation is yet another possible application."); col. 4:6-11 ("FIG. 14 is a corresponding flowchart illustrating the processing of demographic or other data utilizing a reconfigurable server for implementing the system and method of the	Information that identifies a particular segment of a population. **Intrinsic Evidence:** '687 patent, 1:38-40 ("web sites employ various methods to allow their content to be varied."). **Extrinsic Evidence:** http://www.dictionary.com/browse/demographic?s=t (demographic: 3. a specific segment of a population having shared characteristics.) U.S. Patent No. 6,182,068 to Culliss Declaration of Henry Houh ¶¶ 202-204. Henry Houh Deposition Tr. (10.17.18) at 163:15-166:25.

	present invention and which results in significantly improved access and data processing times."); col. 20:52-21:8 ("With reference additionally now to FIG. 13, a flowchart is shown illustrating a conventional data processing sequence 310 in a conventional application of a typical web site server 306 as depicted in the preceding figure. The sequence 310 begins with the input of a number 'N' of demographic data elements for processing by the typical web site server 306… With reference additionally now to FIG. 14, a corresponding flowchart is shown illustrating the processing of demographic or other data utilizing the reconfigurable server 308 of FIG. 12 in a significantly faster data processing sequence 330. The processing sequence 330 again begins with the input of N demographic data elements or other secure socket, database or other data for processing by the site server at input step 332."). U.S. Patent No. 6,128,663. IPR2014-00039, Exs. 1003, 1027, 1028.	

SRC Labs, LLC v. Microsoft Corp.
Joint Claim Construction

Claim Language (Disputed Terms in **Bold**) `524 Patent	Plaintiff's Proposed Construction and Evidence in Support	Defendant's Proposed Construction and Evidence in Support
"**memory module bus**" Found in claim numbers: '524 Patent: 1, 15	a bus connecting the memory controller to a module in the memory subsystem **Intrinsic Evidence:** '524 patent: Abstract; Fig. 5, col. 1:29-37, col. 2:34-38 ("To this end, by placing a MAP element (in, for example, a DIMM physical format) in one of the PC's DIMM slots, it's field programmable gate array ("FPGA") could accept the normal memory "read" and "write" transactions and convert them to a format used by an interconnect switch or network."), col. 3:4-21 ("Because both the disk's PCI port and the MAP element DIMM slots are controlled by the PC memory controller, no processor bus bandwidth is consumed by this transfer."), 5:39-57 ("By, for example, placing the MAP element 112 in the memory subsystem or memory space, it can be readily accessed through the use of memory "read" and "write" commands, which allows the use of a variety of standard operating systems."), col. 8:55-9:5 ("Because both the disk's PCI bus 210 and the DIMM MAP element 212 and DIMM slots 214 are controlled by the PC memory controller 204, no processor bus bandwidth is consumed by this transfer."), col. 9:25-26 ("wherein said memory module bus comprises an DIMM bus.").	A bus used to communicate with a memory module (or with components within that module). A memory module is a module used as memory for a data processor, but which may be used for other purposes. **Intrinsic Evidence:** '524 patent, Abstract ("A switch/network adapter port ('SNAP') for clustered computers employing multi-adaptive processor ('MAPTM', a trademark of SRC Computers, Inc.) elements in a dual in-line memory module ('DIMM') or RambusTM in-line memory module ('RIMM') format to significantly enhance data transfer rates over that otherwise available through use of the standard peripheral component interconnect ('PCI') bus."), 2:1-38 ("One of the most commonly used memory formats in PCs today is the dual inline memory module ('DIMM') format. These modules are-presently [sic] available in what is called a double data rate ('DDR') format and PCs using this format incorporate a memory bus that can provide up to 1.6 GB/sec."), 7:56-59 ("The controller 204 is also conventionally coupled to a number of DIMM slots 214 by means of a much higher bandwidth DIMM bus 216 capable of data transfer rates of substantially 2.1 GB/sec. or greater."); *see also* 1:29-67, 4:7-13, 7:34-41, 8:4-29.

	'152 patent, prosecution history at MS_SRC-SRMT0474517 ("The present invention provides a processor element for a memory module bus that enables a direct data connection between an external device and the memory module bus. The SIMMBUS of Klingelhofer is simply a memory bus and memory bus by itself fails to provide any mechanism or link by which to signal the memory controller that data on the bus is available."). **Extrinsic Evidence:** SRC Computers, Inc., "The Unified Computing Architecture," 2004. Computer Dictionary, Microsoft Press (3rd ed. 1997) (definition of "control unit"). IEEE std 610.10-994, IEEE Standard Glossary of Computer Hardware Terminology, 1995. Deposition of El-Ghazawi at 60-62. Deposition of Henry Houh at 118:1-14 (216 "would be an example of a memory module bus").	**Extrinsic Evidence:** Tarek El-Ghazawi Deposition Tr. (10.18.18) at 25:9-28:7, 60:9-62:20, 63:11-64:5. Declaration of Henry Houh ¶¶ 38, 57-61, 97, 159-163. U.S. Patent No. 5,673,204 to Klingelhofer; *see also* Fig. 1, 5:20-27.
"a direct data connection coupled to said field programmable gate array for **providing said altered data directly from said memory module bus to an**	This term has its plain and ordinary meaning and need not be construed. In the alternative, claim 1 should be construed to mean: an FPGA that receives and operates on data directly from a memory module bus,	Transferring the altered data directly from the memory module bus to an external device coupled thereto. For clarity, this claim term also requires that:

