345 / 200.77	Class	Subclass	ISSUE CLASSIFICATION
08/702911			

[redacted]

5812789

UTILITY SERIAL NUMBER 08/702911	PATENT DATE SEP 22 1998	PATENT NUMBER 5812789

SERIAL NUMBER	FILING DATE	CLASS	SUBCLASS	GROUP ART UNIT	EXAMINER
08/702,911	08/28/96	~~348~~ 395	~~200.18~~ 200.77	~~26__~~ 2756	E. RAMIREZ

APPLICANTS

RAUL Z. DIAZ, PALO ALTO, CA; JEFFERSON E. OWEN, FREMONT, CA.

345/200.77

CONTINUING DATA*******************
VERIFIED

No

FOREIGN/PCT APPLICATIONS**********
VERIFIED

No

FOREIGN FILING LICENSE GRANTED 10/29/96

Foreign priority claimed ☐ yes ☒ no 35 USC 119 conditions met ☐ yes ☒ no Verified and Acknowledged — Examiner's Initials	AS FILED →	STATE OR COUNTRY	SHEETS DRWGS.	TOTAL CLAIMS	INDEP. CLAIMS	FILING FEE RECEIVED	ATTORNEY'S DOCKET NO.
		CA	5	49	3	[illegible]	[illegible]

ADDRESS

LISA K JORGENSON
SGS THOMSON MICROELECTRONICS INC
1310 ELECTRONICS DRIVE
CARROLLTON TX 75006

TITLE

VIDEO AND/OR AUDIO DECOMPRESSION AND/OR COMPRESSION DEVICE THAT SHARES A MEMORY INTERFACE

U.S. DEPT. of COMMERCE • Patent and Trademark Office–PCT-436L (rev. 7-94)

PARTS OF APPLICATION FILED SEPARATELY		Applications Examiner 4/10/98
NOTICE OF ALLOWANCE MAILED: 3/30/98	Assistant Examiner	CLAIMS ALLOWED — Total Claims: 33; Print Claim: 1
ISSUE FEE — Amount Due: $1320.00; Date Paid: 7/2/98	ELLIS B. RAMIREZ PRIMARY EXAMINER Primary Examiner	DRAWING — Sheets Drwg.: 5; Figs. Drwg.: 6; Print Fig.: 2
Label Area	PREPARED FOR ISSUE	ISSUE BATCH NUMBER: C25

WARNING: The information disclosed herein may be restricted. Unauthorized disclosure may be prohibited by the United States Code Title 35, Sections 122, 181 and 368. Possession outside the U.S. Patent & Trademark Office is restricted to authorized employees and contractors only.

Form PTO-436A
(Rev. 8/92)

(FACE)

08/702911

PATENT APPLICATION

08702911

APPROVED FOR LICENSE
SEP 25 96
INITIALS

CONTENTS

Date Entered or Counted			Date Received or Mailed
	1.	Application 5 papers.	
	2.	IDS	11-4-96
	3.	IDS	11-19-96
	4.	Petition to Secure Filing (411)	7/3/97
	5.	Petition dismissed (411)	8/11/97
9.2.97	6.	BEJ (3)	9/3/97
	7.	Extension of Time (3)	3/9/98 cm 3/3/98
	8.	Amendment A	3/9/98 cm 3/3/98
	9.	Notice of Allowability	3/30/98
	10.	Supple. Notice of Allow	4/10/98
7/13/98	11.	Formal Drawings (5 shts) set 1	6/29/98
	12.	PTO Grant SEP 22 1998	
9/22/98	13.	Sup Dec	6/29/98
	14.		
	15.		
	16.		
	17.		
	18.		
	19.		
	20.		
	21.		
	22.		
	23.		
	24.		
	25.		
	26.		
	27.		
	28.		
	29.		
	30.		
	31.		
	32.		

(FRONT)

Staple Issue Slip Here

POSITION		ID NO.	DATE
CLASSIFIER		12	9/25/96
EXAMINER		230	10-[illegible]
TYPIST	Strunk	318	10/29/96
VERIFIER		446	10/[illegible]
CORPS CORR.			
SPEC. HAND			
FILE MAINT.			
DRAFTING			

445

INDEX OF CLAIMS

Claim Final	Claim Original	Date [illegible]	Date 3/26/98						
	1	✓	✓						
	2		✓						
	3		✓						
	4		✓						
	5		✓						
	6		✓						
	7		✓						
	8		✓						
	9		✓						
	10		✓						
	11		✓						
	12		✓						
	13		✓						
1	14		=						
2	15								
3	16								
4	17								
5	18								
6	19								
7	20								
8	21								
9	22								
10	23								
11	24								
12	25								
13	26								
14	27								
15	28								
16	29								
17	30								
18	31								
19	32								
20	33								
21	34								
22	35								
23	36								
24	37								
25	38								
26	39								
27	40								
28	41		=						
	42		✓						
	43		✓						
29	44		✓						
30	45		=						
31	46								
32	47								
33	48		=						
	49		✓						
	50								

SYMBOLS

✓ .... Rejected
= .... Allowed
- (Through numberal) Canceled
\+ .... Restricted
N .... Non-elected
I .... Interference
A .... Appeal
O .... Objected

Claim Final	Claim Original	Date							
	51								
	52								
	53								
	54								
	55								
	56								
	57								
	58								
	59								
	60								
	61								
	62								
	63								
	64								
	65								
	66								
	67								
	68								
	69								
	70								
	71								
	72								
	73								
	74								
	75								
	76								
	77								
	78								
	79								
	80								
	81								
	82								
	83								
	84								
	85								
	86								
	87								
	88								
	89								
	90								
	91								
	92								
	93								
	94								
	95								
	96								
	97								
	98								
	99								
	100								

(LEFT INSIDE)

PATENT NUMBER	ORIGINAL CLASSIFICATION				
	CLASS 345	SUBCLASS 100.77			
APPLICATION SERIAL NUMBER 08/[illegible],911	CROSS REFERENCE(S)				
	CLASS	SUBCLASS (ONE SUBCLASS PER BLOCK)			
APPLICANT'S NAME (PLEASE PRINT) Diaz et al					
IF REISSUE, ORIGINAL PATENT NUMBER					

INTERNATIONAL CLASSIFICATION					
G	0	6	F		17/00
					/
					/
					/

GROUP ART UNIT 2756

ASSISTANT EXAMINER (PLEASE STAMP OR PRINT FULL NAME)

PRIMARY EXAMINER (PLEASE STAMP OR PRINT FULL NAME) Ellis B. Ramirez

PTO 270 (REV. 5-91)

ISSUE CLASSIFICATION SLIP

U.S. DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE

[redacted]

SEARCHED

Class	Sub.	Date	Exmr.
395	200.77 200.82 507 890 729	8/20/07	Q
348	402 407 10	8/20/07	Q
	Update	3/26/08	Q

SEARCH NOTES

	Date	Exmr.

INTERFERENCE SEARCHED

Class	Sub.	Date	Exmr.
395	200.77	3/26/08	Q

(RIGHT OUTSIDE)

PATENT APPLICATION SERIAL NO. 08/702911

U.S. DEPARTMENT OF COMMERCE
PATENT AND TRADEMARK OFFICE
<u>FEE RECORD SHEET</u>

340 BA 19-1353 10/28/96 08702911
34011 101 188.00CH

340 TL 10/02/96 08702911
1 101 1,356.00 CK

PTO-1556
(5/87)

08/702911

MAIL ROOM AUG 24 1996 PAT. & TM OFFICE

MAIL ROOM AUG 26 1996 PAT. & TM OFFICE

SGS-THOMSON MICROELECTRONICS

August 23, 1996

THE ASSISTANT COMMISSIONER FOR PATENTS
Washington, D.C. 20231

Re: Inventor(s): Raul Z. Diaz and Jefferson E. Owen

For: Video and/or Audio Decompression and/or Compression Device that Shares a Memory Interface

Our File No: 96-S-11

Sir:

Enclosed with this transmittal letter are:

(1) Subject patent application with Declaration and Power of Attorney;
(2) Five (5) sheets of informal drawings;
(3) Certificate of Express Mail;
(4) Assignment and Recordation Cover Sheet;
(5) Check in the amount of $1,396.00;
(6) Return postcard which we would appreciate your date stamping and returning to us upon receipt;

The total filing fee has been calculated as follows:

Basic fee	=	$ 750.00
Recordation of Assignment	=	40.00
29 claims in excess of 20	=	638.00
2 independent claim in excess of 3	=	156.00
Total filing fee	=	$1.584.00

I authorize the Commissioner to charge any additional fees which may be required, or credit any overpayment to Account No. 19-1353. A duplicate copy of this sheet is enclosed.

Irena Lager
Reg. No. 39,260

SGS-THOMSON Microelectronics, Inc. • 1310 Electronics Drive • Carrollton, TX 75006-5039
Telephone (214) 466-6000, Telex 730643

ST002-CAR

08/702911

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Docket No. 96-S-011

In Re Application of:

Raul Z. Diaz and Jefferson E. Owen

For: **Video and/or Audio Decompression and/or Compression Device that Shares a Memory Interface**

CERTIFICATE OF EXPRESS MAIL

"EXPRESS MAIL" NO. EG947362259US

Date of Deposit: August 23, 1996

I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Assistant Commissioner for Patents, Box Patent Application, Washington, D.C. 20231.

Signature of person mailing paper or fee

VIDEO AND/OR AUDIO DECOMPRESSION AND/OR COMPRESSION DEVICE THAT SHARES A MEMORY INTERFACE

Cross-reference to Related Applications

This application contains some text and drawings in common with pending U.S. Patent Applications entitled: "Video and/or Audio Decompression and/or Compression Device that Shares a Memory" by Jefferson E. Owen, Raul Z. Diaz, and Osvaldo Colavin S/N 08/702,910 Filed on 08/26/96 (~~Attorney's Docket No. 96-S-01~~2), and has the same effective filing date and ownership as the present application, and to that extent is related to the present application, which is incorporated herein by reference.

Background

The present invention relates to the field of electronic systems having a video and/or audio decompression and/or compression device, and is more specifically directed to sharing a memory interface between a video and/or audio decompression and/or compression device and another device contained in the electronic system.

The size of a digital representation of uncompressed video images is dependent on the resolution, and color depth of the image. A movie composed of a sequence of such images, and the audio signals that go along with them, quickly becomes large enough so that uncompressed such a movie typically cannot fit entirely onto conventional recording medium, such as a CD. It is also typically now prohibitively expensive to transmit such a movie uncompressed.

It is therefore advantageous to compress video and audio sequences before they are transmitted or stored. A great deal of effort is being expanded to develop systems to compress these sequences. There are several coding standards currently used that are based on the discrete cosine transfer algorithm including MPEG-1, MPEG-2, H.261, and H.263. (MPEG stands for "Motion Picture Expert Group", a committee of the International Organization for Standardization, ISO.) The MPEG-1, MPEG-2, H.261, and H.263 standards are decompression protocols that describe how an encoded bitstream is to be decoded. The encoding can be done in any manner, as long as the resulting bitstream complies with the standard.

Video and/or audio compression devices (hereinafter encoders) are used to encode the video and/or audio sequence before it is transmitted or stored. The resulting bitstream is decoded by a video and/or audio decompression device (hereinafter decoder) before the video and/or audio sequence is displayed. However, a bitstream can only be decoded by a decoder if it complies to the standard used by the decoder. To be able to decode the bitstream on a large number of systems it is advantageous to encode the video and/or audio sequences to comply to a well accepted decompression standard. The MPEG standards are currently well accepted standards for one way communication. H.261, and H.263 are currently well accepted standards for video telephony.

Once decoded the images can be displayed on an electronic system dedicated to displaying video and audio, such as television or digital video disk (DVD) player, or on electronic systems where image display is just one feature of the system, such as a computer. A decoder needs to be added to these systems to allow them to display compressed sequences, such as received images and associated audio, or ones taken from a storage device. An encoder needs to be added to allow the system to compress video and/or audio sequences, to be transmitted or stored. Both need to be added for two way communication such as video telephony.

A typical decoder, such as an MPEG decoder 10 shown in Figure 1a, contains video decoding circuitry 12, audio decoding circuitry 14, a microcontroller 16, and a memory interface 18. The decoder can also contain other circuitry depending on the electronic system the decoder is designed to operate in. For example, when the decoder is designed to operate in a typical television the decoder will also contain an on screen display (OSD) circuit.

Figure 1b shows a better decoder architecture, used in the STi3520 and STi3520A MPEG Audio/MPEG-2 Video Integrated Decoder manufactured by SGS-THOMSON Microelectronics. The decoder has a register interface 20 instead of a microcontroller. The register interface 20 is coupled to an external microcontroller 24. The use of a register interface 20 makes it possible to tailor the decoder 10 to the specific hardware the decoder 10 interfaces with or change its operation without having to replace the decoder by just reprogramming the register interface. It also allows the user to replace the microcontroller 24, to upgrade or tailor the microcontroller 24 to a specific use, by just replacing the microcontroller and reprogramming the register interface 20, without having to replace the decoder 10.

The memory interface 18 is coupled to a memory 22. A typical MPEG decoder 10 requires 16 Mbits of memory to operate in the main profile at main level mode (MP at ML). This typically means that the decoder requires a 2Mbyte memory. Memory 22 is dedicated to the MPEG decoder 10 and increases the price of adding a decoder 10 to the electronic system. In current technology the cost of this additional dedicated memory 22 can be a significant percentage of the cost of the decoder.

An encoder also requires a memory interface 18 and dedicated memory. Adding the encoder to an electronic system again increases the price of the system by both the price of the encoder and its dedicated memory.

A goal in the semiconductor industry is to reduce the die area of an integrated circuit device for a given functionality. Some advantages of reducing the die area is the increase in the number of the die that can be manufactured on same size silicon wafer, and the reduction in price per die resulting therefrom. This results in both an increase in volume and reduction in price of the device.

Many of the functional circuits described above for Figure 1a and Figure 1b take up a lot of die space. However, each of them is needed to make the respective decoder operate.

Figure 1c shows a computer 25 containing a decoder 10, a main memory 168 and other typical components such as a modem 199, and graphics accelerator 188. The decoder 10 and the rest of the components are coupled to the core logic chipset 190 through a bus 170. The bus is typically a PCI (peripheral component interface) or ISA (industry standard architecture) bus, and each component contains an appropriate interface for interfacing with the bus.

When any component needs access to the memory 168 either to read from or write to the main memory 168, it generates a request which is placed on the bus 26. When the request is a write the data to be written is also placed on the bus 26. The request is processed in the core logic chipset 190 and the data is then either written to or read from the main memory 168. When data is read from the main memory 168 the data is now placed on the bus and goes to the component that requested the read.

There are typically many components in the computer systems that may require access to the main memory 168, and they are typically all coupled to the same bus 174, or possibly several buses 170, ~~188~~ 198 connected together by a PCI bridge 192, if there are not enough connectors on one bus to accommodate all of the peripherals. However, the addition of each bus is very expensive. Each request is typically processed according to a priority scheme. The priority scheme is typically based on the priority given to the device and the order in which the requests are received. Typically, the priority scheme is set up so no device monopolizes the bus, starving all of the other devices. Good practice suggests that no device on the bus require more than approximately 50% of the bus's bandwidth.

The minimum bandwidth required for the decoder 10 can be calculated based on the characteristics and desired operation of the decoder. These characteristics include the standard to which the bitstream is encoded to comply with, whether the decoder is to operate in real time, to what extent frames are dropped, and how the images are stored. Additionally, the latency of the bus that couples the decoder to the memory should be considered.

If the decoder does not operate in real time the decoded movie would stop periodically between images until the decoder can get access to the memory to process the next image. The movie may stop quite often between images and wait.

To reduce the minimum required bandwidth and still operate in real time, the decoder 10 may need to drop ~~frame~~ frames. If the decoder 10 regularly does not decode every frame then it may not need to stop between images. However, this produces very poor continuity in the images. This is problematic with an image encoded to the MPEG-1 or MPEG-2 standard~~s~~, or any standards that uses temporal compression. In temporal (interpicture) compression some of the images are decoded based on previous images and some based on previous and future images. Dropping an image on which the decoding of other images is based is unacceptable, and will result in many poor or even completely unrecognizable images.

The computer can also contain both a decoder and encoder to allow for video telephony, as described above. In this case not operating in real time would mean that the length of time between the occurrence of an event, such as speaking, at one end of the conversation until the event is displayed at the other end of the conversation is increased by the time both the encoder and then the decoder must wait to get access to the bus and the main memory. Not being able to operate in real time means that there would be gaps in the conversation until the equipment can catch up. This increases the time needed to have a video conference, and makes the conference uncomfortable for the participants.

One widely used solution to allow a component in a computer system to operate in real time is to give the component its own dedicated memory. Thus, as shown in Figure 1c, the decoder 10 can be given its own dedicated memory 22, with a dedicated bus 26 to connect the decoder 10 to its memory 22. The dedicated memory 22, its controller and the pins to control this memory significantly increase the cost of adding a decoder 10 to the computer.

Summary of the Invention

The present application discloses an electronic system that contains a first device and video and/or audio decompression and/or compression device capable of operating in real time. Both the first device and the video and/or audio decompression and/or compression device require a memory interface. The video and/or audio decompression and/or compression device shares a memory interface and the memory with the first device. In the preferred embodiment of the invention the shared memory interface contains an arbiter. The arbiter and DMA engines of the video and/or audio decompression and/or compression device and of the first device are configured to arbitrate between the two devices when one of them is requesting access to the memory. This allows the use of one memory interface to control the access of both the video and/or audio decompression and/or compression device and the first device to the memory.

When the video and/or audio decompression and/or compression device used in an electronic system, such as a computer, already containing a device that has a memory interface the video and/or audio decompression and/or compression device can share that memory interface and the memory of the device and the memory interface and memory of the video and/or audio decompression and/or compression device can be eliminated. Eliminating this memory interface reduces the die area without changing the critical dimensions of the device. Therefore increasing the volume and reducing the cost of the decoder or encoder. Eliminating the memory greatly reduces the cost of adding the video and/or audio decompression and/or compression device to the electronic system while not requiring the video and/or audio decompression and/or compression device to be connected to the system bus, allowing the video and/or audio decompression and/or compression device to operate in real time.

An advantage of the present invention is significant cost reduction due to the fact that the video and/or audio decompression and/or compression device does not need its own dedicated memory but can share a memory with another device and still operate in real time.

Another significant advantage of the present invention is that the die space needed for the video and/or audio decompression and/or compression device is smaller because the memory interface on the video and/or audio decompression and/or compression device is eliminated.

A further advantage of the present invention is that the video and/or audio decompression and/or compression device can share the memory of the device with which it is sharing the memory interface more efficiently.

Another advantage of the present invention is that the cost of producing a video and/or audio decompression and/or compression device is reduced because the memory interface on the video and/or audio decompression and/or compression device is eliminated.

Another advantage of the present invention is that the video and/or audio decompression and/or compression device can be monolithically integrated into the first device and no extra packaging or pins are needed for the video and/or audio decompression and/or compression device, and no pins are needed for the first device to connect to the video and/or audio decompression and/or compression device, saving pins on both devices and producing a better connection between the two devices.

Other advantages and objects of the invention will be apparent to those of ordinary skill in the art having reference to the following specification together with the drawings.

Brief Description of the Drawings

Figure 1a and 1b are electrical diagrams, in block form, of prior art decoders.

Figure 1c is an electrical diagram, in block form, of a computer system containing a decoder according to the prior art.

Figure 2 is an electrical diagram, in block form, of an electronic system containing a device having a memory interface and an encoder and decoder.

Figure 3 is an electrical diagram, in block form, of a computer system containing a core logic chipset designed for the CPU to share a memory interface with an encoder and decoder.

Figure 4 is an electrical diagram, in block form, of a computer system containing a graphics accelerator designed to share a memory interface with an encoder and/or decoder.

Detailed Description of the Preferred Embodiment

Figure 2 shows an electronic system 40 containing a first device 42 having access to a memory 50 through a memory interface 48, and a decoder 44 and encoder 46, having access to the same memory 50 through the same memory interface 48. First device 42 can be a processor, a core logic chipset, a graphics accelerator, or any other device that requires access to the memory 50, and either contains or is coupled to a memory interface. Any parts common to Figures 1 through 4 are indicated using the same numbering system. In the preferred embodiment of the invention, electronic system 40 contains a first device 42, a decoder 44, an encoder 46, a memory interface 48, and a memory 50. Although, either the decoder 44 or encoder 46 can be used in the decoder/encoder 45 without the other. For ease of reference, a video and/or audio decompression and/or compression device 45 will hereinafter be referred to as decoder/encoder 45. The decoder/encoder 45 may be a single device, or cell on an integrated circuit, or may be two separate devices, or cells in an integrated circuit. In the preferred embodiment of the invention, the first device 42, decoder/encoder 45, and memory interface 48 are on one integrated circuit, however, they can be on separate integrated circuits in any combination.

The decoder 44 includes a video decoding ~~12 circuit~~ circuit 12 and an audio decoding circuit 14, both coupled to a register interface 20. The decoder 44 can be either a video and audio decoder, just a video, or just an audio decoder. If the decoder 44 is just a video decoder it does not contain the audio decoding circuitry 14. The audio decoding can be performed by a separate audio codec coupled to the first device 42, or through software. In the preferred embodiment of the invention, when the decoder/encoder 45 is in a system containing a processor and is coupled to the processor, the audio decoding is performed in software. This frees up space on the die without causing significant delay in the decoding. If the audio decoding is performed in software, the processor should preferably operate at a speed to allow the audio decoding to be performed in real time without starving other components of the system that may need to utilize the processor. For example, currently software to perform AC-3 audio decoding takes up approximately 40% of the bandwidth of a 133 MHz Pentium. The encoder 46 includes a video encoding circuit 62 and an audio encoding circuit 64, both coupled to a register interface 20. The encoder 46 can be either a video and audio encoder, just a video, or just an audio encoder. If the encoder 46 is just a video encoder, it does not contain the audio encoding circuitry 64. The audio encoding can be performed by a separate audio codec coupled to the first device 42, or through software. In the preferred embodiment of the invention, when the decoder/encoder 45 is in a system containing a processor and is coupled to the processor, the audio encoding is performed in software presenting ~~software. Presenting~~ the same advantages of freeing up space on the die without causing significant delay in the encoding. The register interfaces 20 of the decoder 44 and encoder 46 are coupled to a processor.

The decoder 44 and encoder 46 are coupled to the direct memory access (DMA) engine 52. The decoder and encoder can be coupled to the same DMA engine as shown in Figure 2, or each can have its own DMA engine, or share a DMA engine with another device. When the decoder/encoder 45 are two separate devices or cells, decoder 44 and encoder 46 can still be coupled to one DMA engine 52. When the decoder/encoder is one device or is one cell on an integrated circuit, the DMA engine 52 can be part of the decoder/encoder 45, as shown in Figure 2. The DMA engine 52 is coupled to the arbiter 54 of the memory interface 48.

The first device 42 also contains a DMA engine 60. The DMA engine 60 of the first device 42 is coupled to the arbiter 54 of the memory interface 48. The arbiter is also coupled to the refresh logic 58 and the memory controller 56. The memory interface 48 is coupled to a memory 50. The memory controller 56 is the control logic that generates the address the memory interface 48 ~~access~~ accesses in the memory 50 and the timing of the burst cycles.

In current technology, memory 50 is typically a DRAM. However, other types of memory can be used. The refresh logic 58 is needed to refresh the DRAM. However, as is known in the art, if a different memory is used, the refresh logic 58 may not be needed and can be eliminated.

The decoder/encoder 45 is coupled to the memory 50 through devices, typically a bus 70, that have a bandwidth greater than the bandwidth required for the decoder/encoder 45 to operate in real time. The minimum bandwidth required for the decoder/encoder 45 can be calculated based on the characteristics and desired operation of the decoder, including the standard to which the bitstream is encoded to comply with, whether the decoder/encoder 45 is to operate in real time, to what extent frames are dropped, and which images are stored. Additionally, the latency of the bus 70 that couples the decoder/encoder 45 to the memory 50 should be considered.

A goal is to have the decoder/encoder 45 operate in real time without dropping so many frames that it becomes noticeable to the human viewer of the movie. To operate in real time the decoder/encoder 45 should decoder and/or encode images fast enough so that any delay in decoding and/or encoding cannot be detected by a human viewer. This means that the decoder/encoder 45 has a required bandwidth that allows the decoder/encoder 45 to operate fast enough to decode the entire image in the time between screen refreshes, which is typically 1/30 of a second, with the human viewer not being able to detect any delay in the decoding and/ or encoding. To operate in real time the required bandwidth should be lower than the bandwidth of the bus. In order not to starve the other components on the bus, i.e. deny these components access to the memory for an amount of time that would interfere with their operation, this required bandwidth should be less the entire bandwidth of the bus. Therefore a fast bus 70 should be used. A fast bus 70 is any bus whose bandwidth is equal to or greater that the required bandwidth. There are busses, in current technology, including the ISA bus, whose bandwidth is significantly below the bandwidth required for this.

In the preferred embodiment of the invention the decoder/encoder 45 is coupled to the memory 50 through a fast bus 70 that has a bandwidth of at least the bandwidth required for the decoder/encoder 45 to operate in real time, a threshold bandwidth. Preferably the fast bus 70 has a bandwidth of at least approximately twice the bandwidth required for the decoder/encoder 45 to operate in real time. In the preferred embodiment the fast bus 70 is a memory bus, however any bus having the required bandwidth can be used.

The decoder/encoder 45 only requires access to the memory during operation. Therefore, when there is no need to decode or encode, the first device 42, and any other devices sharing the memory 50 have exclusive access to the memory, and can use the entire bandwidth of the fast bus 70.

In the preferred embodiment, even during decoding and encoding the decoder/encoder 45 does not always use the entire required bandwidth. Since the fast bus 70 has a bandwidth a little less than twice the required bandwidth the decoder/encoder 45 uses at most 60% of the bandwidth of the fast bus 70.

The required bandwidth is determined based on the size and resolution of the image, and the type of frame (I, P, or B). In the preferred embodiment the decoder/encoder typically will be using less than 40% of the bandwidth of the fast bus 70. This frees up the remaining bandwidth to be used by the other devices the decoder/encoder 45 is sharing the memory 50 with.

The decoder/encoder 45 can decode a bitstream formatted according to any one or a combination of standards. In the preferred embodiment of the invention the decoder/encoder 45 is a multi-standard decoder/encoder capable of decoding and encoding sequences formatted to comply to several well accepted standards. This allows the decoder/encoder 45 to be able to decode a large number of video and/or audio sequences. The choice of which standards the decoder/encoder 45 is capable of decoding bitstreams formatted to and of encoding sequences to comply to is based on the desired cost, efficiency, and application of the decoder/encoder 45.

In the preferred embodiment, these standards are capable of both intrapicture compression and interpicture compression. In intrapicture compression the redundancy within the image is eliminated. In interpicture compression the redundancy between two images are eliminated and only the difference information is transferred. This requires the decoder/encoder 45 to have access to the previous or future image that contains information needed to decode or encode the current image. These precious and/or future images need to be stored then used to decode the current image. This is one of the reasons the decoder/encoder 45 requires access to the memory, and requires a large bandwidth. The MPEG-1 and MPEG-2 standards allow for decoding based on both previous images and/or future images. Therefore for a decoder/encoder 45 capable of operating in real time to be able to comply with the MPEG-1 and MPEG-2 standards it should be able to access two images, a previous and a future image, fast enough to decode the current image in the 1/30 of a second between screen refreshes.

An MPEG environment is asymmetrical; there are much fewer encoders than decoders. The encoders are very difficult and expensive to manufacture and the decoders are comparatively easy and cheap. This encourages many more decoders than encoders, with the encoders in centralized locations, and decoders available such that every end user can have a decoder. Therefore, there are many receivers but few transmitters.

For video telephony and teleconferencing each end user has to be able to both receive and transmit. H.261, and H.263 are currently well accepted standards for video telephony. An encoder that can encode sequences to comply to the H.261 and H.263 standards is less complicated, having a lower resolution and lower frame rate than an encoder that complies to the MPEG-1 or MPEG-2 ~~standards. Possibly~~ standards, possibly making the quality of the decoded images somewhat lower than those from an encoder that complies with the MPEG-1 or MPEG-2 standards. Such an encoder, since it should be inexpensive and operate in real time, is also less efficient than an encoder to encode sequences to comply to the MPEG-1 or MPEG-2 standards. ~~Meaning that compression~~ This means that the compression factor, which is the ratio between the source data rate and the encoded bitstream data rate, of such an encoder is lower for a given image quality than the compression factor of an MPEG encoder. However, because such an encoder is less ~~complicated is it~~ complicated, it is much cheaper and faster than an encoder capable of complying with the MPEG-1 and/or MPEG-2 standards.

This makes video telephony possible, since both a long delay in encoding the signal and a cost that is prohibitively expensive for many users is unacceptable in video telephony.

In the preferred embodiment, the decoder/encoder 45 is capable of decoding a bitstream formatted to comply to the MPEG-1, MPEG-2, H.261, and H.263 standards, and encoding a sequence to produce a bitstream to comply to the H.261, and H.263 standards. This allows the decoder/encoder 45 ~~to be able~~ to be used for video telephony. Having the encoding comply ~~The encoding to comply~~ to the H.261 and H.263 standards but not the MPEG-1 and MPEG-2 standards balances the desire to reduce the cost of transmission and storage by encoding to produce the highest compression factor and the desire to keep cost low enough to be able to mass market the device.

The decoder/encoder 45 is preferably monolithically integrated into the first device as shown in Figure 3 and Figure 4. In Figure 3 the decoder/encoder 45 is integrated into a core logic chipset 150. In Figure 4 the decoder/encoder 45 is integrated into a graphics accelerator 200. Although, the decoder/encoder 45 can be separate from the first device 42, as shown in Figure 2.

Figure 3 shows a computer where the decoder/encoder 45 and the memory interface 48 are integrated into a core logic chipset 150. The core logic chipset 150 can be any core logic chipset known in the art. In the embodiment shown in Figure 3 the core logic chipset 150 is a PCI core logic chipset 150, which contains a PCI core logic device 158, the processor interface 154, and bus interfaces 156 for any system busses 170 to which it is coupled. The core logic chipset 150 can also contain a accelerated graphics port (AGP) 160 if a graphics accelerator 200 is present in the computer, and an enhanced integrated device electronics (EIDE) interface 186. The core logic chipset 150 is coupled to a processor 152, peripherals, such as a hard disk drive 164 and a DVD CD-ROM 166, a bus, such as a PCI bus 170, and a main memory 168.

In this embodiment, the main memory 168 is the memory 50 to which the memory interface 48 is coupled to. The main memory 168 is coupled to the memory interface 48 through a memory bus 167. In current technology the memory bus 167, which corresponds to the fast bus 70, for coupling a core logic chipset to a memory, is capable of having a bandwidth of approximately 400 Mbytes/s. This bandwidth is at least twice the bandwidth required for an optimized decoder/encoder 45, allowing the decoder/encoder 45 to operate in real time.

The core logic chipset 150 can also be coupled to cache memory 162 and a graphics accelerator 200 if one is present in the computer. The PCI bus 170 is also coupled to the graphics accelerator 200 and to other components, such as a local-area network (LAN) controller 172. The graphics accelerator 200 is coupled to a display 182, and a frame buffer 184. The graphics accelerator can also be coupled to an audio codec 180 for decoding and/or encoding audio signals.

Figure 4 shows a computer where the decoder/encoder 45 and the memory interface 48 are integrated into a graphics accelerator 200. The graphics accelerator 200 can be any graphics accelerator known in the art. In the embodiment shown in Figure 4, the graphics accelerator 200 contains a 2D accelerator 204, a 3D accelerator 206, a digital to analog converter 202, and bus interfaces 210 for any system busses 170 to which it is coupled. The graphics accelerator 200 can also contain an audio compressor/decompressor 208. The graphics accelerator 200 is coupled to a display 182, and a frame buffer 184.

In this embodiment, the frame buffer 184 is the memory 50 to which the memory interface 48 is coupled. The frame buffer 184 is coupled to the memory interface 48 through a memory bus 185. In this embodiment, memory bus 185 corresponds to the fast bus 70. In current technology the memory bus 185, for coupling a graphics accelerator to a memory, is capable of having a bandwidth of up to 400 Mbytes/s. This bandwidth is more that twice the bandwidth required for an optimized decoder/encoder 45. This allows the decoder/encoder 45 to operate in real time.

The graphics accelerator 200 can also be coupled to an audio codec 180 for decoding and/or encoding audio signals. The PCI bus 170 is also coupled to a chipset 190, and to other components, such as a LAN controller 172. In the present embodiment the chipset is a PCI chipset, although it can be any conventional chipset. The chipset 190 is coupled to a processor 152, main memory 168, and a PCI bridge 192. The PCI bridge bridges between the PCI bus 170 and the ISA bus 198. The ISA bus 198 is coupled to peripherals, such as a modem 199 and to an EIDE interface 186, which is coupled to other peripherals, such as a hard disk drive 164 and a DVD CD-ROM 166. Although, if the peripherals are compatible to the PCI bus the EIDE interface 186 can be integrated in to the PCI chipset 190 and the peripherals 164, 166 can be coupled directly to the PCI chipset, eliminating the PCI bridge 192 and the ISA bus 198.

Referring to Figure 2, the operation of the memory interface 48 during a memory request will now be described. During operation the decoder/encoder 45, the first device 42, and the refresh logic 58, if it is present, request access to the memory through the arbiter 54. There may also be other devices that request access to the memory 50 through this arbiter. The arbiter 54 determines which of the devices gets access to the memory 50. The decoder gets access to the memory in the first time interval and the first device gets access to the memory in the second time interval. The DMA engine 52 of the decoder/encoder 45 determines the priority of the decoder/encoder 45 for access to the memory 50 and of the burst length when the decoder/encoder 45 has access to the memory. The DMA engine 60 of the first device determines its priority for access to the memory 50 and the burst length when the first device 42 has access to the memory.

The decoder/encoder 45 or one of the other devices generates a request to access the memory 50. The request will be transferred to the arbiter 54. The state of the arbiter 54 is determined. The arbiter typically has three states. The first state is idle, when there is no device accessing the memory and there are no requests to access the memory. The second state is busy when there is a device accessing the memory and ~~the are no a request~~ there are no requests to access the memory. The third state is queue when there is a device accessing the memory and there is another request to access the memory.

It is also determined if two requests are issued simultaneously. This can be performed either ~~before of after~~ before or after determining the state of the arbiter. Access to the memory is determined according to the following chart.

F1-30X

Arbiter state	Simultaneous requests	Action
Idle	Yes	One of the requests gets access to the memory based on the priority scheme, and the other request is queued.
Busy	Yes	Both requests are queued in an order based on the priority scheme.
Queue	Yes	Both requests are queued in an order based on the priority scheme.
Idle	No	The device gets access to the memory.
Busy	No	The request is queued.
Queue	No	The requests are queued in an order based on the priority scheme.

The priority scheme can be any priority scheme that ensures that the decoder/encoder 45 gets access to the memory 50 often enough and for enough of a burst length to operate properly, yet not starve the other devices sharing the memory. The priority of the first device, device priority, and the priority of the decoder/encoder 45, decoder priority, is determined by the priority scheme. This can be accomplished in several ways.

To operate in real time, the decoder/encoder 45 has to decode an entire image in time to be able to display it the next time the screen is refreshed, which is typically every 1/30 of a second. The decoder/encoder 45 should get access to the memory to store and retrieve parts of this and/or of past and/or future images, depending on the decoding standard being used, often enough and for long enough burst lengths to be able to decode the entire image in the 1/30 of a second between screen refreshes.

There are many ways to this. One way to do this is to make the burst length of the first device, and any other device like the screen refresh that shares the memory and memory interface, [hereinafter sharing device] have short burst lengths, and to make sure that the same device is not the next device to get access to the memory when other devices have been waiting for a long time. Another way is to preempt the sharing device if its burst length exceeds a burst length threshold and again to make sure that the same device is not the next device to get access to the memory when other devices have been waiting for a long time. Preferably, when the preemption is used the sharing device would be preempted when its burst length exceeds 16 words. A third way is to limit the bandwidth available to the sharing devices, this way the decoder/encoder 45 always has enough bandwidth to operate in real time. Preferably the bandwidth of the sharing devices is limited only when the decoder/encoder 45 is operating. In the preferred embodiment a memory queue, such as a FIFO, in the decoder/encoder 45 generates an error signal when it falls below a data threshold. The error is sent to the CPU 152 and the CPU 152 can either shut down the system, drop ~~a image~~ an image frame or resume the decoding/encoding process.

There are also many ways to make sure that the same device is not the next device to get access to the memory when other devices have been waiting for a long time. This both ensures decoder/encoder 45 gets access to the memory 50 often enough, yet not starve the other devices sharing the memory. One way to do this is to disallow back to back requests. Another is to have shifting priority, where a particular request starts with a lower priority when first made and the priority increases with the length of time the request is in the queue, eventually reaching a priority above all of the other requests. In the preferred embodiment, the decoder/encoder 45 has a one clock cycle delay between requests to allow a sharing device to generate a request between the decoder/encoder requests.

In the preferred embodiment the burst length of the decoder is relatively short, approximately four to seventeen words. This allows the graphics accelerator more frequent access to the memory to ensure that the display is not disturbed by the sharing of the memory interface 48 and memory 50 when the decoder/encoder is in the graphics accelerator 200.

An electronic system 40, shown in Figure 2, containing the first device 42, the memory interface 48 coupled to a memory 50 and to the first device 42, a decoder/encoder 45 coupled to the memory interface 48, where the decoder/encoder 45 shares the memory interface 48 with the first device 42 provides several advantages. Referring to Figure 2 and Figure 1b simultaneously, the decoder 44, and encoder 46, according to the preferred embodiment of the invention do not need their own memory interfaces 18, as was needed in the prior art. Eliminating the memory interface 18 results in reducing the die size. This allows both a reduction in the price per die of the decoder, or encoder, and an increase in the volume of the product that can be produced.

Additionally, because the decoder/encoder 45 shares the memory interface 48 of the first device it also shares its memory 50. This eliminates the dedicated memory 22 that was necessary in the prior art for the decoder/encoder to operate in real time, resulting in significant reduction in the cost of the device. Allowing the decoder/encoder 45 to share the memory 50 with a first device 42 and to allow the decoder/encoder 45 to access the memory 50 through a fast bus 70 having a bandwidth of a least the bandwidth threshold permits the decoder/encoder to operate in real time. This allows the decoder/encoder to operate in real time and reduces stops between images and dropping frames to a point where both are practically eliminated. This produces better images, and eliminates any discontinuities and delays present in the prior art.

Furthermore, as the geometry used for devices decreases and the functionality of device increases the number of pads required in them increases. This at times requires the die size to be dictated by the number of pads and their configuration, leaving empty space on the die. This is typically the situation for core logic chipsets. In current technology, the pad requirements of a core logic chipset require the chipset to be one-third larger than required for the functional components of the core logic chipset. That means that one-third of the die space is empty. Incorporating the decoder/encoder 45 into the core logic chipset 150, as shown in Figure 3 provides the added advantage of effectively utilizing that space, without adding any extra pins to the core logic chipset 150. It also provides better connections between the decoder/encoder 45 and the core logic chipset 150.

Further background on compression can be found in: International Organization for Standards, INFORMATION TECHNOLOGY - CODING OF MOVING PICTURES AND ASSOCIATED AUDIO FOR DIGITAL STORAGE MEDIA AT UP TO ABOUT 1.5 MBITS/S, Parts 1-6, International Organization for Standards; International Standards Organization, INFORMATION TECHNOLOGY - GENERIC CODING OF MOVING PICTURES AND ASSOCIATED AUDIO INFORMATION, Parts 1-4, International

Organization for Standards; Datasheet "STi3500A" Datasheet of SGS-THOMSON Microelectronics; STi3500A - Advanced Information for an MPEG Audio/ MPEG-2 Video Integrated Decoder" (June 1995); Watkinson, John, COMPRESSION IN VIDEO AND AUDIO, Focal Press, 1995; Minoli, Daniel, VIDEO DIALTONE TECHNOLOGY, McGraw-Hill, Inc., 1995. Further background on computer architecture can be found in Anderson, Don and Tom Shanley, ISA SYSTEM ARCHITECTURE, 3rd ed., John Swindle ed., MindShare Inc., Addison-Wesley Publishing Co., 1995. All of the above references incorporated herein by reference.

While the invention has been specifically described with reference to several preferred embodiments, it will be understood by those of ordinary skill in the prior art having reference to the current specification and drawings that various modifications may be made and various alternatives are possible therein without departing from the spirit and scope of the invention.

For example:

Although the memory is described as DRAM the other types of memories including read-only memories, SRAMs, or FIFOs may be used without departing from the scope of the invention.

Any conventional decoder including a decoder complying to the MPEG-1, MPEG-2, H.261, or H.261 standards, or any combination of them, or any other conventional standard can be used as the decoder/encoder.

27

WE CLAIM:

1. An electronic system coupled to a memory and to a first device that requires access to the memory, the electronic system comprising:

a decoder that requires access to the memory sufficient to maintain real-time operation; and

a memory interface coupled to the decoder, for selectively providing access for the first device and the decoder to the memory.

2. The electronic system of claim 1, wherein the memory interface is coupled to the first device.

3. The electronic system of claim 1, wherein the memory interface is coupled to the memory.

4. The electronic system of claim 1, wherein the decoder comprises a video decoder.

5. The electronic system of claim 1, wherein the decoder is capable of decoding a bitstream formatted to comply with the MPEG-2 standard.

6. The electronic system of claim 1, wherein the memory interface further comprises an arbiter for selectively providing access for the first device and the decoder to the memory.

7. The electronic system of claim 1, further comprising an encoder coupled to the memory interface.

8. The electronic system of claim 7, wherein the decoder, the encoder and the memory interface are monolithically integrated into the first device.

9. The electronic system of claim 7, wherein the encoder is capable of producing a bitstream that complies with the H.263 standard.

10. The electronic system of claim 1, wherein the decoder and the memory interface are monolithically integrated into the first device.

11. The electronic system of claim 1, further comprising a fast bus coupled to the memory, to the decoder and to the first device.

12. The electronic system of claim 11, wherein the fast bus has a bandwidth of greater than a threshold bandwidth.

13. The electronic system of claim 11, wherein the fast bus comprises a memory bus.

14. An electronic system coupled to a memory, comprising:

a first device that requires access to the memory;

a decoder that requires access to the memory sufficient to maintain real time operation;

a fast bus coupled to the first device and the decoder; and

a memory interface for coupling to the memory, and coupled to the first device, and to the decoder, the memory interface having an arbiter for selectively providing access for the first device and the decoder to the memory.

15. The electronic system of claim 14, wherein:

the first device is capable of having a variable bandwidth; and

the decoder is capable of having a variable bandwidth.

16. The electronic system of claim 14, wherein the decoder comprises a video decoder.

17. The electronic system of claim 14, wherein the decoder is capable of decoding a bitstream formatted to comply with the MPEG-2 standard.

18. The electronic system of claim 14, further comprising an encoder coupled to the memory interface.

19. The electronic system of claim 18, wherein the decoder, the encoder and the memory interface are monolithically integrated into the first device.

20. The electronic system of claim 18, wherein the encoder is capable of producing a bitstream that complies with the H.263 standard.

21. The electronic system of claim 14, wherein the decoder and the memory interface are monolithically integrated into the first device.

22. The electronic system of claim 14, wherein the first device is a processor chipset.

23. The electronic system of claim 22, wherein the processor chipset is coupled to a processor.

24. The electronic system of claim 14, wherein the first device is a graphics accelerator.

25. The electronic system of claim 14, wherein the decoder is capable of decoding a bitstream formatted to comply with the MPEG-2 standard.

26. The electronic system of claim 14, ~~wherein the fast bus has a bandwidth of greater~~ than a threshold bandwidth.

27. The electronic system of claim 14, wherein the fast bus comprises a memory bus.

28. A computer comprising:

an input device;

an output device;

a memory;

a first device that requires access to the memory;

a decoder that requires access to the memory sufficient to maintain real time operation; and

a memory interface coupled to the memory, to the first device, and to the decoder, the memory interface having a means for selectively providing access for the first device and the decoder to the memory.

29. The computer of claim 28, wherein:

the first device is capable of having a variable bandwidth; and

the decoder is capable of having a variable bandwidth.

~~30~~ 17. The computer of claim ~~28~~ 15, wherein the decoder comprises a video decoder.

~~31~~ 18. The computer of claim ~~28~~ 15, wherein the decoder is capable of decoding a bitstream formatted to comply with the MPEG-2 standard.

~~32~~ 19. The computer of claim ~~28~~ 15, wherein the memory interface further comprises an arbiter for selectively providing access for the first device and the decoder to the memory.

~~33~~ 20. The computer of claim ~~28~~ 15, further comprising an encoder coupled to the memory interface.

~~34~~ 21. The computer of claim ~~33~~ 20, wherein the decoder, the encoder and the memory interface are monolithically integrated into the first device.

~~35~~ 22. The computer of claim ~~33~~ 20, wherein the encoder is capable of producing a bitstream that complies with the H.263 standard.

~~36~~ 23. The computer of claim ~~28~~ 15, wherein the decoder and the memory interface are monolithically integrated into the first device.

~~37~~ 24. The computer of claim ~~28~~ 15, wherein the first device is a processor chipset.

~~38~~ 25. The computer of claim ~~37~~ 24, wherein the processor chipset is coupled to a processor.

39. The computer of claim 28, wherein the first device is a graphics accelerator.

40. The computer of claim 28, wherein the decoder is capable of decoding a bitstream formatted to comply with the MPEG-2 standard.

41. The computer of claim 28, further comprising a fast bus coupled to the memory, to the decoder and to the first device.

42. The computer of claim 41, wherein the fast bus has a bandwidth of greater than a threshold bandwidth.

43. The electronic system of claim 41, wherein the fast bus comprises a memory bus.

44. In an electronic system having a first device coupled to a memory interface and a memory coupled to the memory interface, the first device having a device priority and capable of generating a request to access the memory, a method for selectively providing access to the memory comprising the steps of:

providing a decoder coupled to the memory interface, the decoder capable of operating in real time, having a decoder priority and capable of generating a request to access the memory;

providing an arbiter having an idle, a busy and a queue state;

generating a request by the decoder to access the memory;

determining the state of the arbiter;

providing the decoder access to the memory responsive to the arbiter being in the idle state;

queuing the request responsive to the arbiter being in the busy state; and

queuing the request responsive to the arbiter being in the queue state in an order responsive to the priority of the decoder request and the priority of any other queued requests.

45. The method of claim 44, further comprising the steps of:

determining the number of requests issued simultaneously;

responsive to number of requests issued simultaneously being greater than one:

selectively providing access to the memory responsive to the arbiter being in the idle state, and the priority of the simultaneously issued requests;

queuing the simultaneously issued requests responsive to the arbiter being in the busy state in an order responsive to the priority of the simultaneously issued requests;

queuing the simultaneously issued requests responsive to the arbiter being in the queue state in an order responsive to the priority of the simultaneously issued requests and the priority of any other queued requests.

46. The method of claim 45, wherein the step of determining the number of requests issued simultaneously is performed prior to the step of determining the state of the arbiter.

47. The method of claim 44, further comprising the step of preempting the first device access to the memory and providing the decoder access to the memory responsive to the first device having a burst length above a burst length threshold.

48. The method of claim 44, wherein the decoder priority increases

responsive to the length of time the request issued by the decoder is queued.

49. In an electronic system having a first device coupled to a memory interface and a memory coupled to the memory interface, a method for selectively providing access to the memory comprising the steps of:

providing a memory management system;

providing a decoder coupled to the memory interface, the decoder capable of operating in real time;

providing access to the memory at a first time interval to the decoder; and

providing access to the memory at a second time interval to the first device.

08/702911

ABSTRACT

An electronic system that contains a first device that requires a memory interface and video and/or audio decompression and/or compression device that shares a memory interface and memory with the first device while still permitting the video and/or audio decompression and/or compression device to operate in real time is disclosed.

Docket No. 96-S-11

DECLARATION FOR PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address, and citizenship are as stated below next to my name,

I believe I am the original, first, and sole inventor (if only one name is listed below) or an original, first, and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

Video and/or Audio Decompression and/or Compression Device that Shares a Memory Interface

the specification of which is attached hereto.

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the patentability of this application in accordance with Title 37, Code of Federal Regulations, Section 1.56(a).

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

2 I hereby appoint Lisa K. Jorgenson, Reg. No. 34,845 and Irena Lager, Reg. No. 39,260 to prosecute this application and to transact all business in the U.S. Patent and Trademark Office in connection therewith.

Please send all correspondence to:

Lisa K. Jorgenson
Reg. No. 34,845
SGS-Thomson Microelectronics, Inc.
1310 Electronics Drive
Carrollton, TX 75006
(214) 466-7414

1-00 Inventor's Signature: [signature]
Full Name of First Joint Inventor: Raul Zegers Diaz
Date of Signature: 8/20/96
Residence and Post Office Address:
750 Montrose Ave.
Palo Alto, CA 94303

Citizenship: United States of America

Inventor's Signature: ________________
Full Name of Second Joint Inventor: Jefferson Eugene Owen
Date of Signature: ________________
Residence and Post Office Address:
44177 Bowers Court
Freemont, CA 94539

Citizenship: United States of America

Inventor's Signature: ___________________
Full Name of First Joint Inventor: Raul Zegers Diaz
Date of Signature: ________________
Residence and Post Office Address:
750 Montrose Ave.
Palo Alto, CA 94303

Citizenship: United States of America

2-00 Inventor's Signature: [signature]
Full Name of Second Joint Inventor: Jefferson Eugene Owen
Date of Signature: Aug 22, 1996
Residence and Post Office Address:
44177 Bowers Court
Freemont, CA 94539

Citizenship: United States of America

IIR/702911

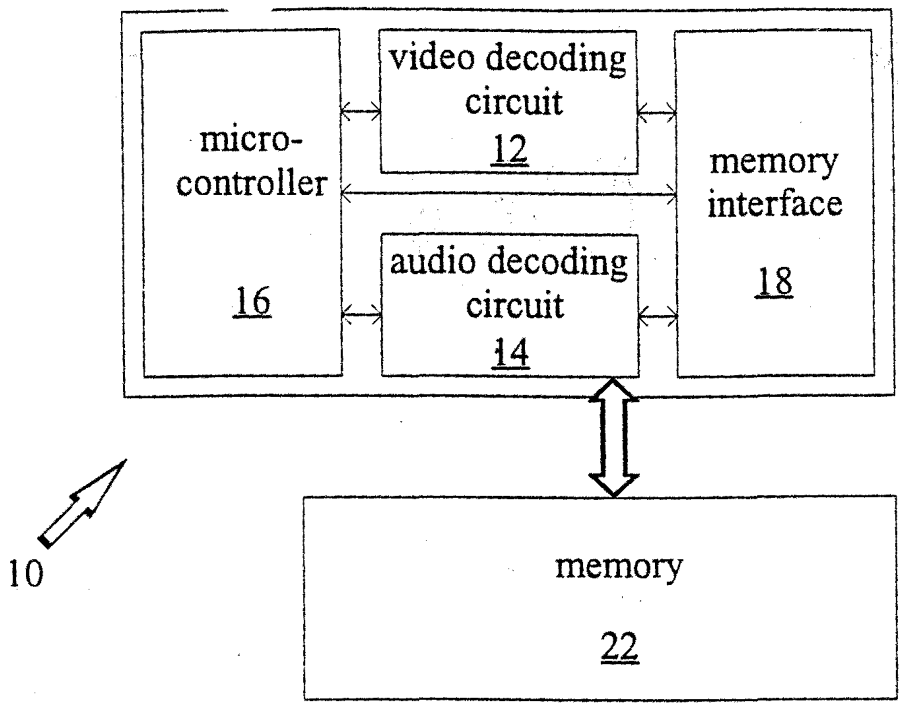


Figure 1a
(Prior Art)

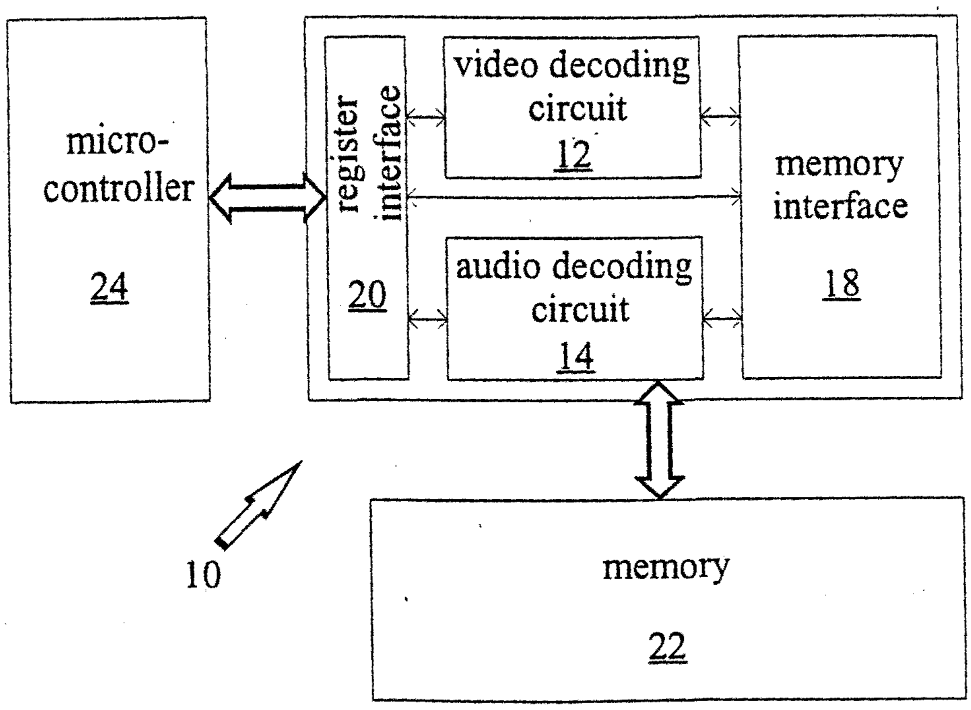


Figure 1b
(Prior Art)

6

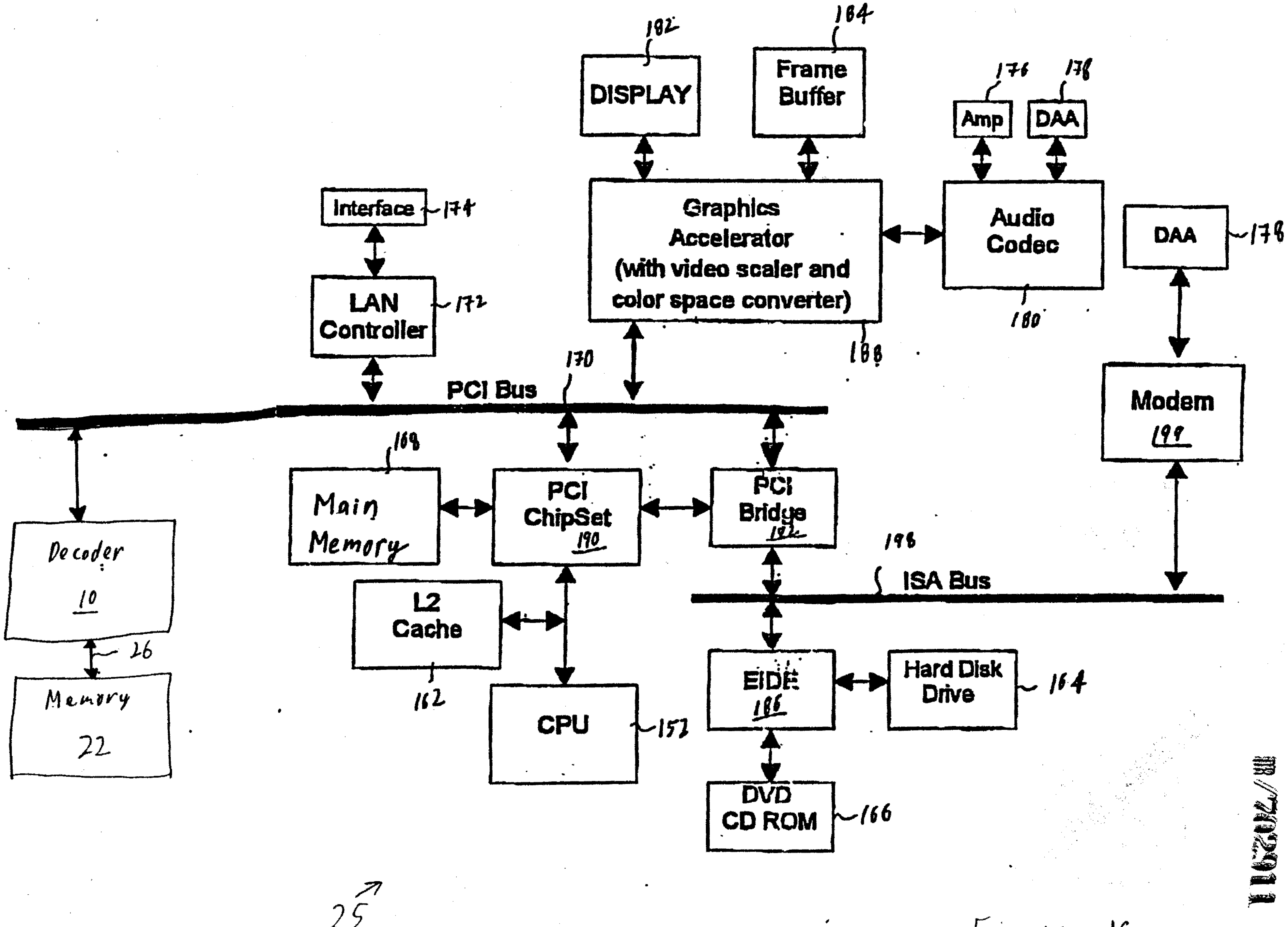


Figure 1C
(Prior Art)

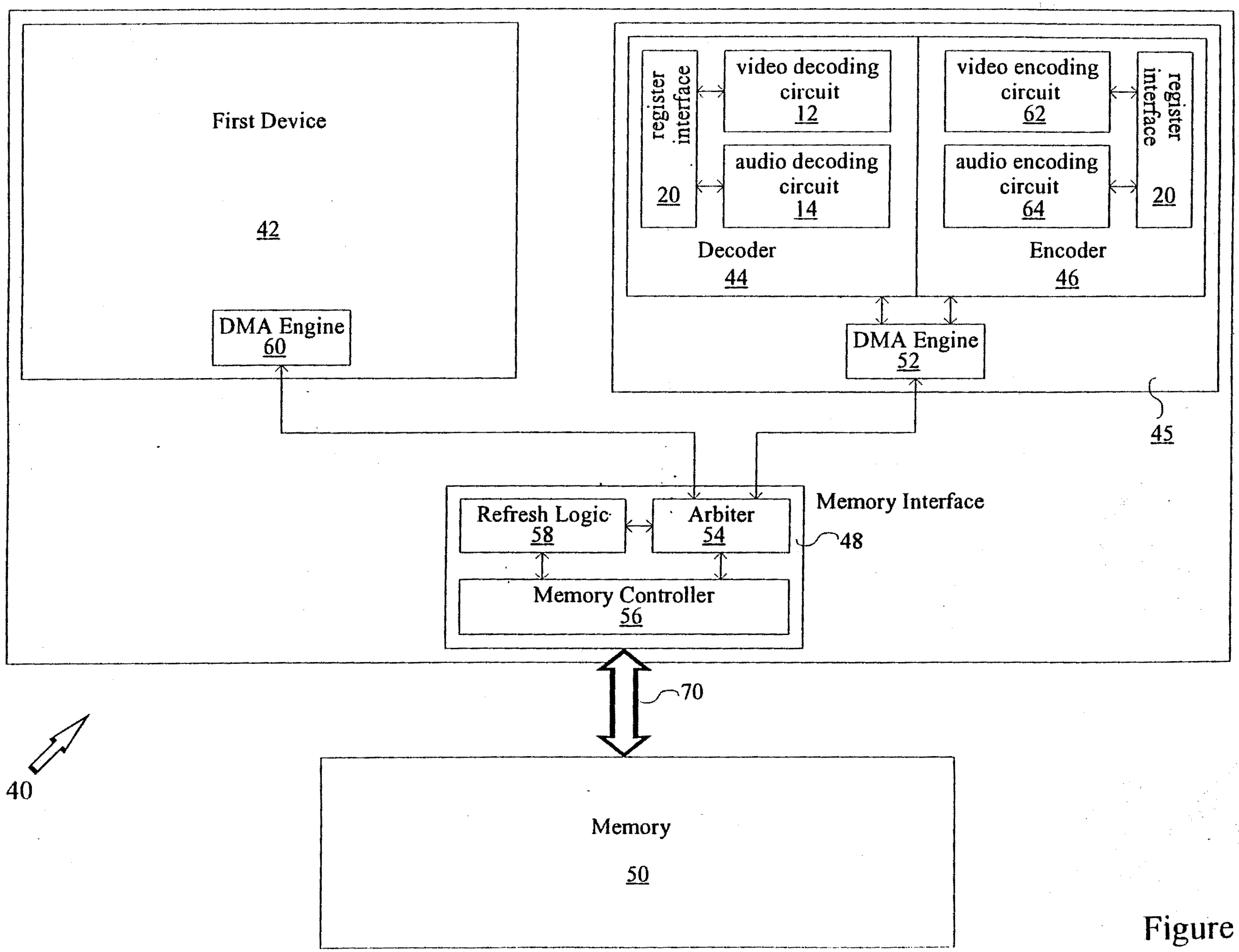

First Device
42
DMA Engine
60
register interface
20
video decoding circuit
12
audio decoding circuit
14
Decoder
44
video encoding circuit
62
audio encoding circuit
64
register interface
20
Encoder
46
DMA Engine
52
45
Memory Interface
Refresh Logic
58
Arbiter
54
48
Memory Controller
56
70
40
Memory
50

Figure 2

09/702911

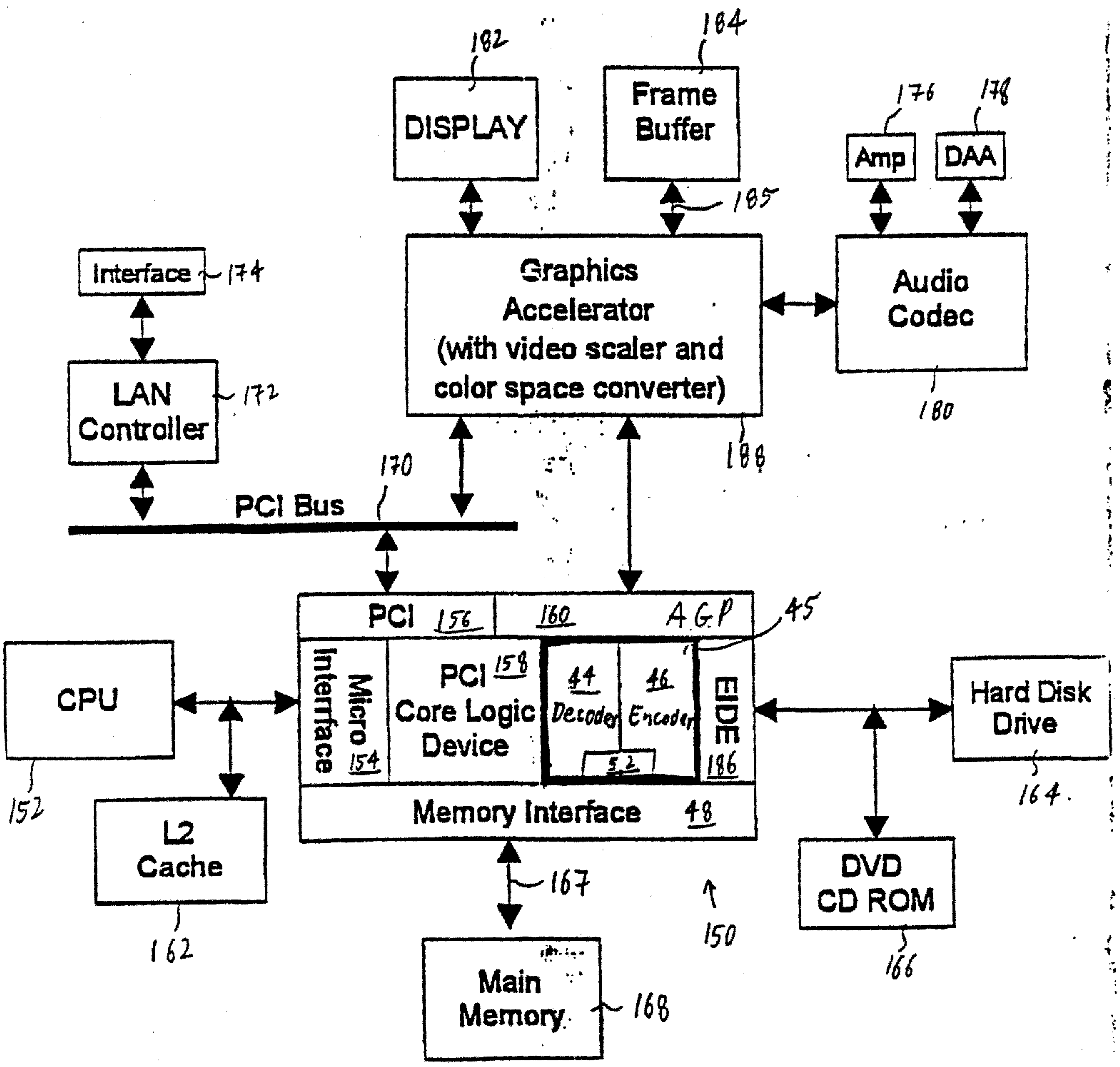
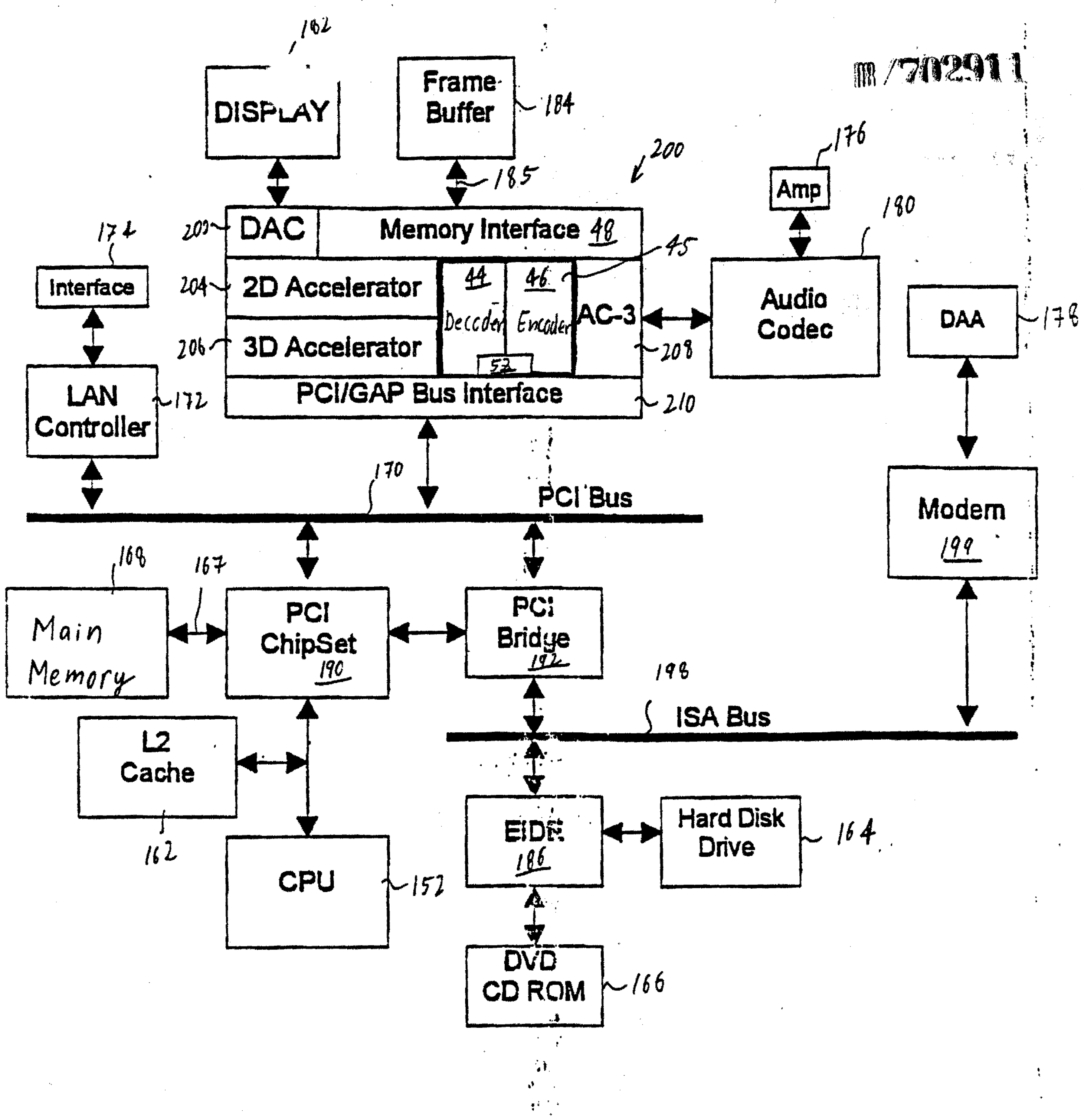


Figure 3



182
DISPLAY
Frame Buffer
184
185
200
176
Amp
180
174
202
DAC
Memory Interface 48
45
Interface
204
2D Accelerator
44 Decoder
46 Encoder
AC-3
Audio Codec
DAA
178
206
3D Accelerator
52
208
LAN Controller
172
PCI/GAP Bus Interface
210
170
PCI Bus
Modem 199
168
167
Main Memory
PCI ChipSet 190
PCI Bridge 192
198
ISA Bus
L2 Cache
162
CPU
152
EIDE 186
Hard Disk Drive
164
DVD CD ROM
166

Figure 4

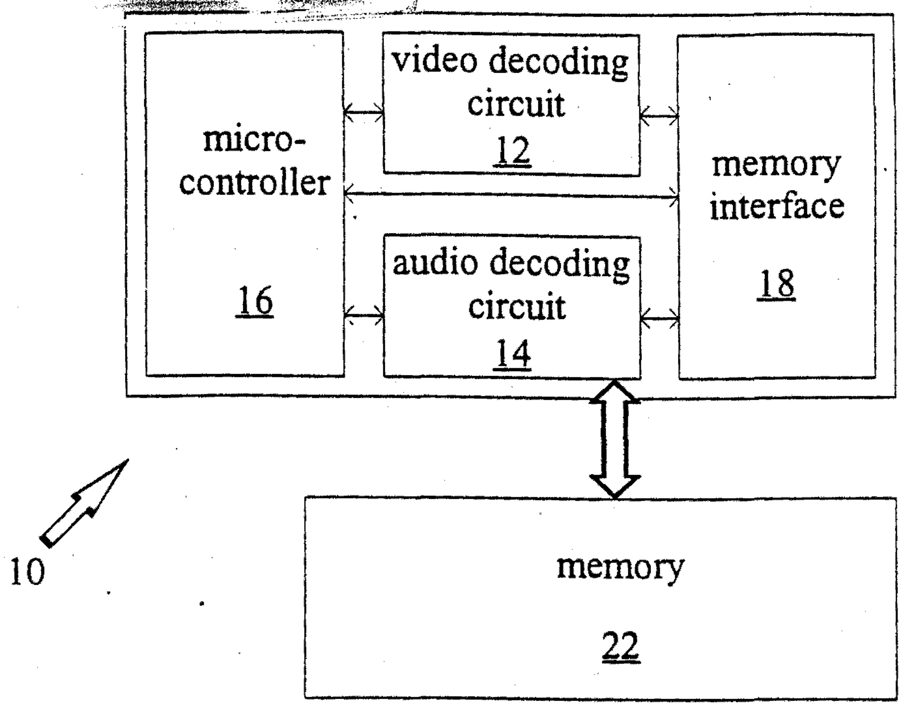


Figure 1a
(Prior Art)

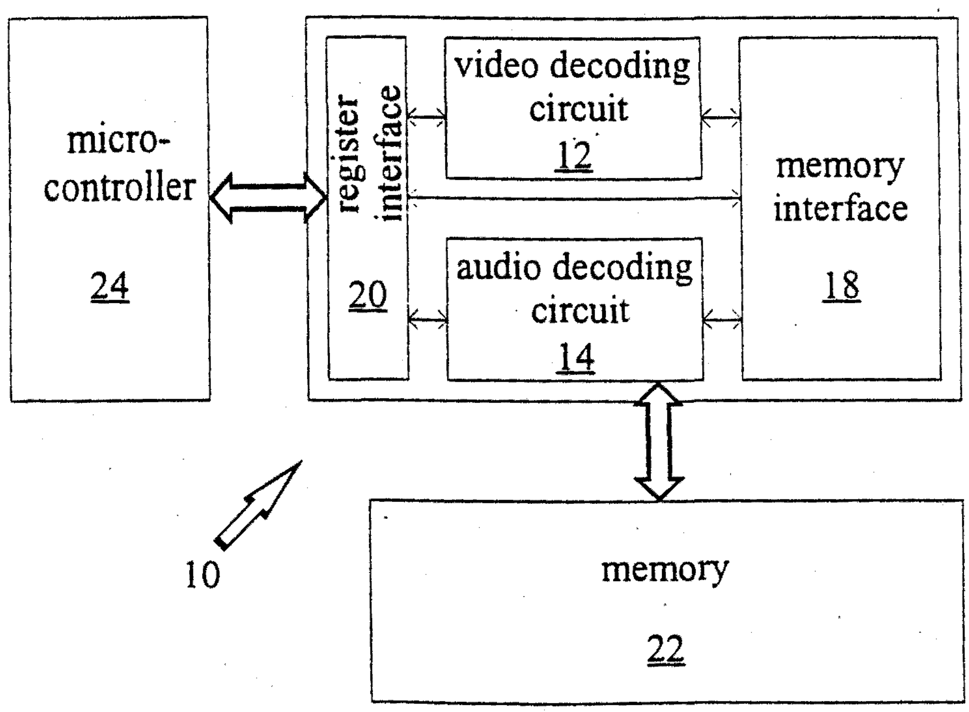


Figure 1b
(Prior Art)

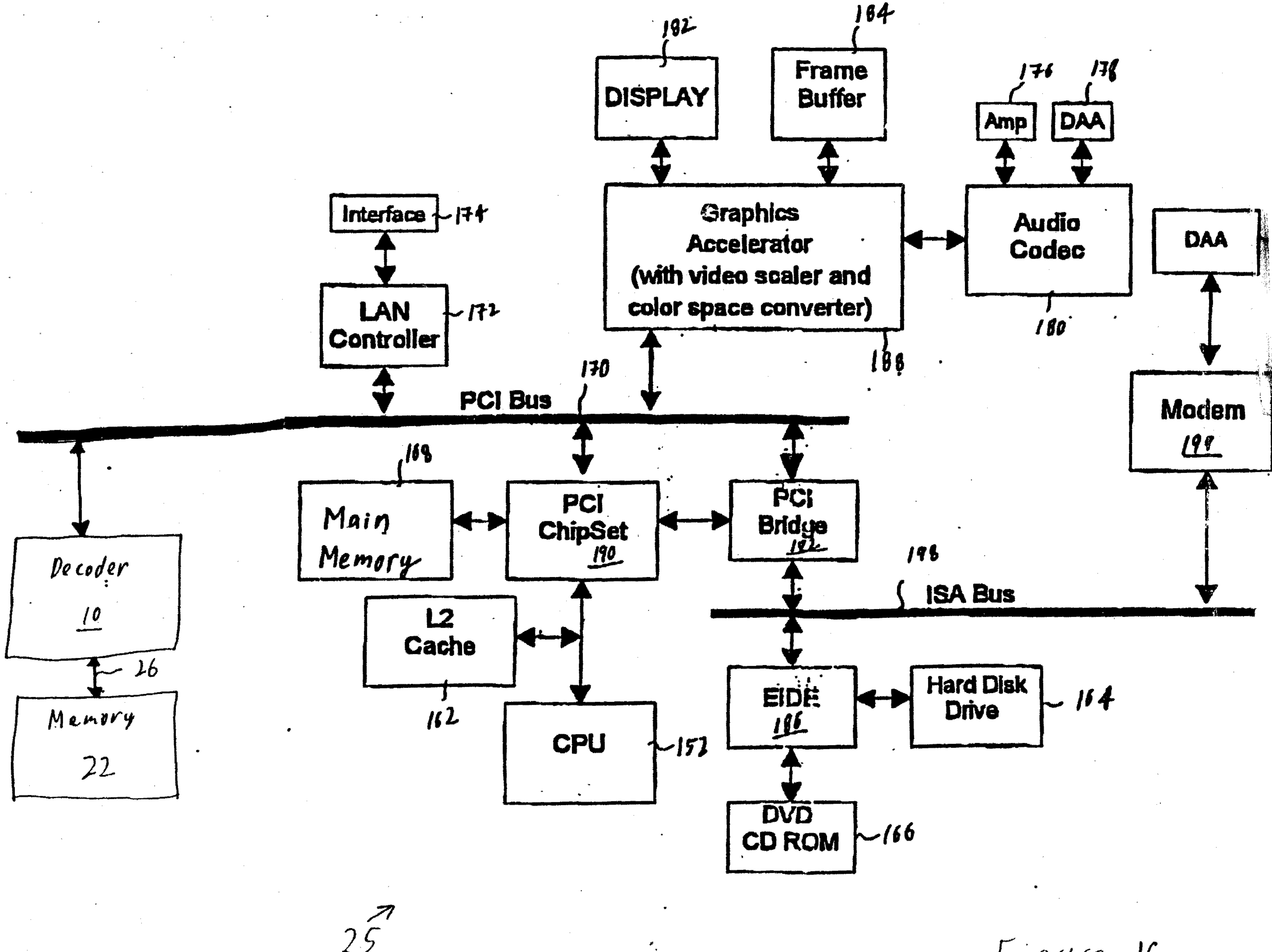
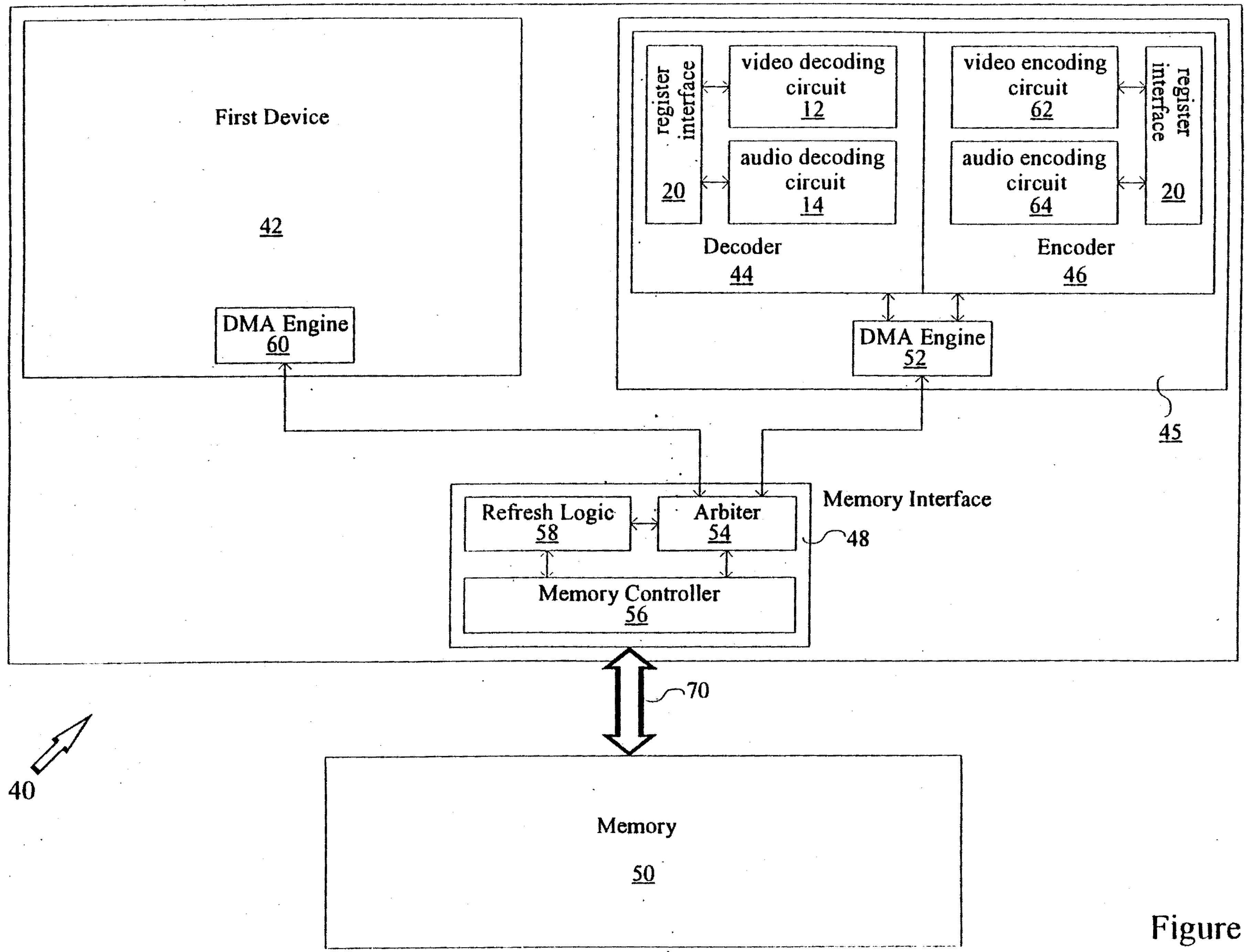


Figure 1C
(Prior Art)



First Device
42
DMA Engine
60
register interface
20
video decoding circuit
12
audio decoding circuit
14
Decoder
44
video encoding circuit
62
audio encoding circuit
64
register interface
20
Encoder
46
DMA Engine
52
45
Memory Interface
Refresh Logic
58
Arbiter
54
48
Memory Controller
56
70
40
Memory
50

