(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
12 February 2004 (12.02.2004)

PCT

(10) International Publication Number
WO 2004/013897 A2

(51) International Patent Classification[7]: H01L

(21) International Application Number:
PCT/US2003/024242

(22) International Filing Date: 1 August 2003 (01.08.2003)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
10/213,038 5 August 2002 (05.08.2002) US

(71) Applicant: **MICRON TECHNOLOGY, INC.** [US/US]; Micron Technology, Inc., 8000 South Federal Way, Boise, ID 83716-9632 (US).

(72) Inventor: **JEDDELOH, Joseph, M.**; 4302 Reiland Lane, Shoreview, MN 55126 (US).

(74) Agents: **BULCHIS, Edward, W.** et al.; Dorsey & Whitney LLP, 1420 Fifth Avenue, Suite 3400, Seattle, WA 98101 (US).

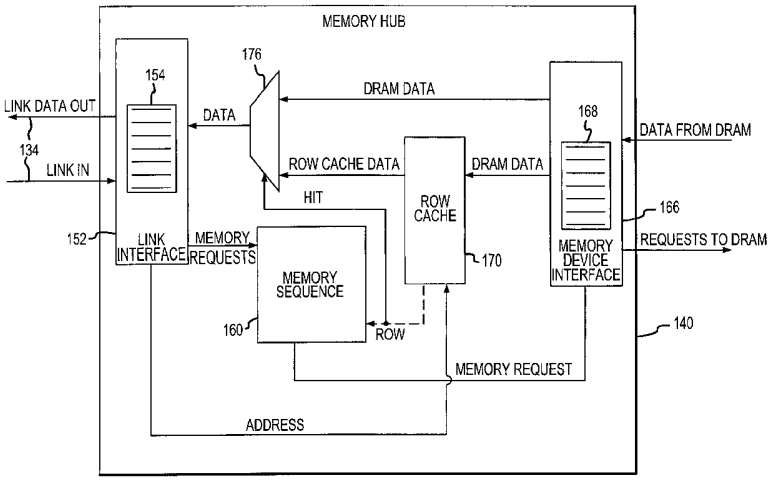
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: MEMORY HUB AND ACCESS METHOD HAVING INTERNAL ROW CACHING



(57) Abstract: A computer system includes a controller coupled to a plurality of memory modules each of which includes a memory hub and a plurality of memory devices. The memory hub includes a row cache memory that stores data as they are read from the memory devices. When the memory module is not being accessed by the controller, a sequencer in the memory module generates requests to read data from a row of memory cells. The data read responsive to the generated read requests are also stored in the row cache memory. As a result, read data in the row being accessed may be stored in the row cache memory even though the data was not previously read from the memory device responsive to a memory request from the controller.

WO 2004/013897 A2

MEMORY HUB AND ACCESS METHOD HAVING INTERNAL ROW CACHING

TECHNICAL FIELD

This invention relates to computer systems, and, more particularly, to a computer system having a memory hub coupling several memory devices to a processor or other memory access device.

BACKGROUND OF THE INVENTION

Computer systems use memory devices, such as dynamic random access memory ("DRAM") devices, to store instructions and data that are accessed by a processor. These memory devices are normally used as system memory in a computer system. In a typical computer system, the processor communicates with the system memory through a processor bus and a memory controller. The processor issues a memory request, which includes a memory command, such as a read command, and an address designating the location from which data or instructions are to be read. The memory controller uses the command and address to generate appropriate command signals as well as row and column addresses, which are applied to the system memory. In response to the commands and addresses, data are transferred between the system memory and the processor. The memory controller is often part of a system controller, which also includes bus bridge circuitry for coupling the processor bus to an expansion bus, such as a PCI bus.

Although the operating speed of memory devices has continuously increased, this increase in operating speed has not kept pace with increases in the operating speed of processors. Even slower has been the increase in operating speed of memory controllers coupling processors to memory devices. The relatively slow speed of memory controllers and memory devices limits the data bandwidth between the processor and the memory devices.

In addition to the limited bandwidth between processors and memory devices, the performance of computer systems is also limited by latency problems that increase the time required to read data from system memory devices. More specifically,

when a memory device read command is coupled to a system memory device, such as a synchronous DRAM ("SDRAM") device, the read data are output from the SDRAM device only after a delay of several clock periods. Therefore, although SDRAM devices can synchronously output burst data at a high data rate, the delay in initially providing the data can significantly slow the operating speed of a computer system using such SDRAM devices.