external device coupled thereto" Found in claim numbers: '524 Patent: 1	producing altered data and providing that altered data directly to an external device via a direct data connection **Intrinsic Evidence:** '524 patent: Abstract, Fig. 3, Fig. 5; col. 2:39-47 ("each MAP element may include chain ports to enable it to be coupled to other MAP elements. Through the utilization of the chain port to connect to the external clustering fabric, data packets can then be sent to remote nodes where they can be received by an identical board."), col. 3:6-44, col. 4:28-30, col. 5:50-57 ("it is possible to allow a MAP element 112 to feed results to another MAP element 112 through use of a chain port"), col. 7:65-8:26 ("The DIMM MAP element 212 then may be coupled to another clustered computer MAP element by means of a cluster interconnect fabric connection 220 connected to the MAP chain ports. …Through the utilization of the chain port to connect to the external clustering fabric over connection 220, data packets can then be sent to remote, nodes where they can be received by an identical board."), col. 9:42-10:4 (claim 1). '524 patent, prosecution history at MS_SRC-SRMT0474590-93 ("The following is an examiner's statement of reasons for allowance: The prior art of record fails to teach alone or in combination, a processor element for a memory module bus comprising an FPGA that	The data that is transferred has been previously provided to the FPGA on that same memory module bus and altered by the FPGA, and the transfer is directly from the same memory module bus to an external device coupled to that bus. **Intrinsic Evidence:** '524 patent, 5:50-54. '524 patent file history at MS_SRC-SRMT_0474425 (November 23, 2004 Preliminary Amendment at 3), MS_SRC-SRMT_0474512-4517 (June 6, 2007 Response to March 6, 2007 Office Action at 2-7) ("1. A processor element for a memory module bus of a computer system, said processor element comprising: a field programmable gate array configurable to perform an identified algorithm on an operand provided thereto and operative to alter data provided directly thereto on said memory module bus; and a direct data connection coupled to said field programmable gate array for providing said altered data from said memory module bus directly to an external device coupled thereto." (emphasis in original) (discussing applicant's argument that the Klingelhofer prior art patent did not disclose a direct transfer from the memory module bus to the external device in relation to amended claim 37), MS_SRC-SRMT_0474537 (August 27, 2007 Office Action Summary at 4) ("[T]he applicant again is arguing limitations that are not required by the claim language. The claim recites a direct data connection coupled to the FPGA for providing data from the

	receives and operates on data directly from a memory module bus, producing altered data and providing that altered data directly to an external device via a direct data connection, in combination with the other recited claim elements."), MS_SRC-SRMT0474551-557 ("Examiner suggested that rewording the limitation to state, 'providing said altered data directly from said memory module bus to an external device coupled thereto' would be a positive reciting of this limitation. … the Examiner confirmed that this wording would overcome Klingelhoffer since Klingelhoffer includes a buffer (VRAM) interposed between the memory module bus and the external device."); MS_SRC-SRMT0474511-18. ("independent claim 37 recites, among other things, 'a <u>direct</u> data connection coupled to said field programmable gate array for providing said altered data <u>from said memory module bus directly</u> to an external device coupled thereto.'"), MS_SRC-SRMT0474517 ("The present invention provides a processor element for a memory module bus that enables a <u>direct</u> data connection between an external device and the memory module bus."). **<u>Extrinsic Evidence:</u>** Chris Orth, "User Hardware Interfacing to the Chain Port of the SRC MAP® Rev D Board," 2003, SRC00004427-46.	memory module bus directly to an internal device, and not providing for the direct transfer of data from the memory module bus to the external device. There is an important distinction in what is claimed compared to what is argued by the applicant. The claim only requires that the data be directly provided to the external device and that the data come from the memory module bus. The data [in Klingelhofer] does come from the memory module bus, and is temporarily stored in the VRAM before being directly transferred to the external device. The claim does not require a direct transfer from the memory module bus to the external device."), MS_SRC-SRMT_0474552, 4555-4556 (October 26, 2007 Response to August 27, 2007 Office Action at 2, 5-6) ("1. A processor element for a memory module bus of a computer system, said processor element comprising: a field programmable gate array configurable to perform an identified algorithm on an operand provided thereto and operative to alter data provided directly thereto on said memory module bus; and a direct data connection coupled to said field programmable gate array for providing said altered data <u>directly</u> from said memory module bus ~~directly~~ to an external device coupled thereto." (emphasis added) (indicating further revisions to amended claim 37), ("The Examiner's response to the third argument presented in the response of June 6, 2007, was discussed during the interview. Namely, that the second limitation of claim 37 does not require a direct transfer of data from the memory module bus to the external device. The Examiner suggested that

	SRC Computers, Inc., "The Unified Computing Architecture," 2004, SRC00004583-587. David Caliga and David Barker, "Delivering Acceleration: The Potential for Increasing HPC Application Performance Using Reconfigurable Logic" SRC00004359-378. Declaration of Dr. El-Ghazawi at ¶ 50. Deposition of Dr. El-Ghazawi at 103-106. Deposition of Henry Houh at 118-149.	rewording the limitation to state, "providing said altered data directly from said memory module bus to an external device coupled thereto" would be a positive reciting of this limitation. The impact of this amendment was discussed with respect to Klingelhoffer . . . ."). **Extrinsic Evidence:** Tarek El-Ghazawi Deposition Tr. (10.18.18) at 74:12-21, 78:20-25, 79:13-80:25, 82:6-83:15, 84:5-85:12, 87:14-89:12, 89:22-95:3. Declaration of Henry Houh ¶¶ 173-180.

SRC Labs, LLC et al v. Amazon.com et al
Joint Claim Construction

DISPUTED CONSTRUCTIONS

Claim Language `324 Patent `800 Patent	Plaintiff's Proposed Construction and Evidence in Support	Defendant's Proposed Construction and Evidence in Support
"systolic" / "systolically" Found in claim numbers: '324 Patent: 1	This term has its plain and ordinary meaning and need not be construed. In the alternative, this term may be construed as: rhythmically computing and passing data in a transport triggered manner **Intrinsic Evidence:** '324 patent: Fig. 2; Fig. 4; Figs. 7A-7D; Fig. 8A; col. 6:1-30 ("With reference additionally now to FIG. 4A, a simplified logic flowchart is provided illustrating a conventional sequential processing operation 400 in which nested Loops A (first loop 402) and B (second loop 404) are alternately active on different phases of the process…In contrast to the sequential processing operation 400 (FIG. 4A) the solution to the problem of most effectively utilizing available resources is to have an application evaluate a problem in a data flow sense. That is, it will 'pass' a subsequent dimension of a given problem through the first loop 412 of logic concurrently with the previous dimension of data being processed through the second loop 414. In practice, a 'dimension' of data can be: multiple vectors of a problem, multiple planes of a problem,	"rhythmically computing and passing data directly between processing elements without a program counter or clock that drives the movement of data and operating in a manner that is transport triggered, *i.e.*, by the arrival of a data object" **Intrinsic Evidence:** '324 patent, prosecution history, e.g.,: 12/16/2005 Response to Office Action at 12 (MS_SRC-SRMT_0472991) ("Systolic sequential parallelism utilizes an array of processing elements (typically multiplier-accumulator chips) in a pipeline structure. The 'systolic,' coined by H.T. Kung of Carnegie-Mellon, refers to the rhythmic transfer of data through the pipeline, like blood flowing through the vascular system. Such an approach inherently accomplishes calculations by using a serialized approach. As recited in Gupta, '. . . the algorithm selects a set of FUs [Functional Units] to be instantiated in the data path, one by