Figure 2

PRINT OF DRAWINGS
AS ORIGINALLY FILED

09/702911

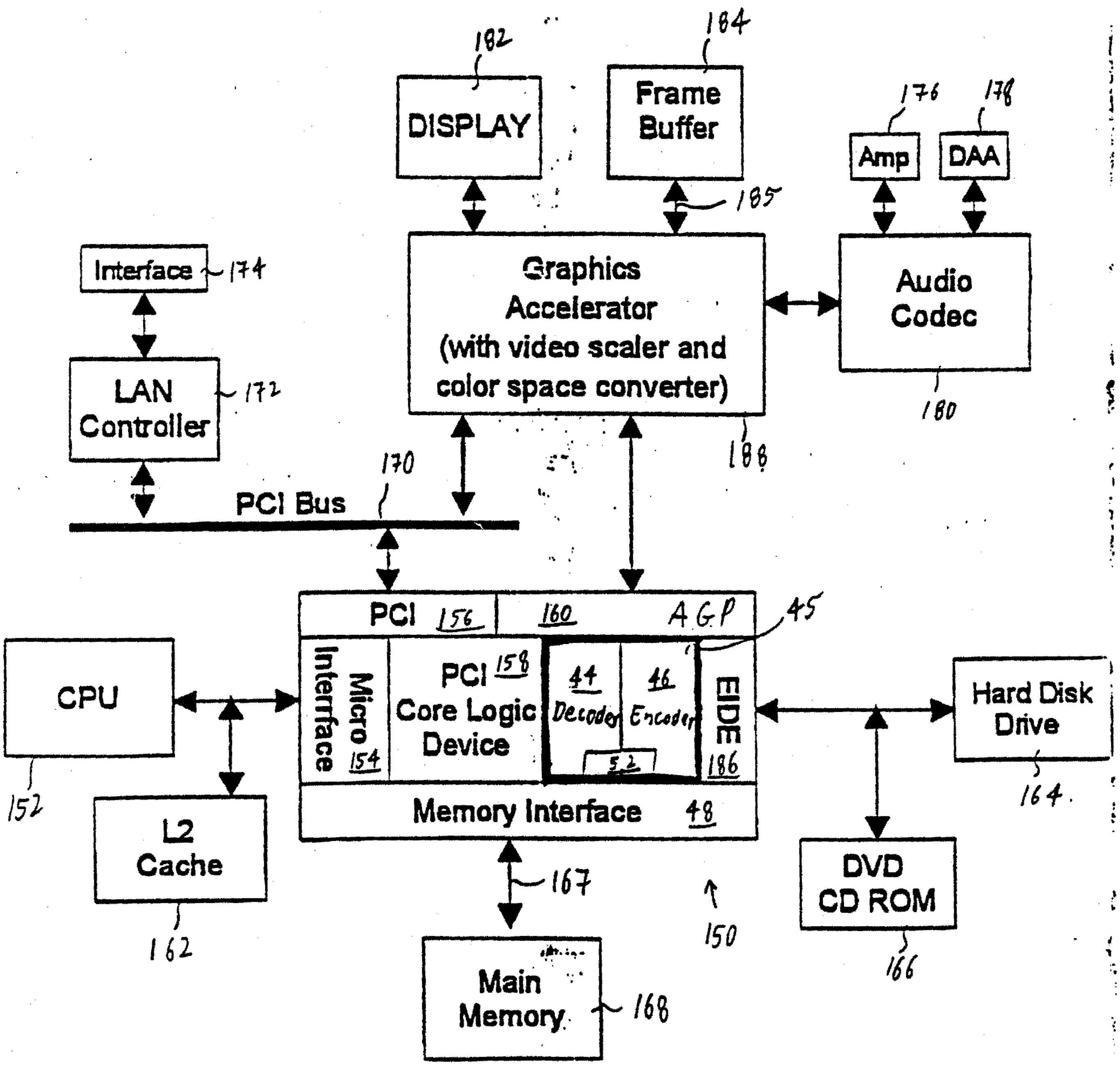


Figure 3

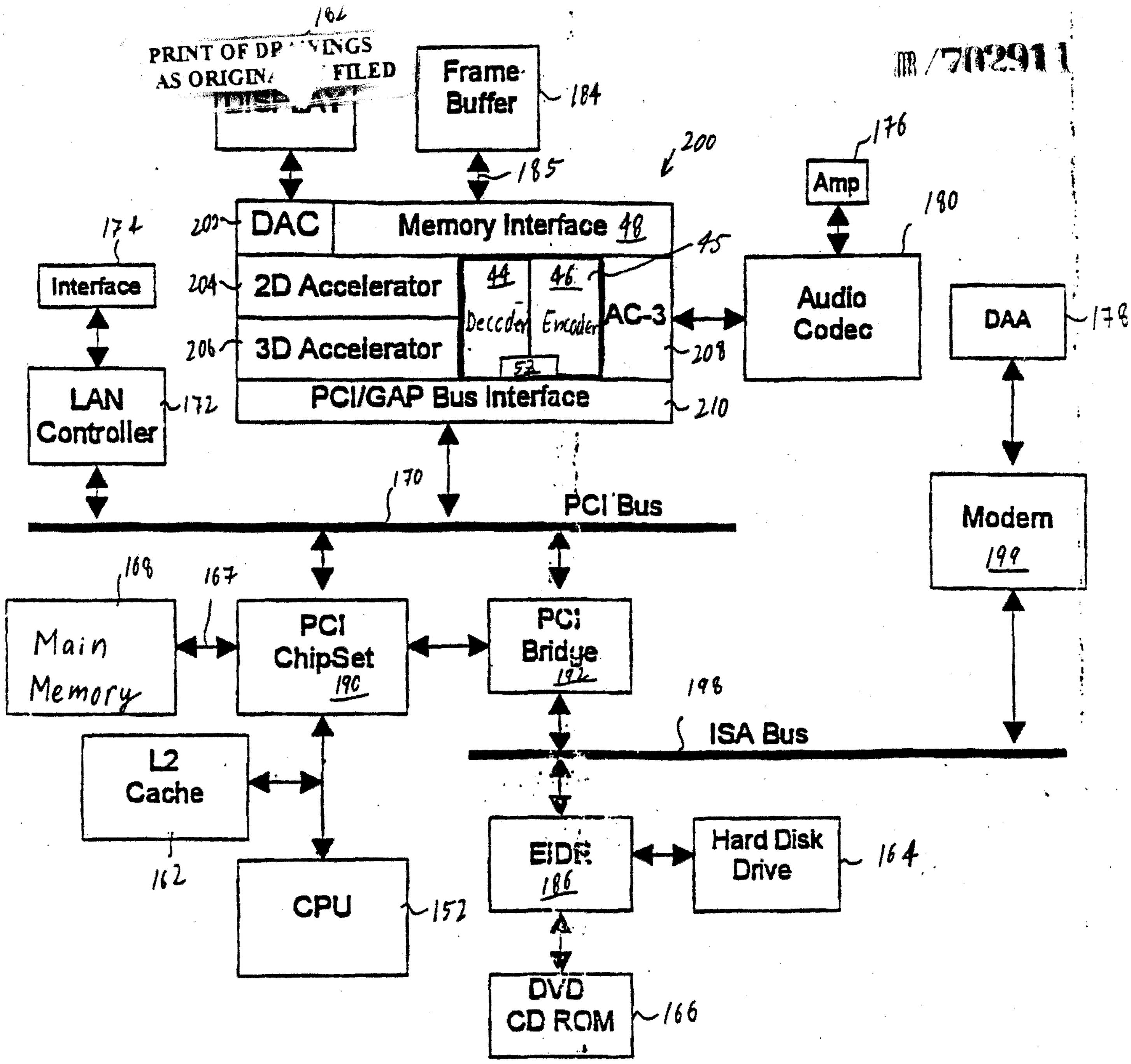

182
Frame Buffer
184
185
200
Amp
176
180
DAC
202
Memory Interface 48
45
174
Interface
204
2D Accelerator
44
Decoder
46
Encoder
AC-3
Audio Codec
DAA
178
206
3D Accelerator
52
208
LAN Controller
172
PCI/GAP Bus Interface
210
170
PCI Bus
Modem
199
168
167
Main Memory
PCI ChipSet
190
PCI Bridge
192
198
ISA Bus
L2 Cache
162
EIDE
186
Hard Disk Drive
164
CPU
152
DVD CD ROM
166

Figure 4

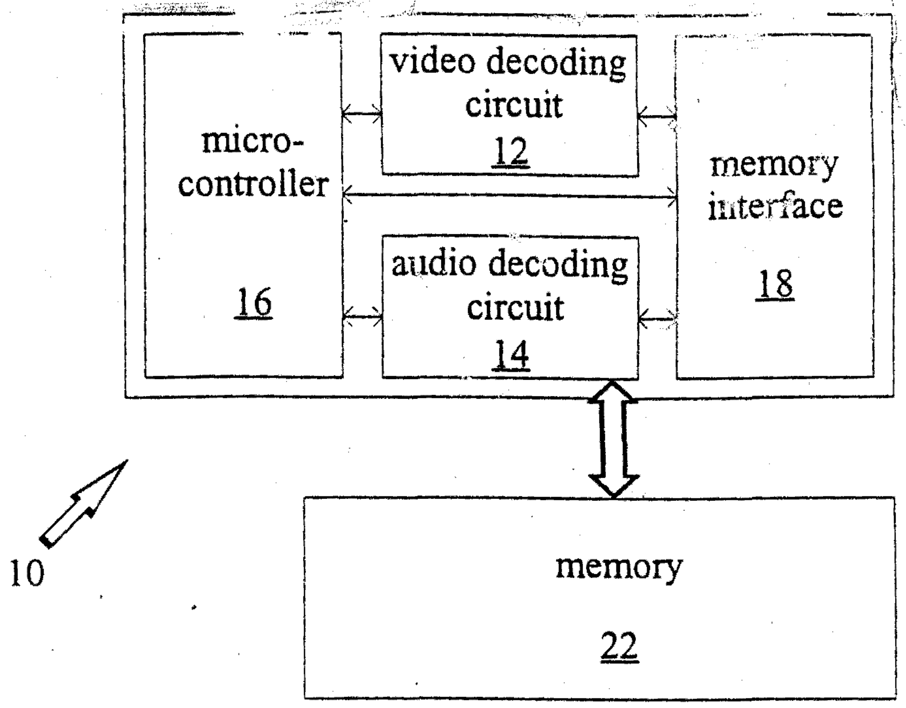


Figure 1a
(Prior Art)

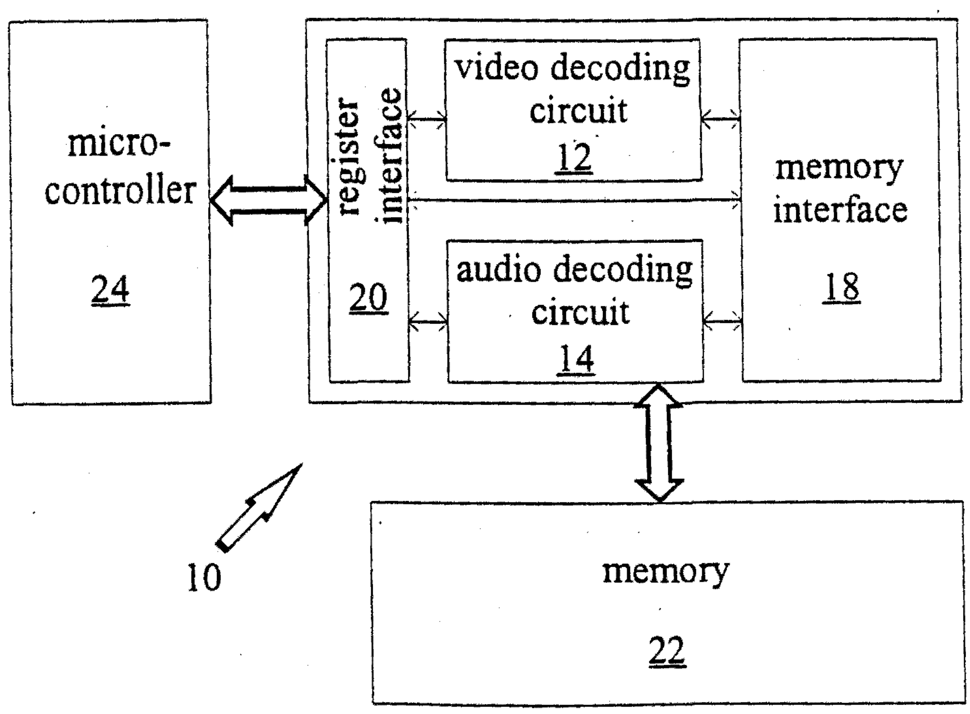


Figure 1b
(Prior Art)

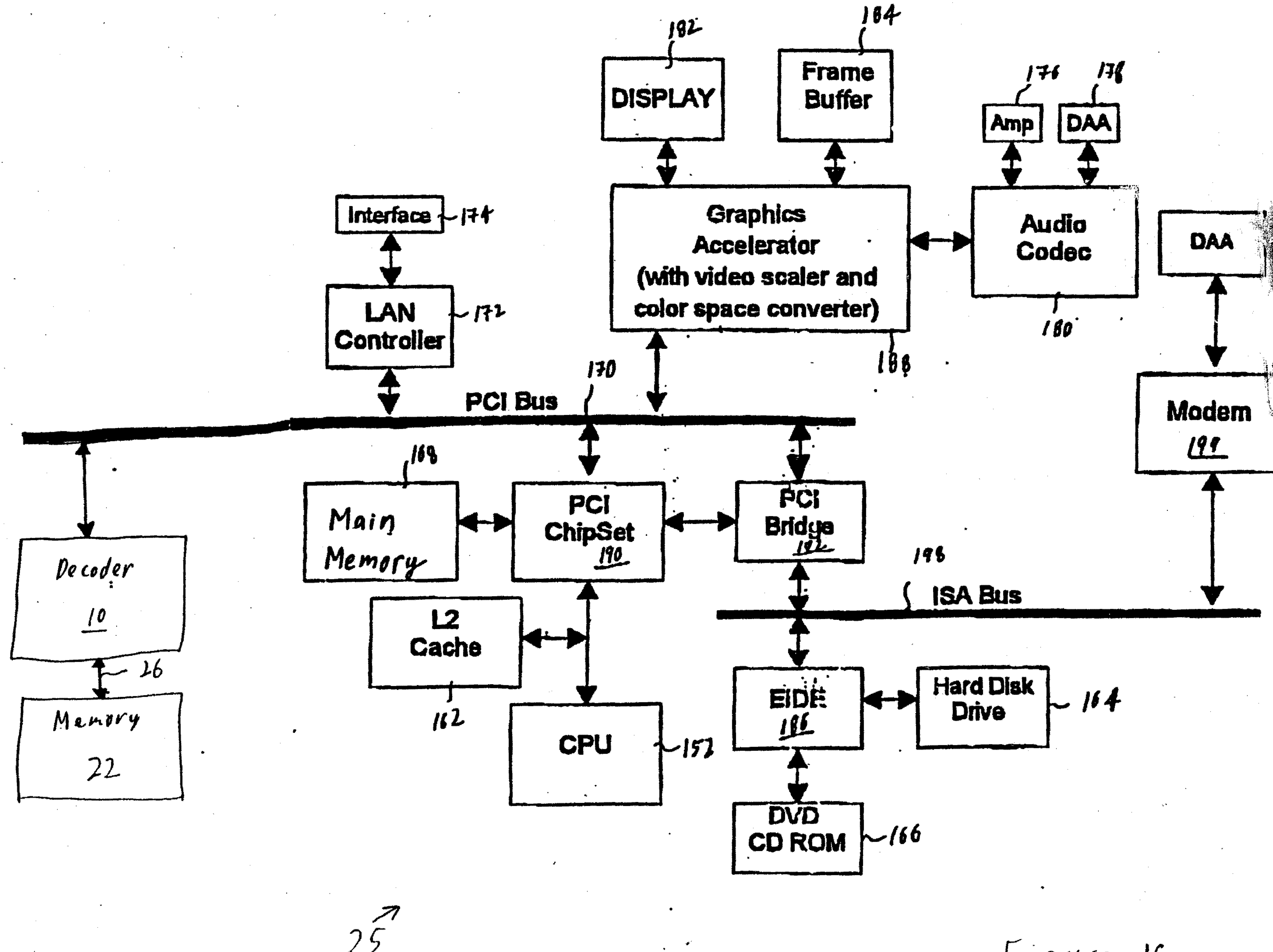
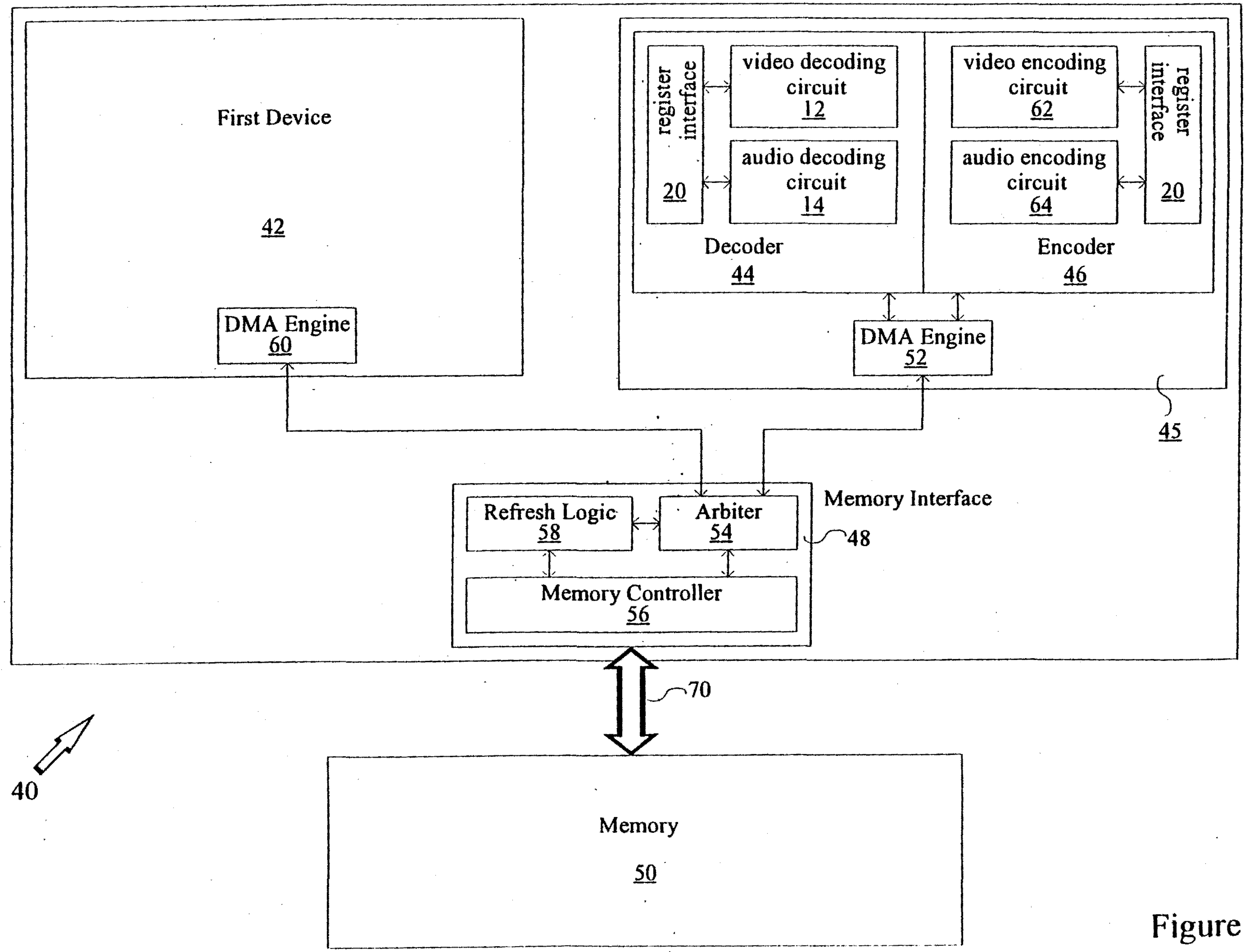


Figure 1C
(Prior Art)



First Device
42
DMA Engine
60
register interface
20
video decoding circuit
12
audio decoding circuit
14
Decoder
44
video encoding circuit
62
audio encoding circuit
64
register interface
20
Encoder
46
DMA Engine
52
45
Refresh Logic
58
Arbiter
54
Memory Interface
48
Memory Controller
56
70
40
Memory
50

Figure 2

PRINT OF DRAWINGS AS ORIGINALLY FILED

09/702911

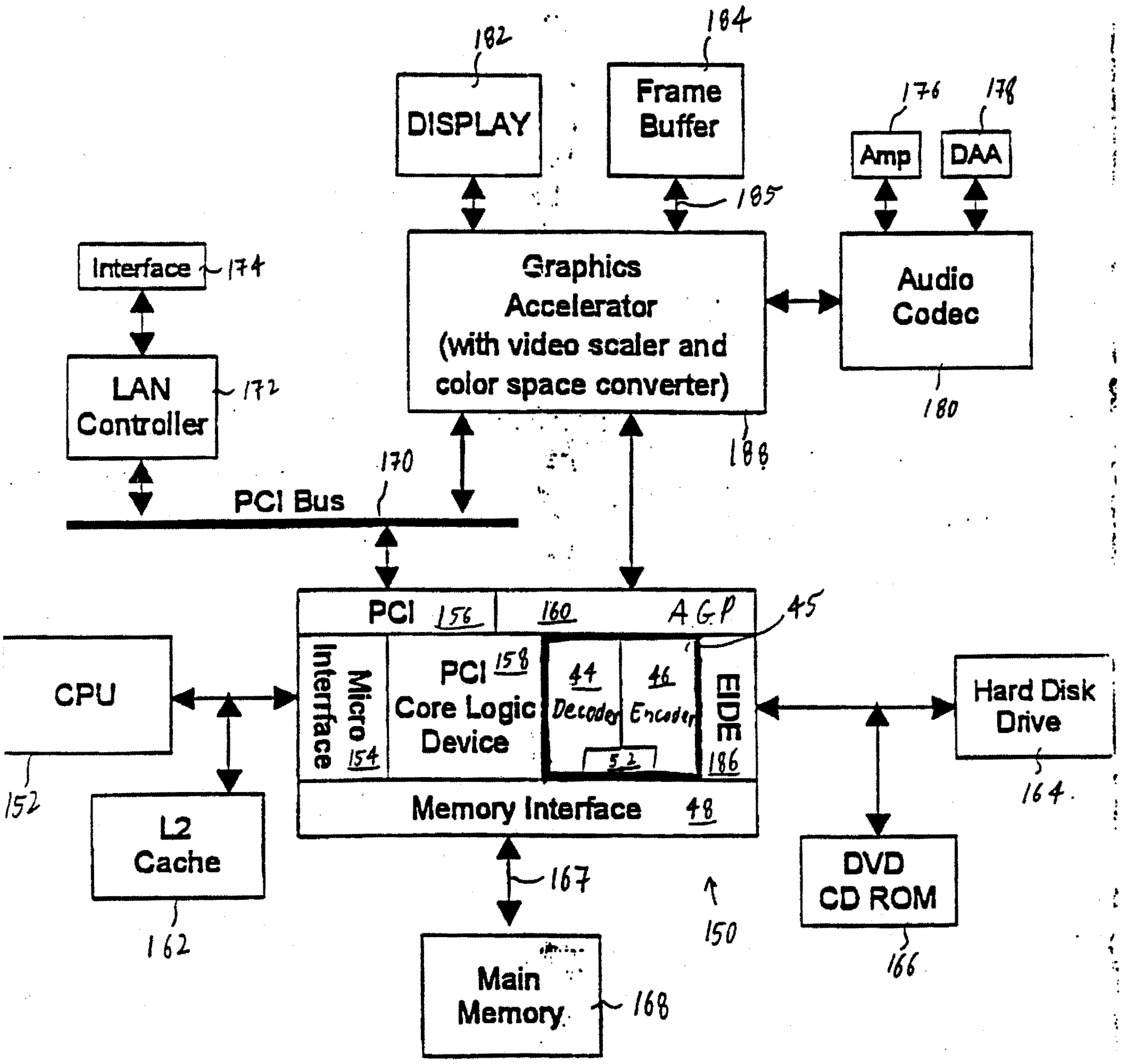


Figure 3

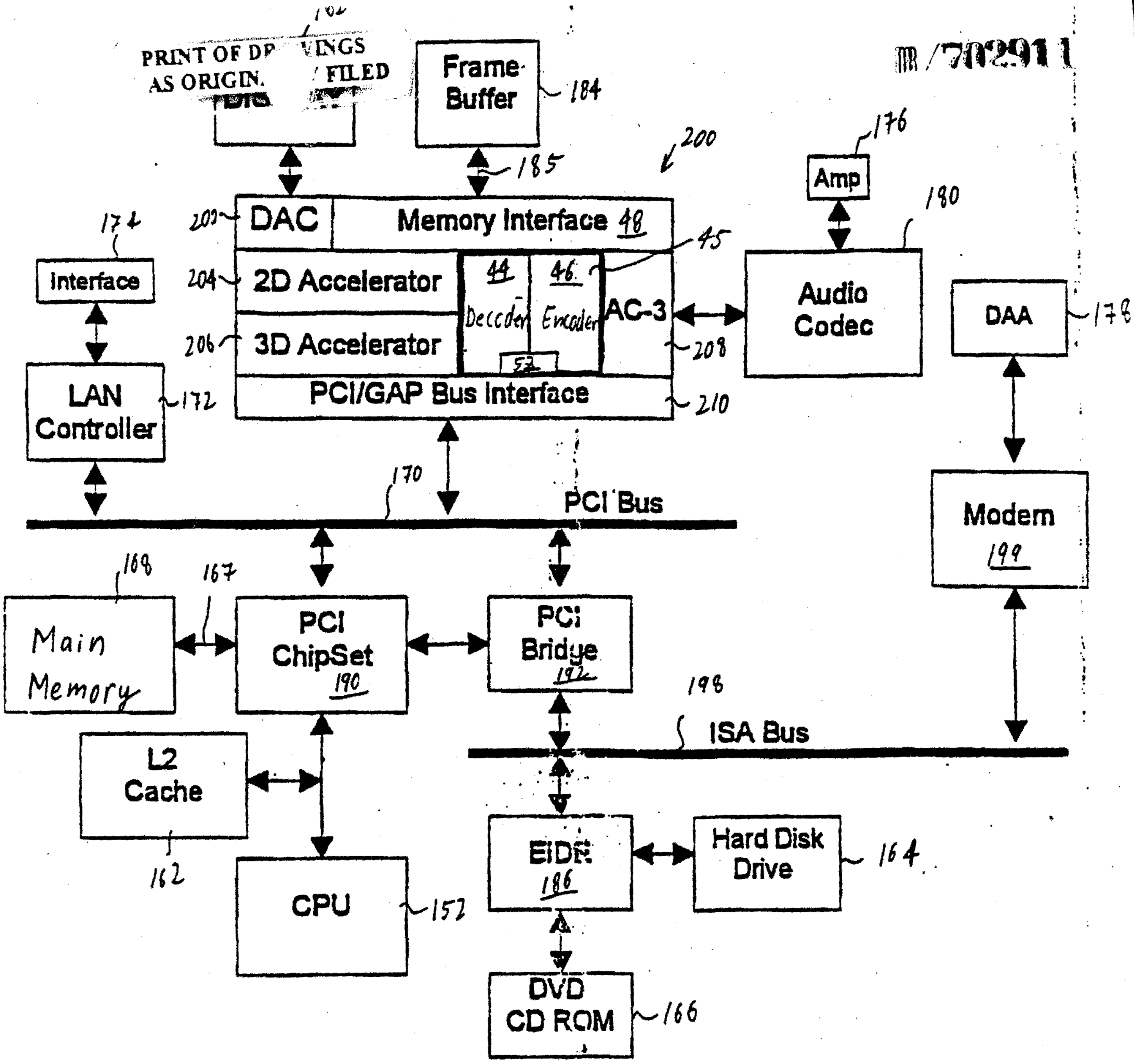

Frame Buffer
184
185
200
176
Amp
180
DAC
Memory Interface 48
45
174
202
Interface
204
2D Accelerator
44
Decoder
46
Encoder
AC-3
Audio Codec
DAA
178
206
3D Accelerator
52
208
LAN Controller
172
PCI/GAP Bus Interface
210
170
PCI Bus
Modem
199
168
167
Main Memory
PCI ChipSet
190
PCI Bridge
192
198
ISA Bus
L2 Cache
162
CPU
152
EIDE
186
Hard Disk Drive
164
DVD CD ROM
166

Figure 4

GP 2612

#28 7/14/97

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

MAIL ROOM NOV 4 1996 PAT. & TRADEMARK OFF.

In Re the Application of:
Raul Diaz et al.

Serial No.: 08/702,911

Filed: August 23, 1996

2318- 2/28/97

Examiner: NOT ASSIGNED

Art Unit: 2318

Docket No: 96-S-11

RECEIVED MAR 05 1997 GROUP 2300

For: Video and/or Audio Decompression and/or Compression Device that Shares a Memory Interface

INFORMATION DISCLOSURE STATEMENT UNDER C.F.R. 1.97

Honorable Assistant Commissioner
for Patents
Washington, D.C. 20231

RECEIVED NOV 13 96 GROUP 2600

Sir:

Applicants request that the information listed on the attached Form PTO-1449 be considered by the Office during the pendency of the above entitled application, pursuant to 37 C.F.R. 1.97.

Please charge any fees necessary for prosecution of the present application to deposit account no. 19-1353. If any extension of time is required, such extension is hereby requested. Please charge any additional required fee for extension of time to Applicant's Deposit Account No. 19-1353.

CERTIFICATE OF MAILING (37 CFR 1.8a)

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to the: Assistant Commissioner for Patents, Washington, D.C. 20231, on Oct. 31, 1996.

Mary L. Nenez
Signature

In accordance with 37 C.F.R. 1.97(h), the filing of this Information Disclosure Statement shall not constitute an admission that any information cited therein is, or is considered to be, material to patentability as defined in 37 C.F.R. 1.56(b). In the interest of full and complete disclosure to the Office, some or all of the art cited herein may not be considered by Applicant(s) or the Undersigned to be material under the new standards of materiality defined in 37 C.F.R 1.56(b), enacted March 16, 1992, but may be material under the old standard of materiality defined in 37 C.F.R. 1.56(a), last amended on November 28, 1988, or may merely be technical background which may be of interest to the Examiner. In accordance with 37 C.F.R. 1.97(g), the filing of this Information Disclosure Statement shall not be construed to mean that a search has been made.

This Information Disclosure Statement is being filed under 37 C.F.R. § 1.97(b) within three months of the filing date of the application, or before the mailing date of a first office action on the merits. No fee or certification is required.

Respectfully submitted,

Irena Lager
Registry No. 39,260
SGS-THOMSON Microelectronics, Inc.
1310 Electronics Drive
Carrollton, Texas 75006
(972) 466-7511

Attorney for Applicants

Form PTO 1449 Rev 7-80	U.S. Dept of Commerce Patent & Trademark Off.	Atty. Docket No. 96-S-11	Serial No. 08/702,911
		Applicant Raul Diaz et al.	
LIST OF PRIOR ART CITED BY APPLICANT (Use several sheets if necessary)		Filing Date August 23, 1996	Group

U.S. PATENT DOCUMENTS

EXAMINER INITIAL		DOCUMENT NUMBER	DATE	NAME	CLASS/ SUBCLASS	FILING DATE
[initials]	AA	5,459,519	10/17/95	Scalise et al.	348/431	
	AB					

FOREIGN COUNTRIES

		DOC. NO.	DATE	COUNTRY	CLASS/ SUBCLASS	TRANSLATION
[initials]	AC	0 673 171 A2	09/20/95	Europe	— —	English

OTHER PRIOR ART (INCLUDING AUTHOR, TITLE, DATE, PERTINENT PAGE, ETC.)

[initials]	AD	Bheda, H. and P. Srinivasan, "A High-Performance Cross-Platform MPEG Decoder," Digital Video Compression on Personal Computers: Algorithms and Technologies. SPIE Proceedings, February 7-8, 1994, Vo. 2187, pp. 241-248.
[initials]	AE	Bursky, D., "Highly Integrated Controller Eases MPEG-2 Adoption," Electronic Design, August 21, 1995, Vol. 43, No. 17, pp. 141-142.
[initials]	AF	Galbi, D. et al., "An MPEG-1 Audio/Video Decoder With Run-Length Compressed Antialiased Video Overlays," 1995 IEEE International Solid-State Circuits Conference, pp. 286-287, 381.
[initials]	AG	Maturi, G., "Single Chip MPEG Audio Decoder," IEEE Transactions on Consumer Electronics, Vol. 38, No. 3, August 1992, pp. 348-356.
[initials]	AH	Butler, B. and T. Mace, "The Great Leap Forward," PC Magazine, October 11, 1994, pp. 241-244, 246, 248, 250, 253-254, 256, 260-261, 264, 266-268, 273-275, 278.
[initials]	AI	Doquilo, J., "Symmetric Multiprocessing Servers: Scaling the Performance Wall," Infoworld, March 27, 1995, pp. 82-85, 88-92.
[initials]	AJ	Video Electronics Standards Association, "VESA Unified Memory Architecture Hardware Specifications Proposal," Version: 1.0p, October 31, 1995, pp. 1-38.
[initials]	AK	Video Electronics Standards Association, "VESA Unified Memory Architecture VESA BIOS Extensions (VUMA-SBE Proposal," Version: 1.0p, November 1, 1995, pp. 1-26.
	AL	Giorgis, T., "SMP Network Operating Systems," Computer Dealer News, Vol. 12, No. 16, August 8, 1996.

EXAMINER	[signature]	DATE CONSIDERED	8/20/97

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with PEP 609; Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

MAIL ROOM NOV 19 1996 82 PAT. & TRADEMARK OFF.

T H

2412 #3 2/14/97

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re the Application of:
Raul Diaz et al.

Examiner:

Serial No.: 08/702,911

Art Unit:

Filed: August 23, 1996

Docket No: 96-S-11

For: Video and/or Audio Decompression and/or Compression Device that Shares a Memory Interface

INFORMATION DISCLOSURE STATEMENT UNDER C.F.R. 1.97

Honorable Assistant Commissioner
for Patents
Washington, D.C. 20231

Sir:

Applicants request that the information listed on the attached Form PTO-1449 be considered by the Office during the pendency of the above entitled application, pursuant to 37 C.F.R. 1.97.

Please charge any fees necessary for prosecution of the present application to deposit account no. 19-1353. If any extension of time is required, such extension is hereby requested. Please charge any additional required fee for extension of time to Applicant's Deposit Account No. 19-1353.

CERTIFICATE OF MAILING (37 CFR 1.8a)
I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to the: Assistant Commissioner for Patents, Washington, D.C. 20231, on NOV 11, 1996.
Kim Lawson Signature

In accordance with 37 C.F.R. 1.97(h), the filing of this Information Disclosure Statement shall not constitute an admission that any information cited therein is, or is considered to be, material to patentability as defined in 37 C.F.R. 1.56(b). In the interest of full and complete disclosure to the Office, some or all of the art cited herein may not be considered by Applicant(s) or the Undersigned to be material under the new standards of materiality defined in 37 C.F.R 1.56(b), enacted March 16, 1992, but may be material under the old standard of materiality defined in 37 C.F.R. 1.56(a), last amended on November 28, 1988, or may merely be technical background which may be of interest to the Examiner. In accordance with 37 C.F.R. 1.97(g), the filing of this Information Disclosure Statement shall not be construed to mean that a search has been made.

This Information Disclosure Statement is being filed under 37 C.F.R. § 1.97(b) within three months of the filing date of the application, or before the mailing date of a first office action on the merits. No fee or certification is required.

Respectfully submitted,

Irena Lager
Registry No. 39,260
SGS-THOMSON Microelectronics, Inc.
1310 Electronics Drive
Carrollton, Texas 75006
(972) 466-7511

Attorney for Applicants

Sheet 1 of 1

Form PTO 1449 Rev 7-80	U.S. Dept of Commerce Patent & Trademark Off.	Atty. Docket No. 96-S-011	Serial No. 08/702,911
		Applicant Raul Diaz et al.	
LIST OF PRIOR ART CITED BY APPLICANT (Use several sheets if necessary)		Filing Date August 23, 1996	Group

MAIL ROOM 82 NOV 19 1996 PAT. & TRADEMARK OFFICE

U.S. PATENT DOCUMENTS

EXAMINER INITIAL	DOCUMENT NUMBER	DATE	NAME	CLASS/ SUBCLASS	FILING DATE
AA					
AB					
AC					
AD					
AE					
AF					
AG					
AH					
AI					
AJ					
AK					
AL					

FOREIGN COUNTRIES

	DOC. NO.	DATE	COUNTRY	CLASS/ SUBCLASS	TRANSLATION
AM					
AN					

OTHER PRIOR ART (INCLUDING AUTHOR, TITLE, DATE, PERTINENT PAGE, ETC.)

	AO	King, A., *Inside Windows 95*, Microsoft Press, Redmond, Washington, 1994, pp. 85-90.
	AP	"MPEG Video Overview," *SGS-THOMSON Microelectronics Technical Note*, April 1992, pp. 1-4.
	AQ	

EXAMINER [signature] DATE CONSIDERED 8/20/97

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with PEP 609; Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

MAIL ROOM FEB 26 1997 PAT. & TRADEMARK OFF.

RECEIVED MAR 19 1997 GROUP 2300

#45

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:
Raul Z. Diaz et al.

Docket No.: 96-S-011

Serial No.: 08/702,911

2318 2/28/97

Examiner:

Filed: August 23, 1996

Art Unit: ~~2612~~ 2318

For: Video and/or audio decompression and/or compression device that shares a memory interface

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

PETITION TO SECURE FILING DATE AS OF

MAILING DATE VIA EXPRESS MAIL

Applicant petitions that this application be accorded the filing date on which the papers were sent "Express Mail Post Office to Addressee" mailing label no. EG947362259US on August 23, 1996.

CERTIFICATE OF MAILING (37 CFR 1.8a)
I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to the: Assistant Commissioner for Patents, Washington, D.C. 20231, on Feb 24, 1997.

[signature]
Signature

SUBMISSIONS

Submitted herewith is:

1. A copy of the executed Express Mail certificate with mailing label number EG947362259US.
2. A copy of the Express Mail Receipt No. EG947362259US with a "date in" of August 24, 1996 as entered by the U.S. Postal Service.
3. Declaration of Kimberley K. Larson.
4. A copy of Collection Management System, Collection Point Inventory by Address (CPIA) for the U.S. Post Office, Dallas District.

Applicant respectfully requests that the above-referenced application be accorded a filing date of August 23, 1996 as shown by the attached declaration of Kimberley K. Larson and the attached CPIA from the U.S. Post Office. Applicant had a reasonable basis to believe that the correspondence placed in the Express Mail envelope and deposited in a U.S. Postal Service Mail box on August 23, 1996 would be picked up that same day and, therefore, the Express Mail label would have a "date in" of August 23, 1996. Therefore, Applicants request that the application be accorded a filing date of August 23, 1996 as specified in 37 C.F.R. §1.10(c) in effect on August 23, 1996.

PETITION FEE

The petition fee (37 CFR 1.17(h)) is hereby authorized to be charged to Deposit Account No. 19-1353 of SGS-THOMSON Microelectronics, Inc.

REQUEST FOR REFUND OF PETITION FEE

Because no defect exists in applicants' previous submission, a refund of the petition fee is respectfully requested.

Respectfully submitted,

Irena Lager
Reg. No. 39,260
SGS-THOMSON Microelectronics, Inc.
Mail Station 2346
1310 Electronics Dr.
Carrollton, Texas 75006
(972) 466-7511

Attorney for Applicant

RECEIVED
MAR 19 1997
GROUP 2300

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re Application of:
Raul Z. Diaz et al.

Docket No.: 96-S-011

Serial No.: 08/702,911

Examiner:

Filed: August 23, 1996

Art Unit: 2612

For: Video and/or audio decompression and/or compression device that shares a memory interface

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

DECLARATION OF KIMBERLEY K. LARSON

I, KIMBERLEY K. LARSON, do hereby take oath and swear as follows:

(1) I am employed as a patent secretary in the Patent Department of SGS-THOMSON Microelectronics, Inc.

(2) As part of my duties, I am responsible for the preparation of certain documents for transmittal to the Patent and Trademark Office, including ensuring that proper documents are present to be transmitted, organizing the documents to be transmitted and placing these documents in envelopes for transmittal.

(3) On August 23, 1996, I prepared an Express Mail mailing label bearing mailing label number EG947362259US addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231. I also prepared and signed a Certificate of Mailing by Express Mail dated August 23, 1996 with express mail mailing label number EG947362259US, a copy of which is attached hereto.

(4) Pursuant to the requirements of 37 CFR 1.10, I placed the correspondence in an Express Mail envelope and deposited the Express Mail envelope in a U.S. Express Mail mailbox on Friday, August 23, 1996 around 4:45 p.m. This was prior to the last designated pick-up time of 5:00 p.m for this U.S. Express Mail mailbox. Therefore, the certificate of Express Mailing and the application was deposited with the U.S. Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on August 23, 1996 addressed to the Assistant Commissioner for Patents, Box Patent Applications, Washington, D.C. 20231. Attached is a copy of the collection times sent via facsimile from Vincent Lewis of the Consumer Affairs/Claims Division of the U.S. Postal Service. This facsimile shows the last collection time of 5:00 p.m. for the post office on 13904 Josey Lane where the Express Mail package was deposited.

(5) I am unaware as to the circumstances that caused our express mail package not be picked up and processed by the U.S. Postal Service until the next day, August 24, 1996. The above-referenced application was accorded a filing date of August 26, 1996, instead of August 23, 1996. A copy of the return postcard bearing a cancelled date of August 24, 1996, and an actual receipt date of August 26, 1996 is enclosed, along with a copy of the Express Mail Receipt No. EG947362259US with a postmark date of August 24, 1996.

I declare further that all statements made herein of my own knowledge are true; that all statements made herein on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such wilful false statements may jeopardize the validity of this application and any registration resulting therefrom.

Signed at Carrollton, Texas this 24 day of February, 1997.

By: Kimberly K. Larson

Kimberley K. Larson

STATE OF TEXAS
COUNTY OF DALLAS

Subscribed and sworn to before me this 24th day of February, 1997

Mary L. Hiner

Notary Public

MARY L. HINER
Notary Public, State of Texas
My Commission Expires 02/22/99

/30/96
:01:15

COLLECTION MANAGEMENT SYSTEM

COLLECTION POINT INVENTORY BY ADDRESS
DALLAS DISTRICT OPS
961 W BETHEL RD
COPPELL TX 75099-9331

FARMERS BRANCH STATIO:: (214)247-4591

ID	Box Address Description of Add:ess Last Box in the Area Nearest Express Mail Box	Service Class Type of Box Button Number	Area of Box	Weekdays Coll by	AM	Coll by	PM	Saturdays Coll by	AM	Coll by	PM	Holidays Coll by	AM	Coll by	PM
10	11400 LUNA RD WESTWOOD BUSINESS PARK ENTRANCE 13904 JOSEY LN 13904 JOSEY LN	MIXED STANDARD 0000004AC84A	BUS			3478 3488	2:00 4:30			3490S	12:00				
11	14280 MARSH LN SE MARSH & SPRING VALLEY 13904 JOSEY LN 13904 JOSEY LN	MIXED STANDARD 0000004AC51D	RES			3474 3484	2:00 4:30			3491S	12:00				
13	7 MEDICAL PARKWAY NE WEBB CHAPEL & MEDICAL PKY 13904 JOSEY LN 13904 JOSEY LN	STAMPED STANDARD 0000004AC74A	BUS			3473 3483	2:00 4:30			3491S	12:00				
17	7 MEDICAL PARKWAY NE WEBB CHAPEL & MEDICAL PKY 13904 JOSEY LN 13904 JOSEY LN	METERED STANDARD 0000004ACF8C	BUS			3473 3483	2:00 4:30			3491S	12:00				
16	2270 VALLEY VIEW LN SOUTHEAST VALLEY VIEW VALLEY BRANCH 13904 JOSEY LN 13904 JOSEY LN	MIXED STANDARD 0000004AC5EE	RES			3478 3488	2:00 5:00			3490S	12:00				
16	2270 VALLEY VIEW LN SOUTHEAST VALLEY VIEW VALLEY BRANCH 13904 JOSEY LN 13904 JOSEY LN	EXPRESS EXPRESS 0000004AD43A	RES			3478 3488	2:00 5:00			3490S	12:00				
15	2570 VALLEY VIEW LN 2570 VALLEY VIEW & WILLIAM DODSON 13904 JOSEY LN 13904 JOSEY LN	MIXED STANDARD 0000004AC50F	RES			3478 3488	2:30 6:00			3490S	12:30				
18	2656 VILLA CREEK DR ONE METRO SQUARE WEST LOBBY 13904 JOSEY LN 13904 JOSEY LN	MIXED WALL 0000004ACA53	BUS			3479 3489	2:00 4:30								
18	2665 VILLA CREEK DR TWO METRO SQUARE 13904 JOSEY LN 13904 JOSEY LN	MIXED WALL 0000004ACB00	BUS			3479 3489	2:00 4:30								
1	2695 VILLA CREEK DR TWO METRO SQUARE 13904 JOSEY LN 13904 JOSEY LN	MIXED WALL 0000004AC3DA	BUS			3479 3489	2:00 4:30								
4			S			3479 3489	2:00 4:30								

OPTIONAL FORM 99 (7-90)

FAX TRANSMITTAL

# of pages ▶

To Kim LArson	From Vincent R. Lewis
Dept./Agency	Phone # 972-393-6700
Fax #	Fax # 972-393-6770

NSN 7540-01-317-7368 5099-101 GENERAL SERVICES ADMINISTRATION

SGS-THOMSON
MICROELECTRONICS

FAX Transmission **PATENT DEPARTMENT**

From:	IRENA LAGER	Date:	07/03/97
To:	OFFICE OF PETITIONS	Time:	10:40 AM
Company:	USPTO	FAX #:	703-308-6916

pages sent including this page: 13

Message:

RE: SERIAL NO.: 08/702,911
DOCKET NO.: 96-S-11

Following is a Status Report, and all documents referenced in the Status Report, for the above-referenced case.

FAX RECEIVED
JUL 3 1997
PETITIONS OFFICE

VOICE: 972-466-7511 FAX: 972-466-7044

MS 2346 - 1310 ELECTRONICS DRIVE- CARROLLTON, TX 75006

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re Application of:

Raul Diaz et al. — Art Unit: 2612

Serial No.: 08/702,911 — Docket No.: 96-S-11

Filed: August 23, 1996

FOR: Video and/or Audio Decompression and/or Compression Device That Shares a Memory Interface

STATUS REPORT

Hon. Assistant Commissioner
for Patents
Washington, D.C. 20231

Sir:

Please provide us with a Status Report on the above-referenced matter. A copy of the "Petition to Secure Filing Date as of Mail Date Via Express Mail" mailed February 24, 1997 along with a copy of the stamped return postcard indicating a receipt date of February 26, 1997 is enclosed. Please charge any fees necessary for prosecution of the present application to deposit account no. 19-1353.

Respectfully submitted:

Irena Lager
Reg. No. 39,260
Attorney for Applicant

FAX RECEIVED
JUL 3 1997
PETITIONS OFFICE

SGS-THOMSON Microelectronics, Inc.
1310 Electronics Drive/MS 2346
Carrollton, TX 75006
972-466-7511

CERTIFICATE OF MAILING
37 CFR 1.8(a)

I hereby certify that the following papers are being facsimile transmitted to the Patent and Trademark Office (fax No. 703-308-6916) on the date shown below.

July 3, 1997 — Kim Larson

Date — Signature

JUL 03 '97 10:32AM SGS-THOMSON LEGAL

Received in the U.S.P.T.O.:
Re: Raul Diaz et al.
Serial No.: 08/702,911
Docket No.: 96-S-11

1. Petition to Secure Filing Date as of Mailing Date Via Express Mail
2. Declaration of Kimberley K. Larson
3. Express Mail Certificate
4. Express Mail Receipt

Mailed: February 24, 1997

MAIL ROOM
82 FEB 26 1997
PAT. & TRADEMARK OFF.

RECEIVED
MAR 18 1997

#4

2400 6/23/97

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re Application of:
Raul Z. Diaz et al.

Docket No.: 96-S-011

Serial No.: 08/702,911

Examiner:

Filed: August 23, 1996

Art Unit: 2612

For: Video and/or audio decompression and/or compression device that shares a memory interface

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

PETITION TO SECURE FILING DATE AS OF

MAILING DATE VIA EXPRESS MAIL

FAX RECEIVED
JUL 3 1997
PETITIONS OFFICE

Applicant petitions that this application be accorded the filing date on which the papers were sent "Express Mail Post Office to Addressee" mailing label no. EG947362259US on August 23, 1996.

CERTIFICATE OF MAILING (37 CFR 1.8a)
I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to the: Assistant Commissioner for Patents, Washington, D.C. 20231, on Feb 24, 1997.

[signature]
Signature

SUBMISSIONS

Submitted herewith is:

1. A copy of the executed Express Mail certificate with mailing label number EG947362259US.
2. A copy of the Express Mail Receipt No. EG947362259US with a "date in" of August 24, 1996 as entered by the U.S. Postal Service.
3. Declaration of Kimberley K. Larson.
4. A copy of Collection Management System, Collection Point Inventory by Address (CPIA) for the U.S. Post Office, Dallas District.

Applicant respectfully requests that the above-referenced application be accorded a filing date of August 23, 1996 as shown by the attached declaration of Kimberley K. Larson and the attached CPIA from the U.S. Post Office. Applicant had a reasonable basis to believe that the correspondence placed in the Express Mail envelope and deposited in a U.S. Postal Service Mail box on August 23, 1996 would be picked up that same day and, therefore, the Express Mail label would have a "date in" of August 23, 1996. Therefore, Applicants request that the application be accorded a filing date of August 23, 1996 as specified in 37 C.F.R. §1.10(c) in effect on August 23, 1996.

PETITION FEE

The petition fee (37 CFR 1.17(h)) is hereby authorized to be charged to Deposit Account No. 19-1353 of SGS-THOMSON Microelectronics, Inc.

REQUEST FOR REFUND OF PETITION FEE

Because no defect exists in applicants' previous submission, a refund of the petition fee is respectfully requested.

Respectfully submitted,

Irena Lager
Reg. No. 39,260
SGS-THOMSON Microelectronics, Inc.
Mail Station 2346
1310 Electronics Dr.
Carrollton, Texas 75006
(972) 466-7511

Attorney for Applicant

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re Application of:
Raul Z. Diaz et al.

Docket No.: 96-S-011

Serial No.: 08/702,911

Examiner:

Filed: August 23, 1996

Art Unit: 2612

For: Video and/or audio decompression and/or compression device that shares a memory interface

Assistant Commissioner for Patents
Washington, D.C. 20231

FAX RECEIVED
JUL 3 1997
PETITIONS OFFICE

Sir:

DECLARATION OF KIMBERLEY K. LARSON

I, KIMBERLEY K. LARSON, do hereby take oath and swear as follows:

(1) I am employed as a patent secretary in the Patent Department of SGS-THOMSON Microelectronics, Inc.

(2) As part of my duties, I am responsible for the preparation of certain documents for transmittal to the Patent and Trademark Office, including ensuring that proper documents are present to be transmitted, organizing the documents to be transmitted and placing these documents in envelopes for transmittal.

(3) On August 23, 1996, I prepared an Express Mail mailing label bearing mailing label number EG947362259US addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231. I also prepared and signed a Certificate of Mailing by Express Mail dated August 23, 1996 with express mail mailing label number EG947362259US, a copy of which is attached hereto.

(4) Pursuant to the requirements of 37 CFR 1.10, I placed the correspondence in an Express Mail envelope and deposited the Express Mail envelope in a U.S. Express Mail mailbox on Friday, August 23, 1996 around 4:45 p.m. This was prior to the last designated pick-up time of 5:00 p.m for this U.S. Express Mail mailbox. Therefore, the certificate of Express Mailing and the application was deposited with the U.S. Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on August 23, 1996 addressed to the Assistant Commissioner for Patents, Box Patent Applications, Washington, D.C. 20231. Attached is a copy of the collection times sent via facsimile from Vincent Lewis of the Consumer Affairs/Claims Division of the U.S. Postal Service. This facsimile shows the last collection time of 5:00 p.m. for the post office on 13904 Josey Lane where the Express Mail package was deposited.

(5) I am unaware as to the circumstances that caused our express mail package not be picked up and processed by the U.S. Postal Service until the next day, August 24, 1996. The above-referenced application was accorded a filing date of August 26, 1996, instead of August 23, 1996. A copy of the return postcard bearing a cancelled date of August 24, 1996, and an actual receipt date of August 26, 1996 is enclosed, along with a copy of the Express Mail Receipt No. EG947362259US with a postmark date of August 24, 1996.

I declare further that all statements made herein of my own knowledge are true; that all statements made herein on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such wilful false statements may jeopardize the validity of this application and any registration resulting therefrom.

Signed at Carrollton, Texas this 24 day of February, 1997.

By: Kimberley K. Larson
Kimberley K. Larson

STATE OF TEXAS
COUNTY OF DALLAS

Subscribed and sworn to before me this 24th day of February, 1997

Mary L. Hiner
Notary Public

MARY L. HINER
Notary Public, State of Texas
My Commission Expires 02/22/99

Received in the U.S.P.T.O.:

Re: Raul Diaz and Jeff Owen
Docket No.: 96-S-11
For: Video and/or audio decompression and/or compression device that shares a memory interface
Enclosed:

1. Patent specification, claims & Abstract
2. Five (5) pages of drawings
3. Declaration and Power of Attorney
4. Transmittal Letter + 1 copy
5. Certificate of Mailing
6. Assignment and Assignment Recordal Form
7. Check in the amount of $1,396.00

Mailed: August 23, 1996

RECEIVED
SEP 16 1996

DALLAS DISTRICT OPS
951 W BETHEL RD
COPPELL TX 75099-9331

FARMERS BRANCH STATION (214)247-9591

ID	Box Address Description of Address Last Box in the Area Nearest Express Mail Box	Service Class Type of Box Button Number	Area of Box	Weekdays Coll by	AM	Coll by	PM	Saturdays Coll by	AM	Coll by	PM	Holidays Coll by	AM	Coll by	PM
10	11400 LUNA RD WESTWOOD BUSINESS PARK ENTRANCE 13904 JOSEY LN 13904 JOSEY LN	MIXED STANDARD 0000004AC84A	BUS			3478 3486	2:00 4:30			34908	12:00				
11	14280 MARSH LN SE MARSH & SPRING VALLEY 13904 JOSEY LN 13904 JOSEY LN	MIXED STANDARD 0000004AC51D	RES			3478 3484	2:00 4:30			34915	12:00				
13	7 MEDICAL PARKWAY NE WEBB CHAPEL & MEDICAL PKY 13904 JOSEY LN 13904 JOSEY LN	STAMPED STANDARD 0000004AC74A	BUS			3473 3483	2:00 4:30			34918	12:00				
17	7 MEDICAL PARKWAY NE WEBB CHAPEL & MEDICAL PKY 13904 JOSEY LN 13904 JOSEY LN	METERED STANDARD 0000004ACF6C	BUS			3473 3483	2:00 4:30			34919	12:00				
18	2270 VALLEY VIEW LN SOUTHEAST VALLEY VIEW VALLEY BRANCH 13904 JOSEY LN 13904 JOSEY LN	MIXED STANDARD 0000004AC5EE	RES			3476 3488	2:00 6:00			34908	12:00				
18	2270 VALLEY VIEW LN SOUTHEAST VALLEY VIEW VALLEY BRANCH 13904 JOSEY LN 13904 JOSEY LN	EXPRESS EXPRESS 0000004AD43A	RES			3478 3488	2:00 6:00			34908	12:00				
18	2570 VALLEY VIEW LN 2570 VALLEY VIEW & WILLIAM DODSON 13904 JOSEY LN 13904 JOSEY LN	MIXED STANDARD 0000004AC50F	RES			3478 3488	2:30 6:00			34908	12:30				
[illegible]	2055 VILLA CREEK DR ONE METRO SQUARE WEST LOBBY 13904 JOSEY LN 13904 JOSEY LN	MIXED WALL 0000004ACA5A	BUS			3479 3489	2:00 4:30								
[illegible]	2005 VILLA CREEK DR TWO METRO SQUARE 13904 JOSEY LN 13904 JOSEY LN	MIXED WALL 0000004AC800	BUS			3479 3489	2:00 4:30								
	2005 VILLA CREEK DR TWO METRO SQUARE 13904 JOSEY LN 13904 JOSEY LN	MIXED WALL 0000004AC3DA	BUS			3479 3489	2:00 4:30								
			S			3479 3489	2:00 4:30								

OPTIONAL FORM 99 (7-90)

FAX TRANSMITTAL # of pages ▶

To Kim LArson
From Vincent R. Lewis
Dept/Agency
Phone # 972-393-6700
Fax #
Fax # 972-393-6770

NSN 7540-01-317-7368 5099-101 GENERAL SERVICES ADMINISTRATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Docket No. 96-S-011

In Re Application of:

Raul Z. Diaz and Jefferson E. Owen

For: **Video and/or Audio Decompression and/or Compression Device that Shares a Memory Interface**

CERTIFICATE OF EXPRESS MAIL

"EXPRESS MAIL" NO. EG947362259US

EG947362259US

Date of Deposit: August 23, 1996

I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Assistant Commissioner for Patents, Box Patent Application, Washington, D.C. 20231.

Signature of person mailing paper or fee

POST OFFICE TO ADDRESSEE EXPRESS MAIL EMS

EG947362259US

ORIGIN (POSTAL USE ONLY)

INTERNATIONAL SHIPMENTS ONLY

Customs Form and International Mail Manual

PO ZIP

Date In Mo. Day Yr.

Time In AM PM

Weight lbs. ozs.

No Delivery Weekend Holiday

Day of Delivery Next Second

12 Noon 3 PM

Int'l Alpha Country Code

Acceptance Clerk Initials

Flat Rate Envelope

Postage $

Total Postage & Fees

SEE REVERSE SIDE FOR THE SERVICE GUARANTEE AND LIMITS ON THE INSURANCE COVERAGE

CUSTOMER COPY

CUSTOMER USE ONLY

METHOD OF PAYMENT: X750-295

Express Mail Corporate Acct. No.

Federal Agency Acct. No. or Postal Service Acct. No.

WAIVER OF SIGNATURE (Domestic Only) I wish delivery to be made without obtaining the signature of the addressee or the addressee's agent (if in the judgement of the delivery employee, the article can be left in a secure location) and I authorize the delivery employee to sign that the shipment was delivered and understand that the signature of the delivery employee will constitute valid proof of delivery.

NO DELIVERY: WEEKEND HOLIDAY

Customer Signature

FROM: (PLEASE PRINT) PHONE

Lisa K. Jorgenson
SGS-THOMSON Microelectronics
1310 Electronics Dr.
MS 2346
Carrollton, TX 75006

96-8-13

TO: (PLEASE PRINT) PHONE

Assistant Commissioner for Patents
Box Patent Applications
Washington, D.C. 20231

LABEL 11-B 5/94

For Pickup or Tracking Call 1-800-222-1811

UNITED STA. ES DEPARTMENT OF COMMERCE
Patent and Trademark Office
ASSISTANT SECRETARY AND COMMISSIONER OF
PATENTS AND TRADEMARKS
Washington, D.C. 20231

#5

Lisa K. Jorgenson
SGS Thomson Microelectronics Inc.
1310 Electronics Drive
Carrollton, TX 75006

COPY MAILED

AUG 11 1997

OFFICE OF PETITIONS
A/C PATENTS

In re Application of	:	
Raul Z. Diaz et al.	:	DECISION DISMISSING
Application No. 08/702,911	:	PETITION
Filed: August 26, 1996	:	
Attorney Docket No. 96-S11	:	

This is a decision on the petition filed February 26, 1997, requesting that the above-identified application be accorded a filing date of August 23, 1996, rather than the presently accorded filing date of August 26, 1996.

Petitioners request the earlier filing date on the basis that the application was purportedly deposited in Express Mail service on August 23, 1996, pursuant to the requirements of 37 CFR 1.10. Petitioners acknowledge that the date of deposit in Express Mail shown on petitioners' Express Mail receipt is August 24, 1996, a Saturday, but argue that the application was actually deposited in an Express Mail drop box on August 23, 1996, before the last scheduled pick up for the day.

Paragraph (c) of 37 CFR 1.10 stated on August 23, 1996, that:

> "the...Office will accept the certificate of mailing by 'Express Mail' and accord the paper or fee the certificate date under 35 U.S.C. 21(a)... without further proof of the date on which the mailing by 'Express Mail' occurred unless a question is present regarding the date of mailing."

However, on May 16, 1995, the Commissioner waived the requirement for a certificate of mailing under 37 CFR 1.10. See 1174 Off. Gaz. Pat. Office 92. The Commissioner noted that the certificate of mailing under 37 CFR 1.10 is redundant in view of the fact that the date of deposit in Express Mail is entered as the "Date-In" on the Express Mail label by the U.S. Postal Service (USPS). Thus, the PTO considers the date "the paper or fee is shown to have been deposited as Express Mail" to be the "Date-In" on the Express Mail label.

Placing the "Date-In" on the Express Mail label or receipt by the postal clerk establishes that the package was actually received by the USPS. That is the date that verifies that the package was actually mailed. Accordingly, the application was accorded a filing date of August 26, 1996, the next business following the date of deposit in Express Mail service.

Petitioners allege that the date of deposit in Express Mail shown by petitioners' Express Mail receipt is a USPS error. In support, the petition is accompanied by a declaration of Kimberley K. Larson stating her recollection that the Express Mail package she deposited six (6) months earlier was deposited in an Express Mail drop box at about 4:45 p.m. and that the last pickup for the day at that particular drop box was 5:00 p.m.

The arguments and evidence presented have been considered, but are not persuasive. Petitioners' Express Mail receipt is considered to be more probative of the correct date of deposit than the declaration presented with the petition, because the Express Mail receipt was completed by the USPS contemporaneously with the processing of the Express Mail package by the USPS while the Larson declaration was made six (6) months after the date of mailing. Postal employees are presumed to discharge their duties in a proper manner. Charlson Realty Co. v. United States, 690 F.2d 434, 442 (Ct. Cl. 1967). Therefore, in view of the August 24, 1996 "date-in" shown on the Express Mail receipt it is concluded that the Express Mail package in question must have been deposited either after the last scheduled pickup for the day on August 23, 1996, or on Saturday, August 24, 1996. It is petitioners' burden to establish to the satisfaction of the Commissioner that the August 24, 1996 "Date-In" on Express Mail label No. EG947362259US is the result of an error on the part of an employee of the USPS. However, no statement from the USPS has been presented verifying that any error was made by the USPS in the processing of petitioners' Express Mail package. Petitioners were made aware of the date of deposit in Express Mail acknowledged by the USPS upon return of petitioners' Express Mail receipt to counsel's office. It is not understood why petitioners did not obtain a statement in writing from the USPS acknowledging an error in the processing of petitioners' Express Mail package immediately after receiving the Express Mail receipt. It is also unfortunate that petitioners chose to deposit a paper as important as a patent application in Express Mail without immediately obtaining an Express Mail receipt showing the desired date of deposit. Since the date of mailing is established by the rule as the date shown by the Express Mail receipt and petitioners' receipt shows a date of mailing of August 24, 1996, the application was correctly accorded a filing date of August 26, 1996.

The petition is dismissed.

Any request for reconsideration should be filed within TWO MONTHS of the date of this decision in order to be considered timely. See 37 CFR 1.181(f). This time period may not be extended pursuant to 37 CFR 1.136(a) or (b).

Further correspondence with respect to this matter should be addressed as follows:

By mail:	Assistant Commissioner for Patents Box DAC Washington, D.C. 20231
By FAX:	(703) 308-6916 Attn: Special Program Law Office
By hand:	One Crystal Park, Suite 520 2011 Crystal Drive Arlington, VA

The application is being returned to Examining Group 2400 for examination in due course with the presently accorded filing date of August 26, 1996.

Telephone inquiries specific to this matter should be directed to the undersigned at (703) 305-9282.

John F. Gonzales
Senior Legal Advisor
Special Program Law Office
Office of the Deputy Assistant Commissioner
for Patent Policy and Projects

JFG

UNITED STATES DEPARTMENT OF COMMERCE
Patent and Trademark Office
Address: COMMISSIONER OF PATENTS AND TRADEMARKS
Washington, D.C. 20231

NM

APPLICATION NUMBER	FILING DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NO.
08/702,911	08/26/96	DIAZ	R 96-S-11

B3M1/0903

LISA K JORGENSON
SGS THOMSON MICROELECTRONICS INC
1310 ELECTRONICS DRIVE
CARROLLTON TX 75006

EXAMINER	
RAMIREZ, E	
ART UNIT	PAPER NUMBER
2414	6

DATE MAILED: 09/03/97

This is a communication from the examiner in charge of your application.
COMMISSIONER OF PATENTS AND TRADEMARKS

OFFICE ACTION SUMMARY

☐ Responsive to communication(s) filed on ______.

☐ 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 D.C. 11; 453 O.G. 213.

A shortened statutory period for response to this action is set to expire 03 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) 1-49 is/are pending in the application.

Of the above, claim(s) ______ is/are withdrawn from consideration.

☐ Claim(s) ______ is/are allowed.

☒ Claim(s) 1-49. is/are rejected.

☐ Claim(s) ______ is/are objected to.

☐ Claims ______ 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 Reference Cited, PTO-892

☒ Information Disclosure Statement(s), PTO-1449, Paper No(s). 2 & 3

☐ 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 --

PTOL-326 (Rev. 10/95)

1. This Application has been examined.

2. This application has been filed with informal drawings which are acceptable for examination purposes only. Formal drawings will be required when the application is allowed.

3. 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 --
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

4. Claims 1-49 are rejected under 35 U.S.C. 102(B) as being clearly anticipated by Lin et al..

The publication of Lin discloses in Figure one a MPEG II decoder which employs memory interface which uses bus arbitration to allocate resources between the decoder and another device. The arbitration scheme, see figure 3, allocates access to the memory on a priority level. The basic requirement of having the interface and the decoder be monolithic is presumed from figures one since both component are found in the rectangle box, and the introduction which discloses uses in " TV set-top boxes, PC add-on and entertainment machines."

5. 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.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103© and potential 35 U.S.C. 102(f) or (g) prior art under 35 U.S.C. 103(a).

6. Claims 1-49 are rejected under 35 U.S.C. 103(a) as being unpatentable over Retter et al. (US Patent 5,557,538) in view of Harney (US Patent 5,522,080).

Retter discloses the prior art devices employed in the compression and decompression of video and audio data. Retter diploses most of the requirements of the claimed invention with the exception of having a memory interface employing an arbitration routine. Though not expressly reciting an arbitration routine the patent does concern itself with managing " an internal bidirectional bus on which all data transfers between the external DRAM buffer and all the internal units." It would stand to reason that Retter must determine who has priority of the DRAM or must settle contentions if one arises when clients are accessing the same resource simultaneously. The Patent to Harney discloses a an arbitration scheme in the same environment as the patent to Retter:

> Additionally, it is permitted to impose no relative local priority within Group III if all three types of request are equally likely and equally important. In this alternate embodiment of method x the transfers

of Group III are processed on a first come, first served basis. There is by, definition arbitration between transfers involving the system bus because system 300 contains two hi-directional burst buffers which allow up to sixteen word transfers to be captured. It also contains a set of dual scaler buffers, which are used to capture scaler accesses. These buffers allow scaler and burst request to be performed even if the bus resource is unavailable.
With respect to Group III local arbitration , the band width requirements are highly dependent upon system configuration.
The functions performed by block transfer controller 368 during block data transfers of data between global memory 366 and local memories 362a-n include arbitration of transfer requests of competing global memory 366 and execution units 360a-n, address generation and control for two-dimensional block transfers between global memory 366 and local memories 362a-n, control for scalar, first-in first-out, and statistical decoder transfers between local memories 362a-n and global memory 366 and address generation and control for block instruction load following cache miss.
A number of different types of transfers requiring input/output access arbitration by block transfer controller 368 may take place between global memory 366 and local memories 362a-n. These include fetching instructions from global memory 366. Image processor system 300 initializes the process by downloading instructions from system memory 364 to global memory 366. On power up of image processor 300 the instructions are loaded from system memory 364 or global memory 366 into the controller.
Different types of transfers may be prioritized by block transfer controller 368 as follows, proceeding from highest priority to lowest priority: (1) instruction, (2) scalar, first-in, first-out and statistical decoder, and (3) block transfer. Thus block transfer controller 368 not only prioritizes and arbitrates within image processor 300 based upon whether a request is an instruction type of request or a data type of request. Block transfer controller 368 also prioritizes and arbitrates based upon the subtypes of data. Column 21, lines 1-30.

Harney recognizes the problem associated with systems, like Retter, which have multiple processors requiring access to the same resource. In column 3, lines 9-12, Harney discloses that to allow each datapath (i.e., process) to have a independent access to external memory is impractical for semiconductor implementation. A sharing of resources is proposed and elucidated upon by the patent.

Therefore, it would have been obvious to those of ordinary skill in the art, at the time of the invention, to combine the Retter patent with the patent of Harney because when the same resource is required by multiple users and since it would be impractical to have independent or exclusive access, an arbitration scheme would insures the most optimal means of assuring prompt resolution of contentions and a judicial process for allocating the memory pie.

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to **Primary Examiner Ellis B. Ramirez, Esq.,** whose telephone number is (703) 305-9786. The examiner can normally be reached on Monday-Friday from 7:30 AM to 6:00 PM . If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, E. Voeltz, can be reached on (703) 305-9714. Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the Group receptionist whose telephone number is (703) 305-3800.

Any response to this action should be mailed to:

Commissioner of Patents and Trademarks
Washington, D.C. 20231

or faxed to:

(703) 308-9051, (for formal communications intended for entry)

Or:

(703)308-5356 (for informal or draft communications, please label "PROPOSED" or "DRAFT")

Hand-delivered responses should be brought to Crystal Park II, 2121 Crystal Drive, Arlington. VA., Sixth Floor (Receptionist).

ELLIS B. RAMIREZ
PRIMARY EXAMINER
GROUP 2400

Notice of References Cited

Application No.	Applicant(s)		
08/702,911	Diaz et al.		
Examiner		Group Art Unit	
Ellis B. Ramirez		2414	Page 1 of 1

U.S. PATENT DOCUMENTS

	DOCUMENT NO.	DATE	NAME	CLASS	SUBCLASS
A	5,557,538	9/17/96	Retter et al.	348	402
B	5,598,525	1/28/97	Nally et al.	395	507
C	5,027,400	6/25/91	Baji et al.	348	10
D	5,371,893	12/6/94	Price et al.	395	729
E	5,623,672	4/22/97	Poppat	395	729
F	5,522,080	5/28/96	Harney et al.	395	729
G	5,621,893	4/15/97	Joh	395	200.82
H	4,774,660	9/27/88	Conforti	395	729
I	4,894,565	1/16/90	Marquardt	395	729
J					
K					
L					
M					

FOREIGN PATENT DOCUMENTS

	DOCUMENT NO.	DATE	COUNTRY	NAME	CLASS	SUBCLASS
N						
O						
P						
Q						
R						
S						
T						

NON-PATENT DOCUMENTS

	DOCUMENT (Including Author, Title, Source, and Pertinent Pages)	DATE
U	"ON THE BUS ARBITRATION FOR MPEG 2 VIDEO DECODER" ; VLSI Tech, System And Application, !995 Symposium	1995
V	"A LOW COST GRAPHICS AND MULTIMEDIA WORKSTATIONM CHIP SET"; IEEE Micro	1994
W		
X		

#7
S. Bond
3/20/98

MAR 09 1998

PATENT

I hereby certify that on the date specified below, this correspondence is being deposited with the United States Postal Service as first-class mail in an envelope addressed to the Assistant Commissioner for Patents, 2011 Jefferson Davis Highway, Washington, DC 20231.

March 3, 1998 — Date

E. Russell Tarleton (signature) — E. Russell Tarleton

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants	:	Raul Z. Diaz et al.
Application No.	:	08/702,911
Filed	:	August 26, 1996
For	:	VIDEO AND/OR AUDIO DECOMPRESSION AND/OR COMPRESSION DEVICE THAT SHARES A MEMORY INTERFACE

Examiner	:	E. Ramirez
Art Unit	:	2414
Docket No.	:	96-S-11 (850063.517)
Date	:	March 3, 1998

RECEIVED MAR 17 98 GROUP 2600

Assistant Commissioner for Patents
2011 Jefferson Davis Highway
Washington, DC 20231

PETITION FOR AN EXTENSION OF TIME

UNDER 37 C.F.R. § 1.136(a)

Sir:

Applicants herewith petition the Assistant Commissioner of Patents under 37 C.F.R. § 1.136(a) for a three-month extension of time for filing the response to the Examiner's Action dated September 3, 1997, from December 3, 1997 to March 3, 1998. Submitted herewith is a check in the amount of $950 to cover the cost of the extension.

Any deficiency or overpayment should be charged or credited to Deposit Account No. 19-1090. This petition is being submitted in triplicate.

Respectfully submitted,
Raul Z. Diaz et al.
SEED and BERRY LLP

E. Russell Tarleton (signature)

E. Russell Tarleton
Registration No. 31,800

ERT:jb
Enclosures:
Two copies of this Petition

6300 Columbia Center
701 Fifth Avenue
Seattle, Washington 98104-7092
(206) 622-4900
Fax: (206) 682-6031
users:\joanneb\ert98\0092

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants	:	Raul Z. Diaz et al.
Application No.	:	08/702,911
Filed	:	August 26, 1996
For	:	VIDEO AND/OR AUDIO DECOMPRESSION AND/OR COMPRESSION DEVICE THAT SHARES A MEMORY INTERFACE

Examiner	:	E. Ramirez
Art Unit	:	2414
Docket No.	:	96-S-11 (850063.517)
Date	:	March 3, 1998

Assistant Commissioner for Patents
2011 Jefferson Davis Highway
Washington, DC 20231

GENERAL AUTHORIZATION UNDER 37 C.F.R. § 1.136(a)(3)

Sir:

With respect to the above-identified application, the Assistant Commissioner is authorized to treat any concurrent or future reply requiring a petition for an extension of time under 37 C.F.R. § 1.136(a)(3) for its timely submission as incorporating a petition therefor for the appropriate length of time. The Assistant Commissioner is also authorized to charge any fees which may be required, or credit any overpayment, to Deposit Account No. 19-1090.

Respectfully submitted,
Raul Z. Diaz et al.
SEED and BERRY LLP

E. Russell Tarleton

E. Russell Tarleton
Registration No. 31,800

ERT:jb
SEED and BERRY LLP
6300 Columbia Center
701 Fifth Avenue
Seattle, Washington 98104-7092
(206) 622-4900
FAX: (206) 682-6031

users:\joanneb\ert98\0091

PATENT

I hereby certify that on the date specified below, this correspondence is being deposited with the United States Postal Service as first-class mail in an envelope addressed to the Assistant Commissioner for Patents, 2011 Jefferson Davis Highway, Washington, DC 20231.

March 3, 1998	E. Russell Tarleton
Date	E. Russell Tarleton

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants	:	Raul Z. Diaz et al.
Application No.	:	08/702,911
Filed	:	August 26, 1996
For	:	VIDEO AND/OR AUDIO DECOMPRESSION AND/OR COMPRESSION DEVICE THAT SHARES A MEMORY INTERFACE

Examiner	:	E. Ramirez
Art Unit	:	2414
Docket No.	:	96-S-11 (850063.517)
Date	:	March 3, 1998

RECEIVED MAR 17 98 GROUP 2600

Assistant Commissioner for Patents
2011 Jefferson Davis Highway
Washington, DC 20231

PETITION FOR AN EXTENSION OF TIME

UNDER 37 C.F.R. § 1.136(a)

Sir:

Applicants herewith petition the Assistant Commissioner of Patents under 37 C.F.R. § 1.136(a) for a three-month extension of time for filing the response to the Examiner's Action dated September 3, 1997, from December 3, 1997 to March 3, 1998. Submitted herewith is a check in the amount of $950 to cover the cost of the extension.

Any deficiency or overpayment should be charged or credited to Deposit Account No. 19-1090. This petition is being submitted in triplicate.

Respectfully submitted,
Raul Z. Diaz et al.
SEED and BERRY LLP

E. Russell Tarleton

E. Russell Tarleton
Registration No. 31,800

ERT:jb
Enclosures:
Two copies of this Petition

6300 Columbia Center
701 Fifth Avenue
Seattle, Washington 98104-7092
(206) 622-4900
Fax: (206) 682-6031
users:\joanneb\ert98\0092

MAR 0 9 1998

PATENT

I hereby certify that on the date specified below, this correspondence is being deposited with the United States Postal Service as first-class mail in an envelope addressed to the Assistant Commissioner for Patents, 2011 Jefferson Davis Highway, Washington, DC 20231.

March 3, 1998 — E. Russell Tarleton

Date — E. Russell Tarleton

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants	:	Raul Z. Diaz et al.
Application No.	:	08/702,911
Filed	:	August 26, 1996
For	:	VIDEO AND/OR AUDIO DECOMPRESSION AND/OR COMPRESSION DEVICE THAT SHARES A MEMORY INTERFACE
Examiner	:	E. Ramirez
Art Unit	:	2414
Docket No.	:	96-S-11 (850063.517)
Date	:	March 3, 1998

RECEIVED MAR 17 98 GROUP 2600

Assistant Commissioner for Patents
2011 Jefferson Davis Highway
Washington, DC 20231

PETITION FOR AN EXTENSION OF TIME
UNDER 37 C.F.R. § 1.136(a)

Sir:

Applicants herewith petition the Assistant Commissioner of Patents under 37 C.F.R. § 1.136(a) for a three-month extension of time for filing the response to the Examiner's Action dated September 3, 1997, from December 3, 1997 to March 3, 1998. Submitted herewith is a check in the amount of $950 to cover the cost of the extension.

Any deficiency or overpayment should be charged or credited to Deposit Account No. 19-1090. This petition is being submitted in triplicate.

Respectfully submitted,
Raul Z. Diaz et al.
SEED and BERRY LLP

E. Russell Tarleton

E. Russell Tarleton
Registration No. 31,800

ERT:jb
Enclosures:
Two copies of this Petition

6300 Columbia Center
701 Fifth Avenue
Seattle, Washington 98104-7092
(206) 622-4900
Fax: (206) 682-6031
users:\joanneb\ert98\0092

FORM PTO-1083

S ,-THOMSON MICROELECTRONI INC.
1310 Electronics Drive
Carrollton, Texas 75006-5039
Phone (972) 466-6000
Fax (972) 466-7044

Gau 2414 ~~/~~ 2756

MAR 0 9 1998

Docket No.: **96-S-11**
Date: **March 3, 1998**

#8

In re application of **Raul Z. Diaz et al.**
Application No.: **08/702,911**
Filed: **August 26, 1996**
For: **VIDEO AND/OR AUDIO DECOMPRESSION AND/OR COMPRESSION DEVICE THAT SHARES A MEMORY INTERFACE**

ASSISTANT COMMISSIONER FOR PATENTS
2011 JEFFERSON DAVIS HIGHWAY
WASHINGTON DC 20231

RECEIVED
MAR 17 98
GROUP 2600

Sir:

Transmitted herewith is **an amendment** in the above-identified application.

[X] A Petition for an Extension of Time for three months is enclosed.
[X] No additional claim fee is required.
[] The fee has been calculated as shown.

	(Col. 1) CLAIMS REMAINING AFTER AMENDMENT		(Col. 2) HIGHEST PREV. PAID FOR	(Col. 3) PRESENT EXTRA	SMALL ENTITY RATE	SMALL ENTITY ADDITIONAL FEE		OTHER THAN A SMALL ENTITY RATE	OTHER THAN A SMALL ENTITY ADDITIONAL FEE
TOTAL	* 33	MINUS	** 49	0	x 11	$	OR	x 22	$
INDEP.	* 3	MINUS	*** 5	0	x 41	$		x 82	$
[ ] FIRST PRESENTATION OF MULTIPLE CLAIMS					+ 135	$	OR	+ 270	$
EXTENSION OF TIME FEE						$			$ 950
TOTAL ADDITIONAL FEE						$	TOTAL		$ 950

* If the entry in Col. 1 is less than the entry in Col. 2, write "0" in Col. 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, write "20" in this space.
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, write "3" in this space.
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found from the equivalent box in Col. 1 of a prior amendment or the number of claims originally filed.

[] Please charge my Deposit Account No. 19-1090 in the amount of $_. A duplicate copy of this sheet is enclosed.
[X] A check in the amount of $ 950 is attached.
[X] The Assistant Commissioner is hereby authorized to charge payment of the following additional fees associated with this communication or credit any overpayment to Deposit Account No. 19-1090 . A duplicate copy of this sheet is enclosed.
  [X] Any filing fees under 37 CFR 1.16 for the presentation of extra claims.
  [X] Any patent application processing fees under 37 CFR 1.17.

03/11/1998 BHINES 00000030 08702911
01 FC:117 950.00 OP

Respectfully submitted,

SGS-Thomson Microelectronics, Inc.

E. Russell Tarleton

E. Russell Tarleton
Registration No. 31,800

users:\joanneb\ert98\0093

FORM PTO-1083

S-THOMSON MICROELECTRONI , INC.
1310 Electronics Drive
Carrollton, Texas 75006-5039
Phone (972) 466-6000
Fax (972) 466-7044

#8

MAR 09 1998

Docket No.: **96-S-11**
Date: **March 3, 1998**

In re application of **Raul Z. Diaz et al.**
Application No.: **08/702,911**
Filed: **August 26, 1996**
For: **VIDEO AND/OR AUDIO DECOMPRESSION AND/OR COMPRESSION DEVICE THAT SHARES A MEMORY INTERFACE**

ASSISTANT COMMISSIONER FOR PATENTS
2011 JEFFERSON DAVIS HIGHWAY
WASHINGTON DC 20231

RECEIVED MAR 17 98 GROUP 2600

Sir:

Transmitted herewith is an **amendment** in the above-identified application.

[X] A Petition for an Extension of Time for three months is enclosed.
[X] No additional claim fee is required.
[] The fee has been calculated as shown.

	(Col. 1)		(Col. 2)	(Col. 3)	SMALL ENTITY			OTHER THAN A SMALL ENTITY	
	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST PREV. PAID FOR	PRESENT EXTRA	RATE	ADDITIONAL FEE	OR	RATE	ADDITIONAL FEE
TOTAL	* 33	MINUS	** 49	0	x 11	$		x 22	$
INDEP.	* 3	MINUS	*** 5	0	x 41	$		x 82	$
[] FIRST PRESENTATION OF MULTIPLE CLAIMS					+ 135	$	OR	+ 270	$
EXTENSION OF TIME FEE						$			$ 950
TOTAL ADDITIONAL FEE						$	TOTAL		$ 950

* If the entry in Col. 1 is less than the entry in Col. 2, write "0" in Col. 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, write "20" in this space.
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, write "3" in this space.
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found from the equivalent box in Col. 1 of a prior amendment or the number of claims originally filed.

[] Please charge my Deposit Account No. 19-1090 in the amount of $_. A duplicate copy of this sheet is enclosed.
[X] A check in the amount of $ 950 is attached.
[X] The Assistant Commissioner is hereby authorized to charge payment of the following additional fees associated with this communication or credit any overpayment to Deposit Account No. 19-1090 . A duplicate copy of this sheet is enclosed.
  [X] Any filing fees under 37 CFR 1.16 for the presentation of extra claims.
  [X] Any patent application processing fees under 37 CFR 1.17.

Respectfully submitted,

SGS-Thomson Microelectronics, Inc.

E. Russell Tarleton

E. Russell Tarleton
Registration No. 31,800

users:\joanneb\ert98\0093

#8a
S. Sand
3/20/98

1

PATENT

MAR 09 1998

I hereby certify that on the date specified below, this correspondence is being deposited with the United States Postal Service as first-class mail in an envelope addressed to the Assistant Commissioner for Patents, 2011 Jefferson Davis Highway, Washington, DC 20231.

March 3, 1998 — Date

E. Russell Tarleton — E. Russell Tarleton

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants	:	Raul Z. Diaz et al.
Application No.	:	08/702,911
Filed	:	August 26, 1996
For	:	VIDEO AND/OR AUDIO DECOMPRESSION AND/OR COMPRESSION DEVICE THAT SHARES A MEMORY INTERFACE
Examiner	:	E. Ramirez
Art Unit	:	2414
Docket No.	:	96-S-11 (850063.517)
Date	:	March 3, 1998

Assistant Commissioner for Patents
2011 Jefferson Davis Highway
Washington, DC 20231

RECEIVED
MAR 17 98
GROUP 2600

AMENDMENT

Sir:

In response to the Office Action dated September 3, 1997, please extend the period of time for response three months, to expire on March 3, 1998. Enclosed are a Petition for an Extension of Time and the requisite fee. Please amend the application as follows:

In the Specification:

On page 5, line 11, please replace "188" with -- 198 --.

On page 5, line 17, please replace "suggest" with -- suggests --.

On page 6, line 5, please replace "frame" with -- frames --.

On page 6, line 8, please replace "standards" with -- standard --.

On page 6, line 11, please delete the "," after "acceptable".

On page 12, line 21, please replace "12 circuit" with -- circuit 12 --.

On page 13, line 15, please replace "software. Presenting" with – software, presenting --.

On page 14, line 5, please replace "access" with -- accesses --.

On page 15, please delete the "," after "busses".

On page 15, line 23, please delete the "," after "memory".

On page 17, lines 18 and 19, please replace "standards. Possibly" with -- standards, possibly --.

On page 17, line 23, please replace "Meaning that compression" with -- This means that the compression --.

On page 17, line 26, please replace "complicated is it" with -- complicated, it is --.

On page 18, line 8, please delete "to be able".

On page 18, line 9, please replace "The encoding to comply" with -- Having the encoding comply --.

On page 18, line 10, please insert -- standards -- between "MPEG-2" and "balances".

On page 21, line 14, please replace "the are no a request" with -- there are no requests --.

On page 21, line 19, please replace "before of after" with -- before or after --.

On page 23, line 21, please replace "a image" with -- an image --.

In the Claims:

Please cancel claims 1-13, 42-43, and 49, and amend claims 14, 26, 28, 41, and 44 as follows:

a1

1 14. (Amended) An electronic system coupled to a memory, comprising:

a first device that requires access to the memory;

a decoder that requires access to the memory sufficient to maintain real time operation;

[a fast bus coupled to the first device and the decoder;] and

a memory interface for coupling to the memory,. and coupled to the first device[,] and to the decoder, the memory interface having an arbiter for selectively providing access for the first device and the decoder to the memory <u>and a shared bus coupled to the memory, the first device, and the decoder, the bus having a sufficient bandwidth to enable the decoder to access the memory and operate in real time when the first device simultaneously accesses the bus</u>.

35

13 ~~26~~. (Amended) The electronic system of claim 1 ~~14~~, wherein the [fast] bus has a bandwidth of [greater than a threshold] at least twice the bandwidth required for the decoder to operate in real time.

15 ~~28~~. (Amended) A computer comprising:

processing means;

an input device connected to the processing means;

an output device connected to the processing means;

a memory connected to the processing means;

a first device that requires access to the memory;

a decoder that requires access to the memory sufficient to maintain real time operation; and

a memory interface coupled to the memory, to the first device, and to the decoder, the memory interface having a means for selectively providing access for the first device and the decoder to the memory and a shared bus coupled to the decoder, the first device, and the memory, the shared bus having a sufficient bandwidth to enable the decoder to operate in real time while sharing access to the bus.

28 ~~41~~. (Amended) The computer of claim 15 ~~28~~, [further comprising a fast] wherein the shared bus [coupled to the memory, to the decoder to the first device] has at least twice the required bandwidth for the decoder to operate in real time.

29 ~~44~~. (Amended) In an electronic system having a first device coupled to a memory interface and a memory coupled to the memory interface, the first device having a device priority and capable of generating a request to access the memory, a method for selectively providing access to the memory comprising the steps of:

providing a decoder coupled to the memory interface[,] through a bus having sufficient bandwidth to enable the decoder [capable of operating] to operate in real time while sharing access to the bus, having a decoder priority and capable of generating a request to access the memory;

providing an arbiter having an idle, a busy and a queue state;

generating a request by the decoder to access the memory;

determining the state of the arbiter;

providing the decoder access to the memory responsive to the arbiter being in the idle state for the decoder to operate in real time;

queuing the request responsive to the arbiter being in the busy state; and

queuing the request responsive to the arbiter being in the queue state in an order responsive to the priority of the decoder request and the priority of any other queued requests.

REMARKS

In the first Office Action, claims 1-49 were rejected under 35 U.S.C. § 102(b) in view of Lin et al., which is an article entitled "On the Bus Arbitration for MPEG II Video Decoder." Claims 1-49 were also rejected under 35 U.S.C. § 103(a) over Retter et al. (U.S. Patent No. 5,557,538) in view of Harney (U.S. Patent No. 5,522,080).

