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.
10/459,797	06/12/2003	Mark M. Leather	00100.02.0053	4148

29153 7590 07/23/2009

ADVANCED MICRO DEVICES, INC.
C/O VEDDER PRICE P.C.
222 N.LASALLE STREET
CHICAGO, IL 60601

EXAMINER	
HSU, JONI	

ART UNIT	PAPER NUMBER
2628	

MAIL DATE	DELIVERY MODE
07/23/2009	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.

UNIFIED 1011

Office Action Summary	Application No.	Applicant(s)	
	10/459,797	LEATHER ET AL.	
	Examiner	Art Unit	
	JONI HSU	2628	

-- 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 *13 May 2009*.

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) *1-7,10-22,24 and 25* 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4a) Of the above claim(s) _____ is/are withdrawn from consideration.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) *1-7,10-22,24 and 25* 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 *5/13/09*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Continued Examination Under 37 CFR 1.114

1. 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. Since this application is eligible for continued examination under 37 CFR 1.114, 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 May 13, 2009 has been entered.

Information Disclosure Statement

2. The information disclosure statement (IDS) submitted on May 13, 2009 was filed after the mailing date of the application on June 12, 2003. 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 Arguments

3. Applicant's arguments, see p. 8-12, filed May 13, 2009, with respect to the rejection(s) of claim(s) 1-7, 10-22, 24, and 25 under 35 U.S.C. 103(a) have been fully considered and are persuasive. Therefore, the rejection has been withdrawn. However, upon further consideration, a new ground(s) of rejection is made in view of MacInnis (US006570579B1) in view of Perego (US006864896B2).

4. As per Claim 1, Applicant argues that in the "Response to Arguments" section, the Examiner states "Claim 1 recites 'at least two graphics pipelines on a same chip' and this is what Perego teaches'". However, in the rejection, it is Furtner (US006778177B1) that is cited as

teaching graphics pipelines are on a same chip. As it currently stands, Applicants are unable to determine which reference the Examiner is using (p. 8).

In reply, the Examiner points out that Furtner was used because it more explicitly teaches the limitation. However, the Examiner agrees that this may be confusing, and is no longer applying Furtner to teach this limitation.

5. Applicant argues that Perego states that as to the scalability which is a requirement of Perego, "this ability to add and remove memory modules 304 provides an upgradeable and scalable memory and computing architecture." Perego requires the separate memory controller 310 to be separate from the separate memory controller that is coupled to the memory modules 304. Therefore, the teaching from MacInnis that the memory controller is also on the same chip cannot be combined with Perego because removing the scalability renders the Perego invention inoperable for its intended purpose (p. 8-9).

In reply, the Examiner agrees. However, new grounds of rejection are made so that MacInnis is used as the main reference and the teaching of the repeating tile pattern from Perego is combined into the main reference MacInnis. Therefore, Perego is now only being used for the teaching of the repeating tile pattern, not for the teaching of the scalability. Since Perego is now only being used as a secondary reference, the Examiner is no longer relying on the entire device of Perego for the rejection.

6. As per Claim 4, Applicant argues that there is no front end circuitry described related to the CPU 308 in Perego since admitted in the "Response to Arguments" section, the CPU of Perego does not generate pixel data (p. 10-11).

In reply, the Examiner points out that Perego teaches that the CPU sorts the primitive data according to the spatial region of the rendering surface (e.g., the x and y coordinates) covered by that primitive, and the rendering surface is divided into multiple rectangular regions of pixels (col. 5, lines 19-27). One of ordinary skill in the art would understand that data pertaining to the x and y coordinates covered by the primitive would include vertex data. The main memory is used to store data which are referenced during the execution of the programs (col. 1, lines 18-21). Therefore, the **CPU** is operative to receive vertex data (data pertaining to the x and y coordinates covered by the primitive) from the main memory and **generate pixel data** corresponding to a primitive to be rendered (col. 5, lines 19-27; col. 1, lines 18-21). Perego teaches that rendering engine 312 is coupled to the CPU, and is operative to receive and process a rectangular region of pixel data (col. 3, line 64-col. 4, line 2; col. 5, lines 19-44). Since the CPU is coupled to the rendering engine 312, and the CPU and the rendering engine 312 both perform graphics processing, and, the CPU performs the operations of the front end circuitry, and the rendering engine 312 performs the operations of the back end circuitry, the CPU and the rendering engine 312 are considered to be part of a graphics pipeline, and the CPU is considered to be the front end circuitry included in the graphics pipeline.