One approach to alleviating the memory latency problem is to use multiple memory devices coupled to the processor through a memory hub. In a memory hub architecture, a system controller or memory controller is coupled to several memory modules, each of which includes a memory hub coupled to several memory devices. The memory hub efficiently routes memory requests and responses between the controller and the memory devices. Computer systems employing this architecture can have a higher bandwidth because a processor can access one memory device while another memory device is responding to a prior memory access. For example, the processor can output write data to one of the memory devices in the system while another memory device in the system is preparing to provide read data to the processor. Although computer systems using memory hubs may provide superior performance, they nevertheless often fail to operate at optimum speed for several reasons. For example, even though memory hubs can provide computer systems with a greater memory bandwidth, they still suffer from latency problems of the type described above. More specifically, although the processor may communicate with one memory device while another memory device is preparing to transfer data, it is sometimes necessary to receive data from one memory device before the data from another memory device can be used. In the event data must be received from one memory device before data received from another memory device can be used, the latency problem continues to slow the operating speed of such computer systems.

One technique that has been used to reduce latency in memory devices is a cache memory, which stores data recently accessed from system memory. The cache memory is generally in the form of a static random access memory ("SRAM"), which has a substantially shorter access time compared to dynamic random access memory

("DRAM") typically used as system memory. Furthermore, the SRAM cache memory is generally coupled directly to the processor through a processor bus rather than through a system controller or the like as is typical with DRAM system memory. As a result of the faster speed of cache memory and the closer proximity of cache memory to the processor, the use of cache memory can greatly reduce the latency of memory read operations.

Although conventional cache memory has reduced memory access latencies in conventional computer systems, cache memory has not been used in a manner that provides optimum performance in computer systems using memory hubs. In particular, the limited storage capacity of typical cache memories compared to the vastly larger capacity of typical memory hub system memories makes cache memory of lesser value since a cache hit is less likely to occur. This problem is exacerbated by the difficulty in transferring data to cache memory that is likely to be the subject of subsequent memory requests. More specifically, it is difficult to couple the data that will subsequently be needed from all of the memory modules through the memory controller to the processor and then from the processor to the cache memory. Also, it can be difficult to maintain cache coherency in a computer system using memory hubs and a cache memory coupled to the processor through the processor bus, and it can require significant hardware resources to maintain cache coherency. Furthermore, the time required to maintain cache coherency can slow memory performance to the extent that much of the performance advantages of using cache memory can be lost.

There is therefore a need for a computer architecture that provides the advantages of a memory hub architecture and also minimize this latency problems common in such systems, thereby providing a memory devices with high bandwidth and low latency.

SUMMARY OF THE INVENTION

A plurality of memory modules are coupled to a controller in a computer system. Each of the memory modules includes a plurality of memory devices and a

memory hub. The memory hub comprises a link interface coupled to the controller and a memory device interface coupled to the memory devices. The link interface receives memory requests from the controller for access to a row of memory cells in at least one of the memory devices. The link interface transfers the memory requests to the memory device interface, which then couples the memory requests to the memory devices for access to a row of memory cells in at least one of the memory devices. The memory device interface then receives the read data from the memory devices responsive to at least some of the memory requests. Each of the memory hubs also includes a row cache memory coupled to the memory device interface for receiving and storing read data responsive to at least one of the memory requests. A sequencer that is also included in the memory hub is coupled to the link interface, the memory device interface and the row cache memory. The sequencer generates and couples to the memory device interface memory requests to read data from memory cells in a row of memory cells being accessed responsive to a memory request transferred from the link interface to the memory device interface. The read data from the memory cells in the row of memory cells being accessed responsive to the generated memory requests are also stored in the row cache memory. The sequencer preferably generates the memory requests when memory requests are not being received from the controller.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram of a computer system according to one example of the invention in which a memory hub is included in each of a plurality of memory modules.

Figure 2 is a block diagram of a memory hub used in the computer system of Figure 1.

DETAILED DESCRIPTION OF THE INVENTION

A computer system 100 according to one example of the invention is shown in Figure 1. The computer system 100 includes a processor 104 for performing various computing functions, such as executing specific software to perform specific

calculations or tasks. The processor 104 includes a processor bus 106 that normally includes an address bus, a control bus, and a data bus. The processor bus 106 is typically coupled to cache memory 108, which, as previously mentioned, is usually static random access memory ("SRAM"). Finally, the processor bus 106 is coupled to a system controller 110, which is also sometimes referred to as a "North Bridge" or "memory controller."

The system controller 110 serves as a communications path to the processor 104 for a variety of other components. More specifically, the system controller 110 includes a graphics port that is typically coupled to a graphics controller 112, which is, in turn, coupled to a video terminal 114. The system controller 110 is also coupled to one or more input devices 118, such as a keyboard or a mouse, to allow an operator to interface with the computer system 100. Typically, the computer system 100 also includes one or more output devices 120, such as a printer, coupled to the processor 104 through the system controller 110. One or more data storage devices 124 are also typically coupled to the processor 104 through the system controller 110 to allow the processor 104 to store data or retrieve data from internal or external storage media (not shown). Examples of typical storage devices 124 include hard and floppy disks, tape cassettes, and compact disk read-only memories (CD-ROMs).