Applicants respectfully disagree with the bases for the rejections and request reconsideration and withdrawal of the rejections.

Some of the technical differences between the applied references and embodiments of the invention will now be discussed. Of course, these discussed differences regarding the embodiments, which are disclosed in detail in the patent specification, do not define the scope of interpretation of any of the claims; where presented below, such discussed differences merely help the Examiner appreciate important claim distinctions discussed thereafter. Embodiments of the present invention are directed to systems and method for coupling memory to a plurality of devices through a single shared bus that enables one of the devices, such as a decoder, to operate in real time. The shared bus has a bandwidth that is at least the required bandwidth for the decoder to operate in real time, and preferably at least twice the size of the required decoder bandwidth.

Lin et al. discusses a scheme for assigning duration rates to input/output tasks to ensure the decoding duration rate is greater than the display input requirements. Lin et al. does not suggest using a single shared bus having a bandwidth of sufficient size to permit real time decoding when sharing the bus with one or more other devices.

Retter et al., U.S. Patent No. 5,557,538, is directed to the management of internal bidirectional busses for data transfers with DRAM and other internal units. The arbitration scheme proposed by Retter et al. does not suggest or disclose a bus having a bandwidth sufficient to permit a decoder to operate in real time while sharing bus access with one or more other devices, and Retter et al. does not disclose the claimed arbitration method for accomplishing real time operation of the decoder.

Harney, U.S. Patent No. 5,522,080, is not concerned with real time operation and does not teach or suggest a bandwidth of at least the required bandwidth for the decoder to operate in real time while simultaneously sharing bus access with one or more other devices. Rather, each of the devices in Harney has a local memory and interface.

Turning to the claims, independent claim 14, as amended, is directed to an electronic system coupled to a memory that comprises a "first device requiring access to the memory", a "decoder that requires access to the memory sufficient to maintain real time operation", and a "memory" interface for coupling to the memory, the memory interface being coupled to the first device and the decoder. Claim 14 further recites the memory interface having an arbiter for selectively providing access for the first device and the decoder to the memory through a shared bus, the "shared bus having a sufficient bandwidth to enable the decoder to access the memory and operate in real time when the first device simultaneously accesses the bus"

Lin et al. does not teach or suggest such a system. More particularly, Lin et al. does not disclose a bus having a sufficient bandwidth to enable the decoder to operate and maintain real time operation when simultaneously accessing the shared bus with the first device. Rather, Lin et al. proposes a scheme for assigning duration rates for input/output tasks that ensures the decoding duration rate is greater than the display input requirements.

Retter et al. and/or Harney fail to make up for Lin et al.'s deficiencies. Retter et al. in combination with Harney fails to teach, disclose, or suggest to one of ordinary skill a shared bus having sufficient bandwidth to enable the decoder to operate in real time and maintain real time operation when simultaneously accessing the bus with the first device. While Retter et al. and Harney propose various methods for accomplishing their particular purposes, they do not suggest maintaining real time operation of a device through shared use of a bus interface to memory. Consequently, applicants respectfully submit that claim 14 is allowable over the references cited and applied by the Examiner.

Claim 15, which depends from claim 14, recites the first device and the decoder as being capable of having a variable bandwidth. Nowhere does Lin et al. or the combination of Retter et al. or Harney disclose or suggest to one of ordinary skill a decoder and first device having variable bandwidths in combination with a shared bus having sufficient bandwidth to enable the decoder to operate in real time when simultaneously accessing the bus with the first device. Claim 26, which depends from claim 14, recites the bus as having a bandwidth at least twice the bandwidth required for the decoder to operate in real time. Lin et al., Harney, and Retter et al. are all silent with respect to providing a bus having at least twice the bandwidth required for a decoder to operate in real time. Applicants respectfully submit that claims 15-26 are allowable for these reasons as well as for the reasons why claim 14 is allowable.

The remaining claims recite limitations similar to those explained above. Claim 28 is directed to a computer comprising a processing means, an input and output device coupled to the processing means along with a memory, and a first device and decoder coupled to a memory interface through a shared bus that has a sufficient bandwidth to enable the decoder to operate in real time. Claim 29, which depends from claim 28, recites the first device and decoder as having a variable bandwidth. Claim 41, which depends from claim 28, recites the bus as having at least twice the required bandwidth for the decoder to operate in real time. Applicants respectfully submit that claims 28-41 are allowable for the reasons why claims 14-26 are allowable.

Claim 44 recites a method for selectively providing access to the memory of an electronic system having a first device coupled thereto, the method comprising the steps of providing a decoder coupled to the memory interface for operating in real time and having a decoder priority and request generation capability, providing an arbiter having an ideal, busy, and queue state, generating a request by the decoder to access memory, determining the state of the arbiter, and providing the decoder access to the memory for the decoder to operate in real time. As discussed above, neither Lin et al. nor the combination of Harney with Retter et al. teach or suggest to one of ordinary skill a method for providing decoder access to a memory through a shared bus that utilizes a bus of "sufficient bandwidth to enable the decoder to operate in real time while sharing access to the bus". In the prior devices described in Lin et al., Retter et al., and Harney, bandwidth is not discussed in the context of providing real time operation for a decoder and maintaining real time operation while sharing access to the bus. None of the applied references taken individually or in any motivated combination thereof teaches the combination of claimed steps in the recited method. Consequently, applicants respectfully submit that claim 44 and dependent claims 45-48 are allowable over the references cited and applied by the Examiner.

Overall, none of the applied references, taken alone or in any combination thereof apparently teach or suggest the claimed features recited in independent claims 14, 28, and 44, and thus such claims are allowable. Since these independent claims are allowable based on the above reasons, the claims which depend from them are likewise allowable. If the undersigned attorney has overlooked a relevant teaching in any of the references, the Examiner is requested to point out specifically where such teaching may be found.

In light of the foregoing remarks, the Applicant respectfully submits that all pending claims are allowable. The Applicant, therefore, respectfully requests the Examiner to reconsider this application and timely allows all pending claims. Examiner Ramirez is encouraged to contact Mr. Tarleton by telephone to discuss the above and other distinctions between the claims and the applied reference, if desired. If the Examiner notes any informalities in the claims, the Examiner is encouraged to contact Mr. Tarleton to expediently correct any such informalities by telephone.

In view of the forgoing, applicants respectfully submit that all of the claims remaining in this application are now clearly in condition for allowance. Consequently, early and favorable action allowing these claims and passing this case to issuance is respectfully solicited.

Respectfully submitted,

Raul Z. Diaz et al.

SEED and BERRY LLP

E. Russell Tarleton

E. Russell Tarleton
Registration No. 31,800

ERT:alb/jp
Enclosures:
- Postcard
- Check
- Form PTO-1083 (+ copy)
- Petition for an Extension of Time (+ 2 copies)
- General Authorization

6300 Columbia Center
701 Fifth Avenue
Seattle, Washington 98104-7092
(206) 622-4900
Fax: (206) 682-6031

WPN/850063/517-AM/V4

UNITED STATES DEPARTMENT OF COMMERCE
Patent and Trademark Office
Address: COMMISSIONER OF PATENTS AND TRADEMARKS
Washington, D.C. 20231

APPLICATION NUMBER	FILING DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NO.
08/702,911	08/26/96	DIAZ	R 96-S-11

LM21/0330

LISA K JORGENSON
SGS THOMSON MICROELECTRONICS INC
1310 ELECTRONICS DRIVE
CARROLLTON TX 75006

EXAMINER	
RAMIREZ,E	
ART UNIT	PAPER NUMBER
2756	9

DATE MAILED: 03/30/98

This is a communication from the examiner in charge of your application.
COMMISSIONER OF PATENTS AND TRADEMARKS

NOTICE OF ALLOWABILITY

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.

☒ This communication is responsive to Papers filed on 3/9/98.

☒ The allowed claim(s) is/are 14-41, and 45-48.

☐ The drawings filed on ______ are acceptable.

☐ 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).

A SHORTENED STATUTORY PERIOD FOR RESPONSE 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 obtained under the provisions of 37 CFR 1.136(a).

☐ Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL APPLICATION, PTO-152, which discloses that the oath or declaration is deficient. A SUBSTITUTE OATH OR DECLARATION IS REQUIRED.

☒ Applicant MUST submit NEW FORMAL DRAWINGS

☒ because the originally filed drawings were declared by applicant to be informal.

☐ including changes required by the Notice of Draftperson's Patent Drawing Review, PTO-948, attached hereto or to Paper No. ______.

☐ including changes required by the proposed drawing correction filed on ______, which has been approved by the examiner.

☐ 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 Draftperson.

☐ Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Any response 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)

- ☐ Notice of References Cited, PTO-892
- ☐ Information Disclosure Statement(s), PTO-1449, Paper No(s). ______
- ☐ Notice of Draftsperson's Patent Drawing Review, PTO-948
- ☐ Notice of Informal Patent Application, PTO-152
- ☐ Interview Summary, PTO-413
- ☐ Examiner's Amendment/Comment
- ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
- ☐ Examiner's Statement of Reasons for Allowance

ELLIS B. RAMIREZ
PRIMARY EXAMINER

PTOL-37 (Rev. 10/95) *U.S. GPO: 1997-417-381/62714

UNITED STATES DEPARTMENT OF COMMERCE
Patent and Trademark Office

NOTICE OF ALLOWANCE AND ISSUE FEE DUE

LM21/0330

LISA K JORGENSON
SGS THOMSON MICROELECTRONICS INC
1310 ELECTRONICS DRIVE
CARROLLTON TX 75006

APPLICATION NO.	FILING DATE	TOTAL CLAIMS	EXAMINER AND GROUP ART UNIT		DATE MAILED
08/702,911	08/26/96	032	RAMIREZ, E	2756	03/30/98

First Named Applicant: DIAZ, RAUL Z.

TITLE OF INVENTION: VIDEO AND/OR AUDIO DECOMPRESSION AND/OR COMPRESSION DEVICE THAT SHARES A MEMORY INTERFACE

ATTY'S DOCKET NO.	CLASS-SUBCLASS	BATCH NO.	APPLN. TYPE	SMALL ENTITY	FEE DUE	DATE DUE
2 96-S-11	395-200.770	C25	UTILITY	NO	$1320.00	06/30/98

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.

PATENT AND TRADEMARK OFFICE COPY

PTOL-85 (REV. 10-96) Approved for use through 06/30/99. (0651-0033)

*U.S. GPO: 1998-437-639/80023

UNITED STATES DEPARTMENT OF COMMERCE
Patent and Trademark Office

NOTICE OF ALLOWANCE AND ISSUE FEE DUE

LM21/0410

LISA K JORGENSON
SGS THOMSON MICROELECTRONICS INC
1310 ELECTRONICS DRIVE
CARROLLTON TX 75006

APPLICATION NO.	FILING DATE	TOTAL CLAIMS	EXAMINER AND GROUP ART UNIT		DATE MAILED
08/702,911	08/26/96	033	RAMIREZ, E	2756	03/30/98

First Named Applicant: DIAZ, RAUL Z.

TITLE OF INVENTION: VIDEO AND/OR AUDIO DECOMPRESSION AND/OR COMPRESSION DEVICE THAT SHARES A MEMORY INTERFACE

ATTY'S DOCKET NO.	CLASS-SUBCLASS	BATCH NO.	APPLN. TYPE	SMALL ENTITY	FEE DUE	DATE DUE
2 96-S-11	395-200.770	C25	UTILITY	NO	$1320.00	06/30/98

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.

PATENT AND TRADEMARK OFFICE COPY

PTOL-85 (REV. 10-96) Approved for use through 06/30/99. (0651-0033)

*U.S. GPO: 1998-437-639/80023

UNITED STATES DEPARTMENT OF COMMERCE
Patent and Trademark Office
Address: COMMISSIONER OF PATENTS AND TRADEMARKS
Washington, D.C. 20231

APPLICATION NUMBER	FILING DATE	FIRST NAMED APPLICANT	ATTORNEY DOCKET NO.
08/702,911	08/26/96	DIAZ	R 96-S-11

EXAMINER
RAMIREZ,E

LM21/0410

LISA K JORGENSON
SGS THOMSON MICROELECTRONICS INC
1310 ELECTRONICS DRIVE
CARROLLTON TX 75006

ART UNIT	PAPER NUMBER
2756	10

DATE MAILED: 04/10/98

This is a communication from the examiner in charge of your application.
COMMISSIONER OF PATENTS AND TRADEMARKS

Supplemental

NOTICE OF ALLOWABILITY (NOA)

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.

- [x] This communication is responsive to Supplemental NOA.
- [x] The allowed claim(s) is/are 14-41, and 44-48. (renumbered 1-33)
- [ ] The drawings filed on ______ are acceptable.
- [ ] 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).

A SHORTENED STATUTORY PERIOD FOR RESPONSE 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 obtained under the provisions of 37 CFR 1.136(a).

- [ ] Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL APPLICATION, PTO-152, which discloses that the oath or declaration is deficient. A SUBSTITUTE OATH OR DECLARATION IS REQUIRED.
- [ ] Applicant MUST submit NEW FORMAL DRAWINGS
  - [ ] because the originally filed drawings were declared by applicant to be informal.
  - [ ] including changes required by the Notice of Draftperson's Patent Drawing Review, PTO-948, attached hereto or to Paper No. ______.
  - [ ] including changes required by the proposed drawing correction filed on ______, which has been approved by the examiner.
  - [ ] 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 Draftperson.

- [ ] Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Any response 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)

- [ ] Notice of References Cited, PTO-892
- [ ] Information Disclosure Statement(s), PTO-1449, Paper No(s). ______
- [ ] Notice of Draftsperson's Patent Drawing Review, PTO-948
- [ ] Notice of Informal Patent Application, PTO-152
- [ ] Interview Summary, PTO-413
- [ ] Examiner's Amendment/Comment
- [ ] Examiner's Comment Regarding Requirement for Deposit of Biological Material
- [ ] Examiner's Statement of Reasons for Allowance

ELLIS B. RAMIREZ
PRIMARY EXAMINER

PTOL-37 (Rev. 10/95)
*U.S. GPO: 1997-417-381/62714

OIPE JUN 29 1998 JC52 PATENT & TRADEMARK OFFICE

N2 (1)

B #11 7/13/98

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the application of:

Applicant	: Raul Z. Diaz, et al	Docket No	: 96-S-011
Serial No	: 08/702,911	Group	: 2756
Filed	: August 26, 1996	Examiner	: E. Ramirez
For	: Video and/or Audio Decompression and/or Compression Device That Shares a Memory Interface	Batch No.	: C25

TRANSMITTAL OF FORMAL DRAWINGS

OFFICIAL DRAFTSMAN
Hon. Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

In response to the "Notice of Allowability" (POL 37) mailed March 30, 1998, in the above-referenced patent application, please find enclosed for filing five (5) sheet(s) of formal drawings.

I hereby authorize the Commissioner to charge any fees which may be required to Deposit Account No. 19-1353. A duplicate copy of this sheet is enclosed.

Respectfully submitted,

Lisa K. Jorgenson

Lisa K. Jorgenson
Reg. No. 34,845
Attorney for Applicant

SGS-Thomson Microelectronics, Inc.
1310 Electronics Drive/MS 2346
Carrollton, TX 75006
972-466-7414

CERTIFICATE OF MAILING
37 CFR 1.8(a)

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Official Draftsman, Assistant Commissioner for Patents, Washington, D.C. 20231 on the date below:

June 24, 1998 — Date

Angie Rodriguez — Signature

APPROVED BY DRAFTSMAN | O.G. FIG. | CLASS | SUBCLASS

5812789

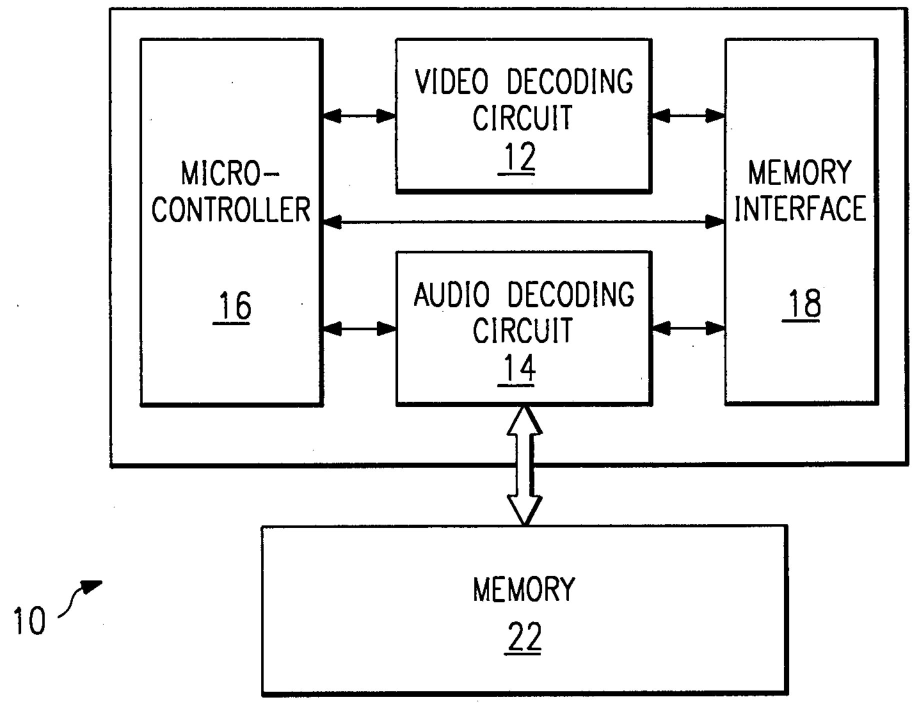


FIG. 1a
(PRIOR ART)

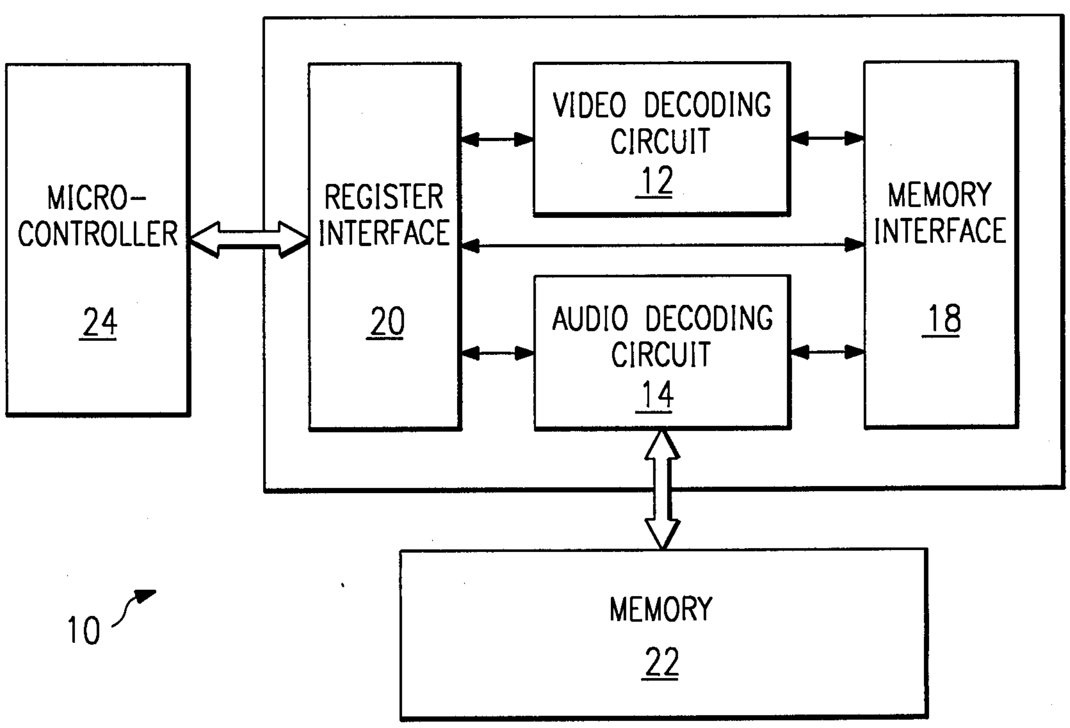


FIG. 1b
(PRIOR ART)

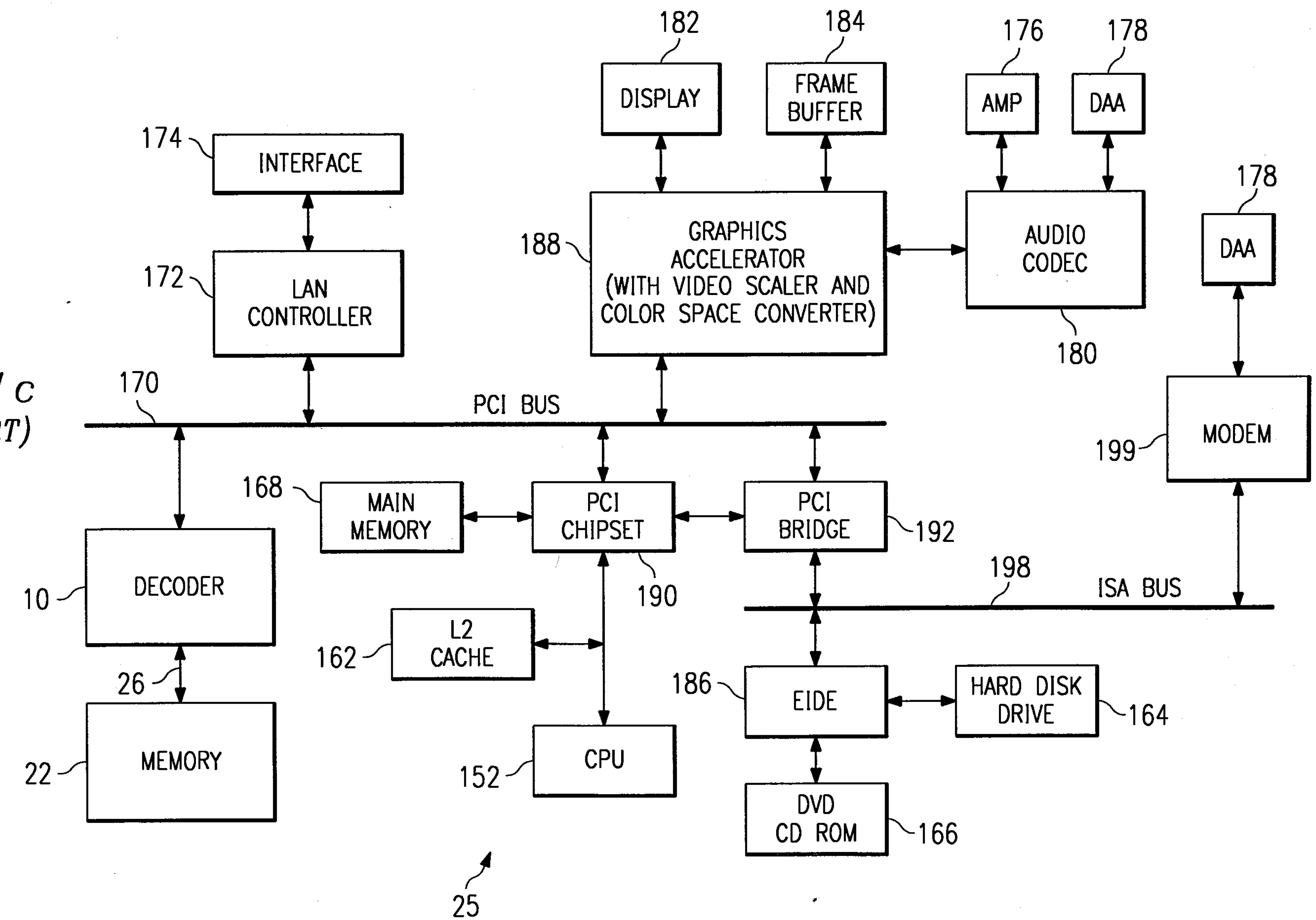


FIG. 1c
(PRIOR ART)

FIG. 2

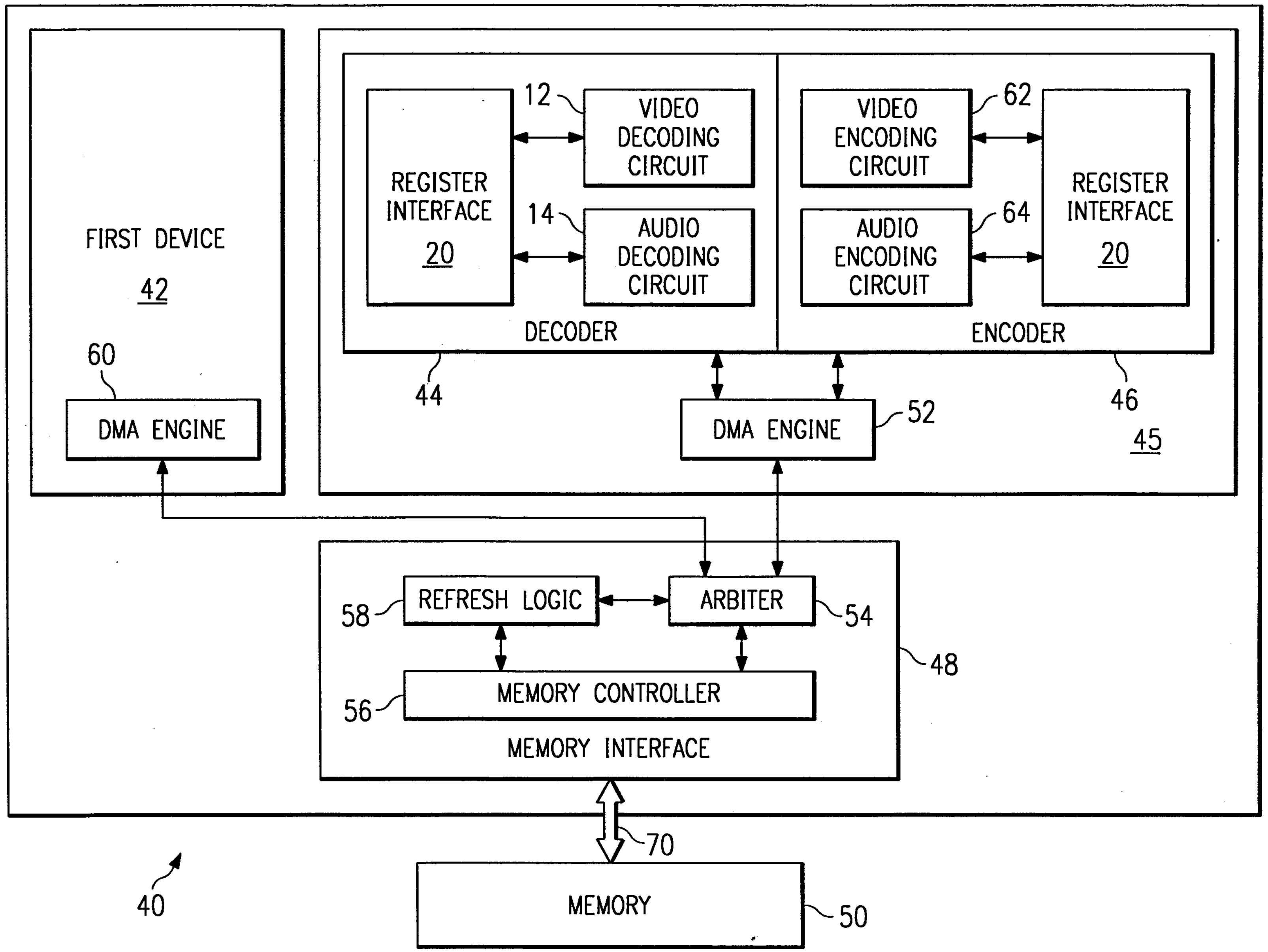
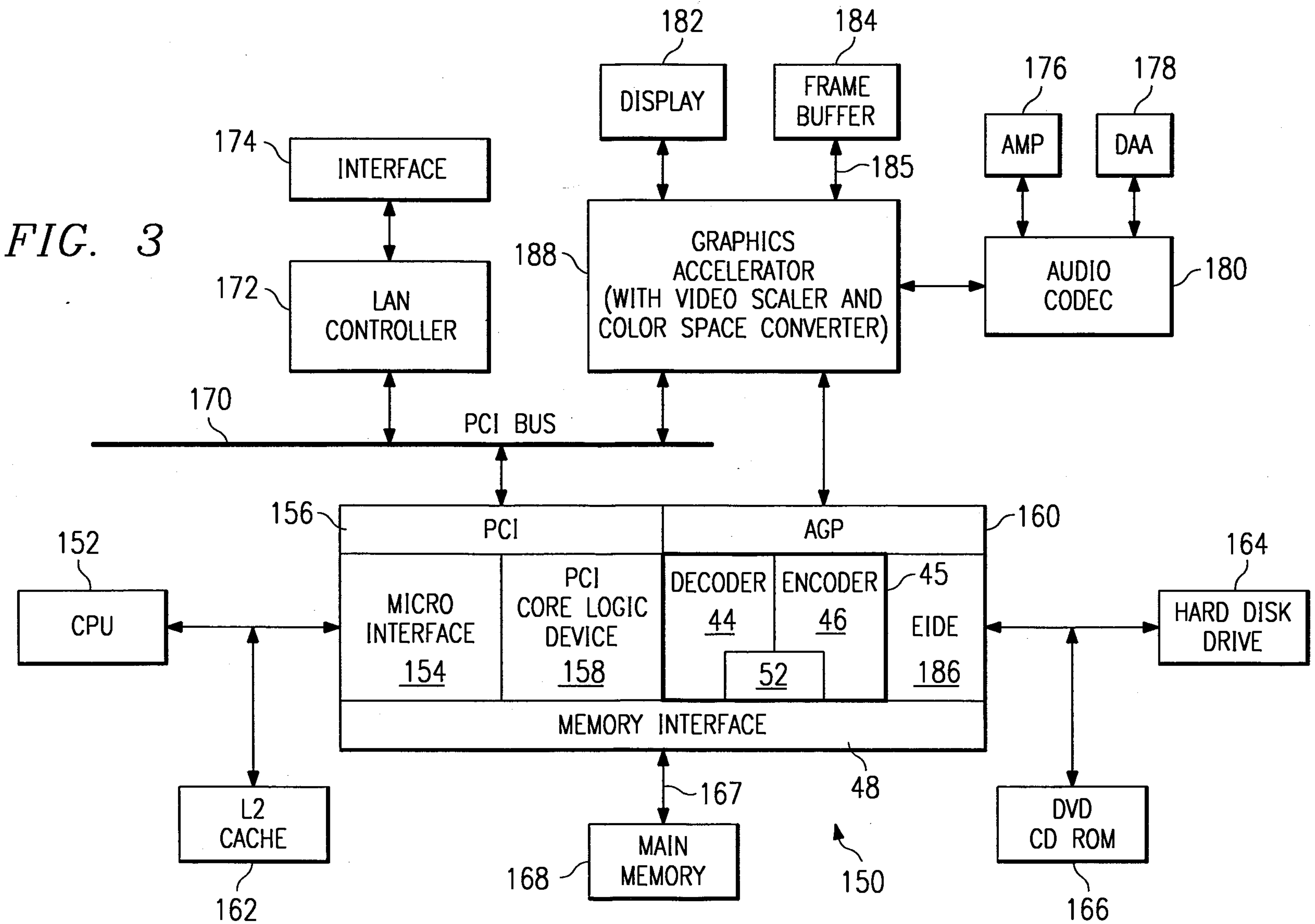

FIRST DEVICE
42
60
DMA ENGINE
REGISTER INTERFACE
20
12
VIDEO DECODING CIRCUIT
14
AUDIO DECODING CIRCUIT
DECODER
44
VIDEO ENCODING CIRCUIT
62
AUDIO ENCODING CIRCUIT
64
REGISTER INTERFACE
20
ENCODER
46
45
DMA ENGINE
52
REFRESH LOGIC
58
ARBITER
54
48
MEMORY CONTROLLER
56
MEMORY INTERFACE
70
MEMORY
50
40

FIG. 3


182
184
176
178
DISPLAY
FRAME BUFFER
AMP
DAA
174
INTERFACE
185
188
GRAPHICS ACCELERATOR (WITH VIDEO SCALER AND COLOR SPACE CONVERTER)
AUDIO CODEC
180
172
LAN CONTROLLER
170
PCI BUS
156
PCI
AGP
160
152
164
CPU
MICRO INTERFACE 154
PCI CORE LOGIC DEVICE 158
DECODER 44
ENCODER 46
45
EIDE 186
52
HARD DISK DRIVE
MEMORY INTERFACE
167
48
L2 CACHE
MAIN MEMORY
DVD CD ROM
168
150
162
166

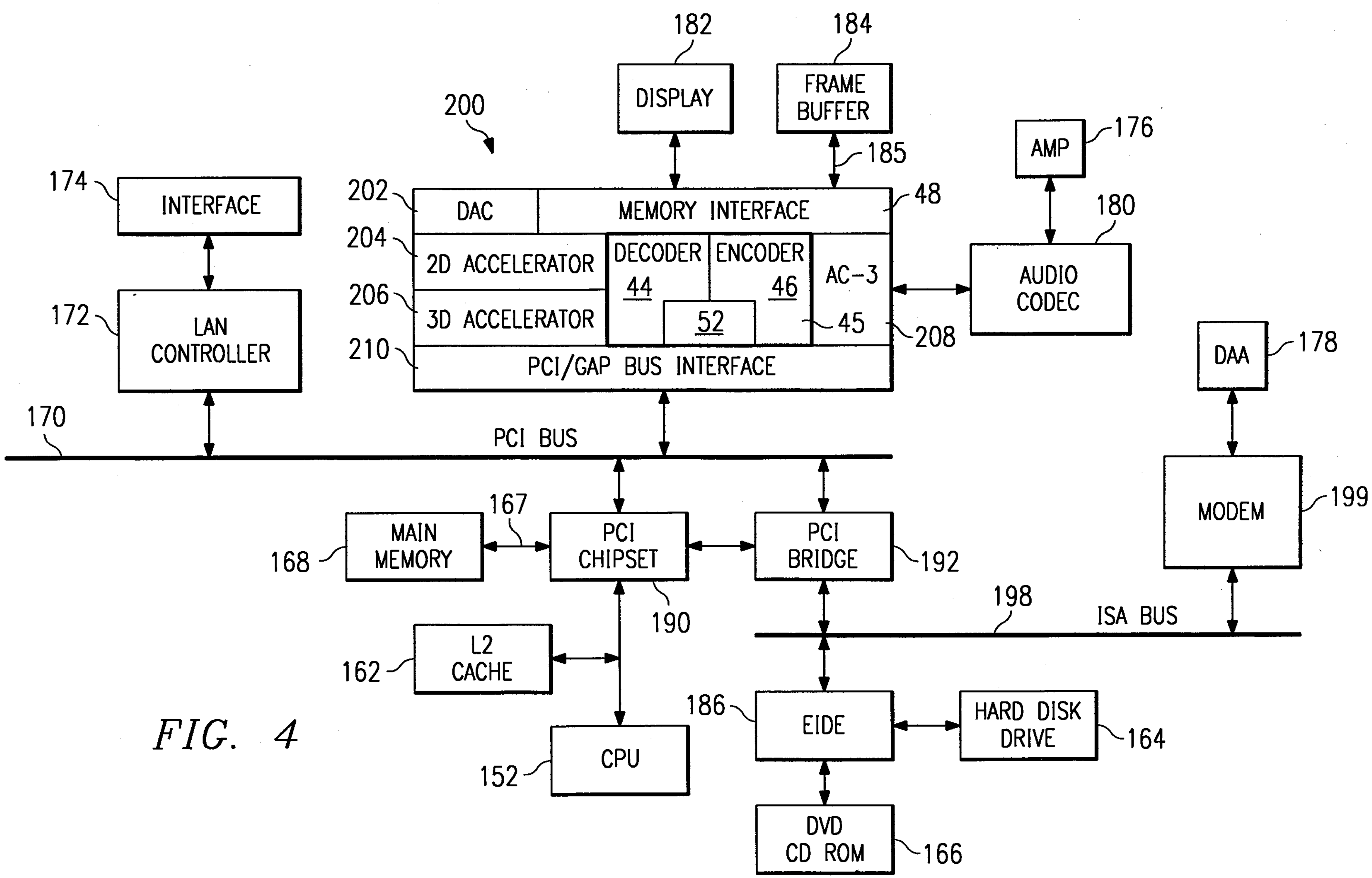

182
184
DISPLAY
FRAME BUFFER
200
185
AMP
176
174
INTERFACE
202
DAC
MEMORY INTERFACE
48
180
204
2D ACCELERATOR
DECODER 44
ENCODER 46
AC-3
AUDIO CODEC
172
LAN CONTROLLER
206
3D ACCELERATOR
52
45
208
DAA
178
210
PCI/GAP BUS INTERFACE
170
PCI BUS
MODEM
199
167
168
MAIN MEMORY
PCI CHIPSET
PCI BRIDGE
192
190
198
ISA BUS
162
L2 CACHE
186
EIDE
HARD DISK DRIVE
164
152
CPU
DVD CD ROM
166

FIG. 4

PTO UTILITY GRANT

Paper Number 12

The United States of America

The Commissioner of Patents and Trademarks

Has received an application for a patent for a new and useful invention. The title and description of the invention are enclosed. The requirements of law have been complied with, and it has been determined that a patent on the invention shall be granted under the law.

Therefore, this

United States Patent

Grants to the person(s) having title to this patent the right to exclude others from making, using, offering for sale, or selling the invention throughout the United States of America or importing the invention into the United States of America for the term set forth below, subject to the payment of maintenance fees as provided by law.

If this application was filed prior to June 8, 1995, the term of this patent is the longer of seventeen years from the date of grant of this patent or twenty years from the earliest effective U.S. filing date of the application, subject to any statutory extension.

If this application was filed on or after June 8, 1995, the term of this patent is twenty years from the U.S. filing date, subject to an statutory extension. If the application contains a specific reference to an earlier filed application or applications under 35 U.S.C. 120, 121 or 365(c), the term of the patent is twenty years from the date on which the earliest application was filed, subject to any statutory extension.

Bruce Lehman

Commissioner of Patents and Trademarks

Attest Mary H. Oliver

Form PTO-1584 (Rev. 2/97)

(RIGHT INSIDE)

FPI-LOM

#13

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the application of:

Applicant	: Raul Z. Diaz, et al	Docket No	: 96-S-011
Serial No	: 08/702,911	Group	: 2756
Filed	: August 26, 1996	Examiner	: E. Ramirez
For	: Video and/or Audio Decompression and/or Compression Device That Shares a Memory Interface	Batch No.	: C25

TRANSMITTAL LETTER

Hon. Assistant Commissioner for Patents
Washington, D.C.

Dear Sir:

Enclosed herewith is a Supplemental Declaration for filing in the above-identified application.

Please charge any fees necessary to deposit account No. 19-1353. A duplicate copy of this sheet is enclosed.

Respectfully submitted,

Lisa K. Jorgenson
Lisa K. Jorgenson
Reg. No. 34,845
Attorney for Applicant

SGS-Thomson Microelectronics, Inc.
1310 Electronics Drive/MS 2346
Carrollton, TX 75006
972-466-7414

CERTIFICATE OF MAILING
37 CFR 1.8(a)
I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231 on the date below:

June 24, 1998
Date

Angie Rodriguez
Signature

094 4/60 7.1.98

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

395/200.770

(13)

In re the application of:

Applicant	: Raul Z. Diaz, et al	Docket No	: 96-S-011
Serial No	: 08/702,911	Group	: 2756
Filed	: August 26, 1996	Examiner	: E. Ramirez
For	: Video and/or Audio Decompression and/or Compression Device That Shares a Memory Interface	Batch No.	: C25

RECEIVED Publishing Division JUN 29 1998

SUPPLEMENTAL DECLARATION

Hon. Assistant Commissioner for Patents
Washington, D.C.

Dear Sir:

I, Raul Z. Diaz, and Jefferson E. Owen, as below-named inventors in the application for letters patent for an improvement in Video and/or Audio Decompression and/or Compression Device That Shares a Memory Interface, Serial No. 08/702,911, filed in the United States Patent and Trademark Office on or about the 26th day of August, 1996, declare that my residence, post office address and citizenship are as stated below next to my name;

I believe I am the original, first and joint inventor of the subject matter which is claimed and for which a patent is sought;

I hereby state that I have reviewed and understand the contents of the above-identified patent application, including the claims;

that said subject matter, including the claims as amended, was part of my invention before the filing of the original application, above identified, for such invention; and that I acknowledge my duty to disclose information of which I am aware which is material to the patentability of this application in accordance with Title 37, Code of Federal Regulations, §1.56(a).

PAT No. 5812789
ISSUE: 9/22/98

I further declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Inventor's Signature: ______________________
Full Name of First Joint Inventor: Raul Z. Diaz
Date of Signature: June 10, 1998
Residence and Post Office Address:
750 Montrose Ave.
Palo Alto, CA 94303

Citizenship: United States of America

Inventor's Signature: ______________________
Full Name of Second Joint Inventor: Jefferson E. Owen
Date of Signature: 6/19/98
Residence and Post Office Address:
44177 Bowers Court
Fremont, CA 94539

Citizenship: United States of America

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the application of:

Applicant	: Raul Z. Diaz, et al	Docket No	: 96-S-011
Serial No	: 08/702,911	Group	: 2756
Filed	: August 26, 1996	Examiner	: E. Ramirez
For	: Video and/or Audio Decompression and/or Compression Device That Shares a Memory Interface	Batch No.	: C25

TRANSMITTAL LETTER

BOX: ISSUE FEE
Hon. Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

Enclosed:
(1) Issue Fee Transmittal PTOL-85B with Certificate of Mailing:
(2) Check in the amount of $1,350.00 for payment of Issue Fee and advanced order of ten (10) copies;
(3) Our return postcard.

The Commissioner is hereby authorized to charge any additional fees which may be required, or credit any overpayment to Deposit Account No. 19-1353. A duplicate copy of this sheet is enclosed.

Respectfully submitted,

Lisa K. Jorgenson
Lisa K. Jorgenson
Reg. No. 34,845
Attorney for Applicant

SGS-Thomson Microelectronics, Inc.
1310 Electronics Drive/MS 2346
Carrollton, TX 75006
972-466-7414

CERTIFICATE OF MAILING
37 CFR 1.8(a)
I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Box Issue Fee, Assistant Commissioner for Patents, Washington, D.C. 20231, on the date below:

June 24, 1998 — Date

Angie Rodriguez — Signature

PART B—ISSUE FEE TRANSMITTAL

142-1320.00
561-30.00
8.

Complete and mail this form, together with ...cable fees, to: **Box ISSUE FEE**
Assistant Commissioner for Patents
Washington, D.C. 20231

4160

MAILING INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE. Blocks 1 through 4 should be completed where appropriate. All further correspondence including the Issue Fee Receipt, 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)

LM21/0410

LISA K JORGENSON RECEIVED
SGS THOMSON MICROELECTRONICS INC
1310 ELECTRONICS DRIVE
CARROLLTON TX 75006 JUL 02 1998

Note: The certificate of mailing below can only be used for domestic mailings of the Issue Fee 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 Issue Fee 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.

Angie Rodriguez (Depositor's name)

(Signature)

(Date)

APPLICATION NO.	FILING DATE	TOTAL CLAIMS	EXAMINER AND GROUP ART UNIT		DATE MAILED
08/702,911	08/26/96	033	RAMIREZ, E	2756	03/30/98

First Named Applicant: DIAZ, RAUL Z.

TITLE OF INVENTION: VIDEO AND/OR AUDIO DECOMPRESSION AND/OR COMPRESSION DEVICE THAT SHARES A MEMORY INTERFACE

ATTY'S DOCKET NO.	CLASS-SUBCLASS	BATCH NO.	APPLN. TYPE	SMALL ENTITY	FEE DUE	DATE DUE
2 96-S-11	395-200.770	C25	UTILITY	NO	$1320.00	06/30/98

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 David V. Carlson
2 Theodore E. Galanthay
3 Lisa K. Jorgenson

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 appropiate when an assignment has been previously submitted to the PTO or is being submitted under separate cover. Completion of this form is NOT a subsititue for filing an assignment.

(A) NAME OF ASSIGNEE STMicroelectronics, Inc.

(B) RESIDENCE: (CITY & STATE OR COUNTRY)

Carrollton, Texas

Please check the appropriate assignee category indicated below (will not be printed on the patent)

☐ individual ☒ corporation or other private group entity ☐ government

4a. The following fees are enclosed (make check payable to Commissioner of Patents and Trademarks):

☒ Issue Fee
☒ Advance Order - # of Copies 10

4b. The following fees or deficiency in these fees should be charged to:

DEPOSIT ACCOUNT NUMBER 19-1353
(ENCLOSE AN EXTRA COPY OF THIS FORM)

☐ Issue Fee
☐ Advance Order - # of Copies ______

The COMMISSIONER OF PATENTS AND TRADEMARKS IS requested to apply the Issue Fee to the application identified above.

(Authorized Signature) Lisa K. Jorgenson (Date) 6/9/98

NOTE; The Issue Fee 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 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, 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

PTOL-85B (REV.10-96) Approved for use through 06/30/99. OMB 0651-0033 **Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE**

PATENT APPLICATION FEE DETERMINATION RECORD Effective October 1, 1995	Application or Docket Number

CLAIMS AS FILED - PART I

FOR	NUMBER FILED (Column 1)		NUMBER EXTRA (Column 2)	SMALL ENTITY RATE	SMALL ENTITY FEE	OR	OTHER THAN SMALL ENTITY RATE	OTHER THAN SMALL ENTITY FEE
BASIC FEE					375.00	OR		750.00
TOTAL CLAIMS	49	minus 20 =	* 29	x$11=		OR	x$22=	638
INDEPENDENT CLAIMS	5	minus 3 =	* 2	x39=		OR	x78=	152
MULTIPLE DEPENDENT CLAIM PRESENT				+125=		OR	+250=	
				TOTAL		OR	TOTAL	1544

* If the difference in column 1 is less than zero, enter "0" in column 2

CLAIMS AS AMENDED - PART II

AMENDMENT A

	CLAIMS REMAINING AFTER AMENDMENT (Column 1)		HIGHEST NUMBER PREVIOUSLY PAID FOR (Column 2)	PRESENT EXTRA (Column 3)	SMALL ENTITY RATE	ADDITIONAL FEE	OR	OTHER THAN SMALL ENTITY RATE	ADDITIONAL FEE
Total	* 33	Minus	** 49	= —	x$11=		OR	x$22=	
Independent	* 3	Minus	*** 5	= —	x39=		OR	x78=	
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM					+125=		OR	+250=	
					TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT B

	CLAIMS REMAINING AFTER AMENDMENT (Column 1)		HIGHEST NUMBER PREVIOUSLY PAID FOR (Column 2)	PRESENT EXTRA (Column 3)	SMALL ENTITY RATE	ADDITIONAL FEE	OR	OTHER THAN SMALL ENTITY RATE	ADDITIONAL FEE
Total	*	Minus	**	=	x$11=		OR	x$22=	
Independent	*	Minus	***	=	x39=		OR	x78=	
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM					+125=		OR	+250=	
					TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT C

	CLAIMS REMAINING AFTER AMENDMENT (Column 1)		HIGHEST NUMBER PREVIOUSLY PAID FOR (Column 2)	PRESENT EXTRA (Column 3)	SMALL ENTITY RATE	ADDITIONAL FEE	OR	OTHER THAN SMALL ENTITY RATE	ADDITIONAL FEE
Total	*	Minus	**	=	x$11=		OR	x$22=	
Independent	*	Minus	***	=	x39=		OR	x78=	
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM					+125=		OR	+250=	
					TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20."
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3."
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

FORM PTO-875
(Rev. 10/95)

Patent and Trademark Office, U.S. DEPARTMENT OF COMMERCE

Form PTO 1130
(REV 2/94)

U.S. DEPARTMENT OF COMMERCE
Patent and Trademark Office

PACE DATA ENTRY CODING SHEET

1ST EXAMINER	270	DATE	10-15
2ND EXAMINER		DATE	

APPLICATION NUMBER	TYPE APPL	FILING DATE MONTH	DAY	YEAR	SPECIAL HANDLING	GROUP ART UNIT	CLASS	SHEETS OF DRAWING
08/702911	1	08	26	96	0	2612	358	5

TOTAL CLAIMS	INDEPENDENT CLAIMS	SMALL ENTITY?	FILING FEE	FOREIGN LICENSE	ATTORNEY DOCKET NUMBER
49	5	0	1356	Y	96-S-11

CONTINUITY DATA

CONT CODE	STATUS CODE	PARENT APPLICATION SERIAL NUMBER	PCT APPLICATION SERIAL NUMBER	PARENT PATENT NUMBER	PARENT FILING DATE MONTH	DAY	YEAR
			PCT / /				
			PCT / /				
			PCT / /				
			PCT / /				
			PCT / /				

PCT/FOREIGN APPLICATION DATA

FOREIGN PRIORITY CLAIMED	COUNTRY CODE	PCT/FOREIGN APPLICATION SERIAL NUMBER	FOREIGN FILING DATE MONTH	DAY	YEAR
N					

```
=> d his

     (FILE 'USPAT' ENTERED AT 13:35:42 ON 22 SEP 1999)
L1         1120 S MULTIPROCESSOR#/TI,AB
L2         1660 S (MAIN MEMORY)/TI,AB
L3          107 S L1 AND L2
L4        12245 S FIFO
L5         5881 S MEMORY(5A)MANAGEMENT
L6         1048 S L5 AND L4
L7            9 S L6 AND L3
L8           30 S L3 AND L4
L9         5786 S MEMORY CONTROLLER
L10          12 S L9 AND L8
L11           3 S L10 AND L5
```

```
=> d his

     (FILE 'USPAT' ENTERED AT 11:38:46 ON 28 SEP 1999)
L1             51 S (FIFO OR QUEUE)(3A)BYPASSING
L2         141355 S PROCESSOR#
L3          13932 S MAIN MEMORY
L4           5827 S MEMORY CONTROLLER
L5           1670 S L4 AND L3 AND L2
L6              2 S L5 AND L1
L7            189 S (FIFO OR QUEUE OR BUFFER)(3A)BYPASSING
L8             19 S L7 AND L2 AND L3 AND L4
L9         104716 S DECOD####
L10            14 S L9 AND L8
L11            14 S L10 AND (IDENTIF? OR TAG#)
L12            12 S L11 AND EMPTY
L13            12 S L12 AND (FULL)
L14            11 S L11 AND ((BUFFER OR QUEUE OR FIFO)(P)(EMPTY))
L15            11 S L14 AND ((BUFFER OR QUEUE OR FIFO)(P)(FULL))
L16            41 S L2(P)L7
L17            14 S L16 AND L3 AND L4
L18          8334 S (BUFFER OR QUEUE OR FIFO)(P)EMPTY
L19         14117 S (BUFFER OR QUEUE OR FIFO)(P)FULL
L20            10 S L19 AND L18 AND L17
L21            10 S L20 AND L9
```

```
=> d his

     (FILE 'USPAT' ENTERED AT 11:38:46 ON 28 SEP 1999)
L1            51 S (FIFO OR QUEUE)(3A)BYPASSING
L2        141355 S PROCESSOR#
L3         13932 S MAIN MEMORY
L4          5827 S MEMORY CONTROLLER
L5          1670 S L4 AND L3 AND L2
L6             2 S L5 AND L1

=> d l6 1- ti,ccls


US PAT NO:      5,796,413 [IMAGE AVAILABLE]             L6: 1 of 2
TITLE:          Graphics controller utilizing video memory to provide
                  macro command capability and enhanced command buffering
US-CL-CURRENT: 345/522, 516, 521

US PAT NO:      5,530,933 [IMAGE AVAILABLE]             L6: 2 of 2
TITLE:          Multiprocessor system for maintaining cache coherency by
                  checking the coherency in the order of the transactions
                  being issued on the bus
US-CL-CURRENT: 711/141; 364/228.3, 229.2, 238.4, 240.1, 243.4, 243.44,
                  264, 264.4, 264.7, DIG.1; 711/3, 119, 121
```

PROCEEDINGS

SPIE—The International Society for Optical Engineering

Digital Video Compression on Personal Computers: Algorithms and Technologies

Arturo A. Rodriguez
Chair/Editor

7–8 February 1994
San Jose, California

Sponsored by

IS&T—The Society for Imaging Science and Technology

SPIE—The International Society for Optical Engineering

In cooperation with

IEEE Computer Society

Published by
SPIE—The International Society for Optical Engineering

SPIE PROCEEDINGS SERIES

Volume 2187

SPIE (The Society of Photo-Optical Instrumentation Engineers) is a nonprofit society dedicated to the advancement of optical and optoelectronic applied science and technology.

A high-performance cross-platform MPEG decoder

Hemant Bheda and Partha Srinivasan

Mediamatics Inc.
Santa Clara, CA 95054

pp. 241-8 = 8
PD = '94

ABSTRACT

We present a high performance implementation of a MPEG decoder, written entirely in a high level language. The decoder implementation fully complies with the MPEG-I standard and decodes all (I, P, B) frame types in MPEG video bitsteams and is portable.

Versions of this decoder are implemented on Windows 3.1, and on Windows NT (X86, MIPS, ALPHA). A comparison of the performance of the decoder between the various platforms is made. We also present a high quality, fast dithering and interpolation algorithm used to convert YCbCr directly into 8 bit palletized images.

We propose a new method called Collaborative Compression, of dealing with compression and decompression tasks at a very low cost to achieve 30 fps SIF performance for desktop applications. Collaborative Compression is a systems approach to partitioning the functionality between CPU-centric (i.e. software) and hardware-assist (VLSI) in order to achieve the optimal cost solution. The CPU provides glue programmability to tie the accelerated and non-accelerated parts of the algorithm together. The advent of high bandwidth, low latency busses (VL Bus and PCI) enable a high speed data pathway between the distributed computational elements.

1. INTRODUCTION

The Motion Picture Experts Group (MPEG) video coding algorithm is a standard for the storage of video and audio data in a compressed digital form. It was defined by the CCITT MPEG group formed with a goal of defining a standard that could compress video and audio to a bandwidth of 1.5Mbit/sec or less. The committee draft dated December 1991[3] has formed the basis for the implementation described in this paper.

The MPEG Decoder we have implemented decodes only the video bitstream. Its primary goal was to explore the possibility of decoding MPEG bitstreams on Personal Computer class platforms. Patel et. al[1] have already presented a software decoder designed primarily for portability across Unix platforms with X Windows. Our software player was designed for portability across PC platforms, with RISC or CISC CPUs and running 16bit or 32bit operating systems, while delivering the highest performance and be fully compliant with the ISO standard. The GUI for the decoder uses the Windows 3.1 API, which makes it work with OS2, Windows NT and Windows 3.1. We have documented the performance of this decoder on Windows 3.1, Windows NT running on X86, MIPS and ALPHA platforms.

Even though the MPEG standard defines one resolution - SIF at 320x240, for the PC environment a defacto standard commonly called QSIF has emerge - at 160x120. Typical PCs do not have enough CPU power to do 30 (or even 15) Frames Per Second (Fps) SIF resolution. So a common practice is to encode a 160x120 (QSIF) resolution image, and then after decoding, to interpolate the output image by a factor of 2 to obtain a 320x240 SIF resolution image. We have also developed a non-real-time MPEG encoder, which is capable of producing bitstreams containing I, P and B frames, with arbitrary M and N parameters. This encoder can also produce video bitstreams with any resolution. Our decoder can also decode any resolution image, and has a user selectable output interpolation factor (1x or 2x).

We have developed a novel method of doing the Color Space Conversion that is especially suited to PC systems using the Windows OS. We present the computational loads presented by the different phases of the MPEG decoding algorithm. Section 3.1 deals with the architecture of the software decoder that allows the core decoder to be ported to any platform that has a 32 bit ANSI-C compiler, and the GUI to be ported to any platform supporting the Windows 3.X API.

Section 3.1 presents the dithering and Color Space Conversion algorithms used in this decoder. Section 4.0 presents the performance results obtained on the various platforms.

2. THE MPEG-1 VIDEO CODING STANDARD

In this paper we refer only to the video bitstream when referring to MPEG-I. The MPEG video coding standard specifies the format of the video bitstream. The standard also specifies the decoding process to be used and the errors tolerable in the process. The process of encoding the bitstream is not specified and is left as an area of differentiation for implementers. We shall attempt to present the MPEG-I standard more from a decoding computational load point of view. Details of the actual standard can be found in the standards document[3].

MPEG uses transform coding in conjunction with motion compensation to achieve its compression goals. It performs spatial compression using transforms (specifically the Discrete Cosine Transform, DCT) and temporal compression using motion compensation. MPEG encoded video frames are basically of three types: (i) Intra Frames (I), (ii) Predictive Frames (P) and (iii) Bi-Directional Frames (B). Each frame type is encoded in a different manner. Each frame is first subdivided in square blocks of 16x16 pixels called macroblocks. Each macroblock of RGB pixels is converted to a 16x16 *Y* block and a 8x8 *Cb* block and an 8x8 *Cr* block representing the Luminance (Y) and Chrominance (Cb, Cr) information respectively.

Intra Frames are also called key Frames. They are coded only spatially and decoding them does not require data from any other frame, past or present. P Frames are coded temporally and spatially. Thus they depend on data from the previous I|P frame in the past. B Frames are dependent on the nearest I|P Frames in the past and the future. Each frame type represents a different type of computational load to a typical PC class machine.

The input video data can be assumed as being represented as a two dimensional array of triplets (each triplet is one pixel with the Red, Green and Blue values). It is also possible to have input data in the form of Luminance and Chrominance arrays. Figure. 1. shows the different stages of the video encoding process for Intra Frames. Each macroblock is transform coded using DCT. The coefficient terms are then quantized and run-length coded. The resulting bitstream is then entropy coded using Huffman codes. This is repeated for every block in the image.

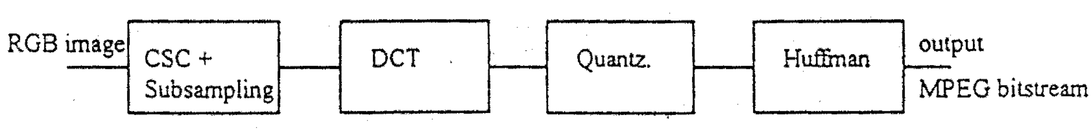


Figure 1.0 Encoding process for Intra Frames

For P and B frames, any macroblock that can be represented from its reference key frame is represented as a motion vector - a tuple representing the relative displacement from its position in the reference frame. In the case of P frames, there is only one reference frame - the nearest I|P frame in the past. For B frames there are two reference frames - the nearest past and future I|P frames. For B frames the current macroblock can be a simple linear function of macroblock from both reference frames. Any block that cannot be so represented is called an Error term, and is coded like a macroblock from an I frame.

2.1 Computational Loads in MPEG

From a computational point of view, each type of frame differs in it computational load. We shall differentiate computational loads as:
(i) arithmetic operations: Those that are predominantly comprised of arithmetic operations (+, -, *) and whose memory access pattern fits in a relatively small cache.
(ii) memory operations: Those whose memory bandwidth utilization is heavy in relation to their arithmetic operation count.

(iii) Bit manipulations: Comprised of manipulating bit fields, extracting, rotating, shifting and masking variable length bit fields. Some processors have special instructions for dealing with bit field operations. Others perform bit field operations by decomposing them into standard arithmetic and logical operations.

The major operations performed in Decoding are:

(1) Variable Length Decoding
(2) Inverse quantization and Zig-Zag scanning
(3) IDCT
(4) Motion Compensation (for P and B frames)
(5) Color Space Conversion (from YCbCr to RGB color space) OPTIONAL
(6) Dithering (reducing from 24 bit RGB color space to 8 bit color space). OPTIONAL

The kind of load presented by each major phase is dependent on the implementation. This in turn allows tradeoffs to be made in order to tune the decoder for different processor architectures.

2.2 Architectural features and their impact on MPEG decoding

As will be seen in the following sections, different architectural features impact the performance of a decoding algorithm implementation. We will attempt to catalog some of the performance axes that influence decoder performance. They are:

1) CPU performance
2) Cache and Memory subsystem performance
3) Bus performance
4) Display subsystem performance

CPU performance affects all compute bound operations such as IDCT, or bit field operations. However if the data required cannot be contained in registers - as is always the case, whether doing IDCT or any other operation, the cache subsystem performance becomes crucial. Operations such as VL decoding, Inverse quantization and IDCT may not fit in the first level cache of the CPU. Second level cache size and performance then become crucial. The motion compensation (frame reconstruction) part is extremely memory dependent. The CPU just constructs the output frame from two input frames, using mostly copying. Given the size of the frames, and the random nature in which the blocks can move, large second level caches are necessary to contain the input and output frames. Once the output frames is ready, it has to be transferred to the display frame buffer. This operation is dependent on the available bandwidth on the bus connecting the display adapter to the CPU. Sometimes, even when adequate bus bandwidth is available, the display adapter itself may not allow efficient access to the display memory from the host bus (e.g. the display memory may be organized to provide higher priority to the graphics accelerator rather than the host CPU bus). We will discuss in Section 3.0, how these architectural features affect our decoder, and what features are best suited to software MPEG decoders.

2.3 Other Software MPEG Decoders

There are a few other software only MPEG compliant video decoders, both commercial and shareware/freeware. Ketan Patel et. al at UC Berkeley have released an excellent portable implementation for Unix platforms into the public domain. There is also a shareware Windows NT based MPEG decoder, which can decode video bitstreams. This is slower than our implementation and also has poor video quality.

Our implementation can decode bitstreams containing any frame type and has been tested with several bitstreams, both from the Internet and from those provided by other collaborators. We also produced bitstreams using our own non-real-time software encoder. This encoder allowed us to play with performance sensitivity to quantization levels, frame type sequencing and so forth.

3. THE SOFTWARE MPEG-I VIDEO DECODER

The primary goal of the decoder was to determine if a full implementation of the decoder would achieve reasonable performance (as measured by frame rate) on a PC class machine. We define this to be a 486Dx2-66 PC, with a 256Kbyte second level cache, 8 or 16 Mbytes of main memory - 70 nS. We also wanted a decoder written in a high level language, for ease of maintenance and portability.

The decoder is partitioned into 2 distinct parts. A GUI based front end, that provides a user friendly player, allowing play, stop, rewind and variable speed fast play functionality in both forward and reverse directions and handles all the display rendering. This GUI front end uses the Windows 3.X API and is portable across all operating systems that provide Windows API servers (such as WABI, OS2, Windows NT). The second part (backend) of the decoder is implemented as a Dynamic Linked Library (DLL).

Windows provides a format called the Device Independent Bitmap (DIB) for representing color bitmaps. It also provides API calls for drawing such DIBs. These routines first convert the DIB from the color system provided to the native color format (eg. 16 bit or 8 bit) and then renders the bitmap.

The percentage of total time spent in the major operations when decoding to a 1x interpolated 8 bit Device Dependent Bitmap (DDB) output format is given. The data was collected on a DEC Alpha system, DECpc AXP 150 running WinNT 3.1 - release build. The compiler used was the release SDK 3.1 compiler for the Alpha architecture.

Major Operation Type	8 bit dithered
VL Decoding	10.5 %
IDCT	16.5%
CSC + Dithering	26.5%
Frame Reconstruction	26.0%
Frame Display	13.5%
Misc.	7%

Table 1. Breakup of time spent in Decoding a video bitstream

The kinds of loads presented by our decoder implementation are as follows:

1) VL Decoding - bit manipulation:
Converted to logical operations, shifting, masking and rotating. This is CPU intensive and register intensive (given the 8 registers on the x86).

2) Inverse quantization and unzigzag:
This stage is performed using table lookups - and is there cache performance sensitive in our implementation.

3) IDCT:
We believe this is sensitive to register pressure in machines with few registers. On machines with hardware single cycle multipliers, this is CPU bound.

4) Color Space Conversion:
We implement this using table lookup - both in the true color (16 bit or 24 bit case) and in the 8 bit dithered output case. This phase is cache and memory latency sensitive. Even though the lookup tables may fit in the primary cache, the reading and writing of the image frame pollutes the cache. Primary write through caches without write buffers suffer in this regard when compared to write through with write buffers. Large primary caches such as those on the MIPS Magnum systems definitely help the performance.

On the whole, our implementation showed a large sensitivity to memory/cache subsystem performance, and to display subsystem speed. For our player the memory to screen bitblt speed is the primary governing factor in display subsystem performance. For this reason - a fast frame buffer access path is better than a very powerful graphics accelerator, coupled with slow host access speeds. This is demonstrated in Section 4.1.

3.1 Dithering

Dithering was a major area where there were large swings in performance depending on the dithering algorithm used. Typical algorithms used for dithering are error diffusion algorithms (Floyd-Steinberg) and ordered dither[2]. We have developed our own dithering algorithm that is similar to the one used by Patel et. al. Windows 3.X and WinNT workstations used for multimedia typically have a 256 color display accelerator card. Out of the 256 colors available, Windows 3.X[4] uses 20 color which occur at pre-defined indices in the color LUT. The first 10 and last 10 indices are used by Windows, ie. colors 0-9 and 246-255. In order to achieve the greatest performance under Windows, the displayed bitmap should used what is called an "Identity Palette". This palette has the 20 fixed Windows colors in their nominal positions i.e 0-9 and 246-255. Only 236 colors at indices 10-245 are available for an application to use.

Typically dithering converts 24 bit RGB images to 8 bit palletized images. In MPEG, the output of the decompression process typically results in a YCbCr image, which is then color converted into an RGB image. This RGB image is then displayed. If the display system cannot accept full color RGB images, the image is then dithered (color sub sampled) to allow for display at lower color resolutions.

Given the additional constraints imposed by Windows 3.X, we finally developed an algorithm that converts directly from YCbCr to a fixed palette with 228 entries in it. This has the advantage of avoiding the intermediate conversion from YCbCr to RGB. This is similar to the algorithm given by Patel, except that we use 228 colors instead of 128. We have also folded the 2X output interpolation into this stage, when we do the look up.

4. PERFORMANCE RESULTS

We present the performance of our decoder as measured in Frames Per second on the different platforms in Table 2. We did not have access to a Pentium based NT platform at the time this was written. The bitstream used had I, P and B frames and a resolution of 320x240 and an average bitrate of 1.0 MBits/sec at 30 frames per second. The output format was 1X interpolated 8 bit dithered output. We also give the performance on QSIF sequence - encoded at 160x120 resolution and displayed with 1X and 2X output interpolation,

CPU-Mhz,Primary Cache, Secondary	OS	320x240	160x120 / 2X	160x120 / 1X
486DX2-66, 8Kb, 256Kb	Win 3.1	5.0	14	26
486DX2-66, 8Kb, 256Kb	Win NT	4.8	13	24
Mips R4400-100, 32Kb, 512Kb	Win NT	10.3	26	42
Alpha AXP-150, 16Kb, 512Kb	Win NT	12.5	29	49

Table 2. Performance in Frames Per Second.

Note: The Mips machine had a direct linear frame buffer, providing the highest display subsystem performance of all the machines. The Intel machines had VESA local bus display adapters. The Alpha machine had an EISA bus based graphics adapter. The source code is the same for all the systems.

We find that none of the machines can do real-time decoding of a SIF frame. However by using bitstreams encoded at QSIF resolution and then by performing a 2X interpolation of the output image, we can perform 30 fps on the RISC platforms. We have not exhausted all the performance gains to be made by playing with aggressive compiler optimization switches. The code is written in 32 bit C and does not utilize the 64 bit data paths available on the MIPS and ALPHA platforms.

4.1 Performance Sensitivity to Memory and Display subsystems

We conducted a side by side performance test of two MIPS based NT machines with identical MIPS R4400 100 Mhz (50 Mhz bus) processors. Both machines had similar Primary caches (on board the R4400). Machine A had a slow 512kb secondary cache and a slow memory controller - identical to those used on 486 Dx2-66 based machines. Machine B had a fast secondary cache of 512Kb and a very fast memory controller. Machine A had an ISA bus based 8 bit graphics card and Machine B had a direct linear frame buffer which the host CPU could access at high speeds. For the same sequence. Machine A had a performance of 6.1 Fps and Machine B had a performance of 10.1 Fps. This was a combination of both the memory subsystem and the display subsystem effects. The gain (Machine B/ Machine A) was 1.65 for this case. We disabled the display of frames to find the effect of memory subsystem effects alone. The speed up dropped to 1.36, still a substantial difference considering the processors are identical and running identical code.

5.0 COLLABORATIVE COMPRESSION

We define collaborative compression to be the optimal balance of software decoding with hardware assist to enable real-time decompression at minimal cost. There are a number of hardware devices on the market that are designed to perform 30 frames per second SIF decoding, such as those from SGS Thompson, C-Cube, IIT etc.

Some of these devices are programmable (IIT) while others are hardwired. Even though these devices themselves are low cost (~ $30), the end cost of a system solution is much higher (~$150). This is due to the extra glue logic required to interface to a PC bus (VL, PCI or ISA), the external memory, DACs etc. On the other hand, we have presented a software only decoder whose performance clearly does not meet real-time requirements. However, it is possible to add a minimum amount of hardware, which when properly designed will together with the existing CPU and display subsystem, will enable real-time full SIF decoding.

5.1 Color Space Conversion and Stretching

The plethora of compression standards in the Windows arena (such as MS Video 1, Cinepak, Motion JPEG) has already led to the emergence of rudimentary forms of hardware assist such as Color Space Converters (CSC) and stretchers. Examples of such devices are made by Weitek (VideoPower) and Videologic, Tseng Labs. All these devices accelerate YCbCr -> RGB conversion, dithering and also stretching bitmaps from QSIF or SIF to full screen proportions. This is a logical first step, because most of the compression algorithms in vogue today can use these functionality.

5.2 IDCT and Huffman Decoding

The next step would be to accelerated compute intensive portions of the algorithm. The next largest gain will be made by performing the IDCT using hardware assist.

It is best to leave random control operations and other logic to the hands of software running on the host CPU. This enables rapid development of the decompressor, reduces time to market. Also, generic blocks such as IDCT, Huffman, can be re-used by other compression algorithms, by merely changing the glue software which ties the various hardware modules together. The requirement of any hardware module designed for collaborative compression are as follows:

(1) They support concurrency. The CPU must be free to perform other tasks after priming the hardware module. This allows us to pipeline the operations (in software).
(2) Data flow must be one-way. If there is too much data movement required of the CPU, for example to move from one module to the other - e.g. from IDCT to CSC, the speedup gained from the hardware module will be lost in the extra work done in shifting data around. The best model to view this is as several functional units - all sharing the same memory as shown in Figure 2.0 below.

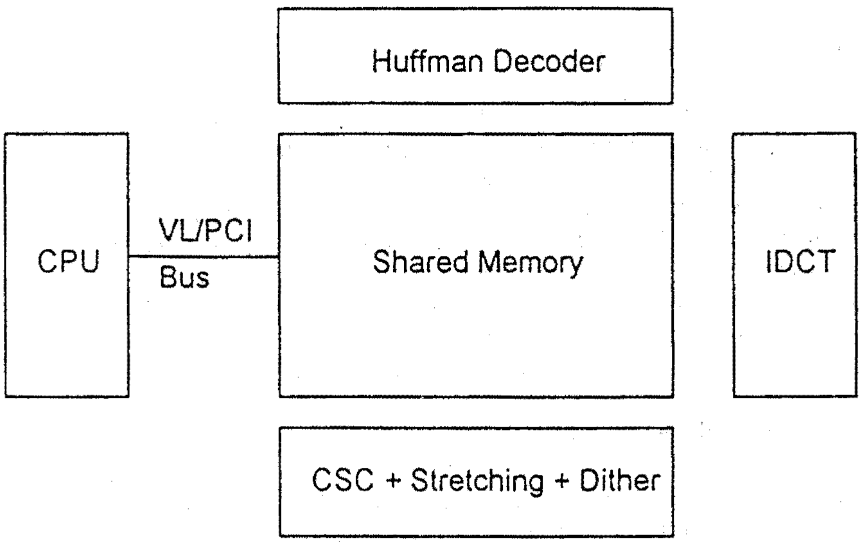


Fig. 2.0 Conceptual view of Collaborative Compression

The CPU performs the task of managing the different units, decoding headers, parsing the bitstream etc. In the initial stages, the data volume is low, and the data resides in CPU main memory anyway (deposited there by a peripheral). In the later stages, as data volume grows, the CPU acts as a re configurable pipeline manager and chains the other functional units using shared memory as the buffer between the units. This shared memory can be on a peripheral card connected to the CPU by a low latency, high bandwidth bus such as PCI.

6.0 CONCLUSION

We have presented the performance of a portable software MPEG player designed for PC platforms. The performance of the player is sensitive to memory/cache system performance and to display subsystem bandwidth. The performance of the software decoder approaches real-time for pseudo-SIF sequences (QSIF interpolated by 2X). IDCT occupies only a sixth of the time, much less than is conventionally expected. The highest performance workstations such as those base on the 300Mhz DEC Alpha should be able to perform real-time decoding of SIF streams. However for volume PC environments, collaborative compression offers the best means of achieving cost effect real-time MPEG video decoding.

7.0 APPENDIX

A. The Decoder Specification

The backend DLL contains all the functionality of the MPEG decoder. It is written entirely in 32bit ANSI-C. The output formats supported are (1) Windows 3.X Device Independent Bitmaps (DIB) which consists of a header and has the first pixel in the lower left corner of the image and (2) a simple array with pixel arranged in row major form, top left corner pixel leading (which we shall call DDB). For both these formats the following bit depths are supported:

(1) 8 bit dithered (DIB and DDB)
(2) 16 bit DIB (5,5,5) and DDB (5,5,5 ; 5,6,5; etc)
(3) 24 bit DIB and DDB
(4) 32 bit DIB and DDB

The following operations are also supported:

Play (I and P only, I, P and B)

Stop
Rewind
Fast Forward (by playing only I frames)
Fast Reverse (by playing on I frames)
Random Seek

The player optionally allows the user to control the decoding of various frame types to provide a scalable performance knob by skipping the decoding of B frames.

Following is a listing of the API supported by the decoder:

1) MpgDecInfo
2) MpgDecStart
3) MpgDecEnd
4) MpgDecFrame
5) MpgSeekFrame
6) MpgRewind

This API allows for random seeking, so that the player can implement fast forward, reverse play type of operation. For bitstreams containing only I Frames, this is a relatively simple operation. For bitstreams containing I, P and B frames the MpgSeekFrame operation becomes a little complicated since we cannot decode P or B frames without decoding the reference frames on which they are dependent. We overcame this problem by implementing the MpgSeekFrame function to always find the nearest I frame in the direction of the seek.

7.0 REFERENCES

[1] Patel, Ketan, et. al., "Performance of a Software MPEG Video Decoder", Proceedings of the ACM Multimedia conference, 1993.

[2] Ulichney, R., *Digital Halftoning*, MIT Press, Cammbridge, Mass. 1987.

[3] "Coded Representation of Picture, Audio and Multimedia/Hypermedia Information", Committee Draft of Standard ISO/IEC 11172, December 6 1991.

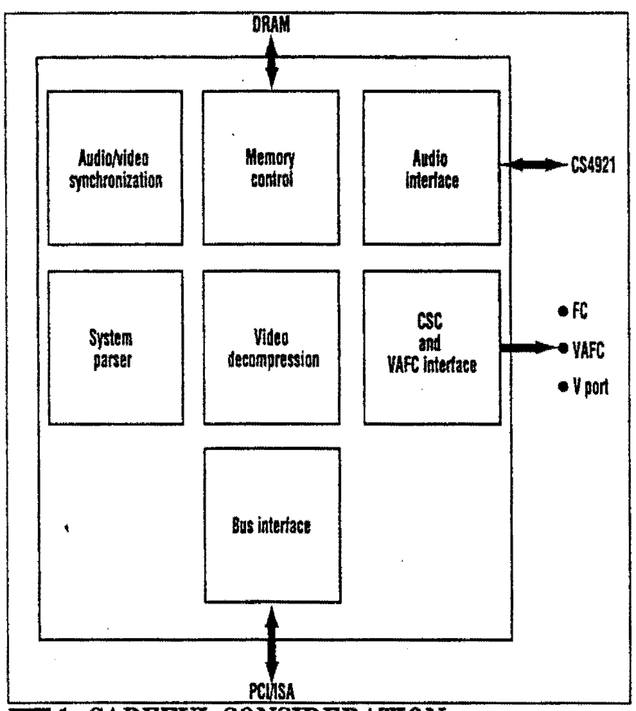
[4] "Microsoft Windows 3.1 SDK: Programmers Ref. Vol 1. - Overview", Microsoft Press, 1993.

Highly Integrated Controller Eases MPEG-1 Adoption

MPEG-1 Decoder Lowers The Cost Of Adding Video And Audio To PCs.

DAVE BURSKY

The drive to add multimedia capabilities to the personal computer, either by offering add-in cards or by building the capability directly on the motherboard, is forcing card and chip suppliers to find new ways to reduce costs. Individual add-in MPEG-1 decoder cards, although reasonable at several hundred dollars, must have their costs cut in half. The aim is to trim the user's cost of adding in MPEG-1 decoding to $100 for an off-the-shelf card, and even less if the system manufacturer is to include the capability as part of the base feature set of the PC.



1. CAREFUL CONSIDERATION to system partitioning has resulted in a highly-integrated MPEG-1 decoder developed by Cirrus Logic, the CL-GD5520. To simplify system design, the chip includes video decompression logic, a host-system interface to ISA PCI buses, DRAM control logic, system parsing control and audio/video synchronization logic. A multi-featured output port provides a VESA advanced feature connector (VAFC), a standard feature connector interface, or a proprietary V-port enhanced interface for transferring video images.

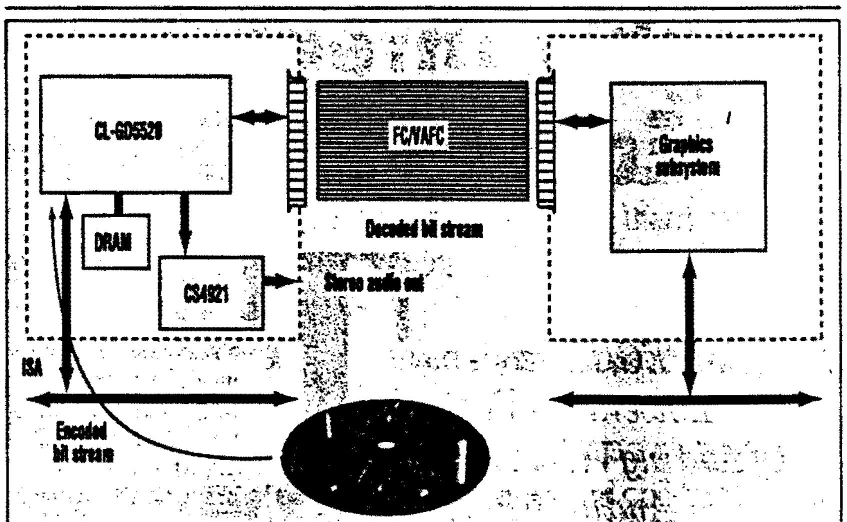
With that in mind, designers at Cirrus Logic studied system partitioning issues and came up with a three-chip solution that trims the cost of a full MPEG-1 subsystem to less than $50 in components *(Fig. 1)*. The three chips include the newly-designed CL-GD5520 MPEG-1 video decoder chip, the already available CS4921 audio decoder, and a commodity, 256-kword by 16-bit DRAM. The DRAM buffer can be expanded by adding a second 256k by 16 DRAM. The larger buffer improves the quality of the displayed video and allows the subsystem to handle larger audio/video streams.

With the three chips, designers can build systems that decode full-motion MPEG video from a variety of video sources. That includes CDs, MPEG-1 CD-i movies, DOS OM-1 (Open MPEG consortium) compatible titles, and Microsoft Windows MPEG MCI standard video.

Designed from the ground up to offer the simplest interface in the PC environment, the CD-GL5520's MPEG-1 video decompressor is based on the MPEG-1 core technology licensed from CompCore Inc. The core is surrounded with all the functions it needs to communicate with the rest of the system at data rates of 80 Mpixels/s off the video port, and at up to 132 Mbytes/s on the host-bus interface. Unlike several other highly integrated MPEG-1 chips that incorporate the audio playback channel on the chip, designers at Cirrus Logic decided to keep the function off the chip. That's because the economics of integrating the sound onto the video decoder chip shows that the all-in-one approach doesn't really lower the cost of materials.

The decompression chip also includes both PCI and ISA host-bus interfaces (including PCI bus master-

HIGHLY-INTEGRATED MPEG DECODER



2. JUST THREE CHIPS are needed to implement a full MPEG audio and video solution. These include Cirrus Logic's CL-GD5520 video decoder and CS4921 audio decoder, and a 512-kbyte DRAM (one 256-kword by 16-bit DRAM).

ing), a VESA advanced feature connector (VAFC) video output (which can also implement an FC or V-port interface), and an integrated color-space converter *(Fig. 2)*. Furthermore, due to its built-in arbitrary scaling and zoom capabilities, the chip can deliver windows of almost any size without degrading image quality. Pixel interpolation is used in the X direction and pixel replication in the Y direction when the images are resized.

To ensure that the MPEG-1 video and audio channels stay synchronized, a key issue in multimedia playback, designers at Cirrus Logic incorporated dedicated synchronization logic on the chip. The chip also includes a system parser to structure the MPEG-1 system-layer bitstream. For the audio channel, the chip delivers the parsed audio stream to the companion CS4921 audio codec developed by Crystal Semiconductor, Austin, Texas, a subsidiary of Cirrus Logic.

Support for both Windows 95 and the Plug-and-Play initiative is also included in the decoder chip. In addition, the PCI-bus mastering capability allows video data transfers to take place directly over the PCI bus, and directly to the graphics-controller's frame buffer. That eliminates the need for a ribbon cable that goes over the top of the cards. This solution also allows the graphic controller to display graphics at higher resolutions.

The integrated color-space conversion circuitry supports 5:5:5, 5:6:5, 8-bit error-diffused 3:3:2, and true-color 8:8:8 RGB formats, as well as 16-bit 4:2:2 YUV and AccuPak 8-bit YUV formats. That broad format support allows the video data to be delivered to just about any display controller subsystem. The chip also supports both NTSC and PAL video resolutions, thus expanding the potential market beyond the U.S. border.

Cirrus Logic's designers have also been busy developing the extensive driver support the chips will need for integrating them into a PC. Drivers are available for Windows 95, Windows 3.11, OM-1 DOS, VideoCD, and CD-i. The CL-GD-5520 is also fully compatible with the company's previously available graphics controllers and audio components. Reference design kits are available so that designers can quickly check compatibility MPEG-based software titles. Software development kits are also available for users who wish to incorporate MPEG video into their applications. □

PRICE AND AVAILABILITY

The CL-GD5520 is housed in a 208-lead PQFP. It sells for $32.00 apiece in lots of 1000 units. Samples are now available.

Cirrus Logic Inc., 3100 West Warren Ave., Fremont, CA 94538-6423; Saul Altabet, (510) 252-6286. CIRCLE 500

HOW VALUABLE?	CIRCLE
HIGHLY	525
MODERATELY	526
SLIGHTLY	527

ELECTRONIC DESIGN

August 21, 1995 Volume 43, Number 17

ELECTRONIC DESIGN (USPS 172-080; ISSN 0013-4872) is published twice monthly except for 3 issues in May and 3 issues in October by Penton Publishing Inc., 1100 Superior Ave., Cleveland, OH 44114-2543. Paid rates for a one year subscription are as follows: $105 U.S., $185 Canada, $210, $255 International. Second-class postage paid at Cleveland, OH, and additional mailing offices. Editorial and advertising addresses: ELECTRONIC DESIGN, 611 Route #46 West, Hasbrouck Heights, NJ 07604. Telephone (201) 393-6060. Facsimile (201) 393-0204. Printed in U.S.A. Title registered in U.S. Patent Office.

For subscriber change of address and subscription inquiries, call (216) 696-7000. Mail your subscription requests to: Penton Publishing Subscription Lockbox, P.O. Box 96732, Chicago, IL 60693.
POSTMASTER: Please send change of address to ELECTRONIC DESIGN, Penton Publishing Inc., 1100 Superior Ave., Cleveland, OH 44114-2543.

ELECTRONIC DESIGN

August 21, 1995 Volume 43, Number 17

DEPARTMENTS

Jesse H. Neal Editorial Achievement

Awards: 1967 First Place Award
1968 First Place Award
1972 Certificate of Merit
1975 Two Certificates of Merit
1976 Certificate of Merit
1978 Certificate of Merit
1980 Certificate of Merit
1986 First Place Award
1989 Certificate of Merit
1992 Certificate of Merit

COVER ART: MARGARET ENDRES, BRUCE JABLONSKI, JOHN BEUKAMANN

FA 17.1: An MPEG-1 Audio/Video Decoder with Run-Length Compressed Antialiased Video Overlays

4S

Dave Galbi, Everett Bird, Subroto Bose, Eric Chai, Yen-Ning Chang, Pierre Dermy, Nishendra Fernando, Jean-Georges Fritsch, Eric Hamilton, Barry Hu, Ernest Hua, Frank Liao, Ming Lin, Ming Ma, Edward Paluch, Steve Purcell, Hisao Yanagi, Sun Yang, *Miranda Chow, *Takeya Fujii, *Akio Fujiwara, *Hiroyuki Goto, *Keiji Ihara, *Shinichi Isozaki, *Janny Jao, *Isami Kaneda, *Masahiro Koyama, *Tomoo Mineo, *Izumi Miyashita, *Goichiro Ono, *Shinji Otake, *Akihiro Sato, *Hideo Sato, *Akira Sugiyama, *Katsunori Tagami, *Kenji Tsuge, *Tomoyuki Udagawa, *Koji Yamasaki, *Sadahiro Yasura, *Tsuyoshi Yoshimura

C-Cube Microsystems, Milpitas, CA
*JVC, Yamato, Japan

This chip decodes MPEG-1 audio and video in real-time when connected to a single 80ns 256kx16 DRAM. The features of the chip are summarized in Table 1. A block diagram is shown in Figure 1. An MPEG-1 system stream, optionally embedded in a CD data stream, is sent to the chip on either an 8b host bus or a serial bus. The host interface contains a code FIFO that buffers input bit streams before they are written to the audio, video or overlay bit-stream buffers in DRAM. The MPEG system stream is processed by interrupting the on-chip CPU after a packet of compressed data has been written to DRAM. The CPU reads the system stream headers out of the code FIFO and initiates a block transfer of the next packet of compressed data to DRAM. The chip uses less than 5% of the clock cycles for system stream processing. The chip alternates between audio decoding and video decoding, with the audio portion using 15% of the clock cycles and video using 80%.