7. As per Claim 24, Applicant argues that Perego does not describe multiple backend circuitry on a common chip nor common front end circuitry and memory controller on a common chip (p. 11).

In reply, the Examiner points out that Perego shows in Fig. 8 that in one embodiment, a memory module 800 contains two different rendering engines 802 and 810 (Fig. 8; col. 6, lines 61-62), and therefore at least two graphics pipelines (802, 810) are on a same memory module

800. From Fig. 8 and from the description in Perego, one of ordinary skill in the art would understand that a memory module is equivalent to a chip. Therefore, Perego teaches that two graphics pipelines are on the same chip, and so first backend circuitry (first rendering engine 312) and second backend circuitry (second rendering engine 312) (col. 3, lines 65-67; col. 4, liens 1-53; col. 5, lines 32-44) are on a common chip. In response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.,* 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). MacInnis is used to expressly teach common front end circuitry and memory controller are on a common chip.

8. As per Claim 20, Applicant argues that Perego does not teach passing the same pixel data to both of the two graphics pipelines on a same chip. (See for example, Specification describing shared front end). Perego does not teach determining the pixels within a set of tiles of the repeating pattern which are to be processed by corresponding graphics pipeline on the same chip in response to pixel data (p. 12).

In reply, the Examiner points out that Perego teaches a shared front end (308, Fig. 3) that passes the pixel data to both of the two graphics pipelines (first rendering engine 312 and second rendering engine 312) (col. 3, line 63-col. 4, line 2; col. 5, lines 19-44), similarly as what is described in the instant specification, and so Perego teaches passing the same pixel data to both of the two graphics pipelines. Perego shows in Fig. 8 that in one embodiment, a memory module 800 contains two different rendering engines 802 and 810 (Fig. 8; col. 6, lines 61-62), and therefore at least two graphics pipelines (802, 810) are on a same memory module 800. From

Fig. 8 and from the description in Perego, one of ordinary skill in the art would understand that a memory module is equivalent to a chip. Therefore, Perego teaches that two graphics pipelines are on the same chip. Perego teaches determining the pixels within a set of tiles of the repeating pattern which are to be processed by corresponding graphics pipeline on the same chip in response to pixel data (col. 5, lines 19-44).

Claim Rejections - 35 USC § 103

9. 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.

10. The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

11. Claims 1-4, 7, 10, 12, 14, and 25 are rejected under 35 U.S.C. 103(a) as being unpatentable over MacInnis (US006570579B1) in view of Perego (US006864896B2).

12. As per Claim 1, MacInnis teaches a graphics processing circuit, comprising: a graphics pipeline (58, Fig. 2) on a chip (10); a memory controller (54) on the chip (10), as shown in Fig. 2 (col. 4, lines 65-67; col. 5, lines 36-41), in communication with the graphics pipeline (58),

operative to transfer pixel data between the pipeline (58) and a memory (col. 6, lines 10-13, 59-66).