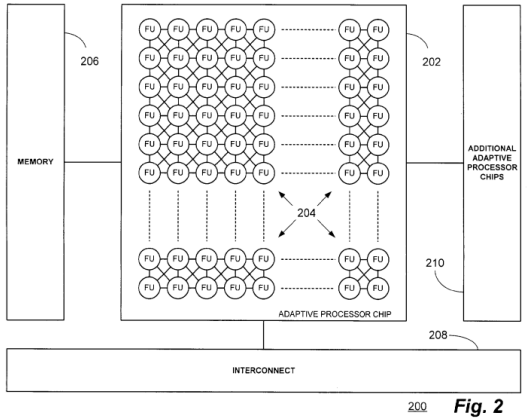
	multiple time steps in a problem and so forth."), col. 8:27-40. '324 patent, prosecution history at MS_SRC-SRMT0473096-99 ("Similarly the term systolic computation is derived from continual and pulsating pumping of the human heart. In computer architecture a systolic array is an arrangement of data processing units similar to a central processing unit but without a program counter or clock that drives the movement of data. That is because the operation of the systolic array is transport triggered, i.e. by the arrival of a data object. … define a Systolic system as a 'network of processors which rhythmically compute and pass data through the system.'"). David Caliga and David Peter Barker, "Delivering Acceleration: The Potential for Increased HPC Application Performance Using Reconfigurable Logic," Nov. 2001, ACM/IEEE SC 2001 Conference, pp. 20. **Extrinsic Evidence:** Jean-Luc Gaudiot, "Data-Driven Multicomputers in Digital Signal Processing," 1987, IEEE, Proceedings of the IEEE, Vol. 75, No. 9, pp. 1220-1234. A Dictionary of Computer Science, Oxford University Press (7th ed. 2016) (Dataflow machine: a computer in which the primitive operations are triggered by the availability of inputs or operands...	one, by looking at the requirement of the operation group cliques provided.'") (emphasis in original); 11/13/2006 Response to Office Action at 13-14 (MS_SRC-SRMT_0473097-98) ("Similarly the term systolic computation is derived from continual and pulsating pumping of the human heart. In computer architecture a systolic array is an arrangement of data processing units similar to a central processing unit but without a program counter or clock that drives the movement of data. That is because the operation of the systolic array is transport triggered, i.e. by the arrival of a data object. Data flows across the array between functional units, usually with different data flowing in different directions. . . . Thus in the Applicant's invention Systolic implementation will connect computational loops such that data from one compute loop will be passed as input data to a concurrently executing compute loop. In the Applicant's invention data computed by computation units or groups of functional units flows seamlessly and concurrently with data being computed by other groups of functional units. Thus, the process claimed by the Applicant therefore significantly increases the computing processes taking place in a reconfigurable processor."); *see also id.* at 12; Claim 1; U.S. Patent No. 6,385,757 to Gupta et al.

	in a dataflow machine there is a flow of data values from operations that produce those values to operations that 'consume' those values as operands."). A Dictionary of Computer Science, Oxford University Press (7th ed. 2016) (systolic array: Data is pulsed from the memory into processor(s) at an array boundary and then pulses through the array…). IEEE std 610.10-994, IEEE Standard Glossary of Computer Hardware Terminology, 1995, pg. 7 ("3.106 asynchronous computer. A computer in which each event or operation is performed upon receipt of a signal generated by the completion of a previous event or operation …."). Kung, H.T., "Why Systolic Architectures?", *IEEE Computer*, Jan. 1982, 37-46 Kung, et al., "Systolic arrays for VLSI," SIAM, Sparse Matrix Proceedings, 1978 1-29. Declaration of Dr. El-Ghazawi (Microsoft) at ¶¶ 14 and 16. Deposition of Henry Houh at 163:5-14.	**Extrinsic Evidence:** Expert report/testimony of Dr. Brad Hutchings, Ph.D., at, e.g., ¶¶ 60-67 Kung, et al., "Systolic Arrays for VLSI," SIAM, 1978, Sparse Matrix Proceedings, 256-282, Gaudiot, Jean-Luc, Data-Driven Multicomputers in Digital Signal Processing, 1987, IEEE, Proceedings of the IEEE, vol. 75, No. 9, 1220-1234, H. T. Kung, "Why Systolic Architectures?," *IEEE Computer*, Jan., 1982, 37-46, U.S. Patent No. 5,956,518, and U.S. Patent No. 5,274,832.
"pass computed data seamlessly" Found in claim numbers:	communicating the computed data over the reconfigurable routing resources **Intrinsic Evidence:**	"to communicate computed data directly" **Intrinsic Evidence:**

’324 Patent: 1 ’800 Patent: 1	’324 patent: Fig. 4; col. 6:1-30 (“With reference additionally now to FIG. 4A, a simplified logic flowchart is provided illustrating a conventional sequential processing operation 400 in which nested Loops A (first loop 402) and B (second loop 404) are alternately active on different phases of the process…In contrast to the sequential processing operation 400 (FIG. 4A) the solution to the problem of most effectively utilizing available resources is to have an application evaluate a problem in a data flow sense. That is, it will ‘pass’ a subsequent dimension of a given problem through the first loop 412 of logic concurrently with the previous dimension of data being processed through the second loop 414. In practice, a ‘dimension’ of data can be: multiple vectors of a problem, multiple planes of a problem, multiple time steps in a problem and so forth.”). ’800 patent: Fig. 4; col. 5:65-6:28 (“With reference additionally now to FIG. 4A, a simplified logic flowchart is provided illustrating a conventional sequential processing operation 400 in which nested Loops. A (first loop 402) and B (second loop 404) are alternately active on different phases of the process… In contrast to the sequential processing operation 400 (FIG. 4A) the solution to the problem of most effectively utilizing available resources is to have an application evaluate a problem in a data flow sense. That is, it will “pass a subsequent dimension of a given problem through the first loop 412 of logic concurrently with the previous dimension of data being processed through the	’324 patent, prosecution history, e.g.: 11/13/2006 Response to Office Action at, e.g., 14 (MS_SRC-SRMT_0473098) (“Thus in the Applicant’s invention Systolic implementation will connect computational loops such that data from one compute loop will be passed as input data to a concurrently executing compute loop. In the Applicant’s invention data computed by computation units or groups of functional units flows seamlessly and concurrently with data being computed by other groups of functional units. Thus, the process claimed by the Applicant therefore significantly increases the computing processes taking place in a reconfigurable processor.”); *see also id.* at 3, 12, 15, 16 ’324 patent at, e.g., Figs. 7A, 7B; 5:40-53 (“As shown, each adaptive processor chip 202 can contain thousands of functional units 204 dedicated to the particular problem at hand. Interconnect between these functional units is created by reconfigurable routing resources inside each chip 202. As a result, the functional units 204 can share or exchange data at much higher data rates and lower latencies than a standard microprocessor 104 (FIG. 1). In addition, the adaptive processor chips 202 can connect directly to the inter-processor interconnect 208 and do not

