Figure 2A:

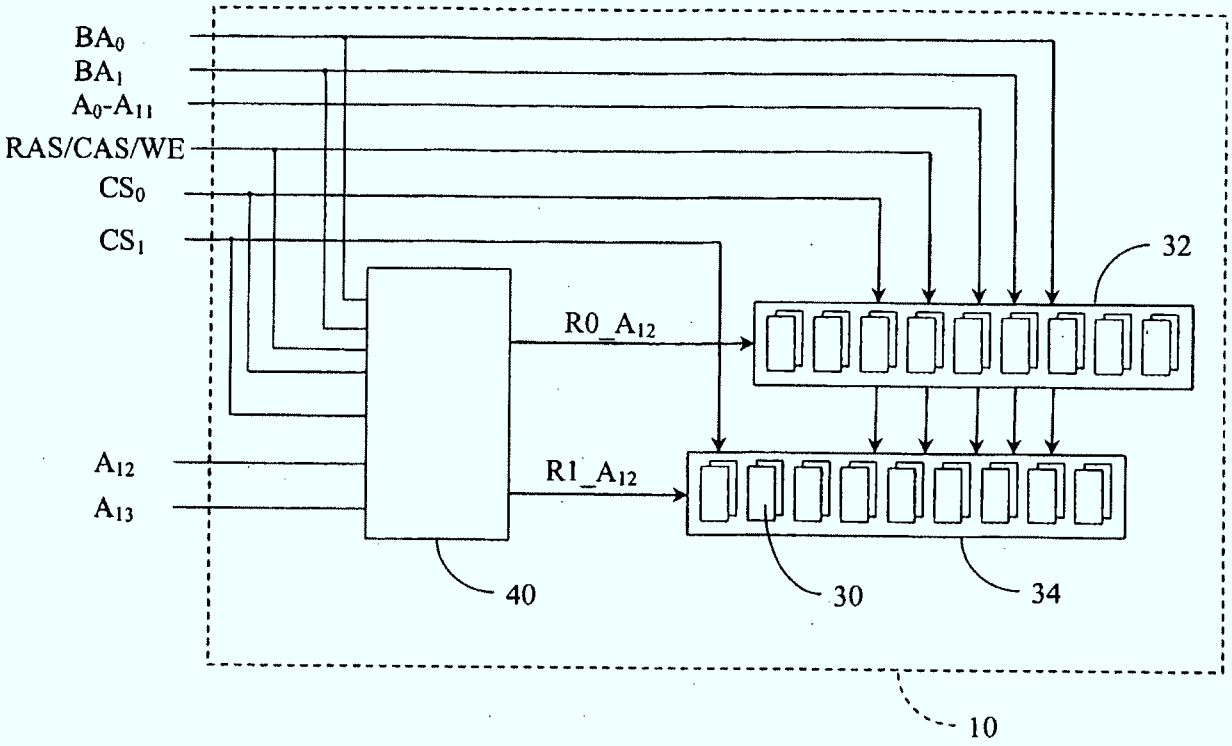


Figure 2B:

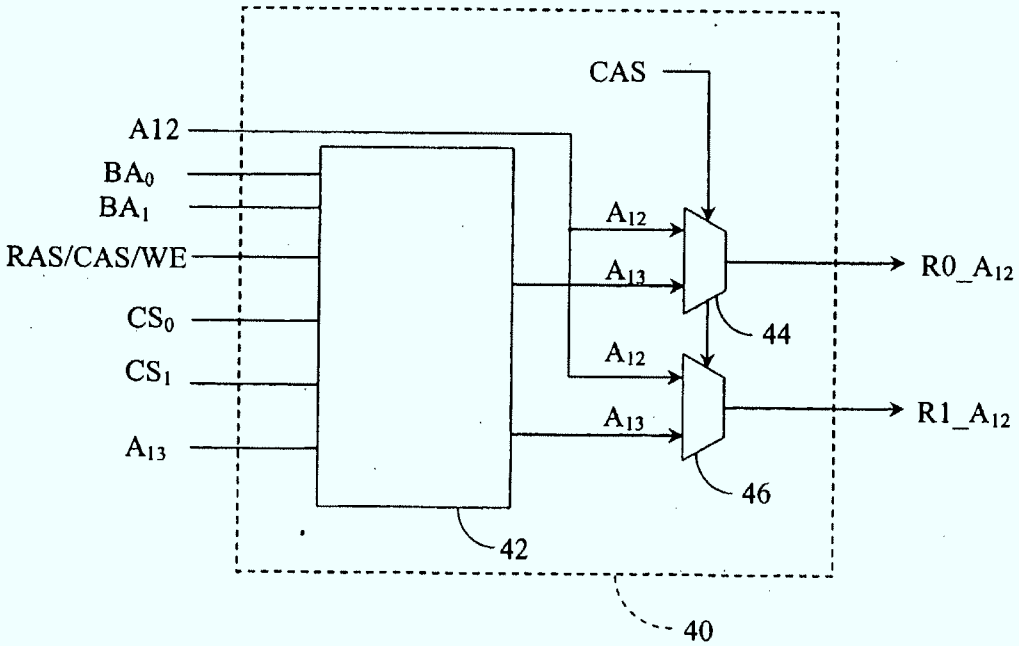


Figure 3A:

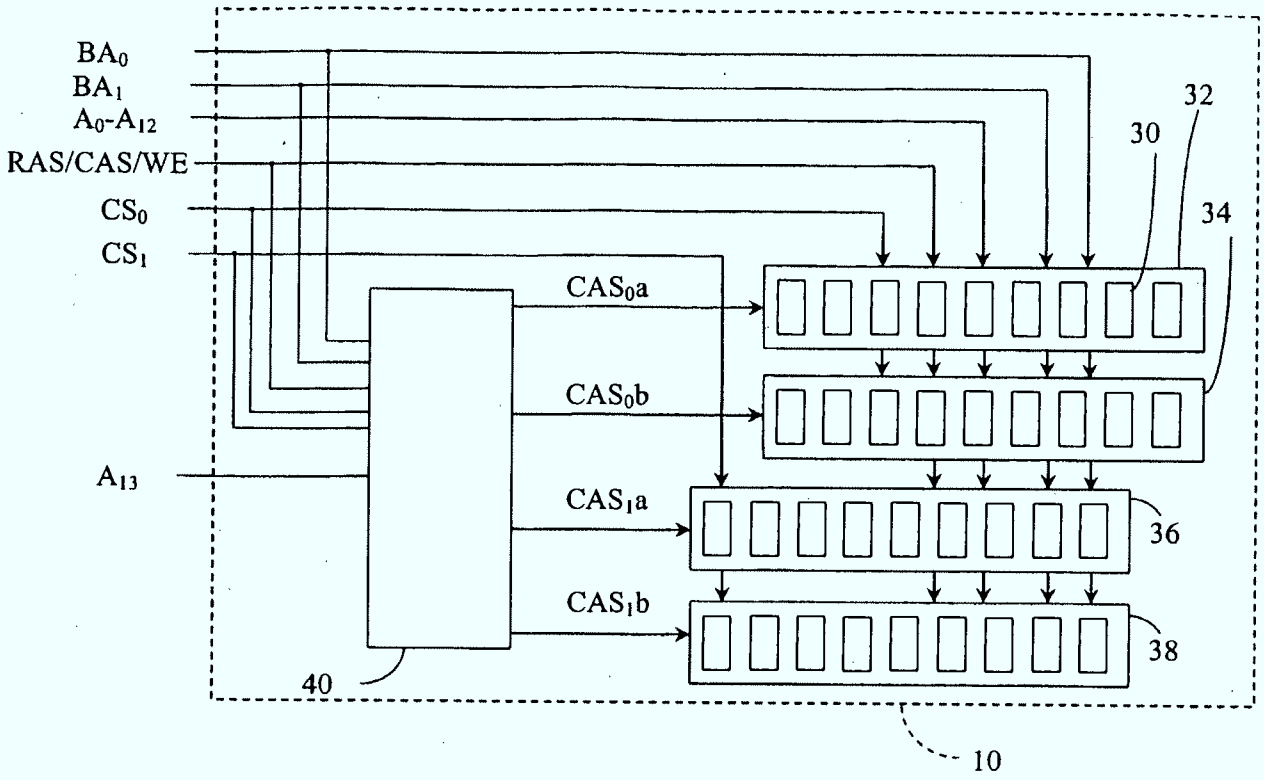


Figure 3B:

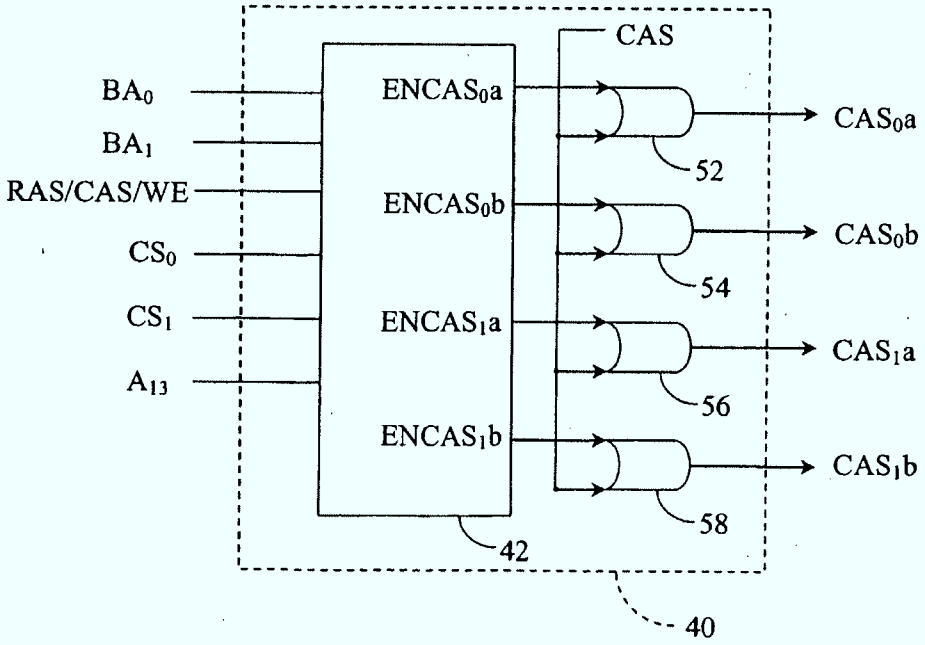


Figure 4A:

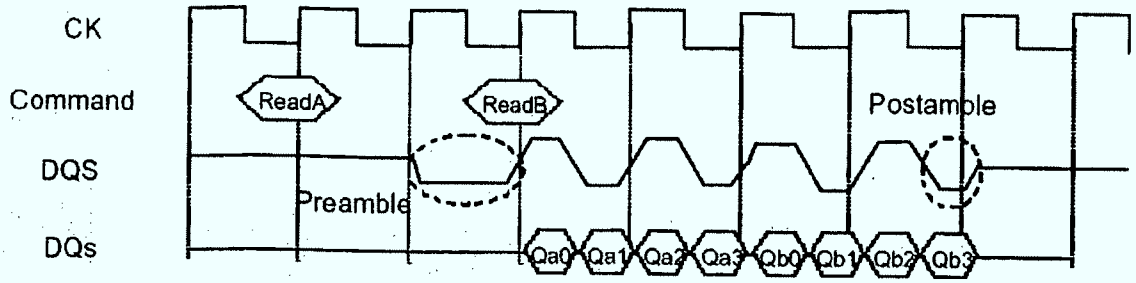


Figure 4B:

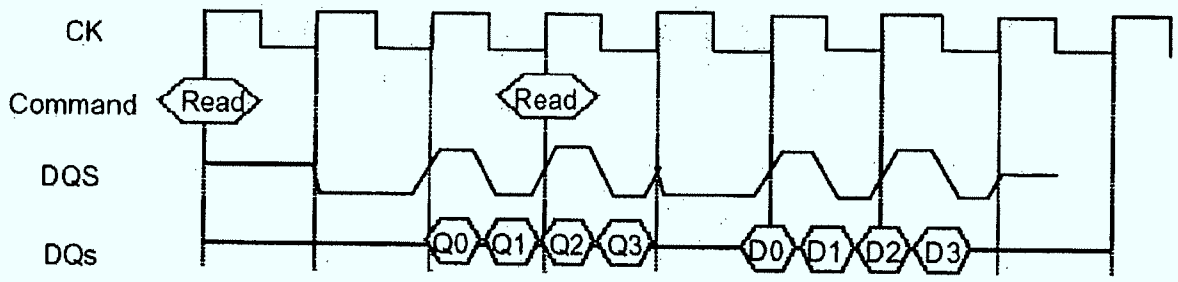


Figure 5:

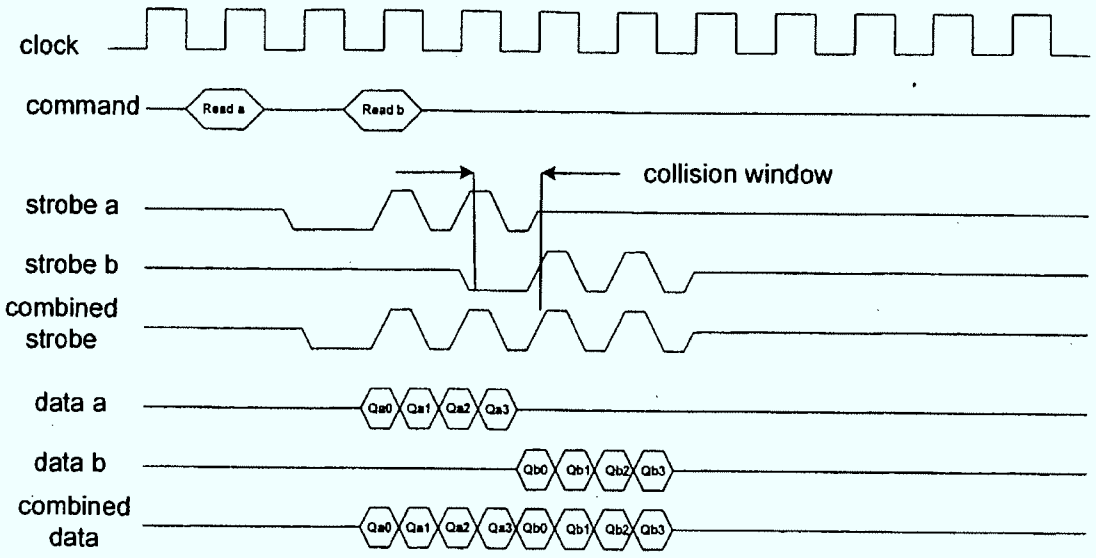

clock
command
Read a
Read b
collision window
strobe a
strobe b
combined strobe
data a
Qa0
Qa1
Qa2
Qa3
data b
Qb0
Qb1
Qb2
Qb3
combined data
Qa0
Qa1
Qa2
Qa3
Qb0
Qb1
Qb2
Qb3

Figure 6A:

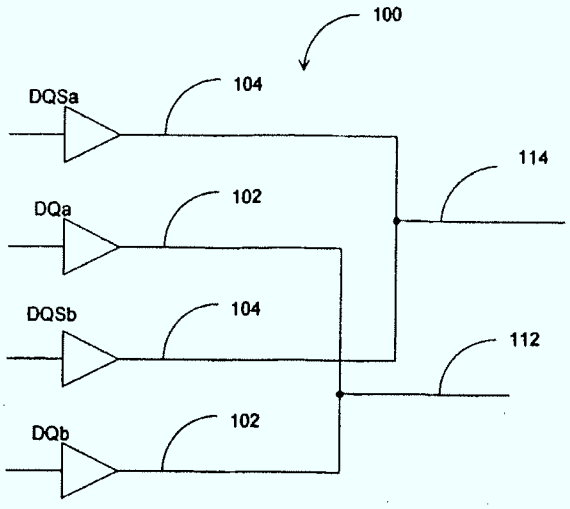


Figure 6B:

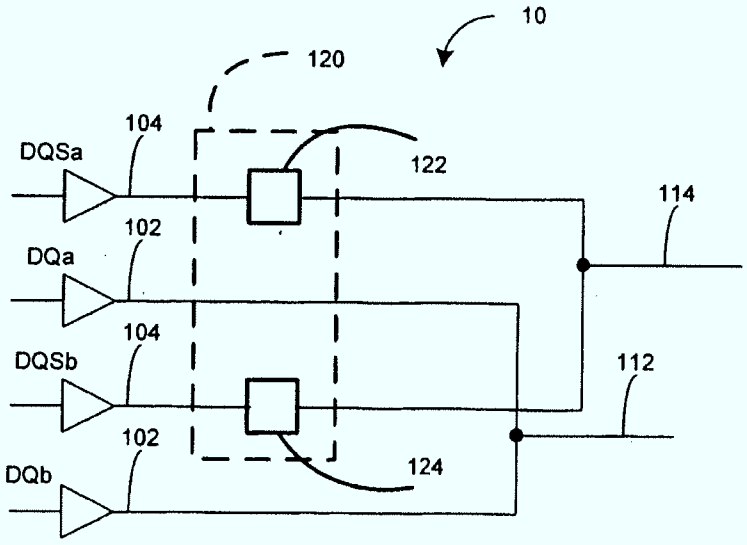


Figure 6C:

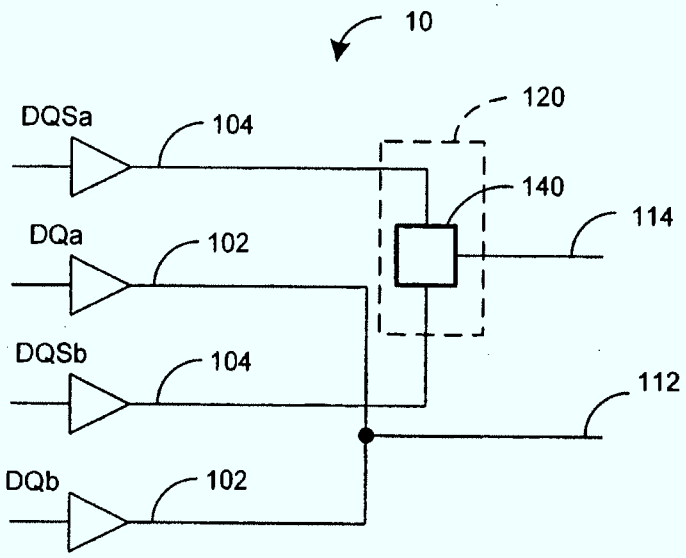


Figure 6D:

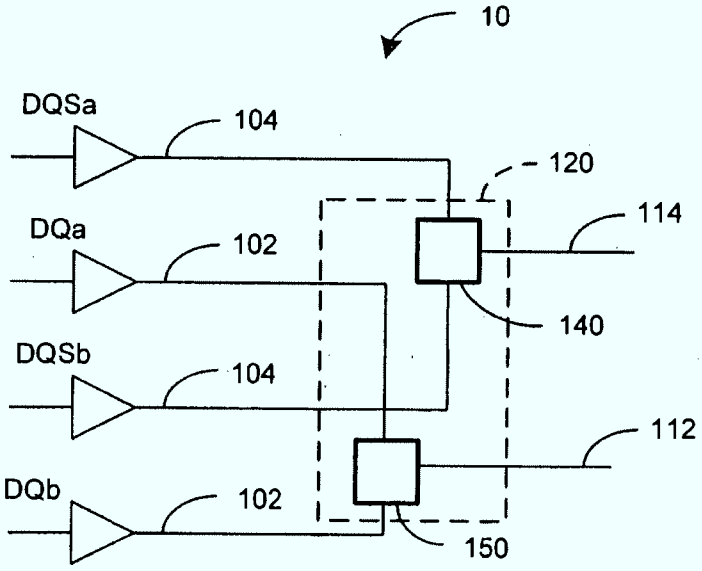


Figure 6E:

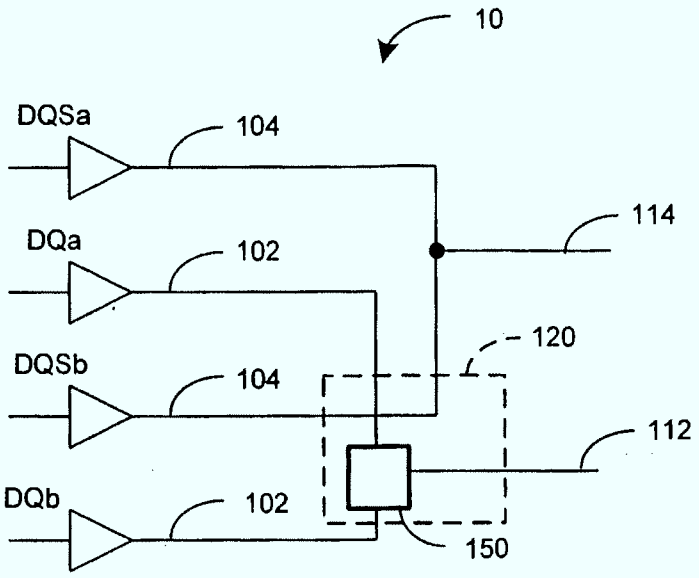


Figure 7:

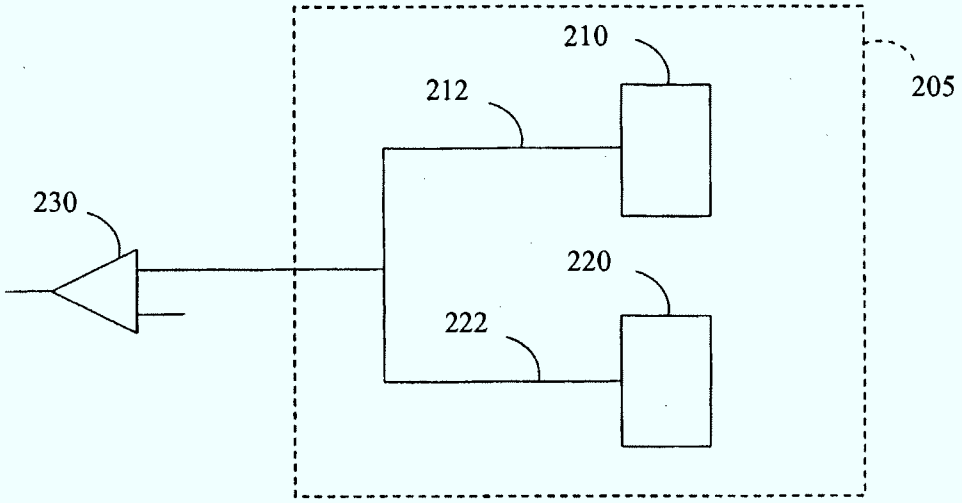


Figure 8:

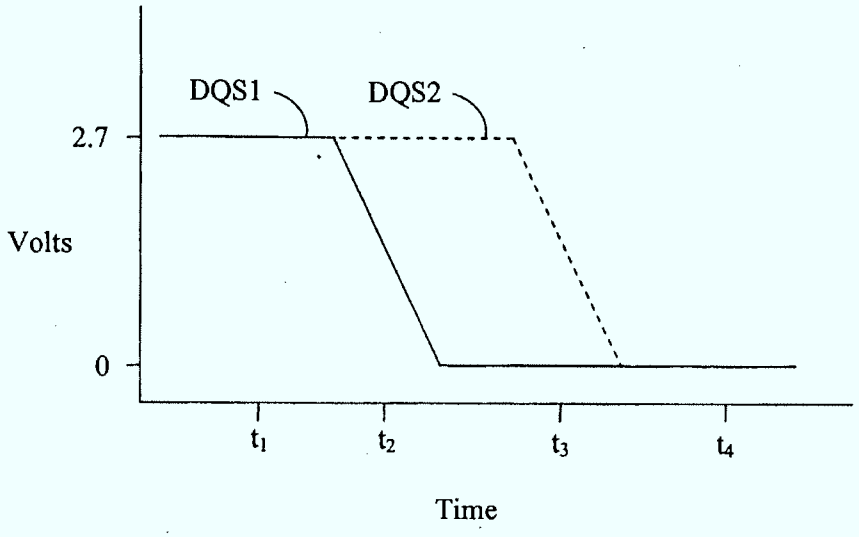


Figure 9:

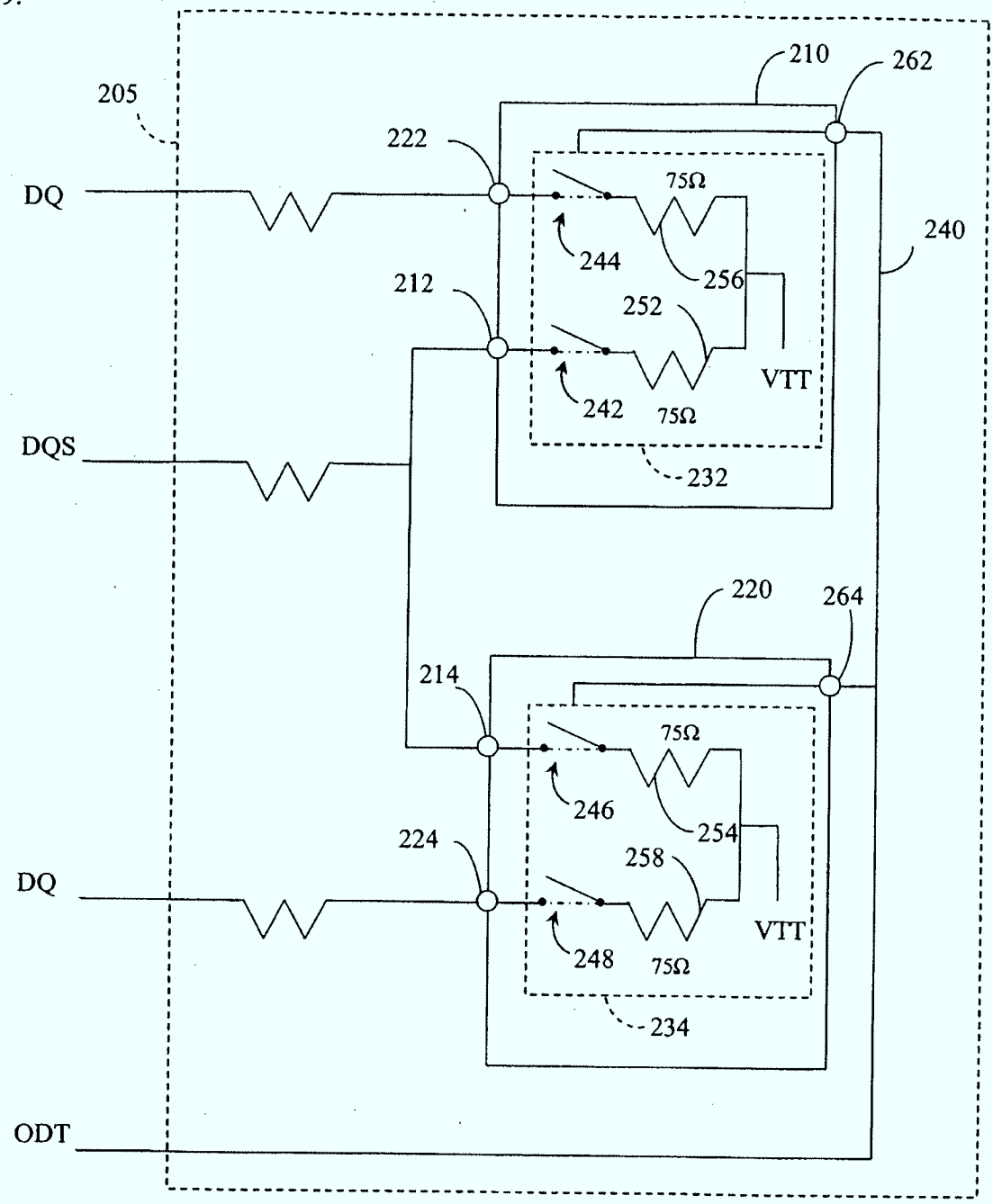

205
210
262
222
DQ
75Ω
240
244
256
212
252
VTT
242
75Ω
DQS
232
220
264
214
75Ω
246
254
224
258
DQ
VTT
248
75Ω
234
ODT

Figure 10:

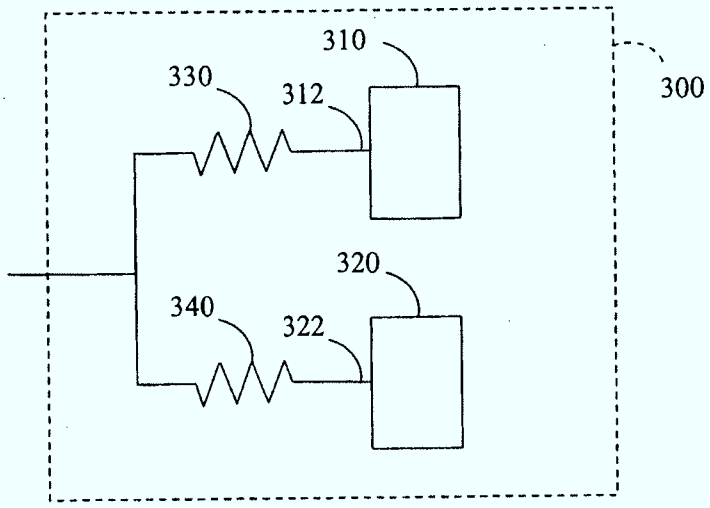


Figure 11A:

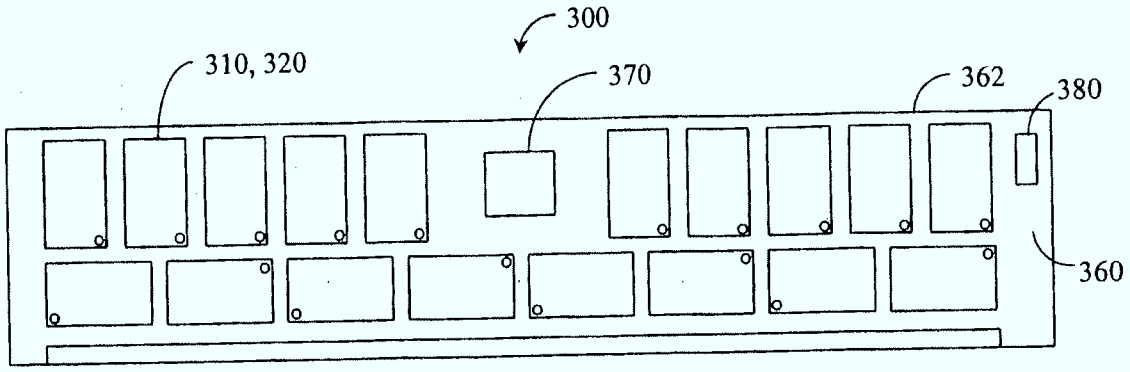


Figure 11B:

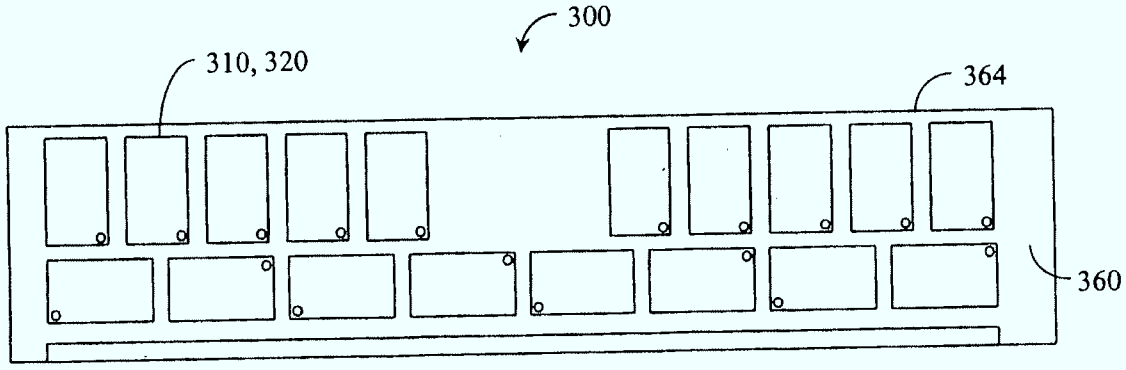


Figure 12A:

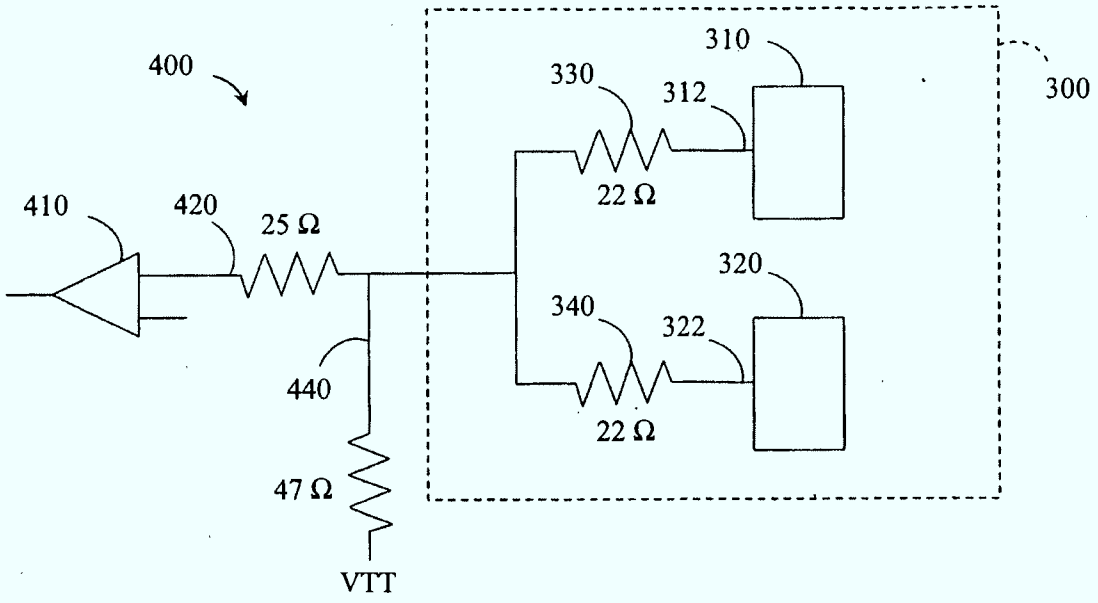


Figure 12B:

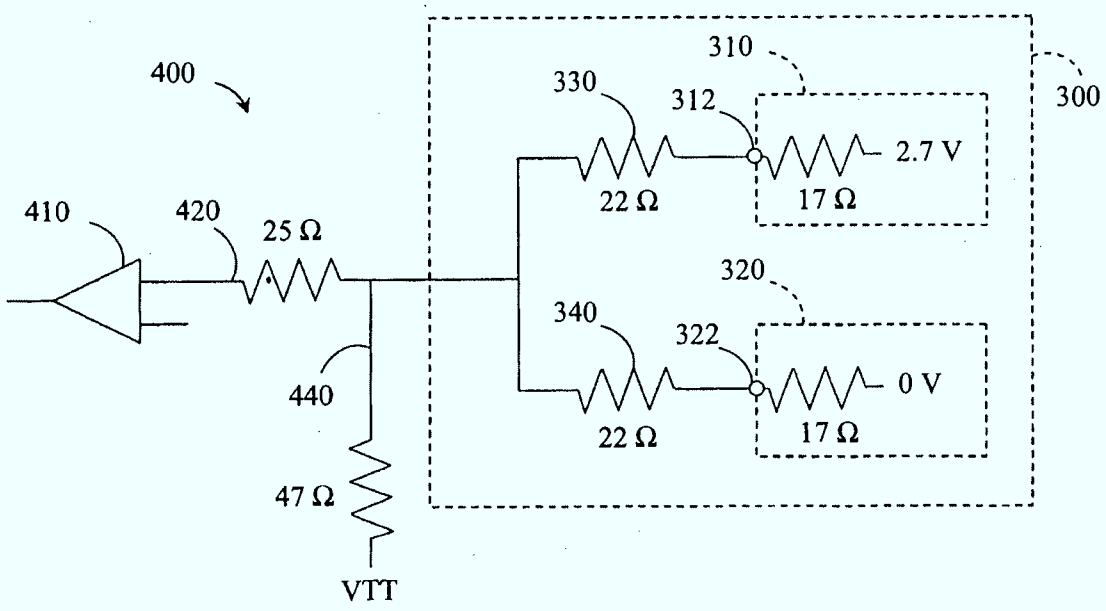


Figure 13:

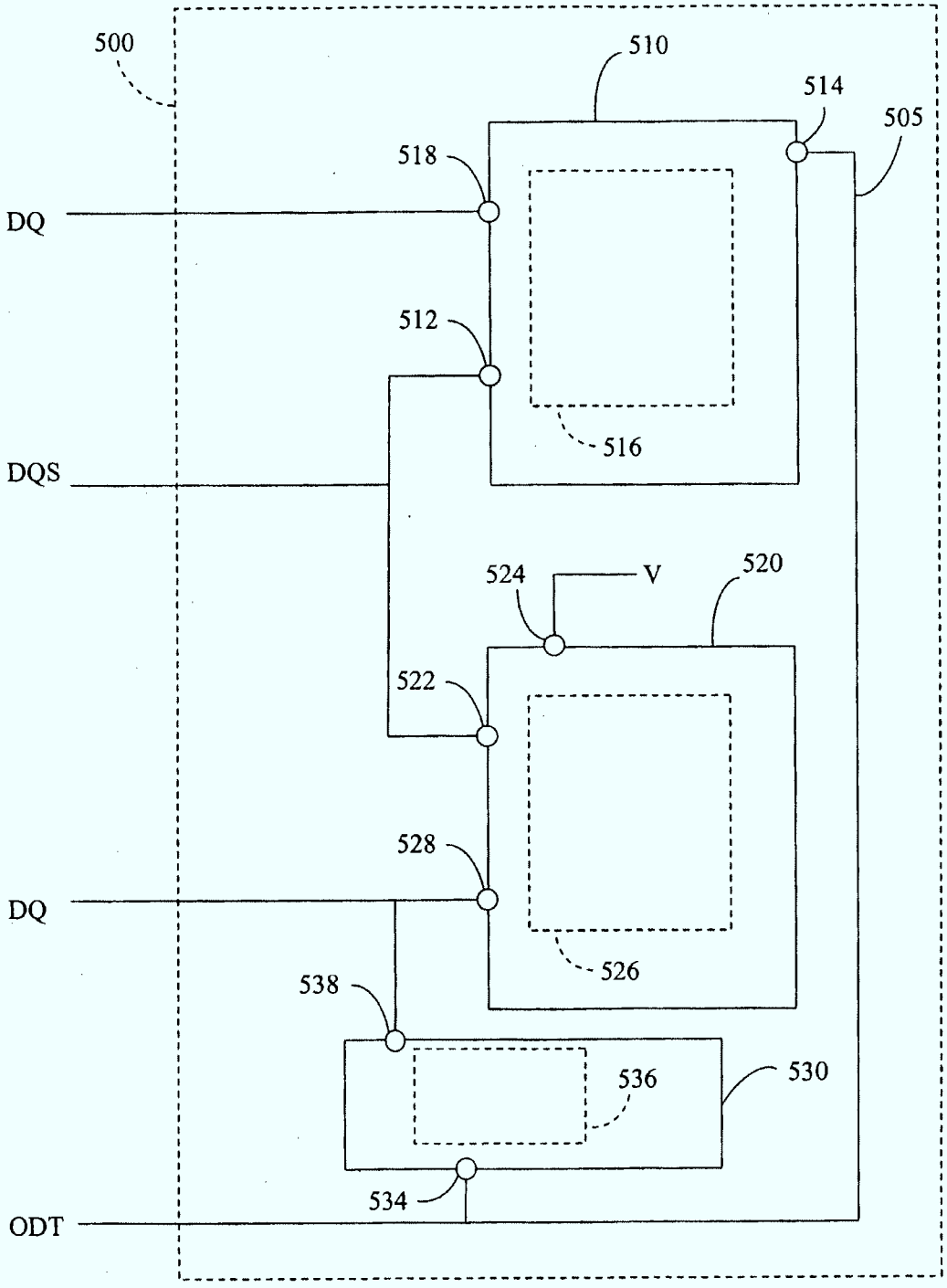
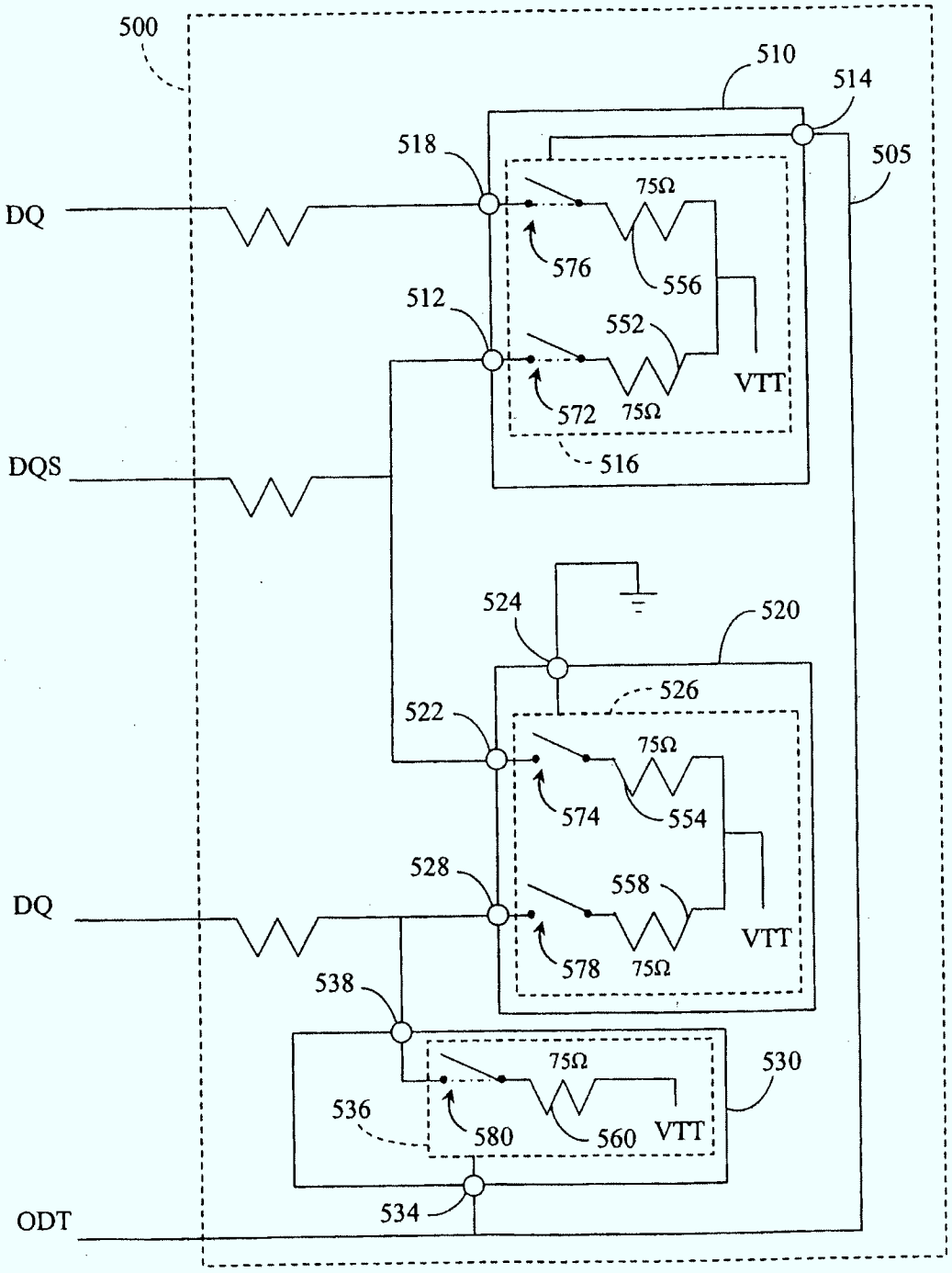

500
510
514
505
518
DQ
512
516
DQS
524
V
520
522
528
DQ
526
538
536
530
534
ODT

Figure 14:


500
510
514
505
518
75Ω
DQ
576
556
512
552
VTT
572
75Ω
516
DQS
524
520
526
522
75Ω
574
554
528
558
DQ
VTT
578
75Ω
538
530
75Ω
536
580
560
VTT
534
ODT

PTO/SB/06 (12-04)
Approved for use through 7/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
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.