However, MacInnis does not teach at least two graphics pipelines on the same chip operative to process data in a corresponding set of tiles of a repeating tile pattern corresponding to screen locations, a respective one of the at least two graphics pipelines operative to process data in a dedicated tile; wherein the repeating tile pattern includes a horizontally and vertically repeating pattern of square regions. However, Perego teaches graphics processing circuit (300, Fig. 3; col. 3, lines 61-63) having at least 2 graphics pipelines (312) operative to process data in corresponding set of tiles of repeating tile pattern corresponding to screen locations, respective one of at least 2 graphics pipelines operative to process data in dedicated tile (col. 5, lines 19-27, 38-44); and memory controller (310, Fig. 3) in communication with at least 2 graphics pipelines 312, operative to transfer pixel data between each of 1st pipeline and 2nd pipeline and shared memories 314 (col. 3, lines 65-67; col. 4, lines 1-10, 48-65). Shared memories 314 are each part of main memory (col. 1, lines 44-54; col. 3, lines 3-6), and so are considered to be one memory. Repeating tile pattern includes horizontally and vertically repeating pattern of regions of square regions, as shown in Fig. 5 (col. 5, lines 19-27, 38-44). Perego describes "Improvements in integrated circuit design and manufacturing technologies allow higher levels of integration, thereby allowing an increasing number of subsystems to be integrated into a single device. This increased integration reduces the total number of components in a system, such as a computer system. As subsystems with high memory performance requirements (such as graphics subsystems) are combined with the traditional main memory controller, the resulting architecture may provide a single high-performance main memory interface" (col. 1, lines 34-43). Perego

shows in Fig. 8 that in one embodiment, a memory module 800 contains two different rendering engines 802 and 810 (Fig. 8; col. 6, lines 61-62), and therefore at least two graphics pipelines (802, 810) are on a same memory module 800. From Fig. 8 and from the description in Perego, one of ordinary skill in the art would understand that a memory module is equivalent to a chip. Therefore, Perego teaches that two graphics pipelines are on the same chip.

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify MacInnis to include at least two graphics pipelines on the same chip operative to process data in a corresponding set of tiles of a repeating tile pattern corresponding to screen locations, a respective one of the at least two graphics pipelines operative to process data in a dedicated tile; wherein the repeating tile pattern includes a horizontally and vertically repeating pattern of square regions because Perego suggests that this parallel processing significantly reduces the processing burden on the memory controller/graphics controller (col. 5, lines 38-46).

13. As per Claim 2, MacInnis does not teaches that the square regions comprise a two dimensional partitioning of memory. However, Perego teaches square regions have two dimensional partitioning of memory (col. 5, lines 19-33). This would be obvious for the reasons given in the rejection for Claim 1.

14. As per Claim 3, MacInnis teaches wherein the memory is a frame buffer (col. 6, line 66-col. 7, line 2).

15. As per Claim 4, MacInnis does not teach that each of the at least two graphics pipelines further includes front end circuitry operative to receive vertex data and generate pixel data corresponding to a primitive to be rendered, and back end circuitry, coupled to the front end

circuitry, operative to receive and process a portion of the pixel data. However, Perego teaches each of at least 2 graphics pipelines includes front end circuitry (308, Fig. 3) operative to generate pixel data corresponding to primitive to be rendered, and back end circuitry (312), coupled to front end circuitry, operative to receive and process portion of pixel data (col. 3, line 64-col. 4, line 2; col. 5, lines 19-44). In order for front end circuitry (308) to generate pixel data, it must receive vertex data. Perego teaches that the CPU sorts the primitive data according to the spatial region of the rendering surface (e.g., the x and y coordinates) covered by that primitive, and the rendering surface is divided into multiple rectangular regions of pixels (col. 5, lines 19-27). One of ordinary skill in the art would understand that data pertaining to the x and y coordinates covered by the primitive would include vertex data. The main memory is used to store data which are referenced during the execution of the programs (col. 1, lines 18-21). Therefore, the CPU is operative to receive vertex data (data pertaining to the x and y coordinates covered by the primitive) from the main memory and generate pixel data corresponding to a primitive to be rendered (col. 5, lines 19-27; col. 1, lines 18-21). Perego teaches that rendering engine 312 is coupled to the CPU, and is operative to receive and process a rectangular region of pixel data (col. 3, line 64-col. 4, line 2; col. 5, lines 19-44). Since the CPU is coupled to the rendering engine 312, and the CPU and the rendering engine 312 both perform graphics processing, and, the CPU performs the operations of the front end circuitry, and the rendering engine 312 performs the operations of the back end circuitry, the CPU and the rendering engine 312 are considered to be part of a graphics pipeline, and the CPU is considered to be the front end circuitry included in the graphics pipeline. This would be obvious for the reasons given in the rejection for Claim 1.