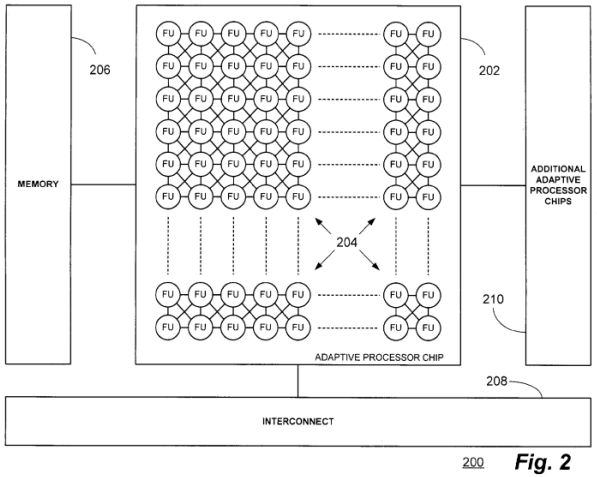
	second loop 414. In practice, a "dimension of data can be: multiple vectors of a problem, multiple planes of a problem, multiple time steps in a problem and so forth."). '324 patent, prosecution history at MS_SRC-SRMT0473096 ("the use of the words 'protocol independent' to impart the ability of the functional units to seamlessly pass computed data between computational loops comprised of functional unit."); MS_SRC-SRMT0473097 ("'protocol' has been replaced with an 'interconnection' between functional units that is established by reconfigurable routing resources inside each chip.""); MS_SRC-SRMT0473086-87 ("communication between other reconfigurable processors within the system would require communication protocol but communication between functional units within an individual reconfigurable processor is free of such a requirement."); MS_SRC-SRMT0473098 ("Data flows across the array between functional units, usually with different data flowing in different directions. David J. Evans in his work, Systolic algorithms. Systolic algorithms, number 3 in Topics in Computer Mathematics, Gordon and Breach, 1991 define a Systolic system as a 'network of processors which rhythmically compute an pass data through the system' Thus in the Applicant's invention Systolic implementation will connect computational loops such that data from one compute loop will be passed as input data to a concurrently executing compute loop. In the	require the data to be passed through multiple chips in a chipset in order to communicate. This is because the adaptive processor can implement whatever kind of interface is needed to accomplish this connection."); 6:21-30 ("In contrast to the sequential processing operation 400 (FIG. 4A) the solution to the problem of most effectively utilizing available resources is to have an application evaluate a problem in a data flow sense. That is, it will 'pass' a Subsequent dimension of a given problem through the first loop 412 of logic concurrently with the previous dimension of data being processed through the second loop 414. In practice, a 'dimension' of data can be: multiple vectors of a problem, multiple planes of a problem, multiple time steps in a problem and so forth."); 7:49-58 ("With reference additionally now to FIG. 7B, the general computation of fluid flow properties in the reservoir simulation process 700 of the preceding figure are illustrated as values are communicated between a group of neighboring cells 710. The group of neighboring cells 710 comprises, in the simplified illustration shown, first, second and third walls of cells 712, 714 and 716 respectively. Each of the walls of cells includes a corresponding number of first, second, third and fourth rows 718, 720, 722 and 724 respectively.");

	Applicant's invention data computed by computation units or groups of functional units flows seamlessly and concurrently with data being computed by other groups of functional units."). David Caliga and David Peter Barker, "Delivering Acceleration: The Potential for Increased HPC Application Performance Using Reconfigurable Logic," Nov. 2001, ACM/IEEE SC 2001 Conference, pp. 7-20. **Extrinsic Evidence:** A Dictionary of Computer Science, Oxford University Press (7th ed. 2016) (Dataflow machine: a computer in which the primitive operations are triggered by the availability of inputs or operands... in a dataflow machine there is a flow of data values from operations that produce those values to operations that 'consume' those values as operands."). IEEE std 610.10-994, IEEE Standard Glossary of Computer Hardware Terminology, 1995, pg. 7 ("3.106 asynchronous computer. A computer in which each event or operation is performed upon receipt of a signal generated by the completion of a previous event or operation …."). Jack B. Dennis, "Data Flow Supercomputers" 1980, Computer, 10.1109/MC.1980.1653418, pp. 52.	7:59-62 ("As shown, the computation of fluid flow properties are communicated to neighboring cells 710 and, importantly, this computation can be scheduled to eliminate the need for data storage."); 10:2-13 ("Reservoir Simulation: These applications, also typically used in the oil and gas production industries, process fluid flow data in the oil and gas subsurface reservoirs to produce extraction models. The application will define a three dimensional ("3d") set of cells that contain the oil and gas reservoir. These programs are ideal candidates to take advantage of parallel or adaptive computing because there are repeated operations on each cell. In addition, information computed for each cell is then passed to neighboring cells. These programs will particularly benefit from the tight parallelism that can be found in adaptive or reconfigurable processors."); 10:55-65 ("Crash Analysis: These applications are typically used in the automotive or aviation industry. The application will partition the entire automobile into components. These components are then subdivided into cells. The application will analyze the effect of a collision on the structure of the automobile. These programs are ideal candidates for parallel computing because there are repeated operations on each cell and they receive computed information from their neighboring cells. These programs will benefit from the tight