Audio and video bit streams are read from DRAM into a decoder FIFO. When decoding video, variable length codes (VLCs) are converted to fixed length codes (FLCs) by the VLC/FLC decoder. The VLC/FLC decoder writes video AC coefficients into ZMEM. When decoding audio, the VLC/FLC decoder extracts subband samples from the bit stream, performs degrouping and writes the results into ZMEM.

The signal processing unit (SPU) receives commands from the CPU and executes these commands in parallel with the rest of the chip. The SPU datapath is shown in Figure 2. The SPU performs three commands:

1. Dequantization and IDCT for video decoding
2. Dequantization and descaling for audio decoding
3. Matrixing and windowing for audio decoding

The Dequant/IDCT command reads an 8x8 block of AC coefficients from ZMEM and writes the results to a double buffered PMEM. During video decoding, the TMEM is used as the quarter-turn memory for the IDCT and the QMEM contains the quantizer matrix. The data flow for the SPU audio commands is shown in Figure 3. The Dequant/Descale command reads a vector of 32 audio subbands from ZMEM and writes the results to 32 locations in TMEM. The other 32 locations in TMEM are used to accumulate 32 partially-decoded audio samples. The Matrix/Window command reads 33 20b matrix results from PMEM and adds the product of matrix results and window coefficients to the partially decoded audio samples in TMEM. The Matrix/Window command then computes 4 20b matrix results that are written to PMEM. The DRAM controller writes matrix results in PMEM to DRAM and fetches previous matrix results for windowing. These DRAM transfers are in parallel with SPU operation. After 8 Matrix/Window commands, TMEM contains 32 decoded audio samples that are written to an audio output buffer in DRAM. The audio output unit receives decoded audio data from DRAM in an 8B FIFO and sends them out to the pins of the chip.

During video decoding, the motion- compensation unit receives reference blocks fetched from DRAM and half-pixel offsets them if needed. The offset reference blocks are added to the IDCT result in PMEM and the sum is stored back into PMEM. The motion compensation unit and SPU work in parallel on opposite halves of PMEM. After the offset reference blocks have been added to PMEM, the resulting decoded pixels are written to DRAM.

The video output unit receives decoded pixels from DRAM in a 112B luminance FIFO and a 128B chrominance FIFO. Luminance and chrominance are horizontally and vertically interpolated by 2x in each direction using a 7-tap horizontal filter and a 3-tap vertical filter. Compressed video overlays are read from DRAM into an overlay FIFO, decompressed and then blended with interpolated MPEG video. Finally, the pixels are optionally converted to RGB and output.

To decode both audio and video with only one 80ns 256kx16 DRAM, the chip must minimize the use of DRAM bandwidth and DRAM space. This is accomplished with the following techniques:

1. Decoded B frames are compressed before being written to DRAM to save about 200kb of DRAM space.
2. Video overlays are compressed to reduce the size of the overlay bit stream buffer in DRAM and to reduce the DRAM bandwidth needed to fetch the overlay bit stream.
3. The on-chip CPU has 96 CPU registers and a 16b instruction word. This gives a 1.9x improvement in instruction density compared to a conventional RISC CPU with 32 registers and a 32b instruction word.
4. The 20b audio matrix results are packed into 1.25 16b words before being written to DRAM.

Decoded B frames are compressed with a lossy DPCM compression technique to save DRAM space. Scan lines are DPCM-decoded in the video output unit. Video overlays are compressed with a run-length code with 4 symbol lengths: 4, 8, 12 and 20 bits. The symbols with N bits cover all runs of at least N/4 pixels so the maximum bit-rate of the compressed overlay bitstream is 4b/pixel. The typical overlay bit rate is 0.6b/coded pixel. The overlay symbols select a shadow color, a text color or transparent (Figure 4). To reduce jaggies and flicker, the MPEG/shadow color boundary and the shadow/text boundary are antialiased using a 2b blend factor indicated by the overlay symbols. The overlay can be gradually faded on or off with a 5b global fade factor. The on-chip CPU has an instruction set designed for instruction density and ease of implementation. The 16b instruction word contains two 6b register addresses and a 4b opcode. There are a total of 96 CPU registers of which 64 are accessible at one time. When a CPU interrupt occurs, 32 interrupt registers are used in place of 32 regular registers. The CPU datapath is 24b wide. CPU instructions are stored in DRAM and are read into a 1024x16 instruction memory as needed. A micrograph is shown in Figure 5.

Acknowledgments

The authors thank the following employees of Matsushita Electronics Corporation for contributions to the chip: K. Hamaguchi, A. Haza, J. Huard, Y. Ochi, Y. Okada, M. Suzuki and A. Yamamoto.

0-7803-2495-1/95/$4.00/© 1995 IE[EE]

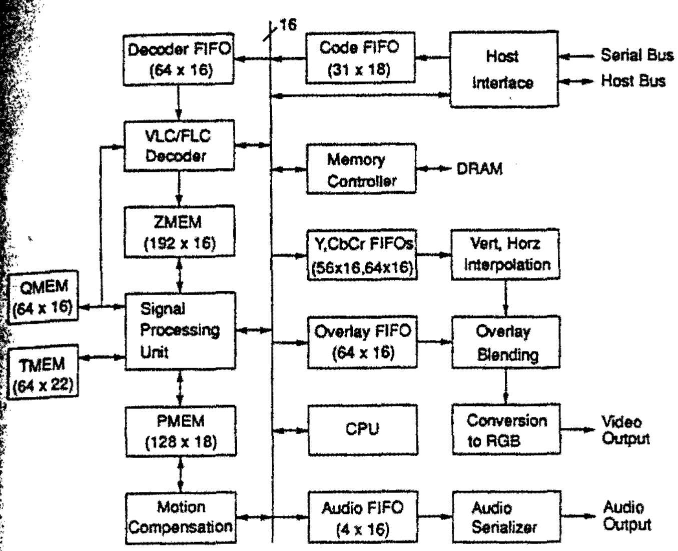


Figure 1: Block diagram.

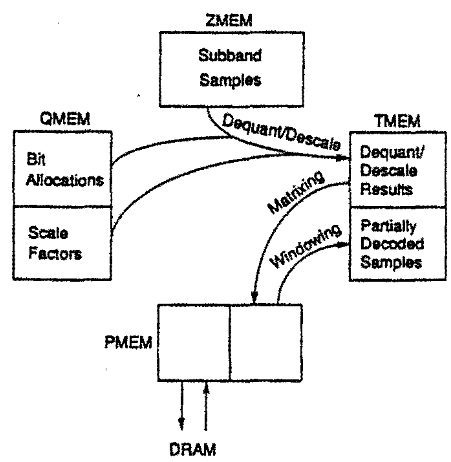


Figure 3: SPU audio commands.

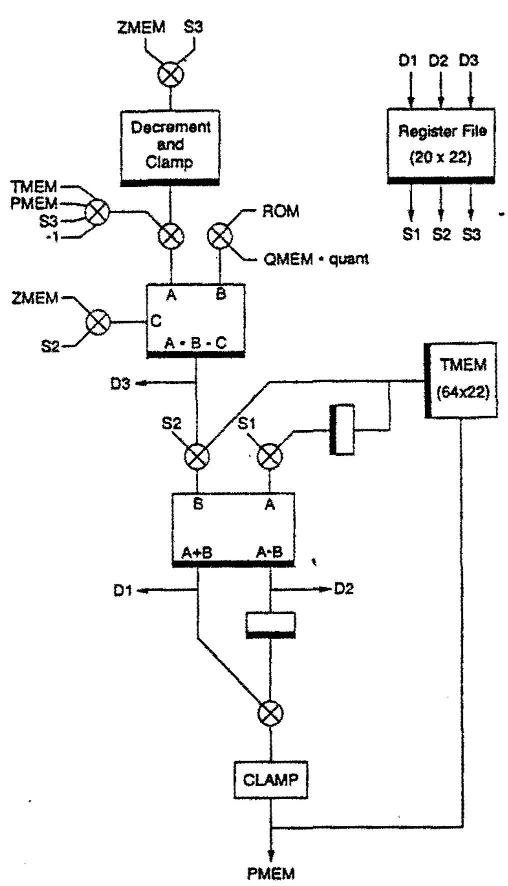


Figure 2: Datapath of signal processing unit.

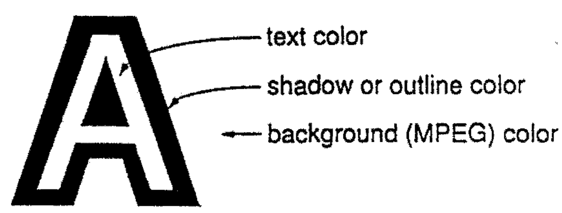


Figure 4: Video overlay colors.

Figure 5: See page 381.

Audio decoding performance	2 channels of 48kHz audio
Video decoding performance	352x240 @ 30Hz or 352x288 @ 25Hz
Video overlay resolution	up to 768x576
Process technology	0.5μm (drawn) 2-layer metal CMOS
Die size	11.5x11.5mm²
Logic transistors	305k
Memory transistors	485k
Clock frequency	40MHz
Operating voltage range	2.7V to 3.6V
Typical power consumption	600mW at 3.3V, Ta = 25°C
Max. power consumption	740mW at 3.6V, Ta = 70°C
Package	128-pin PQFP (18x18mm² body)

Table 1: Feature summary.

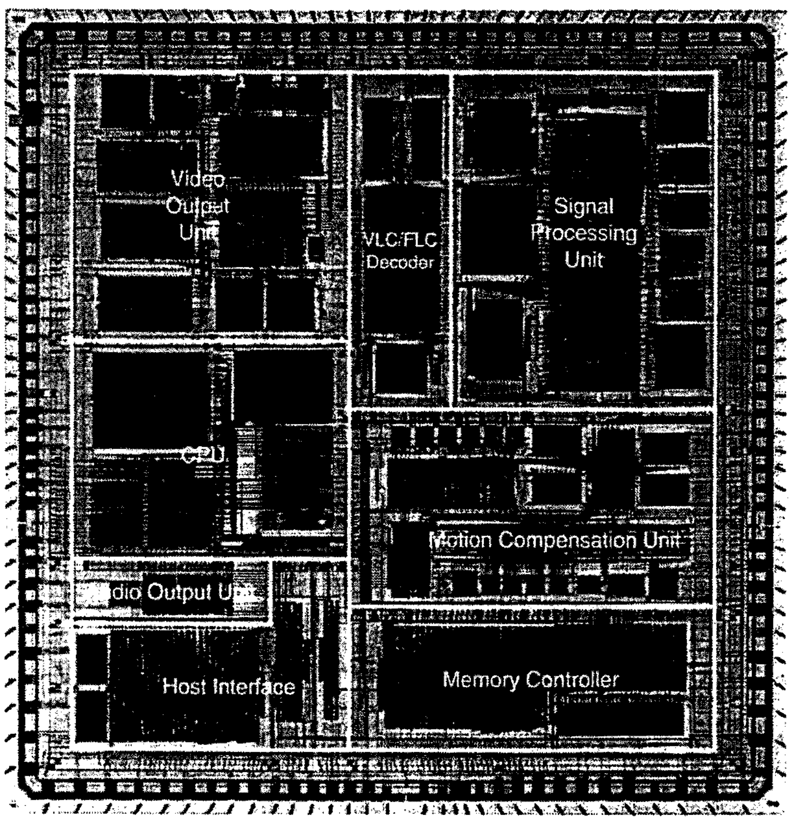


Figure 5: Micrograph of MPEG-1 A/V decoder.

FA 17.2: A Half-pel Precision MPEG2 Motion-Estimation Processor with Concurrent Three-Vector Search

(Continued from page 289)

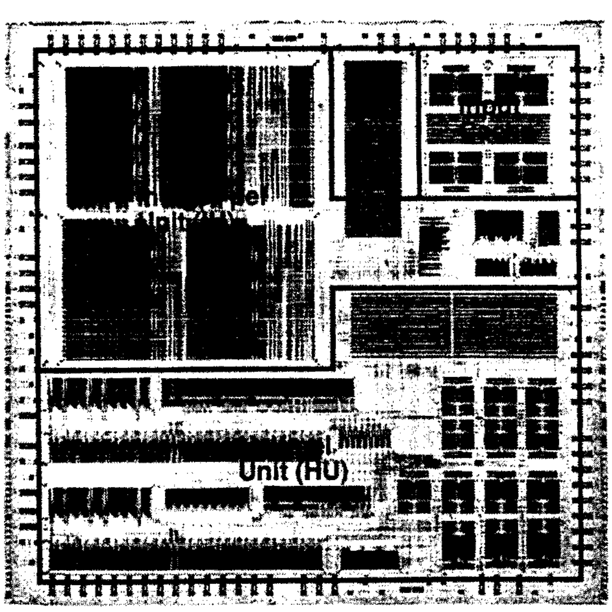


Figure 4: ME2 micrograph.

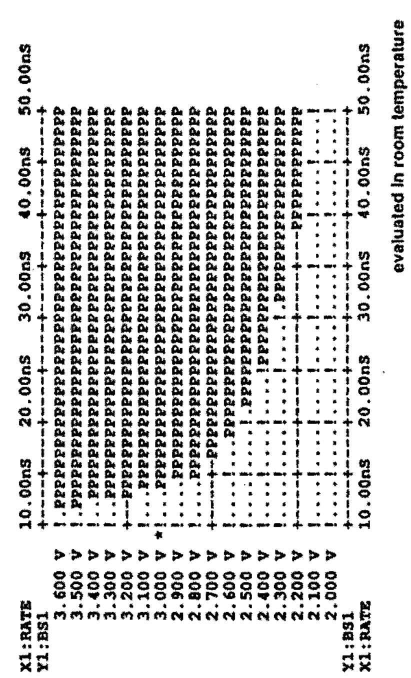


Figure 5: ME2 shmoo plot.

A Y

SINGLE CHIP MPEG AUDIO DECODER

Greg Maturi
LSI Logic Corporation
Milpitas, California

ABSTRACT

An IC has been designed and fabricated which can take an MPEG System or MPEG Audio stream and decode LayerI and Layer II (MUSICAM) encoded audio into 16 bit PCM data. Audio/Video synchronization, cue and review is provided via its external channel buffer.

SUMMARY

The Single Chip MPEG Audio Decoder will take an MPEG Layer I or II (MUSICAM) System or Audio stream, and provide complete decoding into 16 bit serial PCM outputs. In addition, presentation can be delayed and audio frames skipped by means of the channel buffer, an external 256K x 4 DRAM controlled by this IC. This allows coarse synchronization of audio and video for skews up to 1 second for Layer I and 2.5 seconds for Layer II. Control over which frames are played or skipped provides cue /review features. Except for the channel buffer DRAM and a 25 - 30 Mhz clock, no other hardware is required. The IC is controlled by an 8 or 16 bit microprocessor, but can operate as a stand alone device with reduced flexibility.

The IC can receive data up to a 15 Mbits/second either serially or through microprocessor interface (selectable for 8 or 16 bits). An input fifo allows the IC to handle burst rates of up to 7.5 Megabytes/sec for up to 128 bytes. The IC will strip out the audio streams from MPEG system streams and provide presentation time and parametric information to the host. The audio frames will then be stored in the channel buffers.

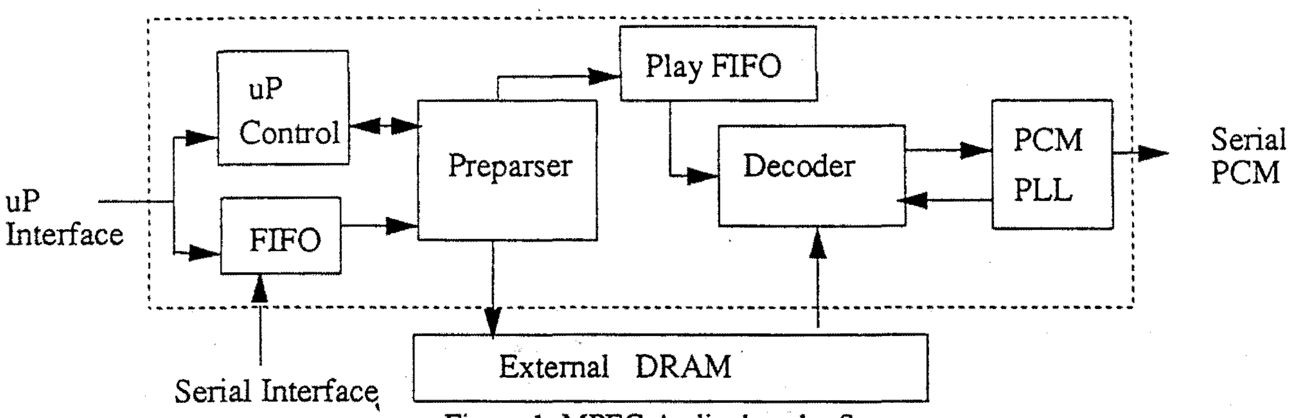


Figure 1. MPEG Audio decoder System

The IC is divided into 4 major parts: the preparser, the decoder, the DRAM controller, and PCM interface.

The preparser performs several functions: system/audio stream synchronization, stripping off of parametric and presentation time headers, syntax checking, CRC checking, and cataloging

Manuscript received June 5, 1992

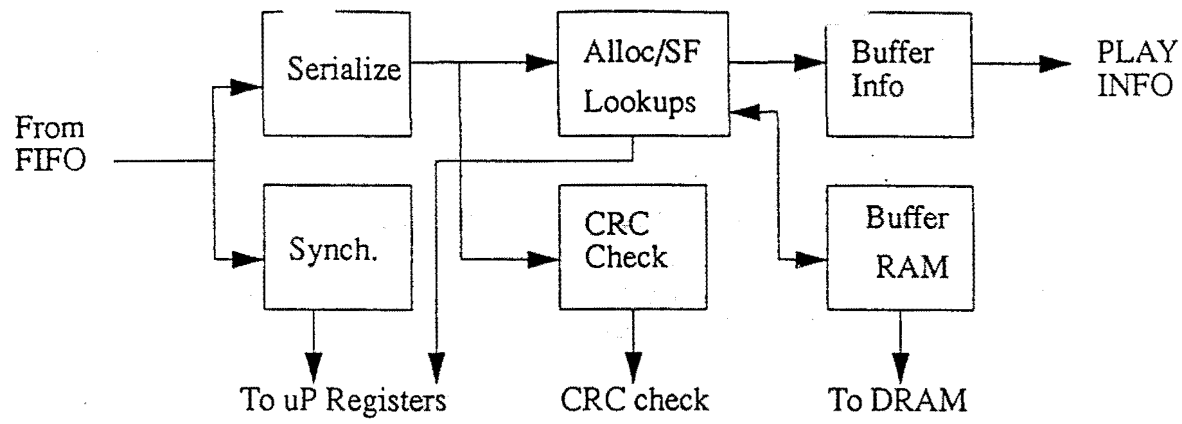


Figure 2.Preparser Architecture

frames. Error concealment (by repeating the last good frame) can be provided automatically. Since the frame must be partially expanded to obtain this information, the frame is stored in the channel buffer in a partially expanded form. A playlist is generated to tell the decoder which frame to decode next. The microprocessor can control which direction to fill the playlist, skip frames in the playlist and which direction for the decode to read the playlist.The decoder does most of the algorithmic work: It performs inverse quantization, scaling, and subband synthesis. It uses a 24 bit architecture. In addition, on Layer II it performs degrouping prior to dequantization. Filter coefficitents,dequantization values, scratchpad and vector memories are internal.

The DRAM controller provides RAS,CAS, address and data to the DRAM. It arbitrates between the preparser and the decoder. It also provides hidden refreshing. This controller requires a 256K x 4 DRAM (100 ns or faster)

The PCM interface buffers the PCM output from the decoder and provides 3 and 4 wire serial output compatible to most serial DACs. The serial clock is generated from the system clock using a fixed point divisor provided by the host (4 bits integer, 16 fractional). The decoder can also be bypassed, allowing serial PCM to be passed directly from input to output.

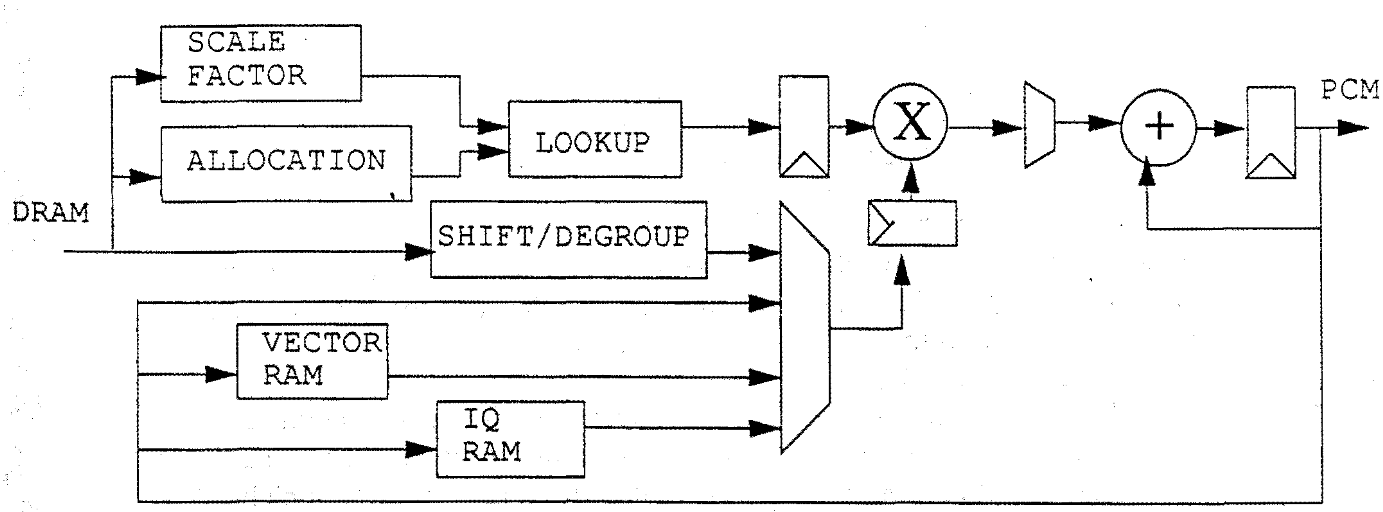


Figure 3.Decoder Architecture

Functional Block Operation

When initialized, the decoder synchronizes itself by monitoring the data stream and locating an audio frame in the data stream.When MPEG data is input, the chip strips away all unneeded information, retaining only the audio and control data. This data is then partially decompressed and stored in a channel buffer. When the appropriate control signals are seen, this stored data is played (fully decompressed and output in PCM format).

These activities are accomplished in general as follows:

Input Synchronization and Buffering

Data in either serial or parallel form enters the MPEG Audio decoder through the Controller Interface. The data is first synchronized to the system clock (SYSCLK), then is sent to the Input Data FIFO. The FIFO buffers data and supplies it to the Preparser. The FIFO can accommodate burst rates up to (input clock)/4 bytes/sec, for bursts of 128 bytes.

System Preparser

The Preparser performs stream parsing. For ISO System Stream parsing and synchronization, it detects the packet start code or system header start code and uses these to synchronize with packets. The parser reads the 16 bit "number of bytes" code in either one of these headers and counts down the bytes following. When count 0 is reached the next set of bytes should be a sync word, If not, the sync word seen was either emulated by audio or private data or a system error. The preparser will not consider itself synchronized until 3 consecutive good syncs have occurred. Likewise, it will not consider itself unsynchronized until 3 false are detected. This hysteresis is detailed in the flowchart in Figure 4. Upon synchronization, the preparser returns the presentation time stamp for use in audio-video synchronization.

Audio Synchronization

If the synchronization code is the selected audio stream or the input stream is only audio, the preparser will then synchronize to the audio stream. It first detects the 12 bit audio sync, if the bitrate is not free format, the bytes remaining in the frame are calculated from the bitrate and sampling frequency (extracted from the parametric values in the bitstream) according to the formula:

bytes = 48 * bitrate/sampling_frequency (I)

bytes = 144 * bitrate/sampling_frequency (II)

This value is loaded into a byte counter. As with the system synchronization, when the counter down counts to zero the preparser verifies the next 12 bits are a sync code. if the padding bit is set the counter will wait 4 bytes on Layer I and 1 byte on layer II before checking for the sync code. The hysteresis is similar to that of the MPEG system synchronization. The audio synchronization is identical for free format except one extra frame is required where the bytes in the frame are counted rather than calculated. Figure 5 shows the audio synchronization.

Storing in Channel Buffer

After synchronization, allocations and scalefactors are separated out and stored in the channel buffer. In layer I there are 32 bit allocations, each allocation 4 bits representing 0 to 15 bits per sample, 1 not allowed. In LayerII there are 8 to 30 allocations 1 to 4 bits in length, representing 0 to 16 bits per sample,1 not allowed. In Layer II, information on whether the samples are grouped (three samples combined into a single sample) is also stored with the allocation.

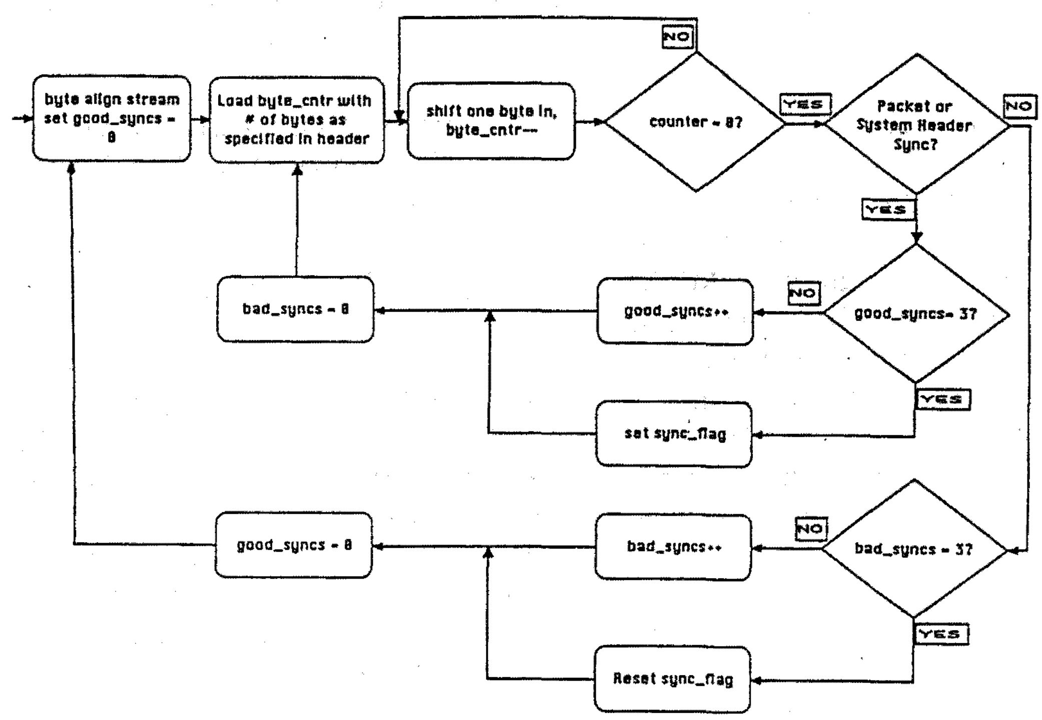


Figure 4. System Sychronization Hysterisis

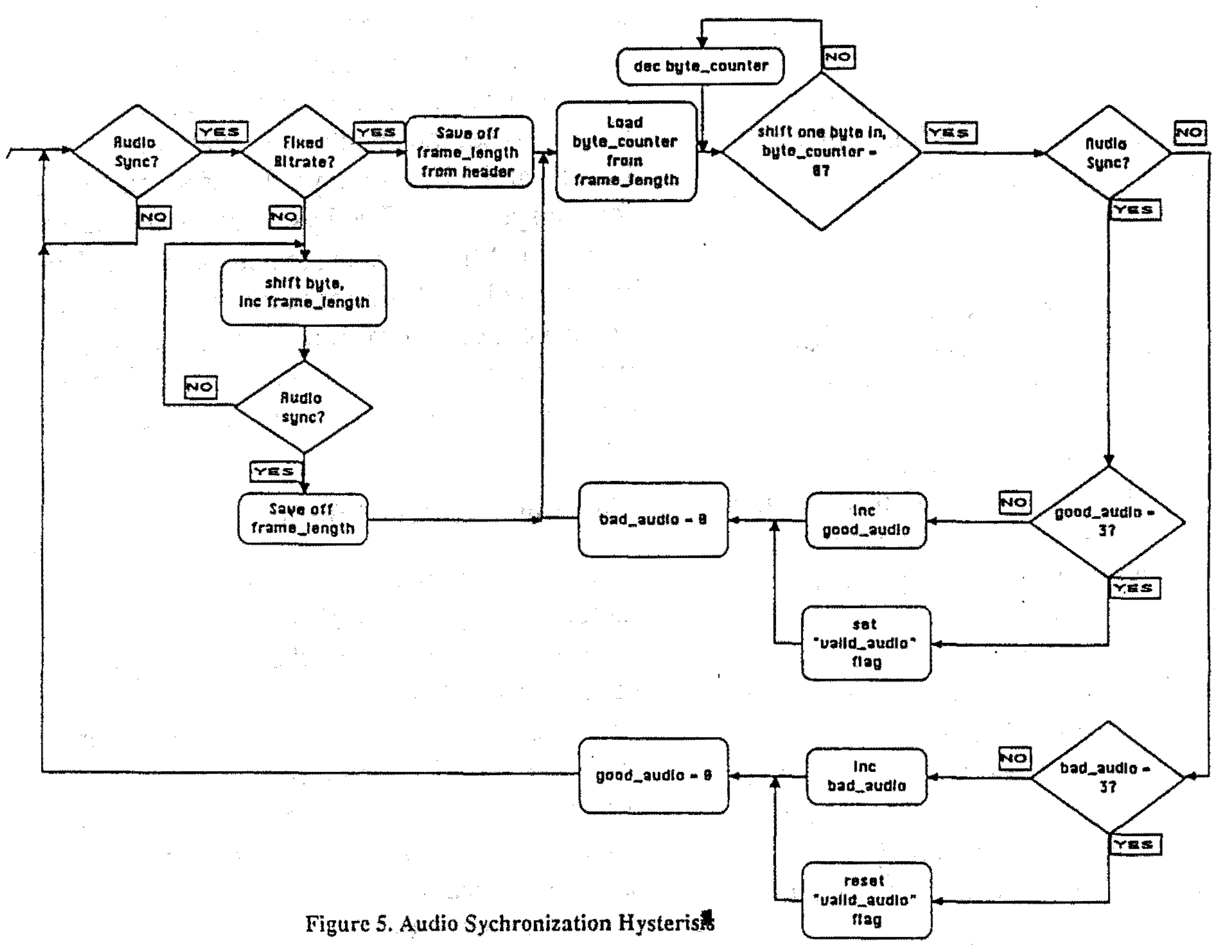


Figure 5. Audio Sychronization Hysterisis

Scalefactors are 6 bits indı to a lookup table, indicating the maximum amplitude of the samples in a subband. In Layer I, there is one scalefactor for each non-zero bit allocation. In Layer II there is 1 to 3 scalefactors per non-zero bit allocation. The actual number is determined by a 2 bit scalefactor select (again 1 per non-zero allocation). The preparser uses this information to separate out the scalefactors. The format that the allocations and scalefactors are stored in the memory is shown in Table I.

Table 1: channel buffer format

Row Address (HEX)	Information Stored
000:03F	allocations (channel 1)
040:07F	first scale index (channel 1)
080:0BF	second scale index (channel 1)
0C0:0FF	third scale index (channel 1)
100:13F	allocations (channel 2)
140:17F	first scale index (channel 2)
180:1BF	second scale index (channel 2)
1C0:1FF	third scale index (channel 2)

As allocations are being written to the channel buffer a small RAM records whether the allocation was non-zero. It then uses this information to separate out scalefactors without having to reread the channel buffer. On Layer II this is also done for scalefactor select bits. The same RAM holds these values for later use in scalefactor decoding.

Samples are left bitpacked when put into the channel buffer, just as received in the bitstream. They are stored immediately after the allocations and scale indices.

Audio data ca. .e parsed at input clock/2 bits per second. The limitation is the DRAM timing.

Parametric Data

The 20 bits following the audio sync word are called parametric data bits. These bits are used by the decoder and presented to the host interface. A maskable interrupt which is asserted as soon as these bits are read from the bitstream lets an optional microprocessor know these bits are available. Table II shows the bit definition.

Table 2: Parametric Data Format

Bit	Data
19 (MSB)	ID
18:17	Layer
16	Protection
15:12	Bitrate
11:10	Sampling Frequency
9	Padding
8	Private
7:6	mode
5:4	mode extension
3	copyright
2	original
1:0	emphasis

Ancillary data

The data immediately following the last data bit until the next frame sync is considered ancillary data. The last bit of data is calculated from the decoded allocations and scale indices. This data is stored in a 16 X 8 bit FIFO. An interrupt indicates valid data in the FIFO, when the FIFO is half full, and when it has overflowed. If the ancillary data is less than 8 bits or a sync word is detected the ancillary bits are left aligned and written to the FIFO.

Play Buffer

The play buffer is a FIFO indicating the location in the channel buffer of the next frame to be played as well as minimum information the decoder needs to decode the samples. Usually, the play FIFO contains consecutive 4K block addresses for layer II and 2K block addresses for Layer I. However, if errors occur, the next address will be the last good frame stored.

The information that is passed in the play buffer is mode and mode extension, and a bit indicating if the frame should be blanked or played. This bit is set if an error occurs and error concealment is not selected. Since bitrate and sampling frequency are not allowed to be changed without resetting the decoder, the frame sizes remain the same. A time equivalent to the frame in error can be silenced with this method.

Decoder Operation

The decoder receives data for full decompression from the channel buffer. The location of this information and other required parameters are provided by the play buffer. The decoder performs all of the following functions: degrouping,dequnatization, denormalization and subband synthesis. Except degrouping, all functions are performed by use of a 2-cycle 24-bit multiplier -accumulator. A ROM provides lookup tables for scalefactors, quantization values, DCT and window coefficients. Two separate RAMS are provided, one for the dequantized coefficients, and one for the vectors generated in the subband synthesis. All memory is 24 bits, a block diagram is shown in Figure 3.

The decoding process begins with a start command being generated from the microprocessor or external start input. At that point the decoder reads parameters and channel buffer address information from the play buffer, and requests data from the channel buffer. The DRAM controller arbitrates between the preparser requesting to write data to the channel buffer and the decoder trying to read data for decompression.

In the first read, the decoder obtains allocation and scalefactor information. In the second read, the decoder obtains 1 to 5 nibbles containing the subband sample. If degrouping is required, the decoder implements the degrouping process:

```
For (i =0;i<3;i++)
{
  Sample[i] = c%nlevels;
  c = (int) c/nlevels;
}
```

by using a serial divider.

Dequantization is then performed by the following equation:

IQ[i] = (Sample[i] + D) * C

where C and D are both in lookup tables indexed by bit allocation.

Next is denormalization:

IN[i] = IQ[i] * scalefactor [scalefactor index]

This process is repeated for 32 samples. Each 24 bit denormalized sample is stored in the

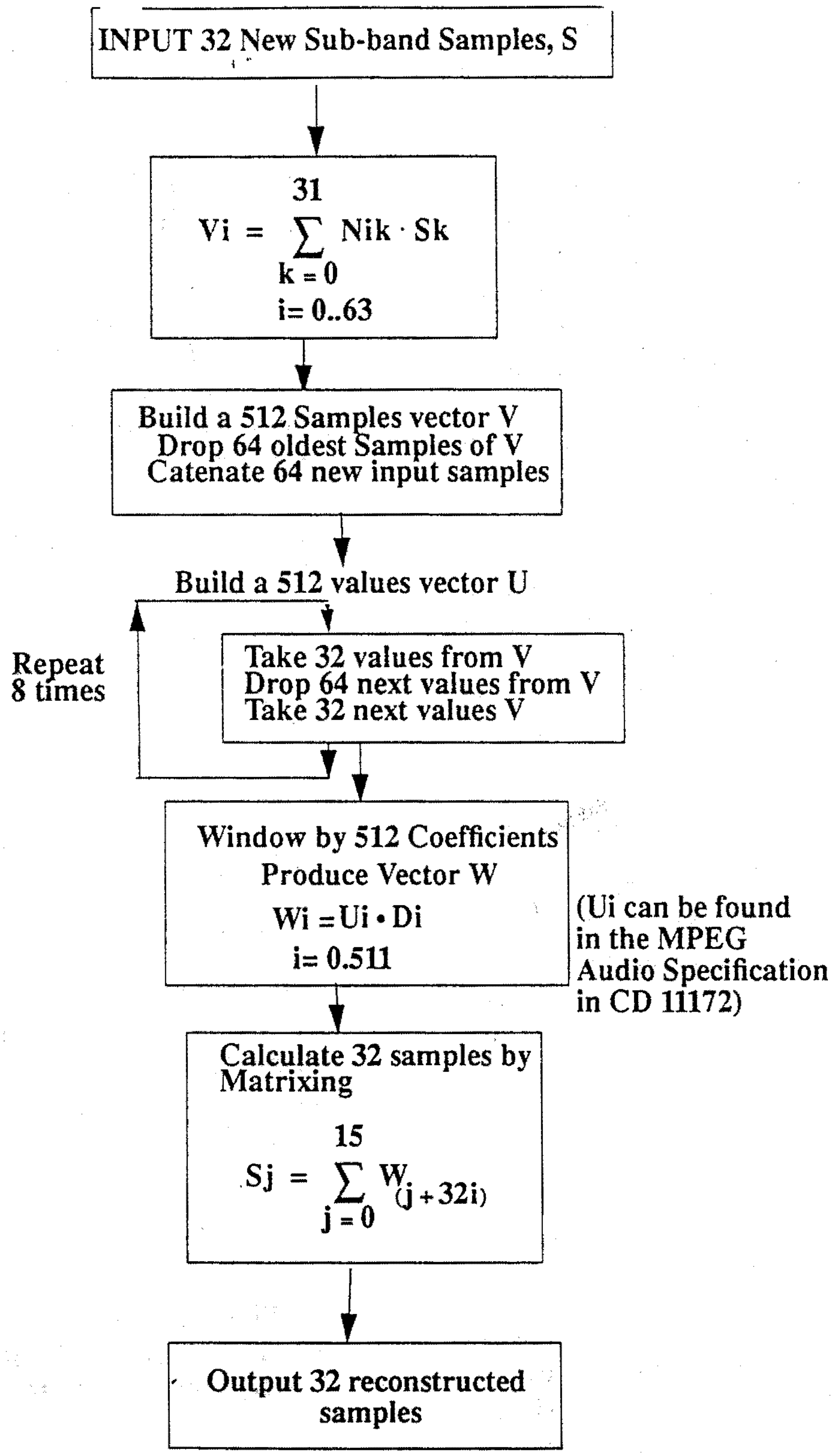


Figure 6. Subband Synthesis

denormalization RAM. These values are then used for subband synthesis.

Subband synthesis

The MPEG standard defines subband synthesis as shown in Figure 6. This process can be broken down into two functions. The first is an odd frequency inverse DCT, the second is a window function (with special addressing). The inverse DCT and window function are performed in parallel. That is the window and calculate samples is performed when the next PCM word has to be shifted out. In between the windowing function, the inverse DCT is performed. To insure that the data used for the window function is not from 2 different sets of sub-samples, the inverse DCT output is stored to a scratchpad portion of the vector ram. When the last of the 32 PCM samples has been transferred, this scratchpad is written to the correct section of the vector RAM.

PCM output

The PCM interface is responsible for obtaining data from the decoder, serializing it, and generating the control signals at the proper time for analog conversion by a serial DAC. In addition, refresh timing is based on the PCM clock.

The PCM contains registers that divide down the input clock to obtain the proper sampling frequency for the output PCM stream. These registers are either loaded on power up or written via the microprocessor port.

The first register is a 4 bit register, that indicates that is used to divide the input clock to obtain 2X the DAC serial clock. The 2nd register is 16 bits, and represents the fractional part of this divisor. Every time the 4 bit register counts down to zero, this fractional register is accumulated. Every time the accumulation exceeds 1 an extra clock cycle is added. Running with the slowest input clock and the fastest sampling frequency, this will produce a 10% variation in the serial clock, but the actual sampling frequency will be accurate to greater than 200 ppm.

As soon as the PCM word is loaded into the parallel to serial register, the PCM interface requests another from the decoder. The decoder completes its current DCT or inverse quantization, and then performs the window function described under subband synthesis. The decoder puts the PCM word into an output register, which the PCM interface will load into the parallel to serial converter when the previous word has been shifted out.

Conclusions:

The MPEG audio IC developed considers system level, interface and rate control issues, rather than just the number crunching involved in MPEG Audio compression. The IC can provide complete decoding from system to PCM with minimal additional hardware.

Acknowledgments:

The author would like to thank Peng Ang, Juergen Lutz and Simon Dolan. I would also like to thank Dave Auld and Neil Mammen for their incessant helpful suggestions during the development of this IC.

References:

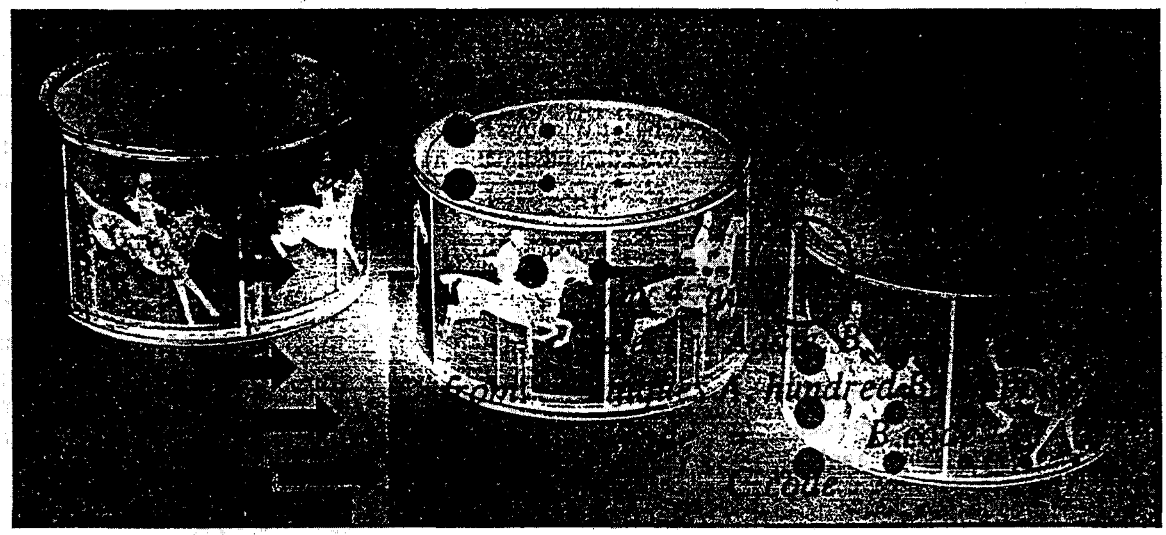
1. ISO CD 11172-3: CODING OF MOVING PICTURES AND ASSOCIATED AUDIO AT UP TO ABOUT 1.5 MBITS/s.

Author Biography:

Greg Maturi received his B.S.E.E. degree in 1981 from the University of Virginia. From 1982 to 1985 he worked for Harris Aerospace division in Imploring, Florida implementing video compression algorithms. From 1985 to 1990, he worked for Texas Instruments, designing DSP circuitry for artificial intellignece systems. In 1991, he joined LSI Logic as an IC designer, where he is responsible for the development of the single chip MPEG Audio Decoder.

A4

SQL DATABASES



The Great Leap *Forward*

The awkward years are over. PC-based SQL database servers have grown up to deliver on the promise of reliability and power.

By Brian Butler and Thomas Mace

Maturity has come quickly to PC-based structured query language (SQL) databases. In last year's roundup, we asked if 32-bit SQL databases for Intel processors were good enough to bet a business on. The answer, coming after months of ups and downs in the test labs, was only a qualified yes. While some of the products shone, many ran into serious difficulties and a few suffered outright failures (for details, see "PC-Based SQL: Time to Commit?", *PC Magazine*, October 12, 1993).

This year's testing offers a much rosier picture. Even though we more than quadrupled the size of our test database and boosted the complexity of our performance tests, all vendors came through with flying colors. While we still saw a wide range of performance

IN THIS REVIEW

Photography by Scott Van Sicklin

results, stability and reliability have improved dramatically across the board.

This is not to say that client/server SQL has become a trivial exercise. Integrating and debugging client and server operating systems, application code, networking components, and the SQL database itself demand real expertise. Even the best database must be backed up by solid client systems, networking cards and cabling, the network operating system and protocols, and the server hardware itself.

The advantages to client/server computing clearly predominate, however. For on-line transaction processing (OLTP) and decision-support applications, client/server offers reasonable hardware costs, faster application development, and for your end users, the familiar PC environment. Upsizing PC databases to client/ server carries with it the benefits of greater reliability, lower network loads, and centralized management. But whichever path you're on, SQL database servers for the Intel platform have made a quantum leap in quality.

OUR REVIEW LINEUP

This story covers most major 32-bit SQL database servers currently available for the Intel platform. All products covered in last year's story receive follow-up coverage and are reviewed in full if they have been released in major new revisions. Our main reviews, based on five months of lab testing, cover Informix OnLine for SCO Unix 5.01, Microsoft SQL Server for Windows NT 4.21, Oracle7 Server for NetWare 7.0.16, Sybase SQL Server for NetWare 10.01, Watcom SQL Network Server for NetWare 3.2, and XDB-Enterprise Server 4 for Windows NT.

We attempted to test and review the ASK Group's Ingres Server for OS/2 6.4.3. Ingres Server had serious difficulties with last year's tests, and many of the problems we found had not been resolved in the version we saw this year. During testing, The ASK Group was acquired by Computer Associates International, which withdrew Ingres Server from the market for debugging (for details, see the sidebar "Ingres Server: Still on Hold").

Client/server offers reasonable hardware costs, faster applications development, and for end users, the familiar PC environment.

Two databases covered in last year's story, Gupta SQLBase Server for NetWare and IBM DB2/2 have not been upgraded in the interim but will ship in major new revisions within the next few months. We were able to put a beta version of SQLBase Server through some of our tests (for details, see the sidebars "Preview: Gupta SQLBase Server" and "Coming Soon: A New DB2/2"). We were also able to examine a beta version of Borland International's new SQL entrant, The Borland InterBase Workgroup Server 4.0 (for details, see the sidebar "Preview: Borland's InterBase").

Cincom Systems and Raima Corp. declined to participate in this story because they could not free up support resources during our test cycle. Btrieve Technologies' NetWare SQL (formerly Novell's NetWare SQL) has not had a significant upgrade since it was last reviewed.

OS UNDERPINNINGS

Vendors were allowed to specify a 32-bit operating system for their server platform. Three OS's are represented this year: Microsoft Windows NT, NetWare, and SCO Unix.

Windows NT, which shipped during the past year, is a new platform for SQL. Its thread-based model, graphical administration tools, and strong networking support worked well for both Microsoft SQL Server and XDB-Enterprise Server. Other vendors, including IBM and Sybase, also plan to ship NT versions of their products.

NetWare, chosen by Oracle, Sybase,

H I G H L I G H T S

SQL Databases

ADVANCED FEATURE SETS are fast becoming standard as SQL databases grow ever more sophisticated. Most of the products we saw support ANSI cursors, triggers, stored procedures, and declarative referential integrity. All support BLOBs and cost-based query optimization.

SYMMETRIC MULTIPROCESSING is clearly the next performance frontier as SMP hardware becomes more common. Some engines use operating-system threads or processes to divide tasks over CPUs; others launch multiple instances of the database itself. Though some servers are SMP-ready, others require that you buy a special SMP version. Upcoming releases will add dedicated support for parallelizing queries, loading, and index creation. The only platform bucking this trend is Novell NetWare, which does not support multiple CPUs.

PRICES ARE DROPPING, driven by increased competition. Much of the pressure comes from sophisticated bundles targeted at the workgroup market. Client/server SQL remains an expensive proposition, however. The need for skilled administrators and the lack of turnkey software solutions means substantial outlays for software development and maintenance.

COMMAND-LINE ISQL, the traditional SQL interface to the database server, has just about seen its day. Most products now ship with menu-driven tools for setup, tuning, and administration. Some sport sophisticated Microsoft Windows–based interfaces, a trend that will be widely copied in the coming year.

and Watcom, remains somewhat controversial because the operating system, database, and any server-based utilities all run at Ring 0, the most privileged level of the Intel 386 protection scheme. In practice, we found NetWare to be problem-free once properly set up. Ring 0 operation is also extremely fast.

SCO Unix, chosen by Informix, is now a mature, highly stable product. Its only drawback is that it demands solid expertise on the administrator's part. OS/2, which was not chosen by any of this year's entrants, seems to be lagging in popularity as a SQL server OS.

ROBUST FEATURE SETS

This year marks the first time that relational database technology on the PC can be considered a broad success. All the tested products showed solid transaction-processing technology, the core of any SQL database. Log managers and lock managers all functioned smoothly, and we saw none of the instabilities, crashes, and data loss that plagued last year's lineup. These products offer what mainframe users take for granted: the ability to recover from a total system shutdown with data integrity intact.

We also saw a clear trend toward converging feature sets where many formerly cutting-edge features are fast becoming commodities. All the reviewed products except Watcom SQL and XDB-Enterprise Server support both triggers and stored procedures, and both Watcom and XDB will add them in upcoming releases. All the reviewed products support storage of binary large objects (BLOBs) in the database. All but XDB-Enterprise support two-phase commit, although only Oracle7 supports it transparently.

There may be not-so-subtle differences in how common features are implemented, however. For example, while all the reviewed products except Microsoft SQL Server support declarative referential integrity, XDB-Enterprise Server offers the most flexible approach. Child records can be automatically updated by changes to a parent record (a feature called *cascading update*) or deleted if the parent is dropped (*cascading delete*). Oracle7 supports cascading deletes but not cascading updates. Informix OnLine, Sybase SQL Server, and Watcom SQL take another tack, simply restricting any operation that tries to remove a parent with references to existing children. Comparable differences exist among implementations of triggers, stored procedures, and two-phase commit.

SPEED LIMITS

How fast a database delivers your data is always a big concern. While this year's performance results look lower than last year's because of our revamped tests and larger test database, performance has actually improved, in some cases significantly. Part of the reason is cost-based optimization, now used by all the reviewed products. Another is the maturing of lock and cache managers.

As giant applications strain the limits of existing servers, the next performance horizon is clearly the use of symmetric multiprocessing (SMP). Intel-based SMP hardware is becoming more common and under optimal conditions can deliver doubled performance when CPU and disk resources are doubled (for more information, see the sidebar "Intel-based SMP: How Strong?").

One performance question left unanswered in last year's story was how well Intel-based SQL servers stack up against heavyweight RISC platforms. In tests using Sybase System 10 that pit five high-end RISC servers against an Intel-based SMP server, we saw the Intel system

• Oracle7 Server for NetWare, Version 7.0.16

Designing a SQL database server is a tremendous challenge. It must provide the safety and integrity of a mainframe. It must be fast, robust, and above all, completely stable. Our Editors' Choice award for SQL database servers goes to Oracle7 Server for NetWare, the product that comes closest to meeting the ideal. Oracle7 is a virtual compendium of the industry's best features, and its solid core technology, including a multiversioning consistency model and row-level locking, give the product a clear performance edge. It ran friction-free through our punishing test suite, finishing in first place in most categories. Its impressive scores were obtained with almost no tuning. Oracle7 ships with a strong suite of administration tools and is well suited for distributed databases. Oracle7 demands deep pockets and professional administration skills, but it remains the overall best choice in high-stress transaction-processing environments.

An honorable mention goes to Microsoft SQL Server for Windows NT. Its strong performance, superb graphical administration tools, easy setup, and tight integration with the Windows NT operating system make for a compelling package. Significant enhancements to the base kernel include implementation of native Windows NT threads, making the package SMP-ready out of the box. Virtually everything needed for success is included in this attractive bundle.

Rounding out the top three contenders, Sybase SQL Server for NetWare delivered superb performance and the benefits of Sybase's sophisticated, feature-rich engine. We look forward to the release of Sybase System 10 add-ons for this product.

Our Contributors: BRIAN BUTLER, who directed testing for this story, is the president of Client/Server Solutions, a St. Louis–based firm specializing in SQL database performance testing and applications development. LORI MITCHELL is an associate project leader, and KASON LEUNG and ANATOLIY NOSOVITSKIY are technical specialists at Ziff-Davis Labs. THOMAS MACE was the associate editor in charge of this story, and MARK JONIKAS was the project leader at Ziff-Davis Labs.

clearly holding its own (for details, see the sidebar "Competing with RISC").

SHINY NEW TOOLS

The days are gone when simple command-line Interactive SQL (ISQL) tools were the state of the art. Administrators increasingly expect a bundled set of menu- or GUI-based tools for database creation, administration, and tuning chores. Long-established vendors such as Informix, Oracle, and Sybase are playing catch-up while Microsoft, with its experience in application interfaces, is a clear leader in sophisticated Windows-based tools. Oracle has shipped Windows-based administration tools with its Workgroup Server product (not reviewed here) and Sybase has Windows tools in beta testing. Watcom and XDB will move to GUI tools in future releases.

SIMPLER PRICING

SQL database prices are clearly on a downward trend, driven by new packages aimed at the departmental and workgroup markets. Pricing models have also gotten simpler. Where most vendors used to charge separately for users, client software, and networking components, all of the products in this story except Informix OnLine are priced on a per-user basis. While prices for the reviewed products vary widely, a price/performance analysis shows most products deliver similar bang for the buck (for details, see the sidebar "The Price of Performance").

Features, price, and performance can create daunting choices. Despite the hurdles, the news is good: PC-based SQL has never been stronger. The reviews that follow will help you find the database best suited to your needs.

Suitability to Task: SQL Databases

SQL databases were created for huge mainframe applications, but in today's PC-based client/server incarnations, they find employment in a wide range of tasks. Qualities that shine in one area can be drawbacks in another, and products that are tuned to excel in certain operations may falter elsewhere. Taking feature sets and test performances into account, we examine each reviewed product from four different perspectives.

For **production OLTP** applications, a database must be absolutely stable and offer excellent multiuser performance, a function of its locking model, cache management, and transaction-log management. We also look for support for triggers, stored procedures, declarative referential integrity, and on-line backup. Support for transparent two-phase commit and symmetric multiprocessing hardware are a plus. Strong scores on our Random Write Transaction Mix test contribute substantially to the rating.

SUITABILITY TO TASK	Company Name
Production OLTP	EXCELLENT
Decision support	GOOD
Workgroup database	FAIR
Connectivity & deployment	POOR

To judge suitability for **decision support** applications, where large volumes of data are regularly moved to a decision-support server for analysis, we look for excellent loading, indexing, and ad hoc query performance. We also look for an efficient cost-based optimizer. Engine support for bidirectional scrollable cursors is a plus for easing development of GUI-based applications. Compatibility with industry-standard mainframe databases earns additional points.

Workgroup database servers frequently exist outside of an IS framework and pose a different set of demands. Here, we look for easy installation, ease of use, few tunables, and high-quality documentation that does not assume expert knowledge on the reader's part. A strong set of visual administration tools is a plus, as are an overall low initial acquisition cost and a good price/performance ratio. Databases that demand professional administration skill and intimate knowledge of the underlying operating system did not fare as well in this category.

Connectivity and deployment affect many other tasks. Here, we look for support for a wide range of network protocols and client environments. Server support for multiple concurrent protocols earns extra points, as do strong tools for monitoring and tuning the network, the operating system, and the database itself. We also look for a good selection of precompilers, a well-documented call-level C API, and the availability of gateways and connectivity products, either from the vendor or from a third party. A robust set of intuitive, GUI-based administration tools is a plus.

Informix Software Inc.

• Informix OnLine for SCO Unix

ALL REVIEWS BY BRIAN BUTLER AND THOMAS MACE

Last year's roundup of SQL databases wasn't easy on Informix. While Informix OnLine for NetWare offered many strong features, it was plagued during our multiuser tests by numerous crashes.

This year, we looked at a major new release on a different platform: Informix OnLine for SCO Unix, Version 5.01. The bugs are gone, testing ran smoothly, and the product's feature set has been significantly enhanced. Informix OnLine's performance scores, which fell in the midrange of the review lineup, are comparable with last year's results. But its high price—more typical of the Unix world than of the competitive PC marketplace—gave it the poorest price/performance ratio of any product in the lineup.

We caught Informix OnLine right before a major new 6.0 release that will address a number of performance issues. The version we reviewed in last year's story, Informix OnLine for NetWare, Version 4.1, has not been upgraded, and the company has no plans to bring it up to date with its Unix cousins.

Informix OnLine has long offered a robust set of engine features, including a cost-based optimizer, engine-driven backward-scrollable cursors, cursor context preservation, mirroring of databases and transaction logs, and on-line backup.

NEW TO THIS RELEASE

The new release adds a number of features that are quickly emerging as industry standards. These include stored procedures (which can return multiple rows), triggers (added in Release 5.01), and declarative referential integrity (Restrict only). Restrict ensures that a user cannot delete parent records that have dependent child records. Automatic deletion of child records is not supported and must be coded using triggers. The database also supports entity integrity by enforcing acceptable data values (including default values) for particular columns. This release does not support group-level security or audit trails, however.

Informix has enhanced the cost-based

Preview: *Borland's InterBase*

By Brian Butler and Thomas Mace

Borland International hasn't exactly been a leader in client/server databases, but the company is staking much of its future on a push into the client/server arena. While Borland's desktop databases and development tools will figure in this strategy, the cornerstone will be The Borland InterBase Workgroup Server, Version 4.0, a SQL database server due for release on a number of platforms this fall. Releases on Microsoft Windows NT and NetWare should be out by the time this article appears.

InterBase, created by InterBase Software Corp., is a technically advanced engine that found an early niche in the on-line complex processing (OLCP) market because of its pioneering support for features such as multiversioning, BLOBs, and multidimensional array data types. But the product languished after its initial sale to Ashton-Tate and up until now has seen little growth under Borland (at press time, InterBase 3.2 was the currently shipping version).

Our look at an early beta of the new InterBase, Version 4.0 for NetWare, revealed an enhanced product repositioned as an upsizing tool. The biggest change is that InterBase can now interface directly with Borland's desktop databases dBASE 5.0 for Windows and Paradox 5.0 for Windows.

STRONG CORE ENGINE

InterBase offers a strong core set of features that includes declarative referential integrity, triggers, stored procedures, event alerters, user-defined functions, a cost-based optimizer, BLOB support, and transparent two-phase commit. InterBase is based on a multiversioning database engine, an approach it shares with Oracle7.

Multiversioning provides transactions with a read-consistent view of the database: A given transaction sees the database as it was at the moment the transaction began, and multiple transactions can see the database in several different consistent states. The main advantage to multiversioning is that read transactions, especially long-running reads typical of decision-support applications, do not acquire locks that block write transactions, improving overall concurrency.

THE DESKTOP CONNECTION

InterBase offers a unique synergy with Borland's desktop database products dBASE for Windows 5.0 and Paradox 5.0 for Windows. Both can connect directly to InterBase, which in turn provides direct engine support for the desktop products' native record-navigation commands (in addition to InterBase's support for standard SQL). This lets developers migrate dBASE and Paradox apps to a client/server environment or use these PC databases as front-end development tools.

> *Borland is staking much of its future on a push into the client/server arena.*

dBASE for Windows and Paradox for Windows share a common local database engine called the Borland Database Engine. This includes an IDAPI (Independent Database API) component for connecting to InterBase and other SQL databases. The IDAPI InterBase driver, called Client/Server Express, provides the direct low-level interface to InterBase. Since InterBase directly supports dBASE's and Paradox's record navigation, you can use commands such as dBASE's Skip -1000 and Go Bottom with InterBase data. The IDAPI component also includes Borland's SQL Link drivers for connecting to Microsoft SQL Server 4.21, Oracle7, Sybase SQL Server 10.01, and ODBC-compliant databases. These drivers also let you use dBASE or Paradox commands with third-party SQL databases, but only through a potentially slower SQL translation. The ODBC component of SQL Link will also let you develop applications for InterBase using non-Borland tools.

PACKAGING

In addition to the pending Windows NT and NetWare versions, ports for DEC Alpha OSF1, HP-UX, Sun OS, and Sun Solaris are scheduled to ship this fall. A Chicago version will ship soon after Microsoft's release of Chicago, and an OS/2 version is due by early next year. Borland also plans to bundle a version of Delphi (the code name for its upcoming Visual Basic competitor) for use in off-line applications development. Pricing is expected to be highly competitive, and client software, including the SQL Link and Client/Server components, ODBC drivers, a set of Windows-based administration tools, and the InterBase C API libraries, will be bundled free with every server.

A few holes remain in the current InterBase strategy. The product cannot operate with the new dBASE for DOS 5.0, so migrating existing dBASE apps to InterBase means porting them to Windows. Also, many of the engine's most advanced features are only accessible through proprietary interfaces. But InterBase's tie-in with Borland's Windows databases is compelling. dBASE and Paradox developers will certainly want to evaluate the product when it ships. □

optimizer to be more intelligent about its choices, and it now lets you set the optimization level of the query. The default is High Optimization, which performs an exhaustive search through all possible access plans and picks the one with the lowest cost. With complex queries involving many tables, this process can be more expensive than the actual execution of the query. In such scenarios, you can select Low Optimization, which will make a quick best guess.

The optimizer did not make any mistakes during our tests, but we did run into a problem updating the optimizer statistics. A bad value placed in the statistics page caused two subsequent queries to crash the server. This was fixed by modifying the statistics page manually.

The new release has also improved Informix OnLine's index-creation speed. Last year, it was the slowest product at indexing our test database by a huge margin. This year its indexing score, while not exactly zippy, was more in line with other competitors'. Under Informix's new indexing scheme, index entries are sorted prior to their insertion into the B+tree structure.

The Informix OnLine engine has always offered strong binary large object (BLOB) support. As with the previous version of the product, BLOBs are stored in a distinct Blobspace, allowing you to tune the associated page size separately for best performance. The maximum allowable BLOB size is 2GB. BLOBs are written directly to disk, not to shared-memory data buffers. This saves space in the transaction logs and keeps the pool of shared-memory data buffers from being swamped. With the optional Informix-OnLine/Optical add-on product, BLOBs can be stored on WORM (write-once-read-many) optical subsystems. Unfortunately, we did not get a chance to test Informix OnLine's BLOB throughput capabilities because of time constraints in our test cycle. This was not due to any problem with the product.

Informix OnLine provides locking by row, page, table, or database and the unique ability to configure locking on a table-by-table basis. You can have one table set for a page-level locking scheme, another with record-level locks, and yet another large lookup table set for table-level locking. Isolation levels are also highly tunable and include support for dirty reads (no isolation), committed read isolation, cursor stability, and repeatable reads.

SUITABILITY TO TASK

Informix OnLine for SCO Unix

Production OLTP	GOOD
Decision support	FAIR
Workgroup database	POOR
Connectivity & deployment	GOOD

Version 5.0 added support for distributed Informix OnLine databases through the separate Informix-Star product. Informix-Star adds a two-phase commit protocol and lets users transparently manipulate multiple Informix OnLine databases at several locations. The current lets you update multiple databases on a single Informix OnLine server instance in a single transaction.

KNOW YOUR UNIX

While database administration does not usually call for much knowledge of the underlying operating system, this version of Informix OnLine demands a good working knowledge of Unix. During installation, we had to modify some SCO kernel parameters to get the package up and running. This process is documented in the machine notes file on the system.

Like most Unix database vendors, Informix recommends that you set up raw file partitions, a task that can be a tricky process. Because the Unix file system has its own cache, the database has no way of ensuring that writes have been physically committed to disk. This can lead to serious integrity problems if the system crashes. Using raw file partitions bypasses the Unix file system, the only way to ensure integrity loss doesn't happen.

Once the database is installed, you can use the supplied DB-Monitor utility to configure various system parameters including buffers, locks, users, and tables. This menu-driven utility also lets you change the server's mode of operation to on-line, off-line, or quiescent mode (a single-user administration mode). DB-Monitor also provides backup, recovery, and a window into virtually everything the engine is doing. It can display a multitude of statistics to aid in the tuning process, such as cache hits, disk reads and writes, and checkpoints.

The database also ships with a menu-driven setup tool called DB-Access for creating databases and tables and executing SQL statements. A set of command-line utilities, which can be driven by scripts, provides additional administrative functions.

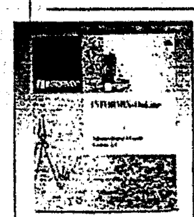
COMING DOWN THE PIKE

We narrowly missed the next major release of Informix OnLine, Version 6.0, which should be shipping on the SCO Unix platform by the time this story appears. Where the 5.0 release is generally targeted at broadening engine functionality, Version 6.0 is primarily aimed at boosting performance.

Informix has rebuilt large portions of the database server, replacing the current process-based engine with an internal multithreaded system. The most important change will be the ability to exploit symmetric multiprocessing hardware through the addition of parallel index creation, parallel thread-level sorts, and parallel backup and restore capabilities. An upgraded 6.0 optimizer will be able to maintain data-distribution histograms. Declarative referential integrity support will be extended to cover cascading deletes.

FACT FILE

Informix OnLine for SCO Unix, Version 5.01



List price: Server software, one development system, 60 client connections, and client software: $29,395. **Requires:** Server: 386-based PC or better, 2MB RAM, 5MB hard disk space, SCO Unix System V 3.24 or later. DOS client: 286-based PC or better, 700K RAM, 3.2MB hard disk space. **In short:** Version 5.01 of Informix OnLine adds significant enhancements to this veteran Unix database. New features include triggers, stored procedures, and declarative referential integrity. Unix knowledge is required, but the bundled administration tools make for easy setup and tuning. Informix OnLine ran smoothly through our benchmark tests, although its performance scores remain in the midrange. Its high price gives it the worst price/performance ratio of the roundup. By the time this story appears, a new Version 6.0 should be available that offers major performance enhancements.

Informix Software Inc., 4100 Bohannon Dr., Menlo Park, CA 94025; 800-331-1763, 415-926-6300; fax, 913-599-8753

481 on reader service card

▪ Preview: *Gupta SQLBase Server*

By Brian Butler and Thomas Mace

Gupta SQLBase Server for NetWare was the worst casualty of last year's testing. Version 5.12 took almost 60 hours to load our database—more than 10 times as long as the next-slowest competitor—and crashed repeatedly on index builds. Testing never got beyond this point.

As we go to press, Gupta is about to ship SQLBase Server for NetWare, Version 6.0—a major new release that will extend the server's feature set and target the problems we encountered. We invited Gupta to run through our Load and Index and Ad Hoc Query tests using a beta of this upcoming release. Testing was done at Gupta Corp. on a Compaq ProLiant configured similarly to our test-bed.

Loading and indexing ran without a hitch, even though our test database is more than four times as large as last year's. Total load-and-index time was also significantly faster, placing SQLBase within reach of other products in this story (although it would still have placed last). SQLBase also ran smoothly through our ad hoc queries and demonstrated times that were reasonable but again were not as fast as the times posted by the other tested products.

PC-CENTRIC

SQLBase was designed from the ground up as a small-footprint engine for PC-based client/server computing. Some of its slickest features, such as bidirectional scrollable cursors, are especially at home in Microsoft Windows–hosted applications.

The new release fleshes out the feature set with stored procedures, triggers, and timer events. SQLBase stored procedures are written in the SQLWindows Application Language (SAL), providing close ties with Gupta SQLWindows—Gupta's well-known front-end development tool. You can specify whether the new SQLBase triggers fire before or after the triggering operation. Timer events are stored procedures that can be set to execute at a specific time or at predetermined intervals.

SQLBase 6.0 shows enhancements to usability as well. New utilities can automate installation: the new SQLEdit utility, a particularly welcome breath of fresh air, automatically handles network configuration for both clients and servers. In previous versions of SQLBase, you had to edit the SQL.INI file manually—a confusing, tedious process.

Support for distributed databases has also been strengthened. SQLBase 6.0 will offer transparent two-phase commit to manage transactions across multiple servers. SQLConsole 2.0, an impressive new Windows-based remote-management utility, will be bundled with the server. This slick tool allows remote tuning, monitoring, and maintenance of multiple servers through its Manager modules.

The Scheduling Manager automates such maintenance tasks as backups. The Alarm Manager monitors the network for more than 20 definable events and automatically executes an appropriate response when necessary. If the event remains unresolved, the alarm can trigger further responses. The Database Object Manager lets you graphically manage every database component, including stored procedures and triggers. SQLTrace is a debugging tool that can trap SQL traffic between a client and the server. You can replay the SQL through SQLTrace's graphical debugger.

Gupta appears to have made great strides with SQLBase 6.0, addressing stability, performance, ease of use, and administration in a single release. □

An even newer release, Version 7.0, was codeveloped with Sequent Computer Systems and is already shipping on Sequent's Symmetry multiprocessing platform. This version adds parallel data query (PDQ) capability and the ability to optimize the partitioning of tables based on the contents of the data.

Informix has long been known for its strong Unix databases, but it has been less active in client/server products for the PC environment. This seems to be changing. The current release, while no screamer, shows major improvements over previous versions. The pending 6.0 release seems poised to add the missing element of top-flight performance.

Microsoft Corp.

• Microsoft SQL Server for Windows NT

For those who've wrestled with mix-and-match client/server environments, Microsoft SQL Server for Windows NT, Version 4.21, offers all the seductions of one-stop shopping. For a very competitive price, it delivers a powerful SQL engine, superb tools, strong networking components, and the benefits of close integration with the Microsoft Windows NT operating system—all in a single box.

There are some caveats, particularly for enterprise applications. The server is not ideally equipped for distributed environments and does not support replication. Its performance, while generally very fast, was surprisingly slow in a few areas critical for decision support. More fundamentally, Microsoft is clearly steering you into a total Windows NT solution, something that may be incompatible with larger enterprise strategies. But for workgroups and larger departments—even those with heavy transaction loads—Microsoft SQL Server is a compelling solution.

MAJOR REWRITE

Although Microsoft SQL Server had its genesis in Version 4.2 of Sybase SQL Server, Microsoft significantly rewrote many important system components for the current release. While the changes are largely targeted at improving integration with Windows NT, they also fixed a few idiosyncrasies and made some important enhancements to the engine. While Microsoft has been careful to preserve full compatibility with the older Microsoft SQL Server for OS/2, Version 4.2, both Microsoft's and Sybase's SQL Server products are now clearly headed in different directions. The only official compatibility between them is at the 4.2 level of DB-Library.

The level of integration between Microsoft SQL Server and Windows NT is high. Microsoft discarded the internal threading engine used in its OS/2 product and has implemented Microsoft SQL Server as a single process using native Windows NT threads. Threads are preemptively scheduled and can be distributed over multiple processors, making Microsoft SQL Server SMP-ready out of the box.

The database also uses Windows NT's asynchronous I/O capabilities to handle physical inputs and outputs concurrently with other operations. As a whole, the database runs as an operating system service that can be started, stopped, or paused from the Windows NT Control Panel. Windows NT also lets Microsoft SQL Server simultaneously support multiple network protocols and connection types, including IPX/SPX, Named Pipes, NetBEUI, sockets, and TCP/IP. Server-based gateways to other databases can be written through the Microsoft Open Data Services (ODS) API. Backups are handled via Windows NT's backup facility. You can dump multiple databases to a single device and schedule on-line backups. Microsoft SQL Server supports any backup devices that are also supported by Windows NT.

Microsoft SQL Server integrates with the Windows NT Performance Monitor to provide a graphical display of database, network, operating system, and hardware performance data such as CPU utilization, I/O activity, cache hits and misses, database users, and network connections. This window into a client/server system takes much of the guesswork out of performance tuning and provides a firm guide when making hardware modifications. The Performance Monitor also lets you

Coming Soon: *A New DB2/2*

By Brian Butler and Thomas Mace

If anyone knows databases, it's IBM. That's why last year's look at IBM's OS/2 database—DB2/2, Version 1—was such a disappointment. DB2/2 proved stable in testing, but it was extremely slow and lacked basic amenities such as the ability to span a database across multiple logical volumes. NetBIOS was the only supported network protocol, and though you could connect to a DOS client, there were no available DOS development tools (all development had to be done in OS/2). A point revision, Version 1.2, improved network connectivity and added ODBC support and DOS client development tools but failed to address other shortcomings.

An important new release, Version 2.0 of DB2/2 is poised to energize this hitherto stodgy product. Enhancements will include a totally revamped query optimizer, flexible tablespace allocation, and a host of new engine features. Version 2.0 will be entering beta testing in the fall and should be generally available early next year. This release will be available for both OS/2 and Microsoft Windows NT.

STARBURST OPTIMIZER

New optimizer technology is high on IBM's list of enhancements. DB2/2's new cost-based optimizer, called Starburst, has been developed by some of the research-team members responsible for IBM's pioneering System R, the original prototype of DB2. IBM claims that Starburst will be the most advanced optimization technology on the market—more advanced than the mainframe version of DB2, with which it will share some technology. An accompanying visual tool will show a graphical representation of the access plan chosen by the optimizer for assistance in tuning. A graphical performance monitor will also be included.

Version 2.0 of DB2/2 will also address the previous one-volume limitation on database size by letting you divide a database into separately managed tablespaces. You will now be able to specify where tables or indexes are created by specifying the tablespace in which they reside. On-line backup capability at the database or tablespace level will also be added.

The new release will bring the engine feature set up to date by adding user-defined functions and data types, triggers, constraints, recursive SQL, and BLOB support. The DB2/2 engine has also been rewritten to support native operating-system threads, making it SMP-ready. While DB2/2 will not have its own server-based 4GL, you will be able to write 3GL stored procedures as DLLs. In addition, the new version will include Distributed Relational Database Architecture (DRDA) Server capability. (The previous release was a DRDA Requester only.) Two-phase commit will be supported.

IBM will also be releasing a set of data-replication products that support replication from multiple sources—including DB2/MVS, DB2/400, IMS, and VSAM—into DB2/2 and DB2/6000 databases. On the networking side, Version 2.0 will add support for TCP/IP.

DB2/2's upcoming feature set looks strong. If IBM delivers performance to match, this product will be a force to be reckoned with. □

How We Tested

Our demanding tests revealed improved performance across the board. Oracle7 took first place overall while Sybase led the pack in multiuser read transactions. Microsoft SQL Server delivered strong results everywhere except in ad hoc queries and load-and-index operations. Watcom SQL and XDB-Enterprise brought up the rear.

To evaluate the SQL relational database management systems in this roundup, we used a heavily modified version of the AS^{3}AP (ANSI-SQL Standard Scalable and Portable) Benchmark Tests for Relational Database Systems, originally developed at Cornell University by Dina Bitton and associates. This set of cross-platform performance tests covers a wide spectrum of typical database operations (although based on ANSI SQL, the AS^{3}AP tests are not an ANSI benchmark.)

For our database server, we used a Compaq ProLiant 4000 equipped with a single 66-MHz Pentium processor card, 128MB of ECC RAM, five 2.1GB Hewlett-Packard disk drives in an external cabinet, four Compaq EISA NetFlex-2 network adapter cards (configured for Ethernet), and a Compaq Smart SCSI-2 Array disk controller. Compaq's hardware RAID 0 striping was available to vendors if they chose to use it. On the client side, we used a network of 60 physical clients comprising a mix of 386- and 486-based machines. All clients were equipped with 8MB of RAM and an NE2000 network card. The network was divided into four segments (15 clients per segment); each segment communicated with a separate network card on the server. The Ad Hoc Query test workstation was a 486/33 PC equipped with 8M of RAM and an NE2000 network card.

All vendors were invited to Ziff-Davis Labs to observe testing and help us tune the database engines. Among the vendors whose products we reviewed, only Informix declined to send a representative. To give Informix equivalent representation during testing, ZD Labs hired Gregory D. Balfanz, an Informix consultant and Unix specialist from Open Systems Engineering of Boerne, Texas, to help us tune the Informix database.

Vendors were allowed to run their products under their choice of Intel-based operating systems and network protocols. Informix Software chose to run its Informix OnLine for SCO Unix 5.01 under Santa Cruz Operation's SCO Version 4.2 using TCP/IP. Originally, Informix chose to run Version 5.02 of the server, but after we encountered a memory-leak bug during our Load testing, the company substituted its 5.01 release. Microsoft ran Microsoft SQL Server for Windows NT 4.21 on Microsoft Windows NT Advanced Server 3.1 using Named Pipes on top of NetBEUI. Oracle Corp. ran Oracle7 Server for NetWare 7.0.16 on NetWare 3.11 using SPX/IPX. Sybase chose to run Sybase SQL Server for NetWare 10.01 on NetWare 3.12 using TCP/IP. Watcom International ran Watcom SQL Network Server for NetWare 3.2 on NetWare 4.01 using SPX/IPX. Finally, XDB Systems ran XDB-Enterprise Server 4 for Windows NT on Windows NT Advanced Server 3.1 using TCP/IP. The Windows NT products applied Service Pack 2 to the operating system. The client-side TCP/IP stack was FTP Software's PC/TCP Plus 2.3.

Our test database consisted of ten tables containing a total of 18.61 million rows. The breakdown of the table sizes was as follows: one table with 7 million rows, one table with 5 million rows, one table with 2 million rows, four tables with 1 million rows each, one table with 100,000 rows, one table with 10,000 rows, and one table with 5,000 .GIF images. We also created two empty tables used for inserts. The database size typically ran well over 2GB when fully loaded and indexed.

The raw data for our test database was generated using the AS^{3}APGen 2.0 program from Dina Bitton and Jeff Millman at DBStar of San Francisco, California. All the tables had the same structure, and each row was approximately 160 bytes long, although the exact values varied by vendor. The test data for each table was supplied in the form of an ASCII comma-delimited file. The data types in the database columns included integer, floating-point, and date, as well as fixed-length and variable-length character strings.

The multiuser tests were automated using the Benchmark SDK utility from Client/Server Solutions of St. Louis, Missouri. All of our multiuser tests measure total system throughput—the amount of work that the system is performing every second—calculated in transactions per second (tps). We generated tps scores using 11 different client-load levels ranging from 1 to 60 simultaneously active network clients.

Beginning with a single client, we ran each client level for 10 minutes. Scores for the first 3 minutes 45 seconds were discarded to allow the database cache to stabilize. During the next 5 minutes, we counted the number of transactions executed. This was followed by a rampdown interval of 1 minute 15 seconds, during which no measurements were made. Before moving to the next client level, we added a 30-second quiet period to allow the network to settle. This overall approach allows us to guarantee accurate and consistent scores. Transactions are processed as quickly as the database allows; test code does not include think time. This generates a workload far greater than 60 real-world clients would produce.

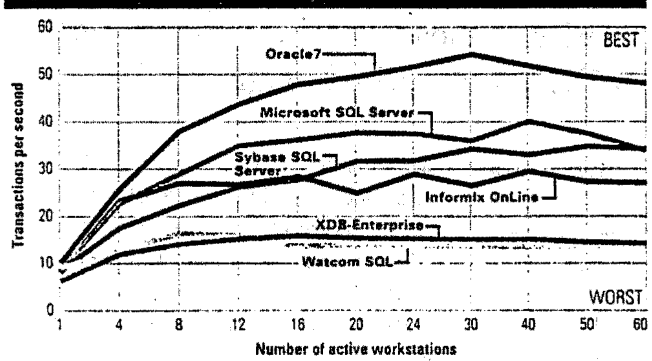
WEIGHING THE RESULTS

This year's testing was based on a significantly larger test database than last year's, and our test queries were considerably more demanding. As a result, this year's raw scores are considerably lower than last year's, despite the use of Pentium-level server hardware and 32MB additional server RAM. The best comparison with last year's results is provided by the Single Random Read test, which was not redesigned for this year's testing. Even assuming that the hardware used this year is twice as fast as last year's (and discounting the larger test database size), we still saw improvements of between 15 and 270 percent.

In general, it is important to realize that a benchmark testing scenario can bring optimizations into play that may not be fully exploited in real-world situations. A good example is the issue of manually striping the database across multiple disks versus using the hardware to stripe it. In a benchmark situation, a vendor can often achieve optimal performance by manually placing the database objects on the disk subsystem because the transactions and access methods are very well defined. Given enough time and intimate knowledge of the database engine, the vendor can find an absolutely optimal balance of inputs and outputs across the disk drives.

This type of optimization is usually achieved through trial and error, which is costly in both time and resources. In the real world, it is very rare that the administrator has comparable knowledge of data access and the database engine—let alone the time for experimentation. Informix and Sybase chose to stripe the database manually, while Microsoft, Oracle, and Watcom used hardware-level striping. XDB chose not to use hardware striping but was able to achieve significant optimization by experimenting with placement of tables and indexes on the disk array.

RANDOM WRITE TRANSACTION MIX

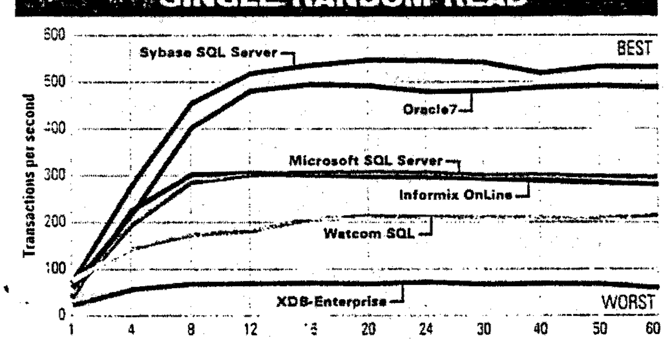


Below we describe the results we observed and attempt to explain these results in terms of each product's features. SQL databases are extremely complex artifacts, and it may not always be possible to isolate the many interrelated factors contributing to observed behavior.

The **Random Write Transaction Mix** test, which accesses six tables in our database, simulates a heavy mixed workload of read and write transactions. This test simultaneously stresses Delete, Insert, Select, and Update functions of the database server. During the test, each station randomly selects and then executes a series of queries from a pool of five possible query types. The randomizer is constructed so that the frequency of execution for query types numbered one through five will be in a ratio of 6:4:4:3:3.

The first transaction updates an integer field in the 7-million-row table via the primary key using a Between operator. The second transaction is a two-way join between two 1-million-row-tables. The third transaction updates an integer field in a 1-million-row table and includes some in-line logic that stores the update in one of the blank tables. The fourth updates the 2-million-row table via an In clause, and the fifth moves a row from the 5-million-row table to a blank table. We used an extensive auditing script to ensure that all the products were actually performing these tests as specified.

SINGLE RANDOM READ



The best performer on this extremely demanding test was Oracle7. Its high score is attributable to its record-locking scheme and efficient log management, features that have been part of the product for quite some time. The engine ran error-free and required very little tuning to achieve the measured performance level. Oracle7 only logs changes to the data, and its support for fast commit and group commits further reduces log-management overhead. The amount of data Oracle7 can write in a group commit is limited only by the operating system. Record-level locking provided optimal concurrency: No deadlocks occurred during the execution of the test. Oracle7 did not use its Discrete Transaction feature on this test.

Following close behind was Microsoft SQL Server. This test highlights how much work Microsoft has done to speed up the transaction-processing aspect of the database's engine. One new feature, *asynchronous checkpoints*, allows translation processing to continue during the checkpoint process. A Lazy Writer feature, also new to this release, lets the database engine clean up dirty database pages in the background, minimizing the work required during the checkpoint process. Microsoft SQL Server also supports group commits of up to eight 2K pages at a time. A factor working against the product may have been its page-level locking scheme.

Since two of our transactions perform an Insert into an empty table, we created a clustered index to avoid contention on the last page. Without a clustered index, the last page of the table is consistently locked, forcing a serialization of operations. With a clustered index, Insert commands are distributed over the range of the table. The only drawback to this approach is the extra overhead for maintaining the index. We avoided deadlocks by using a fill factor on the affected indexes.

Sybase, which came in third, implemented this test with stored procedures accessed via remote procedure calls (RPCs). The company also coded several of our transactions using its newly added support for cursors within the stored procedures itself. Another new feature of the tested NetWare port is Sybase's Buffer Wash mechanism. This is a background process that cleans up dirty pages, guaranteeing a supply of free pages. While checkpoints are essentially unchanged since Version 4.2, the new Buffer Wash feature means that checkpoints must perform significantly less work. Sybase also supports group commits, but only up to a single 2K page at a time. The company used a fill factor to avoid deadlocks. For this test, we allowed Sybase to modify the database schema slightly to make the update columns Not Null. This allowed the company to work around the engine's limitation on update-in-place for nullable columns.

Informix, which came in fourth, used record-level locking on all tables that were updated. Informix performed well once the database and operating system were properly tuned, although the tuning process was not particularly intuitive. The database was stable in operation and we encountered none of the problems with failed checkpoints that plagued it in last year's tests.

Watcom SQL and XDB-Enterprise brought up the rear. Although Watcom SQL supports group commits and only logs changed data to the transaction log, its performance was only about 25 percent as fast as the fastest product. XDB-Enterprise does not support group commits and logs the entire before-and-after image of the row.