16. As per Claim 7, MacInnis does not teach that the at least two graphics pipelines separately receive the pixel data from the front end circuitry. However, Perego teaches at least two graphics pipelines (312) separately receive pixel data from front end circuitry (308) (col. 3, line 64-col. 4, line 2; col. 5, lines 19-44). This would be obvious for the reasons given in the rejection for Claim 1.

17. As per Claim 10, MacInnis does not teach that a first of the at least two graphics pipelines processes the pixel data only in the first set of tiles in the repeating tile pattern. However, Perego teaches first of at least two graphics pipelines (first rendering engine of 312, Fig. 3) processes pixel data only in first set of tiles (tiles labeled "RE0" in Fig. 5) in repeating tile pattern (col. 5, lines 23-44). This would be obvious for the reasons given in the rejection for Claim 1.

18. As per Claim 12, MacInnis does not teach that a second of the at least two graphics pipelines processes the data only in a second set of tiles in the repeating tile pattern. However, Perego teaches second of at least two graphics pipelines (second rendering engine of 312, Fig. 3) processes pixel data only in second set of tiles (tiles labeled "RE1" in Fig. 5) in repeating tile pattern (col. 5, lines 23-44). This would be obvious for the reasons given in the rejection for Claim 1.

19. As per Claim 14, Claim 14 is similar to Claims 4 and 10, except that Claim 14 is for a third and fourth graphics pipeline. Perego teaches four graphics pipelines (col. 5, lines 41-44). So Claim 14 is rejected under the same rationale as Claims 4 and 10.

20. As per Claim 25, Claim 25 is similar in scope to Claim 1, and therefore is rejected under the same rationale.

21. Claims 5, 18, and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over MacInnis (US006570579B1) in view of Perego (US006864896B2), further in view of Kelleher (US005794016A).

22. As per Claim 5, MacInnis and Perego are relied on for teachings for Claim 4.

But, MacInnis and Perego do not explicitly teach at each of 2 graphics pipelines includes scan converter, coupled to back end circuitry, operative to determine portion of pixel data to be processed by back end circuitry. But, Kelleher teaches each of at least 2 graphics pipelines (20A, 20B, Fig. 3; col. 3, lines 22-23; col. 4, lines 9-14) includes scan converter (update stage, Fig. 7), coupled to back end circuitry, operative to determine portion of pixel data to be processed by back end circuitry (col. 8, lines 52-61; col. 9, lines 1-23; col. 6, lines 26-28).

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify MacInnis and Perego so at each of at least two graphics pipelines includes a scan converter, coupled to the back end circuitry, operative to determine the portion of the pixel data to be processed by the back end circuitry because Kelleher suggests scan converters are needed in order to define image data as array of pixels by calculating pixel addresses (col. 9, lines 1-23), as is well-known in the art.

23. As per Claim 18, MacInnis does not teach a bridge operable to transmit vertex data to each of the 1st, 2nd, 3rd and 4th graphics pipelines. But, Kelleher teaches a bridge (38, Fig. 3) operative to transmit vertex data to each of the 1st (20A), 2nd (20B), 3rd (20C) and 4th (20N) graphics pipelines (col. 3, lines 22-23; col. 4, lines 9-14; col. 8, lines 56-65; col. 3, lines 46-50).

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify MacInnis to include a bridge operable to transmit vertex data to each of the

first, second, third and fourth graphics pipelines because Kelleher suggests being able to convert the vertex data to pixel data in parallel, which increases the efficiency of the graphics system (col. 2, lines 31-35; col. 8, lines 56-65; col. 9, lines 1-23).