PATENT APPLICATION FEE DETERMINATION RECORD

Substitute for Form PTO-875

Application or Docket Number: 11173175

APPLICATION AS FILED – PART I

FOR	(Column 1) NUMBER FILED	(Column 2) NUMBER EXTRA	SMALL ENTITY RATE ($)	SMALL ENTITY FEE ($)	OR	OTHER THAN SMALL ENTITY RATE ($)	OTHER THAN SMALL ENTITY FEE ($)
BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A	N/A			N/A	300
SEARCH FEE (37 CFR 1.16(k), (l), or (m))	N/A	N/A	N/A			N/A	
EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A	N/A			N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	20 minus 20 =	* 0	X =		OR	X =	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	4 minus 3 =	* 1	X =		OR	X =	
APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).						
MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))			N/A			N/A	
* If the difference in column 1 is less than zero, enter "0" in column 2.			TOTAL			TOTAL	

APPLICATION AS AMENDED – PART II

AMENDMENT A

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	SMALL ENTITY RATE ($)	SMALL ENTITY ADDITIONAL FEE ($)	OR	OTHER THAN SMALL ENTITY RATE ($)	OTHER THAN SMALL ENTITY ADDITIONAL FEE ($)
Total (37 CFR 1.16(i))	*	Minus	**	=	X =		OR	X =	
Independent (37 CFR 1.16(h))	*	Minus	***	=	X =		OR	X =	
Application Size Fee (37 CFR 1.16(s))									
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))					N/A		OR	N/A	
					TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

AMENDMENT B

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA	RATE ($)	ADDITIONAL FEE ($)	OR	RATE ($)	ADDITIONAL FEE ($)
Total (37 CFR 1.16(i))	*	Minus	**	=	X =		OR	X =	
Independent (37 CFR 1.16(h))	*	Minus	***	=	X =		OR	X =	
Application Size Fee (37 CFR 1.16(s))									
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))					N/A		OR	N/A	
					TOTAL ADD'L FEE		OR	TOTAL ADD'L 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.

APPENDIX C

US006209074B1

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

(10) **Patent No.: US 6,209,074 B1**
(45) **Date of Patent: *Mar. 27, 2001**

(54) **ADDRESS RE-MAPPING FOR MEMORY MODULE USING PRESENCE DETECT DATA**

(75) Inventors: **Timothy Jay Dell**, Colchester; **Mark William Kellogg**, Essex Junction, both of VT (US)

(73) Assignee: **International Business Machines Corporation**, Armonk, NY (US)

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

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **09/067,549**

(22) Filed: **Apr. 28, 1998**

(51) **Int. Cl.**[7] **G06F 12/00**

(52) **U.S. Cl.** **711/170**; 711/5; 711/202; 365/230.04

(58) **Field of Search** 711/5, 170, 202, 711/171, 172; 365/230.04

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,307,320	*	4/1994	Farrer et al.	365/230.01
5,379,304		1/1995	Dell et al.	371/40.1
5,390,308	*	2/1995	Ware et al.	711/5
5,412,788	*	5/1995	Collins et al.	711/157
5,450,422		9/1995	Dell	371/40.1
5,745,914		4/1998	Connolly et al.	711/105
5,765,188	*	6/1998	Cowell	711/115
5,897,663	*	4/1999	Stancil	711/200

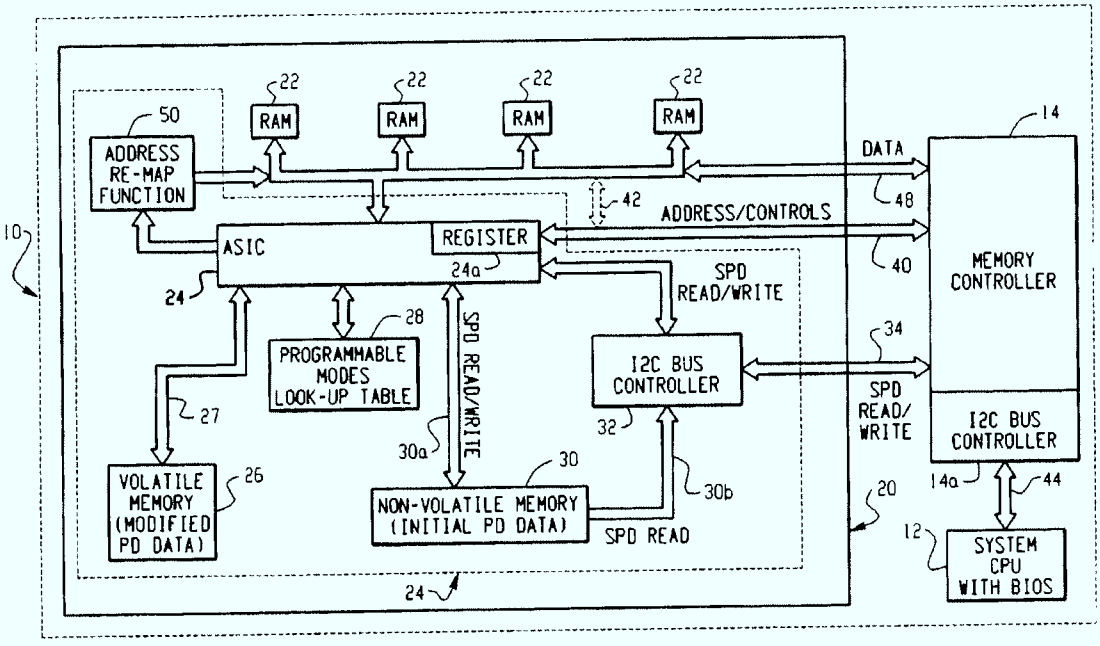
* cited by examiner

Primary Examiner—Do Hyun Yoo
Assistant Examiner—Mehdi Namazi
(74) *Attorney, Agent, or Firm*—William N. Hogg

(57) **ABSTRACT**

A memory module comprising: a plurality of memory devices associated with the module; each of said memory devices being configured in M banks; and a logic circuit for configuring the memory module to operate in a programmable addressing mode; said logic circuit receiving a number of address inputs and a number of bank address signals from a memory controller with said address inputs and bank address input signals corresponding to N bank memory devices; said logic circuit re-mapping at least one of said address inputs as an additional bank address signal to the memory device.

25 Claims, 3 Drawing Sheets



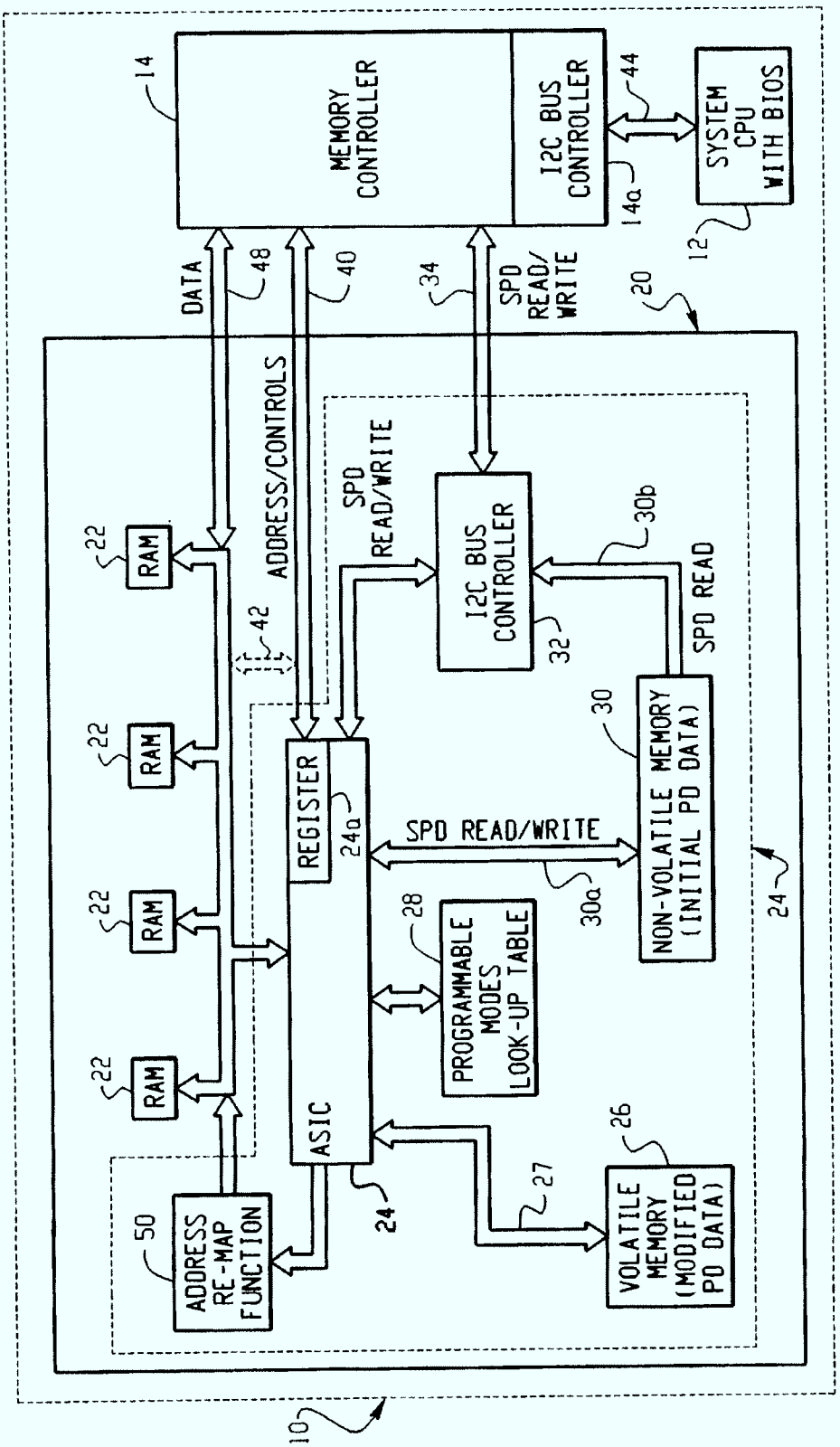

14
MEMORY CONTROLLER
I2C BUS CONTROLLER
44
14a
SYSTEM CPU WITH BIOS
12
DATA
48
40
34
SPD READ/ WRITE
20
ADDRESS/CONTROLS
SPD READ/WRITE
I2C BUS CONTROLLER
30b
32
SPD READ
22
RAM
42
REGISTER
24a
SPD READ/WRITE
30
NON-VOLATILE MEMORY (INITIAL PD DATA)
30a
24
28
PROGRAMMABLE MODES LOOK-UP TABLE
ASIC
26
27
VOLATILE MEMORY (MODIFIED PD DATA)
50
ADDRESS RE-MAP FUNCTION
10

Fig. 1

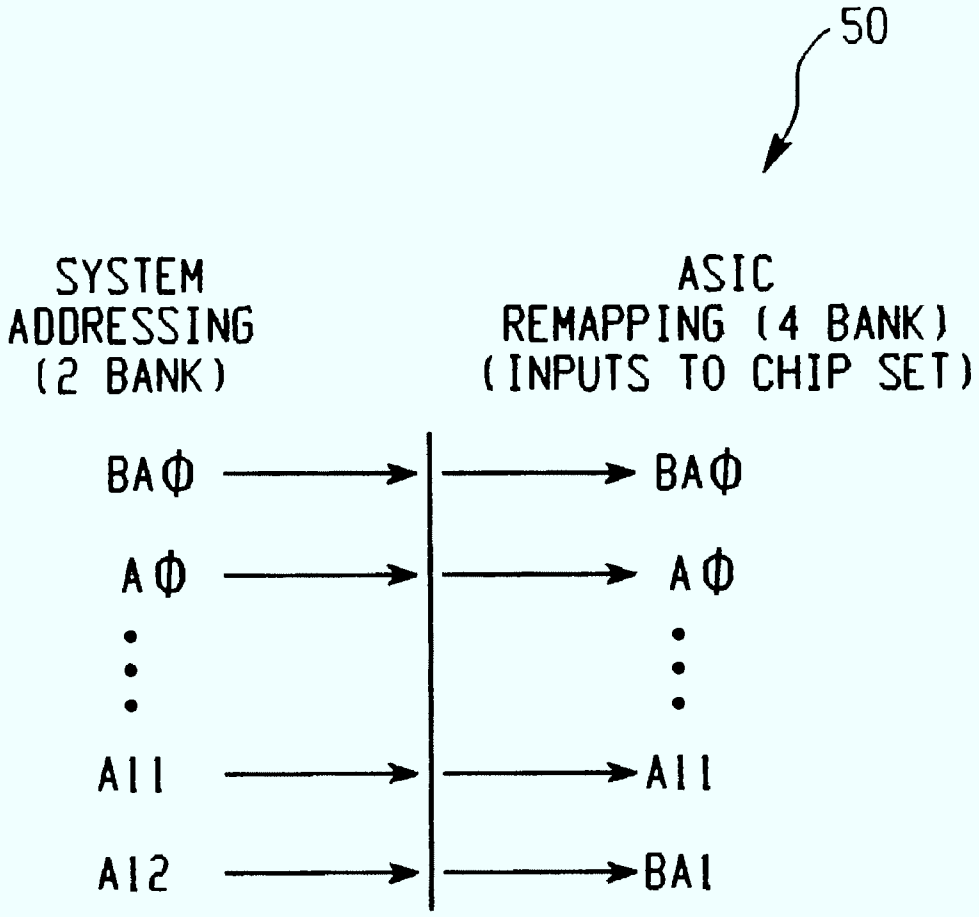

50
SYSTEM
ADDRESSING
(2 BANK)
ASIC
REMAPPING (4 BANK)
(INPUTS TO CHIP SET)
BAΦ
AΦ
A11
A12
BAΦ
AΦ
A11
BA1

Fig. 1A

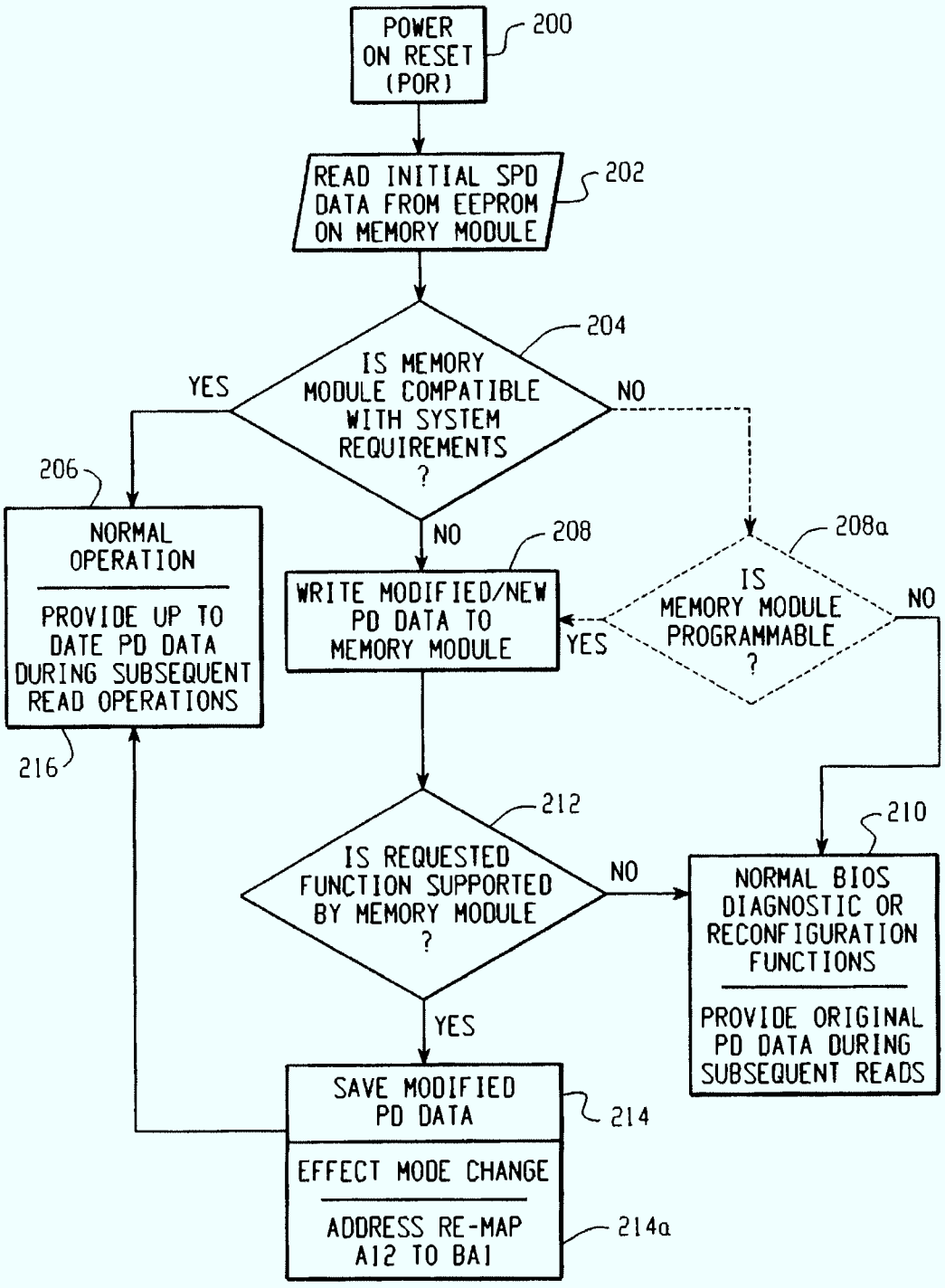

POWER ON RESET (POR)
200
READ INITIAL SPD DATA FROM EEPROM ON MEMORY MODULE
202
IS MEMORY MODULE COMPATIBLE WITH SYSTEM REQUIREMENTS ?
204
YES
NO
206
NORMAL OPERATION
PROVIDE UP TO DATE PD DATA DURING SUBSEQUENT READ OPERATIONS
216
NO
208
WRITE MODIFIED/NEW PD DATA TO MEMORY MODULE
208a
IS MEMORY MODULE PROGRAMMABLE ?
YES
NO
212
IS REQUESTED FUNCTION SUPPORTED BY MEMORY MODULE ?
NO
YES
210
NORMAL BIOS DIAGNOSTIC OR RECONFIGURATION FUNCTIONS
PROVIDE ORIGINAL PD DATA DURING SUBSEQUENT READS
SAVE MODIFIED PD DATA
214
EFFECT MODE CHANGE
ADDRESS RE-MAP A12 TO BA1
214a

Fig. 2

ADDRESS RE-MAPPING FOR MEMORY MODULE USING PRESENCE DETECT DATA

RELATED APPLICATIONS

U.S. application Ser. No. 09/067,420, entitled "Dynamic Configuration of Memory Module Using Presence Detect Data", filed Apr. 28, 1998; U.S. application Ser. No. 08/598,857, now U.S. Pat. No. 5,926,827, entitled "High Density SIMM or DIMM with RAS Address Re-Mapping", filed Feb. 9, 1996; and U.S. application Ser. No. 08/582,080, now U.S. Pat. No. 5,838,122, entitled "Method and Apparatus for Modifying Signals Received by Memory Cards", filed Jan. 2, 1996.

FIELD OF THE INVENTION

The invention relates generally to memory modules for computer systems. More particularly, the invention relates to address re-mapping techniques such as, for example, for system level negotiation of an addressing mode of a memory module by dynamic control of the presence detect data.

BACKGROUND OF THE INVENTION

Computer memory comes in two basic forms: Random Access Memory (hereinafter RAM) and Read-Only Memory (hereinafter ROM). RAM is generally used by a processor for reading and writing data. RAM memory is volatile typically, meaning that the data stored in the memory is lost when power is removed. ROM is generally used for storing data which will never change, such as the Basic Input/Output System (hereinafter BIOS). ROM memory is non-volatile typically, meaning that the data stored in the memory is not lost even if power is removed from the memory.

Generally, RAM makes up the bulk of the computer system's memory, excluding the computer system's hard-drive, if one exists. RAM typically comes in the form of dynamic RAM (hereinafter DRAM) which requires frequent recharging or refreshing to preserve its contents. Organizationally, RAM data is typically arranged in bytes of 8 data bits. An optional 9th bit, a parity bit, acts as a check on the correctness of the values of the other eight bits.

As computer systems become more advanced, there is an ever increasing demand for DRAM memory capacity.

Consequently, DRAM memory is available in module form, in which a plurality of memory chips are placed on a small circuit card, which card then plugs into a memory socket connected to the computer motherboard or memory carrier card. Examples of commercial memory modules are SIMMs (Single In-line Memory Modules) and DIMMs (Dual In-line Memory Modules).

In addition to an ever increasing demand for DRAM capacity, different computer systems may also require different memory operating modes. Present memories are designed with different modes and operational features such as fast page mode (FPM), extended data out (EDO), synchronous DRAM (SDRAM), double data rate SDRAM (DDR SDRAM), parity and non-parity, error correcting (ECC) and non error correcting, to name a few. Memories also are produced with a variety of performance characteristics such as access speeds, refresh times and so on. Further still, a wide variety of basic memory architectures are available with different device organizations, addressing requirements and logical banks. As a result, some memory modules may or may not have features that are compatible with a particular computer system.

In order to address some of the problems associated with the wide variety of memory chip performance, operational characteristics and compatibility with system requirements, memory modules are being provided with presence detect (PD) data. PD data is stored in a non-volatile memory such as an EEPROM on the memory module. A typical PD data structure includes 256 eight bit bytes of information. Bytes **0** through **127** are generally locked by the manufacturer, while bytes **128** through **255** are available for system use. Bytes **0–35** are intended to provide an indepth summary of the memory module architecture, allowable functions and important timing information. PD data can be read in parallel or series form, but serial PD (SPD) is already commonly in use. SPD data is serially accessed by the system memory controller during boot up across a standard serial bus such as an I²C™ bus (hereinafter referred to as an I2C bus). The system controller then determines whether the memory module is compatible with the system requirements and if it is will complete a normal boot. If the module is not compatible an error message may be issued or other action taken.

Some memory devices have the memory cells organized into a number of logical banks which can be individually addressed by the system memory controller. Control of bank selection is accomplished through the use of one or more Bank Address (BA) inputs. If the memory device bank organization is not the same as the system level addressing scheme, the memory module may be incompatible with the system requirements. For example, the system may need a two bank memory chip, but the memory module may include a memory device that is a four bank device.

It is desired, therefore, to provide a memory module that is more flexible in terms of its compatibility with different computer systems, and particularly that permits the computer system dynamically to negotiate available memory module functions and modes, especially an address re-mapping function.

SUMMARY OF THE INVENTION

The present invention contemplates, in one embodiment, a memory module that includes a plurality of memory devices associated with the module; each of said memory devices being configured as M banks; and a logic circuit for configuring the memory module to operate in a programmable addressing mode; said logic circuit receiving a number of address inputs and a number of bank address signals from a memory controller with said address inputs and bank address input signals corresponding to N bank memory devices; said logic circuit re-mapping at least one of said address inputs as an additional bank address signal to the memory device.

The invention also contemplates the methods embodied in the use of such a memory module, and in another embodiment, a method for using an M bank memory device in a computer system that has N bank addressing, comprising the steps of:

a) inputting address signals from a system controller to a logic circuit, said address signals including a number of address inputs and a number of bank address signals;

b) re-mapping at least one of said address inputs as an additional bank address signal; and

c) providing said address inputs, bank address signals and said additional bank address signal as inputs to the memory device.

These and other aspects and advantages of the present invention will be readily understood and appreciated by

those skilled in the art from the following detailed description of the preferred embodiments with the best mode contemplated for practicing the invention in view of the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a functional block diagram of a memory module for a computer system in accordance with the present invention;

FIG. 1A illustrates an exemplary address re-mapping function carried out by the present invention; and

FIG. **2** is a flow chart for a negotiation process at the system level with a memory module using READ/WRITE PD data functions.

DETAILED DESCRIPTION OF THE INVENTION

With reference to FIG. **1**, an embodiment of the invention is illustrated in the environment of a computer system **10**. The computer system **10** can be any computer system that utilizes a memory module having presence detect (PD) data and programmable or selectable memory module functions and modes. Personal computer systems, such as an IBM APTIVA® or IBM PC-300™, could be used for the computer system **10**, to name just two of many examples. The computer system includes a CPU or system controller **12** which can include or be interfaced to a memory controller **14**. In this embodiment, the computer system **10** further includes a memory module **20**, as will be further described hereinafter. The memory controller **14** provides address, data and bus control signals for interfacing the CPU **12** and the memory module **20**. The memory controller **14** includes logic for addressing, receiving, writing and refreshing data in the plurality of memory devices **22** on the module **20**.

In accordance with one aspect of the invention, the memory module **20** is of the type that can be generally categorized as an "intelligent" module, in that the module **20** can operate in a plurality of selectable or programmable modes. The programmable feature of the module **20** is significantly advanced beyond the conventional mode selection criteria available by use of the Mode Register function of conventional memory chips such as synchronous DRAMs (SDRAMs). The memory module **20** can include memory devices or chips such as, for example, SDRAMs with standard Mode Register functions such as, for example, burst type, burst length and CAS Latency. Such chips are used today on memory modules such as, for example, Dual Inline Memory Modules or DIMMs. Other module architectures such as SIMMS could also be used. However, these mode register functions alone do not provide the level of flexibility needed to allow system level control to optimally interface with a number of different memory chip **22** designs and memory module **20** capabilities.

In accordance with one aspect of the invention, the memory module **20** includes a logic circuit **24**. In the embodiment, the logic circuit **24** is realized in the form of an application specific integrated circuit (ASIC). A suitable device for the ASIC **24** is a gate array ASIC such as a TOSHIBA ASIC TC160G. Suitable SDRAM devices **22** are IBM 0316409CT3 available from IBM.

The ASIC **24** includes or communicates with a volatile memory **26** over a bus **27**. The volatile memory **26** is used to store modified SPD data fields, as will be further explained herein.

The ASIC **24** further includes a look-up table **28** or comparable data set function that stores information about the programmable features of the memory module **20**. The ASIC **24** can also receive inputs in the form of signals from jumper wires (not shown). The settings of the jumper wires can indicate various features of the memory chip **22**, including, for example, whether the memory chips **22** are two bank, four bank or eight bank devices. Whether jumper inputs are used will be determined by the design of the memory module **20**, and is not a requirement of the present invention. The present invention can be utilized in any memory module **20** that can detect or otherwise determine the memory device characteristics, either through jumpers or look-up tables or SPD data, to name a few examples.

The use of a logic circuit **24** provides the capability to include a number of system level programmable or selectable features or operating modes. For example, the ASIC **24** can be configured to allow the module **20** to operate in several addressing modes. In one embodiment, the ASIC **24** effects an address re-mapping operation. This allows the system controller **12**, for example, to select or request an addressing option that is compatible with a mode available on the memory module **20**.

For example, SDRAM memories can include a number of internal banks of memory arrays. An ASIC can be configured to allow the use of a **4** bank SDRAM in a system that supports only **2** bank SDRAMs, by effecting an address re-mapping function. For clarity, the letter "M" will be used herein to represent an integer number of internal banks for the memory devices **22**. Additionally, the letter "N" will be used herein to represent an integer number of banks that the system addressing function is expecting the memory devices **22** to have.

By way of example, suppose the memory module **20** has SDRAM devices installed for the memory devices **22**. The ASIC **24** determines the configuration of the memory chips, such as the number of banks, either from jumper input signals, presence detect data, or information available in, for example, module memory **30** or a look-up table **28**. The particular source of the bank configuration information to the ASIC **24** is not critical to the present invention. As will be further described hereinafter, if the system **12** requires an addressing mode that is different from the mode of the memory devices **22**, the ASIC **24** can be used to effect an address re-mapping function in response to a request or negotiation process with the system **12**. In this manner, a four bank memory device **22** could still be used with a system **12** that is expecting or requires a two bank memory device, for example.

In order for the system controller **12** to be able to take advantage of programmable modes in the memory module **20**, the system controller **12** must be able to communicate with the module **20** to effect a mode request. In accordance with a significant aspect of the present invention, a technique is provided that allows the system controller **12** to negotiate an operating mode with the memory module **20**. In the described embodiment, this negotiation is effected by the use of the presence detect function of the memory module **20**.

Memory modules that use SDRAMs typically include a presence detect (PD) function. A non-volatile memory **30** such as an EEPROM is included on the DIMM and stores a PD data field. A typical PD data field includes 256 bytes of information which are further categorized into a number of segments as follows:

BYTE NOS.	DATA
0–35	Module functional and performance information
36–61	Superset data
62	SPD Revision
63	Checksum for bytes 0–62
64–127	Manufacturer's information
128–255	Reserved for system use

The PD data in bytes **0–35** can be used by a system controller to verify compatibility of the memory module **20** and the system requirements. The PD data can be read in serial or parallel format. Although serial PD data (SPD) is used in the exemplary embodiments herein, those skilled in the art will appreciate that the invention can be used with parallel PD data.

The information contained in bytes **0–127** is generally locked by the manufacturer after completion of the module build and test. This ensures that the data is not corrupted or overwritten at a later time.

In the embodiment of FIG. **1**, the system controller **12** accesses SPD data stored in a non-volatile memory **30**. The non-volatile memory **30** may be a separate memory device such as an EEPROM, or may be a memory array that is part of the ASIC logic device **24**. A suitable EEPROM with an integrated I2C bus controller (shown separately in the drawing for clarity) is a FAIRCHILD part no. NM24C03L. The system controller **12** reads the SPD data stored in the non-volatile memory **30** (via a bus **30***b*) by accessing the memory **30** through a standard I2C bus controller **32** and the system memory controller **14** which includes a corresponding I2C controller **14***a*. The I2C bus **34** is an industry standard serial bus, and the I2C bus controllers **14***a*, **32** can be, for example, a PHILLIPS part no. PCF8584 controller. The system I2C bus controller **14***a* may be located on the system mother board or integrated into the memory controller logic **14** as in FIG. **1**. The module I2C bus controller function can be and often is integrated with the non-volatile memory **30** and/or the ASIC device **24**. The system controller **12** accesses the memory controller **14** across a standard bus **44**.

The ASIC **24** also has access to data in the non-volatile memory **30**, via a bus **30** a. This is provided so that the ASIC **24** can, in some applications, be used to re-write the original PD data in the non-volatile memory **30**. Furthermore, in the case where the ASIC device **24** directs PD data to be read from the volatile memory **26**, the appropriate control signal, such as the I2C clock, is simply withheld from the non-volatile memory **30** by the ASIC **24**.

The system memory controller **14** communicates with the module **20** via an ADDRESS AND CONTROL bus **40**. This bus **40** can interface directly with the ASIC circuit **24** as illustrated, or can interface directly with the memory chips **22**, as indicated by the phantom bus **42**. Data flow typically is accomplished directly between the memory controller **14** and the memory chips **22** (as with the bus **48**), however, in some applications the ASIC may also be used, in addition to modifying addresses, for data formatting features such as parity, error correction and so on to name a few examples (in which case the bus **40** could also carry data signals). The present invention thus is not limited in terms of how data and control signals are exchanged between the system and the module **20**, but rather more generally to how the system can negotiate an address re-mapping mode of the module. Thus, although double ended arrows are used to represent data and control flow between the ASIC **24** and the memory chips **22**, this is intended to be exemplary in nature. Those skilled in the art will appreciate that the particular architecture used will depend on the actual programmable features incorporated into the memory module **20**. In some applications, for example, the ASIC **24** will send address and control signals to the memory chips **22**, but the data will flow directly to the memory controller **14**. In another example, control and data signals will flow directly between the memory controller **14** and the memory chips **22**, but the ASIC will provide address re-mapping or other features or controls. Thus, the exact flow of signals will depend on each particular implementation, and the exemplary embodiment of FIG. **1** should not be construed in a limiting sense.

It is further noted that the various circuits indicated as discrete functional blocks, such as blocks **26**, **28**, **30**, **32** and **50** may be part of the overall ASIC device **24**, as represented by the dashed box **24** a around those components.

The system controller **12** initially obtains the SPD data from the non-volatile memory **30** during boot-up after the computer **10** is powered up. A power on reset (POR) operation occurs which resets the module **20** logic to ensure that the preset module operation mode is initiated using the initial or original SPD data stored in the non-volatile memory **30**.

It is another aspect of the invention that the system **12** can originate a negotiation of memory module **20** functions or modes "on the fly", not just during a power on sequence. Although the embodiment described herein is explained in the context of a power on or boot up sequence, this is merely for convenience of explanation, and those skilled in the art will appreciate that the techniques and apparatus described herein allow the system **12** to negotiate a module **20** mode at any time by initiating a new SPD read/write operation and subsequent new mode selection.

In order to effect a negotiation between the system **12** and the memory module **20**, it is preferred but not required that the system controller **12** be able to ascertain whether the module **20** includes programmable features. It is contemplated that one of the PD data bytes, such as byte **61** in the address range for "Superset" will be designated to indicate that the memory module **20** has one or more programmable features (such as, for example, address re-mapping). One reason that it may not be required to include programmable information in a PD data byte is that the system **12** can be designed to request a mode change if needed and the logic device **24** could simply accept or reject the request based on the features available on the module **20**. The use of a byte such as byte **61** to indicate programmable features could speed up the negotiation process, particularly where the module **20** does not have programmable features.

Based on the initial PD data from the non-volatile memory **30**, the system controller **12** can compare the module **20** performance and operational features with the system requirements. This comparison can be effected by the system BIOS as is known. If the module **20** is compatible with the system **12** requirements, normal boot up and operation follows. If, however, the module **20** has module or device functions that are inconsistent with the system **12** requirements, and if the PD data indicates that the module **20** has one or more programmable features, then a negotiation process can be executed by the system **12**. Again, the latter requirement of an affirmative indication in the PD data of programmable features is not required in order to carry out the present invention but is a preferred embodiment.

A negotiation process between the system controller **12** and the module **20** can be implemented as follows. Based on

the system requirements, the system controller **12** writes or transfers modified or requested PD data to the module **20**. The modified PD data corresponds with a requested operating mode or function and can be transferred by a complete PD data field write of all 255 bytes, or alternatively the system controller **12** could write data for only the PD data entries that the system controller **12** desires to change. In either case, the modified PD data is generally transmitted to the logic device **24** by the memory controller **14** and the I2C controller **32**. The ASIC logic device **24** stores the modified PD data in the volatile memory **26**. A volatile memory **26** can be used to store the new PD data because when power is removed it will be preferred to effect a start up sequence with the "original" or initial PD data in the EEPROM **30**. Thus, it is further contemplated that for a system level negotiation, modified or requested PD data will not be written to the EEPROM **30** because it is desirable not to lose the original PD data therein. But, alternative techniques for preserving the original PD data while using the non-volatile memory **30** for the modified PD data, and then re-writing the original PD data back to the memory **30** could be implemented if needed, although such a process may not be feasible in some applications.

After receiving the modified or requested PD data from the system controller **12**, the ASIC logic device **24** can compare the new PD data and its corresponding modes or functions, with permitted modes or functions that are supported by the ASIC device **24**. The permitted functions can be obtained, for example, from the look-up table **28**, jumper wires, or PD data for example, as previously described herein above. This process does not require a "translation" per se of PD data to corresponding functions. For example, the ASIC device **24** can be provided with a look up table **28** or other suitable stored data format that indicates PD data values that it can support. The look-up table **28** may also store data that indicates various operational parameters of the memory chips, which data can be used to analyze additional compatibility features that might otherwise not be available from the conventional PD data and mode register functions.

In the case where the modified PD data corresponds to functions supported on the module **20**, the modified or new PD data is saved in the volatile memory **26** and normal start-up and operation continues under the new mode or function. Thereafter, the ASIC logic device controls the transfer of PD data either from the non-volatile memory **30** or the volatile memory **26** depending on which memory holds the most up-to-date PD data for each PD data byte. The volatile memory **26** can be designed to store all the PD data field entries, in which case PD data transfer can occur from the volatile memory **26** alone. Alternatively, the volatile memory **26** can be used to store only the new up-to-date PD data entries, in which case the ASIC device **24** will use both the non-volatile memory **30** and the volatile memory **26** to transfer PD data to the system controller **12**. In the latter case, it is contemplated that the ASIC device **24** will set a "flag" bit for each SPD address that is re-written by the system **12**. This bit can then be used to direct any future "SPD READ" operations to use the PD data contained in the volatile memory **26** for those addresses.

The system controller **12** may elect to verify that the new mode or function has been entered. In this case, the system performs a READ of the new PD data to verify compatible functions are in use. In general, the system controller **12** would then initiate a power on self test (POST) to ensure the memory module **20** is fully functional.

In the event that the module **20** is not programmable or does not have requested programmable functions supported by the ASIC logic device **24**, the system controller **12** will continue the boot up process with appropriate diagnostics or other initialization processes as normally occurs when incompatible memory devices are detected during power up.

The present invention relates to effecting an address re-mapping function **50** as one example of a programmable mode **28** carried out by the ASIC **24** in response to a negotiated operating mode with the system **12**. In this example, assume that the memory module **20** is a registered DIMM which includes a re-drive/flip-flop register **24** a on all inputs except CLOCK and DATA. The register **24** a can be integral with the ASIC **24** as in FIG. **1**, or separately provided. However, the present invention can be implemented with other module **20** designs, as will be apparent to those skilled in the art. Further assume that the module **20** includes memory devices **22** of **64** Megabit SDRAM devices that are configured as four bank devices. Finally, for this example, assume the system **12** expects or requires a memory device with two bank devices.

As four bank devices, each SDRAM **22** uses twelve address signals (**A0**–**A11**) and two bank address signals (**BA0** and **BA1**). However, the system **12** is expecting memory chips with two banks, and therefore will address the module **20** with thirteen address signals (**A0**–**A12**) and only one bank address signal **BA0**, with the second bank address signal **BA1** missing from the system **12** address inputs to the module **20**. This is represented in FIG. **1**A on the left side of the figure.

The ASIC **24** effects the address re-mapping function **50** by connecting the highest order address signal (in this case **A12**) to the **BA1** input pin of the memory devices **22**, as shown on the right side of FIG. **1**A. The ASIC **24** can effect this by a simple switch circuit. At RAS time, twelve addresses and one bank address (**A0**–**A11** and **BA0**) are provided to the SDRAM **22** in conjunction with the **BA1** signal (available from the **A12** system **12** address input). The ASIC **24** needs to store the **BA1** address applicable to each of the **BA0** options. This allows the ASIC **24** to re-send the **BA1** signal at CAS time to ensure that the correct bank is addressed. The ASIC can use any convenient memory location to store the **BA1** signal for CAS. Note that in this example, no more than two of the possible four banks in the SDRAM **22** are active at once.

Note that although the address re-mapping function **50** is illustrated as a separate functional block **50** in FIG. **1**, this is for ease of description and understanding and need not represent a separate function or circuit within the module **20**. The address re-mapping function **50** can be implemented as part of the addressing circuitry present in the design of the ASIC **24** for interfacing or providing the address signals to the memory device **22**.

Also it should be noted that the exemplary embodiment should not be construed in a limiting sense. The present invention can be used with many different size DRAM devices with different configurations (e.g. 2Mx32, 4Mx16, 8Mx8, 16Mx4 in the 64 Megabit example). Memory devices that contain eight banks, for example, can be used with the present invention for systems **12** expecting four bank devices (in the 64 Megabit example, the third bank address signal **BA2** is provided by a re-map of the high order address **A12**). Thus, in one general aspect of the invention, a 2N bank memory device can be used in systems having addressing for N banks, by effecting a re-mapping of the high order address bit for the missing BA signal.

With reference to FIG. **2**, a suitable control process in accordance with the invention is provided. At step **200** a POR sequence is performed to initialize the memory module

20. At step 202 the system controller 12 accesses the initial PD data stored in the non-volatile memory 30. In the described embodiment, step 202 is a serial PD READ operation via the I2C bus 34 and I2C controller 32.

At step 204 the system controller 12 determines whether the initial operating modes and functions of the memory module 20 are compatible with system level requirements. If YES, normal operation continues at step 206. If NO, the system controller 12 at step 208 writes modified or new PD data to the memory module 20, which new PD data is stored in the volatile memory 26. Shown in dashed lines on FIG. 2 is a related step 208 a for systems wherein a PD data entry is used as a flag or marker to indicate to the system controller 12 whether the module 20 supports programmable functions or modes. If NO, the system enters its normal diagnostic/configuration functions at step 210.

At step 212 the ASIC logic device 24 determines whether the requested function, as indicated by the modified PD data, is supported on the memory module 20. If YES, the up-to-date PD data is stored (step 214) and provided during subsequent READ operations (step 216) during normal operation (step 206). If the requested function is not supported by the memory module 20 as determined at step 212, the system enters the normal diagnostic/configuration functions at step 210, as is the case from step 208 a if the module 20 is not programmable.