		parallelism that can be found in adaptive or reconfigurable processors."); Claim 1; FIG. 2:  ***Fig. 2*** '800 patent at, e.g., 5:38-50 ("As shown, each adaptive processor chip 202 can contain thousands of functional units 204 dedicated to the particular problem at hand. Interconnect between these functional units is created by reconfigurable routing resources inside each chip 202. As a result, the functional units 204 can share or exchange data at much higher data rates and lower latencies than a standard microprocessor 104 (FIG. 1). In addition, the

		adaptive processor chips 202 can connect directly to the inter-processor interconnect 208 and do not require the data to be passed through multiple chips in a chipset in order to communicate. This is because the adaptive processor can implement whatever kind of interface is needed to accomplish this connection."); 7:44-52 ("With reference additionally now to FIG. 7B, the general computation of fluid flow properties in the reservoir simulation process 700 of the preceding figure are illustrated as values are communicated between a group of neighboring cells 710. The group of neighboring cells 710 comprises, in the simplified illustration shown, first, second and third walls of cells 712, 714 and 716 respectively. Each of the walls of cells includes a corresponding number of first, second, third and fourth rows 718, 720, 722 and 724 respectively."); 7:53-56 ("As shown, the computation of fluid flow properties are communicated to neighboring cells 710 and, importantly, this computation can be scheduled to eliminate the need for data storage."); 9:59-10:2 ("Reservoir Simulation: These applications, also typically used in the oil and gas production industries, process fluid flow data in the oil and gas subsurface reservoirs to produce extraction models. The application will define a three dimensional ("3d") set of cells that contain the oil and gas reservoir. These programs are ideal

		candidates to take advantage of parallel or adaptive computing because there are repeated operations on each cell. In addition, information computed for each cell is then passed to neighboring cells. These programs will particularly benefit from the tight parallelism that can be found in adaptive or reconfigurable processors."); 10:43-52 ("Crash Analysis: These applications are typically used in the automotive or aviation industry. The application will partition the entire automobile into components. These components are then subdivided into cells. The application will analyze the effect of a collision on the structure of the automobile. These programs are ideal candidates for parallel computing because there are repeated operations on each cell and they receive computed information from their neighboring cells. These programs will benefit from the tight parallelism that can be found in adaptive or reconfigurable processors."); Claim 1; FIG. 2:

		 ***Fig. 2*** **<u>Extrinsic Evidence:</u>** Expert report/testimony of Dr. Brad Hutchings, Ph.D. at, e.g., ¶¶ 60-67
"instantiating"/ "instantiated" Found in claim numbers: '324 Patent: 1	This term has its plain and ordinary meaning and need not be construed. In the alternative, this term may be construed as: Configuring/configured **<u>Intrinsic Evidence:</u>** '324 patent: Fig. 2; Fig. 4; Figs. 7A-9C; col. 2:1-5 ("Reconfigurable processors instantiate only the	"creating or configuring to perform a defined calculation, each creation or configuration for each calculation is unique" **<u>Intrinsic Evidence:</u>** '324 patent at, e.g., Claim 1.

	functional units needed to solve a particular application, and as a result, have available space to instantiate as many functional units as may be required to solve the problem up to the total capacity of the integrated circuit chips they employ."); col. 2:49-58 ("instantiating at least two of the functional units to perform the calculation"); '800 patent: Fig. 2; Fig. 4; Figs. 7A-9C; col. 2:1-7; col. 2:53-55. '324 patent, prosecution history at MS_SRC-SRMT0473096-97 ("Instantiation is a term well known to one of ordinary skill in the art of reconfigurable processing. A reconfigurable processor is essentially a blank processor that must be configured (instantiated) to conduct a particular task. To instantiate means to create such an instance or configuration by, for example, defining one particular variation of the processor's structure."). David Caliga and David Peter Barker, "Delivering Acceleration: The Potential for Increased HPC Application Performance Using Reconfigurable Logic," Nov. 2001, ACM/IEEE SC 2001 Conference, pp. 20. **Extrinsic Evidence:** U.S. Patent No. 6,434,687 John Villasenor and Brad Hutchings, "The flexibility of configurable computing," 1997, IEEE Signal Processing Magazine, pp 67-84.	'324 patent, prosecution history, e.g.,: 11/13/2006 Response to Office Action at 13 (MS_SRC-SRMT_0473096-97) ("Instantiation is a term well known to one of ordinary skill in the art of reconfigurable processing. A reconfigurable processor is essentially a blank processor that must be configured (instantiated) to conduct a particular task. To instantiate means to create such an instance or configuration by, for example, defining one particular variation of the processor's structure. This involves allocation of a structure with the types specified by a template and the initialization of instance variables with either default values or those provided by a constructor function. In reconfigurable computing a hard macro library file is typically inserted into a design file. A design may include multiple instances of the same library file with each possessing a unique name. Thus in the Applicant's invention the reconfigurable processor is instantiated and designed to perform the defined calculation. Each instantiation for each calculation is unique."). **Extrinsic Evidence:** Expert report of Dr. Brad Hutchings, Ph.D. at, e.g., ¶¶ 60-67

	Katherine Compton and Scott Hauck, "Reconfigurable Computing: A Survey of Systems and Software," June 2002, ACM Computing Surveys, Vol. 34, No. 2, pp. 171–210. IPR2018-01601, Ex. 1003 – Declaration of Dr. Stone at ¶¶ 86-88 IPR2018-01601, Paper 1 at 16-17. Declaration of Henry Houh at ¶¶ 199-200.**Error! Hyperlink reference not valid.** Deposition of Dr. El-Ghazawi (Microsoft) at 67:13-25. Deposition of Dr. Houh at 162:22-163:4. Deposition of Derek Chiou (Microsoft) at 34:2-35:22.	