The **Single Random Read** test, based on a single-record read via the primary key, shows the maximum number of concurrent retrievals the system can handle. This test has not been modified since our last roundup and is included to show how far the products and hardware have come in the interim.

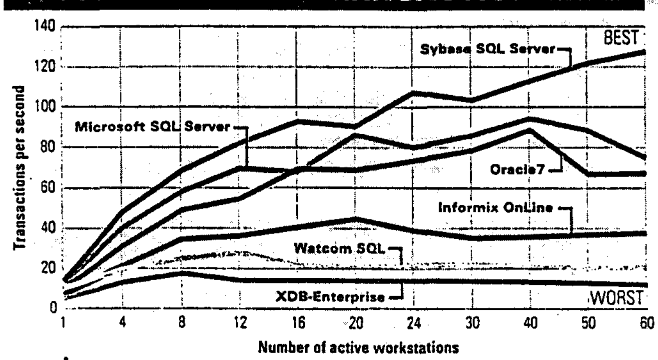
In this test, each workstation selects a random row from a single table that is then fetched across the network and discarded. All active workstations repeat this process at the maximum speed supported by the database. This scenario does not stress every component of a database engine and the results tend to exaggerate the engine's actual transaction-processing power. The small, quick transaction involved does put a significant stress on the network components of the operating system, however. For products tested under Windows NT and SCO Unix—both of which are true protected-mode operating systems—the overhead for privilege-level checking proved to be costly. Records are read in the lowest lock level each product supports, thus permitting the greatest degree of concurrency (we required that each vendor take at least a shared-level lock on the row or page). Since all locks are shared-level, no blocking occurs; multiple clients can access the same row or page without concurrency loss.

The best performer on this test was Sybase SQL Server. Contributing factors are the efficiency of its NetWare Loadable Module (NLM) architecture, Sybase's clustered indexes, and the use of stored procedures. Sybase called its stored procedures via an RPC instead of by using a straight stored procedure call. In an RPC, the function call is translated into a binary representation at the client; a normal stored procedure call is sent across the network as text and translated at the server. Sybase believes that the use of RPCs in CPU-bound situations such as this test can significantly improve performance.

Oracle7, which came in second, did not use a stored procedure due to the simplicity of the transaction. It did open and maintain a cursor, however. Since Oracle7 has the ability to share cursors across multiple clients, this approach allowed clients to execute the transactions without having to reparse and optimize the SQL statement—in effect, the same advantage provided by a stored procedure. Oracle7 also supports a unique method of executing and fetching multiple rows in a single function call, thereby reducing network traffic. Most other products require several function calls to do the job, one to execute the query and others to retrieve the results.

Microsoft SQL Server placed third. It used clustered indexes and stored procedures but did not use RPCs. While clustered indexes usually help Microsoft SQL Server considerably, they were offset by the stress on the operating system's network communications layer.

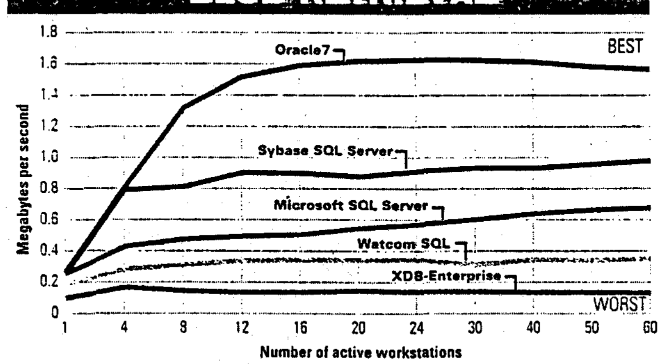
RANDOM READ TRANSACTION MIX



Close behind Microsoft came Informix OnLine. We did not test Informix using its clustered indexes or stored procedures. The opinion of our Informix consultant was that stored procedures would be slower than a prepare/fetch mechanism, and the clustered indexes would have drastically slowed the index creation times.

Watcom SQL came in fourth—earning it the award for being the most improved product since last year. Improved performance was mostly due to the move to NetWare, a true 32-bit environment, and asynchronous I/O, which was not available in the DOS product we tested previously. Watcom does not support stored procedures at this time; a prepare/fetch mechanism was used to avoid the parsing/optimization phase of the query.

BLOB RETRIEVAL



In last place was XDB-Enterprise, which did not use striping. The database table and index used in this test resided on a single drive, something that proved to be the biggest bottleneck. While performance could have improved by moving the index to a separate spindle, this approach would have hurt the product's multiuser test results. This highlights the problem with manually placing database objects: Tuning for one type of operation can hurt performance elsewhere.

The **Random Read Transaction Mix** test, which accesses five tables in the test database, simulates a mixed workload of read-only queries. This test was designed to stress the data-retrieval capabilities of the database engine, and proved to be extremely disk bound. During the test, each station randomly selects and then executes a series of queries from a pool of five possible query types. The randomizer uses the same ratios as for the Random Write test.

The first query is a single record read via primary key (this is identical to the Single Read Transaction test). The second is a Join on the primary key between two 1-million-row tables. The third is a Select on the 7-million-row table using a Between clause. The fourth query is a two-way join between a 1-million-row and a 2-million-row table using a Between as a restriction and a Join on a character field. The fifth query is a two-way Join between a 1-million-row and a 5-million-row table using an In clause; the Join is on a character field.

Sybase SQL Server was the clear winner. Its performance can be attributed to the use of clustered indexes and stored procedures (using RPCs), and the NetWare operating system's low overhead. Sybase SQL Server seemed able to satisfy the transaction requests with less I/O than other vendors, and its stored procedures cut down on network traffic. The package's clustered indexes must also be considered an important contributing factor since two of the queries used a Between clause on a clustered key. Since the data is physically arranged on the disk in clustered order, these queries could be typically resolved in fewer disk inputs and outputs than when using products that do not support clustered indexes. Sybase experimented with using SPX/IPX on this test, but TCP/IP, its original protocol of choice, proved to be slightly more efficient.

Microsoft SQL Server, which also used stored procedures and clustered indexes, came in second. While Sybase and Microsoft coded the test

CONTINUES

ransactions in a similar manner, Microsoft chose iot to use RPCs to call the stored procedures. The verhead of Windows NT may have also played a light role in the performance difference, al-hough, as the transaction size increases, Win-ows NT overhead appears to de-rease.

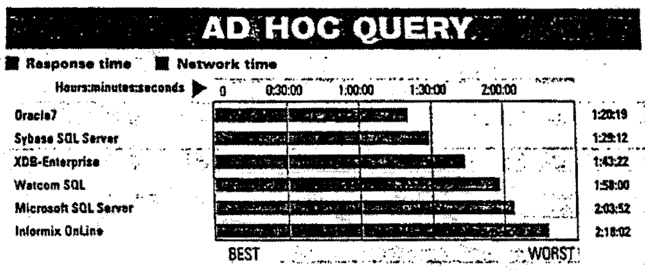
Third-place Oracle7 was able to hare the cursor among the clients, nd its ability to do an execute and tch in one statement also helped erformance. Oracle7 is also unique that it can retrieve multiple rows in single network fetch. But its perfor-iance was bottlenecked on this test y the disk I/O subsystem, something iat could be attributable to nonclus-ered indexes and the nature of the ueries. As an experiment, we added nother five drives and saw tangible nprovements before the system ecame clearly CPU-bound. Other endors may well have achieved omparable improvement.

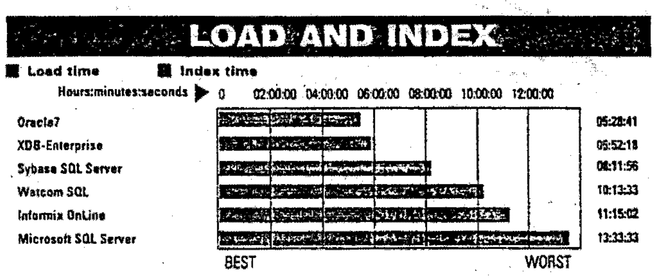
Informix came in fourth. We did ot use the clustered index feature of ie database engine, because of its ffect on the index time. Our Informix onsultant also advised against using iformix's stored procedures since he eels that they are inefficient for our ype of transactions. To achieve optimum perfor-iance, each station used a prepare/fetch mecha-ism, saving the processing overhead of a pars-ig/optimization process. The overhead of SCO nix may also have been a factor.

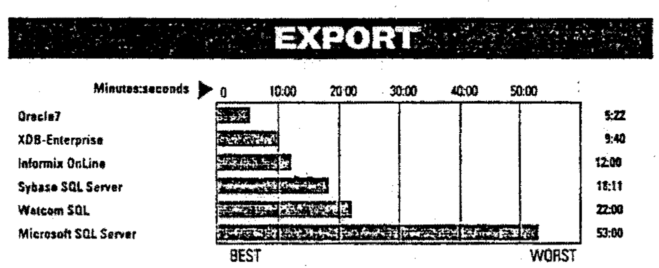
Watcom SQL came in fifth with XDB-Enter-rise pulling up the rear. Watcom SQL did not use lustered indexes to avoid undue load times, and ie product does not currently support stored rocedures. The transactions were coded using a repare/fetch mechanism. XDB-Enterprise's esults may be attributable to the product's ianual distribution of database objects.

Binary large objects (BLOBs) are structures sed for storing images and other large binary elds in the database. The **BLOB Retrieval** test ieasures how fast the client can retrieve these irge structures—in effect, how well the data-ase can utilize the network. All the tested roducts offer a method for fetching large blocks f data in a single network call, and many are able change the default network packet size dynami-ally.

This test used a database table containing 5,000 unique bitmapped images in .GIF format ranging in size from 20K to 150K, with a majority in the 70K range. During the test, clients randomly selected and retrieved a series of images. Images were not displayed, but we required that the full binary file be sent across the network. While the BLOB Retrieval test executes in a similar manner as our other multiuser tests, it measures sus-tained system throughput in megabytes per second (MBps).







Oracle7 was the clear winner with a maximum transfer rate of 1.61 MBps. No tuning of the network packet size was needed to achieve this result. While creating the BLOB table, however, we discovered that Oracle7 was the only product unable to load our BLOB images from a DOS client because the Oracle7 DOS client does not have a mechanism for sending BLOBs piecemeal to the server, and not enough memory could be allocated to load the entire image at once. We used an OS/2 client as a workaround. In experimenting with network protocols, we found that using TCP/IP gave Oracle7 about a 25 percent perfor-mance boost over the SPX/IPX protocol used for official testing (charted numbers show results for SPX/IPX).

Sybase SQL Server came in second with a maximum transfer rate of just under 1 MBps. The package's DB-Library does not support negotiated packet size, so the default packet size of 512 bytes was used. (The company's Open Client does support negotiated packet size, but it is currently available under Windows only.) We experimented with substituting SPX/IPX for the TCP/IP protocol Sybase chose for official tests. We observed a 45 percent performance degradation under SPX/IPX (charted numbers show results for TCP/IP).

Microsoft SQL Server came in third with a maximum transfer rate of 0.68 MBps. Microsoft tuned for this test by increasing the default packet size from 512 bytes to 4K. The negotiated packet size feature allows clients to configure the packet size at connection time. This feature is only supported under Named Pipes.

Watcom SQL placed fourth with a transfer rate of 0.35 MBps, and XDB-Enterprise was last with a transfer rate of 0.17 MBps. Watcom SQL also lets you specify the packet size when the DOS requestor is started. Watcom used a packet size of 1,450 bytes for our entire suite of tests (the default packet size is 512 bytes). While XDB-Enterprise used TCP/IP, the network was not an overriding factor in the package's performance. Since all of XDB-Enterprise's BLOBs resided on a single disk, the server remained strongly disk-bound throughout the test.

We could not obtain results for Informix OnLine due to time constraints. This was not due to any fault in the product.

The **Ad Hoc Query** test measures each product's effectiveness in a decision-support environment. The query mix is submitted from a single 486/33 client, and both the response time (the time for the first row to be returned) and the total elapsed time for each query are recorded. Response time is an important metric in a real-world environment in which the user is waiting to see results. Once the first row is returned, the user can begin scrolling through the data. Total elapsed time is more important in a batch-report-ing environment in which large reports are being printed.

Because of the large number of rows returned by some of our queries, network overhead is in some cases the factor limiting performance. It is also difficult to separate engine processing speed from network overhead since many products return rows to the user before the query is completely resolved.

The Ad Hoc Query test consists of 34 queries that stress six different types of server functions: selects, joins, projections, aggregates, sorts, and subqueries. Ten select queries measure the speed at which a database can selectively scan a table. Nine join queries show how well an optimizer can pick the fastest access path from the available indexes (the joins range from a two-way to a seven-way join). Two projection queries measure how fast a database can determine the number of distinct values in a table. Five aggregate queries calculate a variety of aggregates (minimum, maximum, average, and count). Five sort queries measure how fast the database can sort data sets ranging in size from 10,000 rows to 2 million rows. And finally, three subqueries show the effectiveness of the optimizer in resolving correlated subqueries and outer joins.

Oracle7 takes the top spot on this test with a total time of 1 hour 20 minutes. But when we first ran the test, Oracle7's optimizer made a mistake on the sort query, returning a score of over 10 hours, by far the worst score we saw. The optimizer chose to use an index when it should have performed a table scan. This entailed extra I/O in jumping between index pages and data pages and did not let the database take advantage of its read-ahead mechanism. This error was easily corrected using a Hint, a well-documented method of overriding the optimizer. Because all optimizers are based on statistics, there is always a probability of making a mistake. Consequently, an override mechanism is a must.

While the Oracle7 optimizer was not the most robust we saw on our 34 test queries, the currently shipping Oracle7 Server for SCO Unix, Version 7.1, was able to execute the same queries without hints, indicating that the problems have been addressed.

Interestingly, Oracle7 took second place in both response time and network time, yet the combination of the two made it the fastest. Oracle7 offers a way to tune the query for response time or total time. Due to the nature of our queries, the company chose to tune for total time.

Sybase SQL Server was close behind Oracle7 and was able to the run the queries untouched, something that demonstrates the strength of Sybase SQL Server's optimizer. While Sybase SQL Server only ranked fourth in terms of the response time for the queries, it was the fastest in terms of network time. We briefly substituted SPX/IPX for TCP/IP and saw that this made very little difference in the results. Third-place XDB-Enterprise sports a cost-based optimizer and a read-ahead mechanism. It was able to run the queries unaltered.

Watcom SQL ranked forth, and once again gets the most-improved award, having taken last place in last year's tests. The addition of read-ahead capability and the work done to improve the optimizer have clearly paid off. Watcom SQL had the highest score in terms of response time but the lowest in terms of network-transfer time.

Microsoft SQL Server, which placed fifth, required a little tuning to optimize performance (unoptimized results, not shown here, were in excess of four hours). But in many cases, the company found that tuning for response time hurt the product's total time, and vice versa. Microsoft also found that a smaller packet size (512 bytes) improved many of the smaller queries but slowed queries returning a large number of rows (larger 4K packets improved those). While Microsoft would have preferred to tune for individual queries, our benchmark testing specification did not allow for this.

Informix OnLine pulled up the rear despite their read-ahead mechanism and cost-based optimizer. We also encountered an optimizer bug that caused a server crash on two of the queries (the database was not corrupted by the crash). The problem was in the Update Statistics command that placed an invalid number in the statistics page. The Informix consultant was able to patch the statistics page to work around the problem.

The **Load and Index** test measure how quickly the database system can import 18.11 million rows, and create 33 indexes. This test is of particular interest for judging products used to implement decision-support systems, where the database must be loaded and indexed on a regular basis. Load times for our BLOB table were not included in the load score. The raw data was provided to the vendors in key order.

Vendors were allowed to choose the structure of the indexes, although we specified the columns on which indexes had to be created. Because load-and-index is typically an isolated operation, we allowed vendors to tune specifically for this test, whereas we required them to run all other tests with a single preselected set of runtime values. All tables were loaded serially. It should be noted that real-world load-and-index times can be reduced by using multiple sessions.

All vendors loaded the database directly from the server, thus eliminating network bottlenecks and optimizing load rates. In addition, all vendors provided a mechanism to bypass the transaction log for better performance. All vendors except Watcom also provided a utility or used SQL extensions to perform the load. Watcom's ISQL utility does include a feature to load data but it is not an NLM implementation. To optimize performance, Watcom took advantage of the engine's NetWare interface to write a custom NLM load module. While most users would probably not do this, we felt that this approach might make sense in a decision-support environment. Watcom SQL is the only database in this story that can directly interface with another NLM.

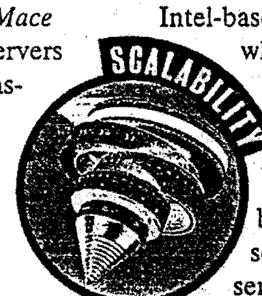
Oracle7 demonstrated its ability to load and index very quickly. While the actual load times lagged behind XDB-Enterprise, Oracle7 quickly made up for lost time with its efficient indexing mechanism. XDB-Enterprise was second overall, and takes top honors in load speed. This may be attributable to Windows NT's asynchronous I/O capabilities and the multithreaded nature of the load utility. Sybase SQL Server placed third, an impressive achievement considering that it created a clustered index on all the tables. While Sybase SQL Server did not have to perform a sort on the data, it did have to move the data physically to put it in a clustered structure. Watcom SQL took fourth place but with the second-fastest index time. Informix OnLine placed fifth, and Microsoft SQL Server placed last. While Microsoft SQL Server did place fourth on the data load, the overhead of creating a clustered index pulled the package to the rear.

The **Export** test measures how fast a database can export a 1-million-row table into comma-delimited ASCII text format. The export was made to a local disk on the server to avoid network overhead. Interestingly, several of the vendors actually took longer to export the table than to load it. This may be due to the overhead of a binary-to-ASCII conversion, which is typically more expensive than ASCII-to-binary. Also, when loading data, a database can cache multiple rows and write them as a single block. Export operations are typically dependent on the operating system's file-system cache. For most users, data export times will not be a significant issue.

—Analysis written by Brian Butler

Intel-Based SMP: *How Strong?*

By Brian Butler and Thomas Mace



Even the fastest Intel-based servers may not be fast enough for massive client/server database applications. The classic gambit for the power-starved has been to forgo the Intel platform in favor of RISC-based symmetric multiprocessing (SMP) servers. (For information on comparative performance of Intel and RISC servers, see the sidebar "Competing with RISC.")

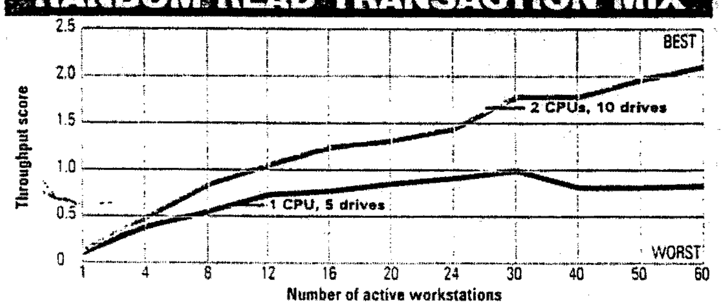
There is an alternative. The emerging class of Intel-based SMP servers delivers substantially improved performance over traditional single-CPU hardware. Scalability testing on the Intel-based Compaq ProLiant 4000, which can accept up to four CPUs on plug-in daughtercards, showed that with an appropriate operating system and database, doubling the number of processors and disk drives in the server can effectively double performance.

CLOCKING SMP ON INTEL

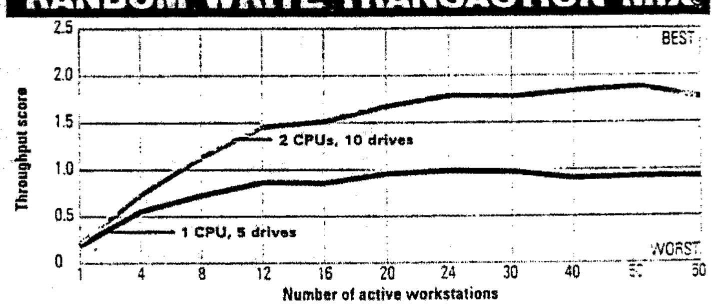
To investigate the benefits of Intel-based SMP database servers, we ran a series of scalability tests using Oracle7 Server for SCO Unix, Version 7.1. Under Oracle7, each client connection to the server is an independent process. The underlying SCO Unix operating system can distribute these processes symmetrically across multiple CPUs.

For testing, we used a subset of the AS^{3}AP database performance tests used for the main reviews. Results from the multiuser portion of the tests show throughput measured in transactions per second (tps). These results (see the accompanying graphs) are shown in normalized form based on the maximum throughput achieved by a single-CPU reference configuration. Query, Load, and Index are timed tests; the charts show the enhanced system's performance as a simple percentage of the reference system's

RANDOM READ TRANSACTION MIX



RANDOM WRITE TRANSACTION MIX



define conditions that can trigger an operating system script. You could use this feature to perform functions such as sending an administrator alert and initiating an automatic backup when a certain percentage of remaining log space is exceeded.

Other modifications to the database server and optimizer include a rewritten lock manager and loosened constraints on update-in-place. The optimizer can now use an available nonclustered index for queries containing an Order By clause. Microsoft has also implemented asynchronous checkpoints so that transactions can continue while a checkpoint is implemented. Dirty data pages are written to disk by a lazy-writer thread, reducing the overhead of the checkpoint operation.

While the server supports triggers and stored procedures, it also adds a powerful new feature called *extended stored procedures* aimed at leveraging workgroup technologies such as e-mail. Extended stored procedures are external Windows NT dynamic link libraries (DLLs) that can be dynamically loaded and executed on the server. For example, you could use this feature within a trigger to broadcast an e-mail message in response to a changed inventory level. Because every thread on the server is under structured exception handling, the server and database are protected from any errors arising from an extended stored procedure. Should a protection fault occur, only the thread would be terminated, not the process.

SUITABILITY TO TASK

Microsoft SQL Server for Windows NT

Production OLTP	EXCELLENT
Decision support	FAIR
Workgroup database	EXCELLENT
Connectivity & deployment	EXCELLENT

Although Microsoft has made improvements to Sybase SQL Server 4.2, it has not adopted some of the significant enhancements that Sybase introduced in its System 10. These include replication, declarative referential integrity, and ANSI cursors. Unlike Sybase System 10, Microsoft SQL Server cannot dump a single database to multiple backup devices.

Microsoft SQL Server also lacks transparent two-phase commit (this feature must be coded via the C interface), row-level locking, and built-in auditing. Remote procedure calls are outside of transaction management, a potential danger since consistency between remote databases cannot be physically guaranteed. While the Windows NT operating system is C2-level secure, the database provides only standard table-level security. Microsoft has committed to shipping a number of enhancements in future releases including declarative referential integrity, bidirectional scrollable cursors, parallel backup, and replication.

ıre.

We began by establishing a refer-
:e score using the ProLiant 4000 in
ı standard test configuration for this
ry: an array of five 2.1GB hard disks
ḋ a single 66-MHz Pentium CPU. In
s configuration, Oracle7's perfor-
.nce is strongly I/O-bound so that
ıply adding a second CPU would
ve had little effect.

To scale performance while keeping
: same balance between CPU and
k loads, we added five additional
ves and a second Pentium CPU and
an our tests. The results of the Ran-
m Read Transaction Mix test show
ıt throughput for the 2-CPU, 10-disk
:tem was well over twice that of the
erence system. The Random Write
ansaction Mix test results show that
: SMP system was just shy of twice as
:t.

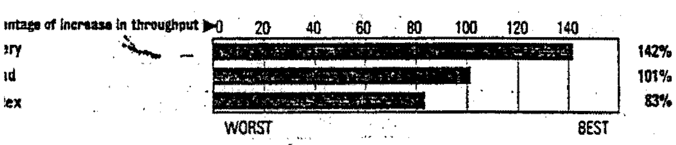
Version 7.1 of Oracle7 for SCO
ıix also supports parallelization of
ery, load, and index operations internally. The engine accomplishes this by dividing operations into separate tasks that are spread across multiple processors. Examples of operations that can be parallelized include table scans, joins, aggregations, and various sort operations.

To see how well the ProLiant 4000's SMP capabilities support these features, we selected queries from our standard Ad Hoc Query test. The results show better than a 140 percent improvement in query execution time on the double-CPU system. Not all queries benefit from parallelization, however. We tried several queries that do not perform table scans or large sorts and saw no performance improvement. The tests also show a doubling of load performance and a substantial improvement in indexing, in part because of the efficiency of parallel sorts.

While the lure of RISC-based servers remains strong, Intel-based database servers with SMP upgradability offer an alternative that is definitely worth investigating. □

QUERY, LOAD, AND INDEX

2 CPUs and 10 drives vs. 1 CPU and 5 drives



ICK TOOLS

ıe superb graphical administration tools
.ndled with the server let the adminis-
ıtor manage the database, the operating
stem, and networking from a single lo-
tion. The SQL Object Manager is a
ck change-management application
at can be used to create stored proce-
ıres, triggers, tables, indexes, rules,
ews, and other database objects. It also
:ludes a bulk-copy program that, unlike
e original command-line bulk copy pro-
am, provides postmortem information
r failed operations. The SQL Object
anager can also generate a transact
ƆL data definition language (DDL)
ript from existing database objects that
n be used to recreate a database on an-
her server or document an existing
ıtabase structure.

The SQL Administrator tool is target-
l at device and database management. You can use it to create databases, devices, and users and to implement security. The ISQL/Windows program provides a basic Interactive SQL (ISQL) server interface with the convenience of a few Windows navigation features. It also lets you create a *showplan*, a graphical display of the access plan for any given query, and displays I/O statistics graphically for tuning and optimization. Standard command-line ISQL is also provided.

All of the Microsoft tools can simultaneously connect to multiple databases, but they cannot administer multiple servers as a group. Like most competing toolsets, they also lack integration; you'll need to switch from one to the other depending on the task at hand. Microsoft plans to roll SQL Administrator and Object Manager into a single tool eventually. Future versions will also support OLE 2.0 drag-and-drop behavior and allow for remote management of server groups.

Although Microsoft SQL Server is positioned as a client/server computing solution for the masses, its overall performance—despite a few gaps—puts it in a league with the industry leaders. Its stability and strong administration tools are benefits in any applications. Except for the fading OS/2 release, Microsoft SQL Server is an NT-only solution and will ultimately be only as scalable as Windows NT itself. But if you are a believer in Windows NT, Microsoft SQL Server is a robust, well-oiled solution.

▶ **FACT FILE**

Microsoft SQL Server for Windows NT, Version 4.21

List price: Server software, one development system, 60 client connections, and client software: $8,690. **Requires:** Server: 386-based PC or better, 16MB RAM, 25MB hard disk space, Microsoft Windows NT 3.1 or later. DOS client: 286-based PC or better, 640K RAM, 1MB hard disk space. **In short:** Microsoft SQL Server offers a compelling combination of a powerful database engine, superb graphical administration tools, excellent connectivity features, and unmatched integration with the Windows NT operating system. Its performance goes beyond workgroup demands and puts it among the top products in our roundup. This product is a Windows NT–only solution, but if your organization can buy into a closed-shop strategy, the integration of server, operating system, and networking components is hard to beat.

Microsoft Corp., One Microsoft Way, Redmond, WA 98052; 800-426-9400, 206-882-8080; fax, 206-936-7329

482 on reader service card

Oracle Corp.

EDITORS' CHOICE

Oracle7 Server for NetWare

Oracle7 Server for NetWare, Version 7.0.16, is a comprehensive, complex package that rolls together just about all the features you'll find in competing products. It is exceptionally fast, eminently stable, and very well suited to both multiuser and decision-support tasks. Oracle7 demands a sizable up-front investment and solid professional skills to get it up and running. But for mission-critical applications, especially in distributed environ-

Competing with RISC

SCALABILITY

By Brian Butler and Thomas Mace

From a PC-centric perspective, the Compaq ProLiant 4000 used as this story's test platform is a powerful machine. But large companies will naturally consider other hardware options when weighing a major client/server investment. This raises a basic question: How well does Intel hardware perform when compared with RISC?

We used ZD Labs' Ritesize IV test suite based on Sybase System 10 RDBMS to pit four RISC-based servers against a ProLiant 4000 equipped with two 66-MHz Pentium CPUs. The evaluated RISC systems were the Data General AViiON 8500 powered by six Motorola 88110 processors; the DEC 3000 Model 800S AXP Deskside Server with a single 200-MHz AlphaAXP 21064; the HP 9000 Series 800 Model G70 with two 96-MHz PA-7100 processors; and the IBM RISC System/6000 POWERserver 590 with a single 66-MHz IBM POWER2 CPU.

The Sybase System 10 database engine we used for testing the platforms supports SMP hardware by launching multiple instances of the database engine. The database server then binds the engines to a particular processor.

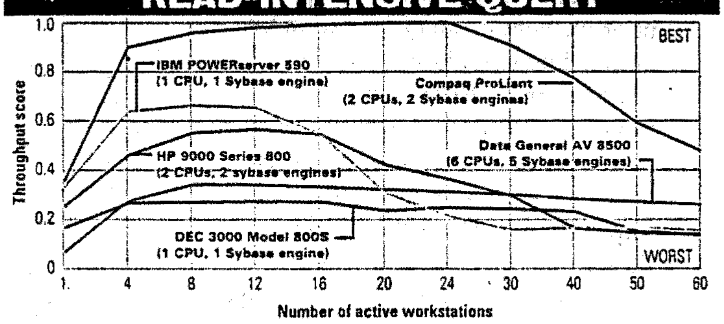
The 60-client test-bed was similar to the one used for review testing. In the accompanying graphs, the scores are shown in normalized form, with the maximum throughput achieved by the Compaq ProLiant in each test indicated as 1.0.

Our Processor-Intensive Query test selects a small number of cached rows. The Read-Intensive Query test performs a join on two tables that exceed the database cache size. The Mixed Workload test runs four transactions: the Processor-Intensive Query, the Read-Intensive Query, an Update transaction, and an Insert.

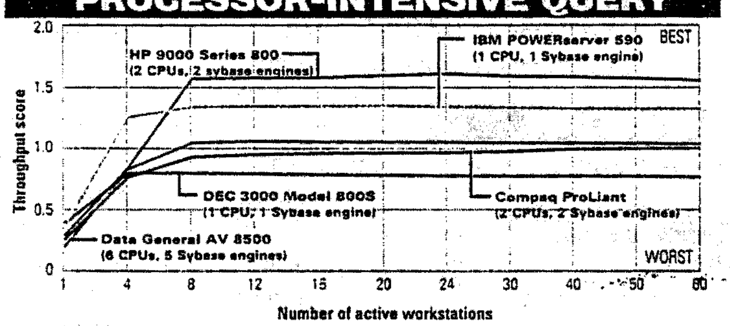
While the overall winner was the HP 9000, the Compaq ProLiant was a competitive midrange performer on both the Mixed Workload and Processor-Intensive Query tests. On the Read-Intensive Query test, it was fastest overall thanks to its strong disk-controller technology.

Examining performance is only part of the process of selecting a database server. Operating system maturity, hardware redundancy, service, and even intangibles such as the company's reputation will play a role. But in a straightforward speed comparison, Intel SMP hardware is clearly in the same league as the RISC-based heavyweights. □

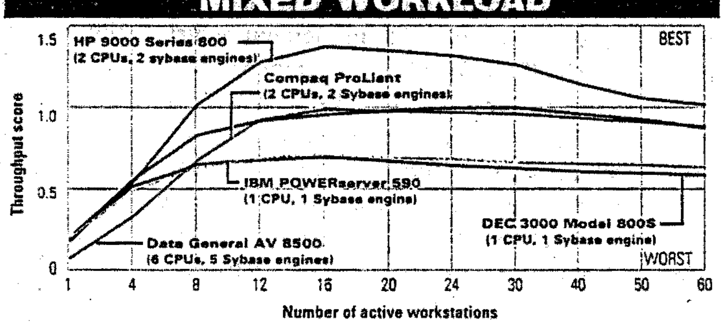
READ-INTENSIVE QUERY



PROCESSOR-INTENSIVE QUERY



MIXED WORKLOAD



ments, you'll be hard pressed to find a more robust solution.

Version 7.0.16 is little changed from the version we saw last year. But, even after a year's time, Oracle7 still looks extremely competitive. A newer release, Version 7.1, is already available on several platforms and is expected for NetWare by the end of this year. The same Oracle7 code base is currently available on about 80 hardware platforms, extending the product's reach from the Microsoft Windows desktop to the mainframe.

PERFORMANCE EDGE

While Oracle7 has adopted many of its competitors' best features, it owes some of its performance edge to technology that is not widely used by other products. For example, Oracle7 implements a multiversioning concurrency model, a unique feature in this roundup (Borland's upcoming InterBase Workgroup Server will offer a similar design).

In a multiversioning scenario, each transaction sees a consistent, unchanging view of the database precisely as it was when the transaction began. If the underlying data is changed by a later transaction, information from rollback segments

Ingres Server: *Still on Hold*

'y Brian Butler and Thomas Mace

.ast year's SQL roundup included a əview of Ingres Server for OS/2, Verion 6.4, which at that time had just een acquired by The ASK Group. We ɔund this product to be intriguing but awed by serious bugs. This year we lanned a follow-up look at a point elease of Version 6.4 designed to ddress the problems we encountered. Vorking with technical representa-ves from The ASK Group, we put the pdated product through our standard ests in preparation for this story. ngres Server made it through our .oad and Index and Ad Hoc query ests without a hitch but in our multiser tests, we ran into significant bugs ıat made the product spontaneously rop clients. The performance numers we were able to generate put ngres Server at the bottom of the test neup.

In the middle of our tests, the ASK Group was acquired by Computer ssociates International (CA), which nmediately withdrew all Intel-based ngres Server products from the market, ıcluding the product we were testing. CA stated that the version we saw was a eta product not ready for release. Customers who received the product vere told that they had received a beta version and that they would get the final release when it was ready.

CA plans to debug the existing Intel ports and release them once they are fixed. As we went to press, a ship schedule had not been announced. Releases for Microsoft Windows NT, NetWare, OS/2, SCO Unix, Solaris, and UnixWare are planned. CA also stated that it will continue to support the existing Ingres installed base.

FUTURE DIRECTIONS

TECHNOLOGY PIONEER

Ingres Server, which had its origins in the Berkeley Ingres prototype, has always had a reputation as one of the most academically strict relational databases. In the past, Ingres has served up impressive technology, pioneering cost-based optimization and many other now-standard database features, including triggers, which Ingres calls *rules*. Other high-end engine features of the version we tested include event alerters, user-defined data types, user-defined functions, stored procedures, and two-phase commit. Ingres also offers a distributed database strategy through its Ingres/Star server.

Ingres Server is unusual in that some of its most powerful features are managed by a server extension product called Ingres Knowledge Management. This component provides Ingres's rules and event alerters and also offers administration of permission levels by individual, group, or application. It also offers a resource-control feature based on the Ingres optimizer for preventing runaway queries.

Although the Ingres Server product we looked at suffered few protection-fault shutdowns during testing, most of the problems we encountered seemed to stem from the Ingres client libraries. CA agrees with our assessment and plans to concentrate its debugging efforts in this area. In the near term, CA plans to work on performance improvements within the existing engine and sees Ingres Server ultimately challenging Oracle and Sybase in mainstream OLTP markets. In the long term, they plan to rearchitect the database to support parallel operations and massively parallel hardware.

Ingres Server has clearly languished in the recent past, but its strong technology deserves a better fate. We look for future releases from CA to turn the product around. □

---

ed to maintain the first transaction's ;istent view. The big advantage to a tiversioning model is that read transɔns do not need to acquire locks that k write transactions, improving overoncurrency.

Vhen locks are needed, Oracle7 uses a -level locking scheme instead of the e conventional page-level locks. The base supports an unlimited number ɔw-level locks that never escalate to ; or table locks. While the page-level ing schemes used in other products adequate for most applications (and ɪretically entail less management head), the page is not a "natural" unit orage. This can increase the difficulty of tuning operations. Row-level locking also provides optimal concurrency, indicated by Oracle7's excellent scores on our Random Write Transaction Mix test.

Oracle7's triggers are similar to those in other products except that the user can stipulate when a trigger executes relative to execution of the SQL statement that fires it. The database also supports declarative referential integrity, and automatic cascading deletes can be set to eliminate child rows when parent rows are deleted.

Since Version 6.0, the log manager has been optimized to support fast commits and group commits. Moreover, only changes to data are logged, not the entire before-and-after image of the row.

SUITABILITY TO TASK

Oracle7 Server for NetWare

Production OLTP	EXCELLENT
Decision support	EXCELLENT
Workgroup database	GOOD
Connectivity & deployment	EXCELLENT

Oracle7's cost-based optimizer does not use histograms, but it does gather a number of statistics from tables and indexes. Using the Analyze command, you can update these statistics based on a subset of the data. This can be useful for huge decision-support databases where a complete table scan would be unduly long.

On our Ad Hoc Query test, the optimizer picked the wrong access method on our sort query. This resulted in a

The Price of Performance

By Brian Butler and Thomas Mace

If all of the SQL database servers in this roundup cost the same, picking out the right one would only be a matter of weighing feature sets. But the wide range of prices here—from a mere $5,390 to almost $30,000—adds to the complexity of your decision. In order to help clarify pricing issues, we look at SQL database prices in two different ways: total cost and bang for the buck.

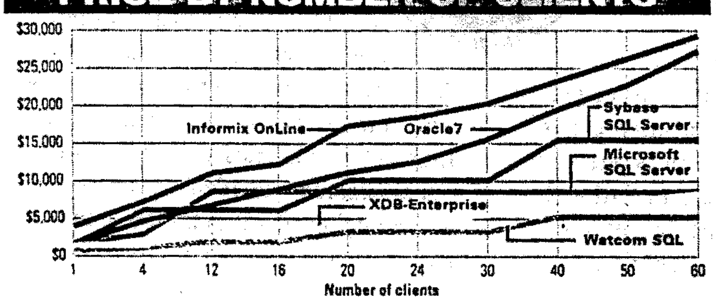
One good piece of news is that SQL database pricing has gotten noticeably simpler. SQL vendors used to be notorious for devising complex schemes with separate pricing for server connections and client libraries. This year, all with the exception of Informix are pricing on a per-user basis, where client software is now essentially free. Workgroup bundles from the major players will soon simplify pricing even further.

STRAIGHT COST

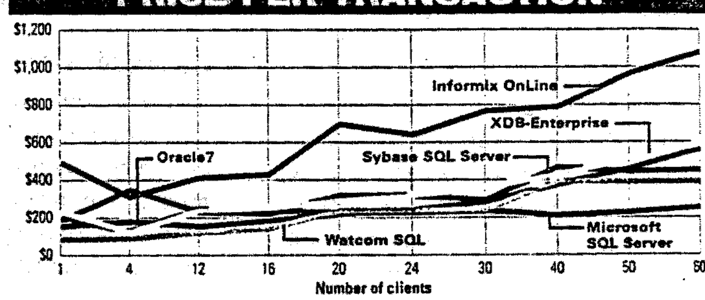
The simplest way to view the price of a database is by its straight deployment cost (see the chart "Price by Number of Clients"). This cost, shown for 1 to 60 users, includes the required number of user licenses, required client software, and one standard 3GL development kit (the cost of network protocol stacks and support is not included).

Prices vary widely and exhibit a

PRICE BY NUMBER OF CLIENTS



PRICE PER TRANSACTION



10-hour run on a test that ultimately took only one hour and 20 minutes to complete. The problem was fixed using Oracle's well-documented Hint mechanism for overriding the optimizer. A nice Hint subtlety is that the mechanism lets you tune queries for best response time (the amount of time required to return the first row of data) or best overall query time.

Stored procedures are available, although they cannot return result sets. But you can send an array of values to a stored procedure. This elegant trick could be used for problems such as inserting multiple line entries in an order table. Stored procedures can be logically grouped together into what Oracle calls a Package. This makes for easier administration, allowing the user to maintain all the stored procedures for a particular application as a single entity, for example. You can define global variables for an entire Package and also grant and revoke permissions at the Package level.

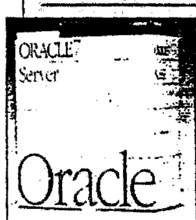
The current release of Oracle7 still lacks the GUI administration tools that recently premiered on Oracle's Workgroup Server product. It does ship with a full-featured character-based administration tool called SQL*DBA. This utility manages tasks such as opening and closing the server and lets you monitor system performance and use, perform backups, and interactively execute SQL statements. SQL*DBA provides two modes of operation: a screen-based interface complete with drop-down menus, and a command-line interface. You can also use SQL*DBA to monitor a variety of system statistics and to create and drop tablespaces and rollback segments.

The current release offers the convenience of role-based security administration. Users can be added to more than one role, and user privileges can be granted or revoked at the role level as needed. While this seems like a simple concept, it marks a big improvement over previous versions of the product in which privileges had to be granted individually for each user. The current version is B2-level secure. A separate product, Trusted Oracle7, is C2-level secure.

Oracle7 has strong support for distributed databases. It supports transparent two-phase commit and controls remote procedure calls (RPCs) as an integral part of transactions. Oracle7

▶ FACT FILE

Oracle7 Server for NetWare, Version 7.0.16



List price: Server software, one development system, 60 client connections, and client software: $27,400. **Requires:** Server: 386-based PC or better, 12MB RAM, 30MB hard disk space, NetWare 3.0 or later. DOS client: 286-based PC or better, 100K RAM, 100K hard disk space. **In short:** Excellent transaction processing speed and a rich feature set add up to one of the most sophisticated databases available. Oracle7 offers virtually all the features of competing products, and its multiversioning consistency model and row-level locking provide excellent concurrency. Despite its high price, Oracle7 still garners an excellent price/performance ratio, but it is not geared toward organizations with limited budgets. This is not a database for the meek, but for the most demanding applications, you'd be hard-pressed to find a better solution.

Oracle Corp., 500 Oracle Pkwy., Redwood Shores, CA 94065; 800-672-2531, 415-506-7000; fax, 415-506-7200

478 on reader service card

number of strategies. Informix OnLine is consistently the most expensive; its price increases in a smooth curve from $3,995 for one user up to $29,395 for 60 users. Oracle7, the second most expensive package, takes a similar approach to pricing. Sybase SQL Server is less expensive and shows a simpler tiered pricing structure. Microsoft SQL Server carries simplification even further, providing for 12 to 60 clients for the same price of $8,690. Watcom SQL, the least expensive package we tested, begins at $790 for one user and rises to a modest $5,390 for 60 users. XDB-Enterprise, while slightly more expensive, closely follows Watcom SQL's pricing.

PRICE/PERFORMANCE

A price/performance analysis shows the products in a very different light (see the chart "Price per Transaction"). To generate this graph, we divided each price by each product's transaction-per-second throughput on our Random Write Transaction Mix test.

The products look much more alike from this perspective. Oracle7, Sybase SQL Server, Watcom SQL, and XDB-Enterprise deliver very similar price/performance ratios across the range of client loads. Microsoft SQL Server is the leader above 20 clients, albeit by a narrow margin. The only standout is Informix OnLine, which offers the poorest mix of price and performance across the board. As with most product groups offering similar bang for the buck, your choice here will be dictated by the performance level you need.

As you calculate server prices, it's important to remember that servers are only a part of the equation. Client/server technology remains an expensive proposition to implement successfully due to the lack of turnkey systems, and by the time that you've factored in software development and support costs, the price of the most expensive server may not look so big. □

also initiates two-phase commit for any RPC outside the Oracle7 since the database has no way of knowing what the result of the RPC will be. Cost-based optimization is available for distributed queries based on statistics and available indexes in the distributed environment. Oracle7 also has a trigger-based replication scheme for making distributed read-only copies of tables, table subsets, or query results. Oracle has announced a more robust symmetric replication technology, which is expected to ship by the end of this year.

Access to non-Oracle7 data through RPCs or SQL is provided by the Oracle Transparent Gateway (formerly SQL*Connect), a set of gateway products for a variety of relational and nonrelational systems.

Its first-place Load and Index test scores are attributable in part to its SQL*Loader, one of the fastest, most functional loaders we used. It supports both direct-path and conventional-path loading. We used direct-path, in which records are written directly to the database block, bypassing most database processing. While direct-path loading has some restrictions, conventional-path loading is always available as a workaround.

The next version of the NetWare product should ship as Version 7.1 before the end of the year. This revision will add Oracle7 Symmetric Replication, and the impressive parallel query–execution, data-loading, and index-creation features already available in the Unix release we used to test CPU scaling (see the sidebar "Intel-based SMP: How Strong?"). Version 7.1 will also include support for user-defined SQL functions and dynamic SQL statements—statements whose contents are not known until runtime. Additional slated improvements include tweaks to the optimizer, encrypted network passwords, and faster database recovery.

NOT FOR THE MEEK

Oracle7 was the fastest database tested for this roundup, taking the lead on five out of seven tests. Almost no tuning was required to achieve these results.

At the same time, this is not a database for the meek. Exploiting its huge array of features demands expertise and time spent with the superb encyclopedic documentation. Oracle7's price/performance ratio is excellent, but its high price is targeted at users who need speed and functionality, not savings. But for bulletproof operation in high-stress transaction-processing environments, Oracle7 is a winner.

Sybase Inc.

• Sybase SQL Server for NetWare

Sybase SQL Server for NetWare, Version 10.01, represents a solid upgrade to Version 4.2, which we reviewed last year. Sybase has made numerous enhancements, tweaks, and fixes to the engine, which proved to be stable and extremely fast in testing.

The jump from Release 4.2 to 10.01 brings Sybase SQL Server's version numbering in line with the company's System 10, an important family of add-on server products designed to address connectivity, replication, administration, and scalability.

NetWare makes an excellent showcase for the database engine's core features and performance, but a lack of add-ons leaves Sybase SQL Server in limbo as a product: While many System 10 components are shipping on other platforms, the only component available for the NetWare product we tested is the Backup Server. Our testing bundle only included the server, Backup Server, bulk copy program, and Interactive SQL (ISQL). Later this fall, Sybase plans to sweeten the offering by releasing NLM versions of some other System 10 components and repackaging the NLM server in separately priced workgroup and enterprise editions.

SUITABILITY TO TASK
Sybase SQL Server for NetWare

Production OLTP	EXCELLENT
Decision support	EXCELLENT
Workgroup database	GOOD
Connectivity & deployment	EXCELLENT

MATURE ENGINE

The NLM version of Sybase SQL Server we tested includes much of the advanced engine technology that first lifted Sybase to prominence. The list includes Sybase SQL Server's stored procedures (which

▶ SUMMARY OF FEATURES CONTINUES

SQL Databases

PC MAGAZINE

Products listed in alphabetical order
■ = YES □ = NO

	Informix OnLine for SCO Unix 5.01	Microsoft SQL Server for Windows NT 4.21	Oracle7 Server for NetWare 7.0.16	Sybase SQL Server for NetWare 10.1	Watcom SQL Network Server for NetWare 3.2	XDB-Enterprise Server 4 for Windows NT
List price*	$29,395	$8,690	$27,400	$15,590	$5,390	$8,490
Cost of standard one-year telephone support	$1,980	$7,500	$1,290	$4,000	$5,000	$2,000
SQL Implementation						
ANSI compatibility:						
ANSI Level 2 with Enhanced Integrity	■	□	■	■	■	■
ANSI Level 2	■	□	□	■	■	■
ANSI Level 1	■	■	□	□	■	■
Full DB2 compatibility	□	□	□	□	□	■
Binary large object (BLOB) data types	■	■	■	■	■	■
User-defined data types	□	■	□	■	□	□
User-defined range limits on data types	□	■	□	■	□	■
Advanced mathematical and statistical functions	■	■	□	■	□	■
User-defined functions and operators	■	□	□	■	□	□
Cost-based/rule-based optimization	■□	■□	■■	■■	■■	■□
Transaction Management						
Locking:						
Record-level	■	□	■	□	■	■
Page-level	■	■	□	■	□	□
Table-level	■	■	■	■	□	■
Adjustable for each table	■	□	□	■	□	□
Automatic lock escalation	□	■	□	■	□	□
Consistency levels supported:						
Cursor stability	■	■	N/A❶	■	■	■
Repeatable reads	■	■	■	■	■	■
Multiversioning	□	□	■	□	□	□
Release locks	■	■	□	□	■	■
Uncommitted reads	■	□	□	□	■	■
Read-only databases	■	■	□	■	■	■
Cost-based deadlock-detection schemes:						
Engine can abort transaction causing the deadlock	■	■	■	■	■	■
Engine can abort via a timeout option	■	■	□	■	□	■
Programming Interface						
Includes call-level interface	■	■	■	■	■	■
ODBC support included	■	■	■	■	■	■
Host-language interface:						
ANSI-compatible cursors	■	□	■	■	■	■
Included SQL precompilers	C	C, COBOL	C, COBOL, FORTRAN	ADA, C, COBOL	C, C++	C, COBOL, DL/1
Backward scrolling in result set	■	■	□	□	■	■
Preserves cursor context after Commit and Rollback	■	■	□	□	■	■
Supports result-set inserts	■	■	□	■	■	□
User can insert, update, and delete using an array of variables	■	□	■	■	□	□
Stores procedures in database	■	■	■	■	□	□
Embedded select, update, delete, insert	■	■	■	■	N/A❷	N/A❷
Supports control and flow logic	■	■	■	■	N/A❷	N/A❷
Supports message and error-code handling	■	■	■	■	N/A❷	N/A❷
Accepts variables and returns values or messages	■	■	■	■	N/A❷	N/A❷
Supports row-at-a-time processing	■	■	■	■	N/A❷	N/A❷
Supports remote stored procedures	■	■	■	■	N/A❷	N/A❷
Performs binding and optimization before runtime	■	■	□	□	■	■
Offers Wait and Nowait for lock to be released	■	□	■	□	■	□
Database Server Environment						
Database server architecture	Multithreaded	Multithreaded	Process-per-user	Multithreaded	Multithreaded	Multithreaded
Portability:						
DOS	□	□	□	□	■	■
Microsoft Windows	□	□	■	□	■	■
Microsoft Windows NT	□	■	■	■	□	■
NetWare	□	□	■	■	■	□
OS/2	□	■	■	■	■	■
Unix	■	□	■	■	□	□
VM	□	□	■	□	□	□
VMS	□	□	■	■	□	□
MVS	□	□	■	□	□	□

*The list price includes server software, 60 client connections with client software, and 1 development system.
N/A ❶--Not applicable: The product implements a multiversioning concurrency model.
N/A❷--Not applicable: The product does not have this feature.

PRODUCT COMPARISON

(A-I)

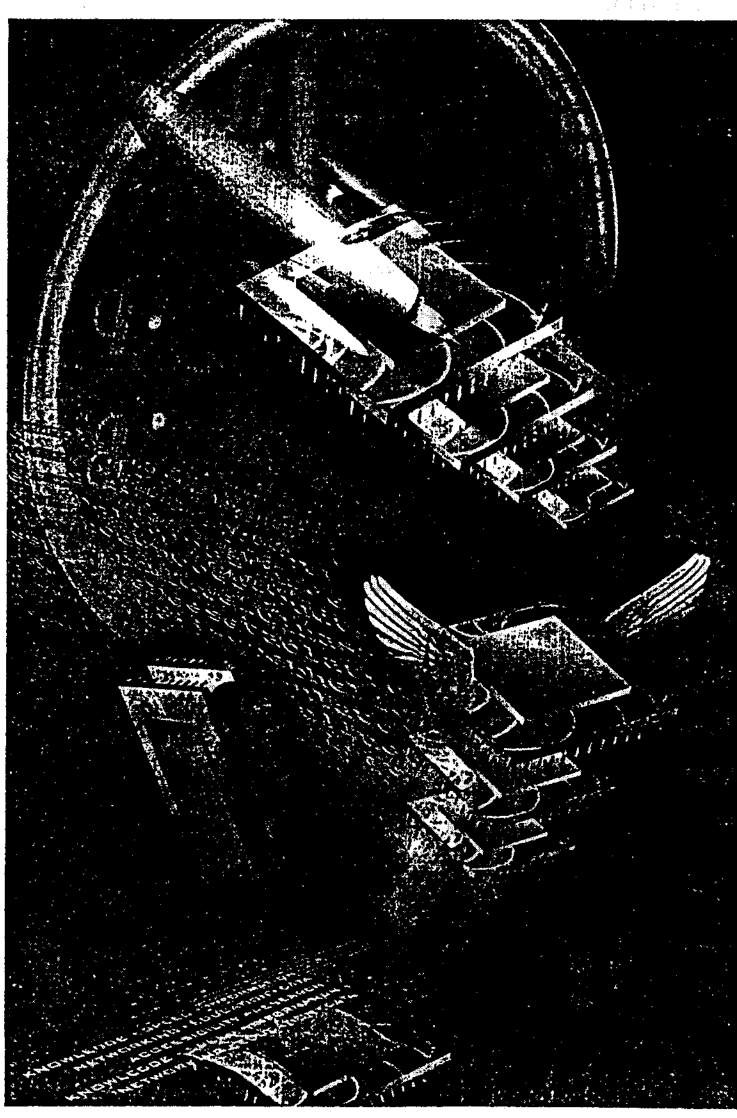
Symmetric multiprocessing servers

Scaling the performance wall

High-performance Pentium-based multiprocessors are catching up to RISC systems, giving multiprocessing network OSes, such as Windows NT, a foothold in what was formerly the sole domain of Unix.

COMPARED

- **Revolution Q-4SMP** Advanced Logic Research Inc.
- **Manhattan P5090** AST Research Inc.
- **ProLiant 2000 5/90** Compaq Computer Corp.
- **PowerEdge XE 590-2** Dell Computer Corp.
- **Poly 500EP2** Polywell Computers Inc.



JESSE DOQUILO

If wimpy, single-processor performance has you climbing the walls, quit climbing and start scaling up with symmetric multiprocessing (SMP) systems. These Pentium-based machines are finally approaching the scalability and performance capabilities of RISC-based systems. And with multiprocessing network operating systems, such as Microsoft Corp.'s maturing Windows NT 3.5 and Novell Inc.'s upcoming NetWare MP, you can take advantage of this new processing power without having to switch to Unix.

Symmetric multiprocessing lets multiple CPUs share a server's memory, interrupts, and devices through a run-time algorithm. How much this boosts performance depends on the application, but you're likely to see at least some improvement across the board. SMP systems probably appeal the most to two groups — those downsizing mainframe applications to client/server systems and those who need to boost an already heavily loaded server. According to our survey of 1,000 *InfoWorld* readers, more than 80 percent of those who use SMP servers use them with a database engine, such as Microsoft's SQL Server or Oracle Corp.'s Oracle7.

Using Windows NT 3.5 as our NOS, we measured how much scalability the five Pentium-based SMP servers in this comparison provided by testing them with one processor and then two. The good news: If your network handles mostly CPU-bound applications, such as online transaction processing (OLTP), these servers offer a way up and out of the performance hole. Advanced Logic Research Inc.'s Revolution Q-4SMP and Compaq Computer Corp.'s ProLiant 2000 5/90 were the most scalable servers by far, performing almost twice as fast on two processors than on one. The Revolution was the upset winner of our speed tests, outperforming even the venerable ProLiant's multiprocessing server by nearly 20 percent in OLTP.

We chose Windows NT 3.5 as the multiprocessing NOS for our benchmark tests because of its focus on scalability. Most of the more than half of our readers using SMP servers are using two processors, but readers projected they might use as many as six processors per server. Because NT 3.5 supports as many as four processors right out of the box, it fit well with our readers' needs. If you need to harness the power of more processors, you can buy NT from a vendor like Sequent Computer Systems Inc., in Beaverton, Ore., which provides NT support for as many as eight processors. In this comparison, only the ProLiant and the Revolution were capable of using more than two processors. The Revolution can use as many as four 100-MHz Pentium chips, and the ProLiant can use as many as four 90-MHz Pentiums.

SCALABILITY IS REAL. Perfect scalability is a 100 percent performance increase between one and two processors. For exam a server with perfect scalability ran 50 transactions on a single processor in 1 hour, it would complete those transactions in 30 minutes using two processors.

None of the servers scaled perfectly, but the Revolution and ProLiant did quite well. The Revolution ran slightly more than 92 percent faster on two processors than one, and the ProLiant ran more than 97 percent faster.

Even at these speeds, these servers still ran about 25 percent slower than a MIPS Technologies Inc.-based machine running Windows NT 3.5, which we used as a point of comparison. (The NEC Technologies Inc. RISCserver 2200 was not yet shipping when we tested; see story, page 89.) But for the first time, Intel-based systems come close to RISC, and that's big news. It means that at least for a while, IS managers can cope with more demanding processing needs by simply moving their applications to Intel machines with more processors. The bad news: If you simply must have that remaining 25 percent speed increase, you'll have to port to a MIPS machine. And even though it runs NT, you'll have to port all of your data, not to mention buy new versions of your applications.

NOT A PANACEA. If you're dealing with I/O-bound applications — such as printing, file transfer, and, to a lesser extent, decision support — an additional processor won't help much. On such applications, we found only minor improvements, usually in the neighborhood of 15 percent. Traveling over the network appears to put a heavy dent in the effect extra processors have in such environments. Expecting them to make a difference would be tantamount to buying a Corvette and expecting it to make rush-hour traffic go away.

Computer manufacturers are well aware of this phenomenon; that's why they like to benchmark their multiprocessing systems on CPU-intensive activities such as database transactions — not on file and print services, which are much more I/O dependent.

Our readers were more concerned about transaction speed than about the more I/O-dependent transactions, according to our survey, so we tested accordingly. As expected, the servers' performance scaled much better in our OLTP test, which we designed to be CPU intensive (see "How we tested," page 85), than in responding to queries, a more I/O-intensive task.

While mulling over the lack of scalability in decision-support operations, we discovered a huge variance in the performance of each server's disk I/O subsystem — which ultimately determines the performance of your server.

Although we did not base any of our scores solely on these disk I/O results, you'll want to pay close attention to them (see chart, page 90) if your database servers perform both OLTP and decision-support operations (such as database queries) on a regular basis.

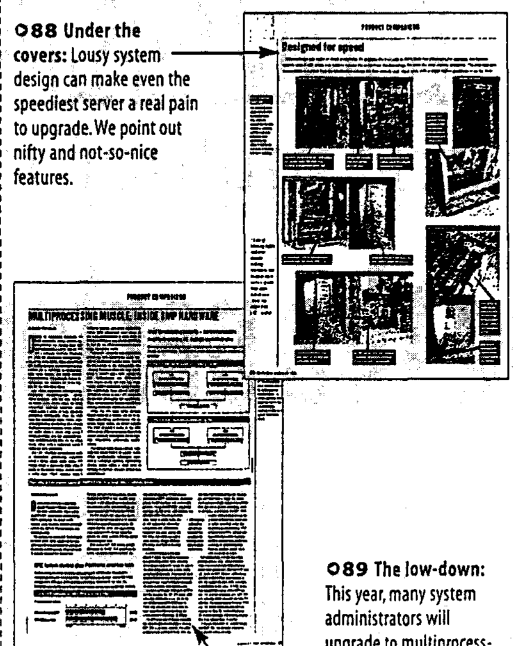
FOUR-PROCESSOR SUPPORT ON SQL SERVER 4.21A? NOT. Our testing turned up some other interesting results. With our scalability testing for two processors completed, we thought we'd fire up a few extra processors and see what even more could do. Using SQL Server 4.21a and Windows NT 3.5 on the Compaq ProLiant 4000 5/66, we tested three and then four processors. To our astonishment, three processors gave us virtually the same performance as two, and using four processors resulted in the same speeds we would have expected with three processors.

We rang Microsoft. It turns out that if you use SQL Server 4.21a's SMPStat parameter — not recommended by Microsoft — you tweak an internal parameter of the software, which in turn tells the software to use all available processors. If you don't, SQL Server treats a 4-processor system as a 2-processor system and a 4-processor system as if it uses only three. We didn't have any problems in our admittedly short tests, but Microsoft warns that using SMPStat could result in a deadly embrace, locking the database. Therefore, it doesn't support the use of SMPStat and won't help you out of any problems it causes. The upcoming Version 6.0 of SQL Server eliminates this quandary by using SMPStat as the default.

We had planned to include a Digital Equipment Corp. Intel-based SMP server (which it sells in addition to its own Alpha chip), but Digital was unable to provide us with one of its machines due to production schedule conflicts.

Also looming large in the SMP server race are multiprocessing systems based on the PowerPC chip. By June, sources expect versions of Windows NT and OS/2 to ship for PowerPC hardware, and both OSes will support SMP out of the box. This month, Microsoft formally released the beta version of Windows NT for the PowerPC, initially running on a Motorola system. (See "Playing with NT on PowerPC promises good times ahead for all users," March 13, page 108.) Regardless of which operating system proves most popular, Intel is feeling the heat from the PowerPC. We'll review the PowerPC servers as they ship.

A guide to this comparison



088 Under the covers: Lousy system design can make even the speediest server a real pain to upgrade. We point out nifty and not-so-nice features.

089 The low-down: This year, many system administrators will upgrade to multiprocessing servers. We explain the RISC alternative.

Contents

Results at a glance

If you can afford it, ask for a server with a dedicated Level 2 cache memory for each processor. The winner of our comparison, Advanced Logic Research Inc.'s **Revolution Q-4SMP**, was one of only two dedicated-cache Pentium systems. The super-fast RISC-based server we tested (see story, page 89) also uses dedicated caches.

There's a lot to like about the Revolution. It won both our online transaction processing (OLTP) and decision-support tests, showing its CPU performance and disk subsystem to be the best of any server we tested. It was the only machine capable of upgrading to four 100-MHz CPUs, aided by ALR's easily-plugged-in CPU boards. Its seven fans should keep the system plenty cool.

The Revolution didn't win in all categories, though. It's not as scalable as Compaq's ProLiant, and we were annoyed by the flimsy construction of its door. ALR was one of only two vendors that did not offer around-the-clock telephone support. The server comes with a five-year warranty, however.

Compaq Computer Corp.'s **ProLiant 2000 5/90** was the only other server capable of using four Pentiums (it uses 90-MHz chips). But you'll have to move the redundant array of independent disks (RAID) to an external drive cage in order to upgrade. The ProLiant is the most scalable of these servers — it showed a near-perfect 97 percent jump in OLTP performance when we added a second CPU. We also liked the system's handy SmartStart CD-ROM, which offered a choice of several network operating systems.

The ProLiant was the least expandable; it could only hold 10 gigabytes (GB) worth of additional hard drives without external drive cages.

Among systems designed for only two CPUs, the AST Research Inc. **Manhattan P5090** scored the highest, primarily because of its ease of use, including graphical utilities and some dealer preconfiguration. Only the Manhattan got an excellent rating for documentation. It also has two PCI slots. But the Manhattan was the only system with 256KB cache, not 512KB. This may be why it landed in last place in our OLTP comparison. Scalability and decision-support scores weren't impressive either.

The Score

Score	
7.3	ALR Revolution Q-4SMP
6.7	Compaq ProLiant 2000 5/90
6.3	AST Manhattan P5090
6.2	Polywell Poly 500EP2
5.8	Dell PowerEdge XE 590-2

The best thing about the Polywell Computers Inc. **Poly 500EP2** is its price. At $14,475, it's more than $7,000 less expensive than any other server in the comparison. Otherwise, the Poly is an average machine, with average performance numbers (it's a respectable transaction processor but had the worst scalability of the bunch) and terrible technical support. Let us say that again: Polywell's technical support was not only the worst in this comparison, it was the worst we've encountered in a while. Representatives were rude, practically hanging up on us even though we called during scheduled support hours.

Dell Computer Corp.'s **PowerEdge XE 590-2** is a mediocre performer. It was the slowest of all the servers at returning query results and performed almost as poorly in our transaction-processing tests. Compared to the other systems, its scalability was unimpressive.

We weren't particularly pleased with some aspects of the system's design, either. To gain access to the memory, you'll need a screwdriver and patience. And be careful where you leave the PowerEdge, because it doesn't have a lock or even a cover for the power switch. On the plus side, once we removed the case, we could easily swap cards and drives in and out without any problems. The PowerEdge has as many EISA slots as the ProLiant and supports two PCI slots as well. It's capable of storing as much as 24GB of data.

RELATED ARTICLES

March 6, page 37
IBM revamps server line
IBM prepares an entry-level dual-processor 90-MHz Pentium LAN server to challenge Compaq's dominant market share.

Feb. 27, page 66
No-fault insurance
We examine four RAID subsystems and tell you how to choose the right RAID.

Jan. 30, page 6
Novell SMP delayed until middle of year
Originally promised in 1989, the NetWare kernel is now due for SMP support this summer.

Dec. 19, 1994, page 1
NOS news is good news
We looked at the areas where SMP hardware helped NOS performance — and where it didn't.

CONTRIBUTORS

Introduction by **Lisa Stapleton** Senior Editor, Enterprise Team and **Laura Wonnacott** Test Developer

Written by **Scott Mace** Senior Editor, LAN Team and **Ayse Sercan** Assistant Editor

Tests developed by **Laura Wonnacott**

Testing by **Jeff Symoens** and **Rod Chapin** Technical Analysts

Edited by **Scott Mace** and **Ayse Sercan**

Report Card

Symmetric multiprocessing servers

	Weighting	ALR Revolution Q-4SMP	AST Manhattan P5090	Compaq ProLiant 2000 5/90
		Advanced Logic Research Inc. Irvine, Calif. (800) 444-4257 or (714) 581-6770; E-mail: sales@alr.com	AST Research Inc. Irvine, Calif. (800) 876-4278 or (714) 727-4141; E-mail via CompuServe: GO ASTSUPPORT	Compaq Computer Corp. Houston, Texas (800) 345-1518 or (713) 370-0670; E-mail via World Wide Web: http://www.compaq/com/
Performance				
Scalability	150	*Very Good* 112.50 92.85 percent gave the Revolution second place.	*Good* 93.75 The Manhattan's 81.60 percent put it in third place.	*Excellent* 150.00 97.34 percent won top honors.
Transaction processing	200	*Good* 125.00 39.29 transactions per minute (tpm) made the Revolution a top on-line transaction processing (OLTP) performer.	*Satisfactory* 100.00 28.88 tpm was the best the Manhattan could do for last place.	*Satisfactory* 100.00 32.80 tpm put the ProLiant in second place.
Decision support (Times in hours:minutes:seconds)	125	*Very Good* 93.75 By finishing in 1:39:58, the Revolution picked up another first place.	*Satisfactory* 62.50 The Manhattan was in fourth place at 2:19:09.	*Satisfactory* 62.50 2:15:05 earned the ProLiant third place.
Setup and ease of use	100	*Good* 100.00 The included EISA utility is simple to use, but the disk configuration utility is more cumbersome. The two are not integrated and the disk can boot the system. The steps for creating a redundant array of independent disks (RAID) are not immediately apparent. The system does come preconfigured, though.	*Very Good* 75.00 AST doesn't preconfigure the system, but some resellers may. The disk configuration utility is GUI-based and requires a mouse, so the current drive configuration is represented in a nice chart. The Manhattan's configuration utilities are not integrated. The system could not boot up from the disk.	*Very Good* 75.00 The CD-ROM-based SmartStart system configuration utility is fully integrated, although it does not come pre-installed. SmartStart includes all supported operating systems with license activation codes for those purchased. The ProLiant system can boot from the CD-ROM and install the OS preconfigured to Compaq's specifications.
Expandability	100	*Very Good* 75.00 The Revolution was the only server we tested that could upgrade to four 100-MHz processors. The 32-bit system memory can be increased to a whopping 1 gigabyte (GB) and the video RAM (VRAM) to 2MB. Three of its 10 EISA slots have VESA local bus extensions, but the system lacks PCI slots. The server supports as much as 22GB using half-height drives or 30GB using hot-swappable drives.	*Good* 62.50 The Manhattan holds a maximum of two 90-MHz Pentium processors. It was the only system to offer only 256KB of cache RAM, instead of 512KB. It can hold 256MB of 32-bit system memory and 512KB of video memory. It has six EISA and two PCI slots (one slot is shared), but one PCI slot is used by the disk array controller that comes standard. The server supports as much as 32GB of storage (hot-swappable).	*Very Good* 75.00 The ProLiant can hold four 90-MHz Pentiums, although going from two to four CPUs required that we move the RAID controllers used in our configuration to an optional ProLiant Storage System. The ProLiant can hold as much as 512MB of 32-bit system memory and 1MB of video memory. It has eight EISA slots but no PCI slots. The server only supports as much as 10GB of disk storage.
System design	75	*Good* 46.88 This system had many of the features we found important, especially a dedicated cache for each CPU. It can use standard memory and includes error detection and correction. Its score was hurt by poorly designed doors and sticky hot-swappable drives. There are two conjunctive power supplies (redundant power supplies are optional) and seven fans. Gaining access to the memory is awkward. Access to drive bays is on the opposite side of the system board.	*Good* 46.88 The Manhattan's design was average, with a shared cache and a single locking door, which was somewhat hard to open and close. The first 16MB of memory features error-correcting code (ECC); AST provides ECC in other components of the system, including cabling. When the single door is closed and locked, access to the side panel screws is restricted. The system has one 300-watt power supply with its own fan; each CPU has a heat sink with a fan; and there is a central cooling fan.	*Good* 46.88 The ProLiant server was the only system besides the Revolution with a dedicated cache for each CPU. The first 32MB of system memory includes ECC. The locking front door swings open and can be detached from the system. The case is designed so the power cannot be on when the front door is open. The side slides off for access to the processors and slots. The system has one power supply, four cooling fans, and heat sinks on the processors. The drives have relatively easy access and glide smoothly in and out of the array chassis. The bus slots and SIMM banks are easy to reach.
Support and pricing				
Documentation	50	*Very Good* 37.50 The Revolution's documentation is well laid out and easy to read, but it's missing some information. The manual has many diagrams and a nice troubleshooting guide.	*Excellent* 50.00 AST's documentation is easy to read and well laid out. It includes an extensive troubleshooting section with a list of error codes, plus a section on system disassembly.	*Very Good* 37.50 Compaq's written documentation is excellent. Although it didn't contain illustrations, step-by-step instructions, or diagrams, it earned extra points for its on-line reference library.
Support policies	50	*Excellent* 50.00 ALR was one of only two vendors without 24-hour, 7-day telephone support. A five-year warranty helps. ALR does offer support on the World Wide Web.	*Excellent* 50.00 AST offers a three-year warranty, free on-site support, and an optional 4-hour response by Memorex Telex. Unlimited free support is available 24 hours a day, 7 days a week through the toll-free line, or via fax, private BBS, CompuServe, and Prodigy.	*Excellent* 50.00 Unlimited free technical support is available. Compaq also offers a three-year, next-day on-site parts and labor warranty, with an optional 4-hour response. Support is available via on-line services. Under the Prefailure Warranty, Compaq will replace any part operating below par before it actually fails.
Technical support	50	*Satisfactory* 25.00 The technical support staff was always available, with minimal hold times, but not all staff members would answer our questions. We sometimes had to call back and ask for someone with specific expertise.	*Good* 31.25 Getting through to technical support was the only pitfall. We had busy signals on several calls and an average 3-minute hold time. Once we got through, the support staff was very pleasant, knowledgeable, and willing to help.	*Satisfactory* 25.00 We waited several minutes on hold for a technical support person. The quality of support varied. Although some technicians were more knowledgeable than others, they all solved our problems. Some were patient and helped us through willingly; others were more abrupt.
Price	100	*Good* 62.50 The price was $22,351 configured as tested with two 100-MHz CPUs. (A base system with one 100-MHz CPU, a 256KB cache, and 16MB of RAM costs $6,995.)	*Good* 62.50 The price was $22,620 configured as tested. (A base system with one 90-MHz CPU, a 256KB cache, 16MB of RAM, and two 1GB drives costs $8,976.)	*Satisfactory* 50.00 The price was $26,999 configured as tested. (Compaq builds to order and had no base price.)
Final score		**7.3**	**6.3**	**6.7**

GUIDE

Rating
Score in points

InfoWorld reviews only finished, production versions of products, never beta-test versions.

Products receive ratings ranging from unacceptable to excellent in various categories. Scores are derived by multiplying the weighting of each criterion by its rating, where:

Excellent = 1.0 - Outstanding in all areas.
Very Good = 0.75 - Meets all essential criteria and offers significant advantages.
Good = 0.625 - Meets essential criteria and includes some special features.
Satisfactory = 0.5 - Meets essential criteria.
Poor = 0.25 - Falls short in essential areas.
Unacceptable or N/A = 0.0 - Fails to meet minimum standards or lacks this feature.

Scores are summed, divided by 100, and rounded down to one decimal place to yield the final score out of a maximum possible score of 10 (plus bonus). Products rated within 0.2 points of one another differ little. Weightings represent average relative importance to *InfoWorld* readers involved in purchasing and using that product category. You can customize the Report Card to your company's needs by using your own weightings to calculate the final score.

The Test Center Hot Pick is *InfoWorld*'s new award for outstanding products we have evaluated in scored stand-alone reviews or product comparisons. To receive the Test Center Hot Pick seal, a product has to offer what *InfoWorld* deems to be a standout feature or technology that is unusually valuable or revolutionary compared to competitors. The product must also score at least satisfactory in all Report Card categories and receive a final score of 7.0 or more.

ell PowerEdge XE 590-2	Polywell Poly 500EP2
ell Computer Corp. ustin, Texas 00) 274-3355 or (512) 338-4400; mail: sales@us.dell.com	Polywell Computers Inc. South San Francisco, Calif. (800) 999-1278 or (415) 583-7222; E-mail: polywell@lx.netcom.com
Good ⊖93.75 .70 percent was fourth best.	Good ⊖93.75 77.68 percent left the Poly in last place.
Satisfactory ⊖100.00 .50 tpm barely kept the PowerEdge out of last ice.	Satisfactory ⊖100.00 32.03 tpm put the Poly smack in the middle.
Poor ⊖31.25 lismal 3:04:41 on this test was the PowerEdge's tet to dead last place.	Good ⊖78.13 The Poly finished our I/O tests with a second-best time of 2:07:17.
iood ⊖62.50 PowerEdge comes unconfigured, but it can be configured by the dealer at an extra cost. Switch tings are fairly accessible. We had to run each con- iration utility separately from the command line. RAID configuration utility is not integrated with and the system configuration. The system could boot from the configuration disk.	Good ⊖62.50 The Poly came with Windows NT Server 3.5 pre-installed. Its disk configuration and EISA utilities are not integrated, and the system cannot boot up from the configuration disks. But the utilities themselves are easy to use.
iood ⊖62.50 Dell can accommodate two 100-MHz Pentium essors, with 512MB of 64-bit system memory 2MB of video memory. The server supports as h as 24GB of storage. It has eight EISA slots and PCI slots, but the two buses share the space re only one card can go.	Good ⊖62.50 The Poly can accommodate as many as two 100-MHz Pentium processors. It holds as much as 512MB of 32-bit system memory and 4MB of video memory. It has four EISA and four PCI slots, though one of those slots can only use one type at a time. The server can support as much as 36GB of storage.
ry Good ⊖56.25 the PowerEdge's bus slots were totally accessible the top of the machine, making up for its shared and lack of ECC memory (which is an option). owerEdge doesn't have a lock or a cover for the r switch. Once the case is off, the bus slots are sible from the top of the machine, and the slide in and out of the array easily. The memory nly be reached by dismounting the backplane screws). The machine has two conjunctive r supplies and four fans. The heat sinks on the ack fans but monitor temperature.	Satisfactory ⊖37.50 The Poly has few refinements and the fewest features of the systems we tested. It has the standard shared cache and does not support ECC. The drive bays are nicely designed, though. Facing out from the front of the case, each bay has an individual keyed lock for security on its hot-swappable drive; there is a fan on each drive, as well. The drives slide in and out easily. The machine has two system fans; one is on the single power supply. The CPU heat sinks also have fans.
ort and pricing	
Good ⊖37.50 werEdge's manuals include lots of step-by- structions, flowcharts, and diagrams, and they arly written. Unfortunately, some features, such Disk Configuration Utility, are undocumented.	Good ⊖31.25 Polywell's documentation is average. A provided three-ring binder can hold all the manuals.
llent ⊖50.00 arranty covers parts for three years and s one-year, next-day on-site labor (provided Dell), plus optional 4-hour turnaround. ire support is unlimited, free, and available 24 hours a day, every day. Support is also e on-line.	Excellent ⊖50.00 The warranty covers five years parts and labor. On-site labor is $200 per year. Technical support is unlimited and free, weekdays from 7 a.m. to 6 p.m. Pacific time and Saturdays from 12 p.m. to 5 p.m. Polywell has a next-day replacement parts program and a money-back guarantee for first-time buyers.
Good ⊖37.50 a support representative, we had to first go an extensive voice-based menu. The techni- re very friendly and knowledgeable. They l extra hints and tips and walked us through res step by step.	Unacceptable ⊖0.00 We spent a lot of time playing phone tag with Polywell's technical support. When we did get through, the staff was less than helpful — even rude — and referred us to component makers.
ctory ⊖50.00 was $26,982 configured as tested. (A base ith one 90-MHz CPU, 8MB of RAM, and a ive costs $7,202.)	Excellent ⊖100.00 The price was $14,475 configured as tested.
	6.2

HOW WE TESTED

To test the five symmetric multiprocessing (SMP) servers in this comparison, we designed benchmark tests that would play to their primary strength — handling CPU-intensive work. We knew from prior testing that SMP servers could slow performance on a network that provides mostly file and print services. (See "Symmetric multiprocessing may not always boost performance," Dec. 19, 1994, page 77.) Additional research, based on discussions with vendors and results from our reader survey, confirmed that these SMPs are most effective when performing CPU-heavy processing chores such as transactions, CAD or CAM work, and statistical analysis.

We tested the SMPs using a tweaked version of the data used for testing the database servers' transaction-processing speed in our Nov. 14, 1994, comparison (page 128). To make the data as scalable and CPU-intensive as possible, we boosted the number of data lookups and calculations. For example, we increased the number of substring searches in our transactions, as well as line items per order, so the database server would not just fetch records but actually compare and manipulate them, processes bound by CPU performance. We eliminated "think" times — a few seconds between each transaction placed in the script to simulate users' pauses.

> ▸ Forty-eight percent of *InfoWorld* readers surveyed already own an SMP server.

SERVER CONFIGURATION. All of the servers we tested adhered to Intel Corp.'s 1.1 SMP specification. We asked each vendor to configure its server with two Pentium processors (dual 90-MHz or dual 100-MHz), 128MB of RAM, three network interface cards (NICs), and a CD-ROM. If a vendor failed to supply the NICs, we installed three of our own Microdyne Corp. NE3200s. We made sure the servers came with Pentium CPUs without the floating-point math error.

Each vendor's disk subsystem consisted of five 1-gigabyte (GB) drives (except Dell Computer Corp.'s — Dell could only configure its PowerEdge XE 590-2 server with five 2GB drives, because it had no 1GB drives at the time). Four of the five drives in each server were configured with RAID Level 5 to provide cost-effective fault tolerance for our database. In each server, we configured one drive without RAID outside the array to provide optimum performance for our workload.

Eighty-four percent of the respondents to our reader survey said they used a database engine with their SMP server. In addition, 70 percent either currently use or plan to implement Microsoft Corp.'s Windows NT 3.5 as their multiprocessing operating system. As a result, we chose NT 3.5 as the multiprocessing operating system for our benchmark tests and Microsoft's SQL Server 4.21a as our database engine. SQL Server is one of the best database servers we've reviewed, and it was a natural choice to easily test how well Windows NT scales.

If an SMP vendor typically installed the network operating system for its customers, we allowed it to install NT to our specification, which did not vary significantly from NT's default installation. We chose NetBEUI as our only network transport, because IPX's optimization for file-and-print services prevents it from fitting a database server-intensive test.

TEST WORLD CENTER

To optimize it for multiprocessing performance, we turned on two functions in SQL Server: Boost SQL Priority and Dedicated MP Performance, both of which let SQL Server know to use the second processor effectively. In addition, we allowed each vendor to tune one hardware-dependent parameter in SQL Server, called maximum asynchronous I/O, which determined the number of outstanding asynchronous requests at any one time in SQL Server. If that setting is too high or low, I/O performance suffers, according to Microsoft.

The RAID 5 array, which we formatted as an NT File System, housed our database test files. We placed the database's transaction log on the single drive outside the array to provide the best performance environment for our on-line transaction processing (OLTP) task. This optimization technique kept the activity of writing the transaction log from interfering with OLTP.

WORKSTATION CONFIGURATION. We configured 40 workstations in four racks of 10. Each rack consisted of four Gateway 2000 Inc. 486/33s, one Dell 386/33, one Dell 486/25SX, and four Hewlett-Packard Co. 486/66s. All workstations contained 8MB of RAM and a 3Com Corp. 3C509 NIC, except the Dells, which the vendor equipped with Standard Microsystems Corp.'s SMC8000 NICs. We installed Microsoft's Network Client 3.0 for DOS on each client, configuring NetBEUI as the network transport. We installed DOS SQL Utilities on each client, configuring Named Pipes as the TSR to communicate with the network layer.

NETWORK CONFIGURATION. The nature of the workload we chose for the servers meant network bottlenecks were highly unlikely. An analysis of our SMP test revealed less than 1 percent network bandwidth utilization when running transactions on 40 clients. In our Dec. 19, 1994 NOS comparison (page 1), we used four network seg-

▸ How we tested, page 90

MEMORY PITFALLS

When buying a server, be careful about the kind of memory you have to use; it can hinder expandability. Some vendors require you to use error-correcting code (ECC), which can end up being very expensive. This is why others — like Dell — don't require it.

> ▸ *According to Compaq, its customers' four most important server requirements are dependability, easy management, ease of service, and a good price-to-performance ratio.*

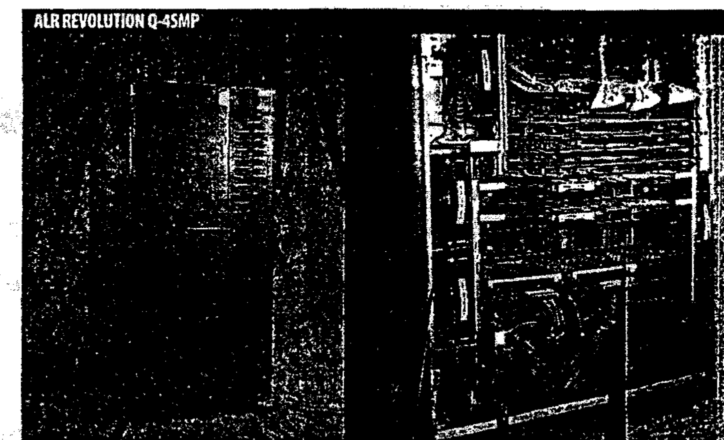
Designed for speed

▸ *System design can make or break a machine. To upgrade the ProLiant to more than two processors, for example, you have to remove your RAID array; you have to remove the cover from the PowerEdge for even the most routine operation. The Poly had easy accessibility in a plain box; the Manhattan echoed the Revolution's sleek black case, with a much higher quality door on the front.*

DRIVE-SWAPPING MADE EASY

Hot-swappable drives can be replaced while the machine is running, saving the administrator the time involved in arranging downtime and maximizing the users' access time. One caveat: It may not be easy to get at the drive that needs swapping.

ALR REVOLUTION Q-4SMP

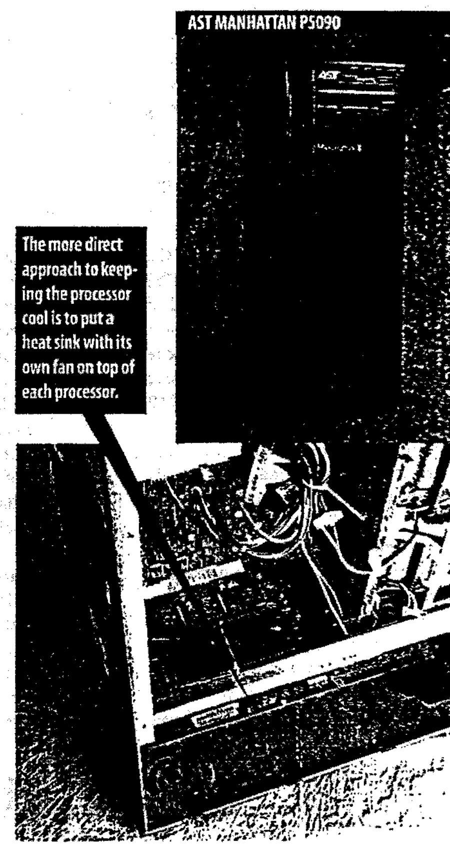


Front doors that lock make for a sleek-looking unit and also protect the server from accidental or unauthorized power-offs.

Plenty of fans keep the RAID array and two processors cool.

The Revolution, like the ProLiant, has room for a total of four processor boards.

AST MANHATTAN P5090



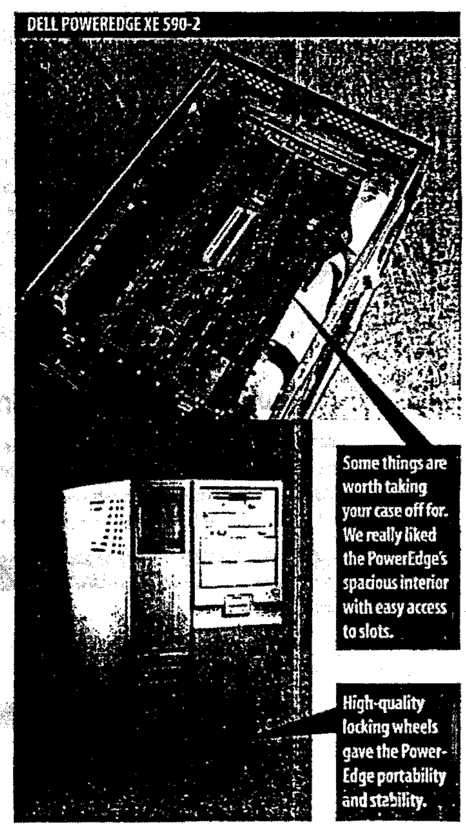
The more direct approach to keeping the processor cool is to put a heat sink with its own fan on top of each processor.