Step 214 can include as part 214*a* the address re-mapping function 50 of the exemplary embodiment (of the 64 Megabit chips 22 with four banks but addressed as two bank devices). In that example, the high order address bit A12 is re-mapped to be the BA1 input to the memory devices 22. This BA1 input is temporarily saved for the CAS sequence and each subsequent sequence for which the ASIC 24 has to address the correct bank.

The invention thus provides techniques for system level negotiation with a programmable memory module by using PD READ/WRITE functions, and in particular an address re-mapping function.

While the invention has been shown and described with respect to specific embodiments thereof, this is for the purpose of illustration rather than limitation, and other variations and modifications of the specific embodiments herein shown and described will be apparent to those skilled in the art within the intended spirit and scope of the invention as set forth in the appended claims.

What is claimed is:

1. A memory module comprising: a plurality of memory devices associated with the memory module; each of said memory devices being configured in M banks; and a logic circuit for configuring the memory module to operate in a programmable addressing mode; said logic circuit receiving a number of address inputs and a number of bank address signals from a memory controller with said address inputs and bank address signals corresponding to N bank memory devices where M and N are integers and can be different; said logic circuit re-mapping at least one of said address inputs as an additional bank address signal to the memory device where M and N are different; and a non-volatile memory for storing initial presence detect (PD) data; a volatile memory for storing modified PD data that corresponds to a requested mode of operation of a memory module input by a system controller; and a bus controller for interfacing the memory module with the system controller.

2. The memory module of claim 1 wherein said system controller operates to read the initial PD data and to write modified PD data to said volatile memory; said modified PD data corresponding to a system controller requested addressing mode of the module; said logic circuit operating to accept said modified PD data when the requested addressing mode is available in the module.

3. The memory module of claim 2 wherein said logic circuit operates to control said modified PD data transfer to the system controller from said volatile and non-volatile memories based on which memory stores up-to-date PD data.

4. The memory module of claim 1 wherein said initial PD data is serially accessed with a serial bus controller that interfaces with the system controller.

5. The memory module of claim 1 wherein said non-volatile memory comprises an EEPROM that stores original serial PD data.

6. The memory module of claim 1 wherein said logic circuit comprise an ASIC (application specific integrated circuit) logic device and includes a non-volatile memory that stores original serial PD data.

7. The memory module of claim 1 comprising an I2C bus controller and an I2C bus; said system controller reading serial PD data from said non-volatile memory after memory power up, and determining compatibility of memory module addressing functions with system requirements.

8. The memory module of claim 1 wherein said system controller writes said modified PD data to said volatile memory that corresponds to a desired addressing mode of the memory module based on system requirements.

9. The memory module of claim 8 wherein said logic circuit operates to accept said modified PD data when the corresponding desired addressing mode is available in the module.

10. The memory module of claim 9 wherein said logic circuit operates to control PD data transfer to the system controller from said volatile and non-volatile memories based on which memory stores up-to-date PD data, said logic circuit setting a flag bit for each modified PD data entry so that PD data that has been modified is read by the system controller from said volatile memory.

11. The memory module of claim 1 comprising a serial bus and serial bus controller for interfacing a system controller with the module so that the system controller can read and write serial PD data of the module to negotiate an addressing mode for the module.

12. The memory module of claim 1 wherein M is greater than N.

13. The memory module of claim 12 wherein said logic circuit detects an address input from a system memory controller and re-maps and saves said address input as a bank address signal.

14. The memory module of claim 12 wherein said logic circuit re-maps an address input into a bank address signal to provide N bank addressing using M bank SDRAM devices.

15. The memory module of claim 1 wherein at least one of said memory devices comprises a synchronous DRAM (SDRAM) memory device.

16. The memory module of claim 1 wherein said logic circuit connects a highest order address input from the memory controller to said additional bank address signal of a memory device being addressed.

17. A computer system comprising: a system controller; a memory module comprising a plurality of memory devices on the module, and a memory module logic circuit for configuring the memory module to operate in a programmable addressing mode; each of said memory devices being configured with M banks; said logic circuit receiving a number of address inputs and a number of bank address

signals from said system controller with said address inputs and bank address signals corresponding to N bank memory devices wherein M and N are integers and can be different; said logic circuit re-mapping at least one of said address inputs as an additional bank address signal to the memory device wherein M and N are different; and wherein said system controller negotiates an addressing mode of the memory module by reading and writing presence detect (PD) data of the memory module.

18. The computer system of claim **17** wherein bus controller comprises a serial bus controller and said system controller operates to read and write serial PD data of the memory module.

19. The computer system of claim **17** wherein said memory module comprises a volatile memory that stores modified PD data written by said system controller that corresponds to a requested addressing mode, and a non-volatile memory that stores initial PD data, said volatile and non-volatile memories being operatively controlled by said logic circuit.

20. The computer system of claim **19** wherein said logic circuit accepts said modified PD data when the requested addressing mode is available in the module, and controls a transfer of PD data from said volatile and non-volatile memories based on which memory stores up-to-date data with respect to each PD data entry.

21. The computer system of claim **17** wherein said logic circuit re-maps addressing signals for said memory module to provide N bank addressing using M bank memory chips in said memory module.

22. A method for system control of an intelligent memory module, comprising:

a) reading presence detect (PD) data from a non-volatile memory on the module;

b) writing modified PD data to a volatile memory based on a requested addressing mode; and

c) controlling transfer of said PD data between the memory module and a system controller based on which memory stores up-to-date PD data.

23. The method of claim **22** wherein a logic circuit on the module re-maps an address input to a bank address signal to provide N bank addressing using M bank memory chips.

24. The method of claim **22** wherein the logic circuit receives a highest order address signal and re-maps and saves said address input as said bank address signal.

25. A method for using an M bank memory device in a computer system that has N bank addressing wherein and M and N are integers and are different, comprising the steps of:

a) inputting address signals from a system controller to a logic circuit, said address signals including a number of address inputs and a number of bank address signals;

b) re-mapping at least one of said address inputs as an additional bank address signal including the step of connecting a highest order address input received from the system controller to one of the bank address signals of the memory device; and

c) providing said address inputs, said bank address signals and said additional bank address signals as inputs to the memory device.

* * * * *

APPENDIX D

US006414868B1

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

(10) **Patent No.: US 6,414,868 B1**
(45) **Date of Patent: *Jul. 2, 2002**

(54) **MEMORY EXPANSION MODULE INCLUDING MULTIPLE MEMORY BANKS AND A BANK CONTROL CIRCUIT**

(75) Inventors: **Tayung Wong**, Fremont; **John Carrillo**, San Jose; **Jay Robinson**, Sunnyvale; **Clement Fang**, Cupertino, all of CA (US)

(73) Assignee: **Sun Microsystems, Inc.**, Palo Alto, CA (US)

(*) Notice: This patent issued on a continued prosecution application filed under 37 CFR 1.53(d), and is subject to the twenty year patent term provisions of 35 U.S.C. 154(a)(2).

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

(21) Appl. No.: **09/327,058**

(22) Filed: **Jun. 7, 1999**

(51) **Int. Cl.**[7] **G11C 5/02**
(52) **U.S. Cl.** **365/51**; 365/193; 365/230.03
(58) **Field of Search** 365/51, 52, 63, 365/230.03, 193, 189.01, 230.01, 222; 711/5, 1

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,025,903	A		5/1977	Kaufman et al.
4,630,230	A		12/1986	Sundet
5,228,132	A		7/1993	Neal et al.
5,260,892	A		11/1993	Testa
5,265,218	A		11/1993	Testa et al.
5,270,964	A		12/1993	Bechtolsheim et al.
5,272,664	A		12/1993	Alexander et al.
5,339,269	A		8/1994	Takagi
5,504,700	A		4/1996	Insley et al.
5,522,064	A		5/1996	Aldereguia
5,686,730	A		11/1997	Laudon et al.
5,745,914	A	*	4/1998	Connolly et al. 711/105
5,771,199	A		6/1998	Lee
5,961,660	A	*	10/1999	Capps, Jr. et al. 714/763
6,038,132	A		3/2000	Tokunaga et al.

FOREIGN PATENT DOCUMENTS

EP	0 744 748	11/1996
EP	0 813 204	12/1997

OTHER PUBLICATIONS

PNY Technologies, "2M ×32 Bit 5V EDO SIMM; Extended Data Out (EDO) DRAM SIMM", 1996.

* cited by examiner

Primary Examiner—Hoai V. Ho
(74) *Attorney, Agent, or Firm*—Conley, Rose & Tayon, PC; B. Noël Kivlin

(57) **ABSTRACT**

A memory expansion module including multiple memory banks and a bank control circuit is disclosed. In one embodiment, a memory module includes a printed circuit board with a connector edge adapted for insertion in an expansion socket of a computer system. Mounted upon the circuit board is a plurality of memory chips, typically Dynamic Random Access Memory (DRAM) chips, which make up an upper bank and a lower bank of memory. A buffer circuit is mounted upon the printed circuit board, for the purpose of driving address signals, Column Address Strobe (CAS) signals, and write enable signals to each of the memory chips. Also mounted upon the printed circuit board is a bank control circuit, which is coupled to the memory chips. An address signal is used as a bank selection input to the bank control circuit, which will drive Row Address Strobe (RAS) signals to the memory chips of the selected memory bank. The bank control circuit is further configured to drive RAS signals to both banks simultaneously during CBR (CAS before RAS) refresh operations, which occur when a CAS signal is asserted before a RAS signal.

12 Claims, 7 Drawing Sheets

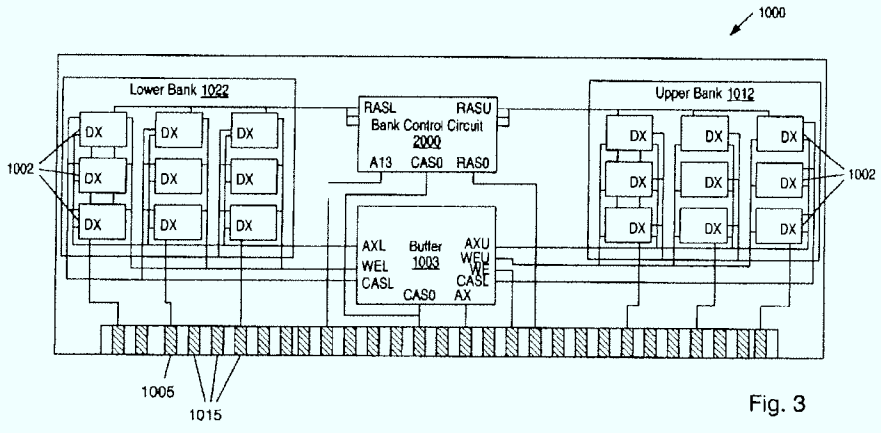


Fig. 3

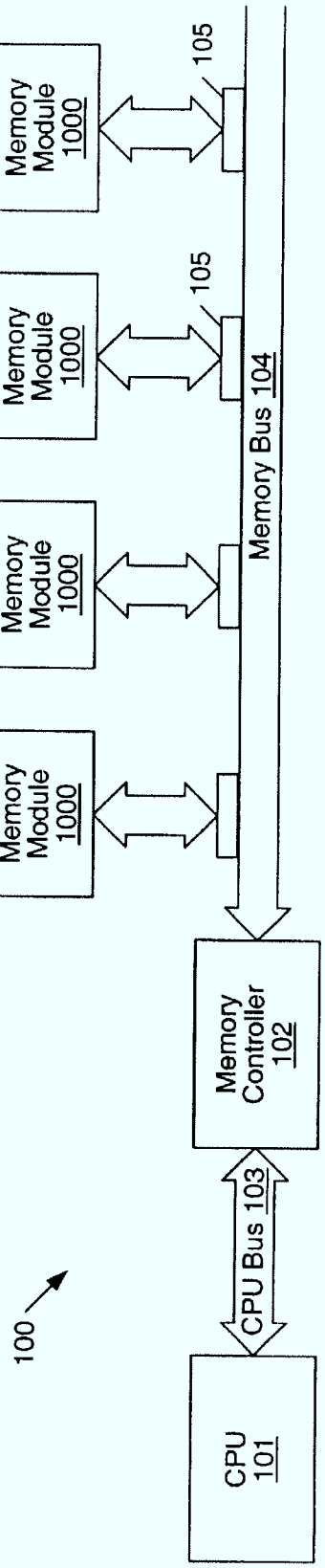

Memory Module 1000
105
Memory Module 1000
105
Memory Module 1000
Memory Module 1000
Memory Bus 104
Memory Controller 102
CPU Bus 103
CPU 101
100

Fig. 1

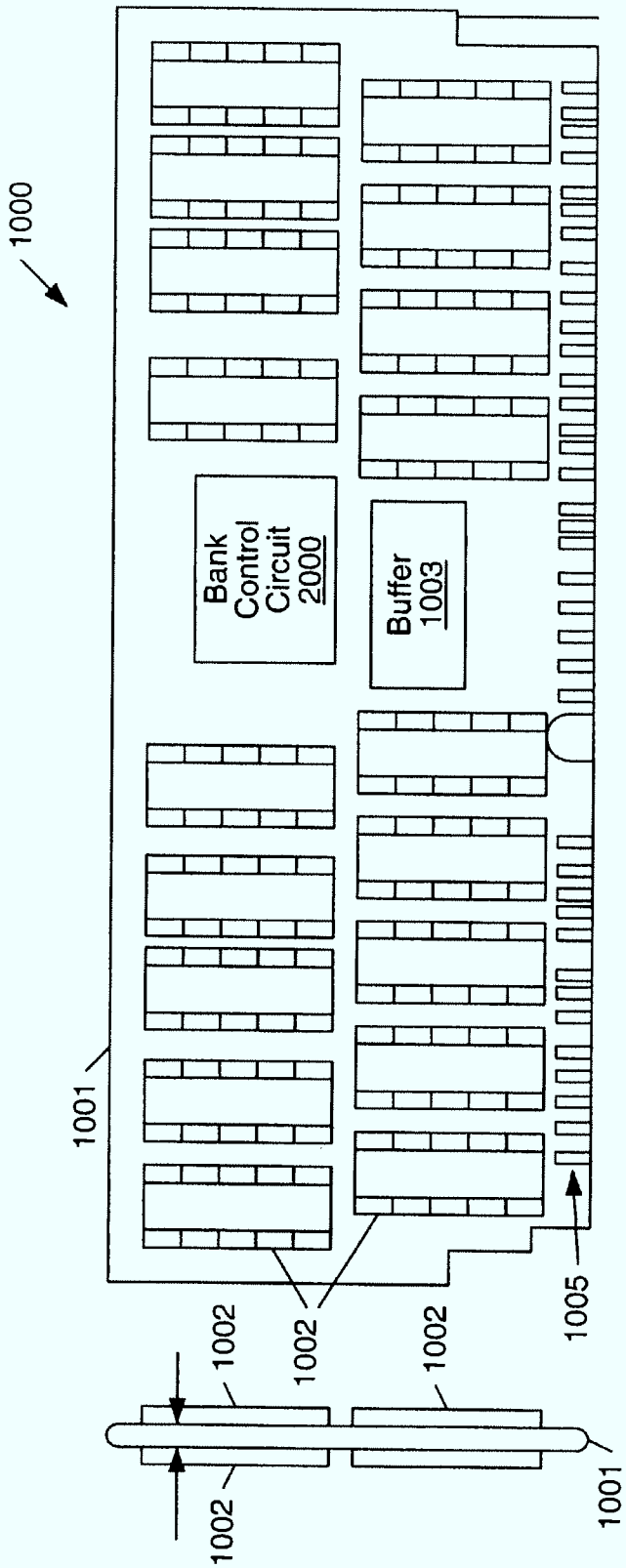

1000
Bank Control Circuit 2000
Buffer 1003
1001
1002
1002
1002
1002
1005
1001

Fig. 2

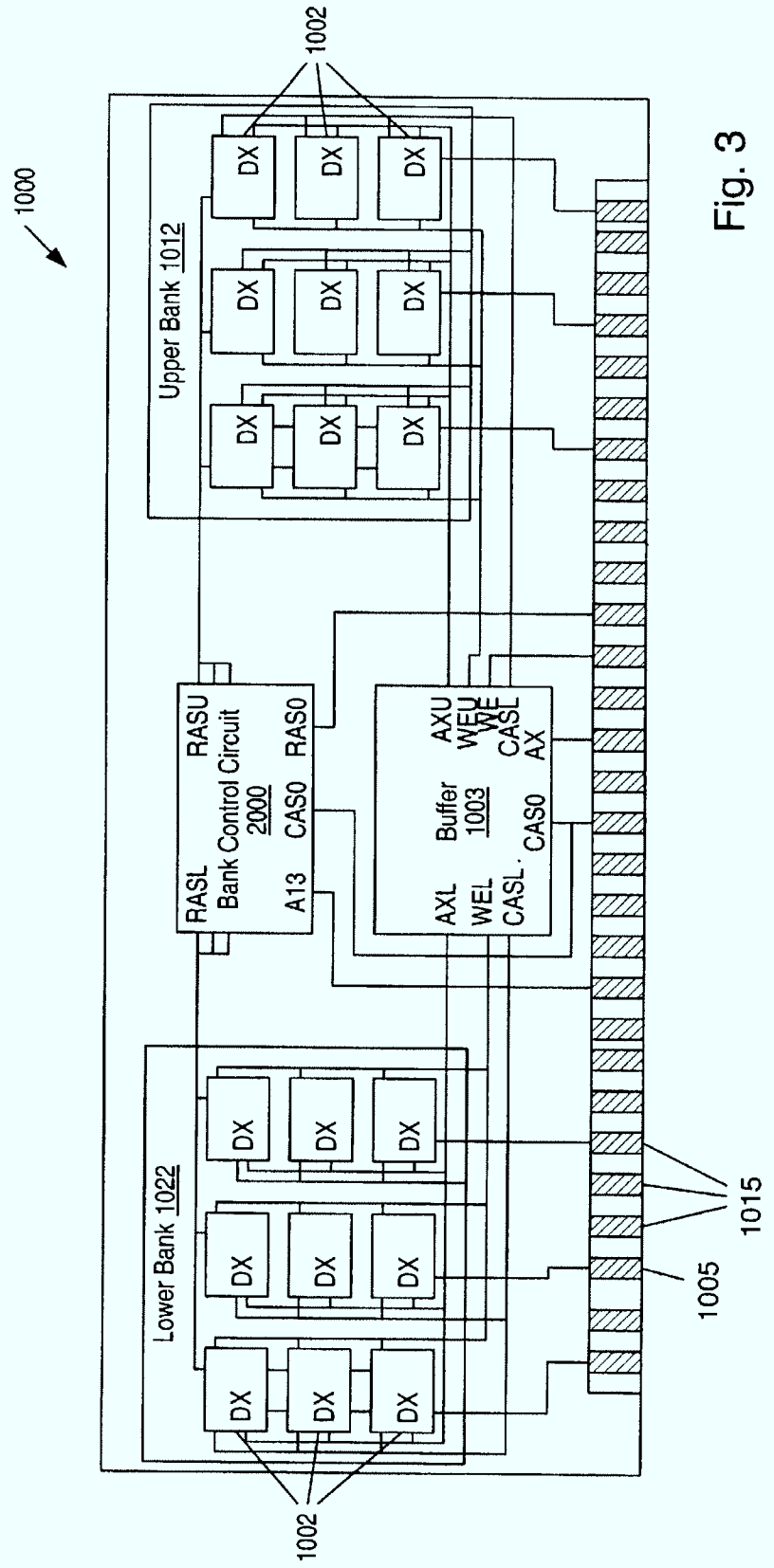

1000
1002
Upper Bank 1012
DX
RASU
RAS0
RASL
Bank Control Circuit
2000
CAS0
A13
AXU
WEU
CASL
AX
Buffer
1003
CAS0
AXL
WEL
CASL
Lower Bank 1022
1015
1005
1002

Fig. 3

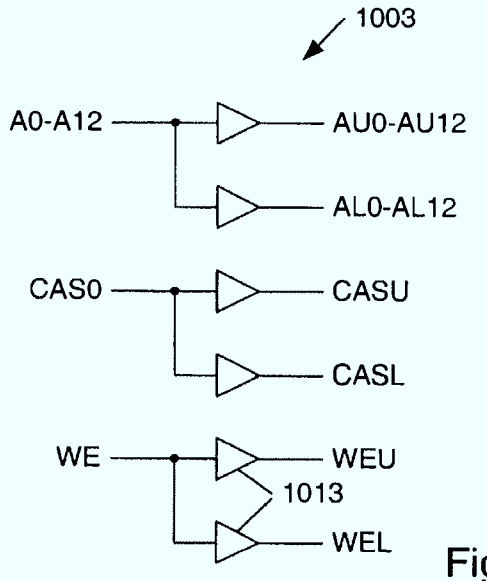


Fig. 4A

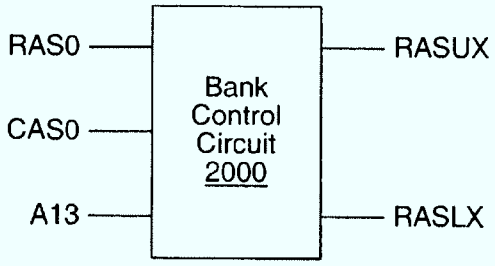


Fig. 4B

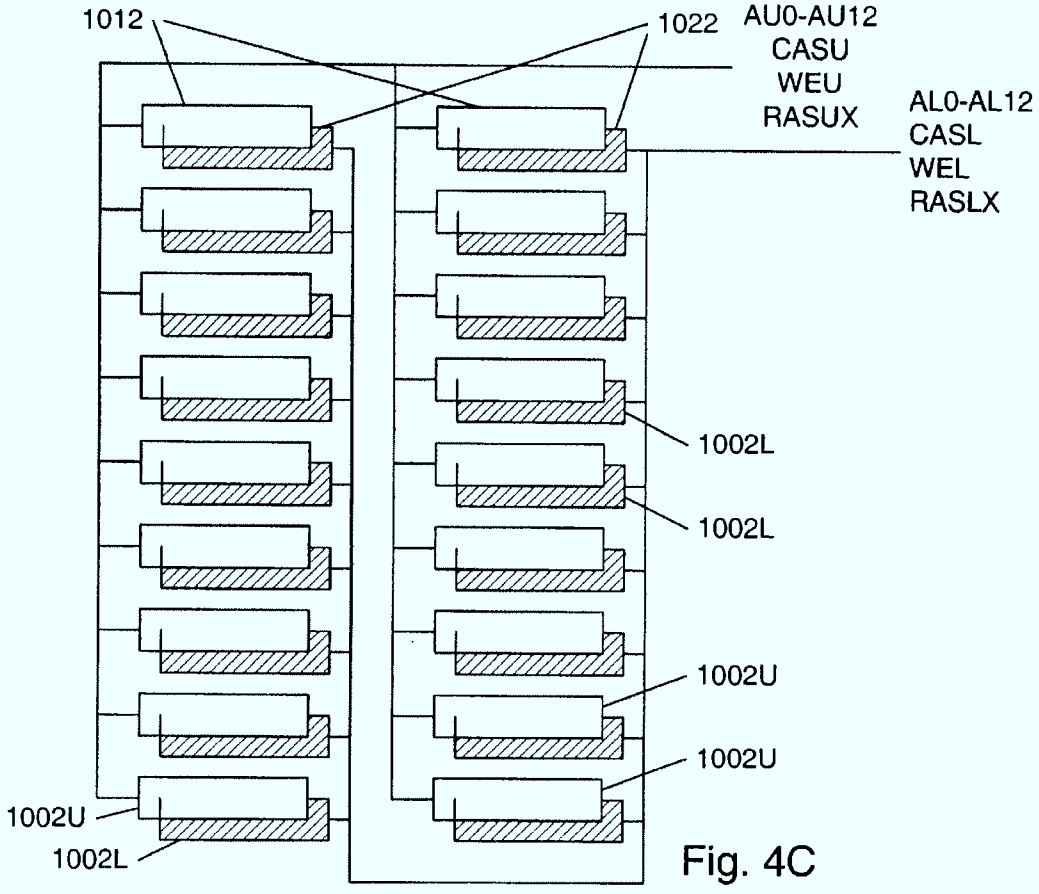


Fig. 4C

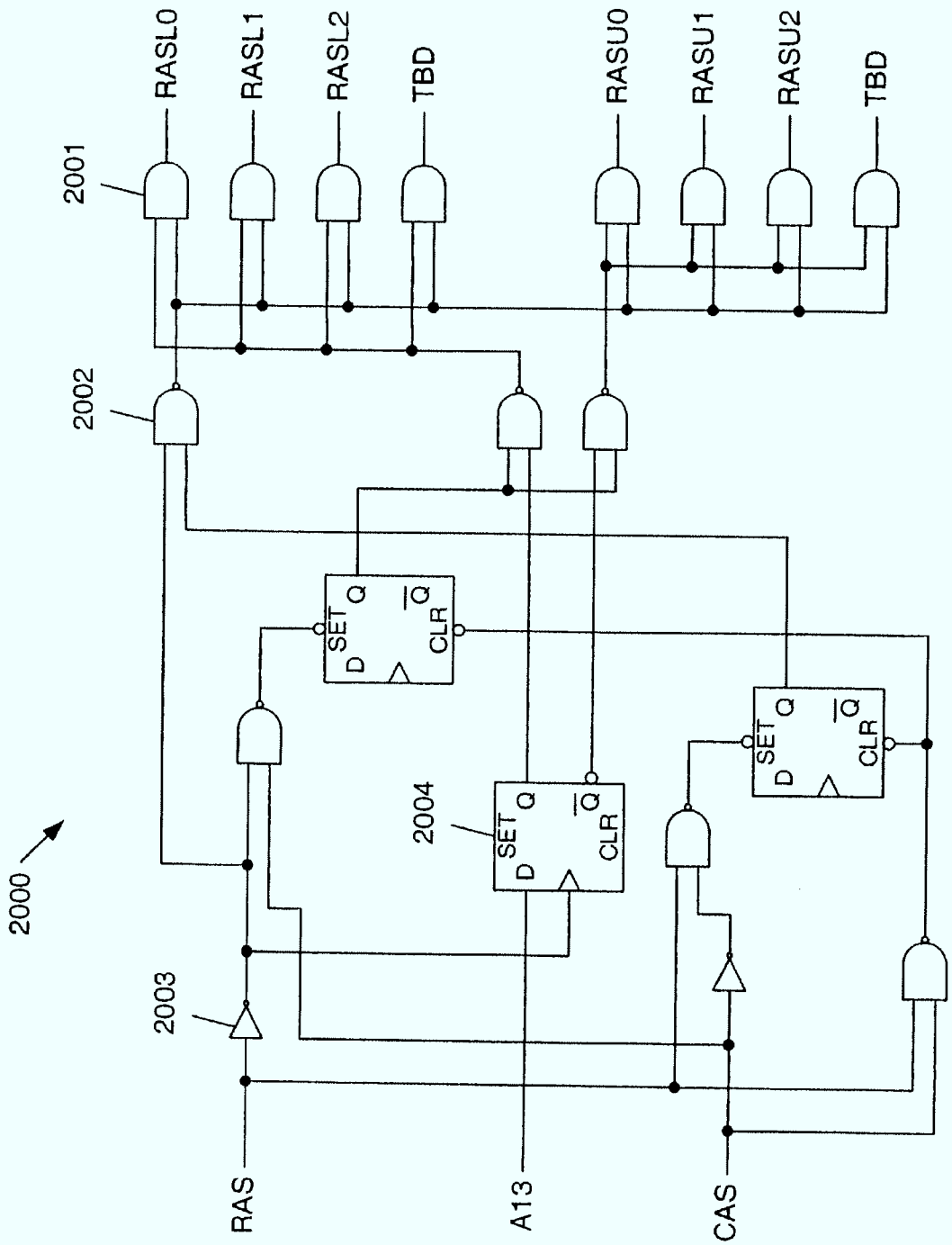

RASL0
RASL1
RASL2
TBD
RASU0
RASU1
RASU2
TBD
2001
2002
2003
2004
2000
SET
CLR
D
Q
RAS
A13
CAS

Fig. 5

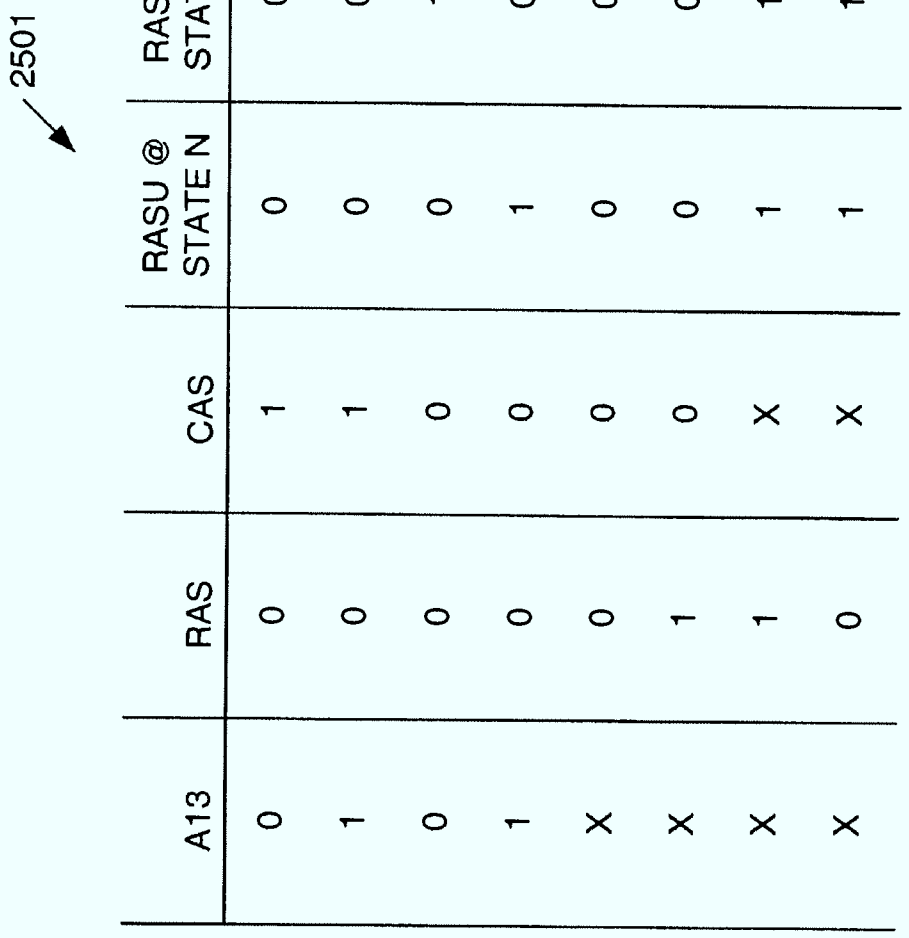

2501
A13
RAS
CAS
RASU @ STATE N
RASL @ STATE N
RASU @ STATE N+1
RASL @ STATE N+1
0 0 1 0 0 1
1 0 1 0 1 0
0 0 0 1 0 0
1 0 0 0 1 0
X 0 0 0 0 0
X 1 0 0 1 1
X 1 X 1 1 1
X 0 X 1 0 0

Fig. 6

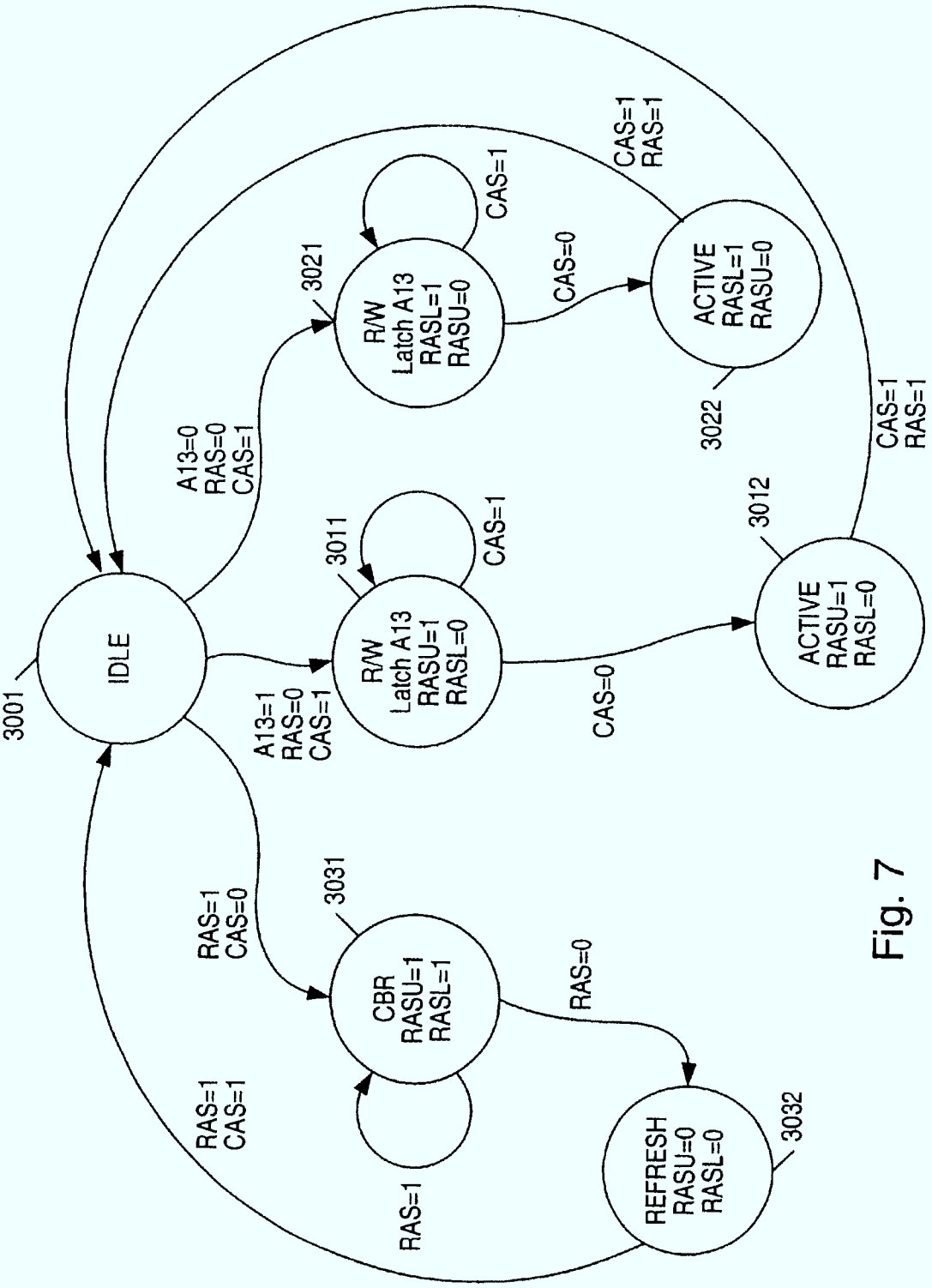

3001
IDLE
A13=0
RAS=0
CAS=1
3021
R/W
Latch A13
RASL=1
RASU=0
CAS=1
CAS=0
ACTIVE
RASL=1
RASU=0
3022
CAS=1
RAS=1
A13=1
RAS=0
CAS=1
3011
R/W
Latch A13
RASU=1
RASL=0
CAS=1
CAS=0
3012
ACTIVE
RASU=1
RASL=0
CAS=1
RAS=1
RAS=1
CAS=0
3031
CBR
RASU=1
RASL=1
RAS=0
RAS=1
CAS=1
RAS=1
REFRESH
RASU=0
RASL=0
3032

Fig. 7

MEMORY EXPANSION MODULE INCLUDING MULTIPLE MEMORY BANKS AND A BANK CONTROL CIRCUIT

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention generally relates to memory hardware for computer systems, and more specifically to memory expansion modules for expanding memory in computer systems.

2. Description of the Relevant Art

Many modern computer systems allow for memory expansion by way of single inline memory modules (SIMMs) and/or dual inline memory modules (DIMMs). SIMMs and DIMMs include small, compact circuit boards that are designed to mount easily into an expansion socket mounted on another circuit board, typically a computer motherboard. The circuit boards used to implement SIMMs and DIMMs include an edge connector comprising a plurality of contact pads, with contact pads typically being present on both sides of the circuit board. On SIMMs, opposing contact pads are connected together (i.e. shorted), and thus carry the same signal, while at least some opposing contact pads on DIMMs are not connected, and thus allowing different signals to be carried. Due to this, higher signal density may be accommodated by DIMMs.

Memory elements mounted on SIMMs and DIMMs are typically Dynamic Random Access Memory (DRAM) chips. DRAM chips store information as a charge on a capacitor, with the charge level representing a logic one or logic zero. Since a capacitor charge will dissipate over time, DRAM chips require refresh cycles on a periodic basis.

To access a location in a DRAM, an address must first be applied to the address inputs. This address is then decoded, and data from the given address is accessed. In modern DRAMs, rows and columns are addressed separately using row address strobe (RAS) and column address strobe (CAS) control signals. By using RAS and CAS signals, row and column addresses can be time-multiplexed on common signal lines, contact pads, and pins of the address bus. This allows a greater number of memory locations that can be addressed without a corresponding increase in the number of required signal lines, contact pads, and pins.

To address a memory location in a DRAM as described above, a RAS signal is asserted on the RAS input of the DRAM, and a row address is forwarded to row decode logic on a memory chip. The contents of all locations in the addressed row will then be sent to a column decoder, which is typically a combination multiplexer/demultiplexer. After row addressing is complete, a CAS signal is asserted, and a column address is sent to the column decoder. The multiplexer in the column decoder will then select the corresponding column from the addressed row, and the data from that specific row/column address is placed on the data bus for used by the computer system.

Although the RAS and CAS signals allow the time-multiplexing of address signals, total memory capacity in a system may be limited by the number of address inputs on the memory chips employed in the system. This is true even if the system address bus is wider than the number of address inputs for an individual memory chip. The use of memory chips with a greater number of address inputs, and hence higher capacity, may disproportionately increase the cost of the desired memory expansion. It would be desirable to increase the memory capacity for such a computer system by adding extra banks of memory without having to change the type of memory chip employed. However, the number of address inputs to the system's memory chips limits the ability to do this. Furthermore, the presence of only one RAS and one CAS signal also limits the ability to expand system memory, as a separate bank of memory typically requires at minimum either a unique RAS or unique CAS signal for each bank. As such, it would be desirable to overcome the limitations described above in order to allow extra banks of memory to be added to a computer system, thereby expanding system memory capacity.

SUMMARY OF THE INVENTION

The problems outlined above may in large part be solved by a memory expansion module including multiple memory banks and a bank control circuit in accordance with the present invention. In one embodiment, a memory module includes a printed circuit board with a connector edge adapted for insertion in an expansion socket of a computer system. Mounted upon the circuit board is a plurality of memory chips, typically Dynamic Random Access Memory (DRAM) chips, which make up an upper bank and a lower bank of memory. A buffer circuit is mounted upon the printed circuit board, for the purpose of driving address signals, Column Address Strobe (CAS) signals, and write enable signals to each of the memory chips. Also mounted on the printed circuit board is a bank control circuit, which is coupled to the memory chips. An address signal is used as a bank selection input to the bank control circuit, which will drive Row Address Strobe (RAS) signals to the memory chips of the selected memory bank. The bank control circuit is further configured to drive RAS signals to both banks simultaneously during CBR (CAS before RAS) refresh operations, which occur when a CAS signal is asserted before a RAS signal. By using the bank control circuit to enable the addition of a second memory bank, a memory expansion can be realized without the need for higher capacity memory chips, which may result in an advantageous cost savings.

In one embodiment of the invention, a Dual Inline Memory Module (DIMM) employs a bank control circuit for bank selection. In this embodiment, the bank control circuit is a programmable logic device (PLD), although this circuit may be implemented in other forms for different embodiments. The bank control circuit receives a RAS signal, a CAS signal, and the selected address bit from expansion socket of a computer system. The bank control circuit drives multiple RAS signals. When the bank control circuit is in an idle state, receiving a RAS signal will cause a memory access operation to begin. The bank of memory to be selected will depend on the logic level of the address input to the bank control circuit. The bank control circuit will then drive RAS signals to the selected memory bank, allowing a row address to be selected. When the memory chips of the selected bank receive a CAS signal, the column address is selected, and the requested memory address is accessed.

If, when in an idle state, the bank control circuit receives a CAS signal, a CBR refresh cycle is begun. The CAS signal is received by both the bank control circuit and a buffer circuit, which drives CAS signals to each of the memory chips. Following this, a RAS signal is received by the bank control circuit, which then drives RAS signals to each of the DRAM chips of the memory module, and the CBR refresh is performed. When both the CAS and RAS inputs to the memory module are deasserted, the bank control circuit returns to an idle state.

Thus, in various embodiments, the memory expansion module with multiple memory banks and a bank control

circuit advantageously allows greater memory capacity by accommodating multiple memory banks. Additionally, memory capacity can be increased without requiring the use of memory chips with a higher address width.

BRIEF DESCRIPTION OF THE DRAWINGS

Other objects and advantages of the invention will become apparent upon reading the following detailed description and upon reference to the accompanying drawings in which:

FIG. **1** is a block diagram illustrating a computer system having a CPU, a memory controller, a CPU bus, and a plurality of memory modules in one embodiment of the invention;

FIG. **2** is a diagram illustrating components associated with an embodiment of a memory module;

FIG. **3** is a block diagram illustrating the electrical connections associated with an embodiment of the memory module;

FIG. **4A** is a schematic illustrating an embodiment of the line driver chip;

FIG. **4B** is a drawing illustrating an embodiment of the bank control circuit;

FIG. **4C** is a block diagram illustrating the upper and lower banks of memory with associated control signal connections;

FIG. **5** is a schematic of a bank control circuit;

FIG. **6** is a truth table that illustrates some of the state transitions of the bank control circuit in FIG. **5**;

FIG. **7** is state diagram illustrating operations of one embodiment of the bank control circuit.

While the invention is susceptible to various modifications and alternative forms, specific embodiments thereof are shown by way of example in the drawings and will herein be described in detail. It should be understood, however, that the drawings and description thereto are not intended to limit the invention to the particular form disclosed, but, on the contrary, the invention is to cover all modifications, equivalents, and alternatives falling with the spirit and scoped of the present invention as defined be the appended claims.

DETAILED DESCRIPTION OF THE INVENTION

Referring now to FIG. **1**, a computer system **100** employing one embodiment of memory modules **1000**, as will be described below, is shown. The computer system includes a CPU **101**, coupled to a memory controller **102** through a CPU bus **103**. The memory controller **102** is coupled to each of the memory modules **1000** through a memory bus **104**. In this embodiment, the memory modules are provided to expand main memory of computer system **100**, and are coupled to memory bus **104** through a set of expansion sockets **105**.