Claim Language '311 Patent	Plaintiff's Proposed Construction and Evidence in Support	Defendant's Proposed Construction and Evidence in Support
"a data maintenance block" Found in claim numbers: '311 Patent: 1	This term has its plain and ordinary meaning and need not be construed. **Intrinsic Evidence:** '311 patent: Figs.1-2, col. 2:4-3:4 ("In accordance with the system and method of the present invention, an FPGA based DRAM controller is utilized in concert with an internally or externally located data maintenance block. In operation, the FPGA drives the majority of the DRAM input/output (I/O) and the data maintenance block drives the self-refresh command inputs. Even though the FPGA reconfigures and the majority of the DRAM inputs are tri-stated, the data maintenance block provides stable input levels on the self-refresh command inputs… This means all the device resources previously reserved for creating a DRAM controller are now free to be used for different functions."), col. 3:40-50 ("Still further particularly disclosed herein is a method for preserving the contents of a DRAM memory associated with a reconfigurable device having a memory controller. The method comprises providing a data maintenance block coupled to the reconfigurable device, coupling the data maintenance block to self-refresh command inputs of the DRAM	"a device separate from the memory controller that drives self-refresh command inputs and stores DRAM memory data when the reconfigurable logic device is reconfigured" **Intrinsic Evidence:** '311 patent at e.g., 1:28-33 ("In order to mitigate design engineering costs and Verification time, it is very common for field programmable gate array (FPGA) designers to implement vendor provided memory controller intellectual property (IP) when including DRAM based memory solutions in their designs.") 1:48-53 ("A significant problem arises in reconfigurable computing when the FPGA is reprogrammed during a live application and the memory controller tri-states all inputs and outputs (I/O) between the FPGA device and the DRAM. The result is corrupted data in the memory sub system.") 1:63-64 ("Disclosed herein is a system and method for preserving DRAM memory contents when a

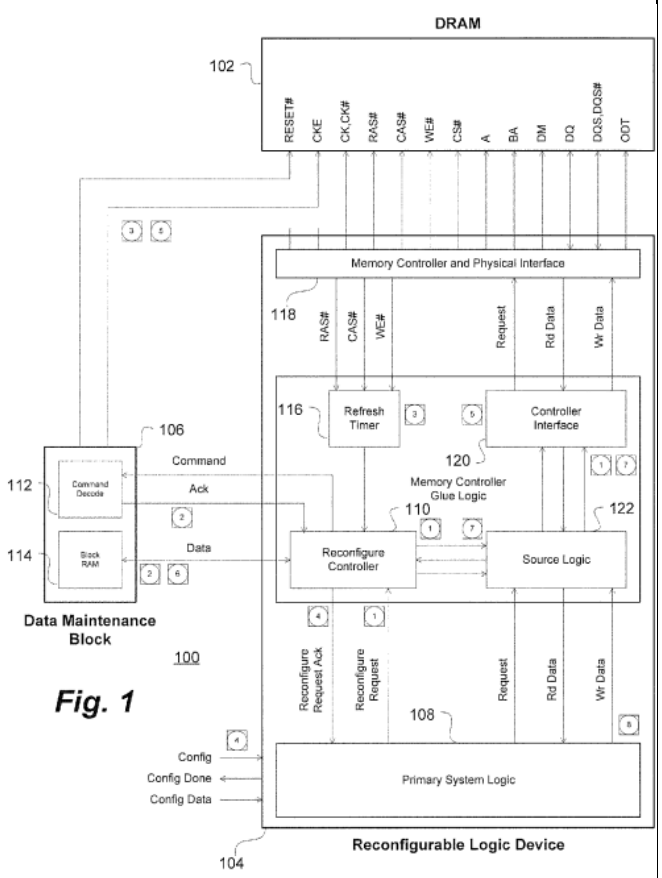
	memory, storing data received from the reconfigurable device at the data maintenance block and maintaining stable input levels on the self-refresh command inputs while the reconfigurable logic device is reconfigured."), col. 4:20-29 ("the data maintenance block 106 may be conveniently provided as a complex programmable logic device (CPLD) or other sperate integrated circuit device or, in alternative embodiments, may be provided as a portion of an FPGA comprising the reconfigurable logic device 104."), col. 4:52-54 ("The data maintenance block 106 is coupled to the DRAM memory 102 to supply reset (RESET#) and clock enable (CKE#) signals thereto."), col. 5:7-6:14 ("As indicated by the operation at numeral 2, the reconfigure Controller 110 stores its block RAM contents in another small section of block RAM 114 located in the data maintenance block 106."). '311 patent, prosecution history at SRC00001873. **Extrinsic Evidence:** John Schewel et al, "High-Speed Computing, Digital Signal Processing, and Filtering Using Reconfigurable Logic," Proceedings SPIE Vol. 2914 (Nov. 1996). IPR2018-01395, Paper No. 1 & Ex. 1003	reconfigurable device, for example an FPGA having a DRAM memory controller, is reconfigured, reprogrammed or otherwise powered down.") 2:4-3:8 ("In accordance with the system and method of the present invention, an FPGA based DRAM controller is utilized in concert with an internally or externally located data maintenance block. In operation, the FPGA drives the majority of the DRAM input/output (I/O) and the data maintenance block drives the self-refresh command inputs. Even though the FPGA reconfigures and the majority of the DRAM inputs are tri-stated, the data maintenance block provides stable input levels on the self-refresh command inputs. Functionally, the data maintenance block does not contain the memory controller and therefore has no point of reference for when and how to initiate the self-refresh commands, particularly the DRAM self-refresh mode. As also disclosed herein, a communication port is implemented between the FPGA and the data maintenance block that allows the memory controller in the FPGA to direct the self-refresh commands to the DRAM via the data maintenance block. Specifically, this entails when to put the DRAM into self-refresh mode and preserve the data in memory. At this point, the DRAM data has been preserved throughout the FPGA reconfiguration via the self-refresh mode initiated by the data maintenance block, but the DRAM controller must now reestablish write/read timing windows and will corrupt specific address

		contents with guaranteed write and read data required during the calibration/leveling process. Consequently, using the self-refresh capability of DRAM alone is not adequate for maintaining data integrity during reconfiguration. (It should be noted that the memory addresses used during calibration/leveling are known and typically detailed in the controller IP specification.) In order to effectuate this, the system transmits a 'reconfiguration request' to the DRAM controller. Once received, glue logic surrounding the FPGA vendor provided memory controller IP issues read requests to the controller specifying address locations used during the calibration/leveling process. As data is retrieved from the DRAM, it is transmitted via the communication port from the FPGA device to a block of storage space residing within the data maintenance block itself or another location in the system. Once the FPGA has been reprogrammed, the DRAM controller has re-established calibration settings and several specific addresses in the DRAM have been corrupted with guaranteed write/read data patterns. At this point, glue logic surrounding the vendor memory controller IP is advised by the data maintenance block (through the communication port) that it has awakened from either an initial power up condition or a reconfiguration condition. If a reconfiguration condition is detected, and before processing incoming DMA requests, the controller retrieves stored DRAM data from the data maintenance block (again through the communication port) and writes it back to the specific address locations