24. As per Claim 24, MacInnis teaches a graphics processing circuit, comprising: front end circuitry (90, Fig. 4) operative to generate pixel data in response to primitive data for a primitive to be rendered (col. 8, lines 49-59); back end circuitry (98), coupled to the front end circuitry (90) (col. 9, lines 15-34), operative to process the pixel data in response to position coordinates (col. 9, lines 35-54). The front end circuitry (90) and the back end circuitry (98) are in the graphics display pipeline (80) (col. 7, lines 55-63), and the graphics display pipeline (80) is equivalent to display engine (58, Fig. 2), which is on graphics chip (10) (col. 4, lines 65-67; col. 5, lines 36-39). A memory controller (54) is also on graphics chip (10) (col. 4, lines 65-67; col. 5, lines 36-39). Therefore, the front end circuitry (90), the back end circuitry (98), and the memory controller (54) are on the chip (10). The memory controller (54) is coupled to the display engine (58) and is operative to transmit and receive the processed pixel data (col. 6, lines 10-13, 59-67; col. 7, lines 1-2). Since the display engine (58) is equivalent to the graphics display pipeline (80) which contains the back end circuitry (98) (col. 6, lines 59-67; col. 7, lines 55-63), the memory controller (54) is coupled to the back end circuitry (98) and is operative to transmit and receive the processed pixel data.

However, MacInnis does not teach first back end circuitry operative to process a first portion of the pixel data; set of tiles of a repeating tile pattern are to be processed by the first back end circuitry, the repeating tile pattern including a horizontally and vertically repeating pattern of square regions; second back end circuitry operative to process a second portion of the

pixel data; set of tiles of the repeating tile pattern are to be processed by the second back end circuitry. However, Perego teaches graphics processing circuit (300, Fig. 3; col. 3, lines 61-63), having front end circuitry (308) operative to generate pixel data in response to primitive data for primitive to be rendered (col. 5, lines. 19-23); 1st back end circuitry (1st rendering engine 312), coupled to front end circuitry 308, operative to process 1st portion of pixel data (labeled "RE0" in Fig. 5) in response to position coordinates; set of tiles of repeating tile pattern are to be processed by 1st back end circuitry, repeating tile pattern including horizontally and vertically repeating pattern of square regions, as shown in Fig. 5; 2nd back end circuitry (second rendering engine 312), coupled to front end circuitry 308, operative to process 2nd portion of pixel data (labeled "RE1" in Fig. 5) in response to position coordinates; set of tiles of repeating tile pattern are to be processed by 2nd back end circuitry (col. 3, line 63-col. 4, line 2; col. 5, lines 19-44); and memory controller (310), coupled to 1st and 2nd back end circuitry (312) operative to transmit and receive processed pixel data (col. 3, lines 65-67; col. 4, lines 1-53; col. 5, lines 32-44). Perego describes "Improvements in integrated circuit design and manufacturing technologies allow higher levels of integration, thereby allowing an increasing number of subsystems to be integrated into a single device. This increased integration reduces the total number of components in a system, such as a computer system. As subsystems with high memory performance requirements (such as graphics subsystems) are combined with the traditional main memory controller, the resulting architecture may provide a single high-performance main memory interface" (col. 1, lines 34-43). Perego shows in Fig. 8 that in one embodiment, a memory module 800 contains two different rendering engines 802 and 810 (Fig. 8; col. 6, lines 61-62), and therefore at least two graphics pipelines (802, 810) are on a same memory module

800. From Fig. 8 and from the description in Perego, one of ordinary skill in the art would understand that a memory module is equivalent to a chip. Therefore, Perego teaches that two graphics pipelines are on the same chip, and so the front end circuitry, first back end circuitry, and second back end circuitry are on the same chip. This would be obvious for the reasons given in the rejection for Claim 1.