Turning now to FIG. **2**, a diagram illustrating components associated with a memory module **1000** is shown. In this particular embodiment, a plurality of memory elements **1002**, typically DRAM (Dynamic Random Access Memory) chips, are surface mounted upon a printed circuit board (PCB) **1001**. On one edge of the printed circuit board is an edge connector **1005**, which includes a plurality of electrical contact pads. These contact pads electrically couple memory module **1000** to the memory bus **104** by mating with a corresponding expansion socket **105** shown in FIG. **1**. Signals passing through the edge connector include data signals, address signals, and control signals. In one embodiment, at least some of the opposing electrical contact pads of edge connector **1005** are not connected (i.e. shorted), thus making the embodiment a Dual Inline Memory Module (DIMM).

Also mounted on PCB **1001** are a buffer chip **1003** (which can also be referred to as a line driver) and a bank control circuit **2000**. Buffer chip **1003** may be used to provide additional fan-out capability for control signals, address signals, and data signals. A bank control circuit **2000** is also mounted upon PCB **1001**.

A block diagram illustrating the electrical connections associated with one embodiment of the memory module is shown in FIG. **3**. Memory module **1000** includes an edge connector **1005**, a lower memory bank **1022**, and upper memory bank **1012**, a bank control circuit **2000** and a buffer **1003**. Each memory bank includes of a plurality of memory chips **1002**. The edge connector **1005** includes a plurality of electrical contact pads **1015** which convey signals between the memory module and the system memory bus. Edge connector **1005** is adapted for mounting in a socket within a computer system. Buffer **1003** receives signals WE (write enable), CAS**0** (Column Address Strobe **0**), and a plurality of address signals, shown as AX. Buffer circuit **1003** drives a plurality of address signals AXL and AXU, which are conveyed to the lower memory bank **1022** and upper memory bank **1012**, respectively. WEL and WEU are write enable signals driven by buffer **1003** to a lower memory bank **1022** and an upper memory bank **1012**, respectively. CASL and CASU are CAS signals driven by buffer **1003** to the lower memory bank **1022** and upper memory bank **1012**, respectively. The bank control circuit **2000** is configured to receive an address signal A**13** for selecting the upper and lower bank. Address signal A**13**, in this embodiment, is the most significant address bit of an address bus that is 14 bits wide. Bank control circuit **2000** is also configured to receive a CAS**0** signal and a RAS**0** (Row Address Strobe **0**) signal. A plurality of data lines, represented in the drawing as DX, convey data signals between the memory chips **1002** and system memory bus **104** of FIG. **1**. In this particular embodiment, the data path is 144 bits wide.

FIG. **4A** further illustrates an internal configuration of an embodiment of a buffer chip **1003**. In the embodiment shown, buffer chip **1003** receives a plurality of address signals, A**0**–A**12**, a CAS**0** signal, and a WE signal. The signals are passed through buffers **1013**, generating corresponding signals that will be provided to an upper bank and a lower bank of memory chips. For example, the input signal WE produces two output signals, WEL and WEU for a lower and an upper bank of memory, respectively. The signals are split and buffered in this manner in order to provide sufficient signal strength for control and address inputs at the memory chips.

One embodiment of bank control circuit **2000** is shown in FIG. **4B**. In this embodiment, bank control circuit **2000** receives input signals RAS**0**, CAS**0**, and address signal A**13**. Bank control circuit **2000** drives a plurality of RASLX and RASUX signals to the lower and upper memory banks, respectively. Depending on the combination of inputs received by bank control circuit **2000**, either the RASUX or RASLX signal groups can be asserted exclusively for memory access operations. Another combination of inputs will assert all RASUX and RASLX signals in order to perform a CBR (Columns before Rows) refresh cycle.

The arrangement of the upper and lower memory banks in one embodiment of the memory module is illustrated in FIG.

4C. Each bank comprises a plurality of memory chips **1002**. The memory chips are connected in such a manner as to form an upper bank **1012** and a lower bank **1022**. The upper lower banks each include a plurality of memory chips **1002**U and **1002**L, respectively. Note that in the embodiment shown, the shaded memory chips are in the lower bank, while those that are shown as unshaded are part of the upper bank. Address, CAS, RAS, and WE signals exclusive to each bank are driven to each of memory chips **1002**.

Turning now to FIG. **5**, a schematic of one embodiment of the bank control circuit **2000** is shown. This particular embodiment of bank control circuit **2000** is a programmable logic device (PLD. In this embodiment, bank control circuit **2000** comprises a plurality of AND gates **2001**, NAND gates **2002**, inverters **2003**, and flip-flops **2004** (D-type flip-flops in this embodiment). Bank control circuit **2000** drives multiple RAS signals for each memory bank in order to provide sufficient signal drive strength to each of the memory chips. The operation of the bank control circuit will be further illustrated in FIG. **6** and FIG. **7**.

It should be noted that alternative embodiments of the bank control circuit are possible using other types of electronic circuitry.

FIG. **6** is a truth table **2501** that illustrates some of the state transitions that may occur in bank control circuit **2000** in one embodiment of the memory module. The first two rows of the truth table illustrate bank selection for memory access operations. Note that the logic levels are active low, i.e. considered asserted when logic 0, for this embodiment. In the first row of the truth table, a RAS signal is asserted, with A**13** being a logic 0, causing the upper bank to be selected in the next state. Conversely, when A**13** is a logic 1, the assertion of the RAS signal selects the lower bank in the next state. When a CAS signal is asserted before a RAS signal, as shown in row **6**, all RASU and RASL signals are deasserted, and the system waits for the RAS signal to be asserted. When the RAS signal is asserted subsequent to the CAS signal, all RASU and RASL signals are driven low, and a CBR refresh cycle is performed.

Operation of one embodiment of the bank control circuit is further illustrated with a state diagram in FIG. **7**. When no memory access operations or refresh cycles are occurring, the bank control circuit is in an idle state **3001**. When the bank control circuit receives an active low RAS signal, a memory access operation is initiated. Selection of the particular bank is dependent upon the state of address signal A**13**. For example, if A**13** is a logic 0, the upper bank will be selected. The bank control circuit will transition to state **3021** and drive RASU signals to DRAM chips in the upper bank. When the DRAM chips of the upper bank receive the RAS signals, a row address is selected based on the address signals driven to them. When the active low CAS signal is asserted, the system will transition to state **3022**, where a column address is selected. The data from the upper bank is then active. At the end of the memory cycle, both the CAS and RAS signals are deasserted, and the system returns to idle state **3001**. The sequence is identical, with the exception of the state A**13**, for lower bank access.

A CBR refresh cycle will occur when a CAS signal is received before a RAS signal. When in idle state **3001**, an active low CAS signal is received, the bank control circuit transitions to state **3031**. All RASU and RASL signals will be deasserted in this state. When the bank control circuit receives an active low RAS signal, a transition to state **3032** will occur. At this state, all RASU and RASL signals are asserted, thereby selecting both banks of memory for the refresh operation. Following completion of the refresh, the RAS and CAS signals are deasserted, and the system returns to the idle state **3001**.

While the present invention has been described with reference to particular embodiments, it will be understood that the embodiments are illustrative and that the invention scope is not so limited. Any variations, modifications, additions, and improvements to the embodiments described are possible. These variations, modifications, additions, and improvements may fall within the scope of the inventions as detailed within the following claims.

What is claimed is:

1. A memory module comprising:

a printed circuit board including a connector edge adapted for insertion in a computer system;

a lower bank of memory chips mounted upon said printed circuit board;

an upper bank of memory chips mounted upon said printed circuit board;

a bank control circuit coupled to said upper and lower banks of memory chips, wherein said bank control circuit is configured to receive a row address strobe (RAS) signal and a column address strobe signal, and at least one address signal, wherein said bank control circuit is configured to selectively provide at least one corresponding RAS output signal to either said upper bank of memory chips or said lower bank of memory chips depending upon said address signal to thereby allow either said lower bank of memory chips or said upper bank of memory chips to be selectively accessed during a given memory operation;

wherein said bank control circuit includes a latch having a latch set input,

wherein said bank control circuit further includes logic configured to assert a latch set signal that is provided to said latch set input in response to said CAS signal being asserted before said RAS signal, wherein assertion of the latch set signal causes an output of said latch to be set;

wherein said bank control circuit is configured to assert said corresponding RAS output signals concurrently to both said upper bank and said lower bank in response to an assertion of said RAS signal while said output of said latch is set; and

a line driver chip, wherein said line driver chip is configured to drive electrical signals, and wherein said line driver chip is further configured to duplicate said electrical signals for driving to said upper bank of memory chips and said lower bank of memory chips.

2. The memory module as recited in claim **1**, wherein said edge connector comprises a set of electrical contact pads for conveying said electrical signals.

3. The memory module as recited in claim **2**, wherein said connector edge includes contact pads for receiving control signals, said control signals comprising at least one column address strobe (CAS) signal, at least one row address strobe (RAS) signal, and at least one write enable (WE) signal.

4. The memory module as recited in claim **3**, wherein said electrical signals include address signals, and, wherein said address signals form an address bus, and, wherein said address bus is at least 14 bits wide.

5. The memory module as recited in claim **2**, wherein said electrical signals include data signals, wherein data signals form a data path, and wherein said data path is at least 144 bits wide.

6. The memory module as recited in claim **1**, wherein said bank control circuit is configured to provide as output a plurality of RAS Upper (RASU) signals and a plurality of RAS Lower (RASL) signals.

7. The memory module as recited in claim **6**, wherein said RASU signals correspond to said upper bank of memory, and said RASL signals correspond to said lower bank of memory.

8. The memory module as recited in claim **7**, wherein a unique combination of input signals to said bank control circuit will select said lower bank of memory for memory access operations by asserting said RASL output signals.

9. The memory module as recited in claim **8**, wherein a unique combination of said input signals to said bank control circuit will select said upper bank of memory for memory access operations by asserting said RASU output signals.

10. The memory module as recited in claim **9**, wherein said bank control circuit is configured to receive a Column Address Strobe (CAS) signal, and, wherein said bank control circuit is configured such that if said CAS signal is asserted before said RAS signal, both of said memory banks will be selected for a CBR (CAS before RAS) refresh cycle by asserting said RASL signals and said RASU signals.

11. The memory module as recited in claim **6**, wherein said bank control circuit is a state machine, and wherein said CAS and RAS signals are used to toggle said bank control circuit through a predetermined sequence of states.

12. The memory module as recited in claim **1**, wherein said memory module is a dual inline memory module (DIMM).

* * * * *

APPENDIX E

United States Patent [19]

Nielsen

[11] **4,368,515**

[45] **Jan. 11, 1983**

[54] **BANK SWITCHABLE MEMORY SYSTEM**

[75] Inventor: **Carl J. Nielsen, Saratoga, Calif.**

[73] Assignee: **Atari, Inc., Sunnyvale, Calif.**

[21] Appl. No.: **261,301**

[22] Filed: **May 7, 1981**

[51] **Int. Cl.**[3] **G06F 9/30**

[52] **U.S. Cl.** **364/200**

[58] **Field of Search** ... 364/200 MS File, 900 MS File; 371/10, 11

[56] **References Cited**

U.S. PATENT DOCUMENTS

3,737,860	6/1973	Sporer .	
3,781,812	12/1973	Wymore et al.	364/200
4,042,911	8/1977	Bourke et al.	364/200
4,064,554	12/1977	Tubbs	364/200
4,118,773	10/1978	Raguin et al.	364/200
4,145,745	3/1979	De Bijl et al.	364/200
4,149,239	4/1979	Jenkins et al.	364/200
4,158,227	6/1979	Baxter et al.	364/200
4,181,933	1/1980	Benysek	364/200

OTHER PUBLICATIONS

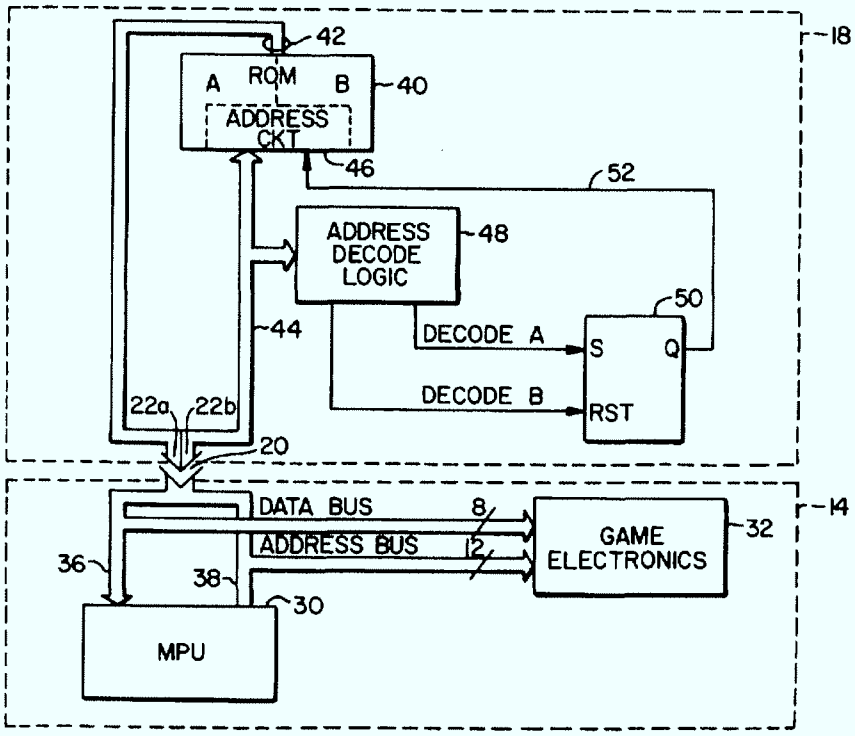
Poppendieck, M. et al., "Memory Extension Techniques for Minicomputers," Computer, May 1977, pp. 68-75.

Primary Examiner—Joseph F. Ruggiero
Assistant Examiner—Thomas M. Heckler
Attorney, Agent, or Firm—Townsend and Townsend

[57] **ABSTRACT**

A decoding circuit is coupled to the signal lines that communicate address signals to a memory unit. When a predetermined address is communicated, the decoding circuit produces a supplemental signal that is coupled to the memory unit and used to select one of a plurality of groups of memory locations. The communicated address signals specify the memory location of the selected group to be accessed.

11 Claims, 6 Drawing Figures



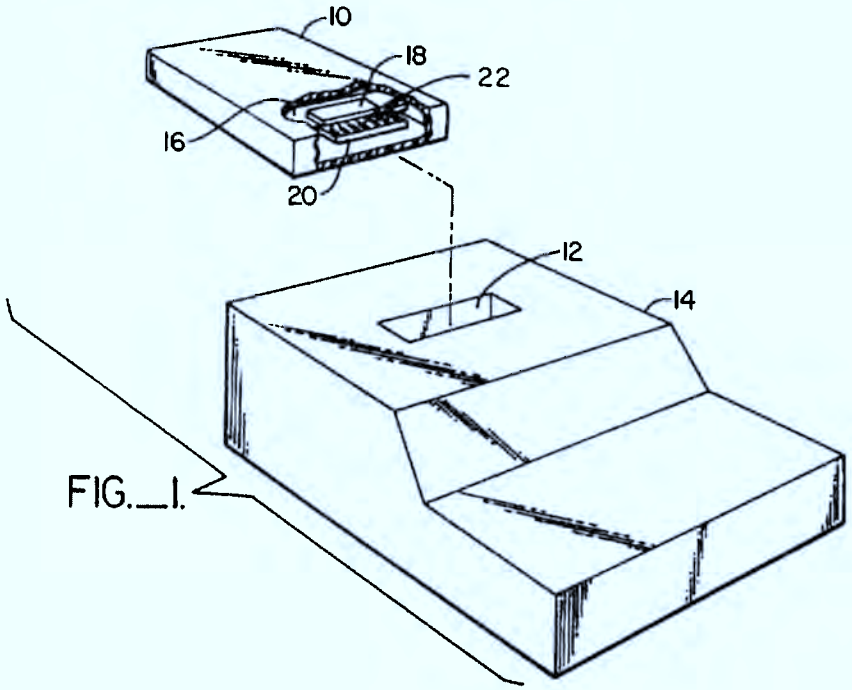

10
18
22
16
20
12
14

FIG._1.

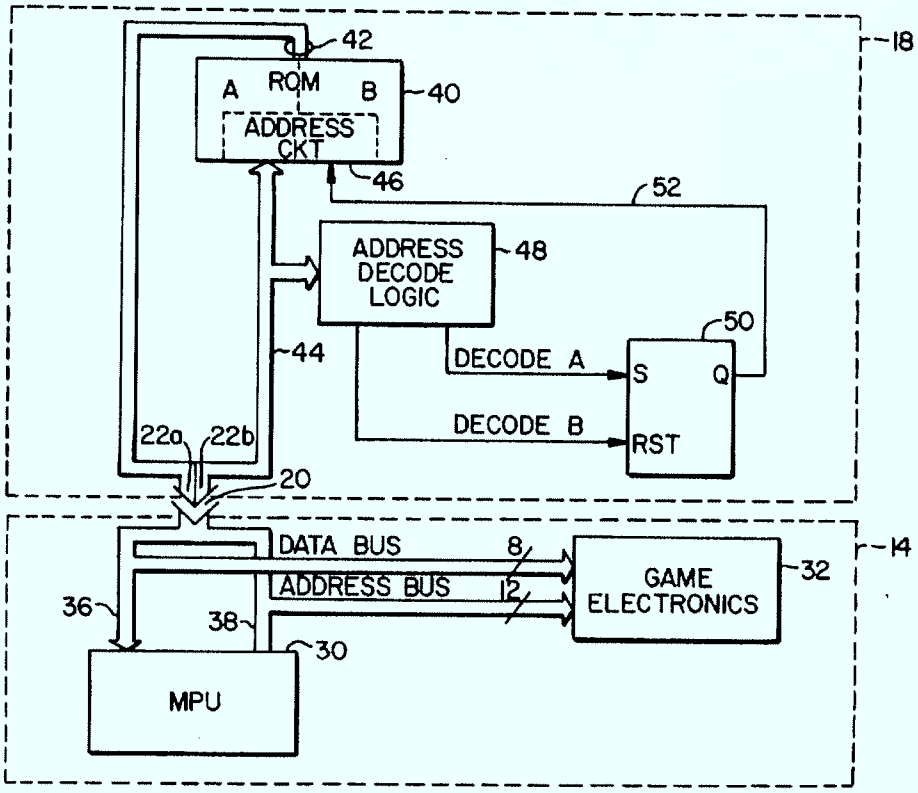

18
42
A ROM B
40
ADDRESS CKT
46
52
ADDRESS DECODE LOGIC
48
50
44
DECODE A
S
Q
DECODE B
RST
22a
22b
20
14
DATA BUS
8
ADDRESS BUS
12
GAME ELECTRONICS
32
36
38
30
MPU

FIG._2.

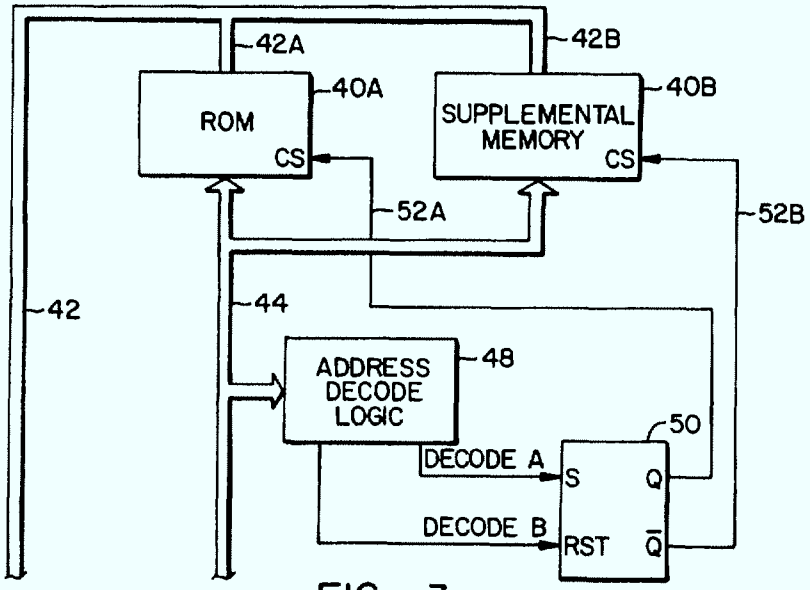

42A
42B
ROM
40A
CS
SUPPLEMENTAL MEMORY
40B
CS
52A
52B
42
44
ADDRESS DECODE LOGIC
48
50
DECODE A
S
Q
DECODE B
RST
Q̄

FIG._3.

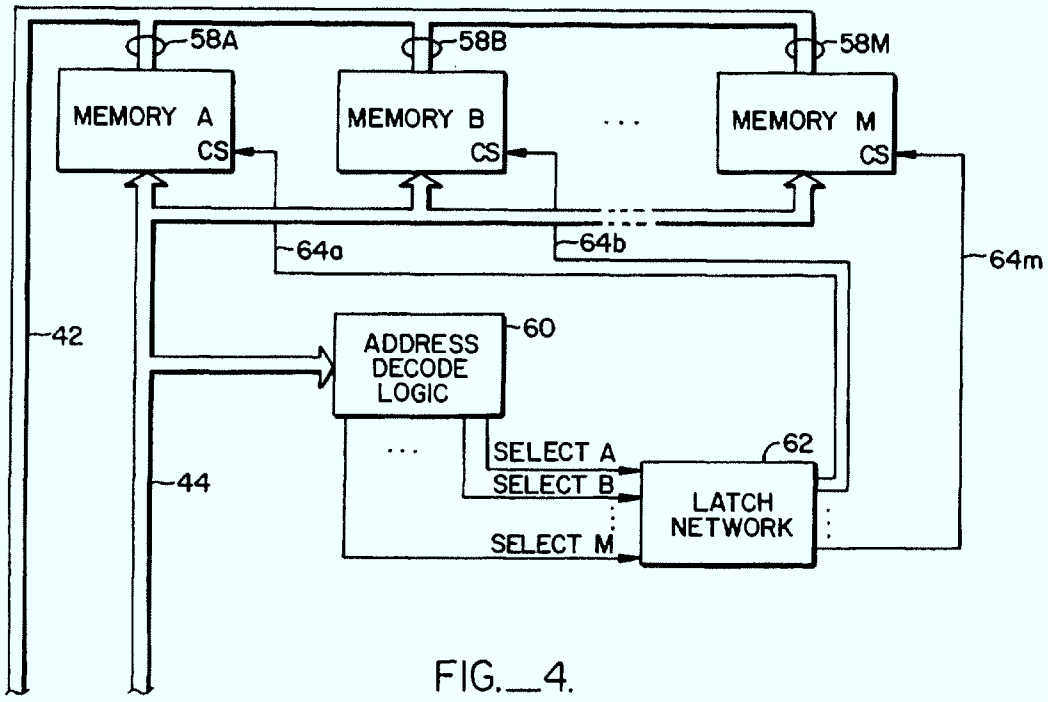

58A
58B
58M
MEMORY A
CS
MEMORY B
CS
MEMORY M
CS
64a
64b
64m
42
ADDRESS DECODE LOGIC
60
SELECT A
SELECT B
SELECT M
62
LATCH NETWORK
44

FIG._4.

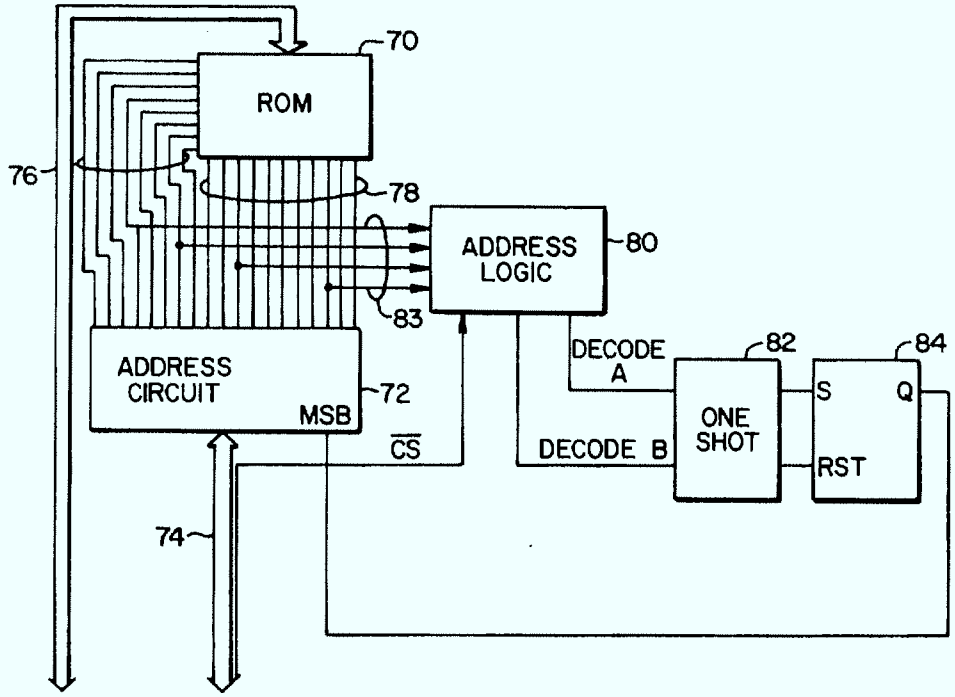

70
ROM
76
78
ADDRESS LOGIC
80
83
ADDRESS CIRCUIT
72
MSB
DECODE A
82
84
ONE SHOT
S
Q
RST
DECODE B
CS
74

FIG._5.

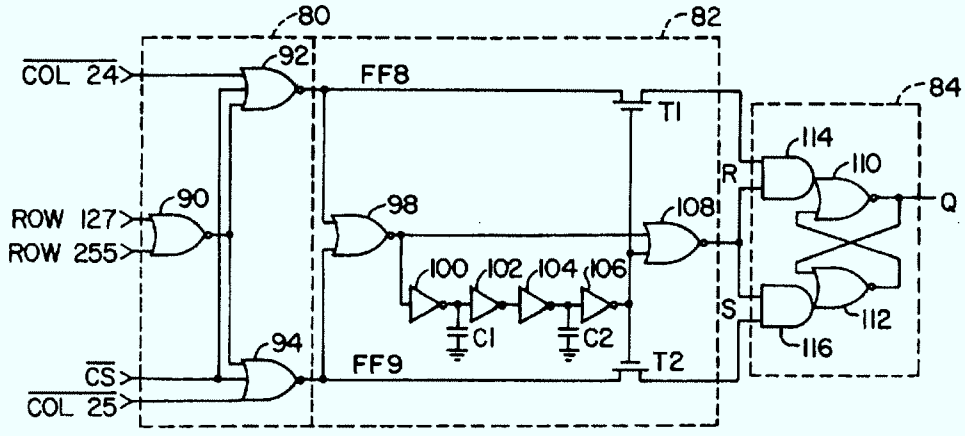

80
82
COL 24
92
FF8
T1
84
114
110
R
Q
ROW 127
ROW 255
90
98
108
100
102
104
106
C1
C2
S
112
116
94
FF9
T2
CS
COL 25

FIG._6.

BANK SWITCHABLE MEMORY SYSTEM

REFERENCE TO RELATED PATENTS

This application is generally related to subject matter of the type shown in U.S. Pat. No. 4,112,422, entitled Method and Apparatus for Generating Moving Objects on a Video Display Screen issued Sept. 5, 1978 to Mayer et al. The Mayer et al. patent describes a microprocessor and associated game console electronics for generating signals used to control the position and movement of images of objects on the display screen of a video game.

BACKGROUND OF THE INVENTION

This invention generally relates to digital systems that use a fixed number of signal lines for communicating addresses to a digital memory element, or other digital storage device having a plurality of addressable memory locations, and more particularly to an apparatus and method that increases the number of available addresses capable of being used to address the memory element.

Recent electronic advances, particularly in the digital arts, have witnessed a proliferation of a wide variety of digital systems, from large scale systems incorporating a number of processing units to consumer goods incorporating microprocessors. On the consumer side, for example, the TV game industry has for some time been producing video games for home use that incorporate microprocessors to maintain and control game play operation.

One form of such a video game currently enjoying substantial popularity today includes a console unit containing the microprocessor and other electronic circuitry that receives player input information from player manipulated elements (i.e., paddles, joysticks, and the like) and generates electronic signals that are used to drive a TV display unit. The game console is provided with a receptacle that removably receives an inexpensive cartridge. The cartridge contains an electronic microcircuit, including a read-only-memory (ROM) that stores the program of the video game to be played. With a plurality of such interchangeable cartridges, a player can program the microprocessor of the video game to execute any one of a large selection of video games.

One of the potential problems with any digital system and one which has specifically developed in the video game industry, resides in the limit of the addressable memory space of the system, i.e., the number of individually addressable memory locations which can be uniquely addressed by the processor unit. This limit is related to the number of signal lines used to make up the address bus that conducts address signals to the memory space. For example, the video game type referred to above couples a portion of the system's address bus, consisting of 12 signal lines, via appropriate wiring and a connector plug to the ROM of the microcircuit contained in the cartridge. This provides for a maximum of 2^{12} or 4,096 uniquely addressable ROM memory locations for containing the program instructions used by the microprocessor to define the video game. As experience is gained, and programming technique improves, it has become desirable to increase the number of addressable memory locations in individual cartridges. However, conventional addressing techniques are limited by

the number of address signal lines available at the game console/cartridge connector.

Accordingly, it is desirable to increase the number of addressable memory locations without changing the number of address signal lines in the current connector.

SUMMARY OF THE INVENTION

The present invention provides a bank switching memory and method for increasing the number of individual address locations that can be addressed in a digital system. The present invention expands the available memory space beyond that capable of being addressed by a conventional addressing having a unique memory location associated with a unique address. Specifically, the invention is used to expand the number of ROM memory locations contained in the game cartridge of a video game system without requiring additional address lines.

According to the present invention, supplemental address decode logic is coupled between the address bus and the memory element. The address decode logic monitors the address signals communicated on the address bus and, when a preselected address is detected, a selection signal is generated. The selection signal is applied to the address circuit of the ROM, together with the address bus, preferably as the most significant bit (MSB) of the address. This selection signal is terminated in response to the detection of another preselected address. Thus, an existing binary system having an address bus limited to N signal lines for addressing a maximum of 2^N memory locations is now capable of addressing 2^{N+1} memory locations.

In the preferred embodiment of the present invention, the address decode logic includes a flip-flop which has applied to its set/reset inputs thereof pulses generated when corresponding ones of the selected addresses are detected to latch the occurrence of the pulses. The output of the flip-flop forms the selection signal. A first predetermined address is communicated on the address bus to set the flip-flop, designating one portion of the 2^{N+1} ROM memory locations to be accessed by the signals communicated on the address bus, and a second predetermined address signal causes the flip-flop to be reset, designating the other portion of the 2^{N+1} ROM memory locations to be addressed.

The preferred embodiment of the invention, including the program ROM with which it operates, is incorporated in a single microcircuit "chip" that is mounted within the video game cartridge. By supplementing the program ROM contained in video game cartridges, existing video game apparatus can be programmed for more complex video games; existing video games can be improved to operate faster; and cartridges can be made to contain a greater number of individual games. These are merely some of the advantages that flow from increasing the memory space available for storing the microinstructions used to direct microprocessor control of the game.

However, the present invention need not be limited to increasing the number of program ROM memory locations in an alternative embodiment. The microcircuit chip housed within the cartridge can be structured to include the program ROM and a random-access-memory (RAM), the invention being used to select which memory (i.e., program ROM or RAM) will be accessed, as well as generate the read/write signal required by RAM. Thereby, an existing system can be provided with additional RAM memory as needed. As

will be seen, the concept can be expanded to provide additional addressing capability for increased program memory space.

The preferred embodiment of the invention is set forth in detail in the following description which, when read in conjunction with the accompanying drawing, will make evident additional objects, features and advantages of the present invention.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a perspective view of a game cartridge containing a microcircuit constructed in accordance with the present invention and a game console for receiving the cartridge;

FIG. 2 is a block diagram of a game console and a cartridge constructed in accordance with the present invention;

FIGS. 3, 4 and 5 are block diagrams of preferred embodiments of the invention; and

FIG. 6 is one example of a detailed schematic diagram demonstrating one implementation of portions of the circuit shown in FIG. 5.

DETAILED DESCRIPTION OF THE INVENTION

FIG. 1 is a perspective view of a commercial video game system that includes a cartridge **10** which is designed to be removably inserted in a socket **12** of a game console **14**. The cartridge is portable and contains a circuit board **16** which carries an electronic microcircuit **18**. Circuit board **16** includes a connector portion **20** with a number of printed circuit leads **22** thereon that establish electrical connection between the microcircuit **18** carried by the circuit board **16** and a connector contained within console **14**.

Referring to FIG. 2, the electronics housed within console **14** is schematically illustrated as including a microprocessor unit (MPU) **30** which functions to maintain game play control over game electronics **32**. Communication between MPU **30** and game electronics **32** is established via a conventional 8-bit data bus **36** and a 12-bit address bus **38**. Specifically, MPU **30** and game electronics **32** comprise apparatus for generating moving objects for a video game display as described in U.S. Pat. No. 4,112,422 entitled "Method and Apparatus for Generating Moving Objects on a Video Display Screen," issued Sept. 5, 1978 to Mayer et al. and incorporated by reference.

FIG. 2 also illustrates the microcircuit that is housed within cartridge **18** including a read-only-memory (ROM) **40** which contains the program instructions used to direct operation of MPU **30**. Data output lines **42** of ROM **40** are electrically connected to data bus **36** by connector leads **22***a* on connector **20**. Similarly, 12-bit address bus **38** is electrically coupled to address circuit **46** of program ROM **40** via 12 connector leads **22***b*.

The 12 address signal lines **44** are also conducted to a supplemental addressing circuit, including an address decode logic **48**. Address decode logic **48** is coupled to DECODE A and DECODE B signal lines that are connected to the set (S) and reset (RST) inputs of a flip-flop **50** respectively. The Q output of flip-flop **50** is coupled to an address circuit **46** of ROM **40** via a signal line **52**, where it combines with the 12 address signal lines **44** to become a 13th address line for addressing ROM **40**.

ROM **40** is accessed by address signals generated by MPU **30**. These address signals are conducted on signal line **44** to address circuits **46** of ROM **40** where they are supplemented with the signal conducted from the Q output of flip-flop **50** on signal line **52**. The Q output signal generated by the flip-flop **50** functions as the most significant bit (MSB) of a 13-bit address formed to designate a memory location of the ROM **40**. Stated differently, and as illustrated in FIG. 2, the supplementary signal generated by the flip-flop **50** divides the memory locations of the ROM **40** into two identifiable groups: group A and group B. Group A consists of those memory locations addressable by the address signals that are communicated on the address signal lines **44** when the Q output of the flip-flop **50** is a logic ZERO; group B consists of those memory locations that are addressed when the Q output of the flip-flop **50** is a logic ONE.

Selection between these two groups of memory locations contained in ROM **40** is effected by designating a pair of the 12 bit addresses (hereinafter address A and address B) as addresses that will cause address decode logic **48** to generate signals that are conducted on the DECODE A or DECODE B signal lines respectively. Thus, assuming that the Q output of the flip-flop **50** is a logic ZERO, the 12-bit addresses conducted to the address circuit **46** of the ROM **40** will access those memory locations associated with this condition of flip-flop **50**, that is, group A. Each 12-bit address conducted on the address signal lines **44** is also applied to the address decode logic **48** which determines whether the presently conducted address is address A or address B. If not, the address decode logic **48** remains inactive and no signals are conducted on the DECODE A and DECODE B signal lines, leaving flip-flop **50** in its present state.

If, however, it is desired to access one of the group B memory locations, address B is conducted on the address signal lines **44**. The address decode logic **48** decodes the address and issues on the DECODE B signal line a pulse that is applied to the S input of the flip-flop **50**, setting the flip-flop and causing the Q output to become a logic ONE. All 12-bit addresses conducted thereafter on the signal lines **44** will cause those memory locations associated with this present state of flip-flop **50** to be accessed, i.e., the memory locations of group B, until address B is conducted to the ROM **40**, switching flip-flop **50** and accessing the alternate group of memory locations.

Similarly, once flip-flop **50** is set to address group B memory locations, address decode logic **48** will generate a DECODE A signal in response to address A appearing on the address signal lines which will reset flip-flop **50**, set the Q output to a logical one and result in the addressing at GROUP A memory locations.

The addresses that are applied to the address circuit **46** will cause the contents of the selected memory location of the ROM **40** to appear on the data output lines **42**. From there, the contents are communicated to the data bus **36** in the game console **14** via connector leads **22***a*.

Typically, in many commercially available memory devices, both ROM and random-access-memory (RAM) are provided with a chip select (CS) pin that allows the particular memory device chip to be selected or deselected, as desired. Many embodiments of these types of memory devices incorporate the CS input in combination with tri-state data output circuitry, allowing the output lines of two or more such devices to be

connected in parallel. Referring to FIG. 3, where like elements are given the same numeral designations as those used in FIG. 2, an alternate embodiment of the present invention utilizing such memory devices is illustrated. As FIG. 3 illustrates, a ROM **40A** is supplemented with a supplemental memory **40B** (which could be either ROM or RAM), and carried by the circuit board **16** within the cartridge **10** (FIG. **1**). The data output lines **42A** and **42B** from the ROM **40A** and supplemental memory **40B**, respectively, are connected in parallel and to the data lines **42**. Address signal lines **44** are conducted to the address circuits (not shown in FIG. 3) of both the ROM **40A** and supplemental memory **40B**, and, as in FIG. 2, to the address decode logic **48**. In turn, address decode logic **48** is coupled to the S and RST inputs of the flip-flop **50** by the DECODE A and DECODE B signal lines, also as in FIG. 2. The Q and $\overline{Q}$ outputs of the flip-flop **50** are respectively connected to the CS inputs of the ROM **40A** and the supplemental memory **40B** by the signal lines **52A** and **52B**.

In the embodiment of FIG. 3, memory locations are physically divided into two physical groups: One group of memory locations resides in ROM **40A**, the other in supplemental memory **40B**. Selection between which of two devices is accessed is made (as with the embodiment of FIG. 2), by designating two addresses to be communicated to cause the address decode logic **48** to issue output signals on the DECODE A or B signal lines corresponding to the received address. For example, a predetermined address A_1 is selected to cause the flip-flop **50** to be set. In turn, the Q output of the flip-flop **50** becomes a logic ONE and the $\overline{Q}$ output a logic ZERO. The respective outputs of the flip-flop **50** are conducted to the respective CS inputs of the ROM **40A** and the supplemental memory **40B**, selecting the data from ROM **40A** to be conducted to the output lines **42A**, and deselecting the supplemental memory **40B**. Alternately, appearance of the predetermined address B on address signal lines **44** will cause the address decode logic **48** to issue a signal on the signal line DECODE B to reset the flip-flop **50**, causing the Q and $\overline{Q}$ outputs of the flip-flop to reverse their binary states, selecting supplemental memory **40B** and deselecting ROM **40A** as the accessed memory device.

It should be evident that the invention need not be limited to selecting between one of only two portions of a designated memory space. Rather, three or more predetermined addresses can be designated for selection of a corresponding number of memory location groups by expanding the address decode logic **48** and the required number of flip-flops.

FIG. **4** illustrates an expanded version of the embodiment of FIG. 3, with some modification. Here, in FIG. **4**, a number of memory devices, memories A–M, are provided, having their respective data output lines **58A–58M** connected in parallel and to the output lines **42**. The address signal lines **44** are coupled to the address circuits (not shown) of each of the memories A–M and to an address decode logic **60** that monitors the address signals communicated on the address signal lines **44**. In response to detection of one of the predetermined addresses, designated to select one of the memories A–M for access, the address decode logic **60** will generate a pulse signal that is conducted on one of the signal lines SELECT A–SELECT M to a latch network **62**. Latch network **62**, which may be in the form of a plurality of flip-flops, one for each of the signal lines SELECT A–SELECT M, temporarily stores the received signal until a different signal is received from the address decode logic **60**. The output lines ***64a–64m*** are respectively connected to the chip select (CS) inputs of the memories A–M.

In operation, address signals are conducted on the address signal lines **44** and appied to the respective memory circuits of memories A–M. Data from the memory location designated by the address will appear on that set of data output lines **58A–58M** corresponding to the memory selected by its CS input. Only one of the memories A–M will generally be selected at any one time and, therefore, a chip SELECT signal will generally only be present on one of the signal lines ***64a–64m*** at any moment in time. Selection of the particular memory A–M is effected basically as described with respect to FIGS. 2 and 3: predetermined addresses corresponding to the memories A–M are conducted on the signal lines **44** to cause the address decode logic to issue a pulse on one of the SELECT A–SELECT M signal lines. The generated pulse is received by the latch network **62**, correspondingly causing a chip select to be conducted on only one of the signal lines ***64a–64m*** to the CS input of the memory A–M corresponding to the decoded predetermined address.

It is well known in this art that, due to propagation delays and other factors inherent in electronic circuitry, changes in the address signal received by the address decode logic **48** (FIGS. 2 and 3) or the address decode logic **60** (FIG. **4**) do not change simultaneously. That is, the changes of state that occur on the individual ones of the signal lines may lead or lag one another so that during such transitions, addresses may momentarily appear that are not intended. Accordingly, some provision must be made in order to prevent the address decode logic in question from erroneously reacting to these transitional signals to inadvertently cause unwanted memory locations to be accessed. One method of preventing such erroneous action is to make the decoding process synchronous; that is, for example, the DECODE A–DECODE B (FIGS. 2 and 3) or the SELECT A–SELECT M (FIG. **4**) signal could be gated by a clock signal. However, this would necessitate a clock signal line for communicating the clock signal to the respective address decode logic. Thus, FIG. **5** illustrates yet another way of preventing such spurious signals.

FIG. **5** is a block diagram of yet another preferred embodiment of the present invention, illustrating a ROM array **70** (i.e., the array of memory locations) that receives specific, decoded row and column signals from an address circuit **72** via row and column signal lines **76, 78**, respectively. The signals generated by the address circuit **72** select the specific memory location of the ROM array **70** that is designated by the address signals communicated to the address circuit on address signal lines **74**.