		corrupted during the calibration/leveling process. Once complete, the DRAM controller in the FPGA is free to begin servicing system memory requests in the traditional fashion. Among the benefits provided in conjunction with the system and method of the present invention is that since the data maintenance block functions to hold the DRAM in self-refresh mode, the FPGA is free to be reprogrammed to perform a very application-specific computing job that may not require DRAM. This means all the device resources previously reserved for creating a DRAM controller are now free to be used for different functions. Further, the overall computer system benefits from the present invention because data previously stored in DRAM has now been preserved and is available for use by the next application that needs it.") 3:21-25 ("Also particularly disclosed herein is a system and method for preserving DRAM data contents in a reconfigurable computing environment when the programmable device is reconfigured with a new design that does not include a DRAM controller.") 3:42-49 ("The method comprises providing a data maintenance block coupled to the reconfigurable device, coupling the data maintenance block to self-refresh command inputs of the DRAM memory, storing data received from the reconfigurable device at the data maintenance block and

		maintaining stable input levels on the self-refresh command inputs while the reconfigurable logic device is reconfigured."); 4:20-29 ("Also illustrated is a data maintenance block 106 in accordance with the present invention for retaining DRAM memory 102 data when the logic device 104 is reconfigured during operation of the computer subsystem 100. In a representative embodiment of the present invention, the data maintenance block 106 may be conveniently provided as a complex programmable logic device (CPLD) or other separate integrated circuit device or, in alternative embodiments, may be provided as a portion of an FPGA comprising the reconfigurable logic device 104."); 5:7-10 ("As indicated by the operation at numeral 2, the reconfigure Controller 110 stores its block RAM contents in another small section of block RAM 114 located in the data maintenance block 106."); 5:12-24 ("At the operation indicated by numeral 3, the reconfigure controller 110 detects a refresh command from the refresh timer 116, waits a refresh cycle time (t) and instructs the data maintenance block 106 to de-assert CKE to the DRAM memory 102. The reconfigure controller 110 asserts the Reconfigure Request Ack signal at the operation indicated by numeral 4 and the reconfigurable logic device 104 is reconfigured. As

		indicated by the operation at numeral 5, the reconfigure controller 110 recognizes a post-reconfigure condition (Ack=High), holds the memory controller and physical interface 118 in reset and instructs the data maintenance block 106 to assert CKE to the DRAM memory 102.") 5:27-36 ("At the operation indicated by numeral 6, the reconfigure controller 110 retrieves the data maintenance block 106 block RAM 114 contents and stores it in a small section of block RAM (not shown) in the reconfigure controller 110. The reconfigure controller 110 detects that the memory controller and physical interface 118 and DRAM memory 102 initialization is complete at the operation indicated by numeral 7 and initiates DMA write requests to restore the memory contents corrupted during the calibration/leveling sequence with the data values read prior to reconfiguration.") 5:63-6:1 ("The reconfigurable application logic 202 is coupled to the data maintenance blocks 106 and DRAM memory 102 as depicted and described previously with respect to the preceding figure and is also illustrated as being coupled to a number of 8 GB ECC static random access memory (SRAM) memory modules 204."); 6:5-8 ("It should be noted that the DRAM memory 102 controller in the reconfigurable application block 202 may be omitted upon subsequent reconfigurations as the DRAM memory 102 data

		contents will be maintained in the data maintenance blocks 106."); Claims 1, 11; FIG. 1: 

		4:52-55 ("The data maintenance block 106 is coupled to the DRAM memory 102 to supply reset (RESET#) and clock enable (CKE#) signals thereto."); Claim 1 ("a reconfigurable logic device having a memory controller coupled to selected inputs and outputs of said DRAM memory"). **Extrinsic Evidence:** Expert report/testimony of Dr. Brad Hutchings, Ph.D. at, e.g., ¶¶ 27-41

Claim Language '867 Patent	Plaintiff's Proposed Construction and Evidence in Support	Defendant's Proposed Construction and Evidence in Support
"A data prefetch unit" Found in claim numbers: '867 Patent: 1, 3, 4	a functional unit that moves data between members of a memory hierarchy. The movement may be as simple as a copy, or as complex as an indirect indexed strided copy into a unit stride memory **Intrinsic Evidence**: '867 patent: Abstract; Figs. 1-14; col. 5:18-57 ("Data prefetch Unit—is a functional unit that moves data between members of a memory hierarchy. The movement may be as simple as a copy, or as complex as an indirect indexed strided copy into a unit stride memory"); col. 7:33-8:52; col. 8:62-9:62. **Extrinsic Evidence**: John Schewel et al, "High-Speed Computing, Digital Signal Processing, and Filtering Using Reconfigurable Logic," Proceedings SPIE Vol. 2914 (Nov. 1996). IPR2019-00103, Paper 1 (Amazon's Petition for inter partes review of the '867 patent). IPR2019-00103, Ex. 1002 (Declaration of Dr. Hutchings ISO of Amazon's IPR petition). Declaration from Dr. El-Ghazawi at ¶¶ 10-23.	No construction of this phrase is necessary in light of the fact that a larger phrase (see below) must be construed. Amazon incorporates by reference evidence in support of the larger phrase (see below).

	Deposition of Dr. El-Ghazawi at 15-46, 56-58, 60, 62-64, 67, 97, 100-101, Declaration from Dr. Homayoun ¶¶ 21-35. Deposition of Dr. Homayoun at 19-20, 26-28, 31-33, 39, 45, 48-58, 83-96, 105-106.	
"a data prefetch unit coupled to the memory, wherein the data prefetch unit retrieves only computational data required by the algorithm from a second memory of second characteristic memory bandwidth and/or memory utilization and places the retrieved computational data in the first memory" Found in claim numbers: '867 Patent: 1	This term has its plain and ordinary meaning and need not be construed. **Intrinsic Evidence**: '867 patent: Abstract; Figs. 1-14; col. 5:18-57 ("Data prefetch Unit—is a functional unit that moves data between members of a memory hierarchy. The movement may be as simple as a copy, or as complex as an indirect indexed strided copy into a unit stride memory"); 6:47-57 ("In this manner, the hardware resources are essentially adapted to conform to the program rather than the program being adapted to conform to the hardware resources."), 7:5-22, 7:33-9:62. '867 prosecution history at SRC00000993-95, SRC00001103-104 **Extrinsic Evidence**: *Bandwidth Efficiency and Utilization Using Direct Execution Logic*, SRC00004528 John Schewel et al, "High-Speed Computing, Digital Signal Processing, and Filtering Using Reconfigurable Logic," Proceedings SPIE Vol. 2914 (Nov. 1996).	*Governed by pre-AIA 35 U.S.C. § 112, para. 6. Indefinite under pre-AIA 35 U.S.C. § 112.* ***Structure***: No disclosed structure ***Function***: "retrieves only computational data required by the algorithm from a second memory of second characteristic memory bandwidth and/or memory utilization and places the retrieved computational data in the first memory" **Intrinsic Evidence** '867 patent at, e.g., 5:44-46 ("Data prefetch Unit—is a functional unit that moves data between members of a memory hierarchy. The movement may be as simple as a copy, or as complex as an indirect indexed strided copy into a unit stride memory."); Claim 1.