COMPAQ PROLIANT 2000 5/90

The ProLiant has room for two more processor boards if you take out this RAID array.

Opening the ProLiant's door makes the server shut off. Luckily, the door locks.

DELL POWEREDGE XE 590-2



Some things are worth taking your case off for. We really liked the PowerEdge's spacious interior with easy access to slots.

High-quality locking wheels gave the PowerEdge portability and stability.

▸ *Lots of blinking lights make for snazzy-looking machines, but because most servers spend their days locked in a closet, we didn't find LEDs useful.*

POLYWELL POLY 500EP2

Individually keyed hot-swappable drives are the Poly's only outstanding feature.

The Poly and the PowerEdge each have a shared bus slot, which can hold an EISA or a PCI card — but not both.

PRODUCT COMPARISON

MULTIPROCESSING MUSCLE: INSIDE SMP HARDWARE

By Laura Wonnacott

DRIVEN BY INCREASED DEMANDS on their networks, many system administrators are buying their first multiprocessing server. But it's not just a decision between Windows NT, OS/2 for Symmetric Multiprocessing, or the soon-to-be-released NetWare MP. Cache designs on multiprocessing servers are as different as condominiums, town houses, and ranch houses. Understanding fundamental design architectures can reduce a lot of the hassles for the first-time buyer.

Like single-processor Pentium PCs, symmetric multiprocessors (SMPs) come equipped with not only 16KB of on-board cache (on the chip), but also a secondary, external hardware cache called Level 2 cache to help eliminate processing bottlenecks. How the SMP uses this secondary cache varies, however, depending on whether it's a dual or multiprocessing machine. Dual-processor SMPs share a single large Level 2 cache, resulting in a less expensive system. Multiprocessor PCs, on the other hand, come with a dedicated Level 2 cache for each processor.

In a CPU-intensive environment, such as transaction processing, dedicated Level 2 cache allows more cache hits (times when a processor finds what it needs in the cache) than a shared Level 2 cache does. That's because when a cache is shared, processor contention (when both processors want access to the same information) is more likely to occur, resulting in one processor having to wait for the other processor to finish using the Level 2 cache.

The results of our on-line transaction processing tests confirm the speed benefits of a dedicated Level 2 cache. Two of the five servers in this comparison (the Compaq ProLiant 2000 5/90 and ALR Revolution Q-4SMP) and a RISC server, the NEC Technologies Inc. RISCserver 2200, which we did not score (see story, below) offer a dedicated Level 2 cache. Not surprisingly, these servers out-performed all others in our CPU-intensive transaction processing test. In addition, these servers proved the most scalable in moving from one to two processors.

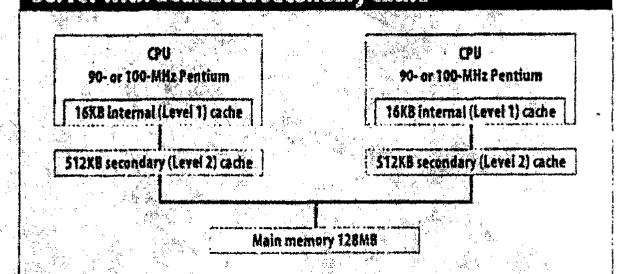
Given the two basic cache designs, there's still a lot a vendor can do to enhance processing performance. For example, larger caches are always helpful. The 2MB of Level 2 cache in the NEC RISCserver 2200 we tested no doubt was at least partially responsible for the machine's superior transaction processing performance.

The slower Intel-based servers typically had about 512KB of Level 2 cache. Other optimization techniques exist, such as Compaq's optional Transaction Blaster, which offers a third level of caching to further enhance processing performance.

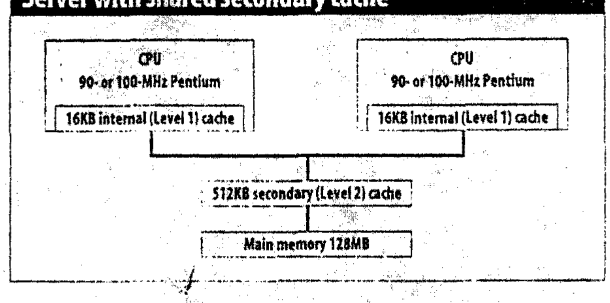
Multiprocessing muscle – two symmetric multiprocessing PC design architectures

We tested servers containing two cache designs. One with a dedicated secondary (Level 2) cache for each processor, the other with a shared secondary (Level 2) cache for all processors.

Server with dedicated secondary cache



Server with shared secondary cache



G. BOHEN / INFOWORLD

MULTIPROCESSING WITH NOVELL

Multiprocessing on Intel architecture is not limited to Windows NT users. Novell's SFT III, with the appropriate NetWare Loadable Module, can use two processors on systems that adhere to its specifications. When SFT III has two processors, it off-loads some of the communications workload (which can cause a significant drop in performance) to the second processor, regaining as much as 15 percent of lost performance.

▸ *Intel's MP specification gives buyers some assurance that future MP operating systems will perform well on their servers, though vendors can enhance this scalability by going beyond the spec on their system designs.*

The need for more speed: MIPS RISC is a fast Intel alternative

By Laura Wonnacott

Do you need even more processing speed? If you're willing to consider a non-Intel architecture, take a look at the RISC symmetric multiprocessing (SMP) architecture based on the chip from MIPS Technologies Inc. We tested a MIPS-based machine using the same test bed and script used for the SMPs in this comparison, and we're impressed.

The machine we tested, NEC Technologies Inc.'s RISCserver 2200, completed 50 transactions on 40 workstations in an average of 8 minutes and 53 seconds, 24 percent and 12 minutes ahead of the fastest two-Pentium server in the comparison, the ALR Revolution Q-4SMP. In fact, the RISCserver was just as fast processing transactions on one processor as the slowest Pentium on two processors (AST's Manhattan P5090).

The RISCserver's disk subsystem performance was not as impressive as its on-line transaction processing performance. We suspect that the server was experiencing some problems with its disk subsystem, because the server reset its SCSI bus several times during our tests.

What makes MIPS RISC servers so fast? One reason for the NEC RISCserver's performance is probably that Microsoft NT, the only operating system it will run, was developed on MIPS architecture. It's not surprising that NT performs well on its native platform. The 64-bit MIPS RISC processors also help; the RISCserver 2200 contains two 200-MHz MIPS CPUs. Its dedicated cache design is similar to the classic SMP architecture in the Compaq ProLiant 2000 5/90 and the Revolution (see story, above). The RISCserver has more Level 2 cache memory, though (2MB per CPU vs. the 512KB in the Intel machines).

In addition to 2MB of Level 2 cache, the caching mechanism does something called "cache snooping," which allows a processor to look for data in the other processors' Level 2 cache before going to main memory. The processor gains speed without hurting overall performance, because it can "snoop" without locking out the other processor from its own cache.

So why aren't buyers flocking to MIPS? It can't be the price. The NEC RISCserver 2200's estimated street price is $32,195 for our test configuration (with RAID Level 5). That's only about $5,000 more than the most expensive Intel-based server we reviewed (Dell's PowerEdge XE 590-2).

The difference in architecture no doubt has something to do with buyers' shyness. MIPS RISC is certainly not Intel. The utility programs have a different flavor from the old-style Intel-based utilities. For example, you can't boot from a floppy. The RISCserver starts based on what's stored in nonvolatile RAM and comes up with something called an ark menu, as defined by the MIPS specification. To change the system's configuration, you choose an option called Run Setup from the ark menu after booting. Setup contains most of the system's configuration. To change the EISA configuration, you'll need to choose Run Program and then specify the A: drive and program name.

▸ **RISC vendors such as Sun provide customized C++ compilers for building multi-threaded SMP applications.**

But moving from a single to multiple processors was a snap. Unlike with Intel, there's no need to load a different kernel when adding processors. Microsoft's NT and SQL Server on MIPS have an identical look and feel to that of their Intel versions. In fact, the support for Intel, MIPS, and Alpha is distributed on the same CD-ROM, so you may already have a copy of the MIPS version. The greatest difference with MIPS is in byte ordering, which you can't see from the interface. This means you can't merely move your SQL Server databases onto MIPS by copying the files. SQL Server includes an SQL Transfer Manager that lets you migrate data from one platform to another.

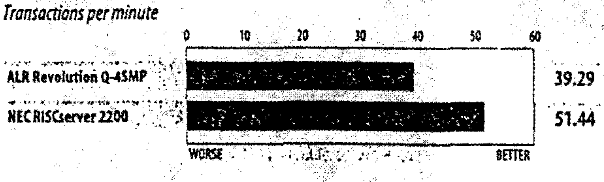
If you use Microsoft's NT and SQL Server, then MIPS RISC is a fast Intel alternative that doesn't cost much more.

RISC takers should give Pentiums another look

Pentium-based machines are finally approaching RISC performance. We pitted the strongest performer in this comparison, the ALR Revolution Q-4SMP, against NEC Technologies Inc.'s RISCserver 2200, which was still in beta when we ran our tests. The Revolution only did 24 percent fewer transactions per minute than the NEC RISC machine.

Fastest Pentium vs. NEC RISCserver 2200

Transactions per minute



▸ How we tested (from page 85)
ments to remove the cable as a bottleneck and target overall server performance. In this comparison, we used three segments, because with less I/O and more CPU-intensive tests, we didn't have as much network traffic.

We distributed three standard racks per network segment, and the fourth rack was distributed across the existing three segments. This isolated server CPU performance from other network performance variables.

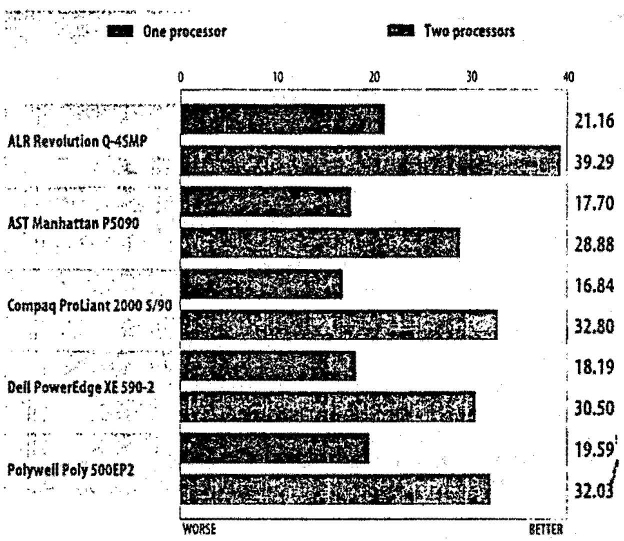
Each segment was supported by a Cabletron Systems Inc. Multi Media Access Center M8FNB concentrator.

Double your processors ...

... And nearly double your fun. The Revolution processed the largest number of transactions per minute on two processors – 39.29. But the ProLiant was the most scalable, nearly doubling its speed when we added the second processor.

Transaction processing

Transactions per minute

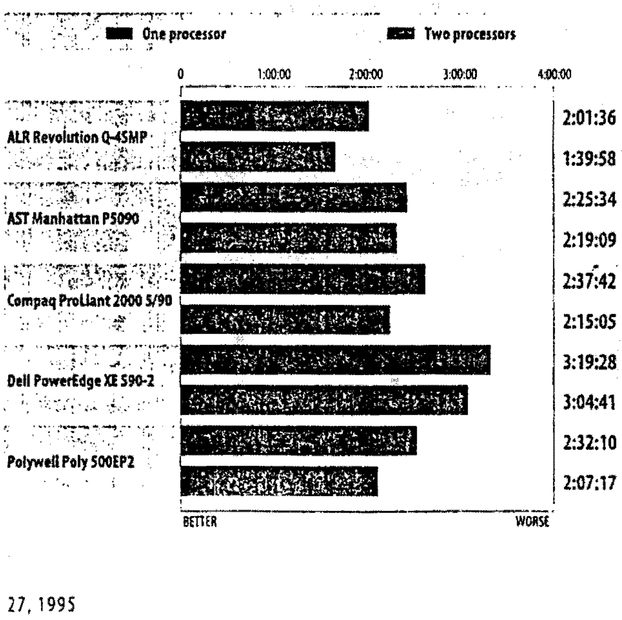


Can't make up their minds fast enough

If decision support makes up the bulk of your network load, think twice before trying to improve performance through symmetric multiprocessors. The Revolution scaled the best when we went to two processors to run our database queries, but none of the servers scaled nearly as well as they did in our on-line transaction processing tests.

Decision support

Time to complete queries on four workstations. Times in hours:minutes:seconds.



G. BOREN / INFOWORLD

PARALLEL VS. SYMMETRIC

Machines are not limited to symmetric multiprocessing; parallel architectures provide even better performance, at the expense of both price and flexibility of platforms.

Distributed processing works over distributed networks but puts a heavy dent in communications overhead — not good news for a network that's already overloaded. Overall, symmetric multiprocessing is the easiest, most flexible, and most cost-effective way to add multiprocessing performance to your network.

THE TESTS. We based the scores for our performance categories (scalability, transaction processing, and decision support) on transaction-processing and query benchmark tests performed on the 40-workstation network.

The transaction-processing script simulated an on-line order-entry system by processing 50 transactions simultaneously across the 40 clients. The queries accessed the same database from four workstations after they processed transactions. We ran the transaction-processing script twice to measure scalability from one to two processors and averaged the results.

Each transaction looked up a customer by identification number or by searching for the name in our customer table. Next, we calculated the next invoice number and created an order by inserting a row into an orders table.

For each part or line item a customer wished to order, we searched our parts table by either an exact part number or a partial description of the part name as supplied by the customer. We updated several quantity fields, such as amount on hand and on back order, if applicable. We then inserted a row into our "parts ordered" table for each line item a customer wished to order. We created an invoice form and updated the sales commission for the appropriate sales representative.

> ▸ Some SMP servers can have more than 1 terabyte of disk space if you add external drive cages via SCSI.

After the 40 workstations completed their transactions, four began processing I/O-bound database requests — our queries. The first workstation processed two sales queries. The second workstation processed two ad-hoc queries. The third workstation selected a set of orders from the orders table and inserted it into a temporary table. The fourth workstation processed a large select query from our parts-ordered table and sorted the results by part number. All the workstations sent query results to the server's disk to test disk subsystem performance.

PERFORMANCE

Scalability

We determined how scalable each server was by measuring its performance after we added a second processor. We first ran our 50 transactions on the 40 clients with the server running NT's uniprocessor kernel. We then ran 50 transactions on 40 clients with the server running NT's multiprocessor kernel with two processors enabled.

We scored scalability as a percentage. Perfect scalability was 100 percent. For example, if a server that ran 50 transactions on a single processor in one hour scaled perfectly, it would complete 50 transactions on two processors in 30 minutes. A server that scaled more than 95 percent we rated excellent; 95 percent to 86 percent earned a very good score; 85 percent to 76 percent earned a good; 75 percent to 66 earned a satisfactory; 65 percent to 56 percent earned a poor; and a server that scaled less than 56 percent was unacceptable.

Transaction processing

To determine test results for each server, we first calculated the average time to run 50 transactions on 40 clients. We then determined the transactions per minute (tpm) for each client and multiplied that by 40 (total number of clients) to arrive at the server's tpm.

A server that processed greater than 49 tpm earned a score of excellent; 49 to 40 tpm earned a score of very good; 39 to 35 tpm earned a score of good; 34 to 25 tpm earned a score of satisfactory; 24 to 20 tpm earned a score of poor; and a server that processed less than 20 tpm was unacceptable.

Decision support

We designed this benchmark test to show how well each server could handle I/O-intensive work on a network that served both decision-support and CPU-intensive requests. We first calculated the average time it took each of the four workstations on our 40-client network to complete our queries. We then figured the averages.

A server that completed our queries in less than 1 hour and 15 minutes received an excellent; a time between 1 hour and 15 minutes and 1 hour and 45 minutes received a very good; between 1 hour and 45 minutes and 2 hours and 15 minutes earned a good; between 2 hours and 15 minutes and 2 hours and 45 minutes earned a satisfactory; between 2 hours and 45 minutes and 3 hours and 15 minutes earned a poor. Anything slower got an unacceptable.

Setup and ease of use

We attempted to capture the experience of setting up the server out of the box. We evaluated how easy it was to get the server running on the network, paying careful attention to both the EISA configuration and the RAID disk subsystem utilities. For a product to receive a score of excellent, the EISA configuration had to be completed, the disk subsystem initialized and operative, and the operating system installed. A server that we could set up by following a few uncomplicated tasks received a score of very good.

Expandability

We looked for expandable server components. The more system memory, cache, slots, drive bays, and different types of hardware buses the server could accommodate after our configuration, the higher the score.

For consistency, we defined a drive as external if we did not have to remove a case, even if it was protected behind a door on the server itself.

System design

We carefully examined each server to determine any significant design advantages or flaws. Servers that offered more than one kind of bus, integrated hard drive interfaces on the system board, and patch-free system boards scored the highest. We also gave bonuses for easy access to parts, dedicated CPU cache designs, error-correcting code (ECC) memory, or hot-swappable drive arrays. We noted how easy it was to add processors or memory to the system. Cases that were hard to open, parts that were difficult to reach, or the lack of adequate fans hurt the score.

Compatibility

SMP servers typically support a variety of operating systems. The more operating systems a server supported, the higher the score.

SUPPORT AND PRICING

Documentation

We looked for clear and concise documentation. We awarded a score of satisfactory if the documentation explained how to set up and configure the server. We also required it to include accurate illustrations and diagrams.

Support policies

We gave a satisfactory score for unlimited free support and a one-year warranty. We gave bonus points for support via fax, on-line services, a money-back guarantee, extended hours, and a toll-free line. We subtracted points for limited or no support.

Technical support

We based technical support scores on the quality of service we received during multiple anonymous calls and on the availability of knowledgeable personnel. We awarded bonus points for extra helpfulness. We subtracted points for unreturned calls or long waits on hold.

Price

We based this score on the price of the server as configured for this comparison, except for the three server NICs (which we omitted because several vendors did not provide NICs). We used the vendor's suggested street price, when available, or suggested retail price. Servers that cost less than $15,000 received an excellent; those that cost $15,001 to $19,999 rated very good; those that cost $20,000 to $24,999 rated good; those that cost $25,000 to $29,999 rated satisfactory; and those that cost more than $30,000 received a score of poor.

SCALABILITY: YOU NEED MORE THAN JUST GOOD HARDWARE

'y Laura Wonnacott

OUR TESTS SHOW that Windows NT scales well (at least as far as four processors; see story, right), but a scalable operating system and server hardware are not ıough to guarantee scalability. Applications must also be designed with scal- ility as an underlying objective. A ıorly designed application executes :necessary code, wasting precious ɔcessing resources.

Writing scalable code begins by ındoning the traditional mindset, in ich code starts at the top and finish- ıt the bottom. An application devel- :r must analyze programs to deter- ıe which portions of code can be cessed simultaneously by different Js.

hreads, the basic unit of execution ıultiprocessor applications, are the to this objective of parallel design. :ads can run on any processor in a :iprocessor system. Splitting a sin- thread into multiple concurrent ıds is a great way to boost a multi- essing server. Older, more tradi- ıl applications often require a :ss to finish before they continue to ıext one. In these older systems, cations and processors cannot the load on a single unit of work, large query.

gle-threading applications will rm the same no matter how many processors are used. The only way to realize a performance gain with a multiprocessor system is to use a multithreading application. Microsoft's SQL Server 4.21a, a multithreaded application designed to take advantage of additional processors, played a vital role in the scalability we saw in this comparison. We tweaked our on-line transaction processing application to better test scalability. Our original transaction wasn't a good test of multithreading, because its think times let the CPU sit idle.

Turning a single-threaded application into a multithreaded application requires a working knowledge of how threads work. Threads exist in three states — waiting (not ready to run), ready, or running. The number of runnable threads is limited by the system's resources; the number of threads running at once is limited by the number of processors in the system.

Many application developers (including us) still aren't familiar with all the programming techniques available for taking advantage of multiple processors. Windows NT provides a number of sophisticated synchronization objects, such as I/O completion ports, multiple synchronization objects, asynchronous I/O, and spinlocks. But as the demand for scalable applications increases, a working knowledge of these features will be essential to writing multiprocessing applications.

Microsoft's dirty little secret

By Laura Wonnacott

According to Microsoft Corp., SQL Server 4.21a supports as many as four processors, the same number the Windows NT 3.5 network operating system supports out of the box. But in some of our ad-hoc testing to see how well the two Microsoft products scaled beyond two processors, we discovered that SQL Server 4.21a's default configuration cripples it so that it's unable to use more than three processors — and Microsoft says changing the default is risky.

We had to conduct the scalability and speed tests in this comparison using only two processors, because that was the most supported by three of the five servers (AST's Manhattan P5090, Dell's PowerEdge XE 590-2, and Polywell's Poly 500EP2). Machines that share cache between two processors, instead of having dedicated cache for each CPU, can't be upgraded to four processors (see story, left). We wanted to see what NT 3.5 and SQL Server could do with the throttle open, so we fired them up on an available Compaq ProLiant 4000 5/66 with four 66-MHz Pentium CPUs and 128MB of RAM, and we ran our transaction processing benchmark test with one, two, three, and then four processors.

A server that scales perfectly doubles its performance by going to two processors. A perfect scale from two to three processors would improve performance 33 percent. Adding a fourth processor would increase performance over three processors only 25 percent but double the performance obtained with two processors, and so on. The scalability we witnessed using NT 3.5 and SQL Server beyond two processors was grim. As the graph below depicts, SQL Server seemed unable to find the third processor at all.

Puzzled, we checked our configuration carefully to make sure we had set SQL Server for dedicated multiprocessor performance. We had. It was only after several discussions with Microsoft that we found out about SMPStat, an undocumented and unsupported parameter set through NT's registry editor that declares the number of CPUs SQL Server can use.

In effect, SMPStat is SQL Server's throttle for multiprocessor performance. When we originally configured SQL Server for dedicated multiprocessor performance, the program automatically set SMPStat to zero, which tells SQL Server to use n-1 processors when the number of processors are greater than two. As a result, when we ran our test with three processors, SQL Server 4.21a did not take advantage of the third processor. When we ran our test with four processors, SQL Server used only three processors. We set SMPStat to -1, which tells SQL Server to use all available processors, and we reran our tests. With the proper setting, SQL Server came very close to perfect scalability.

According to Microsoft, fooling around with SMPStat could cause two program threads to eventually deadlock, bringing the database server to a halt. Had we run a battery of regression tests, we might have seen this happen, but we ran out of time to test, so we only have Microsoft's word to go on.

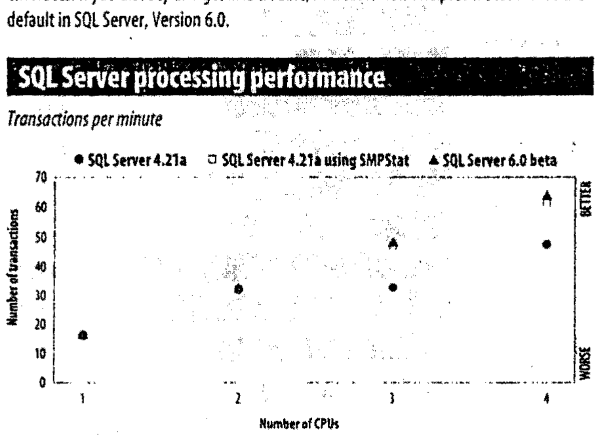
At any rate, SQL Server 6.0, now in beta testing and due the first half of this year, will come with SMPStat set to use all available processors without the risk of a deadlock, according to Microsoft. We tested a beta of Version 6.0 and were able to verify Microsoft's claim — the SQL Server 6.0 beta scaled similarly to SQL Server 4.21a with the SMPStat value set to -1. An interesting aside: The SQL Server 6.0 beta performed our test slightly faster on all multiprocessors than SQL Server 4.21a with SMPStat optimized for performance.

Who knows what CPUs lurk in the heart of SQL Server?

If you liked your Dick Tracy decoder ring, you'll like Microsoft SQL Server 4.21a. This version of SQL Server has a sneaky little undocumented tweak called SMPStat, which tells the machine to use all the available processors. If you don't alter SMPStat correctly, SQL Server 4.21a won't take full advantage of more than two processors. The catch: Microsoft doesn't want you to use SMPStat, which it says could result in deadly embraces. If you disobey and get into trouble, Microsoft won't help. SMPStat will be the default in SQL Server, Version 6.0.

SQL Server processing performance

Transactions per minute



SYMMETRIC MULTIPROCESSING SERVERS

KNOW YOUR RAID LEVELS

RAID 0 is simple data striping; RAID 1, simple disk mirroring. RAID 2 stripes data on mirrored disks; in RAID 3, bytes of data are striped across disks, with one drive storing parity information. With RAID 4, blocks of data are striped across disks, and one drive stores parity information; and in RAID 5, blocks of data and parity information are striped across all drives.

▸ *Each channel on a SCSI controller is the equivalent of a complete controller. By putting two or three channels on one board, you can get the benefits of having several boards on the system without losing the actual slots.*

ıtroduce
-bound
ivities into
ır appli-
ion and
tch scala-
ty run out
ŗas. Add
cessors to a
ver running
I/O-bound
plication,
d you'll
bably
ate even
re of an
bottle-
k.

PRODUCT COMPARISON

Support

	ALR Revolution Q-4SMP	AST Manhattan P5090	Compaq ProLiant 2000 5/90	Dell PowerEdge XE 590-2	Polywell Poly 500EP2
Free telephone support provided by vendor	Yes	Yes	Yes	Yes	Yes
Telephone support hours	Weekdays 6 a.m. to 6 p.m., Saturday 7 a.m. to 1 p.m. Pacific time	24 hours per day, 7 days per week	24 hours per day, 7 days per week	24 hours per day, 7 days per week	Weekdays 7 a.m. to 6 p.m., Saturday noon to 5 p.m. Pacific time
Warranty period	5 years on ALR motherboard, chassis, and parts; 3 years on third-party peripherals, drives, and parts	3 years with on-site support; optional 4-hour response by Memorex Telex	3 years next-day, on-site parts and labor; optional 4-hour response	3 years parts; 1 year next-day on-site labor; optional 4-hour turnaround	5 years parts and labor; 2 years on third-party components
Vendor-provided on-site service	1 year[1]	3 years	3 years	1 year	None
Money-back guarantee	No	No	No	No	Yes
On-line support	In-house BBS, CompuServe, World Wide Web server	In-house BBS, CompuServe, Prodigy	In-house BBS, CompuServe, America Online, World Wide Web server	In-house BBS, CompuServe, World Wide Web server, FTP server	In-house BBS, Internet E-mail
Fax-back support	Yes	Yes	Yes	Yes	Yes
Support policies score	Excellent	Excellent	Excellent	Excellent	Excellent
Technical support score	Satisfactory	Good	Satisfactory	Very Good	Unacceptable

1. $9.95 registration fee.

Features

	ALR Revolution Q-4SMP	AST Manhattan P5090	Compaq ProLiant 2000 5/90	Dell PowerEdge XE 590-2	Polywell Poly 500EP2
Setup and ease of use					
Additional preconfigured components	3 ALR E-Net 32 Ethernet controllers	None	3 Compaq NetFlex Ethernet controllers	5 2GB drives	Znyx 4-port PCI Ethernet controller
Power supply monitoring	No	Voltage monitoring software (requires NetWare); also an LED on the unit	Insight Manager software monitors voltage, temperature	Dell SafeSite software monitors voltage, temperature	No
On-line diagnostics	No	Yes	Yes	Yes	Yes[1]
Expandability					
Maximum number of Pentiums	4 90-MHz or 4 100-MHz	2 90-MHz	4 90-MHz[2]	2 90-MHz or 2 100-MHz	2 100-MHz
Maximum 32-bit system memory	1GB using 64MB SIMMs	256MB	512MB	512MB using 64MB SIMMs (not sold by Dell)	512MB using 64MB or 128MB SIMMs
Maximum external cache RAM	512KB per CPU	256KB shared	512KB per CPU	512KB shared	512KB shared
Maximum video memory	2MB	512KB	1MB	2MB	4MB
EISA slots	10	6	8	8[3]	4[3]
PCI slots	0	2	0	2[3]	4[3]
Proprietary slots (VESA, CPU, memory)	4 (3 EISA slots have VESA local bus extension)	None	4 proprietary EISA (2 CPU boards, 1 memory, 1 modem slot)	None	None
Levels of RAID supported	0, 1, 5	0, 1, 5	0, 1, 4, 5, 10	0, 1, 4, 5, 10	0, 1, 2, 5, 6, 10
Number of free slots in our configuration	5 EISA	3 EISA, 1 PCI	4 EISA and 3 proprietary upgrade slots (2 CPU boards, 1 memory)	4 EISA, 2 PCI[3]	2 EISA, 1 PCI[3]
Total number of external drive bays	11 half-height (with hot-swap option, every 3 hot-swap bays use 2 half-height bays), 2 3.5-inch	8 half-height (6 hot-swappable), 1 3.5-inch	2 half-height	4 half-height	8 half-height (6 hot-swappable)
Total number of internal drive bays	None	None	5 half-height (hot-swappable)	8 (hot-swappable)	2 half-height
Number of free drive bays after configuration	4 half-height, 4 hot-swappable bays[4], and 1 3.5-inch; all external	1 external half-height, 1 external hot-swappable bay	1 external half-height	2 external half-height, 2 internal hot-swappable bays	2 internal half-height
System design[5]					
Integrated hard drive interface	1 IDE	No	Embedded SCSI port	Embedded SCSI port	None embedded but ships with an additional controller
Type of video display	SVGA	SVGA	SVGA	ATI Mach 32 chip on PCI bus	Diamond Viper PCI card with 2MB video RAM 1,280 by 1,024
Number and type of serial ports	2 9-pin	2 9-pin	2 9-pin	2 9-pin	1 25-pin, 1 9-pin
Number of parallel ports	1	1	1	1	1 on add-on board
Types and locations of external SCSI ports	1 SCSI-2 port on controller on the back of the RAID caching controller	1 SCSI-2 port on controller	1 external SCSI-2 port off the motherboard, 1 external SCSI port off the RAID controller	1 embedded SCSI port on motherboard, 1 external SCSI-2 port on back of the system, running off the RAID controller	1 external SCSI-2 port off the RAID controller, 1 external SCSI-2 port off the CD-ROM controller
Keyboard security features	Keyboard disable button, password control	Password control	Password control	Password control	Keyboard lock, password control
Case lock	Yes; one for each side	Yes	Yes	Yes	Each hot-swappable drive has an individual keyed lock. The case itself has no locks.
BIOS manufacturer	Phoenix	Phoenix/AST	Compaq	Phoenix	Award Modular BIOS, Version 4.50G
Error-correcting memory type, amount	128MB error detection and correction (EDC) using standard DRAM	16MB error-correcting code (ECC)	32MB ECC	32MB ECC (optional)	None
Shared or dedicated L2 cache, size	Dedicated cache, 512KB per CPU	Shared 256KB cache	Dedicated cache, 512KB per CPU	Shared 512KB cache	Shared 512KB cache
UPS support	No	No	Yes (optional)	No	Yes
LEDs on system	Power, hard drive activity, memory, local bus slave and master, EISA slave and master	Power, hard drive activity, temperature, voltage, 4-digit post display (behind door)	Power and drive LEDs for each drive	Power, failed drive, diagnostics test	Power, drive for system and each bay

1. We could not get the supplied diagnostics to run.
2. You must remove the RAID array and house it in a separate drive cage in order to fit more than two Pentium boards.
3. One EISA and one PCI card share a slot.

A5

VESA®

VUMA Proposal

(Draft)

Video Electronics Standards Association

2150 North First Street, Suite 440
San Jose, CA 95131-2029

Phone: (408) 435-0333
FAX: (408) 435-8225

VESA Unified Memory Architecture Hardware Specifications Proposal

Version: 1.0p

Document Revision: 0.4p

October 31, 1995

Important Notice: This is a **draft** document from the Video Electronics Standards Association (VESA) Unified Memory Architecture Committee (VUMA). It is only for discussion purposes within the committee and with any other persons or organizations that the committee has determined should be invited to review or otherwise contribute to it. It has not been presented or ratified by the VESA general membership.

Purpose

To enable core logic chipset and VUMA device designers to design VUMA devices supporting the Unified Memory Architecture.

Summary

This document contains a specification for VUMA devices' hardware interface. It includes logical and electrical interface specifications. The BIOS protocol is described in VESA document VUMA VESA BIOS Extensions (VUMA-SBE) rev. 1.0.

Scope

Because this is a draft document, it cannot be considered complete or accurate in all respects although every effort has been made to minimize errors.

Intellectual Property

Trademarks

All trademarks used in this document are the property of their respective owners. VESA and VUMA are trademarks owned by the Video Electronics Standards Association.

Patents

The proposals and standards developed and adopted by VESA are intended to promote uniformity and economies of scale in the video electronics industry. VESA strives for standards that will benefit both the industry and end users of video electronics products. VESA cannot ensure that the adoption of a standard; the use of a method described as a standard: or the making, using, or selling of a product in compliance with the standard does not infringe upon the intellectual property rights (including patents, trademarks, and copyrights) of others. VESA, therefore, makes no warranties, expressed or implied, that products conforming to a VESA standard do not infringe on the intellectual property rights of others, and accepts no liability direct, indirect or consequential, for any such infringement.

Support For This Specification

If you have a product that incorporates VUMA™, you should ask the company that manufactured your product for assistance. If you are a manufacturer of the product, VESA can assist you with any clarification that you may require. All questions must be sent in writing to VESA via:

(The following list is the preferred order for contacting VESA.)

VESA World Wide Web Page:	*www.vesa.org*
Fax:	(408) 435-8225
Mail:	VESA 2150 North First Street Suite 440 San Jose, California 95131-2029

Acknowledgments

This document would not have been possible without the efforts of the members of the 1995 VESA Unified Memory Architecture Committee and the professional support of the VESA staff.

Work Group Members

Any industry standard requires information from many sources. The following list recognizes members of the VUMA Committee, which was responsible for combining all of the industry input into this proposal.

Chairperson

Rajesh Shakkarwar	OPTi

Members

Jonathan Claman	S3
Jim Jirgal	VLSI Technology Inc.
Don Pannell	Sierra Semiconductor
Wallace Kou	Western Digital
Derek Johnson	Cypress
Andy Daniel	Alliance Semiconductor
Long Nguyen	Oak Technology
Robert Tsay	Pacific Micro Computing Inc.
Sunil Bhatia	Mentor Arc
Peter Cheng	Samsung
Alan Mormann	Micron
Solomon Alemayehu	Hitachi America Ltd.
Larry Alchesky	Mitsubishi
Dean Hays	Weitek

Revision History

Initial Revision 0.1p — Sept. 21 '95

Revision 0.2p — Oct 5 '95

- Added sync DRAM support
- Electrical Section
- Boot Protocol
- Reformatted document

Revision 0.3p — Oct 19 '95

- Graphics controller replaced with VUMA device
- MD[n:0] changed to t/s
- Modified Aux Memory description
- Added third solution to Memory Expansion Problem
- Synch DRAM burst length changed to 2/4
- Modified all the bus hand off diagrams
- Added DRAM Driver Characteristics section

Revision 0.4p — Oct 19 '95

- Sync DRAM Burst Length changed to 1/2/4
- DRAM controller pin multiplexing added
- Changed AC timing parameters

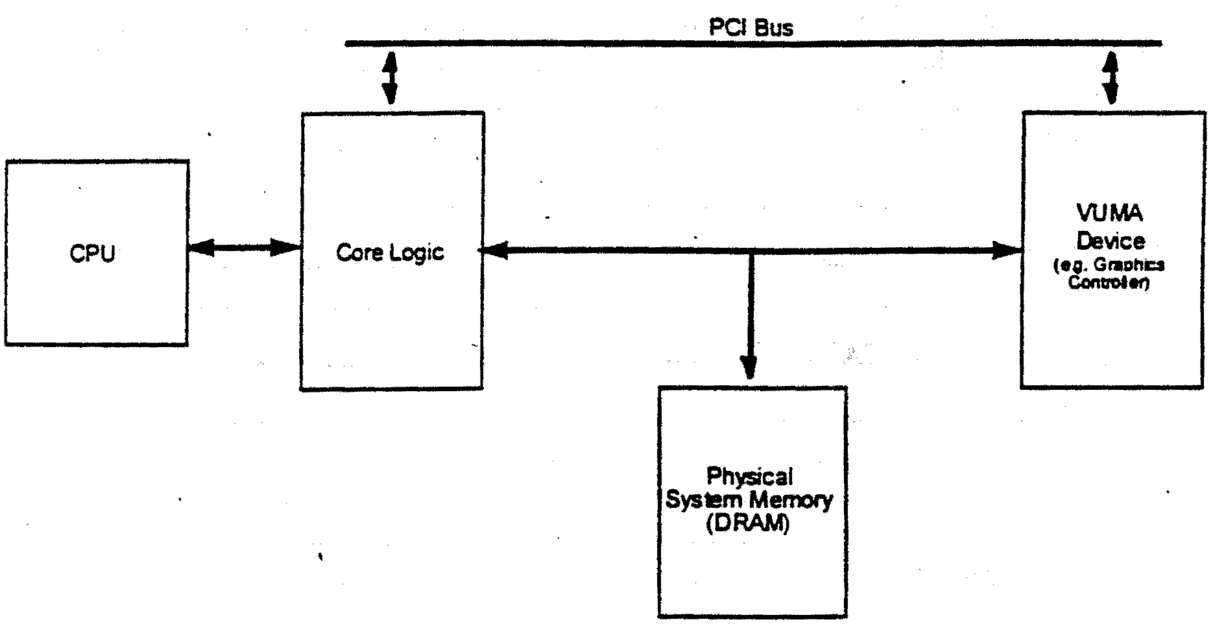
TABLE OF CONTENTS

1.0 Introduction

The concept of VESA Unified Memory Architecture (VUMA) is to share physical system memory (DRAM) between system and an external device, a VUMA device; as shown in Figure 1-1. A VUMA device could be any type of controller which needs to share physical system memory (DRAM) with system and directly access it. One example of a VUMA device is graphics controller. In a VUMA system, graphics controller will incorporate graphics frame buffer in physical system memory (DRAM) or in other words VUMA device will use a part of physical system memory as its frame buffer, thus, sharing it with system and directly accessing it. This will eliminate the need for separate graphics memory, resulting in cost savings. Memory sharing is achieved by physically connecting core logic chipset (hereafter referred to as core logic) and VUMA device to the same physical system memory DRAM pins. Though the current version covers sharing of physical system memory only between core logic and a motherboard VUMA device, the next version will cover an expansion connector, connected to physical system memory DRAM pins. An OEM will be able to connect any type of device to the physical system memory DRAM pins through the expansion connector.

Though a VUMA device could be any type of controller, the discussion in the specifications emphasizes a graphics controller as it will be the first VUMA system implementation.

Figure 1-1 VUMA System Block Diagram



2.0 Signal Definition

2.1 Signal Type Definition

in Input is a standard input-only signal.

out Totem Pole Output is a standard active driver

t/s Tri-State is a bi-directional, tri-state input/output pin.

s/t/s Sustained Tri-state is an active low or active high tri-state signal owned and driven by one and only one agent at a time. The agent that drives an s/t/s pin active must drive it high for at least one clock before letting it float. A pullup is required to sustain the high state until another agent drives it. Either internal or external pullup must be provided by core logic. A VUMA device can also optionally provide an internal or external pullup.

2.2 Arbitration Signals

MREQ#	**in** **out**	MREQ# is out for VUMA device and in for core logic. This signal is used by VUMA device to inform core logic that it needs to access shared physical system memory bus.
MGNT#	**in** **out**	MGNT# is out for core logic and in for VUMA device. This signal is used by core logic to inform VUMA device that it can access shared physical system memory bus.
CPUCLK	**in**	CPUCLK is driven by a clock driver. CPUCLK is in for core logic, VUMA device and synchronous DRAM.

2.3 Fast Page Mode, EDO and BEDO DRAMs

RAS#	s/t/s	Active low row address strobe for memory banks. Core logic will have multiple RAS#s to support multiple banks. VUMA device could have a single RAS# or multiple RAS#s. These signals are shared by core logic and VUMA device. They are driven by current bus master.
CAS[n:0]#	s/t/s	Active low column address strobes, one for each byte lane. In case of pentium-class systems n is 7. These signals are shared by core logic and VUMA device. They are driven by current bus master.
WE#	s/t/s	Active low write enable. This signal is shared by core logic and VUMA device. It is driven by current bus master.
OE#	s/t/s	Active low output enable. This signal exists only on EDO and BEDO. This signal is shared by core logic and VUMA device.

		It is driven by current bus master.
MA[11:0]	s/t/s	Multiplexed memory address. These signals are shared by core logic and VUMA device. They are driven by current bus master.
MD[n:0]	t/s	Bi-directional memory data bus. In case of pentium-class systems n is 63. These signals are shared by core logic and VUMA device. They are driven by current bus master.

2.4 Synchronous DRAM

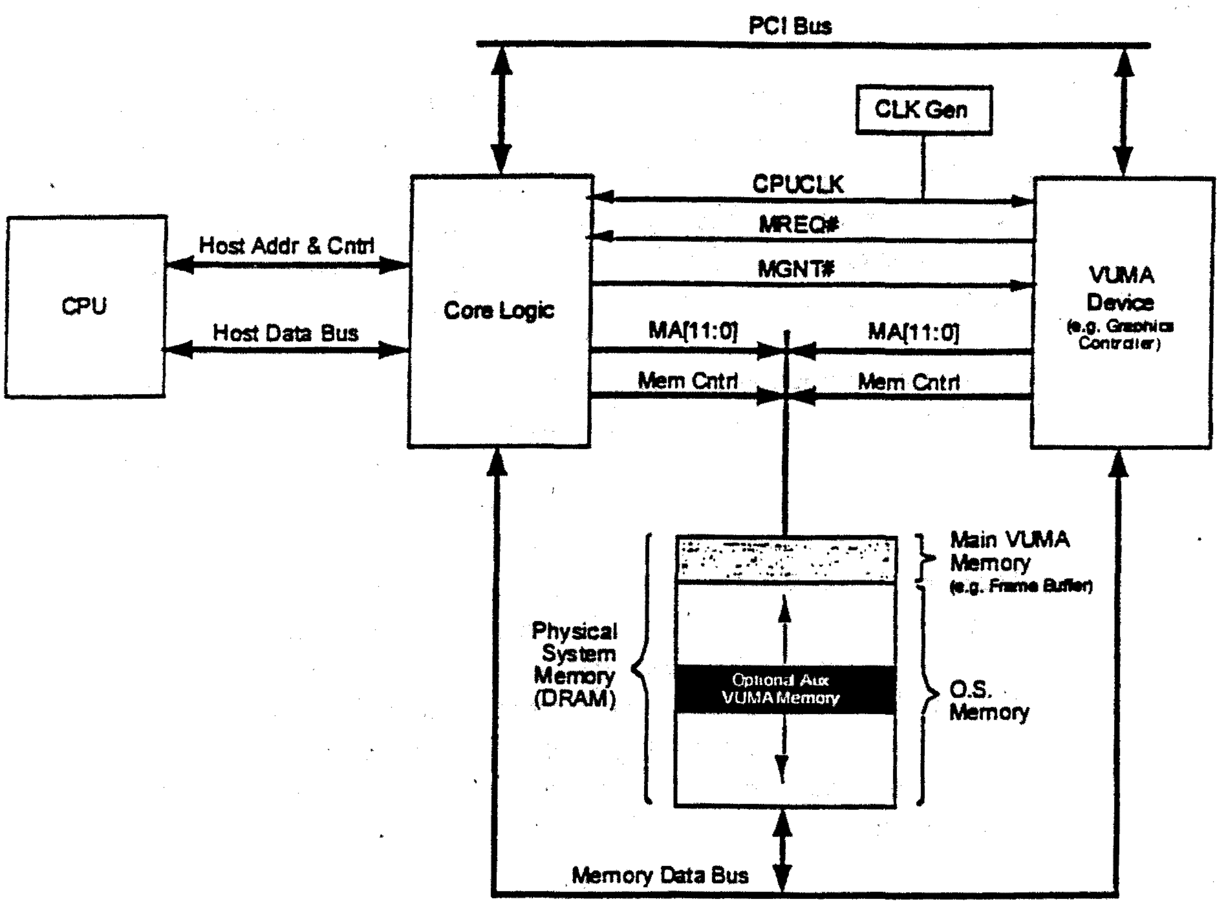
CPUCLK	in	CPUCLK is the master clock input (referred to as CLK in synchronous DRAM data books). All DRAM input/ output signals are referenced to the CPUCLK rising edge.
CKE	s/t/s	CKE determines validity of the next CPUCLK. If CKE is high, the next CPUCLK rising edge is valid; otherwise it is invalid. This signal also plays role in entering power down mode and refresh modes. This signal is shared by core logic and VUMA device. It is driven by current bus master.
CS#	s/t/s	CS# low starts the command input cycle. CS# is used to select a bank of Synchronous DRAM. Core logic will have multiple CS#s to support multiple banks. VUMA device could have a single CS# or multiple CS#s. These signals are shared by core logic and VUMA device. They are driven by current bus master.
RAS#	s/t/s	Active low row address strobe. This signal is shared by core logic and VUMA device. It is driven by current bus master.
CAS#	s/t/s	Active low column address strobe. This signal is shared by core logic and VUMA device. It is driven by current bus master.
WE#	s/t/s	Active low write enable. This signal is shared by core logic and VUMA device. It is driven by current bus master.
MA[11:0]	s/t/s	Multiplexed memory address. These signals are shared by core logic and VUMA device. They are driven by current bus master.
DQM[n:0]	s/t/s	I/O buffer control signals, one for each byte lane. In case of pentium-class systems n is 7. In read mode they control the output buffers like a conventional OE# pin. In write mode, they control the word mask. These signals are shared by core logic and VUMA device. They are driven by current bus master.
MD[n:0]	t/s	Bi-directional memory data bus. In case of pentium-class systems n is 63. These signals are shared by core logic and VUMA device. They are driven by current bus master.

3.0 Physical Interface

3.1 Physical System Memory Sharing

Figure 3-1 depicts the VUMA Block Diagram. Core logic and VUMA device are physically connected to the same DRAM pins. Since they share a common resource, they need to arbitrate for it. PCI/VL/ISA external masters also need to access the same DRAM resource. Core logic incorporates the arbiter and takes care of arbitration amongst various contenders.

Figure 3-1 VUMA Block Diagram



As shown in Figure 3-1, VUMA device arbitrates with core logic for access to the shared physical system memory through a three signal arbitration scheme viz. MREQ#, MGNT# and CPUCLK. MREQ# is a signal driven by VUMA device to core logic and MGNT# is a signal driven by the core logic to VUMA device. MREQ# and MGNT# are active low signals driven and sampled synchronous to CPUCLK common to both core logic and VUMA device.

Core logic is always the default owner and ownership will be transferred to VUMA device upon demand. VUMA device could return ownership to core logic upon completion of its activities or park on the bus. Core logic can always preempt VUMA device from the bus.

VUMA device needs to access the physical system memory for different reasons and the level of urgency of the needed accesses varies. If VUMA device is given the access to the physical system memory right away, every time it needs, the CPU performance will suffer and as it may not be needed right away by the VUMA device, there would not be any improvement in VUMA device performance. Hence two levels of priority are defined viz. low priority and high priority. Both priorities are conveyed to core logic through a single signal, MREQ#.

3.2 Memory Regions

As shown in Figure 3-1, physical system memory can contain two separate physical memory blocks, Main VUMA Memory and Auxiliary (Aux) VUMA Memory. As cache coherency for Main VUMA Memory and Auxiliary VUMA Memory is handled by this standard, a VUMA device can access these two physical memory blocks without any separate cache coherency considerations. If a VUMA device needs to access other regions of physical system memory, designers need to take care of cache coherency.

Main VUMA Memory is programmed as non-cacheable region to avoid cache coherency overhead. How Main VUMA Memory is used depends on the type of VUMA device; e.g., when VUMA device is a graphics controller, main VUMA memory will be used for Frame buffer.

Auxiliary VUMA Memory is optional for both core logic and VUMA device. If supported, it can be programmed as non-cacheable region or write-through region. Auxiliary VUMA Memory can be used to pass data between core logic and VUMA device without copying it to Main VUMA Memory or passing through a slower PCI bus. This capability would have significant advantages for more advanced devices. How Auxiliary VUMA Memory is used depends on the type of VUMA device e.g. when VUMA device is a 3D graphics controller, Auxiliary VUMA memory will be used for texture mapping.

When core logic programs Auxiliary VUMA Memory area as non-cacheable, VUMA device can read from or write to it. When core logic programs Auxiliary VUMA Memory area as write through, VUMA device can read from it but can not write to it.

Both core logic and VUMA device have an option of either supporting or not supporting the Auxiliary VUMA Memory feature. Whether Auxiliary VUMA memory is supported or not should be transparent to an application. The following algorithm explains how it is made transparent. The algorithm is only included to explain the feature. Refer to the latest VUMA VESA BIOS Extensions for the most updated BIOS calls:

1. When an application needs this feature, it needs to make a BIOS call, <Report VUMA

- core logic capabilities (refer to VUMA VESA BIOS Extensions)>, to find out if core logic supports the feature.
2. If core logic does not support the feature, the application needs to use some alternate method.
3. If core logic supports the feature, the application can probably use it and should do the following:

a. Request the operating system for a physically contiguous block of memory of required size.
b. If not successful in getting physically contiguous block of memory of required size, use some alternate method.
c. If successful, get the start address of the block of memory.
d. Read <VUMA BIOS signature string (refer to VUMA VESA BIOS Extensions)>, to find out if VUMA device can access the bank in which Auxiliary VUMA Memory has been assigned.
e. If VUMA device can not access that bank, the application needs to either retry the procedure from "step a" to "step c" till it can get Auxiliary VUMA Memory in a VUMA device accessible bank or use some alternate method.
f. If VUMA device can access that bank, make a BIOS call function <Set (Request) VUMA Auxiliary memory (refer to VUMA VESA BIOS Extensions)>, to ask core logic to flush Auxiliary VUMA Memory block of the needed size from the start address from "step c" and change it to either non-cacheable or write through. How a core logic flushes cache for the block of memory and programs it as non-cacheable/ write through is implementation specific.
g. Use VUMA Device Driver, to give VUMA device the Auxiliary VUMA Memory parameters viz. size, start address from "step c" and whether the block should be non-cacheable or write through.

3.3 Physical Connection

A VUMA device can be connected in two ways:

1. VUMA device can only access one bank of physical system memory - VUMA device is connected to a single bank of physical system memory. In case of Fast Page Mode, EDO and BEDO VUMA device has a single RAS#. In case of Synchronous DRAM VUMA device has a single CS#. Main VUMA memory resides in this memory bank. If supported, Auxiliary VUMA Memory can only be used if it is assigned to this bank.

2. VUMA device can access all of the physical system memory - VUMA device has as many RAS# (for Fast Page Mode, EDO and BEDO)/CS# (for Synchronous DRAM) lines as core logic and is connected to all banks of the physical system memory. Both Main VUMA memory and Auxiliary VUMA Memory (if supported) can be assigned to any memory bank.

4.0 Arbitration

4.1 Arbitration Protocol

There are three signals establishing the arbitration protocol between core logic and VUMA device. MREQ# signal is driven by VUMA device to core logic to indicate it needs to access the physical system memory bus. It also conveys the level of priority of the request. MGNT# is driven by core logic to VUMA device to indicate that it can access the physical system memory bus. Both MREQ# and MGNT# are driven synchronous to CPUCLK.

As shown in Figure 4-1, low level priority is conveyed by driving MREQ# low. A high level priority can only be generated by first generating a low priority request. As shown in Figure 4-2 and Figure 4-3, a low level priority is converted to a high level priority by driving MREQ# high for one CPUCLK clock and then driving it low.

Figure 4-1 Low Level Priority

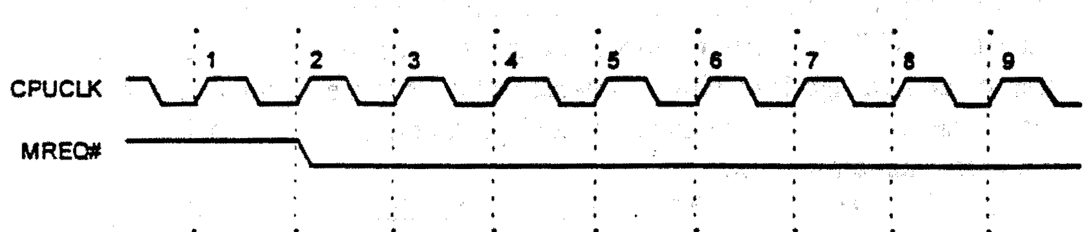


Figure 4-2 High Level Priority

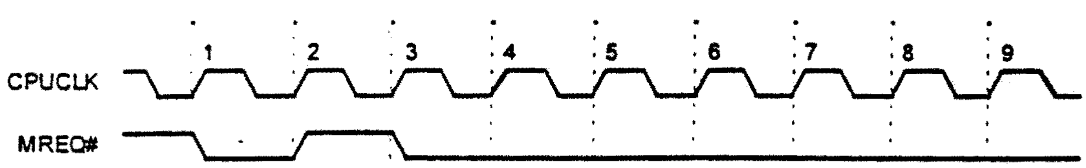
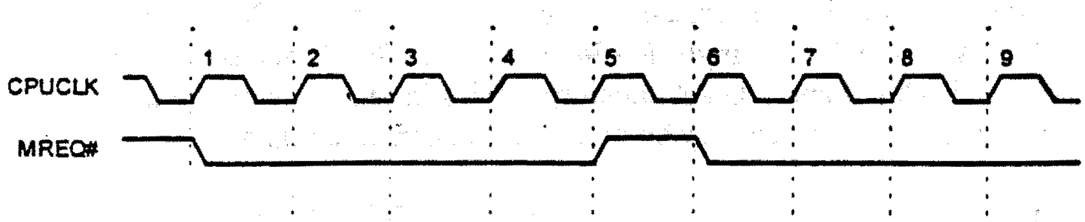


Figure 4-3 A Pending Low Level Priority converted to a High Level Priority



4.2 Arbiter

The arbiter, housed in core logic, needs to understand the arbitration protocol. State

Machine for the arbiter is depicted in Figure 4.4. As shown in Figure 4.4, the arbiter State Machine is resetted with PCI_Reset. Explanation of the arbiter is as follows:

1. HOST State - The physical system memory bus is with core logic and no bus request from VUMA device is pending.

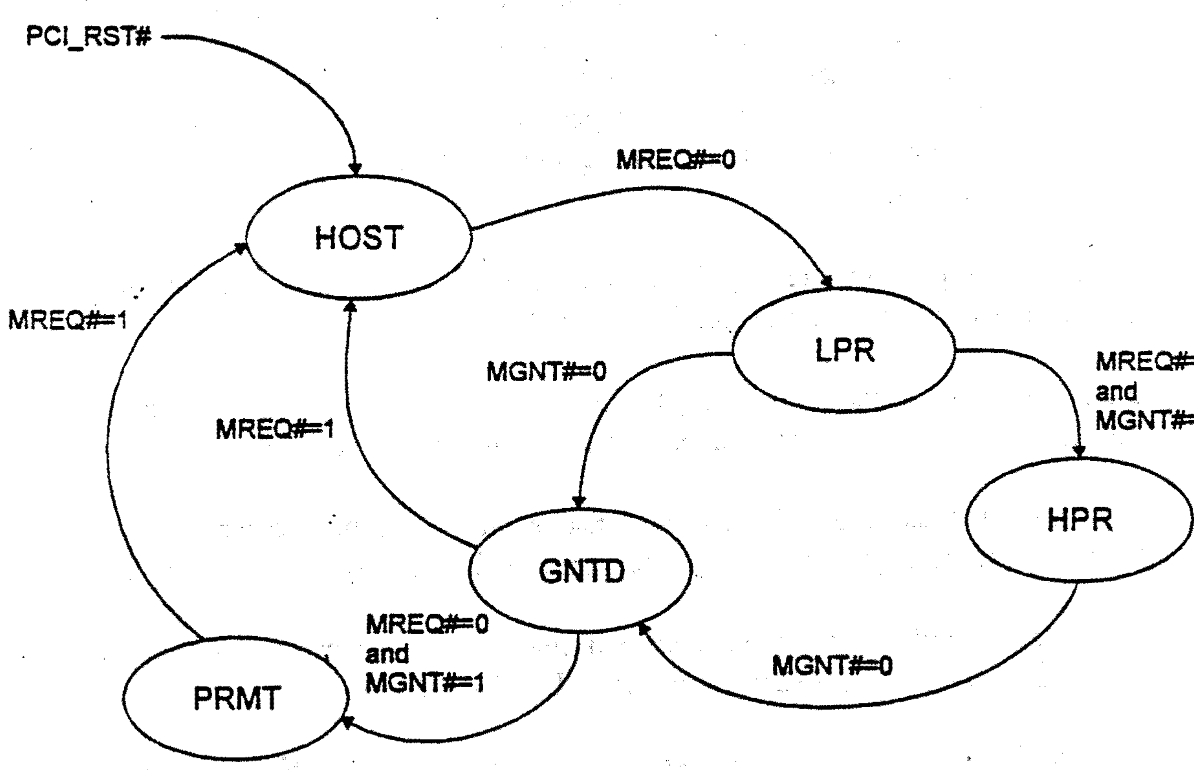
2. Low Priority Request (LPR) State - The physical system memory bus is with core logic and a low priority bus request from the VUMA device is pending.

3. High Priority Request (HPR) State - The physical system memory bus is with core logic and a pending low priority bus request has turned into a pending high priority bus request.

4. Granted (GNTD) State - Core logic has relinquished the physical system memory bus to VUMA device.

5. Preempt (PRMT) State - The physical system memory bus is owned by VUMA device, however, core logic has requested VUMA device to relinquish the bus and that request is pending.

Figure 4.4 Arbiter State Machine



Note:

1. Only the conditions which will cause a transition from one state to another have been

noted. Any other condition will keep the state machine in the current state.

4.2.1 Arbitration Rules

1. VUMA device asserts MREQ# to generate a low priority request and keeps it asserted until the VUMA device obtains ownership of the physical system memory bus through the assertion of MGNT#, unless the VUMA device wants to either raise a high priority request or raise the priority of an already pending low priority request. In the later case,

   a. If MGNT# is sampled asserted the VUMA device will not deassert MREQ#. Instead, the VUMA device will gain physical system memory bus ownership and maintain MREQ# asserted until it wants to relinquish the physical system memory bus.

   b. If MGNT# is sampled deasserted, the VUMA device will deassert MREQ# for one clock and assert it again irrespective of status of MGNT#. After reassertion, the VUMA device will keep MREQ# asserted until physical system memory bus ownership is transferred to the VUMA device through assertion of MGNT# signal.

2. VUMA device may assert MREQ# only for the purpose of accessing the unified memory area. Once asserted, MREQ# should not be deasserted before MGNT# assertion for any reason other than raising the priority of the request (i.e., low to high). No speculative request and request abortion is permitted. If MREQ# is deasserted to raise the priority, it should be reasserted in the next clock and kept asserted until MGNT# is sampled asserted.

3. Once MGNT# is sampled asserted by VUMA device, it gains and retains physical system memory bus ownership until MREQ# is deasserted.

4. The condition, VUMA device completing its required transactions before core logic needing the physical system memory bus back, can be handled in two ways:

   a. VUMA device can deassert MREQ#. In response, MGNT# will be deasserted in the next clock edge to change physical system memory bus ownership back to core logic.
   b. VUMA device can park on the physical system memory bus. If core logic needs the physical system memory bus, it should preempt VUMA device.

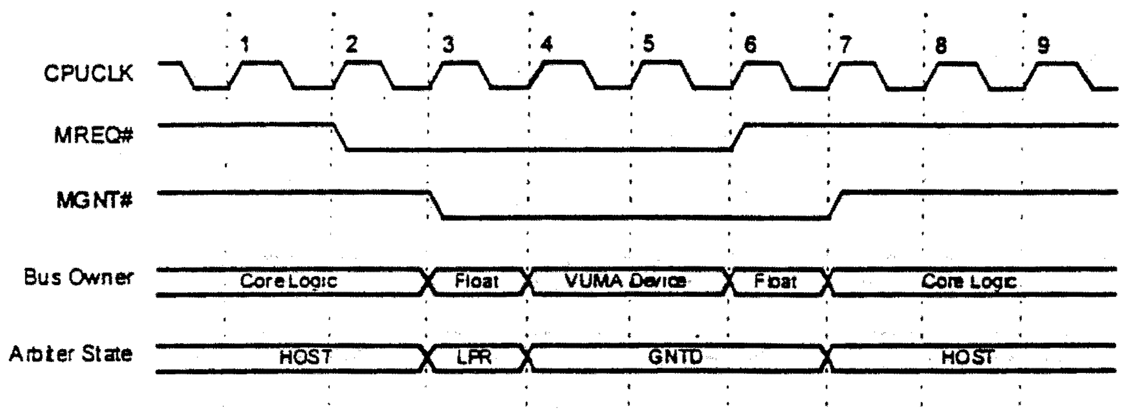
5. In case core logic needs the physical system memory bus before VUMA device releases it on its own, arbiter can preempt VUMA device from the bus. Preemption is signaled to VUMA device by deasserting MGNT#. VUMA device can retain ownership of the bus for a maximum of 60 CPUCLK clocks after it has been signaled

to preempt. VUMA device signals release of the physical system memory bus by deasserting MREQ#.

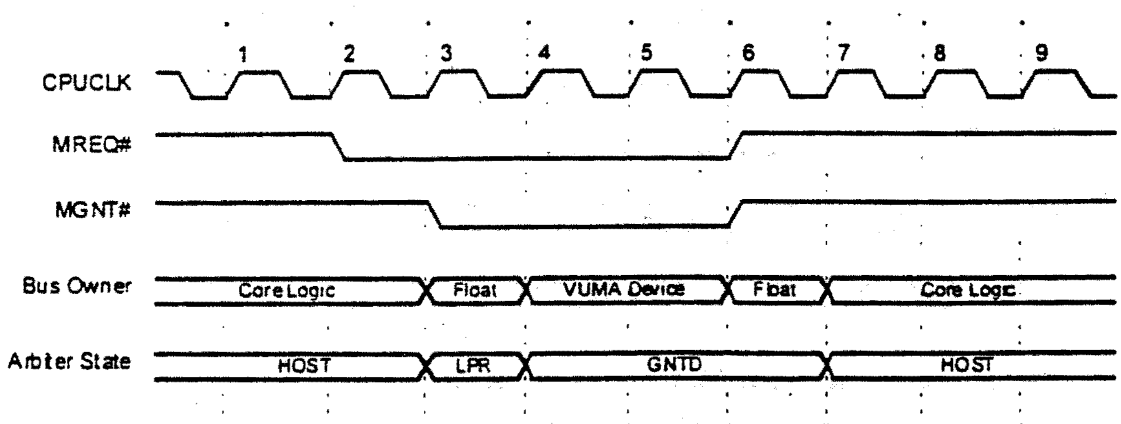
6. When VUMA device deasserts MREQ# to transfer bus ownership back to core logic, either on its own or because of a preemption request, it should keep MREQ# deasserted for at least two clocks of recovery time before asserting it again to raise a request.

4.3 Arbitration Examples

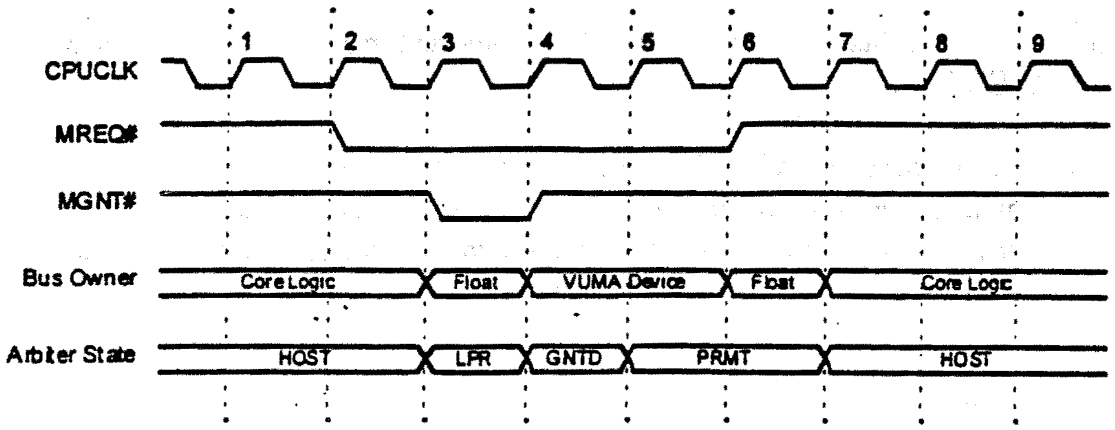
1. Low priority request and immediate bus release to VUMA device



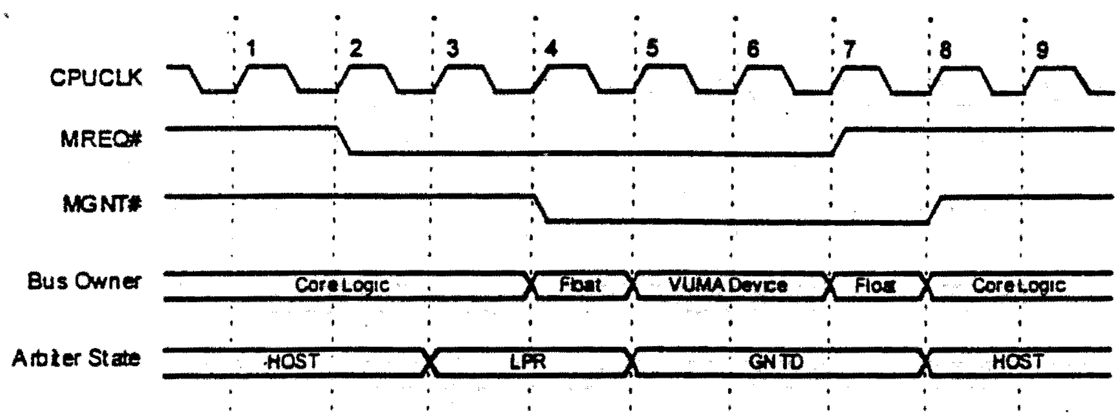
2. Low priority request and immediate bus release to VUMA device with preemption where removal of MGNT# and removal of MREQ# coincide



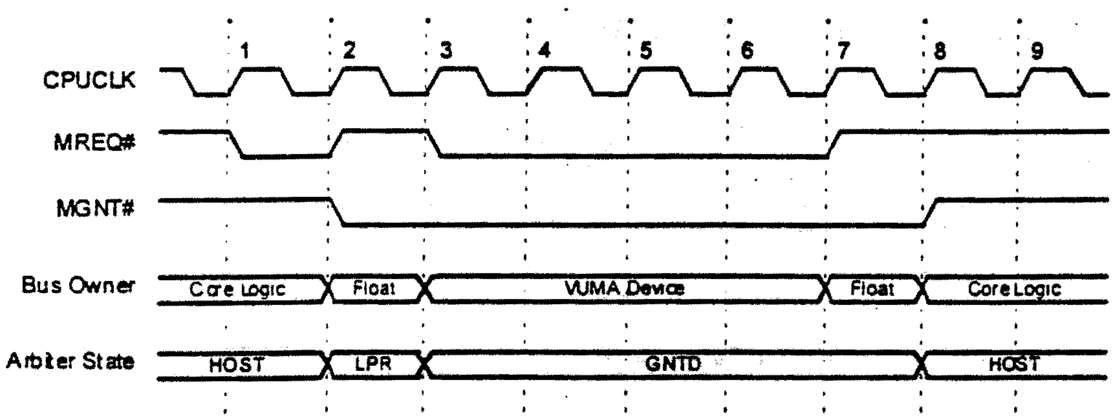
3. Low priority request and immediate bus release to VUMA device with preemption where MREQ# is removed after the current transaction because of preemption



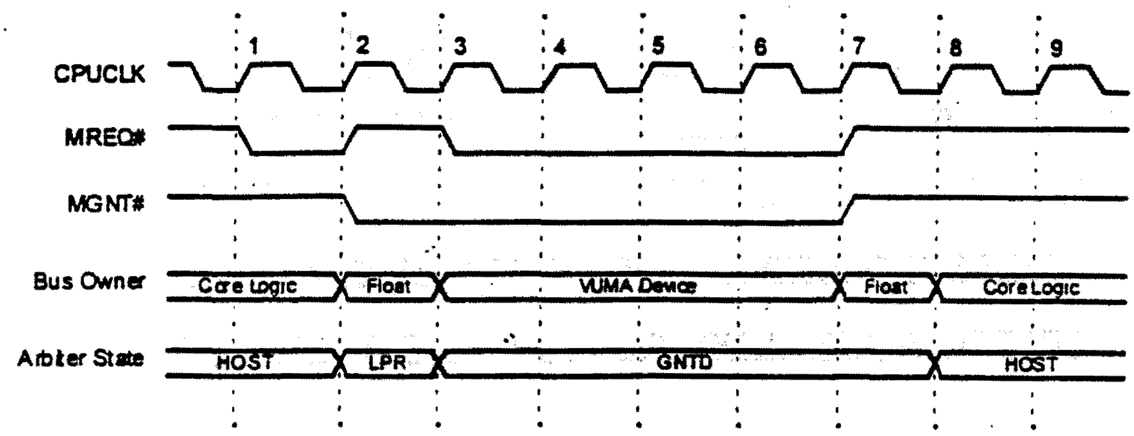
4. Low priority request and delayed bus release to VUMA device



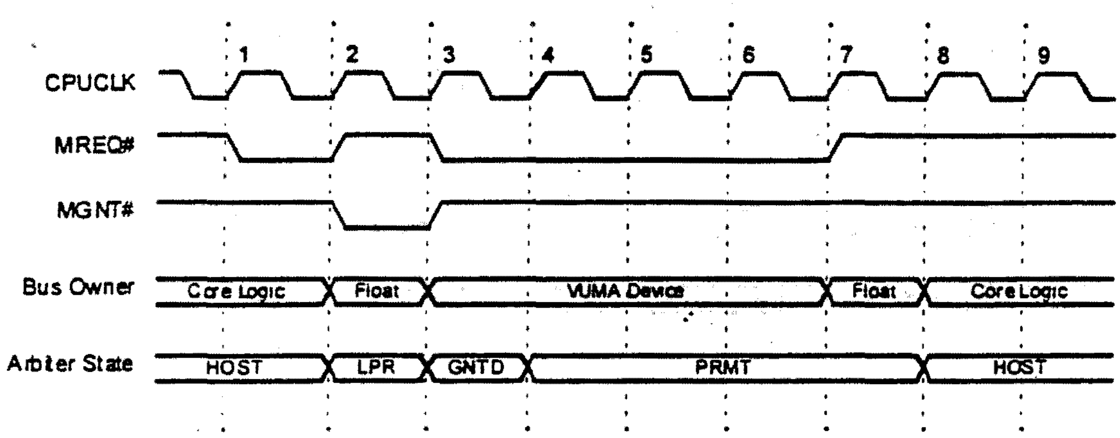
5. High priority request and immediate bus release to VUMA device



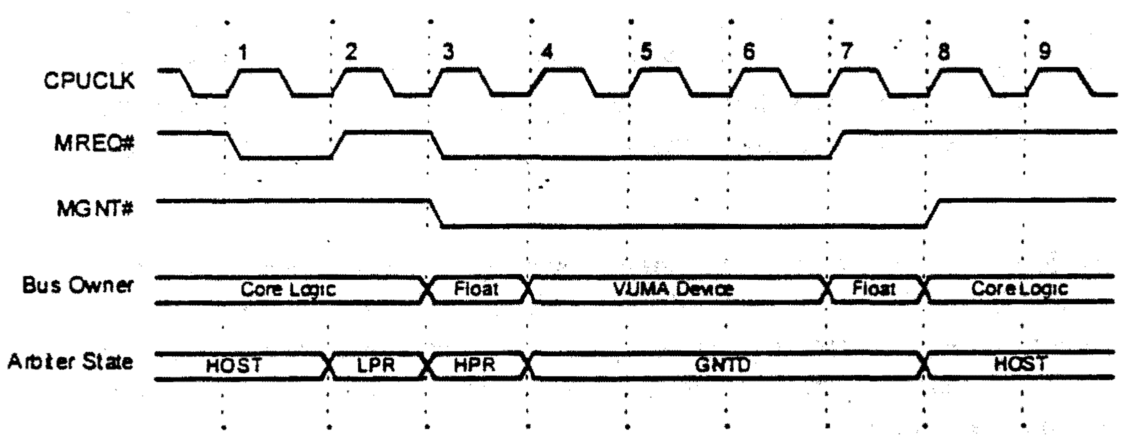
6. High priority request and immediate bus release to VUMA device with preemption where MGNT# and MREQ# removal coincides.



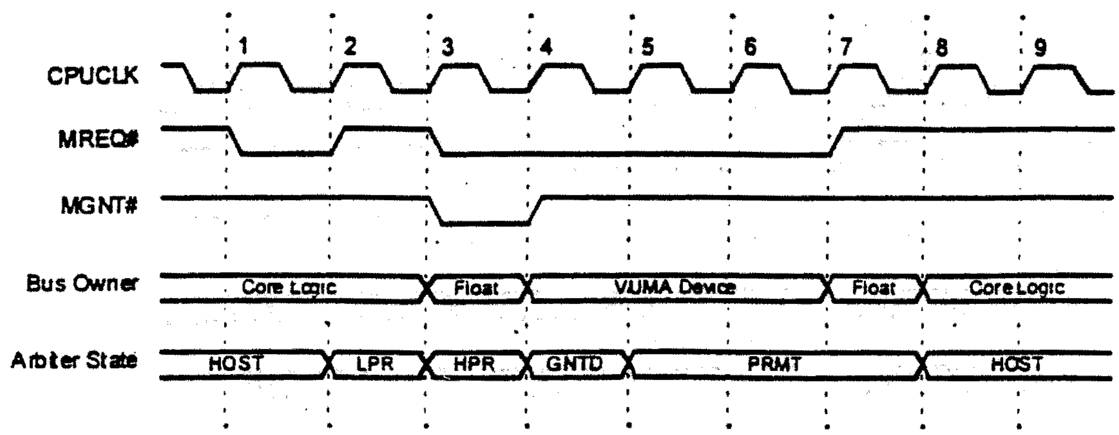
7. High priority request and immediate bus release to VUMA device with preemption where MREQ# is removed after the current transaction because of preemption.



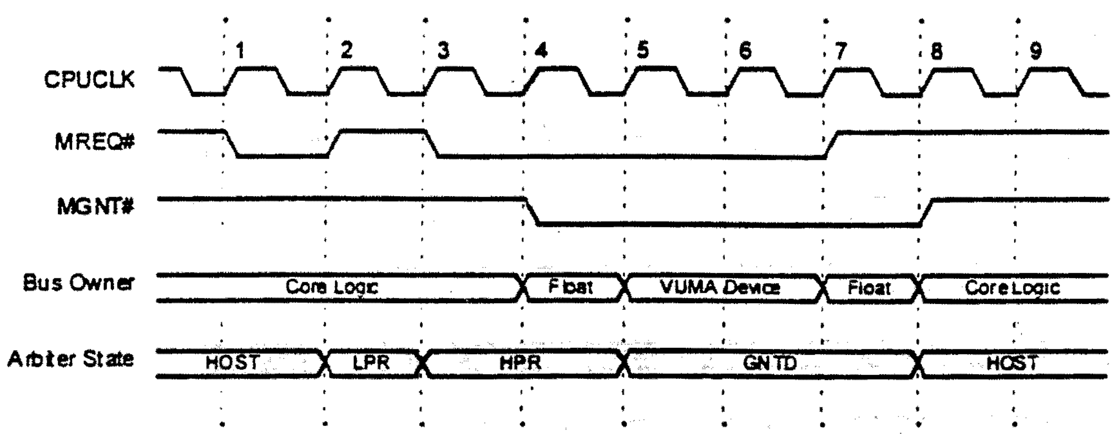
8. High priority request and one clock delayed bus release to VUMA device



9. High priority request and one clock delayed bus release to VUMA device with preemption where MREQ# and MGNT# removal do not coincide



10. High priority request and delayed bus release to VUMA device



4.4 Latencies

1. High Priority Request - Worst case latency for VUMA device to receive a grant after generating a high priority request is 35 CPUCLK clocks, i.e. after arbiter receives a high priority request from VUMA device, core logic does not need to relinquish the physical system memory bus right away and can keep the bus for up to 35 CPUCLK clocks.

2. Low Priority Request - No worst case latency number has been defined by this specification for low priority request. VUMA devices should incorporate some mechanism to avoid a low priority request being starved for an unreasonable time. The mechanism is implementation specific and not covered by the standard. One simple reference solution is as follows:

   VUMA device incorporates a programmable timer. The timer value is set at the boot time. The timer gets loaded when a low priority request is generated. When the timer times out, the low priority request is converted to a high priority request.

3. Preemption Request to VUMA device - Worst case latency for VUMA device to relinquish the physical system memory bus after receiving a preemption request is 60

CPUCLK clocks, i.e. after core logic requests VUMA device to relinquish the physical system memory bus, VUMA device does not need to relinquish the bus right away and can keep the bus for up to 60 CPUCLK clocks.

Design engineers should take in to consideration the above latencies for deciding FIFO sizes.

5.0 Memory Interface

The standard supports Fast Page Mode, EDO, BEDO and Synchronous DRAM technologies.

DRAM refresh for the physical system memory including Main VUMA Memory and Auxiliary VUMA Memory is provided by core logic during normal as well as suspend state of operation.

If VUMA device uses only a portion of its address space as Main VUMA Memory or Auxiliary VUMA Memory, it should drive unused upper MA address lines high.

5.1 Memory Decode

The way CPU address is translated in to DRAM Row and Column address decides the physical location in DRAM where a particular data will be stored. In the conventional architecture this could be implementation specific as there is a single DRAM controller. In unified memory architecture, multiple DRAM controllers (core logic resident and VUMA device resident DRAM controller) need to access the same data. Hence, all DRAM controllers should follow the same translation of CPU address into DRAM Row and Column address. The translation is as shown in Table 5-1.

Table 5-1 Translation of CPU address to DRAM Row and Column addresses

Symmetrical x9, x10, x11, x12

	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
clmn	A26	A24	A22	A11	A10	A9	A8	A7	A6	A5	A4	A3
row	A25	A23	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12

Asymmetrical x8

	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
clmn					A10	A9	A8	A7	A6	A5	A4	A3
row	A22	A21	A11	A20	A19	A18	A17	A16	A15	A14	A13	A12

Asymmetrical x9

	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
clmn				A11	A10	A9	A8	A7	A6	A5	A4	A3
row	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12

Asymmetrical x10

	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
clmn			A22	A11	A10	A9	A8	A7	A6	A5	A4	A3
row	A24	A23	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12

Asymmetrical x11

	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
clmn		A24	A22	A11	A10	A9	A8	A7	A6	A5	A4	A3
row	A25	A23	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12

Synchronous 16Mb

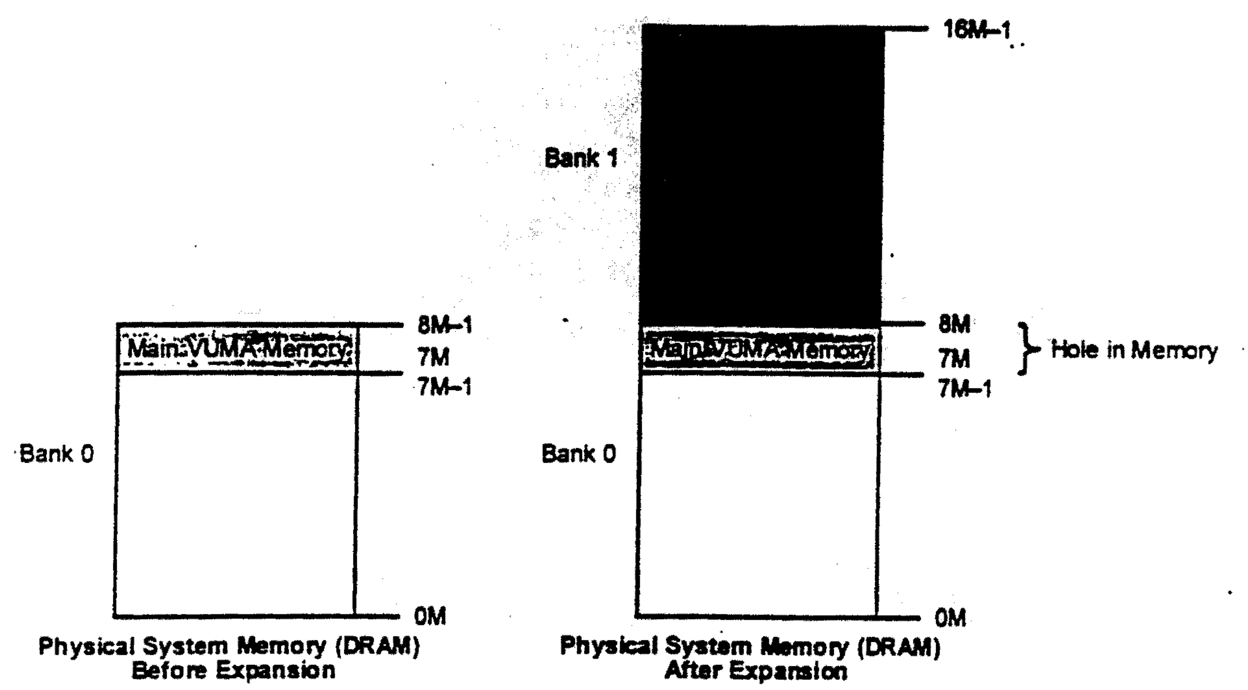
	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
clmn	A11		A24	A23	A10	A9	A8	A7	A6	A5	A4	A3
row	A11	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12

5.2 Main VUMA Memory Mapping

When physical system memory (DRAM) is expanded, unified memory architecture poses a unique problem not existing on the conventional architecture. The problem and three different solutions are described below:

Problem: Main VUMA Memory needs to be mapped at the top of existing memory for any given machine. When physical system memory (DRAM) is expanded, this would cause a hole in the physical system memory as shown in Figure 5-1. The example assumes an initial system with single bank 8MB memory (1MB allocated to Main VUMA Memory) expanded to 16MB memory (1MB allocated to Main VUMA Memory) by adding a bank of 8MB memory. All the numbers mentioned in this discussion are just examples and do not imply to be a part of the standard.

Figure 5-1 Memory Expansion Problem

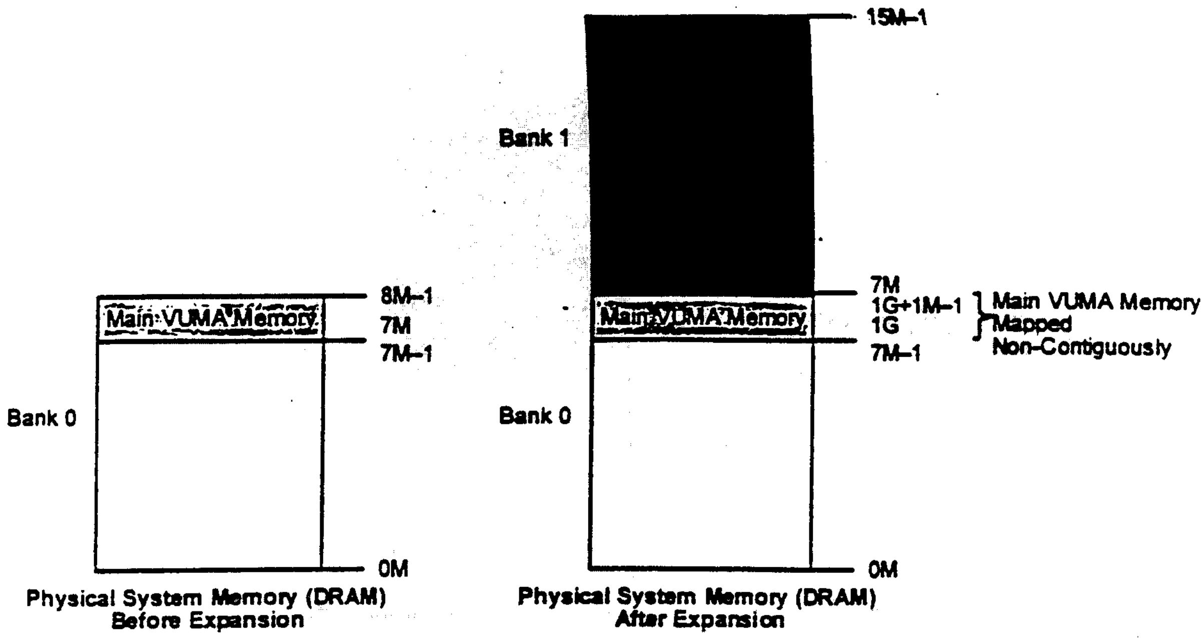


Three solutions are suggested for this problem. BIOS calls defined in VUMA VESA BIOS Extensions support all the three solutions. The BIOS calls are designed in such a way that a VUMA device can find out which of the three solutions is implemented by core logic and can configure the VUMA device appropriately.