Address logic **80** receives a selected number of the row and column signal lines **76, 78** via the signal line group **83** for decoding the preselected addresses. DECODE A and DECODE B signals are generated by the address logic **80** in response to detecting the preselected addresses which indicate a switch from one group or "bank" of memory locations of the ROM array **70** to another. A monostable multivibrator device or "one-shot" **82** receives the DECODE A and DECODE B signals from the address logic **80** and prevents spurious transients on the address line **76, 78** from causing an inadvertent switchover by requiring the DECODE A

and DECODE B signals to stabilize before applying them to the set (S) and reset (RST) inputs of a flip-flop **84**, respectively. The Q output of the flip-flop **84** is in this embodiment coupled to the address circuit **72** as the most significant bit (MSB) of the address applied thereto.

Turning now to FIG. **6**, the detailed schematic diagram of the address logic **80**, oneshot **82**, and flip-flop **84** of FIG. **5** is illustrated. As shown, the address logic **80** takes advantage of the initial decoding performed by the address circuit **72** (FIG. **5**). Here, the addresses (in hexadecimal) FF8 and FF9 are selected as the predetermined addresses used to set or reset the flip-flop **84**. As FIG. **6** indicates, the predetermined hexidecimal address FF8 will, after initial decoding by the address circuit **72**, correspond to activation of the ROW **127** or ROW **255** and COLUMN (COL) **24** signals. Similarly, active ROW **127** or ROW **255** and COL **25** signals will correspond to the predetermined hexadecimal address FF9. In addition, a chip select ($\overline{SC}$) signal is used here to designate selection of the ROM array **70** for access, as opposed to other memory elements (not shown) of the system incorporating the invention. In this example the $\overline{CS}$ signal is an active LOW, i.e., a logic ZERO designates selection of the ROM array **70**.

FIG. **6** shows the address logic **80** as including a two input NOR gate **90** for receiving the ROW **127** and ROW **255** signals from the ROW signal line **76** (FIG. **5**), and performs an ORING function on these signals. Three input NOR gates **92** and **94** function as AND gates. The NOR gate **92** performs an ANDING of the signal produced by the NOR gate **90** and $\overline{COL\ 24}$; the NOR gate **94** ANDS the output of the NOR gate **90** with the $\overline{COL\ 25}$ signal. The $\overline{CS}$ signal functions to enable the NOR gates **92** and **94**. The output of the NOR gates **92** and **94** are signal lines FF8 and FF9, respectively, designating recognition of either the hexadecimal address FF8 or FF9 communicated on the address signal lines **74**.

The two signal lines FF8 and FF9 are both applied to a NOR gate **98**, which forms the input stage of the oneshot **82**. The output of the NOR gate **98** is applied to a delay network comprising four inverters, **100**, **102**, **104** and **106** and capacitors C1 and C2, and to a NOR gate **108**. The output of the last inverter **106** of the delay network is also applied to the NOR gate **108**, as well as the control leads of transfer switches T1 and T2. If the signal produced by the NOR gate **98** remains present for a sufficient length of time, determined by the time for the signal to propagate through the delay provided by the inverters **100–106** and capacitors C1 and C2, the signal is considered valid and NOR gate **108** is activated. At the same time the transfer switches are turned off, so the decoded address signal stored on R or S by signal line FF8 and FF9 sets or resets the flip-flop **84**.

The flip-flop **84** is shown as including a pair of cross-coupled NOR gates **110** and **112** which form the bistable or latching portion of the flip-flop. An AND gate **114** forms a gated reset input of the flip-flop **84** and an AND gate **116** forms the gated set input of the flip-flop. The signals produced by the AND gates **114** and **116** respectively reset or set the latching portion (NOR gate **110** and **112**) of the flip-flop, causing the Q output of the flip-flop to assume a logic ZERO or a logic ONE, as the case may be.

The embodiment of FIGS. **5** and **6** operates as follows: Address signals are continually being formed and communicated on the signal lines **74** to the address circuit **72**. The selected decodes (i.e., $\overline{COL\ 24}$, $\overline{COL\ 25}$, ROW **127**, and ROW **255**) produced by the address circuit **72** are applied to the NOR gate **90**, **92**, and **94** of the address logic **80**. If, at any moment in time, the address signals appearing on signal lines **74** form either of the predetermined addresses FF8 or FF9, and $\overline{CS}$ is a logic ZERO (selecting ROM array **70** for access), a corresponding signal will appear on one of the signal lines FF8 or FF9. This corresponding signal is applied by the signal line FF8 or FF9 to the NOR gate **98** and, after a certain delay the signal propagates through the inverters **100–106** to switch the NOR gate **108** and transfer switches T1 and T2. If the corresponding signal is still present on the signal line FF8 or FF9 at the time the transfer switches T1 and T2 are switched, the signal will be passed to the R or S input of the flip-flop **84**; at the same time, the AND gates **114** and **116** are enabled by the signal produced by the NOR gate **108** and the flip-flop is thereby reset or set, as the case may be.

On the other hand, if the address signal decoded by the NOR gates **92** or **94** is merely a transient, resulting from a transition from one address to another, the signal will not be present when the transfer switches T1 and T2 are switched, and the NOR gate **108** will be found to be disabled when the propagation time expires. Accordingly, the AND gates **114** and **116** of the flip-flop **84** remain disabled and the state of the flip-flop will be left unchanged.

When switching from one group of memory locations to another, using the present invention, it should be evident that two memory locations are accessed by each predetermined "switching" address. For example, in FIGS. **5** and **6** the hexadecimal address FF8 will access the two memory locations: one in that group defined when the Q output of the Flip-flop **84** is a logic ONE, and one defined when the Q output is a logic ZERO. Accordingly, in the preferred embodiment, those memory locations specified by the hexadecimal address signals FF8 and FF9, regardless of the state of the MSB input to the address circuit **72** contain a NO-OPERATION (NOP) instruction or designation.

When the invention is used to expand the memory capacity of that portion of a video game system that is resident in the cartridge **10** (FIG. **1**), the circuitry, including the memory, is fabricated as a single microcircuit chip. Thus, for example, the ROM array **70**, address circuit **72**, address logic **80**, Oneshot **82**, and flop-flop **84** are preferably fabricated as a single integrated circuit chip and packaged in a conventional dual-in-line package (DIP). The package is configured as an 8-bit×8K memory (64K bits) having a pin configuration that is identical to the 8-bit×4K memory package (32K bits) presently carried by existing cartridges. Thus, no changes are needed in the cartridge **10** (FIG. **1**) in order to convert a present 32K system to a 64K system.

Thus, it will be seen that the invention provides for greatly increasing the number of individual addresses that can be generated by a digital system having an address bus for communicating those addresses limited to N individual signal lines, correspondingly expanding the available memory space of the system. The present invention provides for a significant increase in memory space of a digital system without extensive modifications to the system. Thereby, a substantial increase in available memory space is obtained at very little cost and effort. Although several embodiments of the invention have been shown and described by way of example, it will be obvious that other adaptations and modifica-

tions can be made without departing from the true spirit and scope of the invention.

I claim:

1. A memory system comprising:

an address bus for providing a plurality of first address signals;

a supplemental address line;

first memory means coupled to the address bus and to the supplemental address line and having a plurality of memory locations for providing digital signals corresponding to data stored in selected memory locations in response to presence of a second address signal on the supplemental address line, the memory locations being selected in response to first address signals on the address bus;

second memory means coupled to the address bus and to the supplemental address line having a plurality of memory locations for providing digital signals corresponding to data stored in selected memory locations in response to absence of the second address signal on the supplemental address line, the memory locations selected in response to the first address signals on the address bus; and

decoder means coupled to the address bus and to the supplemental address line for providing the second address signal on the supplemental address line in response to detecting a first combination of address signals on the address bus.

2. A memory system as in claim 1 wherein the decoder means terminates the second address signal on the supplemental address line in response to detecting a second combination address signals on the address bus.

3. In a digital system, including a memory means having a plurality of memory locations addressable by an address circuit and an address bus coupled to said address circuit for communicating address signals thereto, supplemental address generating apparatus comprising:

detecting means coupled to said address bus for detecting communication of predetermined ones of said address signals, including means for generating one of a number of supplemental address signals selected in response to detection of a corresponding one of said predetermined address signals; and

means coupling the detecting means to the address circuit of the memory means for conducting the supplemental address signals thereto, the address circuit combining the address signals communicated on the address bus with the supplemental address signals to designate a one of the plurality of memory locations.

4. The supplementary address generating apparatus of claim 3, wherein the memory means comprises a plurality of individual memory elements each having a selection input for receiving a memory element selection signal, and wherein at least a portion of said supplemental address signals is coupled to the selection input of each memory element.

5. The supplementary address generating apparatus of claim 3, wherein the detecting means includes latch means for generating the supplemental address signal.

6. The supplementary address generating apparatus of claim 5, wherein the detecting means includes decoding means interconnecting the address bus and the latch means for generating a pulse signal in response to detection of each one of the predetermined address signals on the address bus, thereby causing the latch means to generate the supplemental address signals.

7. A memory system comprising:

an address bus for providing a plurality of address signals;

a supplemental address line;

a memory array having a plurality of row and column lines and having a plurality of memory locations corresponding to different row and column lines for providing digital signals responsive to data stored in selected memory locations, a distinct memory location being selected in response to signals on each unique row and column line pair;

an address circuit coupled to the address bus, the supplemental address line and to the row and column lines for providing signals on row and column line pairs selected in response to address signals on the address bus and the supplemental address line, a distinct row and column line pair selected in response to each unique combination of address signals on the address bus and supplemental address line;

address logic coupled to the row and column lines for providing a reset signal in response to signals appearing on a first row and column line pair and for providing a set signal in response to signals appearing on a second row and column line pair; and

latch means coupled to the address logic and to the supplemental address line for providing an address signal on the supplemental address line in response to the set signal and for terminating the address signal in response to the reset signal.

8. A memory system as in claim 7 wherein the address logic includes means for logically ORing a pair of the lines from the address circuit such that the selection of either of a first two memory locations will result in a set signal, and the selection of either of a second two memory locations will result in a reset signal, one of the first two memory locations and one of the second two memory locations selected only in response to an address signal on the supplemental address line, the other of the first and the second two memory locations selected only in response to no address signal appearing on the supplemental address line.

9. The memory system of claim 8, wherein the data stored in the memory array includes a plurality of multibit instruction words, and wherein each of the first two and the second two memory locations contains a no-operation instruction.

10. The memory system of claim 7, wherein the address logic includes means for generating the set and the reset signals in response to signals appearing on the first row and column line pair for a time period substantially longer than the period of time needed to detect the signal.

11. In a digital system, including an address bus having N signal lines for conducting address signals and a digital memory coupled to the address bus and having 2^{N+M} memory locations for storing a plurality of data words, the digital memory including a memory address circuit having at least N+M signal inputs, N of said signal inputs being coupled to corresponding ones of the signal lines of the address bus, apparatus for supplementing the address signals conducted on the address bus to selectively address each of said 2^{N+M} memory locations, the apparatus comprising:

decoder means coupled to the address bus for detecting communication thereon of a predetermined number of said address signals, the decoder means including output means for providing supplemental

address signals in response to detection of communication of corresponding ones of the predetermined address signals on the address bus; and

means coupling the decoder output means to the remaining M inputs of the memory address circuit for communicating the supplemental address signals thereto, the memory address circuit being operable to select a one of the memory locations designated by the combination of address signals conducted on the address bus and the supplemental address signals provided by the decoder output means.

* * * * *

APPENDIX F

US 20060117152A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2006/0117152 A1**

Amidi et al. (43) **Pub. Date: Jun. 1, 2006**

(54) **TRANSPARENT FOUR RANK MEMORY MODULE FOR STANDARD TWO RANK SUB-SYSTEMS**

(75) Inventors: **Hossein Amidi**, Lake Forest, CA (US); **Kelvin A. Marino**, Laguna Hills, CA (US); **Satyadey Kolli**, Milpitas, CA (US)

Correspondence Address:
Robert E. Krebs
Thelen Reid & Priest LLP
P. O. Box 640640
San Jose, CA 95164-0640 (US)

(73) Assignee: **SMART Modular Technologies Inc., a California Corporation**

(21) Appl. No.: **10/752,151**

(22) Filed: **Jan. 5, 2004**

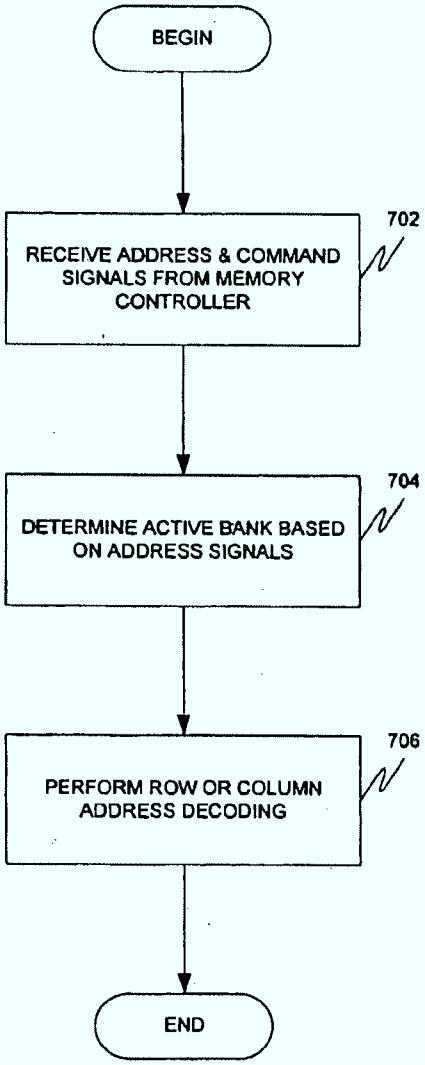
Publication Classification

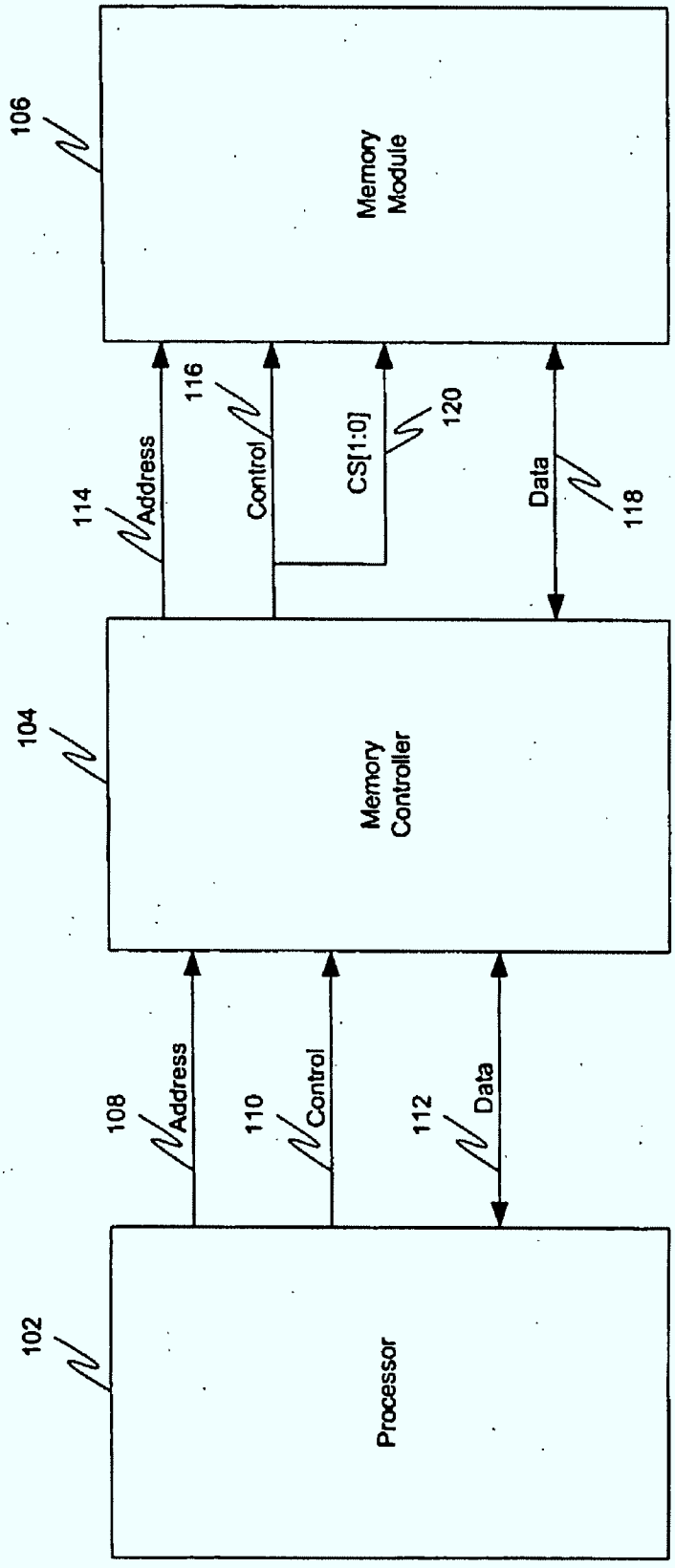
(51) **Int. Cl.**
G06F 13/00 (2006.01)

(52) **U.S. Cl.** **711/154**

(57) **ABSTRACT**

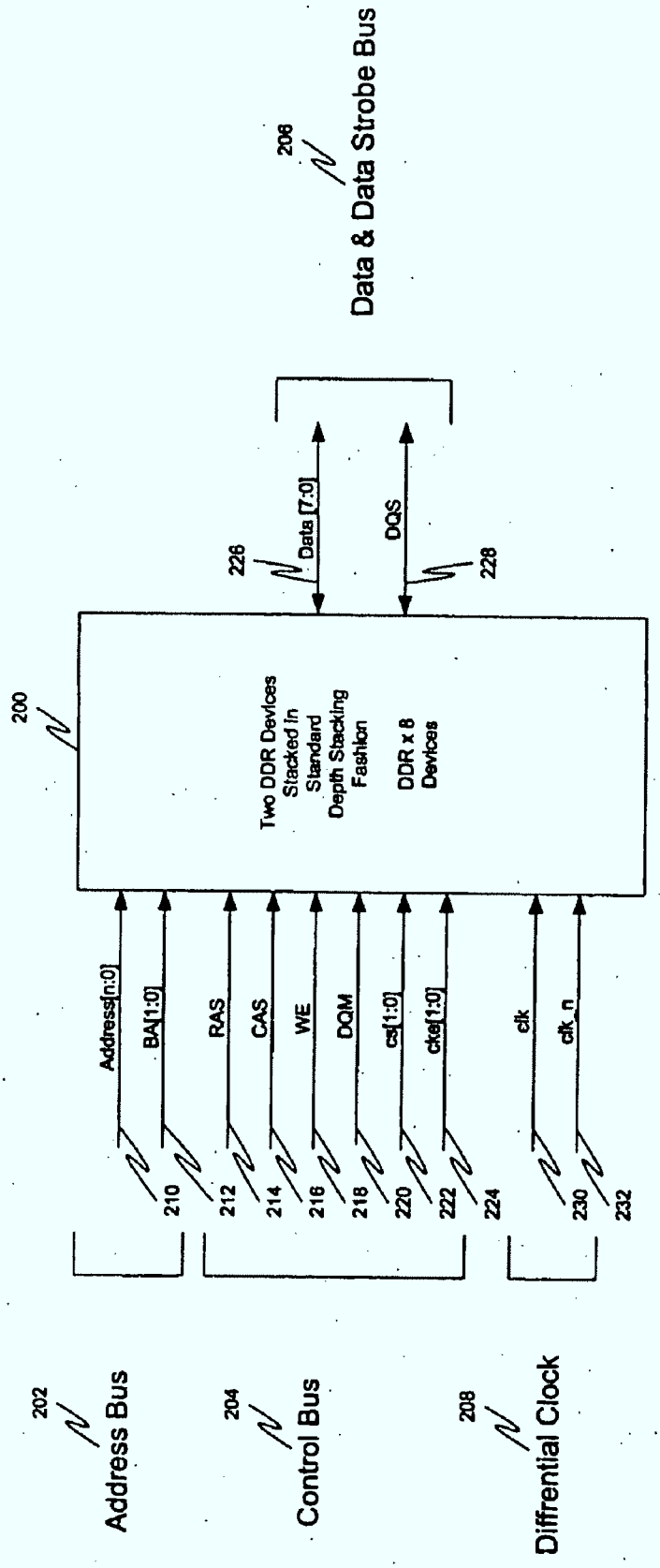
A transparent four rank memory module has a front side and a back side. The front side has a third memory rank stacked on a first memory rank. The back side has a fourth memory rank stacked on a second memory rank. An emulator coupled to the memory module activates and controls one individual memory rank from either the first memory rank, the second memory rank, the third memory rank, or the fourth memory rank based on the signals received from a memory controller.





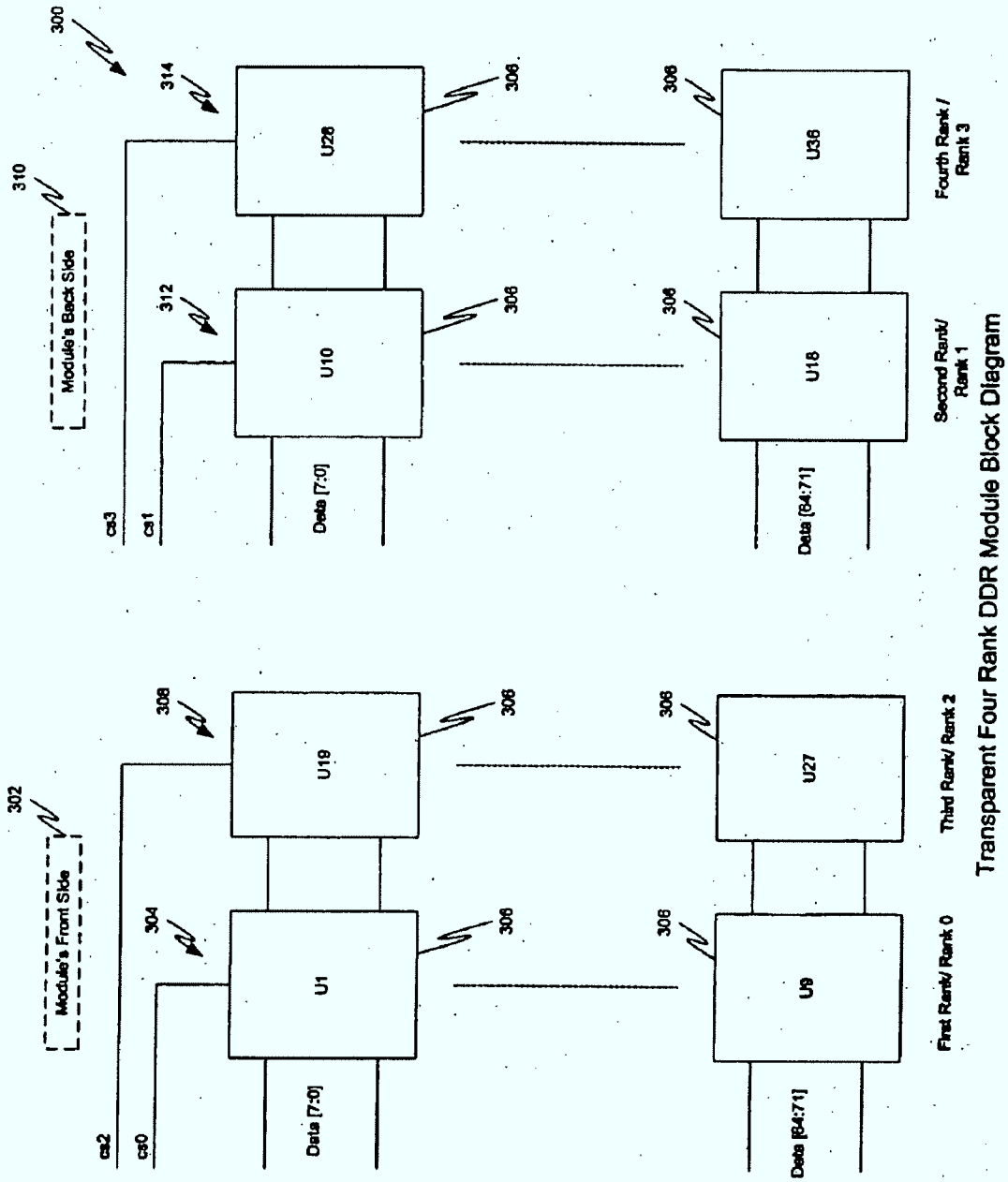
Overall Block Diagram of Standard Memory Interface

FIG. 1 - PRIOR ART



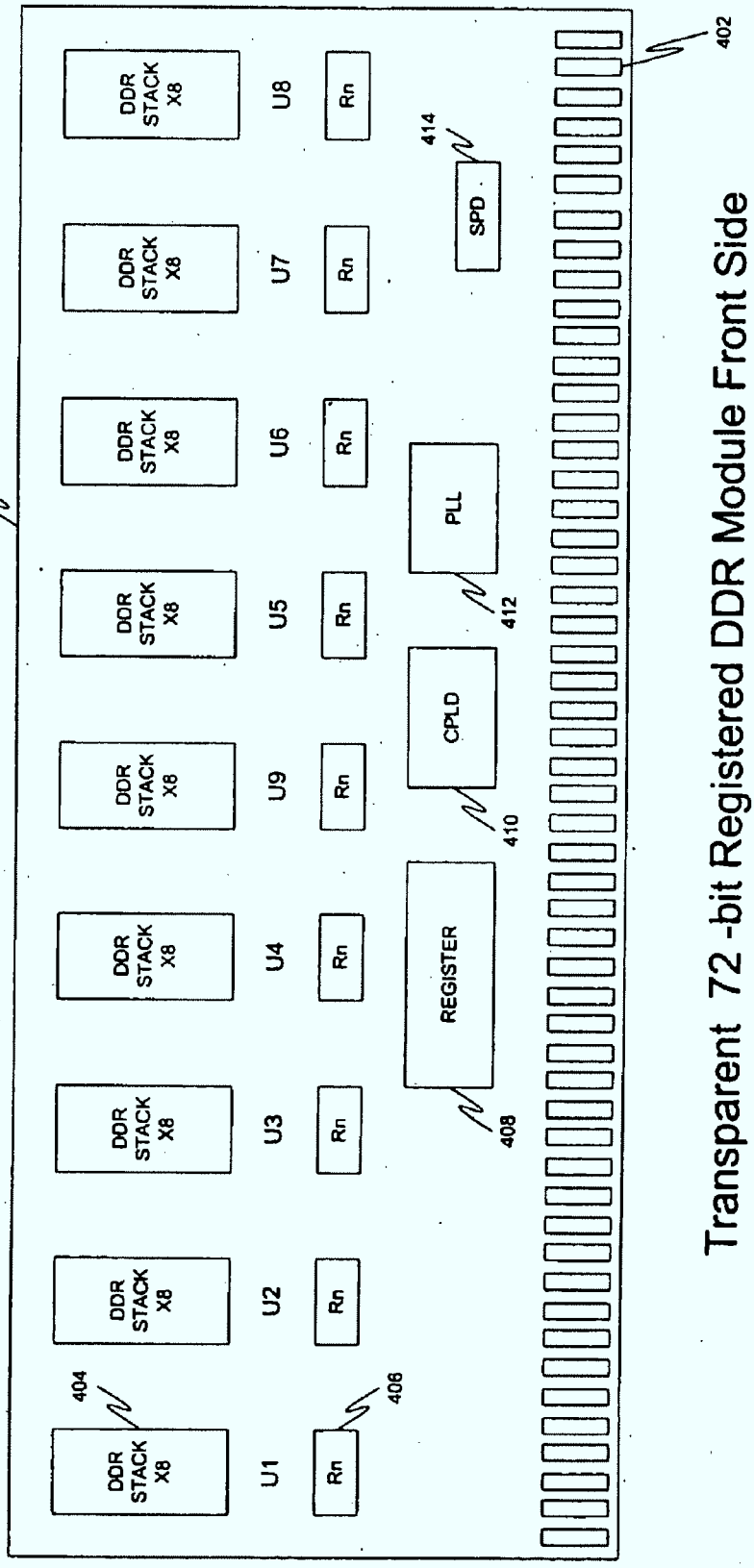
Standard DDR Device Depth Stacking Block Diagram

FIG. 2

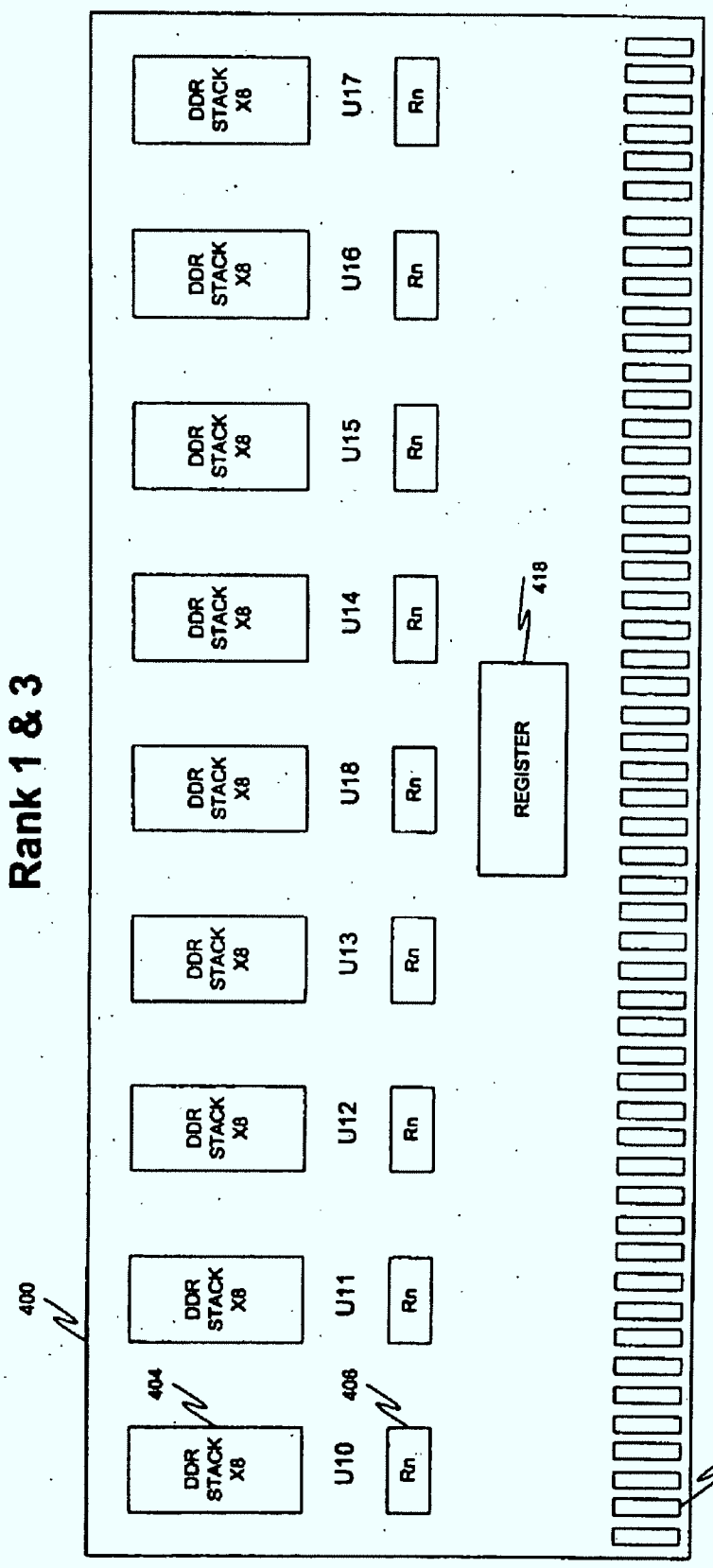


Transparent Four Rank DDR Module Block Diagram

FIG. 3



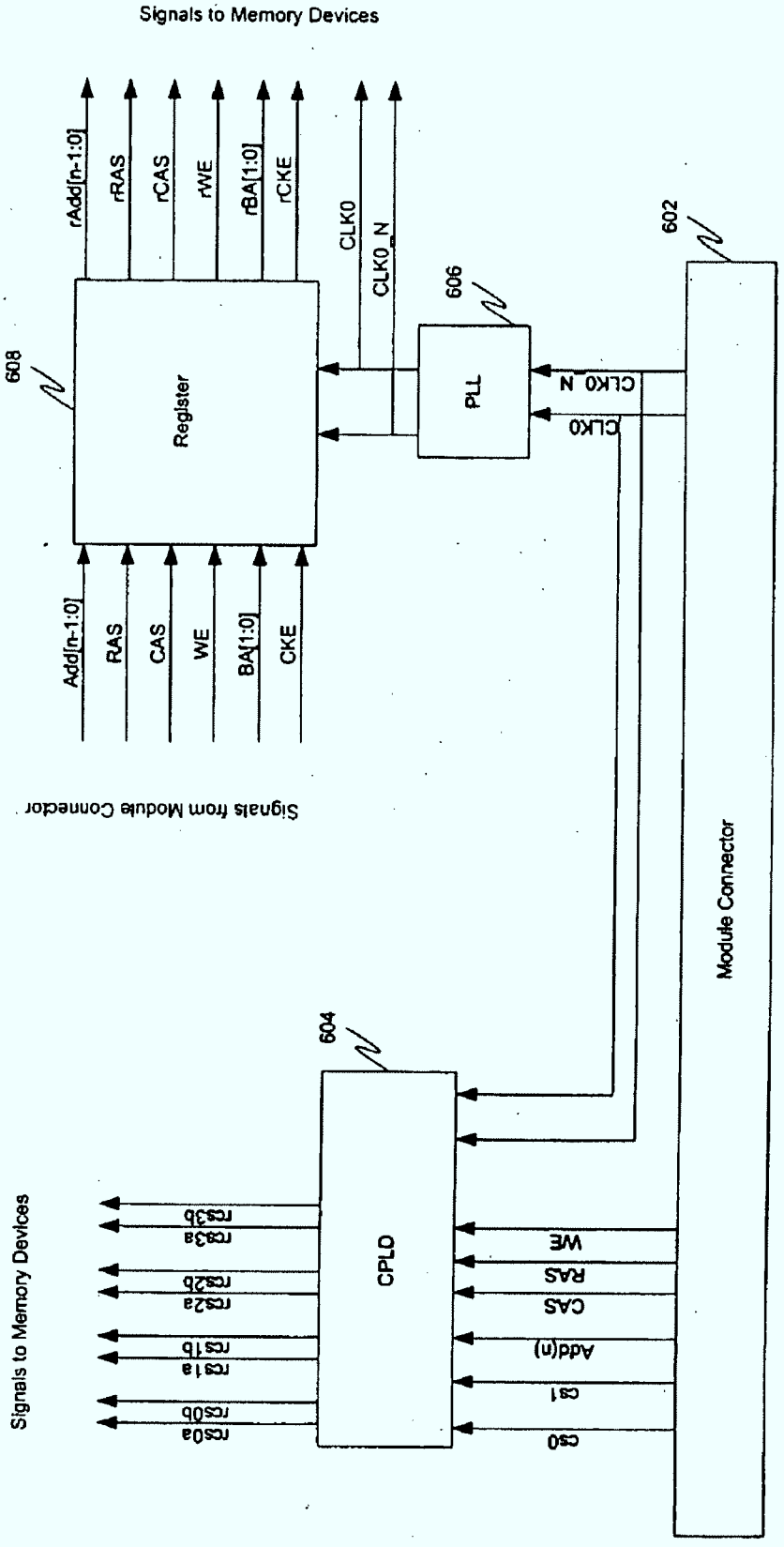
Transparent 72 -bit Registered DDR Module Front Side
FIG. 4A



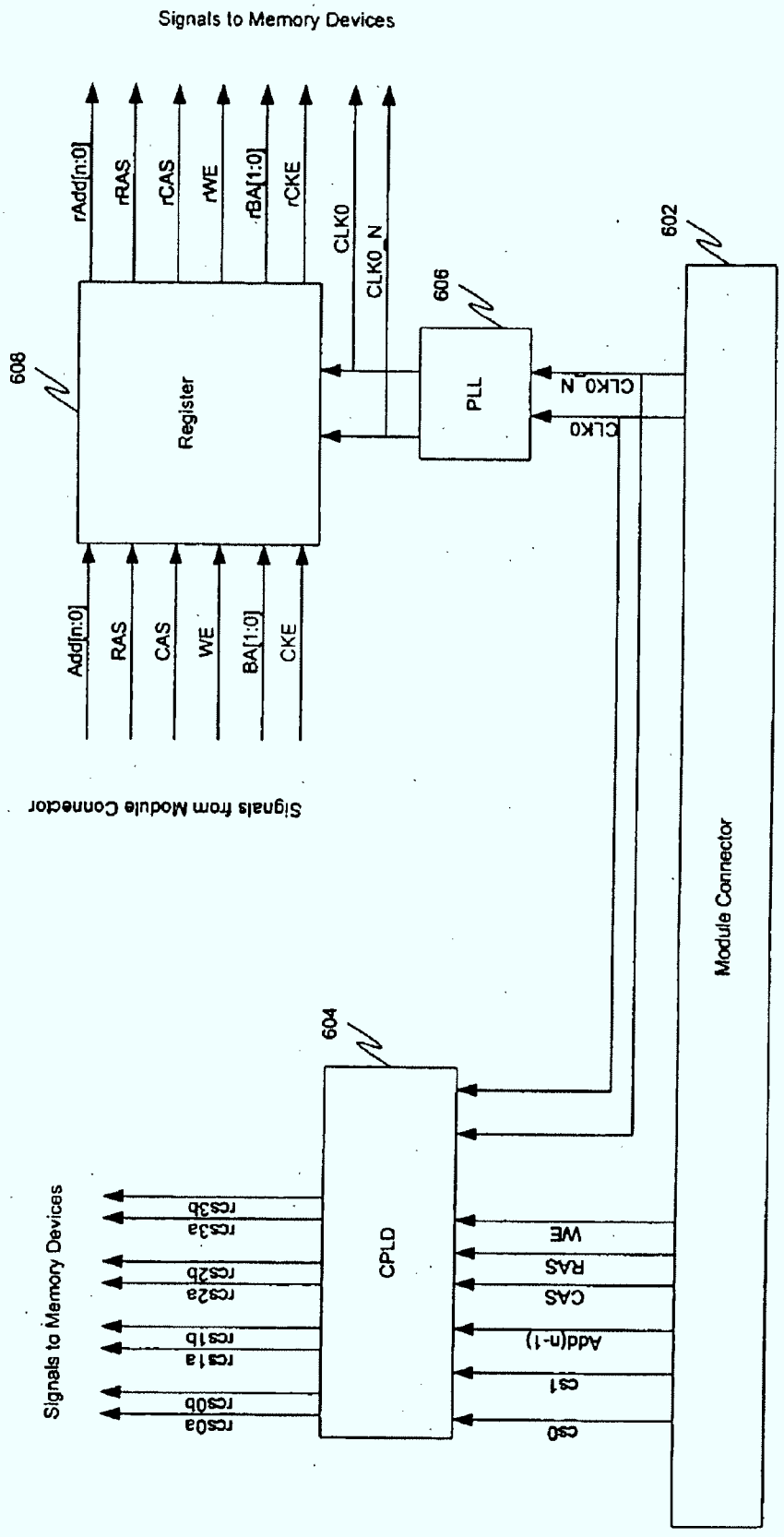
Transparent 72 -bit Registered DDR Module Back Side
FIG.4B

Add(n)	CS1	CS0	Active Bank
0	1	0	0
0	0	1	1
1	1	0	2
1	0	1	3

Truth Table
FIG.5



Row Address Decoding
FIG.6A



Column Address Decoding
FIG.6B

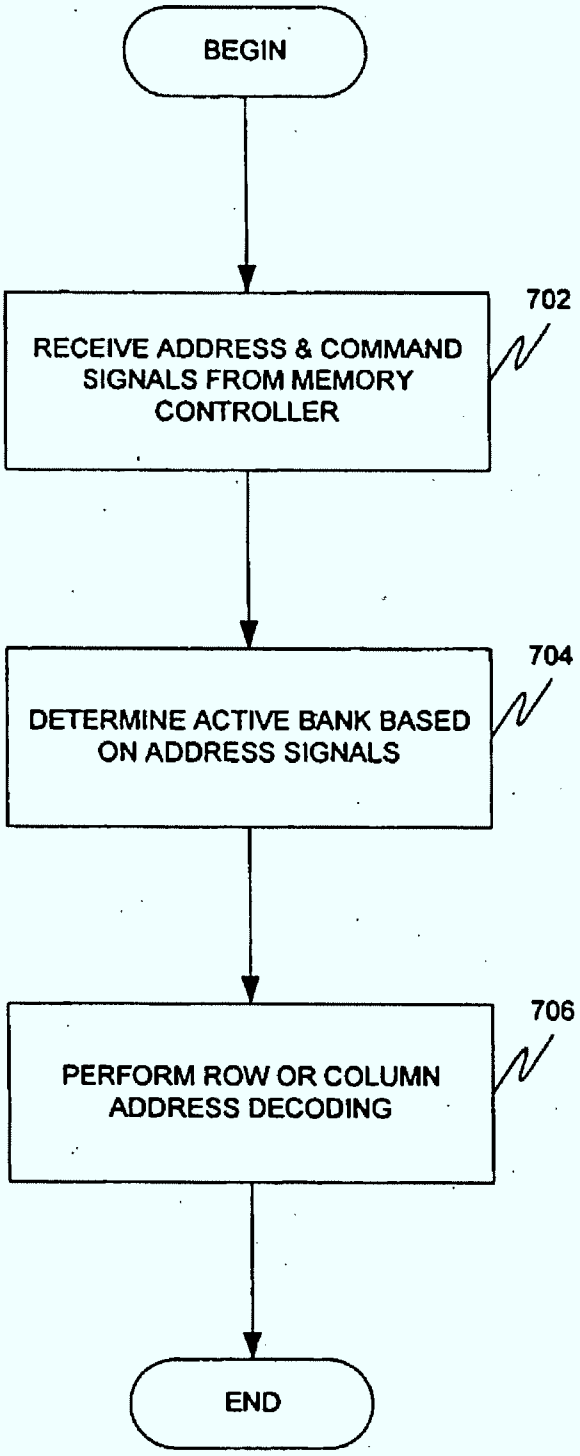
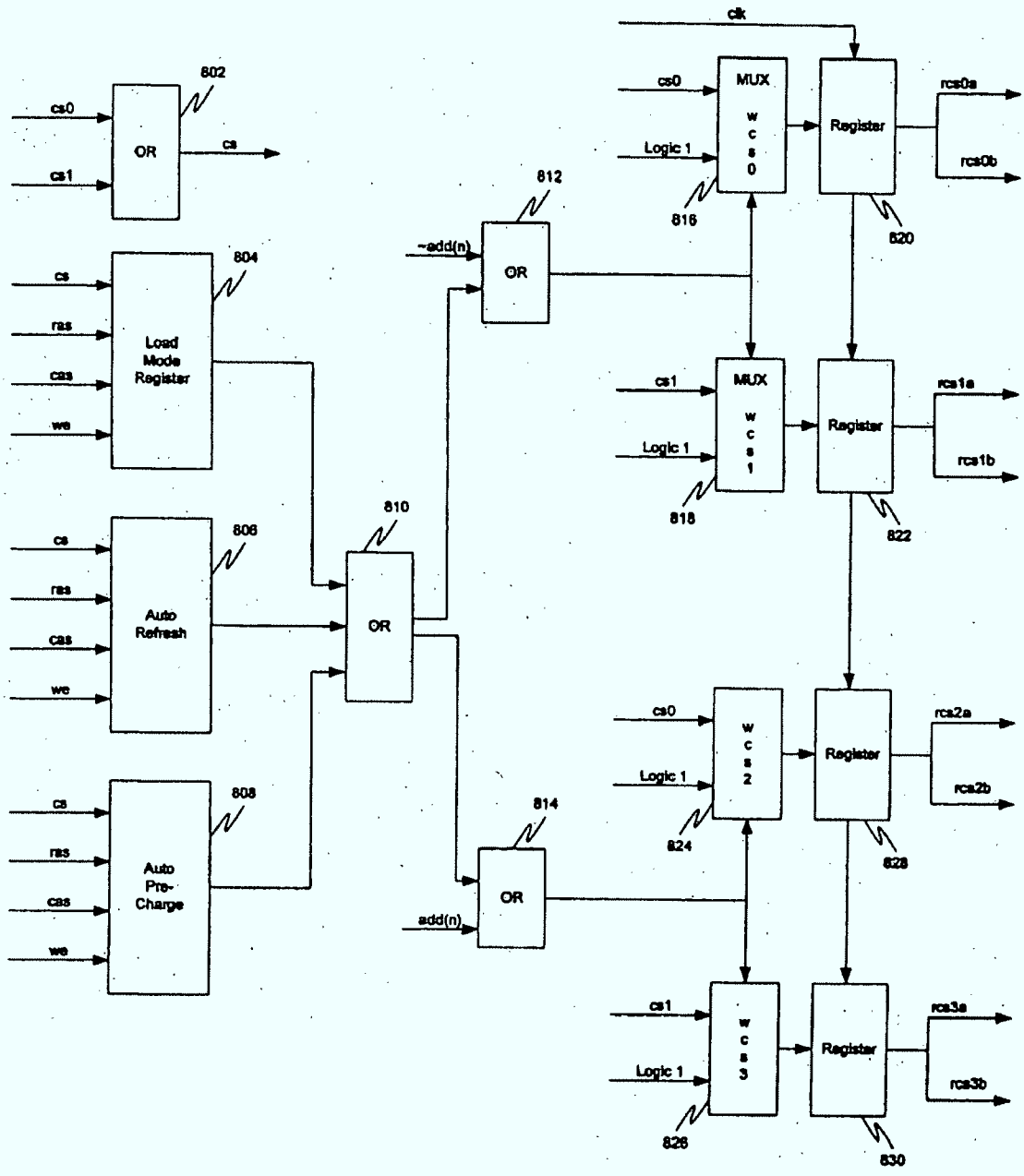

BEGIN
RECEIVE ADDRESS & COMMAND SIGNALS FROM MEMORY CONTROLLER
702
DETERMINE ACTIVE BANK BASED ON ADDRESS SIGNALS
704
PERFORM ROW OR COLUMN ADDRESS DECODING
706
END

FIG.7



Bank EmulatorBlock Diagram
FIG.8

TRANSPARENT FOUR RANK MEMORY MODULE FOR STANDARD TWO RANK SUB-SYSTEMS

FIELD OF THE INVENTION

[0001] The present invention relates to computer memory. More particularly, the present invention relates to a four rank memory module.