However, MacInnis and Perego do not explicitly teach first scan converter and second scan converter. However, Kelleher teaches first scan converter, coupled between front end circuitry (14, Fig. 3) and first back end circuitry (update stage, Fig. 7 in 20A, Fig. 3), operative to determine which set of tiles of repeating tile pattern are to be processed by first back end circuitry (col. 3, lines 22-23; col. 8, line 33-col. 9, line 23), and operative to provide position coordinates to first back end circuitry in response to pixel data (col. 4, lines 60-62; col. 8, lines 52-65; col. 6, lines 36-38); second scan converter, coupled between front end circuitry and second back end circuitry (update stage, Fig. 7 in 20B, Fig. 3), operative to determine which set of tiles of repeating tile pattern are to be processed by second back end circuitry, and operative to provide position coordinates to second back end circuitry in response to pixel data (col. 3, lines 22-23; col. 8, line 33-col. 9, line 23; col. 4, lines 60-62; col. 8, lines 52-65; col. 6, lines 36-38). Therefore, by implement this teaching into the device of Perego, front end circuitry, first back end circuitry, and first scan converter of Perego-Kelleher combination make up one graphics pipeline, and front end circuitry, second back end circuitry, and second scan converter of Perego-Kelleher combination make up another graphics pipeline. Since Perego teaches graphics pipelines are on the same chip, the teaching from Kelleher can be applied to Perego so that the

first scan converter and the second scan converter are also on the same chip. This would be obvious for reasons given in the rejection for Claim 5.

25. Claims 6 and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over MacInnis (US006570579B1) in view of Perego (US006864896B2), further in view of Furtner (US006778177B1).

26. As per Claim 6, MacInnis and Perego are relied upon for the teachings as discussed above relative to Claim 1.

However, MacInnis and Perego do not expressly teach that each tile of the set of tiles further comprises a 16x16 pixel array. However, Furtner teaches that each tile of the set of tiles further comprises a 16x16 pixel array (col. 11, lines 45-48, 64-65).

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify MacInnis and Perego so each tile of set of tiles further has 16x16 pixel array because Furtner suggests depending on number of parallel image-rendering pipelines and depending on memory organization, optimum tile size and shape can be selected (col. 11, lines 45-48, 64-65), and so it would be obvious to modify tile size to be 16x16 pixels if that would be optimum tile size for particular number of parallel image-rendering pipelines and particular memory organization.

27. As per Claim 17, MacInnis does not teach 3rd and 4th graphics pipelines are on separate chips. However, Furtner teaches this limitation (col. 6, lines 47-51).

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify MacInnis so pipelines are on separate chips because Furtner teaches this

makes system more configurable by being able to easily add more graphics pipelines to increase performance (col. 6, lines 29-30, 42-51).

28. Claims 11, 13, 15, and 16 are rejected under 35 U.S.C. 103(a) as being unpatentable over MacInnis (US006570579B1) in view of Perego (US006864896B2), further in view of Kelleher (US005794016A), further in view of Hamburg (US005905506A).

MacInnis and Perego are relied upon for teachings relative to Claim 10.

However, MacInnis and Perego do not explicitly teach scan converter. However, Kelleher teaches first of the at least two graphics pipelines (20A, Fig. 3; col. 3, lines 22-23; col. 4, lines 9-14) further includes scan converter (84, Fig. 7), coupled to front end circuitry (80, 82) and back end circuitry (col. 8, line 52-col. 9, line 23). Scan converter determines which groups of blocks 52 within graphics memory 22 are allocated to and controlled by graphics pipelines (col. 8, lines 52-65; col. 6, lines 26-28). Graphics memory is partitioned into plurality of pixel blocks that are tiled in x-and y-direction of graphics memory (col. 4, lines 60-62). So, scan converter is inherently operative to provide memory addresses or position coordinates of pixels within first set of tiles to be processed by back end circuitry. This would be obvious for the reasons given in the rejection for Claim 5.

But, MacInnis, Perego, and Kelleher do not expressly teach using tile identification data to indicate which tiles are to be processed. But, Hamburg teaches pixel identification line for receiving tile identification data indicating which tiles are to be processed (col. 5, lines 35-52).

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify MacInnis, Perego, and Kelleher to include using tile identification data to indicate which tiles are to be processed because Hamburg suggests advantage of using tile

identification data to easily track storage locations of tile pixel data and being able to easily retrieve data for particular image tile (col. 1, lines 46-54).

29. Claim 19 is rejected under 35 U.S.C. 103(a) as being unpatentable over MacInnis (US006570579B1) in view of Perego (US006864896B2), further in view of Furtner (US006778177B1), further in view of Kent (US 20030164830A1).