Solution 1:

As depicted in Figure 5-2, core logic maps Main VUMA Memory to an address beyond core logic's possible physical system memory range. Main VUMA Memory is mapped non-contiguous to the O.S. memory. As shown in Figure 5-2, Main VUMA Memory is mapped from 1G to (1G+1M-1) and hence even if physical system memory is expanded to the maximum possible size, there will be no hole in the memory. As shown in Figure 5-2. Bank 0 is split with two separate blocks of memory with different starting addresses. If the VUMA device is a graphics controller, and if it wants to look at Main VUMA Memory also as a PCI address space, it can allocate a different address than what has been assigned by core logic (1G in this example).

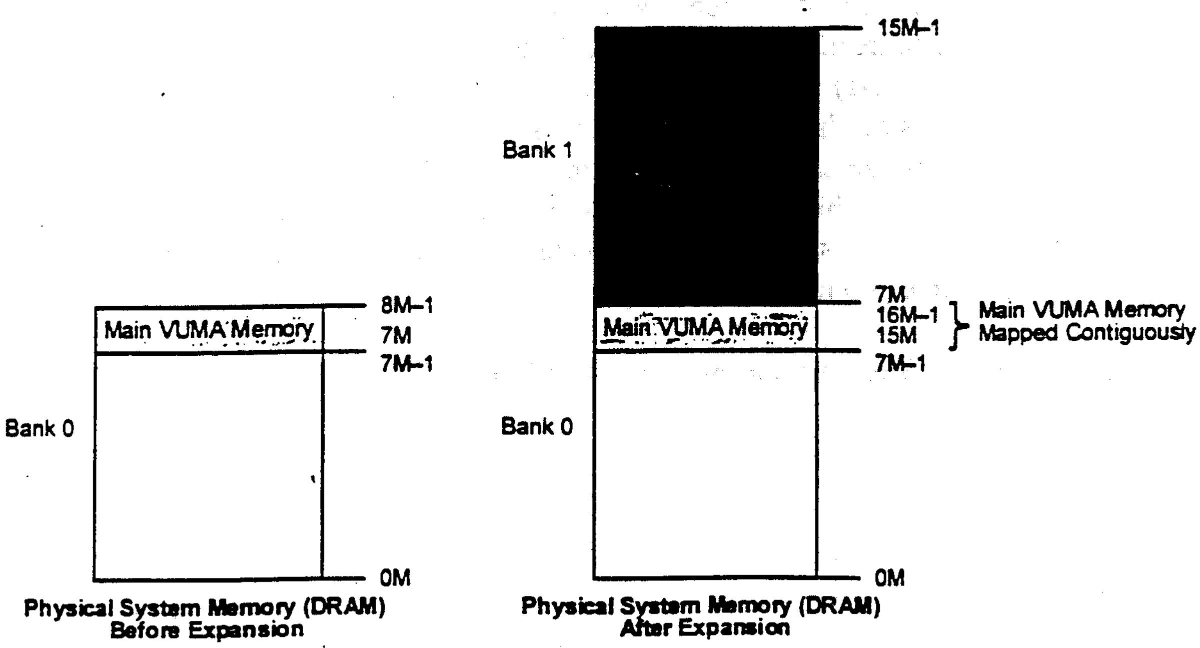
Figure 5-2 Main VUMA Memory mapped non-contiguously



Solution 2:

As depicted in Figure 5-3, core logic maps Main VUMA Memory to the top of memory. Main VUMA Memory is mapped contiguous to the O.S. memory. As shown in Figure 5-3, Main VUMA Memory is mapped from 15 M to (16M-1). As shown in Figure 5-3, Bank 0 is split with two separate blocks of memory with different starting addresses.

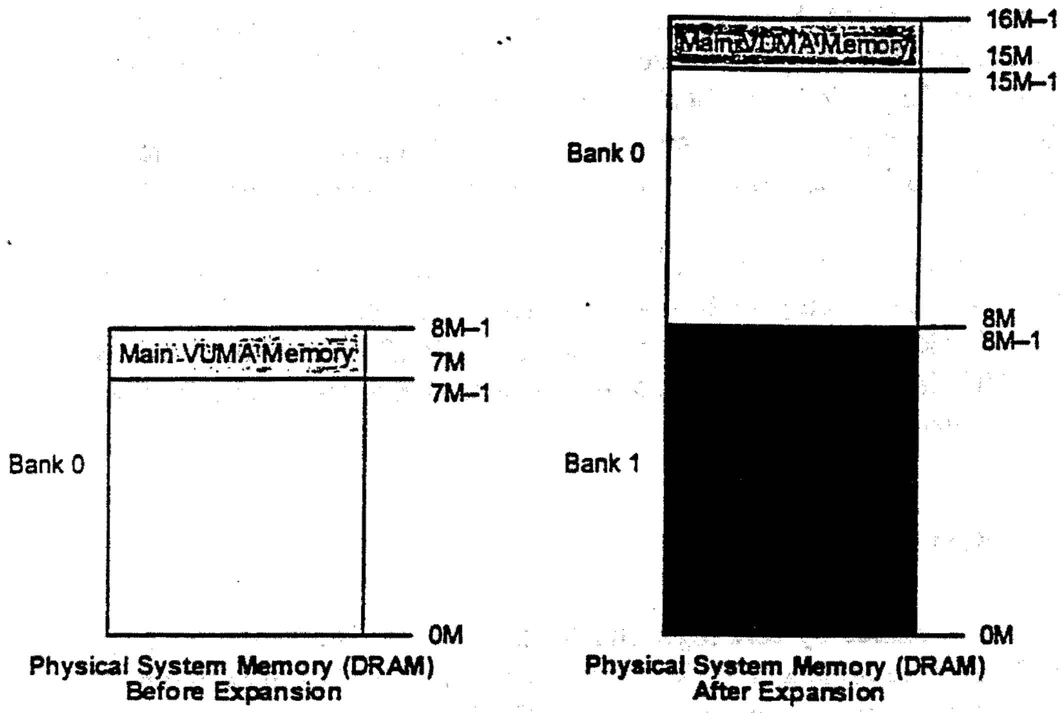
Figure 5-3 Main VUMA Memory mapped contiguously



Solution 3:

As depicted in Figure 5-4, core logic swaps the bank containing main VUMA Memory to the top of memory. As shown in Figure 5-4, Bank 0 is not split with two separate blocks of memory with different starting addresses like in solution 1 and solution 2.

Figure 5-4 Main VUMA Memory bank swapped



5.3 Fast Page EDO and BEDO

The logical interfaces for Fast Page, EDO and BEDO DRAMs are very similar and hence are grouped together. If no specific exception to a particular technology is mentioned, the description in this section applies to all the three types of DRAMs.

BEDO support is optional for both core logic and VUMA device. Various BEDO support scenarios are as follows:

1. Core logic does not support BEDO - Since core logic does not support BEDO, there will not be any BEDO as the physical system memory and hence whether VUMA device supports BEDO or not is irrelevant.

2. Core logic supports BEDO - When core logic supports BEDO, VUMA device may or may not support it. Whether core logic and VUMA device support BEDO or not should be transparent to the operating system and application programs. To achieve the transparency, system BIOS needs to find out if both core logic and VUMA device support this feature and set the system appropriately at boot. The following algorithm

explains how it can be achieved. The algorithm is only included to explain the feature. Refer to the latest VUMA VESA BIOS Extensions for the most updated BIOS calls:

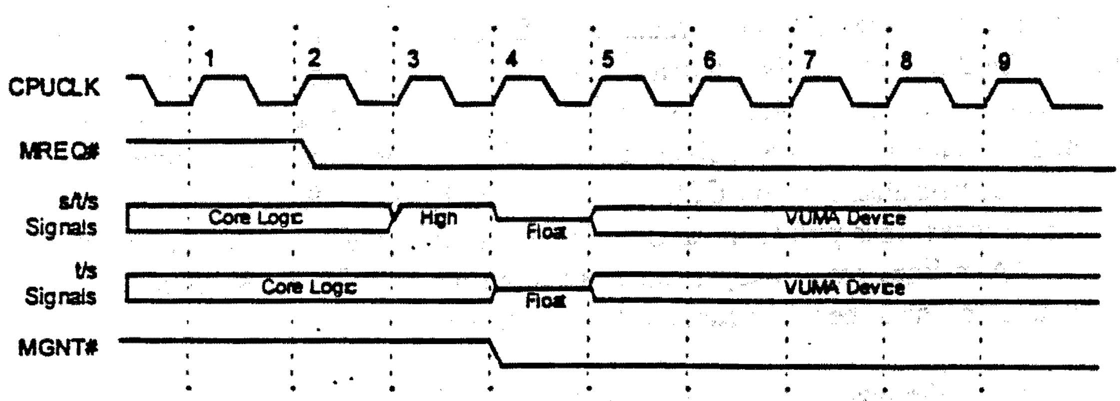
a. Read <VUMA BIOS signature string (refer to VUMA VESA BIOS Extensions)>. Check if VUMA device supports BEDO.
b. If VUMA device does not support BEDO, do not assign BEDO banks for Main VUMA Memory. Assign Main VUMA Memory to Fast Page Mode or EDO bank. Also, if Auxiliary VUMA Memory is assigned by operating system to BEDO banks, do not use it. Either repeat the request for Auxiliary VUMA Memory till it is assigned to Fast Page Mode or EDO bank or use some alternate method.
c. If VUMA device supports BEDO, read <VUMA BIOS signature string (refer to VUMA VESA BIOS Extensions)> to find out if VUMA device supports multiple banks access.
d. If only single bank access supported on VUMA device, exit, as the Main VUMA Memory and Auxiliary VUMA Memory bank is fixed.
e. If multiple banks access is supported and if the RAS for BEDO bank is supported on VUMA device, assign the Main VUMA Memory to obtain the best possible system performance and exit.

5.3.1 Protocol Description and Timing

All the DRAM signals are shared by core logic and VUMA device. They are driven by current bus master. When core logic and VUMA device hand over the bus to each other, they must drive all the shared s/t/s signals high for one CPUCLK clock and then tri-state them. Also, they should tri-state all the shared t/s signals.

The shared DRAM signals are driven by core logic when it is the owner of the physical system memory bus. VUMA device requests the physical system memory bus by asserting MREQ#. Bus Arbiter grants the bus by asserting MGNT#. Also, as mentioned above, before VUMA device starts driving the bus, core logic should drive the s/t/s signals high for one CPUCLK clock and tri-state them. Core logic should also tri-state all the shared t/s signals. The float condition on the bus should be for one CPUCLK clock, before VUMA device starts driving the bus. These activities are overlapped to improve performance as shown in Figure 5-5.

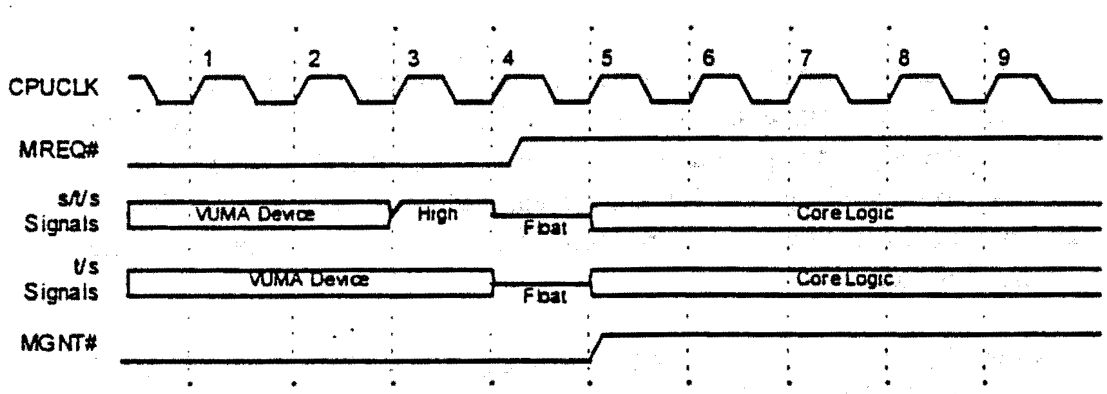
Figure 5-5 Bus hand off from core logic to VUMA device



MREQ# is driven low from clock edge 2. Core logic samples it active on clock edge 3. Arbiter can give the bus right away, so core logic drives all s/t/s signals high from the same clock edge. Core logic tri-states all the shared signals (s/t/s and t/s) and drives MGNT# active from clock edge 4 . VUMA device samples MGNT# active at clock edge 5 and starts driving the bus from the same edge.

The shared DRAM signals are driven by VUMA device when it is the owner of the physical system memory bus. VUMA device relinquishes the physical system memory bus by de-asserting MREQ#. Bus Arbiter gives the bus back to core logic by de-asserting MGNT#. Also, as mentioned above, before core logic starts driving the bus, VUMA device should drive the s/t/s signals high for one CPUCLK clock and tri-state them. VUMA device should also tri-state all the shared t/s signals. The float condition on the bus should be for one CPUCLK clock, before core logic starts driving the bus. These activities are overlapped to improve performance as shown in Figure 5-6.

Figure 5-6 Bus hand off from VUMA device to core logic



VUMA device drives all s/t/s signals high from clock edge 3. VUMA device tri-states all shared signals (s/t/s and t/s) and de-asserts MREQ# from clock edge 4. Core logic samples MREQ# inactive on clock edge 5. Core logic drives all shared signals and deasserts MGNT# from clock edge 5.

5.3.2 DRAM Precharge

When the physical system memory bus is handed off from core logic to VUMA device or vice a versa, the DRAM needs to be precharged before the new master starts driving it. Part of this precharge can be hidden by overlapping with the arbitration protocol. As shown in Figure 5-5 and 5-6, all the DRAM control signals (including RAS# lines) are driven high for and tri-stated for one CPUCLK clock each. When RAS# lines are tri-stated, pull ups on those lines pull them to a logical high. Thus when a new master gets the control, the RAS# lines are already been precharged for two CPUCLK clocks. The rest of the precharge needs to be taken care by the new master.

1. VUMA device gets the bus from core logic - As shown in Figure 5-5, when VUMA device gets the physical system memory bus at clock edge 5, the DRAM has been precharged for two CPUCLK clocks. VUMA device needs to take care of the rest of the DRAM precharge. This precharge can be overlapped by VUMA device over some of its activity e.g. VUMA device may be running at a different clock than the CPUCLK clock and the precharge can be overlapped with the synchronization of MGNT# signal. VUMA device can calculate the number of clocks it needs to precharge the DRAM with the following formula:

No. of VUMA device clocks for DRAM precharge = {RAS# Precharge Time (tRP) - (2 * CPUCLK Clock Time Period)}/ VUMA device Clock Time Period

Example: CPU running at 66.66 MHz, VUMA device running at 50 MHz. 70ns Fast Page DRAM used.

No. of VUMA device clocks for DRAM precharge = {50ns - (2* 15ns)}/ 20ns
= {20ns}/ 20ns
= 1 clock

2. Core logic gets the bus from VUMA device - As shown in Figure 5-6, when core logic gets the physical system memory bus at clock edge 5, the DRAM has been precharged for two CPUCLK clocks. core logic needs to take care of the rest of the DRAM precharge. This precharge can be overlapped by core logic over some of its activity e.g. driving of new row address. Core logic can calculate the number of clocks it needs to precharge the DRAM with the following formula:

No. of CPU clocks for DRAM precharge = {RAS# Precharge Time (tRP) - (2 * CPUCLK Clock Time Period)}/ CPUCLK Clock Time Period

Example: CPU running at 66.66 MHz, VUMA device running at 50 MHz. 70ns Fast Page DRAM used.

No. of CPU clocks for DRAM precharge = {50ns - (2* 15ns)}/ 15ns
= {20ns}/ 15ns
= 2 clock

5.4 Synchronous DRAM

Synchronous DRAM support is optional for both core logic and VUMA device. Various Synchronous DRAM support scenarios are as follows:

1. Core logic does not support Synchronous DRAM - Since core logic does not support Synchronous DRAM, there would not be any Synchronous DRAM as the physical system memory and hence whether VUMA device supports Synchronous DRAM or not is irrelevant.

2. Core logic supports Synchronous DRAM - When core logic supports Synchronous DRAM, VUMA device may or may not be supporting it. Whether core logic and VUMA device support Synchronous DRAM or not should be transparent to the operating system and application programs. To achieve the transparency, system BIOS needs to find out if both core logic and VUMA device support this feature and set the system appropriately at boot. The following algorithm explains how it can be achieved. The algorithm is only included to explain the feature. Refer to the latest VUMA VESA BIOS Extensions for the most updated BIOS calls:

   a. Read <VUMA BIOS signature string (refer to VUMA VESA BIOS Extensions)>. Check if VUMA device supports Synchronous DRAM.
   b. If VUMA device does not support Synchronous DRAM, do not assign Synchronous DRAM banks for Main VUMA Memory. Assign Main VUMA Memory to Fast Page Mode or EDO bank. Also, if Auxiliary VUMA Memory is assigned by operating system to Synchronous DRAM banks, do not use it. Either repeat the request for Auxiliary VUMA Memory till it is assigned to Fast Page Mode or EDO bank or use some alternate method.
   c. If VUMA device supports Synchronous DRAM, read < VUMA BIOS signature string (refer to VUMA VESA BIOS Extensions)> to find out if VUMA device supports multiple banks access.
   d. If only single bank access supported on VUMA device, exit, as the Main VUMA Memory and Auxiliary VUMA Memory bank is fixed.
   e. If multiple banks access is supported and if the CS# for Synchronous DRAM bank is supported on VUMA device, assign the Main VUMA Memory to obtain the best possible system performance and exit.

5.4.1 Programmable Parameters

Synchronous DRAMs have various programmable parameters. Core logic programs Synchronous DRAM parameters to obtain the best possible results. The most efficient way for VUMA device to program its DRAM controller is to make a BIOS call to find out the parameters core logic has decided and program its DRAM controller with the same parameters. Alternately, VUMA device could program its DRAM controller with one or all different parameters. If VUMA device programs its DRAM controller with any different parameters, it is VUMA device's responsibility to reprogram Synchronous DRAM back with the original parameters, before the physical system memory bus is handed off to core logic. In other words, VUMA device is free to change any or all of the parameters, but the change should be transparent to core logic.

How core logic programs various parameters and how VUMA device could inquire them is as follows:

1. **Burst Length -** Burst Length can be programmed as 1, 2 or 4. VUMA device needs to make a BIOS call <Return Memory Speed Type (refer to VUMA VESA BIOS Extensions)> to find out the Burst Length.
2. **CAS Latency -** As CAS latency depends on the speed of Synchronous DRAM used and the clock speed, this standard does not want to fix this parameter. Core logic programs this parameter to an appropriate value. VUMA device needs to make a BIOS call <Return Memory Speed Type (refer to VUMA VESA BIOS Extensions)> to find out the CAS latency.
3. **Burst Ordering -** Most efficient Burst Ordering depends upon the type of CPU used. VUMA device needs to make a BIOS call <Return Memory Speed Type (refer to VUMA VESA BIOS Extensions)> to find out the Burst Order.

5.4.2 Protocol Description and Timing

All the DRAM signals are shared by core logic and VUMA device. They are driven by current bus master. When core logic and VUMA device hand over the bus to each other, they must drive all the shared s/t/s signals high for one CPUCLK clock and then tri-state them. Also, they should tri-state all the shared t/s signals.

Synchronous DRAMs are precharged by precharge command. When the physical system memory bus is handed off from core logic to VUMA device or vice a versa, the DRAM precharge has two options:

1. Precharge both the internal banks before hand-off - This is a simple case where both the internal banks of the active synchronous DRAM bank are precharged and then the bus is handed off.
2. Requesting Master snoops the physical system memory bus and synchronous DRAM internal banks need not be precharged - In this case the requesting master snoops the DRAM address and control signals to track the open pages in the internal banks of the active synchronous DRAM bank. The internal banks of the active synchronous DRAM

are not precharged when the physical system memory bus is handed-off to the requesting master. If needed, the requesting master takes care of precharge after getting the physical system memory bus.

Both core logic and VUMA device have an option of either implementing or not implementing DRAM snoop feature. Whether core logic and VUMA device support DRAM snoop or not should be transparent to the operating system and application programs. To achieve the transparency, system BIOS and VUMA BIOS need to find out if both core logic and VUMA device support this feature and set the system appropriately at boot. The following algorithm explains how it can be achieved. The algorithm is only included to explain the feature. Refer to the latest VUMA VESA BIOS Extensions for the most updated BIOS calls:

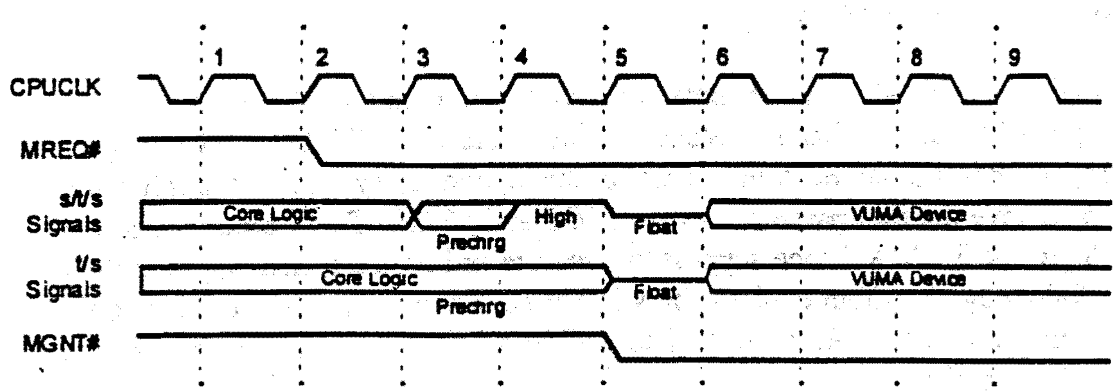
1. System BIOS reads <VUMA BIOS signature string (refer to VUMA VESA BIOS Extensions)>, to find out if VUMA device can snoop the physical system memory bus.
2. If no, System BIOS programs core logic to precharge synchronous DRAM before bus hand-off.
3. If yes, System BIOS programs core logic not to precharge synchronous DRAM before bus hand-off.
4. VUMA BIOS makes a call, <Report VUMA - core logic capabilities (refer to VUMA VESA BIOS Extensions)>, to find out if core logic can snoop the physical system memory bus.
5. If no, VUMA BIOS programs VUMA device to precharge synchronous DRAM before bus hand-off.
6. If yes, VUMA BIOS programs VUMA device not to precharge synchronous DRAM before bus hand-off.

None, only one, or both of core logic and VUMA device can support this feature. When only one of them supports this feature memory precharge will be asymmetrical i.e. there will be precharge before hand-off one way and no precharge the other way.

5.4.2.1 Non-Snoop Cases

The shared DRAM signals are driven by core logic when it is the owner of the physical system memory bus. VUMA device requests the physical system memory bus by asserting MREQ#. Bus Arbiter grants the bus by asserting MGNT#. Also, before VUMA device starts driving the bus, core logic should drive all the shared s/t/s signals high for one CPUCLK clock and tri-state them. Core logic should also tri-state all the shared t/s signals. The tri-state condition on the bus should be for one CPUCLK clock, before VUMA device starts driving the bus. These activities are overlapped to improve performance as shown in Figure 5-7. Since VUMA device does not support DRAM snoop feature, DRAM is precharged before handing off the physical system memory bus as shown in Figure 5-7.

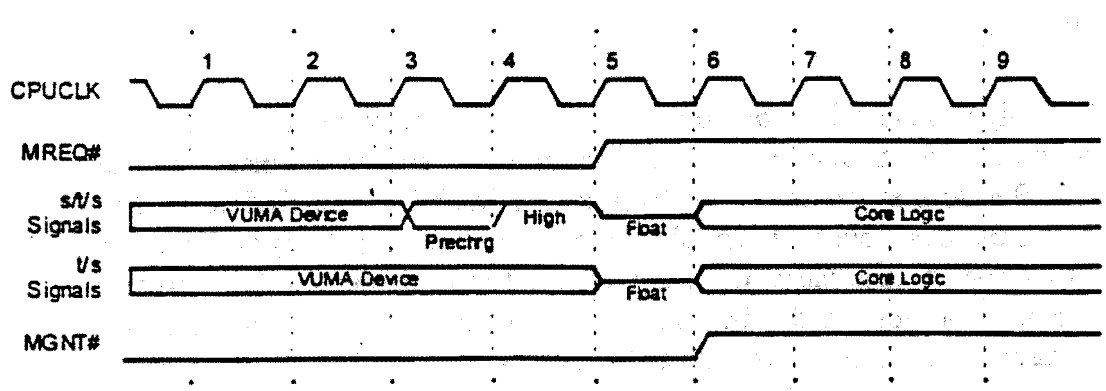
Figure 5-7 Bus hand off from Core Logic to VUMA device



MREQ# is driven low from clock edge 2. Core logic samples it active on clock edge 3. Arbiter can give the bus right away, so core logic gives precharge command to DRAM from the same clock edge. Core logic drives all the shared s/t/s signals high from clock edge 4. Core logic tri-states all the shared signals (s/t/s and t/s) and drives MGNT# active from clock edge 5. VUMA device samples MGNT# active at clock edge 6 and starts driving the bus from the same edge.

The shared DRAM signals are driven by VUMA device when it is the owner of the physical system memory bus. VUMA device relinquishes the physical system memory bus by de-asserting MREQ#. Bus Arbiter gives the bus back to core logic by de-asserting MGNT#. Also, as mentioned above, before core logic starts driving the bus, VUMA device should drive all the shared s/t/s signals high for one CPUCLK clock and tri-state them. VUMA device should also tri-state all the shared t/s signals. The float condition on the bus should be for one CPUCLK clock, before core logic starts driving the bus. These activities are overlapped to improve performance as shown in Figure 5-8. Since core logic does not support DRAM snoop feature, DRAM is precharged before handing off the physical system memory bus as shown in Figure 5-8.

Figure 5-8 Bus hand off from VUMA device to Core Logic



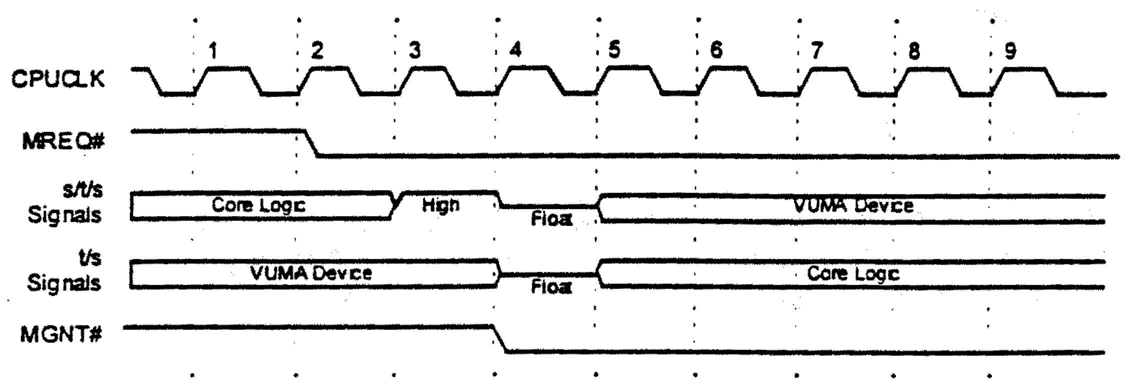
VUMA device gives precharge command from clock edge 3. It drives all shared s/t/s signals high from clock edge 4. It tri-states all shared signals (s/t/s and t/s) and de-asserts

MREQ# from clock edge 5. Core logic samples MREQ# inactive on clock edge 6. Core logic drives all shared signals and deasserts MGNT# from clock edge 6.

5.4.2.2 Snoop Cases

The shared DRAM signals are driven by core logic when it is the owner of the physical system memory bus. VUMA device requests the physical system memory bus by asserting MREQ#. Bus Arbiter grants the bus by asserting MGNT#. Also, before VUMA device starts driving the bus, core logic should drive all the shared s/t/s signals high for one CPUCLK clock and tri-state them. Core logic should also tri-state all the shared t/s signals. The tri-state condition on the bus should be for one CPUCLK clock, before VUMA device starts driving the bus. These activities are overlapped to improve performance as shown in Figure 5-9. Since VUMA device supports DRAM snoop feature, core logic does not precharge DRAM before handing off the physical system memory bus as shown in Figure 5-9.

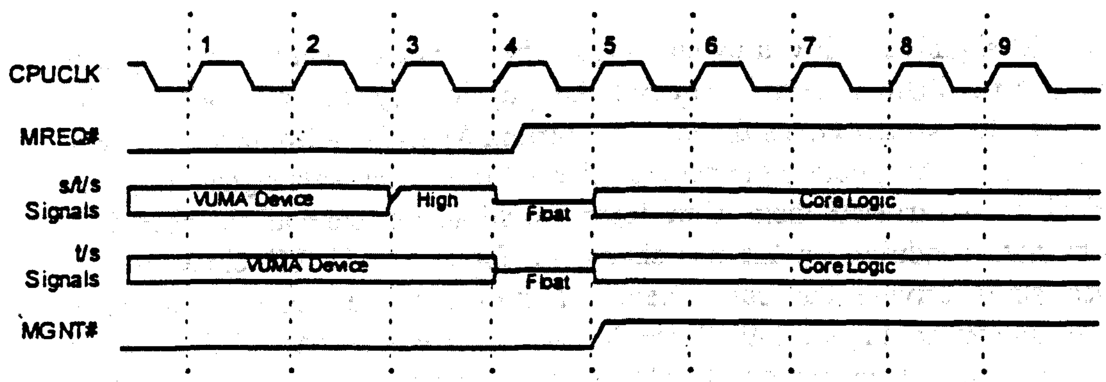
Figure 5-9 Bus hand off from core logic to VUMA device



MREQ# is driven low from clock edge 2. Core logic samples it active on clock edge 3. Arbiter can give the bus right away and since VUMA device supports DRAM snoop feature, core logic drives all the shared s/t/s signals high from the same clock edge. Core logic tri-states all the shared signals (s/t/s and t/s) and drives MGNT# active from clock edge 4. VUMA device samples MGNT# active at clock edge 5 and starts driving the bus from the same edge.

The shared DRAM signals are driven by VUMA device when it is the owner of the physical system memory bus. VUMA device relinquishes the physical system memory bus by de-asserting MREQ#. Bus Arbiter gives the bus back to core logic by de-asserting MGNT#. Also, as mentioned above, before core logic starts driving the bus, VUMA device should drive all the shared s/t/s signals high for one CPUCLK clock and tri-state them. VUMA device should also tri-state all the shared t/s signals. The float condition on the bus should be for one CPUCLK clock, before core logic starts driving the bus. These activities are overlapped to improve performance as shown in Figure 5-10. Since core logic supports DRAM snoop feature, VUMA device does not precharge DRAM before handing off the physical system memory bus as shown in Figure 5-10.

Figure 5-10 Bus hand off from VUMA device to core logic



VUMA device drives all shared s/t/s signals high from clock edge 3. It tri-states all shared signals (s/t/s and t/s) and de-asserts MREQ# from clock edge 4. Core logic samples MREQ# inactive on clock edge 5. Core logic drives all shared signals and deasserts MGNT# from clock edge 5.

5.5 Memory Parity support

Memory Parity support is optional on both core logic and VUMA device. If core logic supports parity it should be able to disable parity check for Main VUMA Memory and Auxiliary VUMA Memory areas while parity check on the rest of the physical system memory is enabled.

5.6 Memory Controller Pin Multiplexing

The logical interfaces for Fast Page, EDO and BEDO DRAMs are very similar but are significantly different than that of Synchronous DRAM. If mother board designers want to mix different DRAM technologies on the same mother board, core logic will have to multiplex DRAM control signals. The meaning of a multiplexed signal will depend on the type of DRAM core logic is accessing at a given time. If a VUMA device supports multiple banks access and mix of different DRAM technologies, it will also have to multiplex DRAM control signals. Both core logic and VUMA devices will have to have same multiplexing scheme. The appropriate JEDEC standard should be followed for multiplexing scheme.

6.0 Boot Protocol

6.1 Main VUMA Memory Access at Boot

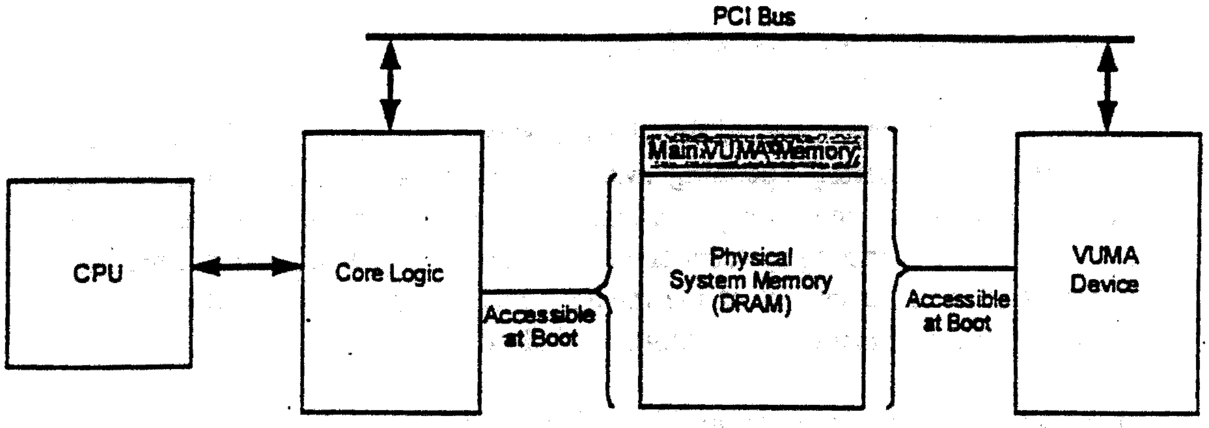
In unified memory architecture, part of the physical system memory is assigned to Main VUMA Memory. The existing operating systems are not aware of unified memory architecture. Also, some of the existing operating systems size memory themselves. This poses a problem as the operating systems after sizing the total physical system memory, will assume that they could use all of the memory and might overwrite Main VUMA Memory. The solution to this problem is explained below:

As shown in Figure 6-1, the solution to this problem is to disable core logic access to Main VUMA Memory area at boot time. In that case even if operating system, sizes the memory, it will find only (total physical system memory - Main VUMA Memory) and will not be aware of the Main VUMA Memory existence. This will avoid operating system ever writing to the Main VUMA Memory area. If VUMA device supports multiple banks access, it can access total physical system memory all the time. If VUMA device supports single bank access, it can access the bank of Main VUMA Memory all the time.

If VUMA device is a graphics controller, it needs a special consideration. Video screen is required during boot and since core logic can not access the Main VUMA Memory, it can not write to it. The problem is solved by programming the graphics controller into a pseudo legacy mode. In this mode graphics controller treats Main VUMA Memory exactly the same way as in non unified memory architecture situations i.e. as if it has its own separate frame buffer. So now, the total system looks just like a non unified memory architecture system and this mode is called as pseudo legacy mode. Core logic performs accesses to video through legacy video memory address space of A000:0 and B000:0. These accesses go on the PCI bus. Graphics controller claims these cycles. Graphics controller still needs to arbitrate for the physical system memory bus. After getting the bus, graphics controller performs reads/writes to Main VUMA Memory (frame buffer). After the system boots, it is still in the pseudo legacy mode. When operating system calls display driver, the driver programs core logic to allow access to Main VUMA Memory and switches the system from pseudo legacy mode to unified memory architecture.

In the case of other type of VUMA devices, device driver needs to program core logic to allow access to Main VUMA Memory.

Figure 6-1 Pseudo Legacy Mode



The following algorithm sums up the boot process in the case of VUMA device being a graphics controller:

1. System BIOS sizes the physical system memory.
2. System BIOS reads the size of Main VUMA Memory at previous boot (where this value is stored is System BIOS dependent, but needs to be in some sort of non volatile memory).
3. System BIOS programs its internal registers to reflect that total memory available is [total physical system memory(from step 1) - Main VUMA Memory at previous boot (from step 2)].
4. System boots and operating system calls display driver.
5. Display driver makes a System BIOS call, <Enable/Disable Main VUMA Memory (refer to VUMA VESA BIOS Extensions)>, to program core logic internal registers to reflect that it can access total physical system memory.
6. Display driver switches VUMA device to unified memory architecture mode.

Even though core logic can not access Main VUMA Memory till the time display driver enables it, core logic is responsible for Main VUMA Memory refresh.

VUMA device should claim PCI Master accesses to Main VUMA Memory till display driver enables core logic access to that area. Core logic should claim PCI Master accesses to Main VUMA Memory after display driver enables core logic access to that area.

6.2 Reset State

On power on reset, both core logic and VUMA device have their unified memory architecture capabilities disabled. MREQ# is de-asserted by VUMA device and MGNT# is de-asserted by core logic. System BIOS can detect if VUMA device supports unified memory architecture capabilities by reading <VUMA BIOS signature string (refer to VUMA VESA BIOS Extensions)>.

7.0 Electrical Speciation

7.1 Signal Levels

This section describes the electrical signal levels for the arbitration signals only. DRAM signal levels depend on the type of DRAM used and hence can not be specified by the standard.

MREQ#	output	5v TTL or 3.3v LVTTL
	input	5v TTL for 5v buffer, 5v tolerant LVTTL for 3.3v buffer
MGNT#	output	5v TTL or 3.3v LVTTL
	input	5v TTL for 5v buffer, 5v tolerant LVTTL for 3.3v buffer
CPUCLK	output	5v TTL or 3.3v LVTTL
	input	5v TTL for 5v buffer, 5v tolerant LVTTL for 3.3v buffer

7.2 AC Timing

This section describes the AC timing parameters for the arbitration signals only. DRAM AC timing parameters depend on the type of DRAM used and hence can not be specified by the standard. Both MREQ# are MGNT# timing parameters are with respect to CPUCLK rising edge.

MREQ#	output	tClk to Out (max)	- 10 ns
		tClk to Out (min)	- 2 ns
	input	Set up time tSU (min)	- 3 ns
		Hold time tH (min)	- 0 ns
MGNT#	output	tClk to Out (max)	- 10 ns
		tClk to Out (min)	- 2 ns
	input	Set up time tSU (min)	- 3 ns
		Hold time tH (min)	- 0 ns
CPUCLK	output	clock frequency (max)	- 66.66 MHz

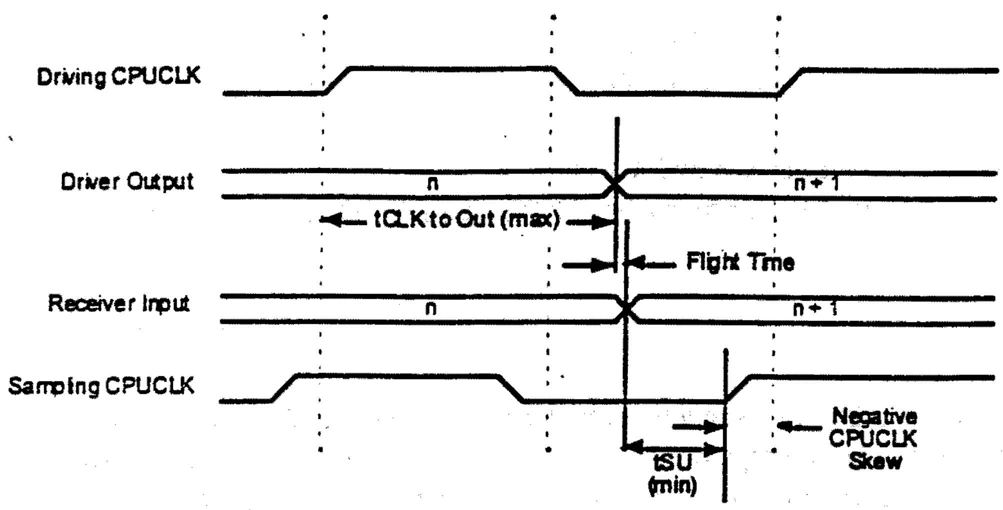
7.2.1 Timing Budget

A margin for signal flight time and clock skew is added to the timing parameters. ± 2ns is allowed for the total of CPUCLK skew and signal flight time. Worst case timing budget calculations for setup and hold time are as follows:

7.2.1.1 Worst case for Setup time

Figure 7-1 shows the worst case for setup time. tClk to Out, flight time and clock skew have converged to reduce available setup time.

Figure 7-1 Worst case for setup time

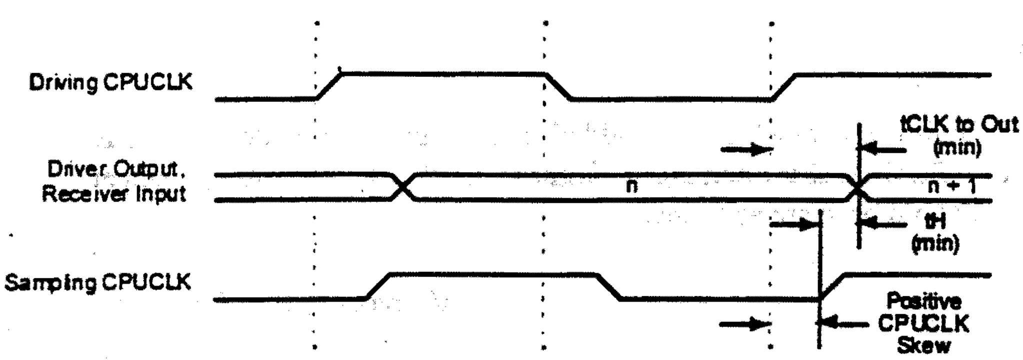


[tCLK to Out (max) + flight time + tSU (min) + negative CPUCLK skew] ≤ CPUCLK period i.e. 15ns @ 66.66 MHz.
[10ns + flight time + 3ns + negative CPUCLK skew] ≤ 15ns
[flight time + negative CPUCLK skew] ≤ 2ns

7.2.1.2 Worst case for Hold time

Figure 7-2 shows the worst case for hold time. tClk to Out and clock skew have converged to reduce available hold time. Positive flight time number helps in this case and hence it is assumed to be zero.

Figure 7-2 Worst case for hold time



[positive CPUCLK skew + tH (min)] ≤ tCLK to Out (min)
[positive CPUCLK skew + 0ns] ≤ 2ns
positive CPUCLK skew ≤ 2ns

7.3 Pullups

All s/t/s signals need pullups to sustain the inactive state until another agent drives them. Core logic has to provide pullups for all the s/t/s signals. VUMA device has as option of providing pullups on some of the s/t/s signals. All t/s signals need pulldowns. Core logic has to provide pulldowns for all the t/s signals. VUMA device has as option of providing pulldowns on the t/s signals. Pullups and pulldowns could either be internal to the chips or external on board.

DRAM Address -	Core logic is responsible for pullups on DRAM Address lines.
DRAM control signals -	Core logic is responsible for pullups on DRAM control signals. VUMA device has as option of providing pullups on them.
DRAM Data Bus -	Core logic is responsible for pulldowns on DRAM data bus. VUMA device has as option of providing pulldowns on them.

Pullups and pulldowns are used to sustain the inactive state until another agent drives the signals and hence need to be weak. Recommended value for pullups and pulldowns is between 50 kohm and 80 kohm.

7.4 Straps

As some VUMA devices and core logic chips use DRAM data bus for straps, DRAM data bus needs to be assigned for straps for different controllers. The assignment of DRAM Data Bus for straps is as follows:

MD [0:19]	VUMA device on Motherboard
MD [20:55]	Reserved
MD [56:63]	Core Logic

All the straps need to be pullups of 10 kohm.

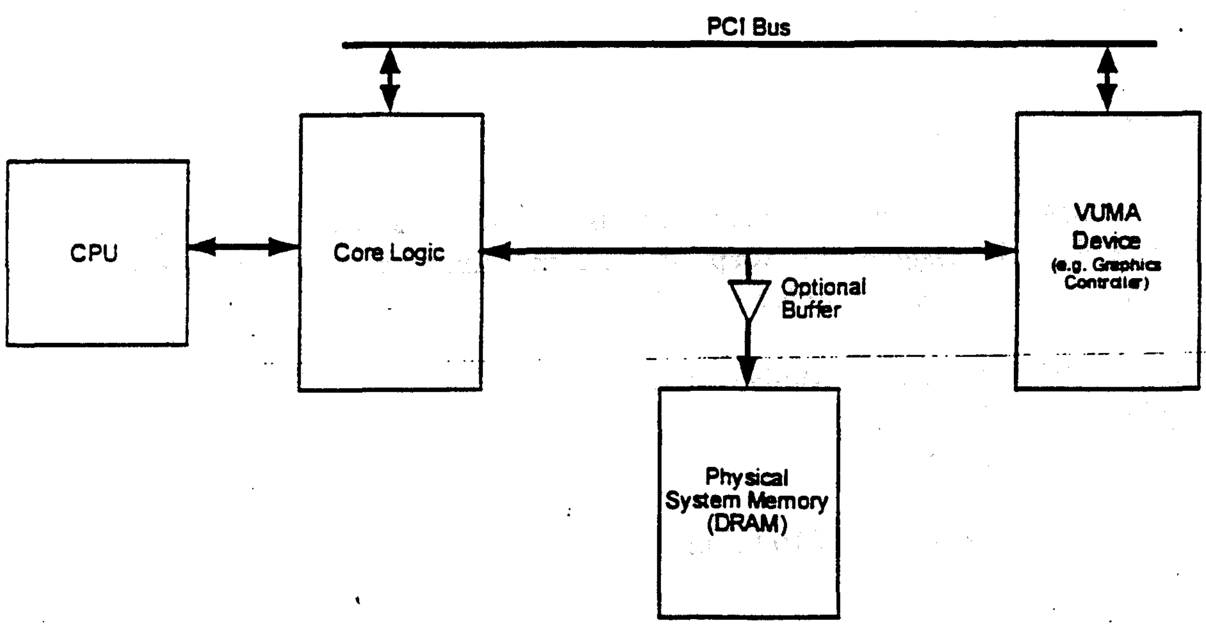
7.5 DRAM Driver characteristics

Loading plays a critical role in DRAM access timing. In case of PC motherboards end users can expand the existing memory of a system by adding extra SIMMs. Hence, typically the total DRAM signal loading is not constant and could vary significantly. Both Core Logic and VUMA device must be able to drive the maximum load that the system motherboard is designed to accommodate. In typical motherboard designs DRAM signal loading can be excessive (on the order of 1000pF for some signals) and hence care must be taken for DRAM driver selection. Some general guide lines for DRAM driver design are as follows:

Slew-rate controlled drivers are recommended. Drivers with selectable current drive (such as 8/16 mA drivers) may be used. This can reduce overshoot and undershoot associated with over-driving lightly loaded signals and can prevent excessive rise and fall time delay due to not providing enough current drive on heavily loaded signals.

As shown in Figure 7-3, buffers may be placed on the system motherboard to reduce the per signal loading and/or provide larger drive strength capabilities. DRAM Write Enable and DRAM Address signals are typically the most heavily loaded signals. Column Address Strobe signals may also become overloaded when more than two DRAM banks are designed into a system. TTL or CMOS buffers (typically 244 type) may be used to isolate and duplicate heavily loaded signals on a per bank basis. 244 type buffers typically have very good drive characteristics as well and can be used to drive all of the heavily loaded DRAM control signals if the Core Logic and/or VUMA device has relatively weak drive characteristics. If external buffers are used, the buffer delays should be taken in to timing considerations.

Figure 7-3 Optional Buffers for DRAM Signals



Wider DRAM devices offer reduced system loading on some of the control signals. x4 DRAMs require four times the physical connections on RAS, MA (Address), and write enables as x16 DRAMs. The reduction in loading can be significant. If the designer has control over the DRAMs which will be used in the system, the DRAM width should be chosen to provide the least loading.

(AL)

VESA®

VUMA Proposal

(Draft)

Video Electronics Standards Association

2150 North First Street, Suite 440
San Jose, CA 95131-2029

Phone: (408) 435-0333
FAX: (408) 435-8225

VESA Unified Memory Architecture
VESA BIOS Extensions (VUMA-SBE)
Proposal

Version: 1.0
Document Revision: 2.2p
November 1, 1995

Important Notice: This is a **draft** document from the Video Electronics Standards Association (VESA) Unified Memory Architecture Committee (VUMA). It is only for discussion purposes within the committee and with any other persons or organizations that the committee has determined should be invited to review or otherwise contribute to it. It has not been presented or ratified by the VESA general membership.

Purpose

To allow the video BIOS and other GUI specific software to control the VUMA hardware without specific knowledge or direct hardware access.

Summary

This document contains a specification for a system and video BIOS interface, VUMA-SVBE. The VUMA-SVBE interface will allow the video BIOS and other GUI specific software to control the VUMA hardware without specific knowledge or direct hardware access. The hardware protocol is described in VESA document VUMA 1.0.

Scope

Because this is a draft document, it cannot be considered complete or accurate in all respects although every effort has been made to minimize errors.

Intellectual Property

Trademarks

All trademarks used in this document are the property of their respective owners. VESA and VUMA are trademarks owned by the Video Electronics Standards Association.

Patents

The proposals and standards developed and adopted by VESA are intended to promote uniformity and economies of scale in the video electronics industry. VESA strives for standards that will benefit both the industry and end users of video electronics products. VESA cannot ensure that the adoption of a standard; the use of a method described as a standard; or the making, using, or selling of a product in compliance with the standard does not infringe upon the intellectual property rights (including patents, trademarks, and copyrights) of others. VESA, therefore, makes no warranties, expressed or implied, that products conforming to a VESA standard do not infringe on the intellectual property rights of others, and accepts no liability direct, indirect or consequential, for any such infringement.

Support For This Specification

If you have a product that incorporates VUMA™, you should ask the company that manufactured your product for assistance. If you are a manufacturer of the product, VESA can assist you with any clarification that you may require. All questions must be sent in writing to VESA via:

(The following list is the preferred order for contacting VESA.)

VESA World Wide Web Page: *www.vesa.org*

Fax: (408) 435-8225

Mail: VESA
2150 North First Street
Suite 440
San Jose, California 95131-2029

Acknowledgments

This document would not have been possible without the efforts of the members of the 1995 VESA Unified Memory Architecture Committee and the professional support of the VESA staff.

Work Group Members

Any industry standard requires information from many sources. The following list recognizes members of the VUMA Committee, which was responsible for combining all of the industry input into this proposal.

VUMA Chairperson

Rajesh Shakkarwar OPTi

Software work group Members

Tim Crawford	Work group leader, Cirrus Logic
Phil Mummah	Phoenix Technologies
Josh Grossman	S3, Inc.
Christopher Rhodes	Award Software

Revision History

Initial Revision — Aug 14 '95

Rev .7 — Aug 24 '95

Rev .8 — Aug 28 '95

Removed segment registers, modified Boot sequence wording.

Rev .9 — Aug 28 '95

Added assumption F, modified issues and moved some to assumptions, remove references to main and aux VUMA memory, removed function for aux VUMA memory, modified speed/type function

Rev 1.0 — Sept 8 '95

Added VUMA device DRAM support, SDRAM parameters in function 6

Rev 1.1 — Sept 11 '95

Add items to Boot sequence
Remove a function call and cleaned up others

Rev 1.2 — Sept 22 '95

Add changes suggested in previous VESA meeting

Rev 1.3 — Sept 29 '95

Added 32 bit I/F, added Aux functions, Added 16 bit protected mode I/F, Modified several functions

Rev 1.4 — Oct 6 '95

Modified assumptions, issues, updated some functions.

Rev 1.5 — Oct 6 '95

Modified the table of contents

Rev 1.6 — Oct 10 '95

Minor modifications to assumptions

Rev 1.7 — Oct 16 '95

Changes to assumptions, & goals.
Made chages to some functions per discusion at last committee meeting.

Rev 1.8 — Oct 18 '95

Made changes to how memory is reported, funcs 1 & 6. Modified ROM signature.

Rev 1.9 — Oct 18 '95

Modified function 6 and assumptions. Other minor typos fixed.

Rev 2.0 — Oct 23 '95

Modified assumptions C, J, L, N. Added assumptions O, P, Q, R. Removed issue 4. Added error code 5, 6. Changed unit of memory in ROM signature from 1K to 64K. Modified register definition in function 0, 1, 2, 6, and 7. Other minor typos fixed.

Rev 2.1 — Oct 24 '95

Added assumption "s".

Rev 2.2 — Nov 1 '95

Added requirement for 32 and 16 bit protected mode call stack selector and IO requirements. Removed issue section. Updated 32-bit and 16 bit interface section. Modified assumption l, o, s. Rewrote sec. 3.1, 3.2.. Modified sec.3.4, 3.5.0,3.5.1, 3.5.6, 3.5.8.

TABLE OF CONTENTS

1.0 Introduction

This document contains specifications for VESA unified memory architecture system and video BIOS interface. The system BIOS VUMA-SBE (System BIOS Extensions) will allow the video BIOS and other GUI specific software to control the VUMA hardware without specific knowledge or direct hardware access. The video BIOS VUMA-VBE (Video BIOS Extensions) will allow the system BIOS and other GUI specific software to access the VUMA hardware without specific knowledge or direct hardware access. The hardware protocol is described in VESA document VUMA 1.0.

Readers of this document should already be familiar with the VESA BIOS extensions and programming at the BIOS level.

2.0 Goals and Assumptions

VUMA-SBE provides a hardware independent means for operating system and configuration utility software to control and get status from the VUMA hardware.

VUMA-SBE services need to be provided as part of the system and video BIOS ROMs since the functions need to be used during system boot up.

2.1 Goals

a. Allow system memory access to non system controller devices. These devices, called VUMA devices will have their own memory controller and access system memory directly. All of system memory is potentially accessible by VUMA devices.

b. Allow multiple devices. Although only one connector is allowed, multiple devices on the motherboard as well as multiple devices on the expansion board are allowed.

c. If a VUMA device that previously has requested memory is taken out of the system, the memory will be returned to the O/S on the next boot.

d. If a VUMA device is replaced by another VUMA device, the system will allocate the same amount of memory, if it meets the minimum requirements of the new board. Otherwise the memory allocated will be increased to the minimum required by the new board.

2.2 Assumptions

a. System BIOS will manage memory allocation requests from VUMA devices.
b. Memory must be sized, typed and contiguous before control is turned over to a VUMA device.
c. VUMA devices will test and initialize their own Main VUMA memory. (This is similar to the way they initialize and test their video memory on conventional VGA devices.)
e. The lowest PCI PFA number will have priority if more than one VGA device is plugged in. The manufacture can decide if the VUMA slot has the highest, lowest or middle PFA number.
f. The Video BIOS must be in shadow ram and writeable when control is passed to the video ROM as defined by the PCI SIG.
g. The values that BIOS reports in function 6 for current, voltage, and speed will be determined at build/compile time.
h. A device driver should insure that when requesting VUMA memory for the next boot that enough memory will be left for the O/S to boot.
i. The device driver should take into consideration memory bandwidth when requesting memory.
j. Memory is installed on the mother board in the bank or banks (RAS/CS) that the VUMA device can access. If a user moves memory to a bank (RAS/CS) that the VUMA controller can not access, the VUMA device will be disabled.
k. On a warm boot the sytem will reallocate VUMA memory for each device.
l. For a multi-function plug in board, only function 0 on the board may require a minimum amount of memory for booting. See section 3.4, point A. Set next boot size call (VUMA-SBE function 2) can only be made using the PFA of function 0 on the board.
m. If a plug in card has a bridge, only the first function of the first device behind the bridge may require a minimum amount of memory for booting. See section 3.4, point A.
n. System BIOS will insure that PCI addresses will not conflict with Main VUMA memory that is placed above system memory. Main VUMA memory could have addresses that are not contiguous with system memory. (See h/w spec.)
o. Main VUMA memory that is contiguous to system memory must be disabled before OS boots.
p. VUMA device driver is responsible for enabling CPU access to Main VUMA memory. Note, all of Main VUMA memory access by the CPU is enabled when any part is enabled, i.e.. all or nothing. Disabling CPU access is not allowed at run time.
q. The Main VUMA memory must be contiguous, but it is not necessary to be contiguous with system memory.
r. If Main VUMA memory is not contiguous with system memory, CPU access does not need to be disabled prior to INT 19h.
s. When requesting Aux VUMA memory, if system memory is being cache by any type of cache, the cache must be cleared by an I/O instruction, not by reading memory. This is necessary since in protected mode a selector will not be available to the BIOS.

2.3 Boot Sequence

1. System BIOS sizes and configures (makes contiguous) all system memory.

2. System BIOS scans ROM space for VUMA devices and determine if there are any devices present that were not present at the last boot. If yes, then add the minimum amount of memory required by that device for booting, to VUMA memory.

3. Allocate VUMA memory. At this point all memory, including VUMA memory is enabled. If this is not possible to allocate all requested memory (possibly memory has been removed between boots) then the system will scan all VUMA ROMs and allocate the minimum necessary to boot.

4. Next, call the entry point to the VGA device. The VGA device tests and initializes it's memory at this time.

5. After the VGA device has initialized itself, control is given back to the system BIOS.

6. System BIOS then continues POST. During POST the system gives control to the other PCI devices (including VUMA devices). They then initialize themselves.

7. When the OS starts it's boot process, it will then load and execute the video driver. If necessary the video driver will then enable the CPU access to memory allocated to the VUMA device.

8. Any changes to the size of the memory allocated to the VUMA device will be requested by the video driver, O/S, or utility/properity sheet. These requests will then be implemented on the next boot.

3.0 VUMA VESA SYSTEM BIOS Extensions (VUMA-SBE)

The new system BIOS calls have that have been defined can be accessed via the following VUMA-SBE interfaces.

3.1 VUMA-SBE 32 bit interface.

Detecting the presence of the 32-bit interface for the VUMA-SBE functions is done using the BIOS32 Service Directory[1]. Use of the service directory involves 3 steps : locating the service directory, using the service directory to get the VUMA services entry point and finally calling the VUMA services to perform the desired function. The BIOS32 Service Directory is a contigous 16-byte data structure which begins on a 16-byte boundary somewhere in the physical address range 0E0000h - 0FFFFFh. It has the following format :

Offset	Size	Description
00h	4 bytes	ASCII Signature String _32_ This puts an underscore at offset 0, a '3' at offset 1, a '2' at offset 2 and another underscore offset 3.
04h	4 bytes	Entry point for BIOS32 Service directory This is a 32-bit physical address through which the service directory can be called.
08h	1 byte	Revision level The current revision level is 00h.
09h	1 byte	Length of data structure in paragraph (i.e., 16-byte) units. The data structure in this revision is 16 bytes long so this field has a value of 01h.
0Ah	1 byte	Checksum This field is a checksum of the complete data structure. It has a value such that when all of the bytes in the data structure are added together in a byte wide sum they add up to 00h.
0Bh	5 bytes	Reserved Must be 0

To locate the service directory a caller must scan 0E0000h to 0FFFFFh on 16-byte boundaries looking for the ASCII signature "_32_" and a valid checksummed data structure. If the service directory is NOT found then 32-bit VUMA support is not present in the BIOS.

[1] The BIOS32 Service Directory is an industry standard and is described by the document **Standard BIOS 32-bit Service Directory Proposal**, Revision 1.0 May24, 1993 available from Phoenix Technologies Ltd., Irvine, CA

To get the VUMA services entry point a CALL FAR to service directory is done using the value specified at offset 04h in the service directory data structure. The following is a list of the entry conditions and the return values when calling the service directory.

INPUT:

EAX	Service Identifier · This is a 4 character string used to identify which 32-bit BIOS service is being sought. For VUMA is it "VUMA" where EAX = 414D5556h (NOTE: This corresponds to mov eax, 'VUMA')
EBX[31:8]	Reserved Must be set to 00h
EBX[7:0]	00h
CS	Code selector set up to encompass the physical page holding the entry point as well as the immediately following physical page. It MUST have the same base. CS is execute only.
DS	Data selector set up to encompass the physical page holding the entry point as well as the immediately following physical page. It MUST have the same base. DS is read-only.
SS	Stack selector must provide at least 1K of stack space and be 32-bit.
I/O	I/O Permissions must be provided so that the BIOS can perform any I/O necessary.

OUTPUT:

AL	Return Code 00h - Requested service is present. 80h - Requested service is NOT present. 81h - Unimplemented function specified in BL.
EBX	Physical address to use as the selector BASE for the service.
ECX	Value to use as the selector LIMIT for the service.
EDX	Entry point for the service relative to the BASE returned in EBX.

Once the VUMA entry point has been found the caller should create an execute-only CODE and read-only DATA selectors based on the values in EBX and ECX. The VUMA entry point can now be invoked using a CALL FAR with the created CODE selector and the offset in EDX. The following additional conditions must exist when calling the VUMA entry point :

- I/O permissions are such that the service can perform any I/O necessary
- The stack is a 32-bit stack and provides at least 1K of stack space
- The appropriate privilege is provided so that the service can enable/disable interrupts are needed

All other register settings are specified to the function being called.

3.2 VUMA-SBE 16 bit interface.

The 16 bit Interface is function-based and all parameters are passed in registers. If a register is not specified as an output parameter for a function, then it will be preserved. All flags are preserved. Function values are passed as input parameters in register BL. Return status is passed back in register AL. A return status of 00h indicates that the function was successful.

Prior to calling into the 16-bit interface in protected mode using the PUSHF / CALL sequence the following requirements must be met :

- CS is an execute-only selector with a BASE of 0F0000h and a LIMIT of 64K.
- DS is a read-only selector with a BASE of 0F0000h and a LIMIT of 64K.
- I/O permissions are such that the service can perform any I/O necessary
- The stack is a 16-bit stack and provides at least 1K of stack space
- The appropriate privilege is provided so that the service can enable/disable interrupts are needed

Entry to the 16 bit interface may be done one of two ways:

1. Entry point to the 16 bit services is F000:F859. To call these services:
   Set up the registers as indicated in the function description. Status information is returned in AX.

   ```
   PUSHF
   CALL FAR F000:F859
   Check results
   ```

2. The 16 bit interface may also be accessed through the INT 15h instruction. The value F401h is passed in the AX register, with the subfunction passed in BL. Status information is returned in AX.

3.3 Status Information for the calls

Every function returns information in the AX register. The format of the status word is as follows:

AL = F4h: Function is supported.
AL = FFh: No error, but function **NOT** supported.
AL = 00h: Function error or not completed yet, see error codes in AH.

AH = 00h: Function call successful.
AH != 00h: Function call failed.
= 01h: Unknown PFA. PFA does not match devices in system.
= 02h: Invalid Input Argument.
= 03h: Too many banks (RAS/CS lines) requested.
= 04h: Requested bank(s) RAS/CS line(s) not supported.
= 05h: Aux Memory not supported.
= 06h: Noncacheable/ write through cache area not available.
= 80h: Function needs to be called again to return additional information.
= FFh: Other unknown error.

3.4 ROM Signature

VUMA devices must have a ROM signature, within the first 1K, "_VUMA_XXxx" where XX is major version and xx is minor version. Following the minor version number:

A. 16 bit value with the minimum amount of memory necessary, in 64Kblocks, for booting. It is not a requirement to have all devices working to boot. Only devices essencial to bring up a system, such as VGA and a boot device (hard drive) are necessary. After booting, a device driver or utility may request additional memory for non-essential devices.
B. 16 bit value with bit map of memory banks (RAS/CS lines) supported. Bit 0 corresponds to bank number 0 etc. If a bit is set then the bank (RAS/CS line) is supported by the VUMA device.
C. 16 bit value for DRAM support. Bit set if supported.
   Bit 0 = Fast Page
   Bit 1 = EDOn
   Bit 2 = SDRAM
   Bit 3 = PN EDO (Burst EDO)
   All other bit are reserved.
D. 8 bit value for features.
   Bit 0 = Snooping supported by VUMA device if set. (See h/w spec for definition of snooping.

3.5 VUMA-SBE Functions

The following defined VUMA-SBE services are not included in the VBE standard documentation.

3.5.0 00h - Report VUMA Core Logic Capabilities

This function should be called before any other VUMA-SBE function is called to ensure that the VUMA system is present, and to inquire the core logic capabilities

Input: (AX is used only when being called by one of the two 16 bit interfaces.)
AH = F4h
AL = 01h
BL = 00h

Output:
AX = Status (see **section 3.3**)
BL = Major BIOS revision = 01h
BH = Minor BIOS revision = 00h
CX = Banks (RAS/CS) that are supported, could have memory installed, by the core logic controller
Bit 0 = bank 0
Bit 1 = bank 1
etc.
DX[3:0] = Core logic capabilities
0 = No special features
Bit 0 = 1 -> Controller supports non-cacheable regions
Bit 1 = 1 -> Controller supports write-thru cache regions
Bit 2 = 0 -> Cannot change at run time from cached to non-cached and back
1 -> Can change at run time from cached to non-cached and back
Bit 3 = 0 -> Cannot change at run time from non-write through to write through and back
1 -> Can change at run time from cached to non-write through and back
DX[4] = Core logic supports snooping, this item is relevant only when synchrounous DRAM is supported
0 = Snooping is NOT supported
1 = Snooping is supported
SI = Bank (RAS/CS) numbers with memory. Bit set if has memory.
Bit 0 = bank 0

Bit 1 = bank 1
etc.

DI = Bank (RAS/CS) numbers with memory and support VUMA.
Bit 0 = bank 0
Bit 1 = bank 1
etc.

3.5.1 01h - Request VUMA Main Memory capabilities

This function returns system controller capabilities.

Input: (AX is used only when being called by one of the two 16 bit interfaces.)
AH = F4h
AL = 01h
BL = 01h

Output:
AX = Status (see **section 3.3**)
BX = Minimum size can allocate in 64 K increments
CX = Maximum size can allocate in 64 K increments
SI = System memory noncacheable or write through area granularity in 64 K blocks. Minimum block size region in system memory that can have L2 cacheable, non-cacheable, or write through cache. This is a basis provided for rounding up Aux memory size request.
0 = Not defined
DI = VUMA main memory size increments from minimum size in 64K. When memory is disabled the CPU does not have access to it but refresh still occurs.
0 = Not defined

3.5.2 02h - Set (Request) VUMA Main Memory for Next Boot

This function sets the size of the Main VUMA memory for the next boot. An input parameter is the memory bank numbers (RAS/CS numbers) that can be accessed by the VUMA device. The banks supported (parameter passed in DX) **must** be the same as reported in the ROM signature as specified in section 3.4 of this document.

Input: (AX is used only when being called by one of the two 16 bit interfaces.)
AH = F4h
AL = 01
BL = 02
CX = PFA number.
CH = Bus Number (0 .. 255)
CL[7:3] = Device number
CL[2:0] = Function Number
DX = Banks (RAS/CS) that are supported by the calling device.
Bit 0 = bank 0
Bit 1 = bank 1
etc.
SI = Size in 64 Kbytes (Will be rounded up by the system BIOS if necessary)

Output:
AX = Status (see **section 3.3**)
DX = Actual size in 64 K bytes allocated.

3.5.3 03h - Get VUMA Main Memory Size for Next Boot for a Device

This function returns the size of Main VUMA memory to be set for the next boot for the selected controller.

Input: (AX is used only when being called by one of the two 16 bit interfaces.)
AH = F4h
AL = 01h
BL = 03h
CX = PFA number
CH = Bus Number (0 .. 255)
CL[7:3] = Device number
CL[2:0] = Function Number

Output:
AX = Status (see **section 3.3**)
DX = Size in 64 K bytes

3.5.4 04h - Get Memory Size for Next Boot for all MAIN VUMA Memory.

This function returns the size of Main VUMA memory to be set for the next boot.

Input: (AX is used only when being called by one of the two 16 bit interfaces.)
AH = F4h
AL = 01h
BL = 04h
Output:
AX = Status (see **section 3.3**)
DX = Size in 64 K bytes

3.5.5 05h - Get Current Memory Size for a Device or for all VUMA Memory

This function returns the size of Main VUMA memory for the selected controller. Note: Value returned in BH is for all of VUMA main memory since all main memory is either enabled or disabled. (Allows CPU access or does not allow CPU access.)

Input: (AX is used only when being called by one of the two 16 bit interfaces.)
AH = F4h
AL = 01h
BL = 05h
CX = PFA number
CH = Bus Number (0 .. 255)
CL[7:3] = Device number
CL[2:0] = Function Number
If CX = FF then return for all devices

Output:
AX = Status (see **section 3.3**)
BH[0] = Memory access for all of Main VUMA memory.
0 -> Memory is not enabled, not visible to the CPU
1 -> Memory is enabled, visible to the CPU
CX = Bit map of bank (RAS/CS) numbers used. Bit set if bank is used.
Bit 0 = bank 0
Bit 1 = bank 1
etc.
DX = Size in 64 Kbytes
SI = upper 16 bits of physical start address
DI = lower 16 bits of physical start address

3.5.6 06h - Return Memory Speed/Type/Location/Size

This function returns information about the type of memory installed for the selected bank (RAS/CS). Fractions greater than 0.5 are rounded up. Fractions 0.5 and lower are rounded down. If a bank is logically divided into more than one area, then the function needs to be called more than once. AX indicates whether the function is done or not. If more than 1 bit in CX is set then error code AH= 03H will be returned, therefore only one bit should be set in CX when calling this function. Note: If a bank has contigous memory, but part of the memory is system memory and part is VUMA memory, the information will also be returned in two steps. BX[14] will reflect the type of memory.

Input: (AX is used only when being called by one of the two 16 bit interfaces.)
AH = F4h
AL = 01h
BL = 06h
CX = Bank (RAS/CS) number
Bit 0 = bank 0
Bit 1 = bank 1
etc.
DX = Serial calling number

Output:
AX = Status (see **section 3.3**, if equal to 80, must call for more info)
BX[15] = Reserved.
BX[14] = MAIN VUMA memory.
0 = not main VUMA memory, is system memory
1 = main VUMA memory.
BX[13:7] = Speed of memory in nano-seconds
0 = undefined, else value.
BX[6:0] = Core logic controller speed in nano-seconds
0 = undefined, else value.
CX[3:0] = Type of memory
0000 = Undefined
0001 = Fast page mode
0010 = EDO
0011 = SDRAM
0100 = PN EDO (Burst EDO)
CX[5:4] = CAS latency
CX[7:6] = Burst
0 = 1
1 = 2
2 = 4
3 = undefined

CX[8]	=	Burst order 0 = Linear 1 = Sequential
CX[15:9]	=	Voltage of memory in tenths of a volt 0 = undefined
DX[7:0]	=	Size in Mega-bytes
SI	=	upper 16 bits of physical start address
DI	=	lower 16 bits of physical start address

3.5.7 07h - Enable/disable Main VUMA memory

The ability to enable/disable CPU access to Main VUMA memory is not required if Main VUMA memory is not contiguous with system memory. If supported, this function enables/disables CPU access to Main VUMA memory. When any device makes this call, all devices that have main VUMA memory will be affected. When CPU access to main VUMA memory is disabled, access to video memory may be done through the PCI bus.
Note: During run time (after Int 19) CPU access can not be disabled.

Input: (AX is used only when being called by one of the two 16 bit interfaces.)
AH = F4h
AL = 01h
BL = 07h
BH[0] = Memory access
0 -> Enable CPU access to VUMA Main memory.
1 -> Disable CPU access to VUMA Main memory. (Can not be done at run time.)

Output:
AX = Status (see section 3.3)

3.5.8 08h - Set (Request)/Free VUMA Aux Memory

This function sets (requests) the size of the aux memory for use at run time. See "How To Access Aux VUMA Memory" in the Appendix to be added at a later time. A physical starting address and size is passed in. This function will flush and then turn off caching for this area or change the area to write through cache.

Input: (AX is used only when being called by one of the two 16 bit interfaces.)
AH = F4h
AL = 01
BL = 08
BH[1:0] = Type of cache
Bit 0 set if non-cachable
Bit 1 set if write-Thru
CX = PFA number
CH = Bus Number (0 .. 255)
CL[7:3] = Device number
CL[2:0] = Function Number
DX = Size in K bytes, free VUMA Aux memory if set to 0
SI = upper 16 bits of physical address
DI = lower 16 bits of physical address

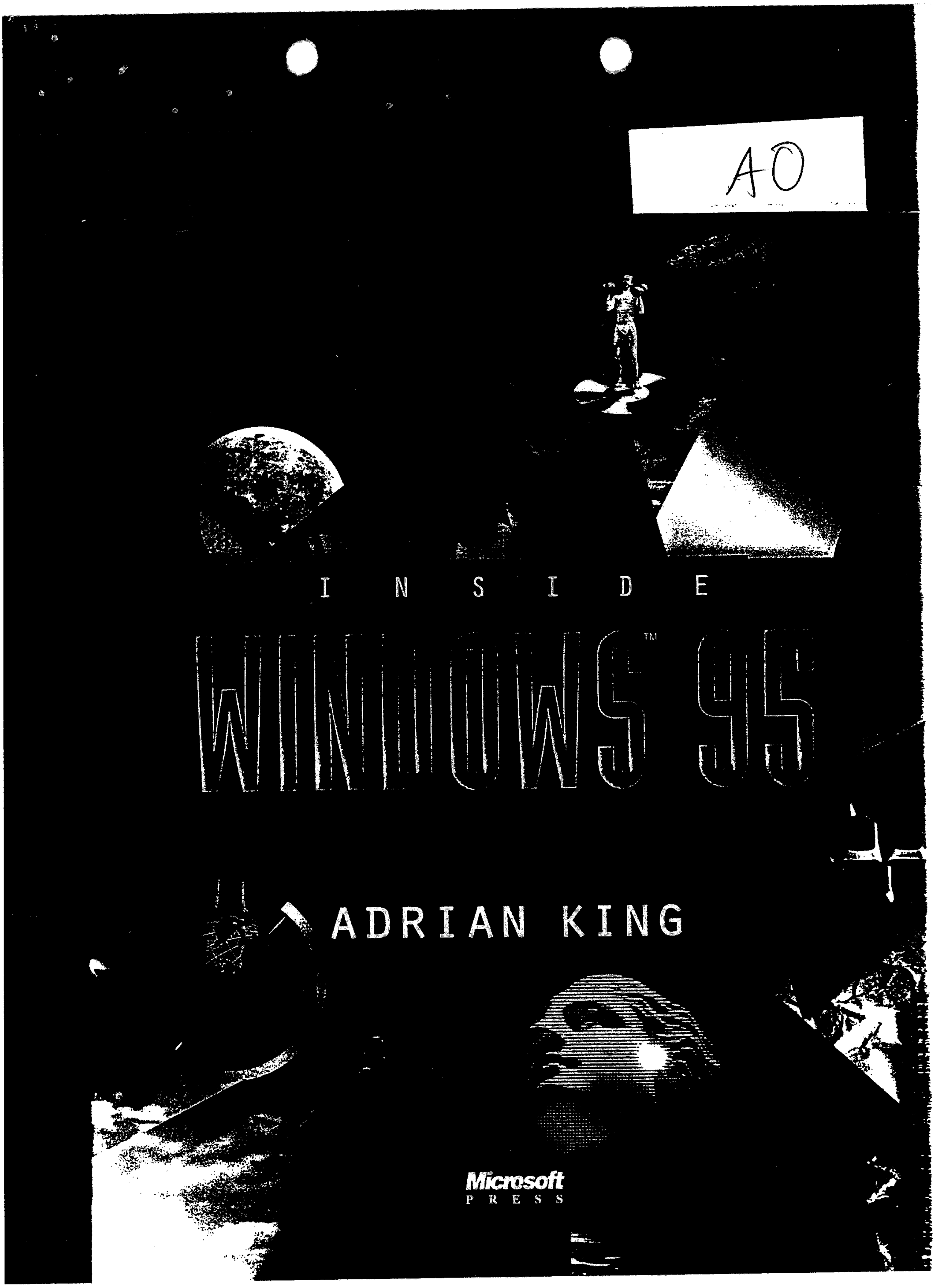
Output:
AX = Status (see **section 3.3**)
DX = Actual size in Kbytes allocated (rounded up by the system BIOS if necessary)

3.5.9 09h - Get VUMA Aux Memory Size

This function returns the size of the aux memory being used by a VUMA device.

Input: (AX is used only when being called by one of the two 16 bit interfaces.)
AH = F4h
AL = 01h
BL = 09h
CX = PFA number
CH = Bus Number (0 .. 255)
CL[7:3] = Device number
CL[2:0] = Function Number

Output:
AX = Status (see **section 3.3**)
DX = Size in Kbytes
SI = upper 16 bits of physical address
DI = lower 16 bits of physical address


AO
INSIDE
WINDOWS™ 95
ADRIAN KING
Microsoft
PRESS

PUBLISHED BY
Microsoft Press
A Division of Microsoft Corporation
One Microsoft Way
Redmond, Washington 98052-6399

Library of Congress Cataloging-in-Publication Data
King, Adrian, 1953-
Inside Windows 95 / Adrian King.
p. cm.
Includes index.
ISBN 1-55615-626-X
1. Windows (Computer programs) 2. Microsoft Windows (Computer file) I. Title.
QA76.76.W56K56 1994
005.4'469--dc20 93-48485
CIP

Printed and bound in the United States of America.

1 2 3 4 5 6 7 8 9 QMQM 9 8 7 6 5 4

Distributed to the book trade in Canada by Macmillan of Canada, a division of Canada Publishing Corporation.

A CIP catalogue record for this book is available from the British Library.

Microsoft Press books are available through booksellers and distributors worldwide. For further information about international editions, contact your local Microsoft Corporation office. Or contact Microsoft Press International directly at fax (206) 936-7329.

Acquisitions Editor: Mike Halvorson
Project Editor: Erin O'Connor
Technical Editors: Seth McEvoy and Dail Magee, Jr.

was originally designed as a standardized format for 32-bit protected mode code modules. There is an API, internal to the base system, that VxDs can use.[11] Obviously, the scope of these functions is at a much lower level than the scope of the services called on directly by applications.

Memory Management

Memory management in Windows takes place at two different levels: a level seen by the application programmer and an entirely different view seen by the operating system. Over the course of different releases of Windows, the application programmer has seen little change in the available memory management APIs. Within the system, however, the memory management changes have been dramatic. Originally, Windows was severely constrained by real mode and 1 megabyte of memory. Then expanded memory provided a little breathing room, and currently the use of enhanced mode and extended memory relieves many of the original constraints. Windows 95 goes further yet and essentially removes all the remaining memory constraints.

Windows 95 continues to support all the API functions present in Windows 3.1, and you can still build and run applications that use the segmented addressing scheme of the 286 processor. However, if you look at the detailed documentation for the Windows 95 memory management API, you'll see that all of the API functions originally designed to allow careful management of a segmented address space are now marked "obsolete." The "obsolete" list includes, for example, all the functions related to selector management. The reason, of course, is the Windows 95 support for 32-bit linear memory and the planned obsolescence of the segmented memory functions—yet another unsubtle hint that the Win32 API is the API you should be using to write Windows applications.

Although use of the 32-bit flat memory model simplifies a lot of Windows programming issues, it would be misleading to say that Windows memory management has suddenly gotten easy.[12] Windows 95 actually has a number of new application-level memory management

11. The Windows Device Driver Kit is the best reference for detailed information on VxDs and the associated API functions.

12. The Windows 95 documentation lists 45 API functions under the heading "Memory Management." The "obsolete" list numbers 28 API functions.

capabilities. All of the functions relate to the management of memory within the application's *address space,* the private virtual memory allocated to the process. The systemwide management of memory is the responsibility of the base system, and the Windows API aims to hide many of the details of the system's lower-level functions.

Application Virtual Memory

Figure 3-3 illustrates the basic layout of a Win32 application's virtual memory. Every Win32 application has a similar memory map, and each such address space is unique. However, it is still not fully protected: the private memory allocated to one Win32 application can be addressed by another application. The Win32 application's private address space is also the region in which the system allocates memory to satisfy application requests at runtime.

The system address space is used to map the system DLLs into the application's address space. Calls to the system DLLs become calls into this region. Applications can also request the dynamic allocation of memory by means of virtual addresses mapped to the shared region. Having virtual addresses mapped to the shared address space caters to the need for controlled sharing of memory with other applications.

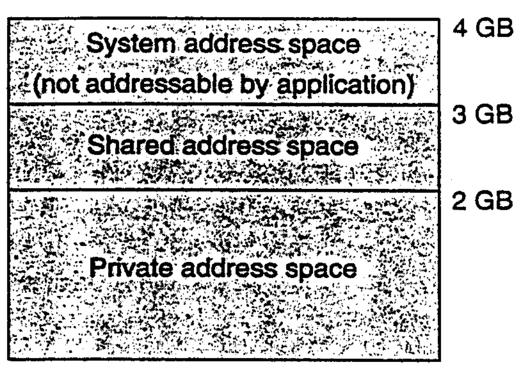


Figure 3-3.
Application virtual memory map.

Requests for memory at runtime fall into one of two categories: the application can make an explicit request for extra memory, or the system can respond to an implicit request for memory—that is, allocate memory to an application as a side effect of allocating some other resource. An implicit request occurs, for example, when an application creates a new window on screen: the system must allocate memory for the data structures used to manage the window. Windows 95 claims memory for resource allocation from a large 32-bit linear region rather than from the restrictive 64K segment used in previous versions of Windows. An ongoing problem in versions through Windows 3.1, running out of memory during resource allocation, has been largely eradicated in Windows 95.

Heap Allocation

In Windows parlance, the term *heap* describes the region of memory used to satisfy application memory allocation requests. In Windows 3.1, the system maintains both a *local heap* and a *global heap*. The local heap is a memory region within the application's address space, and the global heap is a memory region belonging to the system. As an application makes requests for local memory, its address space is adjusted to encompass the newly allocated memory. The system resolves requests for global memory from the same system memory pool used for all applications. It's possible to run out of either or both resources, although the use of a 2-GB address space makes this highly unlikely. Exhaustion of the local heap affects only a single application. Exhaustion of the global heap has systemwide repercussions.

Windows 3.1 programmers have to consider a variety of factors as they decide how to satisfy an application's runtime memory requirements. Windows 3.1 also has a range of API functions for manipulating dynamically allocated segments, and the manipulation of these shifting regions is further complicated by the underlying segmented memory model. It isn't just a chunk of memory that must be allocated. The application also needs a selector so that it can address the memory correctly. Under Windows 95, the Win32 application model does away with all these considerations. Selectors are no longer required—it's simply a 32-bit address that identifies the new memory—and the local and global heaps are merged into a single heap. The API functions that deal with selectors and the manipulation of memory regions in a segmented model all become obsolete.

Windows 95 Application Memory Management

For a Windows programmer, the Win32 API greatly simplifies the most common dynamic memory allocation chores. Furthermore, the increased capability of the underlying 32-bit architecture allowed the Windows designers to add a number of new functions for application memory management.

- Windows 95 provides functions that support private heaps whereby an application can reserve a part of memory within its own address space. The application can create and use as many private heaps as it wishes and can direct the system to satisfy subsequent memory allocation calls from a specific private heap. An application might use the local heap functions to create several different memory pools that each contain data structures of the same type and size.
- Windows 95 provides functions that allow an application to reserve a specific region of its own virtual address space that once reserved won't be used to satisfy any other dynamic memory allocation requests. In a multithreaded application, the 32-bit pointer to this reserved region is a simple way to provide each thread with access to the same memory.
- Memory mapped files allow different applications to share data. An application can open a named file and map a region of the file into its virtual address space. The data in the file is then directly addressable by means of a single 32-bit memory address. Other applications can open the same file, map it into their private address spaces, and reference the same data by means of a single pointer.

System Memory Management

Regardless of changes in the details of application memory management, the Windows programming model has remained pretty consistent through the different product releases. Allocating blocks of memory at runtime, using a reference to a block to manipulate it, and ultimately returning the block to the system for re-use is the way in which Windows programmers have always dealt with dynamic memory requirements. Windows 95 is no different. What has changed, however, is the way in which the system realizes the application's requests for dynamic memory.

Starting with the Windows 3.0 enhanced mode and continuing with the Windows 95 Win32 application model, the Windows API manipulates only the application's virtual address space. This means that an application request for a block of memory will adjust the application's virtual address map but might do absolutely nothing to the system's physical memory. Remember that the 386 deals with physical memory in pages each 4K in size. This page size is reflected in the virtual address space map of every Windows application. If an application requests 100K of memory, for example, its virtual address space will have 25 pages of memory added to it. The system will also adjust the data in its own control structures to reflect the application's new memory map.

However, at the time of allocation, Windows won't do anything to the physical memory in the system. It's only when the application starts to use the memory that the underlying system memory management kicks in and allocates physical memory pages to match the virtual memory references the application makes. If the application allocates but never references a region of its virtual memory space, the system might never allocate any physical memory to match the virtual memory. The ability of the 386 to allow physical memory pages to be used at different times within different virtual address spaces is the basis for the operating system's virtual memory capabilities.

Deep within the system are a range of memory management primitives available to device drivers and other system components that sometimes deal with virtual memory and sometimes force the system to commit actual physical memory pages. But these primitives are specific to the base operating system. Neither applications nor the Windows subsystem knows or cares about physical memory. Applications can force the system to allocate physical memory only by actually using the memory: namely, by reading from and writing to locations within a page. The separation of Windows memory management into the virtual and physical levels is a key aspect of the system. Applications and the Windows subsystems deal with defined APIs and virtual address spaces. The base system deals with physical memory as well as virtual address spaces.

Although physical memory is transparent to an application, its behavior can radically affect the performance of the system. For example, scanning through a two dimensional array of data row by row using C as the programming language will cause memory to be accessed from low to high virtual addresses because C stores two dimensional array data structures in *row major order.* As the memory sweep proceeds, the system will allocate physical memory pages to match the virtual memory accesses. Byte-at-a-time access will cause the system to allocate a new physical page every 4096 references. Other languages—FORTRAN, for example—store two dimensional arrays in *column major order.* Referencing the data row by row will generate memory references to widely scattered memory locations, forcing a much higher frequency of physical page allocation and much-reduced application performance. So, although the programmer doesn't have to worry about matching virtual memory to physical memory, it is a good idea for the programmer to know something about how the underlying system primitives and hardware support the application.