BACKGROUND OF THE INVENTION

[0002] Computers use memory devices for the storage and retrieval of information. These memory devices are often mounted on a memory module to expand the memory capacity of a computer. Sockets on a main board accommodate those memory modules also known as SIMMs or DIMMs.

[0003] **FIG. 1** is a diagram schematically illustrating a standard memory interface system overview in accordance with a prior art. The system **100** includes a processor **102**, a memory controller **104**, and a memory module **106**. The processor **102** communicates with the memory controller **104** with an address bus **108**, a control signal bus **110**, and a data bus **112**. The memory controller **104** communicates with the memory module **106** with a controller address bus **114**, a controller control signal bus **116**, and a controller data bus **118**. Common system implementations **100** have typically two memory chip selects routed per socket. Common memory module **106** may have two chip selects (one per rank) or four chip selects (two per rank). In one implementation, each chip select from the controller is connected to the corresponding chip select on the memory module. In the second implementation, each chip select from the controller is connected to the two chip selects (those that control one rank) on the memory module. The system chip select signals control individual memory modules ranks. The memory module **106** is coupled to the memory controller **104** through a memory socket.

[0004] Standard memory modules such as memory module **106** have either one rank or two rank of memory devices. Each memory device comes in a variety of configurations and families such as 128 Mbit, 256 Mbit, 512 Mbit, and 1024 Mbit DDR SDRAM families. Each of these families is further divided into three distinct flavors such as x4, x8, and x16 data bits. For example, a single 128 Mbit DDR SDRAM family comes in three flavors of:

32 M×4 (32 Mega cell of 4-bit each=32M×4-bit=128 Mbit)

16 M×8 (16 Mega cell of 8-bit each=16M×8-bit=128 Mbit)

8M×16 (8 Mega cell of 16-bit each=8M×16-bit=128 Mbit)

[0005] The example above illustrates that all three different data bits flavors result in the same density of 128 Mbit. As the number of data bits doubles the cell numbers decrease by half. One can build memory modules with similar densities using different data bits flavors.

[0006] One method of building a 512 M Byte standard memory module with ECC (64-bit data plus 8-bit ECC=72-bit) includes using 256 Mbit density families of 32M×**8** to achieve the density of 512 M Byte as follow:

Rank **0**=9×(32M×8) devices=32M×72-bit which equates to 32M×8 Bytes+1 Byte of ECC. This yields a total density of 32M×8 Bytes=256M Byte.

Rank **1**=9×(32M×8) devices=32M×72-bit which equates to 32M×8 Bytes+1 Byte of ECC. This yields a total density of 32M×8 Bytes=256M Byte.

[0007] Therefore, a two rank memory module with 18 device placements will achieve the 512M Byte density. Furthermore, it should be noted that a standard DDR 184-pin memory module can only fit nine TSSOP placements per side, or a total of 18 placements of TSSOP per module, considering both front and back sides based on a standard defined height limits by JEDEC.

[0008] Because memory devices with lower densities are cheaper and more readily available, it may be advantageous to build the above same density memory module using lower densities devices. However, in order to achieve a density of, for example, 512 M Bytes using 128 Mbit density of 16M×8 instead, the memory module needs four ranks configured as follows:

Rank **0**=9×(16M×8) devices=16M×72-bit which equates to 16M×8 Bytes+1 of ECC. This would give us a total density of 16M×8 Bytes=128 M Byte.

Rank **1**=9×(16M×8) devices=16M×72-bit which equates to 16M×8 Bytes+1 Byte of ECC. This would give us a total density of 16M×8 Bytes=128 M Byte.

Rank **2**=9×(16M×8) devices=16M×72-bit which equates to 16M×8 Bytes+1 Byte of ECC. This would give us a total density of 16M×8 Bytes=128 M Byte.

Rank **3**=9×(16M×8) devices=16M×72-bit which equates to 16M×8 Bytes+1 Byte of ECC. This would give us a total density of 16M×8 Bytes=128 M Byte.

[0009] In order to achieve the above configuration, 4 rows of 9 devices each, totaling 36 placements, are required. As mentioned above, on a standard 184-pin DDR memory module, there is only enough space for 18 TSSOP devices.

[0010] The only solution would be, to stack two memory devices together to achieve an extra rank on the same placement space. Although this would solve the placement problem of 36 TSSOP devices, the memory module would still possess four memory ranks. As explained earlier, all standard memory modules have only two chip select signals per memory socket routed. Therefore, such memory module would not be viable.

[0011] A need therefore exists for a transparent four rank memory module fitting into a memory socket having two chip select signals routed. A primary purpose of the present invention is to solve these needs and provide further, related advantages.

BRIEF DESCRIPTION OF THE INVENTION

[0012] A transparent four rank memory module has a front side and a back side. The front side has a third memory rank stacked on a first memory rank. The back side has a fourth memory rank stacked on a second memory rank. An emulator coupled to the memory module activates and controls one individual memory rank from either the first memory rank, the second memory rank, the third memory rank, or the fourth memory rank based on the signals received from a memory controller.

BRIEF DESCRIPTION OF THE DRAWINGS

[0013] The accompanying drawings, which are incorporated into and constitute a part of this specification, illustrate

one or more embodiments of the present invention and, together with the detailed description, serve to explain the principles and implementations of the invention.

[0014] In the drawings:

[0015] **FIG. 1** is a block diagram schematically illustrating a standard memory interface system in accordance with a prior art.

[0016] **FIG. 2** is a diagram schematically illustrating a stacked DDR device in accordance with one embodiment of the present invention.

[0017] **FIG. 3** is a diagram schematically illustrating a transparent four rank DDR memory module in accordance with one embodiment of the present invention.

[0018] **FIG. 4A** is a diagram schematically illustrating the front side of a transparent 72-bit registered DDR module in accordance with one embodiment of the present invention.

[0019] **FIG. 4B** is a diagram schematically illustrating the back side of a transparent 72-bit registered DDR module in accordance with one embodiment of the present invention.

[0020] **FIG. 5** is a truth table for a transparent four rank memory module in accordance with one embodiment of the present invention.

[0021] **FIG. 6A** is a block diagram schematically illustrating a row address decoding system for a transparent four rank memory module in accordance with one embodiment of the present invention.

[0022] **FIG. 6B** is a block diagram schematically illustrating a column address decoding system for a transparent four rank memory module in accordance with one embodiment of the present invention.

[0023] **FIG. 7** is a flow diagram schematically illustrating a method for emulating a two rank memory module.

[0024] **FIG. 8** is a block diagram schematically illustrating a CPLD in a transparent four rank DDR memory module in accordance with one embodiment of the present invention

DETAILED DESCRIPTION

[0025] Embodiments of the present invention are described herein in the context of a memory module. Those of ordinary skill in the art will realize that the following detailed description of the present invention is illustrative only and is not intended to be in any way limiting. Other embodiments of the present invention will readily suggest themselves to such skilled persons having the benefit of this disclosure. Reference will now be made in detail to implementations of the present invention as illustrated in the accompanying drawings. The same reference indicators will be used throughout the drawings and the following detailed description to refer to the same or like parts.

[0026] In the interest of clarity, not all of the routine features of the implementations described herein are shown and described. It will, of course, be appreciated that in the development of any such actual implementation, numerous implementation-specific decisions must be made in order to achieve the developer's specific goals, such as compliance with application- and business-related constraints, and that these specific goals will vary from one implementation to another and from one developer to another. Moreover, it will be appreciated that such a development effort might be complex and time-consuming, but would nevertheless be a routine undertaking of engineering for those of ordinary skill in the art having the benefit of this disclosure.

[0027] In accordance with one embodiment of the present invention, the components, process steps, and/or data structures may be implemented using various types of operating systems (OS), computing platforms, firmware, computer programs, computer languages, and/or general-purpose machines. The method can be run as a programmed process running on processing circuitry. The processing circuitry can take the form of numerous combinations of processors and operating systems, or a stand-alone device. The process can be implemented as instructions executed by such hardware, hardware alone, or any combination thereof. The software may be stored on a program storage device readable by a machine.

[0028] In addition, those of ordinary skill in the art will recognize that devices of a less general purpose nature, such as hardwired devices, field programmable logic devices (FPLDs), including field programmable gate arrays (FPGAs) and complex programmable logic devices (CPLDs), application specific integrated circuits (ASICs), or the like, may also be used without departing from the scope and spirit of the inventive concepts disclosed herein.

[0029] **FIG. 2** is a block diagram of an example of a standard stacked DDR 8 bit memory device **200** used in a memory module in accordance with one embodiment of the present invention. The memory device **200** interfaces with a memory controller (not shown) with three buses: an address bus **202**, a control bus **204**, and a data bus **206**. A differential clock bus **208** is also coupled to the DDR memory device **200**.

[0030] The address bus **202** conveys the following signals: address[n:0]**210** and BA[1:0]**212**.

[0031] The control bus **204** conveys the following signals: RAS **214**, CAS **216**, WE **218**, DQM **220**, CS[1:0]**222**, and CKE[1:0]**224**.

[0032] The data bus **206** conveys the following signals: data signals [7:0]**226** and DQS **228**.

[0033] The differential clock bus **208** includes two signals: clk **230**, and clk_n **232**.

[0034] **FIG. 3** illustrates a block diagram of stacked 8 bit memory devices on the front and back side of a memory module **300** in accordance with one embodiment of the present invention. The memory module's front side **302** includes a first rank **304** of memory devices **306** (U1 through U9). The first rank **304** is stacked with a third rank **308** of memory devices **306** (U19 through U27). As illustrated in **FIG. 3**, data bus [7:0] is connected to both ranks' memory devices **306** (U1 and U19). The remaining data buses are connected to their respective ranks' memory devices **306**. A chip select signal is coupled to each rank of memory devices. As illustrated in **FIG. 3**, chip select signal cs0 is connected to the first rank **304** (from U1 to U9) and chip select signal cs2 is connected to the third rank **308** (from U19 to U27).

[0035] The memory module's back side **310** includes a second rank **312** of memory devices **306** (U10 through U18). The second rank **312** is stacked with a fourth rank **314** of

memory devices **306** (U**28** through U**36**). As illustrated in **FIG. 3**, data bus [7;0] is connected to both ranks' memory devices **306** (U**10** and U**28**). The remaining data buses are connected to their respective ranks' memory devices **306**. A chip select signal is coupled to each rank of memory devices. As illustrated in **FIG. 3**, chip select signal cs**1** is connected to the second rank **312** (from U**10** to U**18**) and chip select signal cs**3** is connected to the fourth rank **314** (from U**28** to U**36**).

[**0036**] **FIG. 3** also illustrates a total of four chip select signals (cs**0**, cs**1**, cs**2**, and cs**3**). As illustrated in **FIG. 2**, the stacked memory device **200** has only two chip select signals: cs[1:0]**222**. In accordance with one embodiment of the invention, the present invention allows the four rank memory modules to communicate with a memory socket having only two chip select signals routed.

[**0037**] **FIG. 4A** illustrates a schematic diagram of the front side of a transparent 72 bit registered DDR module **400** in accordance with one embodiment of the present invention. The memory module **400** includes 92 contact pins **402** on the front side for connecting with a memory socket (not shown). The memory module **400** as illustrated in **FIG. 4A** includes nine 8 bit stacked memory devices **404**, nine corresponding Resistor Network (Rn) **406**, a register **408**, a CPLD **410**, a PLL **412**, and a SPD **414**. The stacking of the 8 bit stacked memory devices **404** was previously illustrated in **FIG. 3**. Memory devices **404** (U**1** through U**9**) are mounted on a surface of the front side of the memory module **400**. Memory devices **404** (U**19** through U**27**) are respectively stacked on memory devices **404** (U**1** through U**9**). Therefore the front side of the four rank memory modules **400** includes two ranks (rank **0** and rank **2**).

[**0038**] The register **408** is used to synchronize the incoming address and control signals with respect to differential clock signals **208** (clk and clk_n). Also, the register **408** may eliminate the loading of 36 devices in case of stacking or loading of 18 devices in case of monolithic memory devices from the main controller by separating the controller side signaling with memory side signal loading fan-out.

[**0039**] The PLL **412** is used to generate a zero-delay buffer off of system side input differential clock signals **208** (clk and clk_n). By using a PLL, the system side will not see the loading effect of either 18 differential clock loads or 36 differentials clock loads in the case of stacking memory devices.

[**0040**] The SPD **414** is a simple I2C interface EEPROM to hold information regarding memory module for BIOS during the power-up sequence.

[**0041**] The CPLD **410** emulates a two rank memory module on the four rank memory module **400**. CPLD **410** allows a system having a memory socket with only two chip select signals routed to interface with a four rank memory module where typically a two rank memory module couples with the memory socket. The CPLD **410** determines which rank from the four ranks to activate based upon the address and command signals from a memory controller coupled to the memory module **410**. The algorithm of CPLD **410** is further described in **FIGS. 5 and 7**.

[**0042**] **FIG. 4B** illustrates a schematic diagram of the back side of the transparent 72 bit registered DDR module **400** in accordance with one embodiment of the present invention. The memory module **400** includes 92 contact pins **416** on the back side for connecting with a memory socket (not shown). Therefore, memory module **400** has a total of 184 different contact pins on the front and back side. The memory module **400** as illustrated in **FIG. 4B** includes nine 8 bit stacked memory devices **404**, nine corresponding Resistor Network (Rn) **406**, another register **418**. The stacking of the 8 bit stacked memory devices **404** was previously illustrated in **FIG. 3**. Memory devices **404** (U**10** through U**18**) are mounted on a surface of the back side of the memory module **400**. Memory devices **404** (U**28** through U**36**) are respectively stacked on memory devices **404** (U**10** through U**18**). Therefore the back side of the four rank memory module **400** includes two ranks (rank **1** and rank **3**).

[**0043**] **FIG. 5** illustrates a truth table used in the process of the CPLD **410** to determine which rank is active (rank **0**, rank **1**, rank **2**, or rank **3**). CPLD **410** utilized three variables to determine the active rank: the highest address number Add(n), a first chip select signal (CS**0**), and a second chip select signal (CS**1**). As previously mentioned, each memory socket includes two active chip select signals (CS**0** and CS**1**). CPLD **410** combines these two variables (CS**0** and CS**1**) with the address number signal **210** from the address bus **202** to determine the active rank. Add(n) includes the highest binary digit from the address number signal **210**. For example, the highest address number Add(n) when the address signal **210** is 10010 would be the first digit to the left: "1".

[**0044**] As illustrated in **FIG. 5**, rank **0** is active when the Add(n) is 0, CS**1** is 1, and CS**0** is 0. Rank **1** is active when the Add(n) is 0, CS**1** is 0, and CS**0** is 1. Rank **2** is active when the Add(n) is 1, CS**1** is 1, and CS**0** is 0. Rank **3** is active when the Add(n) is 1, CS**1** is 0, and CS**0** is 1.

[**0045**] Because the row address and column address may differ between different memory device densities, the CPLD may employ two different decoding schemes: a Row Address Decoding scheme, and a Column Address Decoding scheme. The following non-limiting example is used for illustration purposes.

[**0046**] A 512 MByte memory module may be build with either two rank of 256 MByte density per rank or four ranks of 128 MByte density per rank. However, a 128 Mbit DDR SDRAM has different characteristics from a 256 Mbit DDR SDRAM device.

[**0047**] A 128 Mbit DDR SDRAM (16M×8) has the following characteristics:

Configuration	4M × 8 × 4
Refresh Rate	4K
Row Address	A0–A11
Column Address	A0–A9

[**0048**] A 256 Mbit DDR SDRAM (32M×8) has the following characteristics:

Configuration	8M × 8 × 4
Refresh Rate	8K
Row Address	A0–A12
Column Address	A0–A9

[**0049**] The size of the column addresses (A**0**-A**9**) for both 128 Mbit DDR SDRAM and 256 Mbit DDR SDRAM

devices match. However, the size of the row address for the 128 Mbit DDR SDRAM differs by one row address line from the 256 Mbit DDR SDRAM (A12). The CPLD **410** uses a Row Address Decoding scheme to emulate a two rank based on 256 Mbit DDR SDRAM Device Technology memory module with a four rank based on 128 Mbit DDR SDRAM Device Technology memory module. Under this scheme, address lines A0-A11 go to module register **408** and **418** and address lines A12 goes into CPLD **410** along with CS0 and CS1 for proper decoding. Therefore, the extra address line A12 is used by the CPLD to determine which rank (from the four ranks) is active. The decoding is performed as previously illustrated in **FIG. 5** above. For example, if address bus (A12-A0) has "1000010101010" and CS0 is "0" and CS1 is "1", then rank **2** is activated. In this present example, Add(n) for A12 is "1". The Row Address Decoding scheme is further illustrated in **FIG. 6A**.

[0050] **FIG. 6A** illustrates a method for decoding row addresses in accordance with one embodiment of the present invention. A module connector **602** (for example, the contact pins **402** and **416**) sends signals to the CPLD **604**, PLL **606**, and register **608**. The signals sent to CPLD **604** include cs0, cs1, Add(n), CAS, RAS, WE, CLK0, and CLK0_N. The signals sent to PLL **606** include CLK0, and CLK0_N. PLL relays the CLK0 and CLK0_N signals to register **608** and memory devices **306**. Register **608** also receives the following signals from module connector **602**: Add[n-1;0], RAS, CAS, WE, BA[1:0], CKE.

[0051] As illustrated in the example above, the 256 Mbit memory devices has an extra row address line (A12) when compared to the 128 Mbit memory devices. Register **608** of a four rank memory module emulating a two bank memory module receives an address with an address size matching the address size of the lower density memory devices (128 Mbit), i.e. A0-A11. In other words, the address signal from the module connector **608** does not include the extra row address line A12.

[0052] CPLD **604** also ensures that all commands for a two rank memory module conveyed by the module connector **602** are also performed on the four rank memory modules. For example, CPLD **604** generates rcs2 and rcs3, besides rcs0 and rcs1 off of CS0, CS1 and Add(n) from the memory controller side. CPLD **604** also generates rcs2 when CS0 Auto Precharge all Banks Commands are issued. CPLD **604** also generates rcs3 when CS1 Auto Precharge all Banks Commands are issued. CPLD **604** also generates rcs2 when CS0 Auto Refresh Commands are issued. CPLD **604** also generates rcs3 when CS1 Auto Refresh Commands are issued. CPLD **604** also generates rcs2 when CS0 Load Mode Register Commands are issued. CPLD **604** also generates rcs3 when CS1 Load Mode Register Commands are issued.

[0053] However, as previously mentioned, a memory module may also be built using two device families which only differs in their column address size, and have the same row address size. The following example illustrates this situation and describes the Column Decoding Scheme.

[0054] A 1024 Mbyte memory module may be build with either two rank of 512 MByte density per rank or four ranks of 256 MByte density per rank. However, the 256 Mbit DDR SDRAM has different characteristics from a 512 Mbit DDR SDRAM.

[0055] A 256 Mbit DDR SDRAM (32M×8) has the following characteristics:

Configuration	8M × 8 × 4
Refresh Rate	8K
Row Address	A0–A12
Column Address	A0–A9

[0056] A 512 Mbit DDR SDRAM (64M×8) has the following characteristics:

Configuration	16M × 8 × 4
Refresh Rate	8K
Row Address	A0–A12
Column Address	A0–A9, A11

[0057] The size of the row addresses (A0-A9) for both 256 Mbit DDR SDRAM and 512 Mbit DDR SDRAM devices match. However, the size of the column address for the 256 Mbit DDR SDRAM differ by one address line from the 512 Mbit DDR SDRAM (A11). The CPLD **410** uses the Column Address Decoding scheme to emulate a two ranks 512 Mbit based DDR SDRAM device Technology memory module with a four ranks 256 Mbit based DDR SDRAM device Technology memory module. Under this scheme, address lines A0-A12 go to module register **408** and **418** and address lines A11 goes into CPLD **410** along with CS0 and CS1 for proper decoding. Therefore, the address line A11 is used by the CPLD to determine which rank (from the four ranks) is active. The decoding is performed as previously illustrated in **FIG. 5** above. For example, if address bus (A11, A9-A0) has "100110101010" and CS0 is "0" and CS1 is "1", then rank **2** is activated. In this present example, Add(n) for A11 is "1". The Column Address Decoding scheme is further illustrated in **FIG. 6B**.

[0058] **FIG. 6B** illustrates a method for decoding column addresses in accordance with one embodiment of the present invention. A module connector **602** (for example, the contact pins **402** and **416**) sends signals to the CPLD **604**, PLL **606**, and register **608**. The signals sent to CPLD **604** include CS0, CS1, Add(n-1), CAS, RAS, WE, CLK0, and CLK0_N. The signals sent to PLL **606** include CLK0, and CLK0_N. PLL relays CLK0 and CLK0_N signals to register **608** and memory devices **306**. Register **608** also receives the following signals from module connector **602**: Add[n;0], RAS, CAS, WE, BA[1:0], CKE.

[0059] As illustrated in the example above, the 512 Mbit memory devices has an extra column address line (A11) when compared to the 256 Mbit memory devices. Register **608** of a four rank memory module emulating a two rank memory module receives an address with an address size matching the address size of the lower density memory devices (256 Mbit), i.e. A0-A12.

[0060] CPLD **604** also ensures that all commands for a two rank memory module conveyed by the module connector **602** are also performed on the four rank memory modules. For example, CPLD **604** generates rcs2 and rcs3, besides rcs0 and rcs1 off of CS0, CS1 and Add(n) from the memory controller side. CPLD **604** also generates rcs2 when

CS0 Auto Precharge all Banks Commands are issued. CPLD **604** also generates rcs3 when CS1 Auto Precharge all Banks Commands are issued. CPLD **604** also generates rcs2 when CS0 Auto Refresh Commands are issued. CPLD **604** also generates rcs3 when CS1 Auto Refresh Commands are issued. CPLD **604** also generates rcs2 when CS0 Load Mode Register Commands are issued. CPLD **604** also generates rcs3 when CS1 Load Mode Register Commands are issued.

[0061] It should be noted that the internal circuitry in the CPLD **410** for Row Address Decoding and Column Address Decoding are different. In particular, in the Column Address Decoding scheme, a unique decoding circuitry is required because in a standard DDR memory module there is only one set of address lines and memory organized as a matrix in such that in order to access an x 4, x8 or x16 cell, two set of addresses needs to be provided. First, the Row address needs to be provided with the proper control and command signals then on a separate cycle, the Column address needs to be provided with its proper control and command signals in order to read or write to that particular cell.

[0062] **FIG. 7** illustrates a method for emulating a two rank memory module with a four rank memory module in accordance with one embodiment of the present invention. At **702**, the memory module receives a command signal and address signal from a memory controller. The memory controller addressed the command signal and address signal to a two rank memory module. The command signal includes CS0, CS1, CKE, CAS, RAS, and WE. The address signal includes Add(n). At **704**, the CPLD of the four rank memory module determines which rank should be active based on the address and command signals (cs0, cs1, and Add(n)). This determination was previously described using the truth table illustrated in **FIG. 5**. At **706**, the CPLD performs either a Row Address Decoding scheme or a Column Address Decoding scheme as previously described to relay the command signals.

[0063] It should be noted that the physical address lines and logical address lines are different in this methodology. This is a non-linear addressing versus SRAM which uses linear addressing. In this methodology, a much lower number of pins are used to access the same amount of memory locations as a SRAM device with longer latency due to multiple cycle of providing the Row and Column logical addresses.

[0064] **FIG. 8** is a block diagram schematically illustrating the internal circuitry of a CPLD used in the transparent four rank DDR memory module. CS0 signal and CS1 signal generate a CS internal signal through logical device **802**. The CS signal is transmitted to Load mode register **804**, auto refresh **806**, and auto precharge detection **808** sub circuitries.

[0065] The Load Mode Register circuitry **804** also receives Chip Select (CS) signal, Row Address Strobe (RAS) signal, Column Address Strobe (CAS) signal, and Write Enable signal (WE). This module **804** detects load mode register cycle if inputs are asserted properly to indicate LMR command.

[0066] The Auto Refresh circuitry **806** receives Chip Select (CS) signal, Row Address Strobe (RAS) signal, Column Address Strobe (CAS) signal, and Write Enable signal (WE). This module **806** detects auto refresh cycle if inputs are asserted properly to indicate Auto Refresh command.

[0067] The Auto Precharge circuitry **808** receives Chip Select (CS) signal, Row Address Strobe (RAS) signal, Column Address Strobe (CAS) signal, and Write Enable (WE) signal. This module **808** detects auto precharge cycle if inputs are asserted properly to indicate auto precharge command.

[0068] The output of all three sub circuitries (LMR **804**, auto refresh **806** and auto precharge **808**) will go to a logical device OR **810** which will drive another level of OR logic **812** and **814** along with either highest address line (**814**) or it's inverted state (**812**).

[0069] The inverted state drives both MUX wcs0 and wcs1 blocks **816** and **818** which goes to a respective register **820** and **822** and gets fan-out into rcs0*a* and rcs0*b* or rcs1*a* or rcs1*b* eventually.

[0070] The non-inverted state will drive both MUX wcs2 and wcs3 blocks **824** and **826** which goes to a respective register **828** and **830** and gets fan-out into rcs2*a* and rcs2*b* or rcs3*a* or rcs3*b* eventually.

[0071] Many other families of memory devices or densities of memory devices (not shown) may be used to build the four rank memory module. Those of ordinary skill in the art will appreciate that the example of four rank memory module described above is not intended to be limiting and that other configuration can be used without departing from the inventive concepts herein disclosed.

[0072] While embodiments and applications of this invention have been shown and described, it would be apparent to those skilled in the art having the benefit of this disclosure that many more modifications than mentioned above are possible without departing from the inventive concepts herein. The invention, therefore, is not to be restricted except in the spirit of the appended claims.

What is claimed is:

1. A memory module having a front side and a back side comprising:

a third memory rank stacked on a first memory rank, said first memory rank coupled to the front side;

a fourth memory rank stacked on a second memory rank, said second memory rank coupled to the back side; and

an emulator coupled to the memory module controlling signals to said first memory rank, said second memory rank, said third memory rank, and said fourth memory rank.

2. The memory module of claim 1 wherein said first, second, third, and fourth memory rank, each, includes at least one memory devices.

3. The memory module of claim 1 further comprising a first contact for a first chip select signal and a second contact for a second chip select signal.

4. The memory module of claim 1 wherein said emulator electrically receives said first chip select signal, said second chip select signal, a set of command and/or control signals, a memory address signal, and a clock signal.

5. The memory module of claim 4 wherein said emulator activates a memory rank based on said first chip select signal, said second chip select signal, said memory address signal.

6. The memory module of claim 5 wherein said memory address signal includes the highest address line of said memory address signal under a row address decoding scheme.

7. The memory module of claim 5 wherein said memory address signal includes the highest address line offset by one address of said memory address signal under a column address decoding scheme.

8. The memory module of claim 1 wherein said emulator re-addresses signals originally addressed for a two rank memory module to said four memory ranks.

9. The memory module of claim 1 wherein said emulator includes a Programmable Logic Device.

10. A computing system comprising:

a central processing unit;

a memory controller coupled to said central processing unit;

a memory socket coupled to said memory controller;

a four rank memory module coupled to said memory socket;

an emulator coupled to said four rank memory module, said emulator toggling signals for two rank memory module to said four rank memory module.

11. The computing system of claim 10 wherein said four rank memory module has a front side and a back side, said four rank memory module further comprising:

a third memory rank stacked on a first memory rank, said first memory rank coupled to the front side; and

a fourth memory rank stacked on a second memory rank, said second memory rank coupled to the back side,

wherein said emulator toggles signals between said first memory rank, said second memory rank, said third memory rank, and said fourth memory rank.

12. The computing system of claim 10 wherein said first, second, third, and fourth memory rank, each, includes at least one memory devices.

13. The computing system of claim 10 further comprising a first contact for a first chip select signal and a second contact for a second chip select signal.

14. The computing system of claim 10 wherein said emulator electrically receives said first chip select signal, said second chip select signal, a set of command and/or control signals, a memory address signal, and a clock signal.

15. The computing system of claim 14 wherein said emulator controls one of said first, second, third or fourth memory rank based on said first chip select signal, said second chip select signal, and said memory address signal.

16. The computing system of claim 15 wherein said memory address signal includes the highest address line of said memory address signal under a row decoding scheme.

17. The computing system of claim 15 wherein said memory address signal includes the highest address line offset by one address of said memory address signal under the column decoding scheme.

18. The computing system of claim 10 wherein said emulator re-addresses signals originally addressed for a two rank memory module to said four memory ranks.

19. The computing system of claim 10 wherein said emulator includes a Programmable Logic Device.

20. A method for simulating a two rank memory module using a four rank memory module on one memory socket comprising:

receiving a first chip select signal, a second chip select signal, and an address signal; and

controlling one memory rank of said four rank memory module in response to said first chip select signal, a second chip select signal, and an address signal.

21. The method of claim 20 further comprising:

re-addressing said address signal to said one memory rank.

22. A memory module comprising:

means for receiving a first chip select signal, a second chip select signal, and an address signal; and

means for selecting one memory rank from said four rank memory module in response to said first chip select signal, a second chip select signal, and an address signal.

23. A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform a method for simulating a two rank memory module using a four rank memory module on one memory socket, the method including:

receiving a first chip select signal, a second chip select signal, and an address signal; and

selecting one memory rank from said four rank memory module in response to said first chip select signal, a second chip select signal, and an address signal.

* * * * *

APPENDIX G

Netrino
The Embedded Systems Experts

Course Calendar

Fall 2008 training courses now available for registration. Check out the schedule.

Embedded C Quiz

Test your embedded software skills in our online quiz.

Newsletter Signup

Receive public course dates, how-to articles, and more. Learn more.

Articles

- Categories
  - Embedded C/C++
  - Hardware Interfacing
  - Network Protocols
  - Processes and Tools
  - Programming Algorithms
  - Real-Time Java
  - Real-Time Operating Systems
  - User Interface Design

How Programmable Logic Works

Hardware Interfacing

by **Michael Barr**

In recent years, the line between hardware and software has blurred. Hardware now engineers create the bulk of their new digital circuitry in programming languages such as VHDL and Verilog. This article will help you make sense of programmable logic.

A quiet revolution is taking place. Over the past few years, the density of the average programmable logic device has begun to skyrocket. The maximum number of gates in an FPGA is currently around 500,000 and doubling every 18 months. Meanwhile, the price of these chips is dropping. What all of this means is that the price of an individual NAND or NOR is rapidly approaching zero! And the designers of embedded systems are taking note. Some system designers are buying processor cores and incorporating them into system-on-a-chip designs; others are eliminating the processor and software altogether, choosing an alternative hardware-only design.

As this trend continues, it becomes more difficult to separate hardware from software. After all, both hardware and software designers are now describing logic in high-level terms, albeit in different languages, and downloading the compiled result to a piece of silicon. Surely no one would claim that language choice alone marks a real distinction between the two fields. Turing's notion of machine-level equivalence and the existence of language-to-language translators have long ago taught us all that that kind of reasoning is foolish. There are even now products that allow designers to create their hardware designs in traditional programming languages like C. So language differences alone are not enough of a distinction.

Both hardware and software designs are compiled from a human-readable form into a machine-readable one. And both designs are ultimately loaded into some piece of silicon. Does it matter that one chip is a memory device and the other a piece of programmable logic? If not, how else can we distinguish hardware from software?

I'm not convinced that an unambiguous distinction between hardware and software can ever be found, but I don't think that matters all that much. Regardless of where the line is drawn, there will continue to be engineers like you and me who cross the boundary in our work. So rather than try to nail down a precise boundary between hardware and software design, we must assume that there will be overlap in the two fields. And we must all learn about new things. Hardware designers must learn how to write better programs, and software developers must learn how to utilize programmable logic.

Glossary

Find definitions for technical terms in our **Embedded Systems Glossary**.

A B C D E
F G H I J
K L M N O
P Q R S T
U V W X Y
Z Symbols

Bookmark It

Bookmark this page or share it with a colleague.

BOOKMARK

Types of programmable logic

Many types of programmable logic are available. The current range of offerings includes everything from small devices capable of implementing only a handful of logic equations to huge FPGAs that can hold an entire processor core (plus peripherals!). In addition to this incredible difference in size there is also much variation in architecture. In this section, I'll introduce you to the most common types of programmable logic and highlight the most important features of each type.

PLDs

At the low end of the spectrum are the original Programmable Logic Devices (PLDs). These were the first chips that could be used to implement a flexible digital logic design in hardware. In other words, you could remove a couple of the 7400-series TTL parts (ANDs, ORs, and NOTs) from your board and replace them with a single PLD. Other names you might encounter for this class of device are Programmable Logic Array (PLA), Programmable Array Logic (PAL), and Generic Array Logic (GAL).

PLDs are often used for address decoding, where they have several clear advantages over the 7400-series TTL parts that they replaced. First, of course, is that one chip requires less board area, power, and wiring than several do. Another advantage is that the design inside the chip is flexible, so a change in the logic doesn't require any rewiring of the board. Rather, the decoding logic can be altered by simply replacing that one PLD with another part that has been programmed with the new design.

Inside each PLD is a set of fully connected macrocells. These macrocells are typically comprised of some amount of combinatorial logic (AND and OR gates, for example) and a flip-flop. In other words, a small Boolean logic equation can be built within each macrocell. This equation will combine the state of some number of binary inputs into a binary output and, if necessary, store that output in the flip-flop until the next clock edge. Of course, the particulars of the available logic gates and flip-flops are specific to each manufacturer and product family. But the general idea is always the same.

Because these chips are pretty small, they don't have much relevance to the remainder of this discussion. But you do need to understand the origin of programmable logic chips before we can go on to talk about the larger devices. Hardware designs for these simple PLDs are generally written in languages like ABEL or PALASM (the hardware equivalents of assembly) or drawn with the help of a schematic capture tool.

CPLDs

As chip densities increased, it was natural for the PLD manufacturers to evolve their products into larger (logically, but not necessarily physically) parts called Complex Programmable Logic Devices (CPLDs). For most practical purposes, CPLDs can be thought of as multiple PLDs (plus some programmable interconnect) in a single chip. The larger size of a CPLD allows you to implement either more logic equations or a more complicated design. In fact, these chips are large enough to replace dozens of those pesky 7400-series parts.

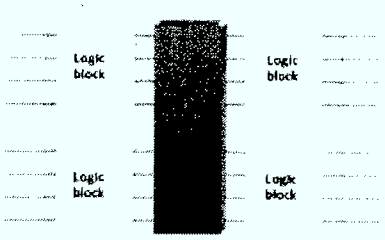


Figure 1. Internal structure of a CPLD

Figure 1 contains a block diagram of a hypothetical CPLD. Each of the four logic blocks shown there is the equivalent of one PLD. However, in an actual CPLD there may be more (or less) than four logic blocks. I've just drawn it that way for simplicity. Note also that these logic blocks are themselves comprised of macrocells and interconnect wiring, just like an ordinary PLD.

Unlike the programmable interconnect within a PLD, the switch matrix within a CPLD may or may not be fully connected. In other words, some of the theoretically possible connections between logic block outputs and inputs may not actually be supported within a given CPLD. The effect of this is most often to make 100% utilization of the macrocells very difficult to achieve. Some hardware designs simply won't fit within a given CPLD, even though there are sufficient logic gates and flip-flops available.

Because CPLDs can hold larger designs than PLDs, their potential uses are more varied. They are still sometimes used for simple applications like address decoding, but more often contain high-performance control-logic or complex finite state machines. At the high-end (in terms of numbers of gates), there is also a lot of overlap in potential applications with FPGAs. Traditionally, CPLDs have been chosen over FPGAs whenever high-performance logic is required. Because of its less flexible internal architecture, the delay through a CPLD (measured in nanoseconds) is more predictable and usually shorter.

FPGAs

Field Programmable Gate Arrays (FPGAs) can be used to implement just about any hardware design. One common use is to prototype a lump of hardware that will eventually find its way into an ASIC. However, there is nothing to say that the FPGA can't remain in the final product. Whether or not it does will depend on the relative weights of development cost and production cost for a particular project. (It costs significantly more to develop an ASIC, but the cost per chip may be lower in the long run. The cost tradeoff involves expected number of chips to be produced and the expected likelihood of hardware bugs and/or changes. This makes for a rather complicated cost analysis, to say the least.)

The development of the FPGA was distinct from the PLD/CPLD evolution just described. This is apparent when you look at the structures inside. Figure 2 illustrates a typical FPGA architecture. There are three key parts of its structure: logic blocks, interconnect, and I/O blocks. The I/O blocks form a ring around the outer edge of the part. Each of these provides individually selectable input, output, or bi-directional access to one of the general-

purpose I/O pins on the exterior of the FPGA package. Inside the ring of I/O blocks lies a rectangular array of logic blocks. And connecting logic blocks to logic blocks and I/O blocks to logic blocks is the programmable interconnect wiring.

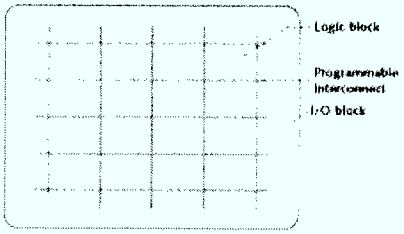


Figure 2. Internal structure of an FPGA

The logic blocks within an FPGA can be as small and simple as the macrocells in a PLD (a so-called fine-grained architecture) or larger and more complex (coarse-grained). However, they are never as large as an entire PLD, as the logic blocks of a CPLD are. Remember that the logic blocks of a CPLD contain multiple macrocells. But the logic blocks in an FPGA are generally nothing more than a couple of logic gates or a look-up table and a flip-flop.

Because of all the extra flip-flops, the architecture of an FPGA is much more flexible than that of a CPLD. This makes FPGAs better in register-heavy and pipelined applications. They are also often used in place of a processor-plus-software solution, particularly where the processing of input data streams must be performed at a very fast pace. In addition, FPGAs are usually denser (more gates in a given area) and cost less than their CPLD cousins, so they are the de facto choice for larger logic designs.

Hardware design and development

The process of creating digital logic is not unlike the embedded software development process you're already familiar with. A description of the hardware's structure and behavior is written in a high-level hardware description language (usually VHDL or Verilog) and that code is then compiled and downloaded prior to execution. Of course, schematic capture is also an option for design entry, but it has become less popular as designs have become more complex and the language-based tools have improved. The overall process of hardware development for programmable logic is shown in Figure 3 and described in the paragraphs that follow.

Perhaps the most striking difference between hardware and software design is the way a developer must think about the problem. Software developers tend to think sequentially, even when they are developing a multithreaded application. The lines of source code that they write are always executed in that order, at least within a given thread. If there is an operating system it is used to create the appearance of parallelism, but there is still just one execution engine. During design entry, hardware designers must think-and program-in parallel. All of the input signals are processed in parallel, as they travel through a set of execution engines-each one a series of macrocells and interconnections-toward their

destination output signals. Therefore, the statements of a hardware description language create structures, all of which are "executed" at the very same time. (Note, however, that the transference from macrocell to macrocell is usually synchronized to some other signal, like a clock.)