	IPR2019-00103, Paper 1 (Amazon's Petition for inter partes review of the '867 patent). IPR2019-00103, Ex. 1002 (Declaration of Dr. Hutchings ISO of Amazon's IPR petition). Declaration from Dr. El-Ghazawi at ¶¶ 10-23. Deposition of Dr. El-Ghazawi at 15-46, 56-58, 60, 62-64, 67, 97, 100-101, Declaration from Dr. Homayoun ¶¶ 21-35. Deposition of Dr. Homayoun at 19-20, 26-28, 31-33, 39, 45, 48-58, 83-96, 105-106.	**Extrinsic Evidence:** Expert report/testimony of Dr. Brad Hutchings, Ph.D. at, e.g., ¶¶ 42-55 Rebuttal expert reports/testimony of Tarek El-Ghazawi, at, e.g., ¶¶ 21-25 El-Ghazaw Deposition Tr. at, e.g., 16-19, 25-26, 32, 39-41 Rebuttal expert reports/testimony of Houman Homayoun at, e.g., ¶¶ 32-41 Houman Homayoun Deposition Tr. at, e.g., 20, 39-40, 47, 58,
"at least the first memory and data prefetch unit are configured to conform to needs of the algorithm" Found in claim numbers: '867 Patent: 1, 3, 4	This term has its plain and ordinary meaning and need not be construed. **Intrinsic Evidence**: '867 patent: Abstract; Figs. 1-14; col. 5:18-57 ("Data prefetch Unit—is a functional unit that moves data between members of a memory hierarchy. The movement may be as simple as a copy, or as complex as an indirect indexed strided copy into a unit stride memory"); 6:47-57 ("In this manner, the hardware resources are essentially adapted to conform to the program rather than the program being adapted to conform to the hardware resources."), 7:5-22, 7:33-9:62	*Indefinite under pre-AIA 35 U.S.C. § 112.* **Intrinsic Evidence:** '867 patent at, e.g., 5:40-41 ("Data prefetch Unit is a functional unit that moves data between members of a memory hierarchy.") 5:34-35 ("Functional Unit is a set of logic that performs a specific operation.")

	'867 prosecution history at SRC00000993-95, SRC00001103-104 **Extrinsic Evidence:** *Bandwidth Efficiency and Utilization Using Direct Execution Logic*, SRC00004528 A Dictionary of Computer Science, Oxford University Press (7th ed. 2016) (definition of Bandwidth), SRC000034372-73. John Schewel et al, "High-Speed Computing, Digital Signal Processing, and Filtering Using Reconfigurable Logic," Proceedings SPIE Vol. 2914 (Nov. 1996). IPR2019-00103, Paper 1 (Amazon's Petition for inter partes review of the '867 patent). IPR2019-00103, Ex. 1002 (Declaration of Dr. Hutchings ISO of Amazon's IPR petition). Declaration from Dr. El-Ghazawi at ¶¶ 21-25. Deposition of Dr. El-Ghazawi at 77-93, 103, 106-110, 111-112. Declaration from Dr. Homayoun ¶¶ 36-41. Deposition of Dr. Homayoun at 60-79, 83-	7:48-62 ("An important feature of the present invention is that many types of data prefetch units can be defined so that the prefetch hardware can be configured to conform to the needs of the algorithms currently implemented by the computational logic. The specific characteristics of the prefetch can be matched with the needs of the computational logic and the format and location of data in the memory hierarchy. For example, FIG. 9A and FIG. 9B show an external memory that is organized in a 128 byte (16 word) block structure. This organization is optimized for stride 1 access of cache based computers. A stride 128 access can result in a very inefficient use of bandwidth from the memory, since an extra 120 bytes of data is moved for every 8 bytes of requested data yielding a 6.25% bandwidth efficiency."); 6:47-57 ("Alternatively, an algorithm can be defined in a high level language then compiled into DEL. DEL can be produced via a compiler from high level programming languages such as C or FORTRAN or may be designed using a hardware definition language such as Verilog, VHDL or a schematic capture tool. Computation is performed by reconfiguring a reconfigurable processor with the DEL and flowing data through the computation. In this manner, the hardware resources are essentially adapted to conform to the program rather than the program being adapted to conform to the hardware resources.");

		8:25-27 ("In this example strided data prefetch units 701 fetch only the required data words from external memory.") 9:2-13 ("Again, an important feature of the present invention is the ability to implement various kinds or styles of prefetch units to meet the needs of a particular algorithm being implemented by computational elements 301. For ease of illustration, each example shows the same set of computational logic, however, in most cases the function being implemented by components 301 would change and therefore alter the decision as to which prefetch strategy is most appropriate. In accordance with the present invention, the prefetch units are implemented in a manner that is optimized for the implemented computational logic."). 9:19-23 ("In each of these examples the data prefetch units are configured to pass information to the intelligent memory controller 601 to identify the type of request that is being made, as well as a data address and parameters, in this case, defining the slice size or Sub-cube size."); Claims 1, 3, 4, 9, 13. **Extrinsic Evidence:**

		Expert report/testimony of Dr. Brad Hutchings, Ph.D. at, e.g., ¶¶ 42-46, 56-59 Rebuttal expert report/testimony of Tarek El-Ghazawi, at, e.g., ¶¶ 21-25 El-Ghazaw Deposition Tr. at, e.g., 83-87, 100 Rebuttal expert report/testimony of Houman Homayoun at, e.g., ¶¶ 36-41 Houman Homayoun Deposition Tr. at, e.g., 69, 77-79