Windows Device Support

The most important aspect of the Windows device driver architecture is its ability to *virtualize* devices. (Yes, it's that word again.) The greatest difference between the device drivers of Windows 95 and Windows 3.1 is the extensive use of protected mode drivers in Windows 95—in fact, it will be unusual if your system uses any real mode drivers at all after you install Windows 95. The use of protected mode for the drivers pays off in terms of both system performance and robustness. The manufacturers of disk devices can adopt a new driver architecture—borrowed from Windows NT—that almost guarantees the availability of a protected mode driver for every hard disk. In addition, new protected mode drivers for CD ROM devices, serial ports, and the mouse make the possibility of needing to support a device with a real mode driver quite remote.

Device Virtualization

The device virtualization capability allows Windows 95 to use the memory and I/O port protection capabilities of the 386 processor to share devices among the different virtual machines. Every MS-DOS VM believes it has full control over its host PC and is unaware of the fact that it might be sharing the screen with other MS-DOS VMs or with the Windows applications running in the System VM. For MS-DOS applications, the display drivers must reside in the lowest level of the operating system. Many MS-DOS applications, particularly those that use the display in a graphics mode or use serial ports, will address the hardware directly. Windows has to intercept all such direct access in order to bring order to a potentially chaotic situation. The MS-DOS application knows nothing of the need to cooperate with other applications and certainly doesn't depend on a system device driver to get the job done. With Windows applications, the system has a slightly easier task since device access is always

AP

SGS-THOMSON
MICROELECTRONICS

TECHNICAL NOTE

MPEG VIDEO OVERVIEW

AN OVERVIEW OF THE MPEG COMPRESSION ALGORITHM

The MPEG standard was developed in response to industry needs for an efficient way of storing and retrieving video information on digital storage media. One inexpensive medium is the CD-ROM which can deliver data at approximately 1.2 Mbps, the MPEG standard was subsequently aimed at this data rate, in fact the data rate is variable and all decoders must be able to decode at rates up to 1.856 Mbps. Although the standard was developed with CD-ROM in mind, other storage and transmission media can include DAT, Winchester Disk, Optical Disk, ISDN and LAN.

Two other relevant international standards were also being developed during the work of the MPEG committee : H.261 by CCITT aimed at telecommunications applications and ISO 10918 by the ISO JPEG committee aimed at the coding of still pictures. Elements of both standards were incorporated into the MPEG standard, but subsequent development work by the committee resulted in coding elements found in neither.

Some of the participants in the MPEG committee include : Intel, Bellcore, DEC, IBM, JVC Corp, THOMSON CE, Philips CE, SGS-THOMSON, Sony Corp, NEC Corp and Matsushita EIC. These are not necessarily be the most important members of the committee but it gives an indication of the relevant importance of the MPEG standard.

Although the MPEG standard is quite flexible, the basic algorithms have been tuned to work well at data rates from 1 to 1.5 Mbps, at resolutions of about 350 by 250 Pixels at picture rates of up to 25 or 30 pictures per second. MPEG codes progressively-scanned images and does not recognise the concept of interlace; interlaced source video must be converted to a non interlace format prior to encoding. The format of the coded video allows forward play and pause, typical coding and decoding methods allow random access, fast forward and reverse play also, the requirements for these functions are very much application dependent and different encoding techniques will include varying levels of flexibility to account for these functions.

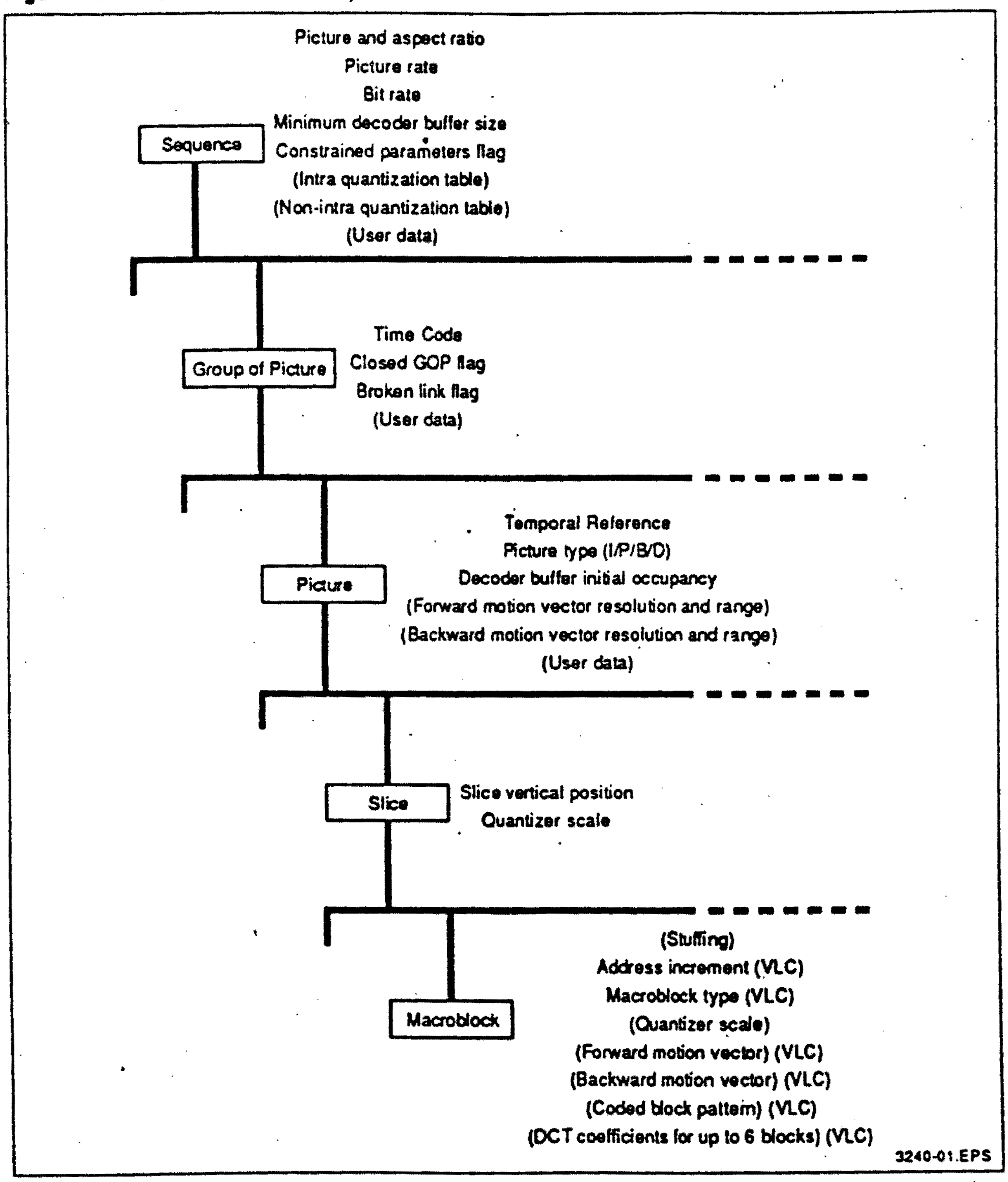
Compression of the digitised video comes from the use of several techniques : Sub sampling of the chroma information to match the human visual system, differential coding to exploit spatial redundancy, motion compensation to exploit temporal redundancy, Discrete Cosine Transform (DCT) to match typical image statistics, quantization, variable length coding, entropy coding and use of interpolated pictures.

ALGORITHM STRUCTURE AND TERMINOLOGY

The MPEG hierarchy is arranged into layers (Figure 1). This layered structure is designed for flexibility and management efficiency, each layer is intended to support a specific function i.e. the sequence layer specifies sequence parameters such as picture size, aspect ratio, picture rate, bit rate etcetera , whereas the picture layer defines parameters such as the temporal reference and picture type.

This layered structure improves robustness and reduces susceptibility to data corruption.

Figure 1 : MPEG Bistream.Hierarchy



For convenience of coding, macroblocks are divided into six blocks of component Pixels - four luma and two chroma (Cr and Cb) (Figure 2).

Blocks are the basic coding unit and the DCT is applied at this block level. Each block contains 64 component Pixels arranged in an 8x8 array (Figure 3)

Figure 2 : Macroblock Structure

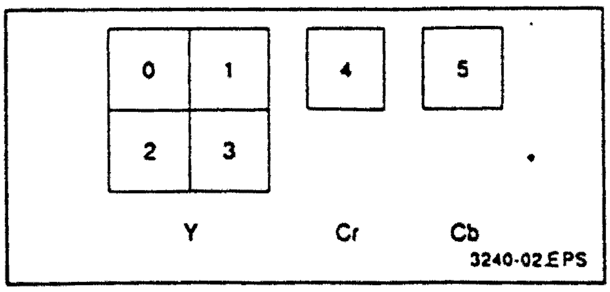
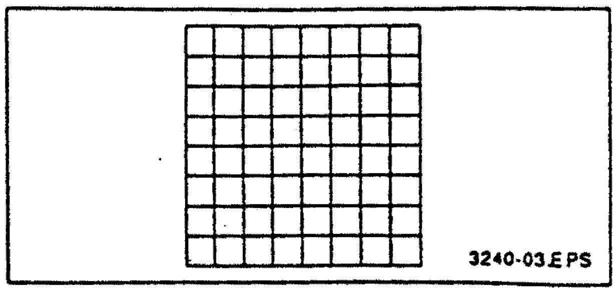


Figure 3 : Block Structure



There are four picture types : I pictures or INTRA pictures, which are coded without reference to any other pictures; P pictures or PREDICTED pictures which are coded using motion compensation from a previous picture; B pictures or BIDIRECTIONALLY predicted pictures which are coded using interpolation from a previous and a future picture and D pictures or DC pictures in which only the low frequency component is coded and which are only intended for fast forward search mode. B and P pictures are often called Inter pictures. Some other terminology that is often used are the terms M and N, M+1 represents the number of frames between successive I and P pictures whereas N+1 represents the number of frames between successive I pictures. M and N can be varied according to different applications and requirements such as fast random access. In Figure 4, M = 3 and N = 12.

A typical coding scheme will contain a mix of I,P and B pictures. A typical scheme will have an I picture every 10 to 15 pictures and two B pictures between succesive I and P pictures: refer to Figure 4.

MPEG COMPRESSION ALGORITHM

The MPEG algorithm is based around two key techniques : temporal compression and spatial compression. Temporal compression relies upon similarity between successive pictures using prediction and motion compensation whereas spatial compression relies upon redundancy within small areas of a picture and is based around the DCT transform, quantization and entropy coding techniques.

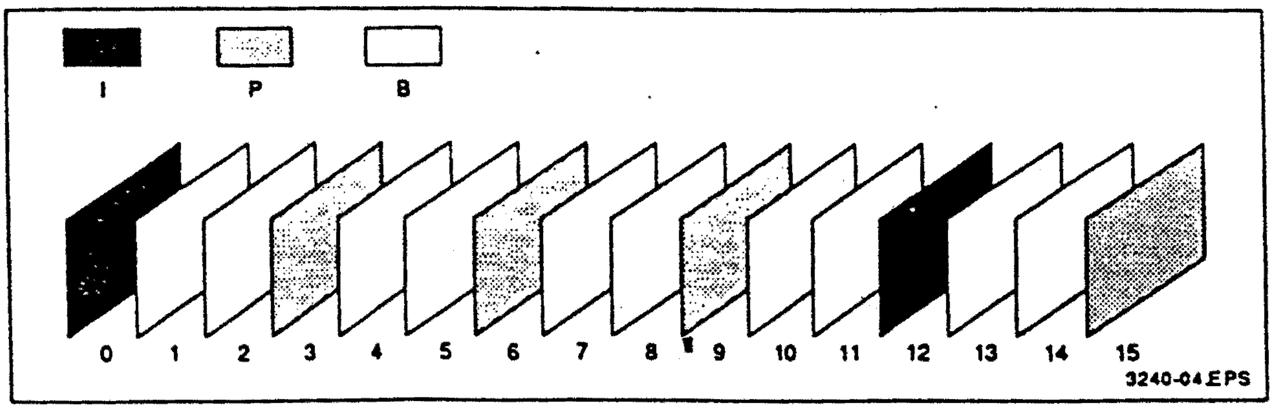
TEMPORAL COMPRESSION

Inter (B and P) pictures are coded using motion compensation, primarily prediction and interpolation.

Prediction

The predicted picture is the previous picture modified by motion compensation. Motion vectors are calculated for each macroblock. The motion vector is applied to all four luminance blocks in the macro block. The motion vector for both chrominance blocks is calculated from the luma vector. This technique relies upon the assumption that within a macroblock the difference between successive pictures can be represented simply as a vector transform (i.e. there is very little difference between successive pictures, the key difference being in position of the Pixels.).

Figure 4 : Typical sequence of pictures in display order



Interpolation

Interpolation (or bidirectional prediction) generates high compression in that the picture is represented simply as an interpolation between the past and future I or P pictures (again this is performed on a macroblock level).

Pictures are not transmitted in display order but in the order in which the decoder requires them to decode the bitstream (the decoder must of course have the reference picture(s) before any interpolated or predicted pictures can be decoded). The transmission order is shown in Figure 6.

Figure 5 : Make up of I, B and P pictures

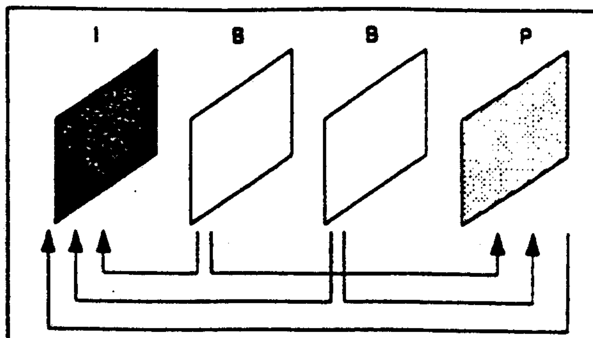
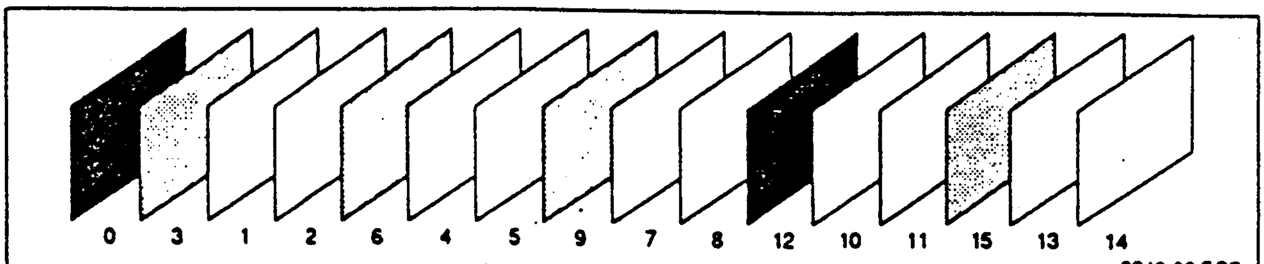


Figure 6 : Typical sequence of pictures in transmission order



SPATIAL COMPRESSION

The spatial compression techniques are similar to those of JPEG , DCT, Quantization and entropy coding. The compression algorithm takes advantage of the redundancy within each block (8 x 8 Pixels).

The resulting compressed datastream is made up of a combination of spatial and temporal compression techniques which best suit the type of picture being compressed. Decoding is controlled through the use of MPEG system codes which are put into the data stream explaining how to reconstruct specific areas of picture - as shown in Figure 1.

CONCLUSION

Through a combination of techniques, MPEG compression is designed to give good quality (typically similar or better quality to VHS) images from such storage media as CD-ROM. The quality is however, dependent upon the type of picture compressed and the level of redundancy within the sequence coded. Picture quality will also depend upon how well the sequence has been coded and which features are required - For Example : For fast random access, N will tend towards zero hence the quality of compression will deteriorate, if random access is not required then the number of P and B frames can increase, hence increasing the potential quality. The standard does not specify a method of compression but a syntax for the compressed data, this allows for differing compression techniques depending upon differing requirements. The decoding techniques are defined due to the nature of the compressed data stream.

This method allows for true flexibility in coding whilst retaining the format and hierarchy ensuring compatibility in the datastream and hence uniform readability.

ORDER CODE :

F3.

On the Bus Arbitration for MPEG 2 Video Decoder

Chia-Hsing Lin and Chein-Wei Jen

Department of Electronics Engineering and Institute of Electronics
National Chiao Tung University

Abstract

A bus arbitration scheme for the MPEG-2 video decoder VLSI developed by NCTU is proposed in this paper. Compared to the traditional pure stochastic bus scheduling scheme, the internal buffer requirement and bus arbitration overheads are reduced due to the deterministic nature of this strategy. This bus arbitration scheme has been verified using Verilog simulator and will be implemented in the NCTU MPEG-2 decoder.

1. Introduction

ISO standard 13818[1] known as MPEG-2 (Moving Pictures Expert Group) have been adopted in many applications like TV set-top boxes, PC add-on cards and entertainment machines. To promote the success of this motion picture standard, it is attractive to develop a single chip decoder, accompanied by DRAMs, to establish a low cost decoding system. The cheapness of standard DRAMs is the main reason for MPEG-2 decoder VLSI to use as the temporal picture buffer. However, the decoder VLSI also has to contain several internal buffers, which will increase the cost of this decoder, to conquer the limited memory bandwidth provided by DRAM. Therefore, it is important to design a suitable bus arbitration scheme for memory access to utilize the bandwidth efficiently in order to reduce the amount of internal buffers.

In this paper, we propose a bus arbitration scheme for MPEG-2 decoder of main profile and main level (MP@ML). We will first give an architectural overview and functional description of NCTU MPEG-2 decoder in the next section. The bottleneck issue of memory access and the proposed bus arbitration scheme will then be presented in section 3. Section 4 demonstrates some simulation results using uninterpreted model[2] and Verilog simulator. Section 5 concludes the paper.

2. MPEG-2 Decoder Design

The architecture of the MPEG-2 video decoder developed by NCTU is shown in Fig.1. The system controller provides controls for other functional units. The decoding pipeline (including variable-length decoder, inverse quantizer, inverse discrete cosine transform unit and motion compensation unit) performs the main MPEG-2 decoding operations. A 64-bit memory data bus is used for the I/O transactions between functional units and external memory (which is used as the VBV buffer and reference picture buffer). The memory I/O transactions are managed by a memory controller. The video interface controls the display timing for video output and performs some post-processing operations like the output format conversion from 4:2:0 to 4:2:2.

To perform the decoding and display processes, the decoder first receives compressed bitstream from host interface to bitstream buffer (BBUF) and transfers them to the VBV buffer, which is located in the external memory. The decoder will then re-read the bitstream from VBV buffer to VLD buffer (VLD BUF) for the requirement of decoding pipeline. If the macroblock currently decoded is nonintra-coded, the decoder may also need to load the reference blocks from reference picture buffer, which is also in the external memory, to perform motion-compensation and interpolation. After adding the results from IDCT and MC units, the decoder will write back the sums to the reference buffer. Finally, at the time to display the previous decoded data, the decoder will read video data again from reference picture buffer to video output buffer (VBUF). Fig.2 shows the timing diagram for each functional unit in the MPEG-2 decoder.

VLSI Tech, Syst. and Appl., 1995 Symposium.

3. Bus Arbitration for Memory Access

3.1 The Problem of limited Memory Bandwidth

In order to reduce the number of DRAMs and the number of I/O pins, the VBV buffer and reference picture buffer share the same external memory port. A memory bandwidth problem occurs because of the several memory I/O transactions (including the bitstream data loading and storing, video output loading, reference picture loading, and predicted data storing), and DRAM refreshing cycles. Also, the overheads introduced by stochastic bus arbitration between different transaction requests will worsen the bus load. The traditional bus arbiter using fixed priority scheme[3] may cause functional units to starve without large internal memories for I/O buffering because of the heavy memory bus load in MPEG-2 with CCIR and higher resolution. In [4], Tatsuhiko, et al. proposed a sophisticated scheme to reduce the memory bottleneck. The basic idea of this scheme is a combination of priority assignment and polling (Fig.3). However, the extra FIFO and internal memories are still required to accommodate the stall of the decoding pipeline due to the stochastic nature of this scheme.

3.2 The Proposed Scheduling Scheme for Memory Access

Unlike the previous pure stochastic scheduling scheme, a "pseudo-deterministic" scheme to allocate the bandwidth for each I/O transaction is proposed in this paper. For each macroblock prediction mode, we analyze the worst case in data transferring and allocate the required duration for each memory I/O in one macroblock period according to the following criteria:

$$N_{video} + N_{load} + N_{store} + N_{btio} + N_{refresh} + N_{overhead} \le N_{MB} \le \frac{\text{clock rate}}{(\text{no. of MBs in a frame}) \times (\text{frame rate})} \quad (1)$$

$$\left(\frac{N_{MB}}{N_{ratio} \times N_{width} \times N_{dis}} + N_{ov} \right) \times N_{access} \times N_{dis} \le N_{video} \quad (2)$$

where

N_{MB} is the number of cycles to decode one macroblock,

N_{video} is the number of cycles to transfer video output data to display buffer,

N_{load} is the number of cycles to read reference blocks from reference picture buffer,

N_{store} is the number of cycles to write predicted macroblock to display buffer,

N_{btio} is the number of cycles to read from and write to VBV buffer,

$N_{refresh}$ is the number of cycles to refresh DRAM,

$N_{overhead}$ is the bus arbitration overhead,

N_{ratio} is the ratio of system clock and video output frequencies.

N_{width} is the width of memory bus,

N_{ov} is the number of DRAM page mode overhead,

N_{access} is the number of cycles to access one word from external memory in page mode, and

N_{dis} is the number of samples to display for one pixel.

Furthermore, to guarantee that the display process does not overrun the decoding process, the decoding rate must be larger than the display rate. Hence one more condition must be held:

$$\frac{\text{No of pixels in one picture}}{\text{No. of samples output in one macroblock time}} \ge \frac{\text{No of pixels in active region of a picture}}{\text{No. of samples in one macroblock}} \quad (3)$$

where

$$\text{No. of samples output in one macroblock time} = \left(\frac{N_{MB}}{N_{ratio} \times N_{width} \times N_{dis}} \right)$$

After we determine suitable time period for each I/O transaction, we can schedule them in the decoding time domain as the state diagram for bus arbitration shown in Fig. 4. The memory controller normally monitors the I/O requests (i.e., polling) to or from VBV buffer and perform the compressed bitstream input and output. While it is time for the transaction of any other I/O process, the bus will be allocated to that process until its transaction encounters end. The memory controller will then return to the state to handle the memory access for VBV buffer input or output.

Fig.5 shows one example of the scheduling scheme for different macroblock prediction modes. Assume that the chip outputs one 8-bit video sample at 27 MHz for 4:2:2 format (converted from 4:2:0 encoded in MPEG-2:MP@ML, 720x480@30Hz). Also, the whole system operates at 27MHz that can access DRAM one word per 1.5 cycles in the fast page mode (cycle time 40ns). The decoder must output 480 bytes of previous decoded data for display and decode 384 bytes of data in one macroblock period (640 cycles@27MHz). While bi-directional-predicted macroblock is encountered (Fig. 5a and Fig. 5b), we will allocate more bus cycles for the loading of predicted blocks, which has relatively larger amount of data to be transferred. Although in this case we limit the bitstream I/O sustained rate to about 200Mbps, the rate is still far lager than the bit rate specified in MPEG-2:MP@ML (i.e., 15Mbps). For intra macroblock, on the other hand, more cycles will be allocated to bitstream I/O transactions because of the relatively lower compression ratio (Fig. 5c) in this type of macroblock. The display process will not overrun the decoding process because the criterion (3) is met:

$$\frac{858 \times 525}{480} = 938 \geq 900 = \frac{720 \times 480}{384}.$$

4. Simulation Results and Implementation

Fig. 6 shows the simulation results of the buffer occupancy using the fixed-priority dynamic scheduling and the proposed scheme. The test video sequence is "Flower garden" with bit rate 15Mbps and the simulation duration is one-frame time. Also, the simulation models are uninterpreted[2] to reduce the simulation time. Furthermore, we use intra-type data for bitstream input/output and inter-type data (frame picture and bidirectionally field-based prediction) for reference and predicted pictures. Although conditions with such heavy bus load for memory access could hardly occur in real case, it is useful to test the robustness of the arbitration scheme.

Here we fix the size of VLD buffer to 1kbit and observe the occupancy of bitstream buffer and video output buffer. Obviously, in the case adopting the fixed-priority dynamic scheduling scheme both of those buffer are larger than the ones in the case using the proposed scheme. Furthermore, although the buffer requirement in the latter case is smaller, the residual time for header decoding in a frame period is still larger than the one in the former case. It is because there exists less arbitration overhead with the proposed scheme. Table 1 summarizes the results.

The proposed bandwidth allocation scheme for memory access has been verified by Verilog simulation. We will implement this scheme in our MPEG-2 VLSI that is currently developed in NCTU.

5. Conclusion

Concluding, compared to the pure stochastic bus arbitration scheme, the proposed scheme reduces the required amount of internal I/O buffer and the overheads of bus arbitration for our MPEG-2 decoder. The only drawback is the little reduction in bitstream I/O sustained rate. We will implement this scheme in the NCTU MPEG-2 decoder.

6. Acknowledgment

This work was supported by the National Science Council under Grant NSC 79-0414-E009-008 and United Microelectronics Corporation.

7. References

[1] ISO/IEC CD 13818, "Generic Coding of Moving Pictures and Associated Audio," (MPEG-2).

[2] J. M. Schoem, editor, "Performance and Fault Modeling with VHDL," Prentice-Hall, 1992.

[3] Thierry Fautier, "VLSI Implementation of MPEG Decoders," in *Proc. IEEE ISCAS 94 Tutorials*, may 1994.

[4] T. Demura, T. Oto, K. Kitagaki, S. Ishiwata, G. Otomo, S. Michinaka, S. Suzuki, N. Goto, M. Matsui, H. Hara, T. Nagamatsu, K. Seta, T. Shimazawa, K. Maeguchi, T. Odaka, Y. Uetani, T. Oku, T. Yamakage and T. Sakurai, "A Single-Chip Video Decoder LSI," in *ISSCC*

Digest of Technical Papers, pp.72-72, Feb 1994.

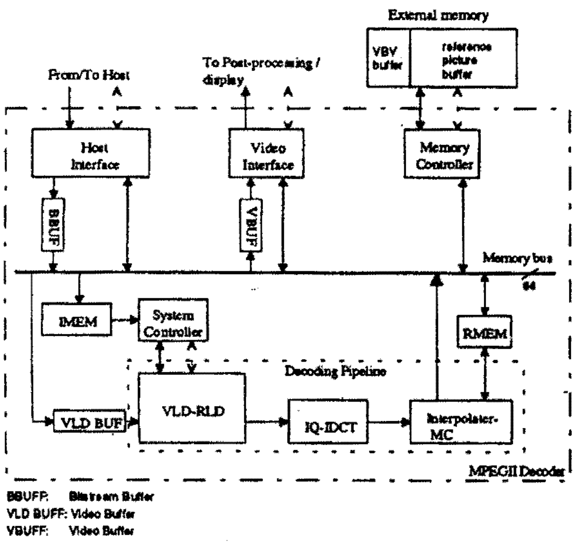


Fig.1 Architecture of the MPEG-2 decoder

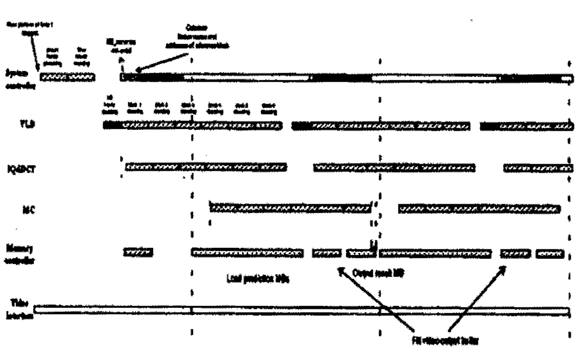


Fig.2 The decoding timing diagram of decoding pipeline (For frame picture, bidirectional field predicted macroblock).

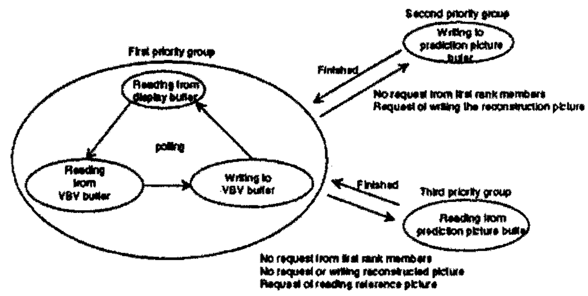


Fig. 3 The bus scheduling scheme proposed by Tatsuhiko Demura, et al.

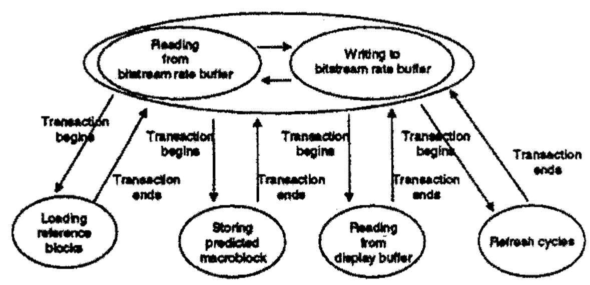


Fig. 4 The state diagram of proposed bus arbitration scheme.

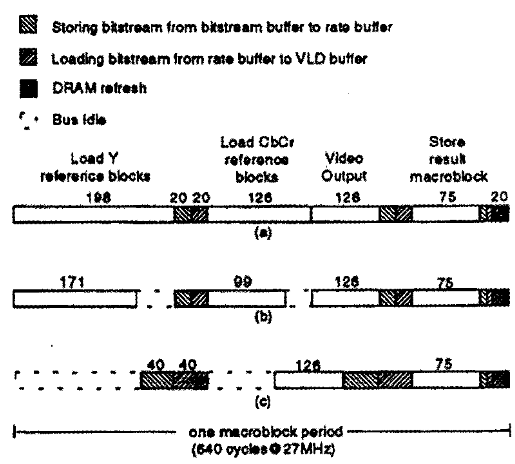
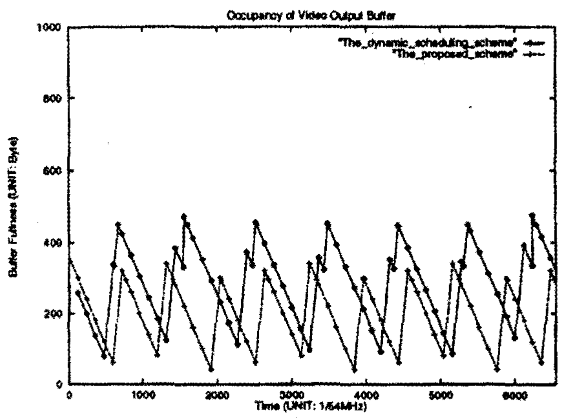
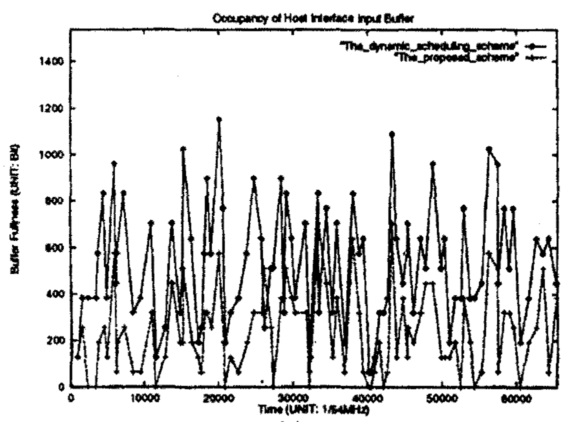


Fig.5 Examples using the proposed bus allocation scheme, (a) for frame picture, bidirectional field predicted macroblock (worst case), (b) for frame picture, bidirectional frame predicted macroblock, (c) for intra macroblock.



(a)

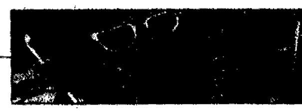


(b)

Fig. 6 Comparison of buffer occupancy in both scheduling schemes, (a) Video Output FIFO, (b) Host Input FIFO.

	The Fixed-Priority Dynamic scheduling Scheme	The Proposed Scheduling scheme	Saving
Size of Bitstream Buffer	1200bits	800bits	33%
Size of Video Output Buffer	480Bytes	360Bytes	25%
Residual Time for Header Decoding	20980cycles @27MHz	29610cycles @27MHz	4% (compared to one-frame time)

Table. 1 Comparison of buffer size and residual time for header decoding in both scheduling schemes.



A Low-Cost Graphics and Multimedia Workstation Chip Set

With just three VLSI parts, our latest workstation class lets designers optimize performance and cost at the system level. Its Hummingbird microprocessor features two-way superscalar execution incorporating two integer units, a floating-point unit, a 1-Kbyte internal instruction cache, an integrated external cache controller, an integrated memory and I/O controller, plus enhancements for little-endian and multimedia applications. Its Artist graphics controller integrates a graphical user interface accelerator, a frame buffer controller, and a video controller on a single chip.

Steve Undy

Mick Bass

Dave Hollenbeck

Wayne Kever

Larry Thayer

Hewlett-Packard

The computer system design approach known as disintegration spurns complex, highly integrated system components in favor of less complex, generic parts. A system house can have different design goals than its component supplier, leading to situations where the component vendor provides features on an integrated part not desired by the system house.

Using standard, off-the-shelf parts lets system designers pick and choose exactly the features they need without having to pay for unwanted ones. Component vendors may also charge premiums for integrated designs, cutting into the profits of the system house, which wants to add value to its products itself. Further, a system house may not want to depend on the availability of a vendor's complicated integrated design when shipping products to customers.

An alternative computer design approach is toward highly integrated parts and systems, a tack we have taken with the low-cost workstations we discuss here. As both a system house and a component vendor to itself, Hewlett-Packard can optimize a design for both performance and cost from a system perspective, giving it more flexibility in deciding where and how to place value-adding features. The system house thus can specify what features must be built into the components to precisely meet overall needs. Schedules, too, are now visible and their risks more controllable. Treating the end product—an entire workstation or server—as a whole rather than just a sum of its parts makes integration another degree of freedom in design optimization. In particular, the goal of the processor design team becomes overall system optimization rather than simply processor subsystem optimization.

A three-chip workstation system

Recently, we introduced a number of entry-level workstations and servers based on the Hummingbird PA7100LC processor. Figure 1 shows a block diagram of one of these computers, the HP 9000 Model 712/60 workstation. Because of integration, this design uses only three very large-scale integration parts. The Hummingbird processor chip connects directly to static cache RAMs and dynamic main memory RAMs. It also connects directly to the other two VLSI parts, named LASI and Artist, via a proprietary system bus. LASI, short for LAN (local-area network) and SCSI (Smaller Computer System Interface), provides a number of built-in I/O connections for the computer—RS-232, 16-bit stereo audio, and a parallel port among others—in addition to the two it was named for. The Artist chip is a graphics subsystem that connects directly to a color monitor.

Integrating so much onto the three VLSI parts was not an arbitrary choice. For example, integrating a memory controller onto the processor chip results in shorter cache miss penalties than

0272-1732/94/$04.00 © 1994 IEEE

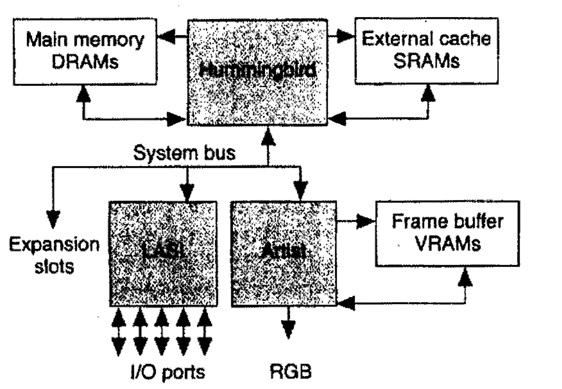


Figure 1. Model 712/60 block diagram.

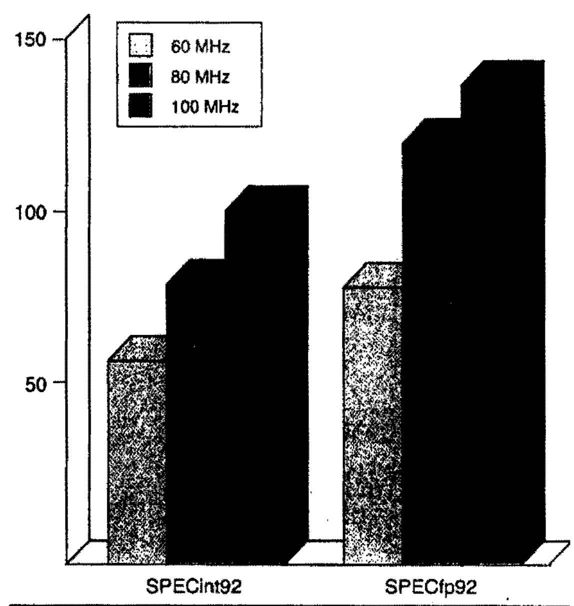


Figure 2. Benchmark performance.

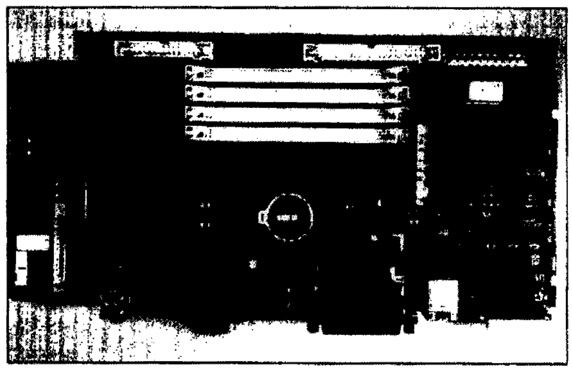


Figure 3. Processor board.

a nonintegrated solution. This, in turn, enables performance improvements—especially for memory-intensive applications. It also allows the design of systems with smaller caches relative to nonintegrated systems, without compromising system performance. The 712/60 uses a fairly small 64-Kbyte external cache. Including a second integer execution unit on the processor also improves performance. The direct connection between the processor and the graphics controller allows for fast data transfers and increased graphics performance. The net effect of integration on performance is that last year's mid-range workstation performance is now available on this year's entry-level workstation. Figure 2 gives SPECint92 and SPECfp92 benchmark performance ratings for the 60-MHz 712/60 and estimated numbers for the 80-MHz Series 800 Model E45. Also shown is the estimated performance for a system running at 100 MHz.

Integration also reduces costs. Figure 3 shows the single processor board used in the 712/60. The 712/60 uses a fraction of the components used in systems built just a couple of years ago. The number of parts used in our processors has steadily decreased from the first CMOS design completed in 1988. Figure 4a (next page) shows this integration trend for processors, with each rectangle representing one VLSI part. The number of components used in graphics controllers has likewise decreased over the years, as Figure 4b shows. As an example, we have incorporated the RAM digital-analog converter, used to generate video signals, directly into the Artist chip, saving both the cost of an external component and the board area it would have occupied. The LASI chip replaces the many separate components needed to provide the I/O connections expected on a workstation.

Hummingbird integrated processor chip

Hummingbird is the fourth in a series of CMOS PA-RISC processors,[1-5] though in many ways, it is a departure from the earlier designs. Rather than concentrating on producing the most performance possible from a given piece of silicon, we designed Hummingbird to be the most cost-effective solution without compromising performance. Hummingbird also uniquely integrates the memory controller, I/O bus controller, and cache controller onto the processor chip.

Design goals. Hummingbird's several design objectives are not just isolated component goals, but are constraints we derived by carefully considering the needs of the entire computer system. For example, the choice to place the memory controller on the processor chip actually increases the cost

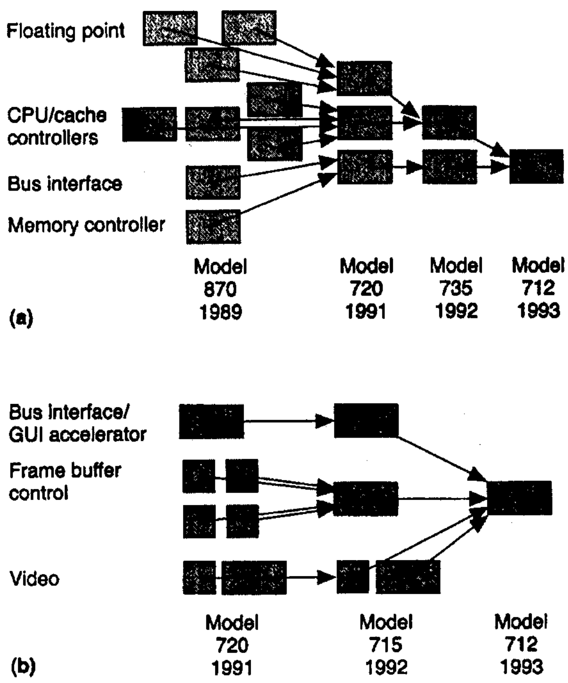


Figure 4. Integration trend: PA-RISC processor (a); graphics (b).

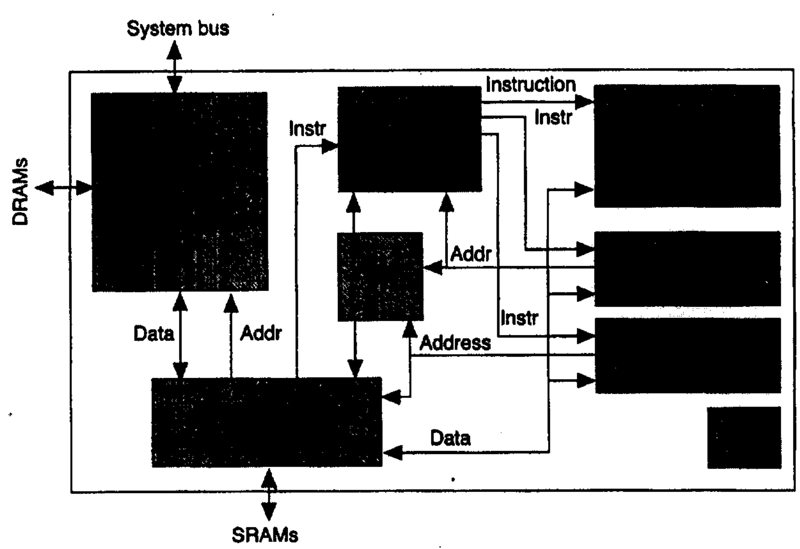


Figure 5. Hummingbird block diagram.

of the processor, but reduces the cost of the system. The goals include

- *Reduced cost—essential for competing in the very cost-sensitive, entry-level workstation market.* Integration of the memory controller helped lower system costs. Equally important, however, was the reduced cost cache organization that we implemented and support for industry-standard SRAMs, DRAMs, and memory SIMMs.
- *Uncompromised performance—considered system wide.* It was important that the cost objective not compromise processing power. Although Hummingbird boasts impressive integer and floating-point performance, it also has features to support high-performance graphics and multimedia applications. Included are a low-latency cache and memory system, new functional units and instructions, and an efficient system bus connection.
- *Inherently scalable—creating an easy upgrade path.* Hummingbird is scalable in clock rate, external cache sizes, main memory sizes, system bus clock ratios, and DRAM timing parameters.
- *Reduced power—achieved largely by using gated clocks and by eliminating dynamic circuit elements.*
- *Architecturally compliant—making Hummingbird completely compliant with the PA-RISC architecture.* Backward compatible with existing implementations, it includes extensions to improve the performance of little-endian and multimedia applications, and connection to standard I/O buses.
- *Improved manufacturability.* We wanted to reduce manufacturing costs and times by using standardized test methodologies and dedicated diagnostic circuitry.

Features. The Hummingbird CPU design leveraged many of its core technologies and features from the PA7100. Thus it has a pipeline design very similar to that of the PA7100, although we made several minor changes. As Figure 5 shows, Hummingbird is a two-way superscalar implementation incorporating two integer execution units, a floating-point execution unit, an internal instruction cache, a controller for external cache, and a main memory and I/O controller. It interfaces directly to static cache RAMs, as well as to standard DRAMs.

Dual-integer superscalar execution. Hummingbird has three execution

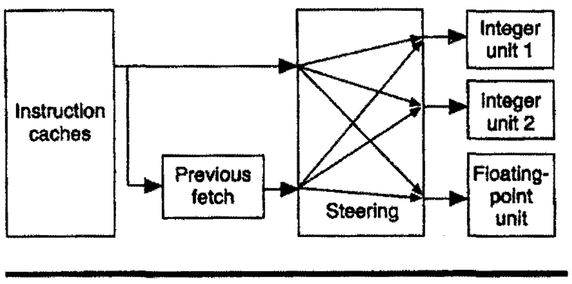


Figure 6. Instruction steering.

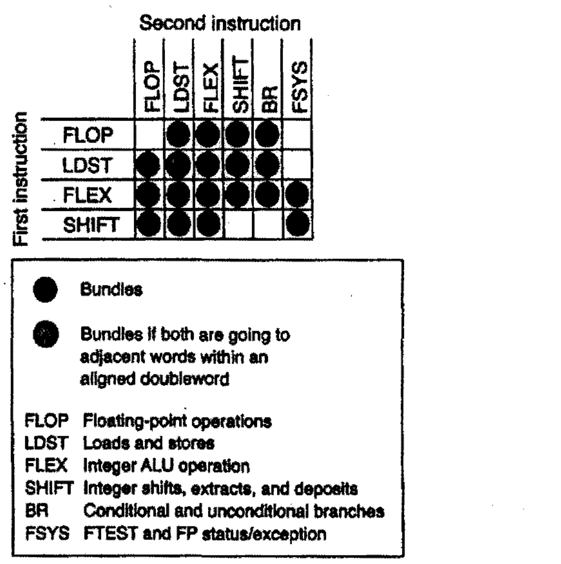


Figure 7. Superscalar bundling chart.

units. The first executes integer arithmetic, shift-type and branch instructions. The second executes integer arithmetic and memory reference instructions (both integer and floating-point). The third executes all floating-point arithmetic instructions.

The ability to execute two integer instructions simultaneously is a new feature for PA-RISC processors, requiring implementation of a second integer arithmetic logic unit. Through careful redesign of the integer datapath and by shrinking the translation look-aside buffer and other blocks from the PA7100, we made room for the second ALU in the chip floor plan. To keep costs down, we did not make the second execution unit as flexible as the first. Only the first execution unit, for instance, has the barrel shifter needed for shift and merge instructions. This relatively small investment in hardware lets us accelerate integer-only software by superscalar execution. The earlier PA7100 processor accelerated mostly floating-point applications by superscalar execution.

Every cycle, the instruction steering block (Figure 6) may issue an instruction to two of the three execution units. On each cycle, the instruction steering block fetches two instructions from the instruction cache. Depending on whether one or two instructions previously went to execution units, those instructions could be several instructions ahead of the program counter. Immediately after fetching instructions, the steering block examines them (along with any instructions from the previous fetch that have not yet executed) to determine which execution units they are to be directed to and whether two instructions may be bundled, that is, issued in the same cycle.

Several considerations arise for determining if two candidate instructions may be bundled. First is functional unit availability. For instance, with only one shifter implemented, only one shift instruction may issue per cycle. This does not tend to limit performance, as shifter use occurs less frequently than does ALU use. Figure 7 shows the combinations of instructions that can be bundled.

The second consideration concerns register dependencies. Even though the instruction steering block can bundle two addition instructions, it cannot do so if the second uses the result of the first. Also, it will not bundle some instructions, especially those that modify global resources such as the TLB or control registers, due to the complexity in determining dependencies. Branches are not bundled with the following instruction. The PA-RISC architecture[6] incorporates a concept called nullification, in which certain instructions can cause the following instruction to be nullified (not executed). The instruction steering block will not bundle instructions that can cause nullification with the following instruction. Like many architectures, PA-RISC uses delayed branching, where the processor fetches the instruction immediately following the branch, regardless of whether or not the branch is taken. We call this instruction the delay-slot instruction; it is not bundled.

Hummingbird has no address alignment constraints on bundles. It also allows the bundling of two load or store instructions referencing adjacent words in memory so long as they do not cross a double-word boundary. In this case, only a single double-word address—generated by the second integer execution unit—suffices for both instructions. Therefore, the two loads or stores may be bundled together. We designed special hardware to detect this case quickly enough to make the decision to bundle. Code that performs loads and stores to linear address ranges—especially procedure calls and context switches—will see an acceleration by this feature.

Cost-effective floating point. The floating-point unit (FPU) design for Hummingbird supports two goals: reduced system cost compared with PA7100-based systems and high performance for graphics. Floating-point performance is critical to

Table 1. Floating-point latencies and issue rates.

	Single precision			Double precision		
	Latency	Issue rate	Stalls	Latency	Issue rate	Stalls
Add/subtract	2	1	0	2	1	0
Multiply	2	1	0	3	2	1
Mpyadd/mpysub	2	1	0	3	2	1
Divide	8	8	7	15	15	14
Square root	8	8	7	15	15	14

graphics performance in PA-RISC systems because a significant amount of graphics processing takes place in the CPU. Fortunately, the PA7100 FPU provided an excellent starting point for performance, so we focused our design decisions on reducing cost without affecting performance for the targeted market. We wanted area reduction in the FPU to enable the integration of new features on the chip, such as the memory controller. We needed reduced power to minimize system power supply costs and cooling fan noise.

In graphics processing, only single-precision (32-bit) floating-point performance is critical. We could thus perhaps sacrifice some double-precision (64-bit) floating-point performance to make room on the chip for the memory controller. The PA7100 FPU has separate units for multiply, divide/square root, and ALU operations. Of these, only the multiplier architecture promised substantial area savings without a major redesign effort. For Hummingbird, we cut the array in half so that single-precision operations make one pass, while double-precision operations circulate their partial products through a second time before the final addition and rounding. Double-precision multiply is now a three-cycle operation, and a new operation can start every two cycles. This change reduced the multiplier power consumption because of the reduced amount of circuitry active on any cycle. Single-precision performance is unaffected and remains a two-cycle operation, where a new operation can start each cycle.

The change to the multiply latency and issue rate brought up an issue in the control logic. The two-cycle latency operations fit inside the normal five-stage pipeline of the CPU. Operations with longer latency require more control logic to avoid pipeline stalls. We elected to take an unconditional pipeline stall on any operation longer than two cycles. In practice, data dependencies often force these stalls anyway, so the performance impact is quite small, even for double-precision floating-point applications. By reducing the control logic we also save area. The number of register dependency comparators fell by 30 percent, and the random logic control core cell count dropped by 15 percent. Treating the long-latency operations in a simple, uniform way greatly simplified the controller design task. Table 1 summarizes latencies and issue rates.

The biggest opportunity for power savings in dynamic circuits comes from making evaluation conditional. This way, Hummingbird only draws current from the supply when precharging the logic following an evaluation cycle. The floating-point data path is composed almost entirely of dynamic logic to satisfy speed and area constraints. Given Hummingbird's lower frequency goal, we buffered the clocks into the three floating-point math units and qualified them with control signals. The three units have separate power switches. Whenever a valid floating-point operation begins, a power token gets passed along with the data, flowing through the pipeline and causing each stage to evaluate only on the cycle it is needed. With a continuous stream of flops, all the stages are active at the same time. However, whenever there are states on which new flops do not start, only those pipe stages with real work to do are active. Even in most floating-point benchmark programs, many states arise in which at least part of each math unit can remain inactive.

Cost- and performance-optimized caches and TLB. Like its predecessors, Hummingbird cycles its external cache at the processor frequency, allowing load instructions to execute every cycle without penalty. Unlike its predecessors, its external cache is combined, containing both instructions and data, and has a small (1-Kbyte) internal instruction cache. Even though we designed Hummingbird for low-cost systems, we had several reasons for retaining a single-cycle external cache. First, we felt that the silicon area on Hummingbird was better spent on other features (such as a second integer ALU and a memory controller) than on a relatively small data cache. Second, for low-cost systems running at moderate frequencies, our design does not require aggressive—costly—SRAM specifications. In fact, systems based on relatively slow 12-ns parts can run up to 66 MHz. The design required only 12 such parts. Lastly, the external cache organization allows for a greater degree of scalability and flexibility than a fixed-size internal cache.

We added the internal instruction cache to supply the needed instruction fetch bandwidth, as both instruction and data caches can be referenced in a single cycle. The caches are virtually indexed and physically tagged. The external cache has a 32-byte cache line size, while the internal cache has an 8-byte cache line size. Developers can configure the external cache size between 8 Kbytes and 2 Mbytes.

Hummingbird implements a two-level instruction cache hierarchy. The first level is the 1-Kbyte internal cache and the second level is one half of the external cache. The first level is a strict subset of the second. Both can provide two instructions every cycle. If a typical operation detects a first-level instruction cache miss, it forwards the instruction fetch to the second-level cache. If the second-level access hits, the cache controller forwards the double-word of instructions to the instruction steering logic while also sending it to the first-level cache for insertion. If the second-level cache indicates a miss, the memory controller begins handling the miss.

Load and store instructions represent only approximately 40 percent of the total instruction mix for PA-RISC processors. Consequently, bandwidth is available to the external cache, which contains both the data cache and the second-level instruction cache. Taking advantage of this extra bandwidth is a prefetching machine that copies instructions from the second-level cache to the first (see Figure 8). Hummingbird will perform this prefetch every cycle that the external cache is not busy satisfying a data reference. The prefetch machine attempts to stay ahead of the program counter so that a first-level miss will not occur. At times, enough data references block the external cache that the prefetching machine cannot keep up with the program counter. If so, the prefetch machine advances to the current instruction fetch address to make future prefetches useful.

Prefetched instructions go into a two-entry queue of instructions to be written to the first-level cache. Writes into the first-level cache from this queue proceed in parallel with reads from the first-level cache. An instruction fetch may use an instruction out of this queue without penalty. If a first-level cache miss is detected at the same time a prefetch is in progress for that address, the instruction goes directly to the instruction-steering logic from the external cache, reducing the normal instruction miss penalty by one cycle. Branches take advantage of this feature by beginning a prefetch to the target of the branch immediately after issuing the target address to the first-level cache. After a branch is taken, the prefetch machine will begin prefetching from the new program counter location.

The data cache on Hummingbird is a conventional single-level external cache. Reads from the external cache require a single processor cycle—even at 100 MHz. Writes, however, require two consecutive cycles. Since store instructions generally must read the tag portion of the cache before writing the data portion, the design uses store pipelining. This optimization technique entails using separate address lines for the external tag and data SRAMs. This, in turn, allows the cache controller to read the tag for a given store at the same time it writes the data for the prior store. Thus store instructions effectively use only two cycles of cache bandwidth employing standard asynchronous SRAMs. The data cache uses another store optimization that involves only stalling the pipeline if a instruction bundle containing a store (which will begin a two-cycle cache sequence) immediately precedes a bundle containing a data reference. In this way, the data cache can effectively hide the extra cycle of cache bandwidth needed by a store instruction if the instructions executed on the following cycle do not need to access the cache.

The advantage of integrating a memory controller on the same chip as the CPU becomes apparent when second-level

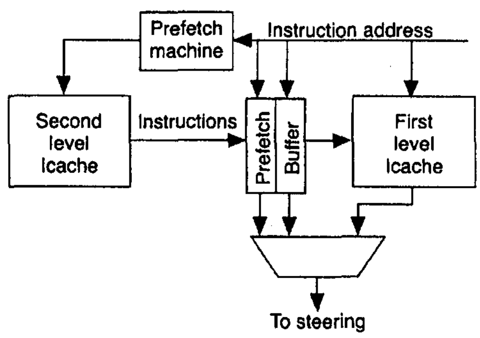


Figure 8. Instruction prefetching.

instruction or data cache misses occur. The cache controller is tightly coupled to the memory controller: the memory controller detects and begins handling a cache miss at the same time the CPU detects the miss. The cache controller uses several techniques to reduce the penalties associated with cache misses. It uses instruction streaming on second-level instruction cache misses, which allows the CPU to continue executing as soon as the first, or critical, double-word arrives from the memory controller. It writes the double-word to both levels of instruction cache while the CPU steps, or continues execution. This will occur for each double-word until all are written. Another feature, called stall-on-use, lets the CPU continue executing after it detects a data cache miss on a load instruction.

The cache controller can handle up to two outstanding cache misses at a time. Even though the CPU will stop stepping after detecting a second cache miss while a cache miss is in progress, it will resume stepping as soon as the cache line move-in for the first miss completes. This feature allows the memory controller to optimize misses to consecutive cache lines.

The virtual memory system for Hummingbird is essentially the same as that on the PA7100. We reduced the TLB from 120 entries to 64 to save area, although it remains fully associative. The TLB also contains eight block TLB entries for mapping large (512 Kbyte to 64 Mbyte) contiguous address ranges.

Tightly coupled memory system. The design of the memory system reflects the system-level design goals of low cost and power with high performance and scalability. We translated the system design goals into the following objectives for the memory system. The memory system should use the lowest cost commodity parts available at any given time. It should enable versatile system design by allowing a wide range of possible main memory sizes for scalability. The memory system should be capable of maintaining good performance levels, at the lowest possible cost, over a wide

Table 2. Memory system performance (at 60 MHz).

Transaction	Value
Miss occurrence to critical data	
Data cache miss	7 cycles
Instruction cache miss:	
Prefetch hit	4 cycles
Prefetch miss	7 cycles
Cache move-in bandwidth	
Page mode	160 Mbytes/s
Nonpage mode	107 Mbytes/s

range of system frequencies. The memory controller design should be simple, helping to achieve low development cost through first-time correctness, small area, and ease of testing.

Integrating the memory controller onto the same die as the CPU provides the memory controller with access to many important CPU internal resources. This enables performance gains that would not be possible in a nonintegrated solution. For example, the memory controller can eavesdrop on the real page number produced by the TLB. It can use this information to drive addresses to the DRAM before the occurrence of a miss is known. This speculative address issue saves a cycle on memory latency for cache misses. Integration also allows more effective use of the fast page mode of the DRAM than would otherwise be possible. Due to the early detection of cache misses, our design can in some cases avoid DRAM precharge penalties that a stand-alone memory controller could not. To further reduce miss penalties, the memory controller returns missing data to the cache in a critical-word-first fashion.

The memory controller implements an instruction prefetching algorithm. This prefetch mechanism occurs between memory and the instruction caches, and is in addition to the second-level cache to first-level cache prefetching described earlier. The algorithm very effectively reduces second-level instruction cache miss penalties, due to the proximity of the prefetch buffer to the CPU core. In the case of an instruction prefetch buffer hit, data can be sourced to the CPU and execution can continue within four CPU cycles of the detection of the second-level instruction cache miss.

The memory controller shares a four-entry transaction queue with the I/O controller. The transaction queue in many cases allows the CPU to continue execution, while the memory controller performs the queued transactions. When a cache miss occurs in which the cache line to be replaced has been modified, the memory controller queues the modified data while fetching the missing data from memory. Only then does it post the modified data to memory. Table 2 shows some performance characteristics of a typical Hummingbird memory system.

The memory controller is versatile enough to allow use of state-of-the-art commodity parts throughout the expected lifetime of the product. Industry-standard DRAM SIMMS form the system's main memory. The main memory data bus is 72-bits wide. Eight of the 72 bits serve for an error correcting code that can correct any single-bit error and detect any double-bit error. Since SIMMs of different types require different address bits to be multiplexed into the row and column addresses, the memory controller implements the address multiplexing function in a programmable fashion, memory card by memory card. This approach maximizes flexibility in the type of memory that may be installed in the system.

We built the memory controller with system scalability in mind. Systems may be built with as few as one, and as many as 16 SIMM slots, providing possible main memory sizes of 4 Mbytes to 2 Gbytes. Delays between DRAM address, control, and data edges are programmable, allowing for tailoring the speed (and cost) of the DRAM used for main memory to system requirements. The design supports DRAMs that implement an extended-data-out mode, providing superior page mode bandwidth at higher system frequencies.

Some systems may require buffering of some or all of the DRAM control lines. All DRAM control lines have programmable sense—active high versus active low—for this reason. The sense of each of the control lines may be programmed independently, allowing maximum system design flexibility.

Although the Hummingbird system caches are smaller than those in previous systems, the cycles per instruction contributions due to cache misses are on the same order as in systems with larger caches. By drastically reducing miss penalties through an integrated approach, our design maintains good performance at a lower system cost.

High-bandwidth I/O system. The I/O system uses a 32-bit bus onto which addresses and data are multiplexed. This substantially lowers the pin count and cost from a nonmultiplexed bus, thus allowing integration of the I/O controller onto the same die as the CPU and memory controller. Tight coupling between the I/O bus and the CPU and memory controller maintains performance, as does an efficient I/O protocol.

The I/O controller performs I/O reads and writes on behalf of the CPU, and direct-memory access on behalf of masters residing on the I/O bus. A transaction queue, shared with the memory controller, receives all CPU I/O requests, allowing the CPU to continue execution, in most cases, while the I/O transaction proceeds. DMA requests always insert directly into the head of the transaction queue. Addresses issue in a speculative manner to the DRAM address bus from the I/O bus when the I/O bus is not granted to the CPU. This not only benefits performance, but also allows the memory controller to handle DMA in the same way that it handles memory requests from the CPU, reducing the design complexity.

We paid particular attention to performance at the system level while designing the I/O system. For example, the processor retains access to main memory while the I/O bus is granted to an external master that is performing DMA. The memory controller alternates between memory requests from the CPU and DMA requests from the I/O system.

The ability of the CPU to quickly move data from main memory to the I/O system is essential for good system graphics performance. The memory and I/O systems work together to allow overlapped execution of processor reads from memory and processor writes to I/O devices. The CPU can attain a bandwidth of 50 Mbytes/s from main memory to I/O by using this technique, without requiring special block move or DMA hardware.

The design of the I/O system also reflects system scalability. We structured the I/O bus to operate properly to a frequency of 40 MHz. The CPU-to-IO bus frequency ratio is programmable to either 2:1 or 3:1. For maximum system design flexibility, we left the system arbitration logic off chip.

New architectural extensions for flexibility and performance. The Hummingbird CPU is completely compliant with the PA-RISC 1.1 architecture.[6] Existing code will automatically be accelerated by the performance features we implemented, although newer compilers take better advantage of the superscalar abilities of the CPU. Besides being backwards compatible, Hummingbird also implements several new extensions to the architecture: little-endian addressing, uncachable memory pages, and multimedia-oriented instructions.

Hummingbird supports both big-endian addressing, which all previous PA-RISC processors implement, and little-endian addressing. The difference between the two modes specifically deals with the order of bytes within larger data quantities and can be conceptualized as whether the most significant, or leftmost, byte in a 4-byte register will be loaded from or stored to byte address 0 or 3. This may seem trivial, but many programs implicitly assume one byte order or the other, therefore representing a roadblock to porting software between computers having different byte-endian addressing. We wanted to tap into the large pool of software written for little-endian processors but still remain compatible with existing PA-RISC code. Thus we added a mode bit to the PA-RISC processor architecture that selects between big- and little-endian byte addressing. Called the E bit, we put it into the processor status word so that it can vary from process to process. A single workstation thus can run both big- and little-endian applications concurrently. The dynamic nature of this bit dictated that memory be one endianess or the other (chosen to be big-endian on Hummingbird) and that data quantities be either byte-swapped or not on transfers between the CPU's registers and memory. In this way, both big- and little-endian software consistently treat a datum correctly that they are processing.

Certain types of software can be better optimized if some memory pages never get loaded into the data cache, for example, a device driver that communicates with an I/O device by reading and writing messages in main memory locations. If memory is always cachable, the driver must execute time-consuming cache flushes to prevent memory writes caused by cache line replacements from corrupting the I/O device's messages. Hummingbird supports uncachable pages because we added another bit, called the U bit, to each TLB entry. This bit controls whether a data cache miss to memory space will cause a move-in of the target memory line or not.

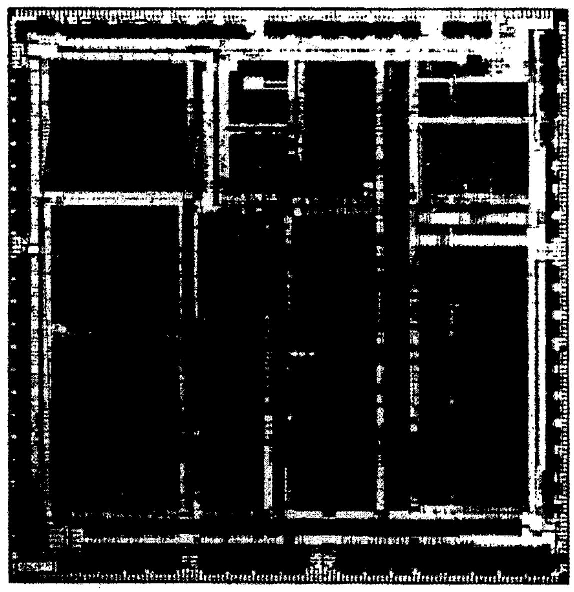


Figure 9. Hummingbird die.

An active area of multimedia research at Hewlett-Packard involves the algorithms used to decompress real-time audio and video information. Performance research suggested that many of the algorithms studied frequently used a few operations: addition and subtraction with either modular arithmetic or saturation, taking the average of two numbers, and multiplication by a small constant. Saturation clips the result to the largest value on positive overflow or clips to the smallest value on negative overflow. We have speeded up all these operations in Hummingbird. Each integer execution unit can execute two of these operations together, meaning that with the two integer units, four operations can occur simultaneously, thus accelerating the various multimedia algorithms substantially. These multimedia-motivated enhancements added insignificant (less than 0.2-percent) silicon area, while improving performance substantially and without requiring a dedicated multimedia accelerator chip.

Figure 9 shows a die photograph of Hummingbird; Table 3 (next page) gives some of the particulars about the chip design.

Table 3. Hummingbird details.

Parameter	Value
Transistors	900,000
Die size	14×14 mm
Metal layers	3
L_{eff}	0.61 μm (NFETs) 0.66 μm (PFETs)
Frequency	0-100 MHz
Power (60 MHz)	9.0W (worst case) 6.8W (typical)
Package	432-pin CPGA 1.8×1.8-inch
Supplies	5V, 3.3V

Artist integrated graphics chip

Coupled to the Hummingbird processor is a single-chip graphics system that complements the capabilities of the processor (see Figure 1).

Design goals. We designed the Artist graphics system to perform well in three areas:

- *Fast 2D graphical user interface.* Nearly every computer user has grown accustomed to running some sort of GUI. Speed is important for general user productivity.
- *Efficient 3D graphics.* HP's PowerShade software enables 3D graphics on even the least expensive workstation systems.
- *Digital video decompression.* Most of today's solutions require significant additional hardware. To meet the cost goals of the target workstation, we needed to provide this capability without additional hardware cost.

While it would be possible to design a graphics subsystem without considering other aspects of the system, the result would most likely be more expensive and slower than a system-oriented approach. Design and partitioning tradeoffs between GUI, 3D graphics, and decompression considerations let us place functionality where it can be provided most efficiently. In most cases, our graphics system design included performance margins to allow for the inevitable improvements in CPU speed.

Graphical user interface. Fast GUI performance is a good example of a system requirement involving hardware features in both the CPU and the graphics subsystem. GUIs use a number of low-level primitive routines that account for a majority of the time spent in typical user interactions. Accelerating these routines with a minimum of hardware to keep costs low presents the real problem.

Our criteria for including special features in the CPU were:[7]

- Is the proposed function best implemented in the CPU, or could the same function be implemented just as effectively in the graphics subsystem?
- Does the envisioned enhancement fit within the general CPU architecture in an economical fashion? (We never considered adding significant cost to the CPU.)
- Does the proposed enhancement provide a significant performance advantage?

One important operation in a GUI is passing data from main memory to the display for painting backgrounds or filling patterns. Graphics hardware cannot alone perform this operation: the CPU and memory system must also be involved. Our approach has the CPU/memory system providing a fast path from memory through the floating-point registers to the system bus (50 Mbytes/s) and the graphics hardware having adequate frame buffer write bandwidth (96 Mbytes/s) from the system bus.

The CPU also had to be able to send data quickly from CPU integer registers to graphics hardware. This sends low-level GUI primitives to the GUI accelerator. High bandwidth streams of CPU writes to I/O addresses have become standard in PA-RISC processors. Providing this capability involved reducing processor penalty cycles associated with I/O references and designing efficient mechanisms to transfer data between the CPU's connection to the memory and I/O controller and the system bus where the graphics controller resides.

Early investigations clearly showed that our cost constraints would not allow us to use hardware acceleration for all other GUI routines. Instead, we accelerated only those routines deemed most important: vectors, rectangles, screen-to-screen block moves, memory-to-screen block moves, text, cursor motion, and pixel formatting. Hardware support for vectors includes a vector drawing engine that can be loaded with a single word per connected polyline segment, while the CPU formats the word and performs the write-to-I/O space. The graphics hardware limit for vector drawing is over 2 million vectors/s.

Found in such areas as window backgrounds and boundaries, rectangle fill is another commonly used operation. Since VRAMs have a fast block mode to draw large, constant-color regions, we added hardware support for this function as well. Software specifies rectangles via a pair of writes. The hardware takes advantage of the four-column block mode to achieve a 425-million pixel/s peak rectangle fill rate with 2-Mbit VRAMs, and 850 million pixels/s peak with 4-Mbit VRAMs.

Since nearly all applications include text, user productivity demands fast text painting and scrolling. Artist lets the CPU provide just four words to define a 6×13 character, then optimizes the VRAM accesses to maximize text performance. An Artist chip can paint over one million characters/s.

Another aspect of fast text scrolling is the ability to move pixels quickly from one location on the display to another. We also use this capability to move entire windows on the display. Because moving pixels from one location to another would be inefficient if all the data had to go through the CPU, Artist includes hardware to handle this operation. With this support, Artist can achieve a block move rate of 47 million pixels/s within the frame buffer.

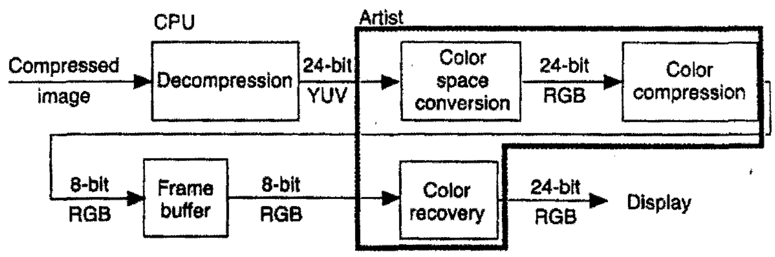


Figure 10. Image decompression pipeline. [Red-green-blue (RGB) and yellow-ultramarine-violet (YUV) are competing color schemes.]

To make all these features work seamlessly and efficiently for the GUI software drivers, we incorporated a number of addressing and data modes. These permit pixel accesses to be any of several pixel configurations (one, four, or 32 pixels per 32-bit word) with arbitrary frame buffer data alignment. Pixel replication can extend single-bit pixels to full depth; either ordered dithering or color compression can reduce 24-bit pixels to eight bits.

A hardware cursor maximizes GUI interactivity by allowing a cursor that does not affect the image bitmap. A hardware cursor can save many of the system CPU cycles spent on the GUI.

Three-dimensional graphics. Consistent with the system-design criteria used with GUI acceleration, Artist offers features to aid in the display of 3D data sets. These include a hardware vector rasterizer for accelerating wireframes and dithering and color compression for displaying 3D solids.

Fast memory-to-frame buffer writes help when double-buffering is required. Software can draw images to a virtual window in main memory, then quickly write them to the frame buffer when complete. The CPU's dual-integer ALUs help 3D graphics solids rendering into main memory in addition to aiding general-purpose processing. CPU floating-point enhancements for graphics, including fast clip checking and parallel multiplication and addition, allow very efficient vector vertex calculations.[7]

Multimedia. A new use of graphics hardware is for the display of images or image sequences that had previously been compressed. To make image retrieval interactive and real-time video sequencing possible, image restoration must be quick. Typically, this process includes variable-length decoding, inverse quantization, inverse discrete cosine transformation, and color-space conversion, as for example, in MPEG video decompression.

System-level design is especially helpful with digital image/video decompression. Since the CPU can do most of the full-motion digital video decoding with some instruction set tuning, dedicated hardware need not be added. Putting the last step of the decompression process (YUV-to-RGB color space conversion) in Artist further improves decompression performance. (See Figure 10.)

Figure 11. Dithering (top) versus color recovery (bottom).

Artist has circuitry to convert the color-space and to color compress the image into its 8-bit frame buffer. The colors are restored as part of the video refresh process; they are true color and appear to be 24-bits deep. The resultant images are much better that ones generated using dithering, a common technique (see Figure 11). The result provides real-time, decompressed, true-color video images on the display with only an entry-level, 8-bit hardware configuration.

Features. Bringing such advanced capabilities into widespread use requires a cost-effective solution. The graphics subsystem described here incorporates acceleration for GUIs, 3D graphics, and digital video with RAM control and video refresh in a single custom VLSI chip. When coupled with four or eight VRAMs (depending on the resolution of the display), this chip provides a complete workstation graphics hardware subsystem. (See Figure 12, next page, for its block diagram and Table 4 for its performance highlights.)

To minimize costs, we put the entire graphics system (except video RAM) on a single chip. Most graphics controller chips require external video clocks or digital-to-analog converters, but we included these in Artist to keep the parts count low. As Figure 13 shows, Artist consists of seven main blocks:

Bus interface/FIFO. Artist connects to the 32-bit multi-

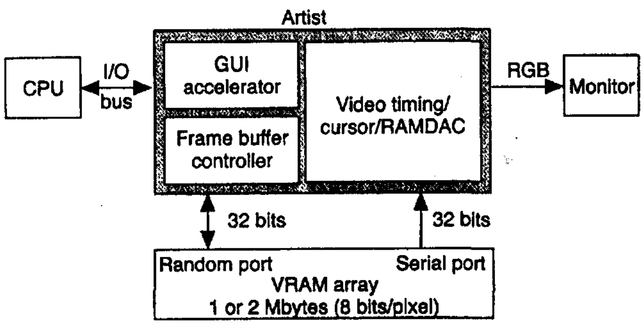


Figure 12. Graphics system block diagram.

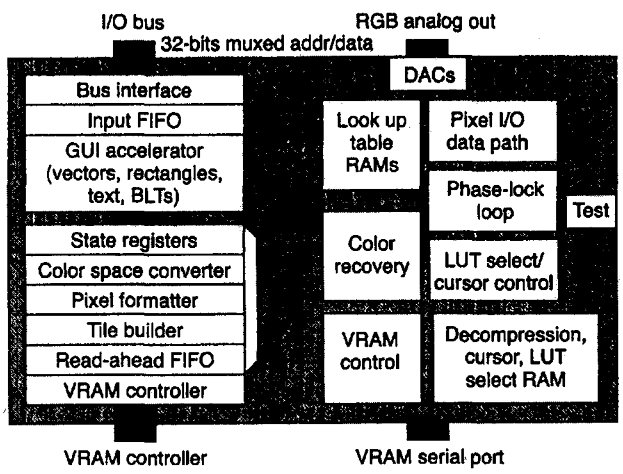


Figure 13. Artist chip block diagram.

plexed address/data system bus connecting the CPU, graphics, and I/O chip. Bus cycles run up to 40 MHz. Artist can accept either one or two data transfers per address cycle, making the peak available bandwidth over 100 Mbytes/s. A 32-deep first-in, first-out memory buffers transactions directed to various parts of the chip.

GUI accelerator. The GUI accelerator consists of an ALU connected to seven registers that manipulate display address and two registers that generate display data. Because these registers operate in a master/slave configuration, one operation can proceed while the next is being set up. Accelerated GUI functions include vector stepping, rectangle filling, text painting, pixel block moving, and lookup table writing.

Address/color formatter. At the output of the GUI accelerator is the address/color formatter that maps graphics data into the frame buffer. This process includes handling various pixel depths, plane masks, color spaces, and data alignments required by the software drivers. Contained in this block are the color converter, color compressor, dither unit, data barrel shifter, and lookup table and cursor data mapper.

Table 4. Artist performance.

Parameter	Value (per second)
Large rectangle fill (peak)	850M pixels
10-pixel, randomly oriented	2.1M vectors
10×10 rectangles	1.7M
Text characters (6×13 pixel)	1M
FB BitBit (unaligned)	47M pixels

Programmable VRAM controller. A VRAM controller at the output of the ACF accesses the random-access port of the VRAMS and initiates data-transfer cycles for updating the VRAM shift registers. Some of the timing parameters are programmable to maintain high levels of performance even when running at a slower clock frequency. Page mode cycles are 37.5 ns, with a clock frequency of 80 MHz. Making extensive use of block mode writes provides further performance optimization.

Video timing generator/PLL. A necessary part of any display controller is the video timing generator. The one built into Artist has programmable timing parameters, including the dot clock frequency itself. Artist reads in lookup-table select bits for each scan line during the horizontal blanking period prior to the display of that line. It supports a wide assortment of resolutions and refresh rates, from 640×480 pixels to 1,280×1,024 pixels with 72-Hz refresh.

Color recovery. Before the video refresh data reaches the lookup tables, it can pass through the color recovery unit. Whether the colors are recovered depends on whether the lookup-table selection bit matches the color recovery enable bit. The lookup tables make a small amount of correction to achieve the final image.

Lookup table/DACs. There are two lookup tables, each with a configuration of three 256 entries. Either all three RAMs can use a single 8-bit index in indexed (pseudocolor) mode or three separate indices can provide true-color decompression mode. Cursor data inserts into the video data stream after the lookup tables so it does not interfere with the images on the display. The DACs have a 75-ohm output so they can have a direct electrical connection to the display.

Chip details. All this circuitry fits on a die measuring 9.7×12 mm in a 208-pin package (See Figure 14). A digital flat-panel port requires a 240-pin package. Table 5 provides additional details about the Artist chip.

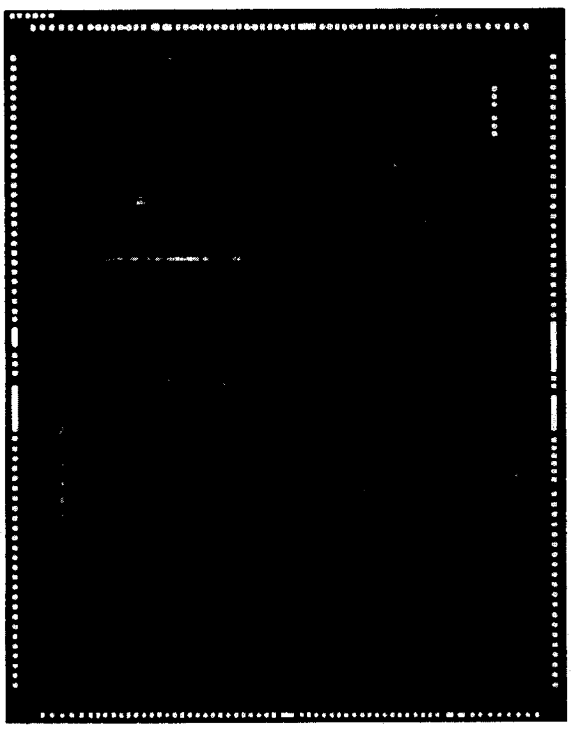


Figure 14. Artist die photograph.

Chip design methodology

Our low-cost goals drove several aspects of the chip designs for the initial systems. The cost of a high performance package becomes a significant portion of the delivered part cost. Reducing power dissipation was also a key consideration during the initial design phase.

The leveraged CPU design made heavy use of local two-phase nonoverlapping clock generators. We migrated this design to one that included a qualifier for each local clock phase, for idling circuits when not active, an especially important consideration for global bus drivers. Thus, for example, Hummingbird does not update or read registers unnecessarily. It uses a custom design approach in large, regularly structured blocks such as RAM, DACs, and most of the data path.

We also designed a special 432-pin ceramic pin-grid array for Hummingbird. Our power reduction strategies enabled us to use a package without bypass capacitors, reducing package and assembly costs. Our Artist packaging strategy involved using commonly available, inexpensive packaging.

For both chips, we synthesized control blocks from both behavioral descriptions and programmable logic array-style equations. We used a three-layer-over-the-top router for composing the artwork for these blocks, a departure from our previous PLA-style designs which markedly improved area efficiency. We used timing analysis and circuit simulation to find paths that needed optimization or custom circuits.

Table 5. Artist details.

Parameter	Value
Transistors	525,000
Die size	9.7x12.1 mm
Metal layers	3
L_{eff}	0.61 µm (NFETs) 0.66 µm (PFETs)
Frequency	40-80 MHz (control) 25-135 MHz (video)
Power	3.5 W (worst case)
Package	208-pin QFP/240-pin QFP
Supplies	5V, 3.3V

We incorporated an aggressive diagnostic capability into Hummingbird that involved piggybacking internal signals onto the system bus during its idle states. By presetting a signal group before running a test, the user can dump all the critical signals, including instruction and data addresses, instructions, and bundling information, virtual translation information, memory and I/O transaction information, and more. These diagnostic signals are driven transparently through the pin driver from their sources at twice Hummingbird's internal frequency.

Artist contains signature generators in several key positions to isolate failures to a single component. A signature in the bus interface verifies proper operation to the graphics system. Signatures on the VRAM random access and serial ports separate VRAM from Artist failures. A signature taken at the input to the DACs helps identify faults in the Artist video section. A crude ADC on the analog video port can identify major errors in the DAC output. Hummingbird also includes a signature generator to accelerate manufacturing tests of the internal instruction cache.

We also included IEEE 1149.1 compliance to help lower board test manufacturing cost. In addition, for Hummingbird, we merged our previous serial test methodology to allow sampling of all scanable nodes on the processor on a specific clock cycle and scanning the sampled values out of the chip while the system continues to run. This greatly aids diagnosis of failures on prototype systems.

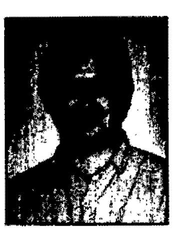
WITH THE PA7100LC VLSI CHIP SET, Hewlett-Packard has pursued a path of high system integration to maximize both cost effectiveness and raw processing power. While performance continues to be an important factor, other design goals such as low cost and low power came into play for this particular design. By performing system wide optimization, we both improved performance and lowered costs as we integrated an entire workstation system into three VLSI chips.

Acknowledgments

We only have space here to thank those who were directly involved with this article: Charlie Kohlhardt, Mark Forsyth, Craig Gleason, Doug Josephson, Joel Lamb, Tom Meyer, Leith Johnson, Pat Knebel, Paul Martin, Tony Barkans, John Wheeler, and Ruby Lee.

References

1. P. Knebel et al., "HP's PA7100LC: A Low-Cost Superscalar PA-RISC Processor," *Compcon Digest of Papers*, Feb. 1993, pp. 441-447.
2. T. Asprey et al., "Performance Features of the PA7100 Microprocessor," *IEEE Micro*, Vol. 13, No. 3, June 1993, pp. 22-35.
3. E. DeLano et al., "A High Speed Superscalar PA-RISC Processor," *Compcon Digest of Papers*, Feb. 1992, pp. 116-121.
4. M. Forsyth et al., "CMOS PA-RISC Processor for a New Family of Workstations," *Compcon Digest of Papers*, Feb. 1991, pp. 202-207.
5. D. Tanksalvala et al., "A 90-MHz CMOS RISC CPU Designed for Sustained Performance," *Digest of Tech. Papers, IEEE Solid-State Circuits Conf.*, Vol. 33, 1990, pp. 52-53.
6. R. Lee, "Precision Architecture," *Computer*, Vol. 22, No. 1, Jan. 1989, pp. 78-91.
7. C. Casey and L. Thayer, "Scalable Graphics Enhancements for PA-RISC Workstations," *Compcon Digest of Papers*, Feb. 1992, pp. 122-128.



Steve Undy is an engineer/scientist for Hewlett-Packard in Fort Collins, Colorado. He has contributed to the design and verification of six PA-RISC processors and was a key designer of the cache system on the Hummingbird processor. He presented the Hummingbird design at Hot Chips V.

Undy received a BS in electrical engineering and a BS in computer engineering from the University of Michigan and an MS in electrical engineering from Purdue University. He is a member of the IEEE Computer Society.

Mick Bass works at the Fort Collins site as a member of technical staff. He has contributed to CPU, memory controller, and I/O controller designs used in PA-RISC workstations. He has also worked on chip and system verification.

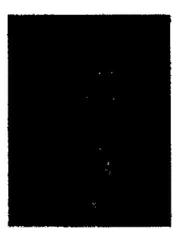
Bass received a BS degree in computer engineering from the University of Illinois at Urbana/Champaign. He is a member of the IEEE.

Dave Hollenbeck is a member of the technical staff at Fort Collins. He has been involved with CPU, memory controller, and system I/O chip designs for PA-RISC systems.

Hollenbeck earned an MSEE and BSEE from the University of South Florida.

Wayne Kever is a member of the technical staff at Fort Collins. He has been involved with the design of four PA-RISC CPU chip sets. His interests include high-speed circuit design, cost/performance trade-offs, and low-power design.

Kever received the BS degree from the University of Oklahoma and the MS degree from Stanford University, both in electrical engineering. He is a member of the IEEE and the IEEE Computer Society.



Larry Thayer is an engineer/scientist at Fort Collins where he has been involved with a number of chip designs for workstation graphics systems. He has published articles and papers in *Byte*, *IEEE Spectrum*, and the proceedings of ACM Siggraph and IEEE Compcon, among others.

Thayer received BS and MS degrees in electrical engineering from Ohio State University.

Direct any questions concerning this article to Steve Undy, Hewlett-Packard, 3404 E. Harmony Rd., Fort Collins, CO 80525; sru@fc.hp.com.

Reader Interest Survey

Indicate your interest in this article by circling the appropriate number on the Reader Service Card.

Low 153 Medium 154 High 155