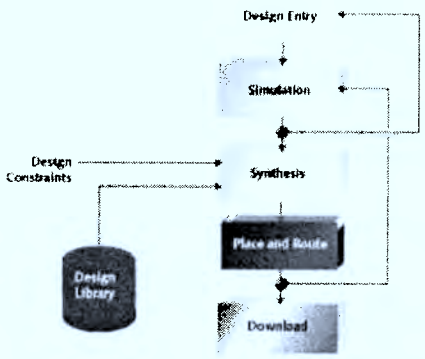


Figure 3. Programmable logic design process

Typically, the design entry step is followed or interspersed with periods of functional simulation. That's where a simulator is used to execute the design and confirm that the correct outputs are produced for a given set of test inputs. Although problems with the size or timing of the hardware may still crop up later, the designer can at least be sure that his logic is functionally correct before going on to the next stage of development.

Compilation only begins after a functionally correct representation of the hardware exists. This hardware compilation consists of two distinct steps. First, an intermediate representation of the hardware design is produced. This step is called synthesis and the result is a representation called a netlist. The netlist is device independent, so its contents do not depend on the particulars of the FPGA or CPLD; it is usually stored in a standard format called the Electronic Design Interchange Format (EDIF).

The second step in the translation process is called place & route. This step involves mapping the logical structures described in the netlist onto actual macrocells, interconnections, and input and output pins. This process is similar to the equivalent step in the development of a printed circuit board, and it may likewise allow for either automatic or manual layout optimizations. The result of the place & route process is a bitstream. This name is used generically, despite the fact that each CPLD or FPGA (or family) has its own, usually proprietary, bitstream format. Suffice it to say that the bitstream is the binary data that must be loaded into the FPGA or CPLD to cause that chip to execute a particular hardware design.

Increasingly there are also debuggers available that at least allow for single-stepping the hardware design as it executes in the programmable logic device. But those only complement a simulation environment that is able to use some of the information generated during the place & route step to provide gate-level simulation. Obviously, this type of integration of device-specific information into a generic simulator requires a good

working relationship between the chip and simulation tool vendors.

Device programming

Once you've created a bitstream for a particular FPGA or CPLD, you'll need to somehow download it to the device. The details of this process are dependent upon the chip's underlying process technology. Programmable logic devices are like non-volatile memories in that there are multiple underlying technologies. In fact, exactly the same set of names is used: PROM (for one-time programmables), EPROM, EEPROM, and Flash.

Just like their memory counterparts, PROM and EPROM-based logic devices can only be programmed with the help of a separate piece of lab equipment called a device programmer. On the other hand, many of the devices based on EEPROM or Flash technology are in-circuit programmable. In other words, the additional circuitry that's required to perform device (re)programming is provided within the FPGA or CPLD silicon as well. This makes it possible to erase and reprogram the device internals via a JTAG interface or from an on-board embedded processor. (Note, however, that because this additional circuitry takes up space and increases overall chip costs, a few of the programmable logic devices based on EEPROM or Flash still require insertion into a device programmer.)

In addition to non-volatile technologies, there are also programmable logic devices based on SRAM technology. In such cases, the contents of the device are volatile. This has both advantages and disadvantages. The obvious disadvantage is that the internal logic must be reloaded after every system or chip reset. That means you'll need an additional memory chip of some sort in which to hold the bitstream. But it also means that the contents of the logic device can be manipulated on-the-fly. In fact, you could imagine a scenario in which the actual bitstream is reloaded from a remote source (via a network of some sort?), so that the hardware design could be upgraded as easily as software.

Applications

Now that you understand the technology, you're probably wondering what all of these FPGAs and CPLDs are doing within the embedded systems. However, their uses are so varied that it's impossible to generalize. Rather, I'll just touch on some of the emerging trends. This should hopefully answer your question, though admittedly indirectly.

Prototyping

Many times a CPLD or FPGA will be used in a prototype system. A small device may be present to allow the designers to change a board's glue logic more easily during product development and testing. Or a large device may be included to allow prototyping of a system-on-a-chip design that will eventually find its way into an ASIC. Either way, the basic idea is the same: allow the hardware to be flexible during product development. When the product is ready to ship in large quantities, the programmable device will be replaced with a less expensive, though functionally equivalent, hard-wired alternative.

Embedded cores

More and more vendors are selling or giving away their processors and peripherals in a form that is ready to be integrated into a programmable logic-based design. They either recognize the potential for growth in the system-on-a-chip area and want a piece of the royalties or want to promote the use of their particular FPGA or CPLD by providing libraries of ready-to-use building blocks. Either way, you will gain with lower system costs and faster time-to-market. Why develop your own hardware when you can buy an equivalent piece of virtual silicon?

The Intellectual Property (IP) market is growing rapidly. It's common to find microprocessors and microcontrollers for sale in this form, as well as complex peripherals like PCI controllers. Many of the IP cores are even configurable. Would you like a 16-bit bus or a 32-bit bus? Do you need the floating-point portion of the processor? And, of course, you'll find all of the usual supporting cast of simple peripherals like serial controllers and timer/counter units are available as well.

Hybrid chips

There's also been some movement in the direction of hybrid chips, which combine a dedicated processor core with an area of programmable logic. The vendors of hybrid chips are betting that a processor core embedded within a programmable logic device will require far too many gates for typical applications. So they've created hybrid chips that are part fixed logic and part programmable logic. The fixed logic contains a fully functional processor and perhaps even some on-chip memory. This part of the chip also interfaces to dedicated address and data bus pins on the outside of the chip. Application-specific peripherals can be inserted into the programmable logic portion of the chip, either from a library of IP cores or the customer's own designs.

Reconfigurable computing

As mentioned earlier, an SRAM-based programmable device can have its internal design altered on-the-fly. This practice is known as reconfigurable computing. Though originally proposed in the late 1960's by a researcher at UCLA, this is still a relatively new field of study. The decades-long delay had mostly to do with a lack of acceptable reconfigurable hardware. On-the-fly reprogrammable logic chips have only recently reached gate densities making them suitable for anything more than academic research. But the future of reconfigurable computing is bright and it is already finding a niche in high-end communications, military, and intelligence applications.

Gate Count
The gate count by itself is almost useless. Different vendors use different measures: number of available gates, equivalent number of NAND gates, equivalent number of gates in a PLD, equivalent number of gates in an ASIC, etc. You simply can't compare these numbers across vendors. A better comparison can be made in terms of numbers of registers (flip-flops) and I/O pins.
Number of I/O Pins
Are there adequate inputs and outputs for your design? This is often a more limiting constraint than gate count, and it very much affects the cost of the chip. As a result,

many manufacturers offer the same part with different numbers of I/O pins.
Cost per Chip
Obviously, cost is a factor if you'll be including a CPLD or FPGA in your production system. Would it be cheaper in the long run to develop a fixed ASIC design and produce a large quantity of them? If you stick with the programmable device, you'll want to use the smallest part with adequate resources for your design.
Available Tools
The most popular Verilog and VHDL simulation and synthesis tools are sold by third party tool vendors. These tools generally have support for a laundry list of common FPGAs and CPLDs. This means that the tools understand the constraints of your particular chip and also understand the timing-relating information that comes out of the place and route tool.
Performance
Generally speaking, CPLDs are faster and introduce more predictable delays than FPGAs. However, that's because their internal structure is less flexible. So you have to give something up for the extra speed. What's typically lost is density. The larger your design, the more likely it is that you'll have to use a slower part. When using an FPGA, the actual performance of your design won't really be known until the final place and route process is complete, since the routing specifics will play a role.
Power Consumption
Power consumption can be an important consideration in any system. EEPROM and Flash-based devices usually require more power than those based on PROM, EPROM, or SRAM technologies.
Packaging
Programmable logic devices are available in all sorts of packages. Your choice of a package will most likely be driven by your need to reduce power consumption, heat dissipation, size, and/or cost.

Table 1. Checklist for programmable logic selection

What's it to ya?

Hopefully, you now have a better understanding of this new kind of software that is really hardware in disguise. (Or is it a new kind of hardware that is really software in disguise?) This should give you a better basis for communicating with hardware designers on partitioning issues like: What functions on your next project should be implemented in dedicated logic, programmable logic, and/or software? I've found that there are valid reasons for choosing all three of these implementation techniques, and that you must pay close attention to the requirements of the particular application. As software and hardware continue to simultaneously expand and overlap, we must all broaden our perspectives and be willing to learn new things.

This article was published in the June 1999 issue of **Embedded Systems Programming**. If you wish to cite the article in your own work, you may find the following MLA-style information helpful:

Barr, Michael. "Programmable Logic: What's it to Ya?," Embedded Systems

Programming, June 1999, pp. 75-84.

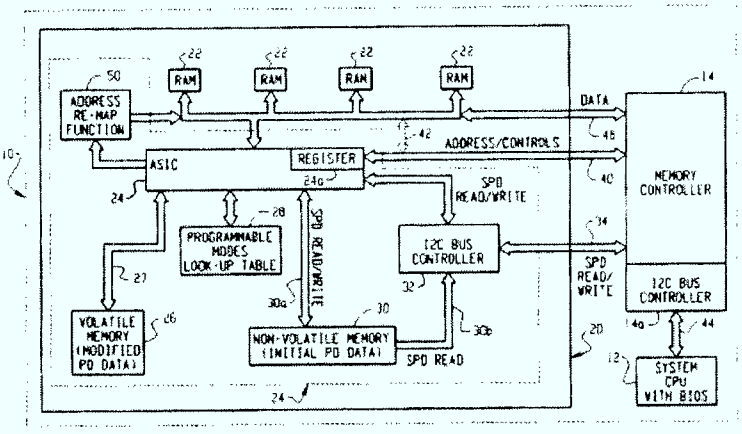
» **webmaster's blog** **Login** or **register** to post comments

Netrino, LLC • 9250 Bendix Road • Columbia, Maryland 21045 • 866.78.EMBED

APPENDIX H

APPENDIX H – CLAIM CHART FOR DELL AND FOR DELL AND BARR

Requester submits that, under the "broadest reasonable interpretation" standard, claims 1-5, 7, and 9-12 of the '386 patent are obvious over Dell and claims 6 and 8 are obvious over Dell and Barr. This Appendix provides an element-by-element comparison of the claims to the prior art, showing how the prior art discloses every element of the claims.

US PATENT 7,289,386	DISCLOSURE US PATENT 6,209,074 (DELL) AND BARR
Claim 1	
1. A memory module connectable to a computer system, the memory module comprising:	"The invention relates generally to memory modules for computer systems." (*1:17-21*) FIG. 1 illustrates a "memory module 20" with "memory chips 22" connected to "system CPU with BIOS 12." "With reference to FIG. 1, an embodiment of the invention is illustrated in the environment of a computer system 10." (3:19-20)  *Fig. 1* "DRAM memory is available in module form, in which a plurality of memory chips are placed on a small circuit card, which card then plugs into a memory socket connected to the computer motherboard or memory carrier card. Examples of commercial memory modules are SIMMs (Single In-line Memory Modules) and DIMMs (Dual In-line Memory Modules)." (*1:46-52*)
a printed circuit board;	"DRAM memory is available in module form, in which a plurality of memory chips are placed on a small circuit card, which card then plugs into a memory socket connected to the computer motherboard or memory carrier card.." (*1:46-50*)

US PATENT 7,289,386	DISCLOSURE US PATENT 6,209,074 (DELL) AND BARR
a plurality of memory devices coupled to the printed circuit board, the plurality of memory devices having a first number of memory devices; and	"The memory module 20 can include memory devices or chips such as, for example, SDRAMS . . . ." (*3:44-45*) In this mapping of claim 1 to Dell, the memory devices of claim 1 are mapped to, and correspond to, the memory cells or memory arrays that are implemented within the distinct memory chips. Thus, where the units of memory in claim 1 are the chips, in Dell the units of memory are the cells (also referred to as arrays) within the chips. "Some memory devices have the memory cells organized into a number of logical banks which can be individually addressed by the system memory controller." (*2:22-24*) "SDRAM memories can include a number of internal banks of memory arrays." (*4:24-25*) As will be seen, the memory cells of Dell correspond to the memory devices of claim 1, and the operations and structures performed with respect to the memory cells of Dell correspond to those performed with respect to the "memory devices" of claim 1. Thus, it would have been obvious to apply the teachings of Dell in the context of banks of memory cells to the parallel context of ranks of memory devices.
a logic element coupled to the printed circuit board,	"In accordance with one aspect of the invention, the memory module 20 includes a logic circuit 24. In the embodiment, the logic circuit 24 is realized in the form of an application specific integrated circuit (ASIC)." (*3:55-58*)
the logic element receiving a set of input control signals from the computer system,	In some embodiments, the logic element (circuit 24) receives address and control signals from the computer system (memory controller 24) over bus 40. "The memory controller 14 provides address, data and bus control signals for interfacing the CPU 12 and the memory module 20. The memory controller 14 includes logic for addressing, receiving, writing and refreshing data in the plurality of memory devices 22 on the module 20." (*3:30-34*) "The system memory controller 14 communicates with the module 20 via an ADDRESS AND CONTROL bus 40. This bus 40 can interface directly with the ASIC circuit 24 as illustrated [in FIG. 1, reproduced above]." (*5:52-55*) "In some applications, for example, the ASIC 24 will send address and control signals to the memory chips 22, but the data will flow directly to the memory controller 14." (*6:5-8*)

US PATENT 7,289,386	DISCLOSURE US PATENT 6,209,074 (DELL) AND BARR
the set of input control signals corresponding to a second number of memory devices smaller than the first number of memory devices,	The module has some number of memory chips. The computer system is expecting memory chips with a smaller number of banks than are actually present. This means the system is expecting a smaller number of memory cells or arrays than are actually present (the "first number"). As explained above, the memory cells of Dell correspond to the memory devices of claim 1 and are the units of memory that are counted and organized into banks. "In this manner, a four bank memory device 22 could still be used with a system 12 that is expecting or requires a two bank memory device, for example." (*4:48-4:51*)
the logic element generating a set of output control signals in response to the set of input control signals, the set of output control signals corresponding to the first number of memory devices,	"In some applications, for example, the ASIC 24 will send address and control signals to the memory chips 22 . . . ." (*6:5-7*) Because they are being sent to the chips themselves, the control signals generated by the logic element (circuit 24) necessarily correspond to the first number of memory devices and to the first (actual) number of memory units (cells or arrays).
wherein the plurality of memory devices are arranged in a first number of ranks, and the set of input control signals corresponds to a second number of ranks of memory modules, the second number of ranks less than the first number of ranks,	"[In] another embodiment, a method for using an M bank memory device in a computer system that has N bank addressing, . . . re-mapping at least one of said address inputs as an additional bank address signal" (2:54-62) The "first number of ranks" of the claim corresponds to the actual number M of banks in the module, i.e., in the chips on the module. The input control signals are generated by the system, which has N bank addressing, and therefore the input control signals correspond to N banks. N is less than M. The correspondence between the banks of Dell and the ranks of the claim is shown, for example, in the following citations. "SDRAM memories can include a number of internal banks of memory arrays. An ASIC can be configured to allow the use of a 4 bank SDRAM in a system that supports only 2 bank SDRAMs" (*Dell, 4:24-28*) "Some memory devices have the memory cells organized into a number of logical banks which can be individually addressed by the system memory controller. Control of bank selection is accomplished through the use of one or more Bank Address (BA) inputs." (*Dell, 2:22-26*)

US PATENT 7,289,386	DISCLOSURE US PATENT 6,209,074 (DELL) AND BARR
wherein the logic element further responds to a first command signal from the computer system by generating a second command signal transmitted to the plurality of memory devices, the first command signal corresponding to the second number of ranks and the second command signal corresponding to the first number of ranks.	"The memory controller 14 includes logic for addressing, receiving, writing and refreshing data in the plurality of memory devices 22 on the module 20." (*3:32-34*) These commands of reading, writing and refreshing data are received as command signals from the memory controller 14 of the computer system 10. "The memory controller 14 provides address, data and bus control signals for interfacing the CPU 12 and the memory module 20." (*3:30-32*) The signals coming from the computer system necessarily correspond to the number of banks that the system is configured for (corresponding to "the second number of ranks") and the signals transmitted to the actual devices (the "second command signal") necessarily correspond to the actual number of banks (corresponding to "the first number of ranks") present on the actual devices. "The invention [contemplates] a method for using an M bank memory device in a computer system that has N bank addressing, comprising the steps of: [¶] a) inputting address signals from a system controller to a logic circuit, said address signals including a number of address inputs and a number of bank address signals; [¶] b) re-mapping at least one of said address inputs as an additional bank address signal; and [¶] c) providing said address inputs, bank address signals and said additional bank address signal as inputs to the memory device." (*2:40-65*)
Claim 2	
2. The memory module of claim 1, wherein the first command signal is a refresh signal or a precharge signal.	"The memory controller 14 provides address, data and bus control signals for interfacing the CPU 12 and the memory module 20. The memory controller 14 includes logic for addressing, receiving, writing and refreshing data in the plurality of memory devices 22 on the module 20." (*3:30-34*)
Claim 3	
3. The memory module of claim 1, wherein the memory devices comprise dynamic random-access memory (DRAM) devices.	"The present invention can be used with many different size DRAM devices with different configurations (e.g. 2M×32, 4M×16, 8M×8, 16M×4 in the 64 Megabit example)." (*8:53-56*) As noted above, the memory devices of the claim correspond to the memory cells or arrays of the memory chips of Dell. These cells are DRAM cells.

US PATENT 7,289,386	DISCLOSURE US PATENT 6,209,074 (DELL) AND BARR
Claim 4	
4. The memory module of claim 1, wherein the set of input control signals comprises a first number of chip-select signals and wherein the set of output control signals comprises a second number of chip-select signals, wherein the first number of chip-select signals is less than the second number of chip-select signals, the memory module simulating a virtual memory module having the second number of memory devices.	The chip-select signals of the claim correspond to the bank address signals of Dell. The memory module with the larger number of banks simulates the module the system is expecting with the smaller number of banks. "[A] four bank memory device 22 could still be used with a system 12 that is expecting or requires a two bank memory device, for example." (*4:48-4:51*) "As four bank devices each SDRAM 22 uses . . . two bank address signals (BA0-BA1). However, the system 12 is expecting memory chips with two banks and therefore will address the module 20 with . . . only one bank address signal BA0, with the second bank address signal BA1 missing from the system 12 address inputs to the module 20." (*8:20-27*) Further, it would have been obvious to have used a chip select signal on devices that use chip select signals in place of the bank addresses that Dell uses on devices that use bank addresses.
Claim 5	
5. The memory module of claim 1, wherein the logic element comprises an application-specific integrated circuit.	"In accordance with one aspect of the invention, the memory module 20 includes a logic circuit 24. In the embodiment, the logic circuit 24 is realized in the form of an application specific integrated circuit (ASIC)." (*3:55-58*)
Claim 6	
6. The memory module of claim 1, wherein the logic element comprises a field-programmable gate array.	Dell teaches a gate array but not a field-programmable gate array: "A suitable device for the ASIC 24 is a gate array ASIC" (3:58-59) Barr teaches the use of a field-programmable gate array to implement logic. "Field Programmable Gate Arrays (FPGAs) can be used to implement just about any hardware design. One common use is to prototype a lump of hardware that will eventually find its way into an ASIC. However, there is nothing to say that the FPGA can't remain in the final product." (Page 3, 4th paragraph) It would have been obvious to use an FPGA to implement the logic element of Dell, which was implemented as an ASIC, because FPGAs are a well understood technology with particular advantages for implementing logic. "It costs significantly more to develop an ASIC [than to develop an FPGA]" (Page 3, 4th paragraph)

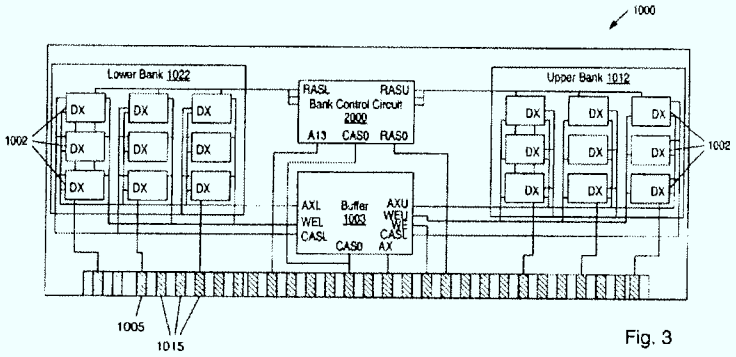
US PATENT 7,289,386	DISCLOSURE US PATENT 6,209,074 (DELL) AND BARR
Claim 7	
7. The memory module of claim 1, wherein the logic element comprises a custom-designed semiconductor device.	"In the embodiment, the logic circuit 24 is realized in the form of an application specific integrated circuit (ASIC)." (*3:56-58*) As is well known to the person of ordinary skill in the art, ASICs are semiconductor devices. Since they are by definition application specific, ASICs are custom-designed.
Claim 8	
8. The memory module of claim 1, wherein the logic element comprises a complex programmable-logic device.	Barr teaches the use of complex programmable-logic devices to implement logic. "Because CPLDs [Complex Programmable-Logic Devices] can hold larger designs than PLDs [Programmable-Logic Devices], their potential uses are more varied. They are still sometimes used for simple applications like address decoding, but more often contain high-performance control-logic or complex finite state machines. . . . Because of its less flexible internal architecture, the delay through a CPLD (measured in nanoseconds) is more predictable and usually shorter." (Page 3, 3rd paragraph) It would have been obvious to use a CPLD to implement the logic element of Dell, because, as taught by Barr, CPLDs are a well understood technology with particular advantages for implementing logic.
Claim 9	
9. The memory module of claim 1, wherein the first number of ranks is four, and the second number of ranks is two.	"For example, the system may need a two bank memory chip, but the memory module may include a memory device that is a four bank device." (*2:29-31*) "An ASIC can be configured to allow the use of a 4 bank SDRAM in a system that supports only 2 bank SDRAMS . . . ." (*4:25-28*) "[A] four bank memory device 22 could still be used with a system 12 that is expecting or requires a two bank memory device, for example." (*4:48-4:51*)
Claim 10	
10. The memory module of claim 1, wherein the first number of ranks is two, and the second number of ranks is one.	"Thus, in one general aspect of the invention, a 2N bank memory device can be used in systems having addressing for only N banks . . . ." (*8:61-63*) The limitation is met for the case of N=1.

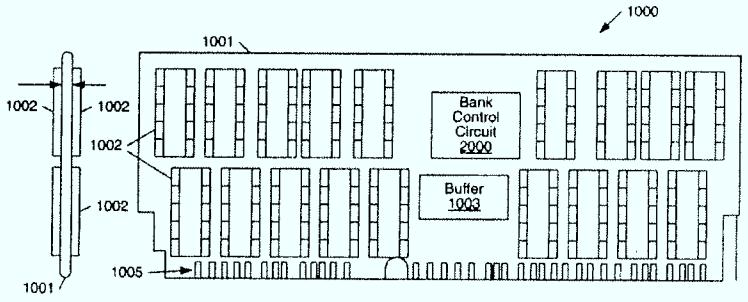
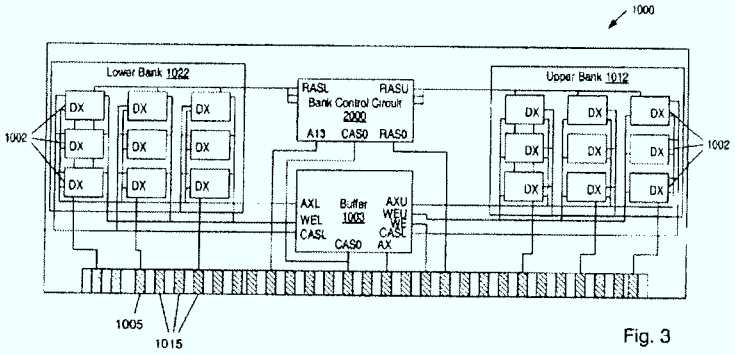
US PATENT 7,289,386	DISCLOSURE US PATENT 6,209,074 (DELL) AND BARR
Claim 11	
11. The memory module of claim 1, wherein the set of input control signals comprises two chip-select signals and an address signal and the set of output control signals comprises four chip-select signals.	"Further assume that the module 20 includes memory devices 22 of 64 Megabit SDRAM devices that are configured as four bank devices. Finally, for this example, assume the system 12 expects or requires a memory device with two bank devices. [¶] As four bank devices, each SDRAM 22 uses . . . two bank address signals (BA0 and BA1). However, the system 12 is expecting memory chips with two banks, and therefore will address the module 20 with . . . only one bank address signal BA0 . . . ." (*8:15-25*) Because the input control signal BA0 selects one of two banks it corresponds to the "two chip select signals" of the claim, which are also capable only of selecting from among two ranks. Similarly, because the output control signals BA0 and BA1 select from among four banks, they correspond to the "four chip-select signals" of the claim, which are also capable only of selecting from among four ranks.
Claim 12	
12. The memory module of claim 1, wherein the printed circuit board is mountable in a module slot of the computer system, the printed circuit board having a plurality of edge connections electrically coupled to corresponding contacts of the module slot.	"Consequently, DRAM memory is available in module form, in which a plurality of memory chips are placed on a small circuit card, which card then plugs into a memory socket connected to the computer motherboard or memory carrier card." (*1:46-50*)

APPENDIX I

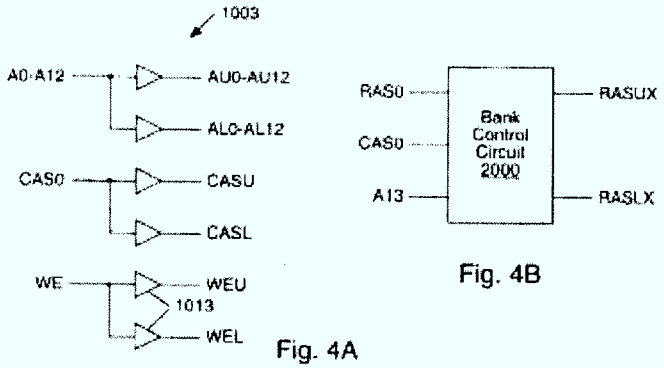
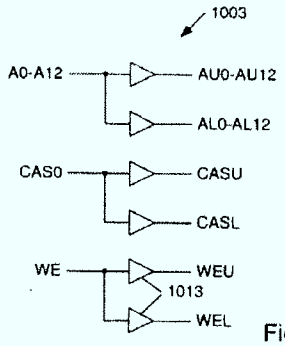
APPENDIX I – CLAIM CHART FOR WONG AND FOR WONG AND BARR

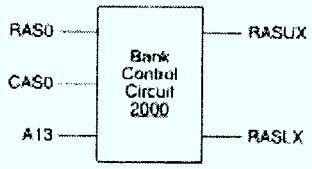
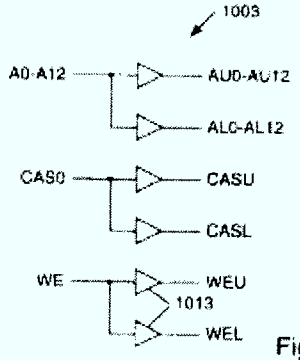
Requester submits that, under the "broadest reasonable interpretation" standard, claims 1-4, 10 and 12 of the '386 patent are anticipated by Wong. In addition, claims 5-8 are obvious in light of Wong and Barr. This Appendix provides an element-by-element comparison of the claims to the references, showing how the prior art discloses every element of the claims. (Any emphasis in quotations from the reference has been added, unless otherwise noted. Citations are to columns and lines of Wong, unless otherwise noted.)

US PATENT 7,289,386	DISCLOSURE US PATENT 6,414,868 (WONG) AND BARR
Claim 1	
1. A memory module connectable to a computer system, the memory module comprising:	"A block diagram illustrating the electrical connections associated with one embodiment of the memory module is shown in FIG. 3. Memory module 1000 includes..." *(4:13-15)*  Fig. 3 The memory module 1000 is connectable to a computer system: "[A] memory module includes a printed circuit board with a connector edge adapted for insertion in an expansion socket of a computer system." *(Abstract)* Shown in FIG. 3, "edge connector 1005 includes a plurality of electrical contact pads 1015 which convey signals between the memory module and the system memory bus. Edge connector 1005 is adapted for mounting in a socket within a computer system." *(4:19-21)*
a printed circuit board;	"Turning now to FIG. 2, a diagram illustrating components associated with a memory module 1000 is shown. In this particular embodiment, a plurality of memory elements 1002, typically DRAM (Dynamic Random Access Memory) chips, are surface mounted upon a printed circuit board (PCB) 1001." *(3:58-63)*

US PATENT 7,289,386	DISCLOSURE US PATENT 6,414,868 (WONG) AND BARR
	 Fig. 2
a plurality of memory devices coupled to the printed circuit board, the plurality of memory devices having a first number of memory devices; and	"Turning now to FIG. 2, a diagram illustrating components associated with a memory module 1000 is shown. In this particular embodiment, a plurality of memory elements 1002, typically DRAM (Dynamic Random Access Memory) chips, are surface mounted upon a printed circuit board (PCB) 1001." *(3:58-63)* FIG. 3 shows the memory module 1000 with 18 memory devices 1002.  Fig. 3
a logic element coupled to the printed circuit board,	The logic element includes the buffer 1003 and the bank control circuit 2000. (The buffer 1003 is also referred to as the buffer chip 1003 and the buffer circuit 1003.) "Also mounted on PCB 1001 are a buffer chip 1003 . . . and a bank control circuit 2000." *(4:7-9)*
the logic element receiving a set of input control signals from the computer system,	The buffer chip 1003 receives input control signals A0-A12, CAS0, and WE and the bank control circuit 2000 receives input control signals RAS0, CAS0, and A13 from a computer system to which the memory module is connected. "[A] memory module includes a printed circuit board with a connector edge adapted for insertion in an expansion socket of a computer system." *(Abstract)* "Signals passing through the edge connector include data signals,

US PATENT 7,289,386	DISCLOSURE US PATENT 6,414,868 (WONG) AND BARR
	address signals, and control signals." *(3:58 –4:2)* "In the embodiment shown, buffer chip 1003 receives a plurality of address signals, A0-A12, a CAS0 signal, and a WE signal." *(4:44-46)* "In this embodiment, bank control circuit 2000 receives input signals RAS0, CAS0, and address signal A13." *(4:56-57)* FIG. 3, reproduced above, shows the signals as being received from the computer system through electrical contact pads 1015. *(4:19-21)*
the set of input control signals corresponding to a second number of memory devices smaller than the first number of memory devices,	The input control signals, which are received from the computer system, include 14 address signals (A0-A13) and one row access strobe (RAS) signal. The memory devices, as seen by the input control signals, are addressed by 14 bits. The as-seen devices therefore have a higher capacity than the devices addressed by 13 bits that are actually used on the memory module. Therefore, for whatever capacity the module has, there would be fewer of the larger as-seen devices (the second number of devices), addressed by 14 bits, than of the smaller actual devices (the larger, first number of devices), addressed by 13 bits. In other words, the input control signal with its 14 bit address corresponds to a second (as-seen, smaller) number of devices than the first (actual) number of devices on the module. "The use of memory chips with a greater number of address inputs, and hence higher capacity, may disproportionately increase the cost of the desired memory expansion." *(1:62-65)* "Buffer 1003 receives . . . a plurality of address signals, shown as AX. Buffer circuit 1003 drives a plurality of address signals AXL and AXU, which are conveyed to the lower memory bank 1022 and upper memory bank 1012, respectively. . . . The bank control circuit 2000 is configured to receive an address signal A13 for selecting the upper and lower bank. Address signal A13, in this embodiment, is the most significant address bit of an address bus that is 14 bits wide." *(4:23-37)* "Address signal A13, in this embodiment [of FIG. 3], is the most significant address bit of an address bus that is 14 bits wide." *(4:35-37)* The remaining address bits, A0 – A12, are designated "AX" in FIG. 3. As illustrated in FIGS. 3, 4A and 4B, address bits A0-A12 go through the buffer 1003 and are used to address the memory devices, unlike A13, which is used by bank control circuit 2000 to select the upper or lower rank of the memory module 1000.

US PATENT 7,289,386	DISCLOSURE US PATENT 6,414,868 (WONG) AND BARR
	 Fig. 4A Fig. 4B
the logic element generating a set of output control signals in response to the set of input control signals, the set of output control signals corresponding to the first number of memory devices,	The buffer chip 1003 (which constitutes the logic element together with the bank control circuit) generates output control signals AU0-AU12, AL0-AL12, CASU, CASL, WEU, and WEL and the bank control circuit 2000 generates output control signals RASL0-2, and RASU0-2 in response to the input control signals A0-A13, CAS0, RAS0, and WE. These signals control and therefore correspond to all the memory devices (the "first number of memory devices") on the module. "FIG. 4A further illustrates an internal configuration of an embodiment of a buffer chip 1003. In the embodiment shown, buffer chip 1003 receives a plurality of address signals, A0-A12, a CAS0 signal, and a WE signal. The signals are passed through buffers 1013, generating corresponding signals that will be provided to an upper bank and a lower bank of memory chips." *(4:43-49)*  Fig. 4A "One embodiment of bank control circuit 2000 is shown in FIG. 4B. In this embodiment, bank control circuit 2000 receives input signals RAS0, CAS0, and address signal A13. Bank control circuit 2000 drives a plurality of RASLX and RASUX signals to the lower and upper memory banks, respectively." *(4:55-60)*

US PATENT 7,289,386	DISCLOSURE US PATENT 6,414,868 (WONG) AND BARR
	 Fig. 4B
wherein the plurality of memory devices are arranged in a first number of ranks, and the set of input control signals corresponds to a second number of ranks of memory modules, the second number of ranks less than the first number of ranks,	FIG. 3, reproduced above, shows the memory devices arranged in two ranks, the lower 1022, and the upper 1012. The term "rank" in the claim has the same meaning as the term "bank" in Wong. The input control signals include only one row access strobe (RAS) signal, and therefore correspond to one rank (the second number) of memory devices. The second number of ranks, one, is less than the first number of ranks, which is two. "[The] presence of only one RAS and one CAS signal also limits the ability to expand system memory, as a separate bank of memory typically requires at minimum either a unique RAS or unique CAS signal for each bank." (*2:3-7*) "By using the bank control circuit to enable the addition of a second memory bank, a memory expansion can be realized without the need for higher capacity memory chips, which may result in an advantageous cost savings." (*2:33-37*)
wherein the logic element further responds to a first command signal from the computer system by generating a second command signal transmitted to the plurality of memory devices, the first command signal corresponding to the second number of ranks and the second command signal corresponding to the first number of ranks.	The signals received by the memory module 1000 from the computer system include a WE (write) signal. (The '386 patent identifies write and refresh signals, among others, as command signals. *See, e.g.*, '386 patent, col. 8, lines 45-55) As shown in FIG. 4A, below, the buffer 1003 (which is part of the logic element) receives one WE signal (corresponding to one, the second number of ranks) and generates two write signals, WEU and WEL (corresponding to two, the first number of ranks).  Fig. 4A

US PATENT 7,289,386	DISCLOSURE US PATENT 6,414,868 (WONG) AND BARR
	In addition, the bank control circuit 2000 receives "receive a row address strobe (RAS) signal and a column address strobe signal, and at least one address signal, wherein said bank control circuit is configured to selectively provide at least one corresponding RAS output signal to either said upper bank of memory chips or said lower bank of memory chips depending upon said address signal to thereby allow either said lower bank of memory chips or said upper bank of memory chips to be selectively accessed during a given memory operation" (*Claim 1*) See, also, the consideration of claim 2, immediately below, in which the claim language is applied to a refresh command.
Claim 2	
2. The memory module of claim 1, wherein the first command signal is a refresh signal or a precharge signal.	In the case of a CBR (Columns Before Rows) refresh command, the first command signal corresponding to one rank (the second number of ranks) are the input CAS and RAS signals, and the second command signal, the refresh commands to both ranks on the module, are the CASU, CASL, RASU, and RASL signals, corresponding to two ranks, referred to in Wong as the upper and lower banks. "One embodiment of bank control circuit 2000 is shown in FIG. 4B. In this embodiment, bank control circuit 2000 receives input signals RAS0, CAS0, and address signal A13. Bank control circuit 2000 drives a plurality of RASLX and RASUX signals to the lower and upper memory banks, respectively. . . . Another combination of inputs will assert all RASUX and RASLX signals in order to perform a CBR (Columns before Rows) refresh cycle." *(4:55-65)* "[Said] bank control circuit is configured to receive a Column Address Strobe (CAS) signal, and, wherein said bank control circuit is configured such that if said CAS signal is asserted before said RAS signal, both of said memory banks will be selected for a CBR (CAS before RAS) refresh cycle by asserting said RASL signals and said RASU signals." (*Claim 10*)
Claim 3	
3. The memory module of claim 1, wherein the memory devices comprise dynamic random-access memory (DRAM) devices.	"[A] memory module includes a printed circuit board with a connector edge adapted for insertion in an expansion socket of a computer system. Mounted upon the circuit board is a plurality of memory chips, typically Dynamic Random Access Memory (DRAM) chips, which make up an upper bank and a lower bank of memory." *(Abstract; see also 2:19-22; 3:59-63)*

US PATENT 7,289,386	DISCLOSURE US PATENT 6,414,868 (WONG) AND BARR
Claim 4	
4. The memory module of claim 1, wherein the set of input control signals comprises a first number of chip-select signals and wherein the set of output control signals comprises a second number of chip-select signals, wherein the first number of chip-select signals is less than the second number of chip-select signals, the memory module simulating a virtual memory module having the second number of memory devices.	The chip select signals of the claim read on the RAS signals of Wong. The '386 patent uses the term "chip select signal" broadly. (*See*, '386 patent, col. 2, lines 36-38: "control signals include, but are not limited to, rank-select signals, also called chip-select signals.") The input control signals received by the module 1000 include only one row address strobe signal (called RAS0). The RAS0 signal is a rank select signal. The "first number of chip select signals" is therefore one. The bank control circuit 2000 outputs two RAS signals, RASLX and RASUX signals. The "set of output control signals" therefore comprises two (the "second number") of chip select signals. The two RASLX and RASUX signals are rank select signals, and therefore chip select, signals, just like the one input RAS0 signal. The "second number of memory devices" is the number of devices that corresponds to the "set of input control signals," as recited earlier, in claim 1. Because the memory module 1000 responds to the input control signals it receives, it necessarily simulates a (virtual) memory module having a number of memory devices that corresponds to the input control signals the module receives. FIGS. 3 and 4B, reproduced above, show that the input control signals include one input RAS signal (RAS0) and two output RAS signals (RASUX and RASLX). The first number, one, is less than the second number, two, of rank select (chip select) signals. "In this embodiment, bank control circuit 2000 receives input signals RAS0, CAS0, and address signal A13. Bank control circuit 2000 drives a plurality of RASLX and RASUX signals to the lower and upper memory banks, respectively." *(4:56-60)* "By using the bank control circuit to enable the addition of a second memory bank, a memory expansion can be realized without the need for higher capacity memory chips, which may result in an advantageous cost savings." *(2:34-38)*
Claim 5	
5. The memory module of claim 1, wherein the logic element comprises an application-specific integrated circuit.	The circuitry described in Wong is specific to Wong's application and so comprises an application-specific circuit. It would have been obvious at the time to implement the logic element found in Wong in an integrated circuit. Barr teaches advantages of the use of an application-specific integrated circuit (ASIC). "It costs significantly more to develop an ASIC, but the cost per chip may be lower in the long run." *(Barr, page 3, 4th paragraph)*

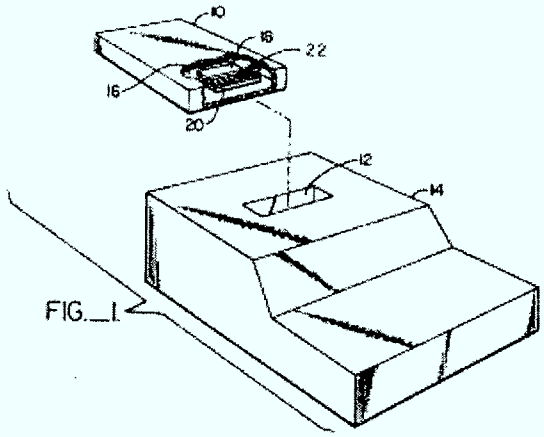
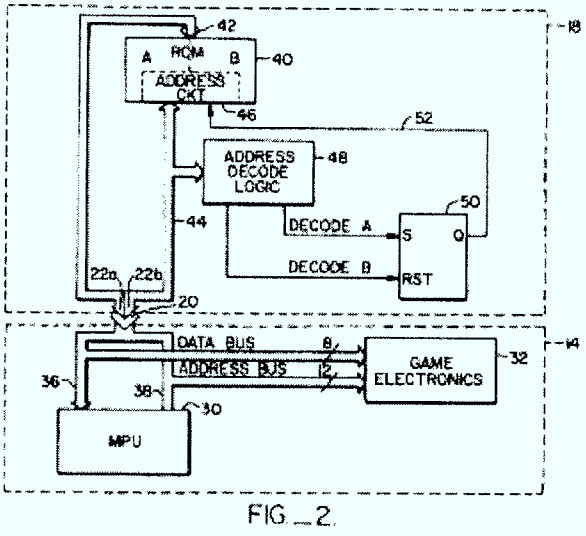
US PATENT 7,289,386	DISCLOSURE US PATENT 6,414,868 (WONG) AND BARR
Claim 6	
6. The memory module of claim 1, wherein the logic element comprises a field-programmable gate array.	Barr teaches the use of a field-programmable gate array to implement logic. "Field Programmable Gate Arrays (FPGAs) can be used to implement just about any hardware design. One common use is to prototype a lump of hardware that will eventually find its way into an ASIC. However, there is nothing to say that the FPGA can't remain in the final product." (*Barr, page 3, 4th paragraph*) It would have been obvious to use an FPGA to implement the logic element of Wong, because, as taught by Barr, FPGAs are a well understood technology with particular advantages for implementing logic. "FPGAs are more flexible than complex programmable-logic devices." (*Barr, page 8*) "It costs significantly more to develop an ASIC [than to develop an FPGA] . . . ." (*Barr, page 3, 4th paragraph*)
Claim 7	
7. The memory module of claim 1, wherein the logic element comprises a custom-designed semiconductor device.	The circuitry described in Wong is a custom design for Wong's application. It would have been obvious at the time to implement the logic element found in Wong in a custom-designed semiconductor device, e.g., an ASIC. Barr teaches advantages of the use of an application-specific integrated circuit (ASIC). "It costs significantly more to develop an ASIC, but the cost per chip may be lower in the long run." (*Barr, page 3, 4th paragraph*)
Claim 8	
8. The memory module of claim 1, wherein the logic element comprises a complex programmable-logic device.	Barr teaches the use of complex programmable-logic devices to implement logic. "Because CPLDs [Complex Programmable-Logic Devices] can hold larger designs than PLDs [Programmable-Logic Devices], their potential uses are more varied. They are still sometimes used for simple applications like address decoding, but more often contain high-performance control-logic or complex finite state machines. At the high-end (in terms of numbers of gates), there is also a lot of overlap in potential applications with FPGAs. Traditionally, CPLDs have been chosen over FPGAs whenever high-performance logic is required. Because of its less flexible internal architecture, the delay through a CPLD (measured in nanoseconds) is more predictable and usually shorter." (*Barr, page 3, 3rd paragraph*) It would have been obvious to use a CPLD to implement the logic element of Wong, because, as taught by Barr, CPLDs are a well understood technology with particular advantages for implementing

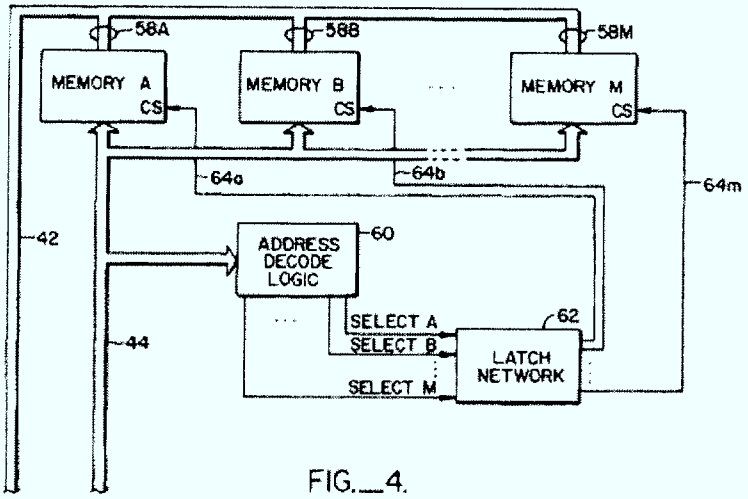
US PATENT 7,289,386	DISCLOSURE US PATENT 6,414,868 (WONG) AND BARR
	logic: "CPLDs are faster and introduce more predictable delays than FPGAs." (*Barr, page 8*)
Claim 10	
10. The memory module of claim 1, wherein the first number of ranks is two, and the second number of ranks is one.	FIG. 3, reproduced above, shows the memory devices arranged in two ranks, the lower 1022, and the upper 1012; and two is the first number of ranks. The second number of ranks is one, because there is only one RAS signal among the input control signals. "[The] presence of only one RAS and one CAS signal also limits the ability to expand system memory, as a separate bank of memory typically requires at minimum either a unique RAS or unique CAS signal for each bank." (*2:3-7*)
Claim 12	
12. The memory module of claim 1, wherein the printed circuit board is mountable in a module slot of the computer system, the printed circuit board having a plurality of edge connections electrically coupled to corresponding contacts of the module slot.	"[A] memory module includes a printed circuit board with a connector edge adapted for insertion in an expansion socket of a computer system." *(Abstract)* "On one edge of the printed circuit board is an edge connector 1005, which includes a plurality of electrical contact pads." *(3:63-65)*

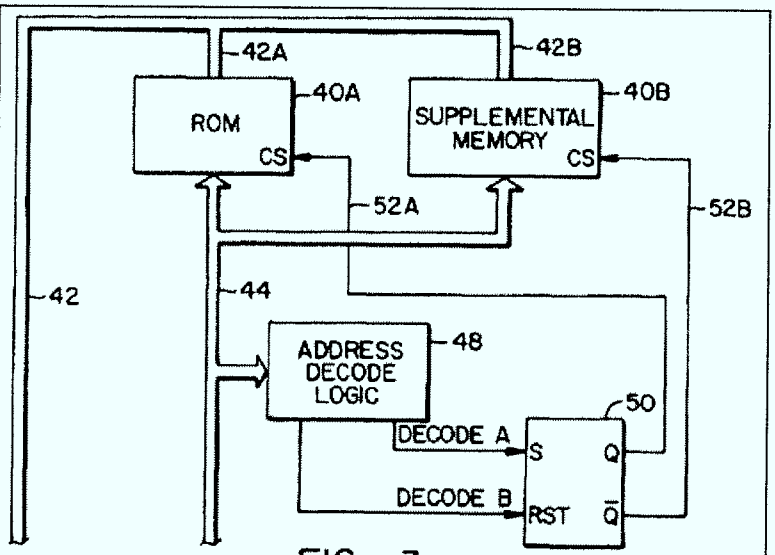
APPENDIX J

APPENDIX J – CLAIM CHART FOR NIELSEN AND FOR NIELSEN AND BARR

Requester submits that claims 1, 3, 10, and 12 are anticipated by Nielsen, that claims 1, 3, 10, and 12 are obvious in light of Nielsen, and that claims 5-8 are obvious in light of Nielsen and Barr. This Appendix provides an element-by-element comparison of claims 1, 3, 5-8, 10, and 12 to the Neilsen reference and, as applicable, the Barr reference, showing how the references discloses every element of these claims. (Any emphasis in quotations from the reference has been added, unless otherwise noted. Citations are to columns and lines of Nielsen, unless otherwise noted.)