MacInnis, Perego, and Furtner are relied upon for the teachings as discussed above relative to Claim 17. MacInnis teaches data includes polygon (col. 58, lines 50-54). Furtner teaches third and fourth graphics pipelines are on separate chips (col. 6, lines 47-51), as discussed for Claim 17.

But, MacInnis, Perego, and Furtner do not teach creating bounding box around polygon and each corner of bounding box is checked against super tile that belongs to each separate chip and if bounding box does not overlap any of super tiles associated with separate chip, then processing circuit rejects whole polygon and processes next one. But, Kent teaches graphics pipeline [0006] calculates bounding box of primitive and testing this against VisRect. If bounding box of primitive is contained in other P10's super tile the primitive is discarded at this stage [0129]. Primitive can be polygon [0088]. Method used is to calculate distance from each subpixel sample point in point's bounding box to point's center and compare this to point's radius. Subpixel sample points with distance greater than radius do not contribute to pixel's coverage. Cost of this is kept low by only allowing small radius points hence distance calculation is a small multiply and by taking a cycle per subpixel sample per pixel within bounding box [0144]. Since method calculates distance from each subpixel sample point in point's bounding box, this must include all corners of bounding box. So, Kent teaches data includes polygon and

graphics pipeline creates bounding box around polygon and wherein each corner of bounding box is checked against super tile that belongs to graphics pipeline and if bounding box does not overlap any of super tiles, then processing circuit rejects whole polygon and processes next one.

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify MacInnis, Perego, and Furtner to include bounding box as because Kent suggests processing super tiles one at a time in order to hide page break costs [0129, 0051].

30. Claims 20-22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Perego (US006864896B2).

31. As per Claim 20, Perego teaches graphics processing method, comprising generating pixel data (col. 5, lines 19-25), which is inherently generated in response to received vertex data. Perego teaches a shared front end (308, Fig. 3) that passes the pixel data to both of the two graphics pipelines (first rendering engine 312 and second rendering engine 312) (col. 3, line 63-col. 4, line 2; col. 5, lines 19-44), similarly as what is described in the instant specification, and so Perego teaches passing the same pixel data to both of the two graphics pipelines. Perego shows in Fig. 8 that in one embodiment, a memory module 800 contains two different rendering engines 802 and 810 (Fig. 8; col. 6, lines 61-62), and therefore at least two graphics pipelines (802, 810) are on a same memory module 800. From Fig. 8 and from the description in Perego, one of ordinary skill in the art would understand that a memory module is equivalent to a chip. Therefore, Perego teaches that two graphics pipelines are on the same chip. Perego teaches determining pixels within set of tiles of repeating tile pattern corresponding to screen locations to be processed by corresponding one of the at least two graphics pipelines (312, Fig. 3) in response to pixel data, repeating tile pattern including horizontally and vertically repeating pattern of

square regions, as shown in Fig. 5; performing pixel operations on pixels within determined set of tiles by corresponding one of at least two graphics pipelines (col. 5, lines 19-44); and transmitting processed pixels to memory controller 310, at least 2 graphics pipelines share memory controller (col. 3, line 65-col. 4, line 25; col. 5, lines 31-44). Perego describes "Improvements in integrated circuit design and manufacturing technologies allow higher levels of integration, thereby allowing an increasing number of subsystems to be integrated into a single device. This increased integration reduces the total number of components in a system" (col. 1, lines 34-39).

32. As per Claim 21, Perego teaches determining pixels within set of tiles of repeating tile pattern to be processed further comprises determining set of tiles that corresponding graphics pipeline is responsible for (col. 5, lines 19-50).

33. As per Claim 22, Perego teaches determining pixels within set of tiles of repeating tile pattern to be processed comprises providing position coordinates of pixels within determined set of tiles to be processed to corresponding one of at least 2 graphics pipelines (col. 5, lines 19-44).

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to JONI HSU whose telephone number is (571)272-7785. The examiner can normally be reached on M-F 8am-5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kee Tung can be reached on 571-272-7794. 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.

JH

/Joni Hsu/
Examiner, Art Unit 2628