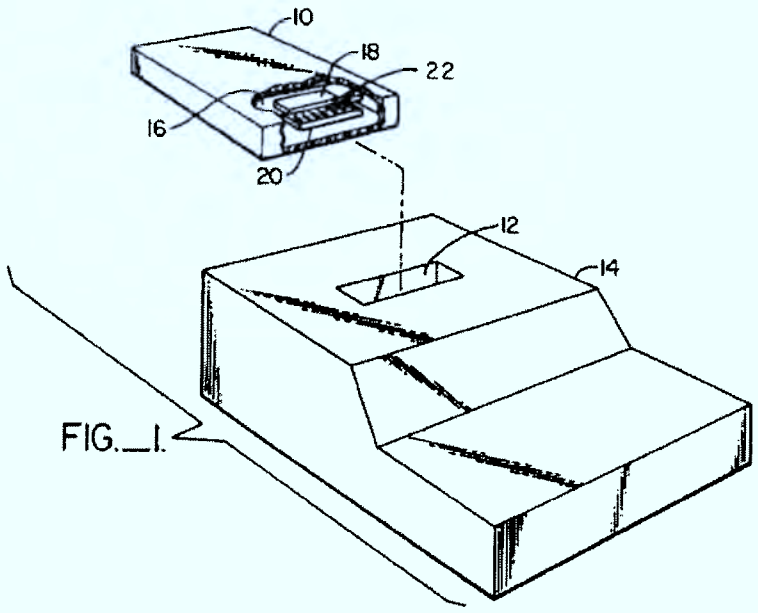
US PATENT 7,289,386	DISCLOSURE OF US PATENT 4,368,515 (NIELSEN) AND BARR
Claim 1	
1. A memory module connectable to a computer system, the memory module comprising:	The module is a cartridge 10 that contains memory and is connectable to console 14, which is a computer system by virtue of its microprocessor unit (MPU) 30. (*See, e.g., 1:26-49*)  FIG. 1  FIG. 2 "FIG. 1 is a perspective view of a commercial video game system that

	includes a cartridge 10 which is designed to be removably inserted in a socket 12 of a game console 14." (*3:26-29*) "Referring to FIG. 2, the electronics housed within console 14 is schematically illustrated as including a microprocessor unit (MPU) 30 which functions to maintain game play control over game electronics 32. . . . Specifically, MPU 30 and game electronics 32 comprise apparatus for generating moving objects for a video game display" (*3:36-44*)
a printed circuit board;	"The cartridge . . . contains a circuit board 16 which carries an electronic microcircuit 18." (*3:29-31*)
a plurality of memory devices coupled to the printed circuit board, the plurality of memory devices having a first number of memory devices; and	"[In] FIG. 4, a number of memory devices, memories A-M, are provided, having their respective data output lines 58A-58M connected in parallel and to the output lines 42." (*5:53-57*) In FIG. 4, the "first number of memory devices" is m.  FIG. 4. The memory and circuits illustrated on FIG. 4 are an example of microcircuit 18 on the printed circuit board 16. *See* FIG. 2, reproduced above. Similarly, in the embodiment of FIG. 3, there are two memory devices, ROM 40A and supplemental memory 40B, so the "first number of memory devices" is two.

	 FIG._3.
a logic element coupled to the printed circuit board,	"According to the present invention, supplemental address decode logic [60] is coupled between the address bus and the memory element." (*2:19-21*) The logic element of FIG. 4, reproduced above, includes address decode logic 60 and network latch 62. The logic element of FIG. 3, reproduced above, includes address decode logic 48 and flip-flop 50.
the logic element receiving a set of input control signals from the computer system,	The address decode logics 48 and 60 of FIGS. 3 and 4, respectively, reproduced above, receive input control signals, more specifically addresses signals, over address bus 44. "The address decode logic monitors the address signals communicated on the address bus . . . ." (*2:21-23*) "Each 12-bit address conducted on the address signal lines 44 is also applied to the address decode logic 48" (*4:28-30*) "The address signal lines 44 are coupled . . . to an address decode logic 60 that monitors the address signals communicated on the address signal lines 44." (*5:56-61*)
the logic element generating a set of output control signals in response to the set of input control signals, the set of output control signals corresponding to the first number of memory devices,	As shown by FIG. 3, in the embodiment of FIG. 3, the address decode logic 48 and flip-flop 50 generate output control signals, namely CS (chip select) signals 52A and 52B, that are transmitted to the memory devices 40A and 40B. The two CS signals correspond to the first number of devices, which for FIG. 3 is two. (*See 5:21-44*) As shown by FIG. 4, in the embodiment of FIG. 4, the address decode logic 60 and latch network 62 generate output control signals, namely CS (chip select) signals 64a-64m, that are transmitted to the m memory devices memories A-M. The m CS signals correspond to the first number of devices, which for FIG. 4 is m. (*See 5:52 – 6:4*)

wherein the plurality of memory devices are arranged in a first number of ranks, and the set of input control signals corresponds to a second number of ranks of memory modules, the second number of ranks less than the first number of ranks,	The plurality of memory devices are the devices on the printed circuit board 16 in the cartridge 10. As shown by FIG. 3, in the embodiment of FIG. 3, the plurality of memory devices are memory devices 40A and 40B, which are arranged in two ranks of one device each. Two is the second number of ranks. Each device is a rank because it has its own chip select signal 52A or 52B, which it does not share with any other memory device. Similarly, as shown by FIG. 4, in the embodiment of FIG. 4, the plurality of memory devices are memory devices A-M, which are arranged in m ranks of one device each. Each device is a rank because it has its own chip select signal, one of 64a-64m, which it does not share with any other memory device. The set of input control signals, the address signals on address bus 44, correspond to one rank, i.a., because no chip select signals are transmitted over the address bus 44 or otherwise from the console 14 to the cartridge 10. One is the first number of ranks. In some embodiments, one chip select may be used; in such embodiments, the set of input control signals still corresponds to one rank. (*See FIGS. 5 and 6, signal CS-bar*) The address lines for the cartridge 10 provide signal lines for only a single memory: "Accordingly, it is desirable to increase the number of addressable memory locations without changing the number of address signal lines in the current connector." (*2:3-5*) (*emphasis added*)
wherein the logic element further responds to a first command signal from the computer system by generating a second command signal transmitted to the plurality of memory devices, the first command signal corresponding to the second number of ranks and the second command signal corresponding to the first number of ranks.	The command signal could be a read signal or a write signal. The logic element can generate a read/write signal for a RAM memory device. "The microcircuit chip housed within the cartridge can be structured to include the program ROM and a random-access memory (RAM), the invention being used to select which memory (i.e., program ROM or RAM) will be accessed, as well as generate the read/write signal required by RAM." (*2:61-67*) For a ROM memory device, the only operation is a read operation. The operation occurs when the address signal is presented to the device. As described above, the address decode logic of FIGS. 3 and 4 receives and responds to the address signals on the address bus by generating a chip select signal that selects the memory device to which the command is applied. The first command signal received from the console 14 corresponds to one rank (the second number of ranks) because it includes at most one chip select signal (*see, e.g., FIG. 5*) and is generated to access only one rank. The second command signal is the one to which only one of the memory devices responds – the one selected by the chip select generated by the flip-flop 50 or the latch network 62. This second command signal corresponds to the second number of ranks, because this second command signal includes a chip select signal that corresponds to – because it selects from among – the second number

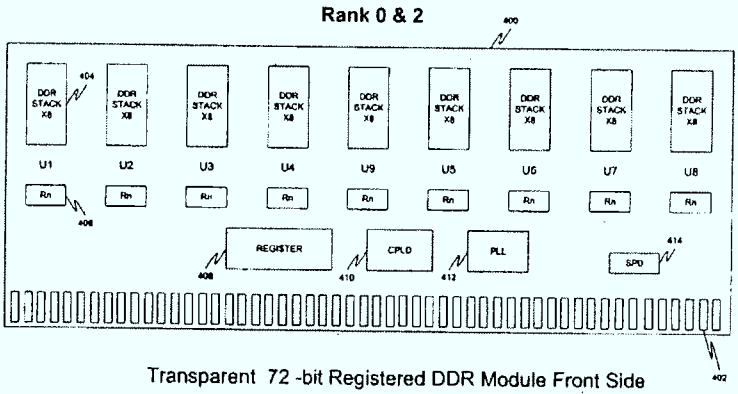
	of ranks and is generated to selected from among the second number of ranks.
Claim 3	
3. The memory module of claim 1, wherein the memory devices comprise dynamic random-access memory (DRAM) devices.	It would have been obvious to use DRAM in the embodiments of Nielsen, because DRAM is a kind of RAM. "As FIG. 3 illustrates, a ROM 40A is supplemented with a supplemental memory 40B (which could be either ROM or RAM)," (*5:5-7*)
Claim 5	
5. The memory module of claim 1, wherein the logic element comprises an application-specific integrated circuit.	The circuitry described in Nielsen is specific to Nielsen's application and so comprises an application-specific circuit. It would have been obvious at the time to implement the logic element found in Nielsen in an integrated circuit. Barr teaches advantages of the use of an application-specific integrated circuit (ASIC). "It costs significantly more to develop an ASIC, but the cost per chip may be lower in the long run." (*Page 3, 4th paragraph*)
Claim 6	
6. The memory module of claim 1, wherein the logic element comprises a field-programmable gate array.	Barr teaches the use of a field-programmable gate array to implement logic. "Field Programmable Gate Arrays (FPGAs) can be used to implement just about any hardware design. One common use is to prototype a lump of hardware that will eventually find its way into an ASIC. However, there is nothing to say that the FPGA can't remain in the final product." (*Page 3, 4th paragraph*) It would have been obvious to use an FPGA to implement the logic element of Nielsen, because, as taught by Barr, FPGAs are a well understood technology with particular advantages for implementing logic. "FPGAs are more flexible than complex programmable-logic devices." (Page 8) "It costs significantly more to develop an ASIC [than to develop an FPGA] . . . ." (*Page 3, 4th paragraph*)
Claim 7	
7. The memory module of claim 1, wherein the logic element comprises a custom-designed semiconductor device.	The circuitry described in Nielsen is a custom design for Nielsen's application. It would have been obvious at the time to implement the logic element found in Nielsen in a custom-designed semiconductor device, e.g., an ASIC. Barr teaches advantages of the use of an application-specific integrated circuit (ASIC). "It costs significantly more to develop an ASIC, but the cost per chip may be lower in the long run." (*Barr, page 3, 4th paragraph*)

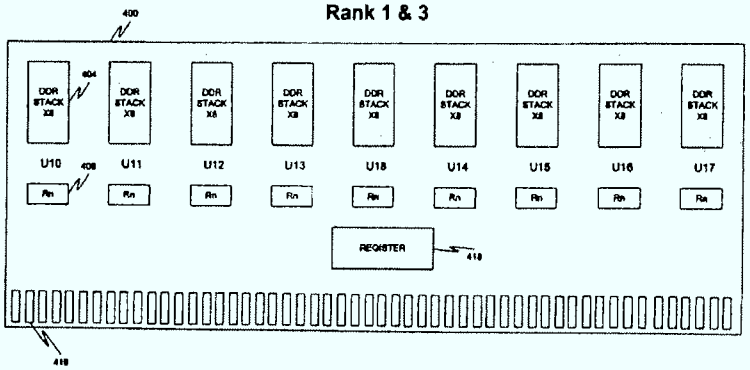
Claim 8	
8. The memory module of claim 1, wherein the logic element comprises a complex programmable-logic device.	Barr teaches the use of complex programmable-logic devices to implement logic. "Because CPLDs [Complex Programmable-Logic Devices] can hold larger designs than PLDs [Programmable-Logic Devices], their potential uses are more varied. They are still sometimes used for simple applications like address decoding, but more often contain high-performance control-logic or complex finite state machines. At the high-end (in terms of numbers of gates), there is also a lot of overlap in potential applications with FPGAs. Traditionally, CPLDs have been chosen over FPGAs whenever high-performance logic is required. Because of its less flexible internal architecture, the delay through a CPLD (measured in nanoseconds) is more predictable and usually shorter." (*Page 3, 3rd paragraph*) It would have been obvious to use a CPLD to implement the logic element of Nielsen, because, as taught by Barr, CPLDs are a well understood technology with particular advantages for implementing logic: "CPLDs are faster and introduce more predictable delays than FPGAs." (*Page 8*)
Claim 10	
10. The memory module of claim 1, wherein the first number of ranks is two, and the second number of ranks is one.	This case is shown in FIG. 3, which was addressed in reference to claim 1, above.
Claim 12	
12. The memory module of claim 1, wherein the printed circuit board is mountable in a module slot of the computer system, the printed circuit board having a plurality of edge connections electrically coupled to corresponding contacts of the module slot.	The "cartridge 10 [is] designed to be removably inserted in a socket 12 of a game console 14. . . . Circuit board 16 includes a connector portion 20 with a number of printed circuit leads 22 thereon that establish electrical connection between the microcircuit 18 carried by the circuit board 16 and a connector contained within console 14." (*3:27-35*)  FIG. 1.

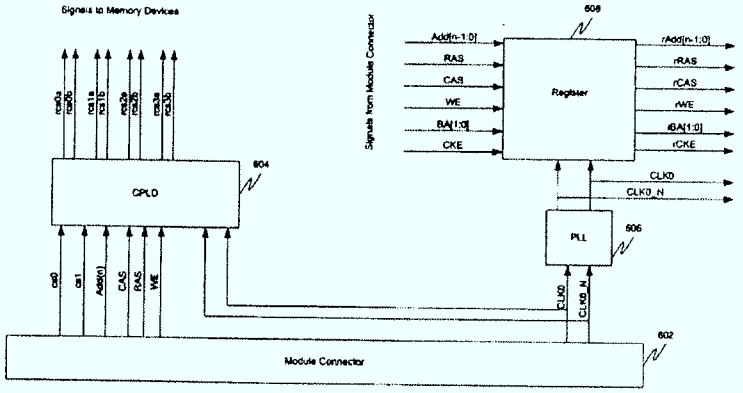
APPENDIX K

APPENDIX K – CLAIM CHART FOR AMIDI

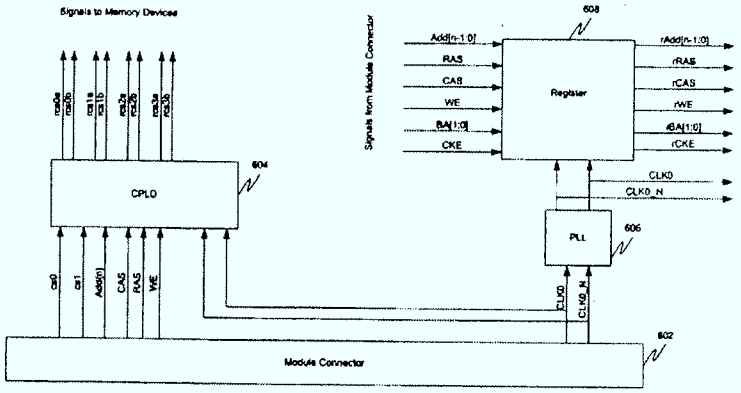
Requester submits that, under the "broadest reasonable interpretation" standard, claims 1-9 and 11-13 of the '386 patent are anticipated by Amidi. This Appendix provides an element-by-element comparison of the claims to the reference, showing how the prior art discloses every element of the claims. (Any emphasis in quotations from the reference has been added, unless otherwise noted.)

US PATENT 7,289,386	DISCLOSURE US PUBLICATION 2006/0117152 (AMIDI)
Claim 1	
1. A memory module connectable to a computer system, the memory module comprising:	"FIG. 4A illustrates a schematic diagram of the front side of a transparent 72 bit registered DDR module 400 . . . The memory module 400 includes 92 contact pins 402 on the front side for connecting with a memory socket *(paragraph 37)* "FIG. 4B illustrates a schematic diagram of the back side of the transparent 72 bit registered DDR module 400..." *(paragraph 42)*  Transparent 72 -bit Registered DDR Module Front Side FIG. 4A

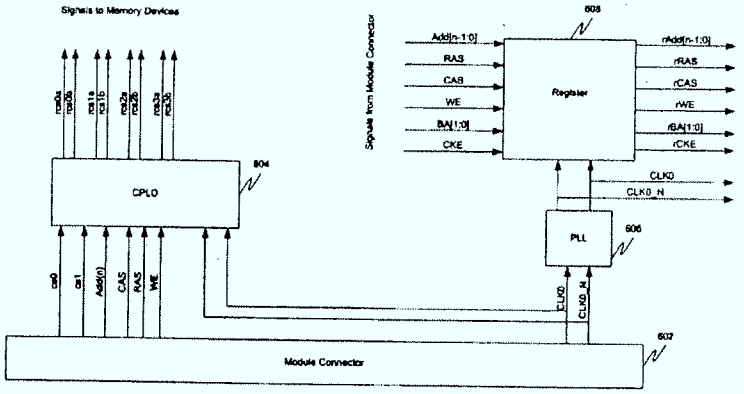
US PATENT 7,289,386	DISCLOSURE US PUBLICATION 2006/0117152 (AMIDI)
	 Transparent 72 -bit Registered DDR Module Back Side FIG.4B
a printed circuit board;	A person skilled in the art would have recognized the memory module 400 illustrated in FIGS. 4A and 4B, reproduced above, as being a printed circuit board.
a plurality of memory devices coupled to the printed circuit board, the plurality of memory devices having a first number of memory devices; and	The module of FIGS. 4A and 4B has four ranks and a total of 36 memory devices U1-U36 (nine devices per rank). Thirty-six is the "first number of memory devices." "Memory devices 404 (U1 through U9) are mounted on a surface of the front side of the memory module 400. Memory devices 404 (U19 through U27) are respectively stacked on memory devices 404 (U1 through U9). Therefore the front side of the four rank memory modules 400 includes two ranks (rank 0 and rank 2)." *(paragraph 37)* "Memory devices 404 (U10 through U18) are mounted on a surface of the back side of the memory module 400. Memory devices 404 (U28 through U36) are respectively stacked on memory devices 404 (U10 through U18). Therefore the back side of the four rank memory module 400 includes two ranks (rank 1 and rank 3)." *(paragraph 42)*
a logic element coupled to the printed circuit board,	The "logic element" of the claim reads on the emulator, which can include a complex programmable logic device and registers, is coupled to the printed circuit board (the memory module). "A transparent four rank memory module has a front side and a back side. The front side has a third memory rank stacked on a first memory rank. The back side has a fourth memory rank stacked on a second memory rank. An emulator coupled to the memory module activates and controls one individual memory rank from either the first memory rank, the second memory rank, the third memory rank, or the fourth memory rank based on the signals received from a memory controller." (*paragraph 12*) "The CPLD 410 emulates a two rank memory module on the four

US PATENT 7,289,386	DISCLOSURE US PUBLICATION 2006/0117152 (AMIDI)
	rank memory module 400." (*paragraph 41*) "The register 408 is used to synchronize the incoming address and control signals with respect to differential clock signals 208..." *(paragraph 38)*
the logic element receiving a set of input control signals from the computer system,	The memory module has contact pins on a connector for receiving input signals from a computer system. "The memory module 400 includes 92 contact pins 402 on the front side for connecting with a memory socket (not shown)." *(page 14, paragraph 37)* Pins 402 are shown in FIG. 4A, reproduced above. The module connector 602 is shown in FIG. 6A:  Row Address Decoding FIG.6A The CPLD 604 receives control signals, including the control signals cs0, cs1, Add(n), RAS, and CAS. The register 608 receives control signals including signals Add(n-1:0), RAS, and CAS. (*Fig. 6A*) The signals correspond to a total of 18 memory devices in the 512 MByte memory module. The input control signals cs0 and cs1 are the chip select signals for the 18 memory devices the computer system thinks are in the module attached to the main board. The logic element would not modify the chip select and address information (the input control signals) if there were actually 18 memory devices.
the set of input control signals corresponding to a second number of memory devices smaller than the first number of memory devices,	The control signals received by the module correspond to a smaller number of larger devices than the first number of memory devices (36 devices) that are actually present on the module. The designs described by Amidi provide a four-rank module with 36 memory devices that provides the same memory capacity as a two-rank memory module with 18 memory devices. That the input control signals include only two chip select signals (described in paragraph

US PATENT 7,289,386	DISCLOSURE US PUBLICATION 2006/0117152 (AMIDI)
	10, quoted below) means that the input control signals correspond to 18 larger 256 Mbit devices (18 is the second number of devices), which is smaller than 36, the number of smaller 128 Mbit devices actually on the module. "[0006] One method of building a 512 M Byte standard memory module with ECC (64-bit data plus 8-bit ECC=72-bit) includes using 256 Mbit density families of 32M×8 [in two ranks of 256 MByte each]. "[0007] Therefore, a two rank memory module with 18 device placements will achieve the 512M Byte density. . . ." "[0008] Because memory devices with lower densities are cheaper and more readily available, it may be advantageous to build the above same density memory module using lower densities devices. However, in order to achieve a density of, for example, 512 M Bytes using 128 Mbit density of 16M×8 instead, the memory module needs four ranks configured as [four ranks of nine devices each, or 128 Mbyte per rank.] "[0009] In order to achieve the above configuration, 4 rows of 9 devices each, totaling 36 placements, are required. As mentioned above, on a standard 184-pin DDR memory module, there is only enough space for 18 TSSOP [Thin-Shrink Small Outline Package] devices. "[0010] The only solution would be, to stack two memory devices together to achieve an extra rank on the same placement space. Although this would solve the placement problem . . . , the memory module would still possess four memory ranks. As explained earlier, all standard memory modules have only two chip select signals per memory socket routed. Therefore, such memory module would not be viable." "[0011] A need therefore exists for a transparent four rank memory module fitting into a memory socket having two chip select signals routed. A primary purpose of the present invention is to solve these needs and provide further, related advantages." (*paragraphs 6-11*)
the logic element generating a set of output control signals in response to the set of input control signals, the set of output control signals corresponding to the first number of memory devices,	The emulator logic (CPLD and registers) necessarily generates output control signals that correspond to the first number of memory devices, because the first number of devices is how many memory devices are actually on the module, and the CPLD and registers control all those devices. "An emulator coupled to the memory module activates and controls one individual memory rank from either the first memory rank, the second memory rank, the third memory rank, or the fourth memory rank based on the signals received from a memory controller." (*Abstract*) For example, the CPLD 604 generates output control signals rcs0,

US PATENT 7,289,386	DISCLOSURE US PUBLICATION 2006/0117152 (AMIDI)
	rcs1, rcs2, and rcs3 and the register 608 generates output control signals including rAdd[n-1:0], rRAS, and rCAS. The output control signals rcs0, rcs1, rcs2, and rcs3 are the chip select signals for the 36 memory devices, corresponding to the claimed "first number of memory devices," that are actually attached to the memory module. "CPLD 604 also ensures that all commands for a two rank memory module conveyed by the module connector 602 are also performed on the four rank memory modules. For example, CPLD 604 generates rcs2 and rcs3, besides rcs0 and rcs1 off of CS0, CS1 and Add(n) from the memory controller side." *(paragraph 52)*  Row Address Decoding FIG.6A In other words, the logic on a four-rank module emulates a two-rank module, so that the module can be used in a system that generates signals for a standard two-rank module, The CPLD translates from the memory controller's understanding of the number of ranks or devices, corresponding to a smaller virtual (emulated) second number of ranks or devices, to the actual output signals sent to the devices on the memory module, corresponding to the larger actual first number of ranks or devices. "Standard memory modules . . . have either one rank or two rank of memory devices." *(paragraph 4)* "Therefore, a two rank memory module with 18 device placements will achieve the 512M Byte density. [¶] Because memory devices with lower densities are cheaper and more readily available, it may be advantageous to build the above same density memory module using lower densities devices. However, in order to achieve a density of . . . 512 M Bytes using 128 Mbit density of 16M×8 instead, the memory module needs four ranks . . . ." *(paragraphs 7-8)* "The CPLD 410 emulates a two rank memory module on the four rank memory module 400. CPLD 410 allows a system having a memory socket with only two chip select signals routed to interface

US PATENT 7,289,386	DISCLOSURE US PUBLICATION 2006/0117152 (AMIDI)
	with a four rank memory module where typically a two rank memory module couples with the memory socket. . . . The algorithm of CPLD 410 is further described in [FIG.] 7." *(paragraph 41)* "FIG. 7 illustrates a method for emulating a two rank memory module with a four rank memory module . . . At 702, the memory module receives a command signal and address signal from a memory controller. The memory controller addressed the command signal and address signal to a two rank memory module. The command signal includes CS0, CS1, CKE, CAS, RAS, and WE. The address signal includes Add(n). At 704, the CPLD of the four rank memory module determines which rank should be active based on the address and command signals (cs0, cs1, and Add(n))." *(paragraph 62)* "The CPLD 410 [emulates] a two rank based on 256 Mbit DDR SDRAM . . . memory module with a four rank based on 128 Mbit DDR SDRAM . . . memory module. *(paragraph 49)*
wherein the plurality of memory devices are arranged in a first number of ranks, and the set of input control signals corresponds to a second number of ranks of memory modules, the second number of ranks less than the first number of ranks,	The input control signals include only two input chip select signals, cs0 and cs1, and correspond to two ranks of memory. The first number of ranks is four, which is the actual (first) number of ranks implemented in the memory module. "[The] CPLD of the four rank memory module determines which rank should be active based on the address and command signals (cs0, cs1, and Add(n))." *(paragraph 62)* "Register 608 of a four rank memory module emulating a two rank memory module..." *(paragraph 59)* "A need therefore exists for a transparent four rank memory module fitting into a memory socket having two chip select signals routed." *(paragraph 11)* "1. A memory module . . . comprising: . . . an emulator coupled to the memory module controlling signals to said first memory rank, said second memory rank, said third memory rank, and said fourth memory rank. . . . . "4. The memory module of claim 1 wherein said emulator electrically receives said first chip select signal, said second chip select signal, a set of command and/or control signals, a memory address signal, and a clock signal. "5. The memory module of claim 4 wherein said emulator activates a memory rank based on said first chip select signal, said second chip select signal, said memory address signal." (*Claims 1, 4 and 5*)
wherein the logic element further responds to a first command signal from the computer system by generating a	Any first command signal received from the computer system (i.e., received by what Amidi refers to as a "transparent four rank module") necessarily corresponds to the second number of ranks, which is two,

US PATENT 7,289,386	DISCLOSURE US PUBLICATION 2006/0117152 (AMIDI)
second command signal transmitted to the plurality of memory devices, the first command signal corresponding to the second number of ranks and the second command signal corresponding to the first number of ranks.	because that is the number of ranks that the module appears to have, and for which the computer system (more particularly, the memory controller 104, *see* FIG. 1) is providing command signals. "The CPLD 410 determines which rank from the four ranks to activate based upon the address and command signals from a memory controller coupled to the memory module 410." *(paragraph 41)* "CPLD 604 also ensures that all commands for a two rank memory module conveyed by the module connector 602 are also performed on the four rank memory modules." *(paragraph 52)* The CPLD and register provide the signals received by the memory devices, including the command signals. Thus, second command signals, generated by the emulator (CPLD and register), necessarily correspond to the first number of ranks, which is four ranks, because that is the number of ranks actually present on the transparent four-rank module. This is illustrated in FIG. 6A, reproduced below. (Note the notations "Signals to Memory Devices".)  Row Address Decoding FIG.6A
Claim 2	
2. The memory module of claim 1, wherein the first command signal is a refresh signal or a precharge signal.	"CPLD 604 also ensures that all commands for a two rank memory module conveyed by the module connector 602 are also performed on the four rank memory modules. . . . CPLD 604 also generates rcs2 when CS0 Auto Precharge all Banks Commands are issued... CPLD 604 also generates rcs2 when CS0 Auto Refresh Commands are issued." *(paragraph 52)* (Note that the reference to 'banks' in the 'Precharge all Banks' command refers to the internal banks of the DRAM and not to what are referred to as 'ranks' in the '386 patent.)

US PATENT 7,289,386	DISCLOSURE US PUBLICATION 2006/0117152 (AMIDI)
Claim 3	
3. The memory module of claim 1, wherein the memory devices comprise dynamic random-access memory (DRAM) devices.	Synchronous dynamic random access memory (SDRAM) is a type of dynamic random access memory (DRAM). "The CPLD 410 uses a Row Address Decoding scheme to emulate a two rank based on 256 Mbit DDR SDRAM Device Technology memory module with a four rank based on 128 Mbit DDR SDRAM Device Technology memory module." *(paragraph 49)*
Claim 4	
4. The memory module of claim 1, wherein the set of input control signals comprises a first number of chip-select signals and wherein the set of output control signals comprises a second number of chip-select signals, wherein the first number of chip-select signals is less than the second number of chip-select signals, the memory module simulating a virtual memory module having the second number of memory devices.	The input control signals include two chip select signals, cs0 and cs1, and the output control signals include four chip select signals, rcs0, rcs1, rcs2, and rcs3. Two is less than four. It should be noted that signals rcs0a and rcs0b, as shown in FIGS. 6A-B, are duplicate (fan-out) signals, and are the same as the rcs0 signal. This is also true for the fan-outs for rcs1, rcs2, and rcs3. This can be seen, for example, on the right side of FIG. 8. *(See also paragraphs 69-70)* "CPLD 604 also ensures that all commands for a two rank memory module conveyed by the module connector 602 are also performed on the four rank memory modules. For example, CPLD 604 generates rcs2 and rcs3, besides rcs0 and rcs1 off of CS0, CS1 and Add(n) from the memory controller side." *(paragraph 52)* The memory module simulates (i.e., emulates) a virtual memory module having, in one embodiment, 18 memory devices (the smaller, "second number of memory devices") in two ranks. (*paragraphs 6-11*) "The CPLD 410 emulates a two rank memory module on the four rank memory module 400. CPLD 410 allows a system having a memory socket with only two chip select signals routed to interface with a four rank memory module where typically a two rank memory module couples with the memory socket." (*paragraph 41*)
Claim 5	
5. The memory module of claim 1, wherein the logic element comprises an application-specific integrated circuit.	"In addition, those of ordinary skill in the art will recognize that devices of a less general purpose nature, such as hardwired devices, field programmable logic devices (FPLDs), including field programmable gate arrays (FPGAs) and complex programmable logic devices (CPLDs), application specific integrated circuits (ASICs), or the like, may also be used without departing from the scope and spirit of the inventive concepts disclosed herein." *(paragraph 28)*

US PATENT 7,289,386	DISCLOSURE US PUBLICATION 2006/0117152 (AMIDI)
Claim 6	
6. The memory module of claim 1, wherein the logic element comprises a field-programmable gate array.	"In addition, those of ordinary skill in the art will recognize that devices of a less general purpose nature, such as hardwired devices, field programmable logic devices (FPLDs), including field programmable gate arrays (FPGAs) and complex programmable logic devices (CPLDs), application specific integrated circuits (ASICs), or the like, may also be used without departing from the scope and spirit of the inventive concepts disclosed herein." *(paragraph 28)*
Claim 7	
7. The memory module of claim 1, wherein the logic element comprises a custom-designed semiconductor device.	"In addition, those of ordinary skill in the art will recognize that devices of a less general purpose nature, such as hardwired devices, field programmable logic devices (FPLDs), including field programmable gate arrays (FPGAs) and complex programmable logic devices (CPLDs), application specific integrated circuits (ASICs), or the like, may also be used without departing from the scope and spirit of the inventive concepts disclosed herein." *(paragraph 28)* Since they are by definition application specific, ASICs are inherently custom designed.
Claim 8	
8. The memory module of claim 1, wherein the logic element comprises a complex programmable-logic device.	"In addition, those of ordinary skill in the art will recognize that devices of a less general purpose nature, such as hardwired devices, field programmable logic devices (FPLDs), including field programmable gate arrays (FPGAs) and complex programmable logic devices (CPLDs), application specific integrated circuits (ASICs), or the like, may also be used without departing from the scope and spirit of the inventive concepts disclosed herein." *(paragraph 28)*
Claim 9	
9. The memory module of claim 1, wherein the first number of ranks is four, and the second number of ranks is two.	"CPLD 410 allows a system having a memory socket with only two chip select signals routed to interface with a four rank memory module where typically a two rank memory module couples with the memory socket." *(paragraph 41)*
Claim 10	
10. The memory module of claim 1, wherein the first number of ranks is two, and the second number of ranks is one.	Amidi teaches doubling the available number of chip select signals, from two to four. It would have been obvious to a person of ordinary skill in the art to apply the techniques of Amidi to double the number of chip select signals from one to two, for modules made for computer systems that provide only one chip select signal.

<table>
<tr><th>US PATENT 7,289,386</th><th>DISCLOSURE US PUBLICATION 2006/0117152 (AMIDI)</th></tr>
<tr><td>Claim 11</td><td></td></tr>
<tr><td>11. The memory module of claim 1, wherein the set of input control signals comprises two chip-select signals and an address signal and the set of output control signals comprises four chip-select signals.</td><td>"CPLD 410 utilized three variables to determine the active rank: the highest address number Add(n), a first chip select signal (CS0), and a second chip select signal (CS1). [¶] As illustrated in FIG. 5, rank 0 is active when the Add(n) is 0, CS1 is 1, and CS0 is 0. Rank 1 is active when the Add(n) is 0, CS1 is 0, and CS0 is 1. Rank 2 is active when the Add(n) is 1, CS1 is 1, and CS0 is 0. Rank 3 is active when the Add(n) is 1, CS1 is 0, and CS0 is 1." (paragraphs 43-44)

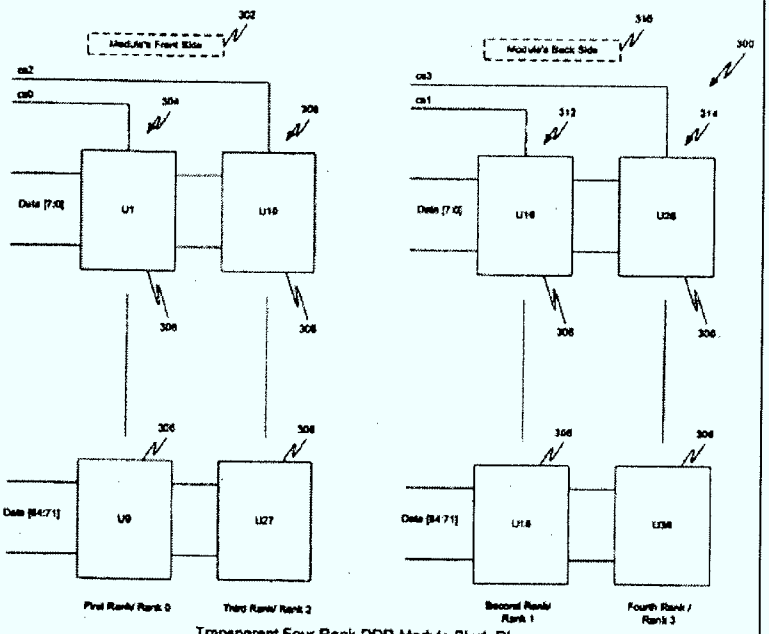
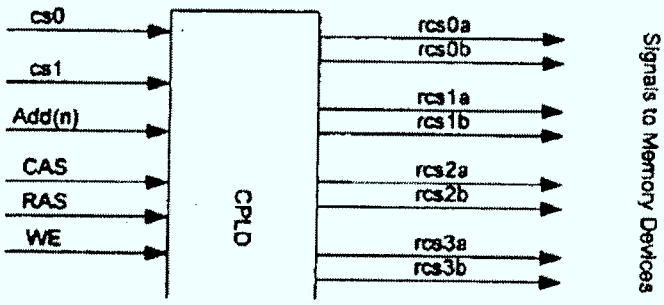
<table>
<tr><th>Add(n)</th><th>CS1</th><th>CS0</th><th>Active Bank</th></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>2</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>3</td></tr>
</table>
Truth Table
FIG.5

(Note that FIG. 5 erroneously refers to ranks 0-3 as banks.)

The input control signals include the two chip select signals CS0 and CS1 and address signal Add(n).

The output control signals include four chip select signals respectively connected to four ranks of memory devices, as illustrated in FIG. 3 and FIG. 6A, for example.

"FIG. 3 also illustrates a total of four chip select signals (cs0, cs1, cs2, and cs3). As illustrated in FIG. 2, the stacked memory device 200 has only two chip select signals: cs[1:0] 222. [The] present invention allows the four rank memory modules to communicate with a memory socket having only two chip select signals routed." (paragraph 36)</td></tr>
</table>

US PATENT 7,289,386	DISCLOSURE US PUBLICATION 2006/0117152 (AMIDI)
	 Transparent Four Rank DDR Module Block Diagram FIG. 3 The portion of FIG. 6A reproduced below illustrates four chip select signals rcs0 (a and b), rcs1 (a and b), rcs2 (a and b), and rcs3 (a and b), generated by the CPLD from two input chip select signals cs0 and cs1 and address signal Add(n). 
Claim 12	
12. The memory module of claim 1, wherein the printed circuit board is mountable in a module slot of the computer system, the printed circuit board having a plurality of edge connections electrically coupled to corresponding contacts of the module slot.	"The memory module 400 includes 92 contact pins 402 on the front side for connecting with a memory socket (not shown)." *(paragraph 37)* "Sockets on a main board accommodate those memory modules also known as SIMMs or DIMMs." *(page 12, paragraph 2)*