The system controller 110 is coupled to several memory modules 130a,b...n, which serve as system memory for the computer system 100. The memory modules 130 are preferably coupled to the system controller 110 through a high-speed link 134, which may be an optical or electrical communication path or some other type of communications path. In the event the high-speed link 134 is implemented as an optical communication path, the optical communication path may be in the form of one or more optical fibers, for example. In such case, the system controller 110 and the memory modules will include an optical input/output port or separate input and output ports coupled to the optical communication path. The memory modules 130 are shown coupled to the system controller 110 in a multi-drop arrangement in which the single high-speed link 134 is coupled to all of the memory modules 130. However, it will be understood that other topologies may also be used, such as a point-to-point coupling

arrangement in which a separate high-speed link (not shown) is used to couple each of the memory modules 130 to the system controller 110. A switching topology may also be used in which the system controller 110 is selectively coupled to each of the memory modules 130 through a switch (not shown). Other topologies that may be used will be apparent to one skilled in the art.

Each of the memory modules 130 includes a memory hub 140 for controlling access to 6 memory devices 148, which, in the example illustrated in Figure 2, are synchronous dynamic random access memory ("SDRAM") devices. However, a fewer or greater number of memory devices 148 may be used, and memory devices other than SDRAM devices may, of course, also be used. The memory hub 140 is coupled to each of the system memory devices 148 through a bus system 150, which normally includes a control bus, an address bus and a data bus.

One example of the memory hub 140 of Figure 1 is shown in Figure 2. The memory hub 140 includes a link interface 152 that is coupled to the high-speed link 134. The nature of the link interface 152 will depend upon the characteristics of the high-speed link 134. For example, in the event the high-speed link 134 is implemented using an optical communications path, the link interface 152 will include an optical input/output port and will convert optical signals coupled through the optical communications path into electrical signals. In any case, the link interface 152 preferably includes a buffer, such as a first-in, first-out buffer 154, for receiving and storing memory requests as they are received through the high-speed link 134. The memory requests are stored in the buffer 154 until they can be processed by the memory hub 140.

When the memory hub 140 is able to process a memory request, one of the memory requests stored in the buffer 154 is transferred to a memory sequencer 160. The memory sequencer 160 converts the memory requests from the format output from the system controller 110 into a memory request having a format that can be used by the memory devices 148. These re-formatted request signals will normally include memory command signals, which are derived from memory commands contained in the memory request received by the memory hub 140, and row and column address signals, which

are derived from an address contained in the memory request received by the memory hub 140. In the event the memory request is a write memory request, the re-formatted request signals will normally include write data signals which are derived from write data contained in the memory request received by the memory hub 140. For example, where the memory devices 148 are conventional DRAM devices, the memory sequencer 160 will output row address signals, a row address strobe ("RAS") signal, an active high write/active low read signal ("W/R*"), column address signals and a column address strobe ("CAS") signal. The re-formatted memory requests are preferably output from the sequencer 160 in the order they will be used by the memory devices 148.

The memory sequencer 160 applies the re-formatted memory requests to a memory device interface 166. The nature of the memory device interface 166 will again depend upon the characteristics of the memory devices 148. In any case, the memory device interface 166 preferably includes a buffer, such as a FIFO buffer 168, for receiving and storing one or more memory requests as they are received from the link interface 152. The memory requests are stored in the buffer 168 until they can be processed by the memory devices 148. However, in the event the memory device interface 166 stores several memory requests, the memory device interface 166 may re-order the memory requests so that they are applied to the memory devices in some other order. For example, the memory requests may be stored in the interface 166 in a manner that causes one type of request, *e.g.*, read requests to be processed before other types of requests, *e.g.*, write requests.

The memory requests are described above as being received by the memory hub 140 in a format that is different from the format that the memory requests are applied to the memory devices 148. However, the system controller 110 may instead re-format memory requests from the processor 104 (Figure 1) to a format that can be used by the memory devices 148. In such case, it is not necessary for the sequencer 160 to re-format the memory request. Instead, the sequencer 160 simply schedules the re-formatted memory request signals in the order needed for use by the memory devices 148. The memory request signals for one or more memory requests are

then transferred to the memory device interface 166 so they can subsequently be applied to the memory devices 148.

As previously explained, one of the disadvantages of using memory hubs is the increased latency they can sometimes create. As also previously explained, a cache memory in the processor 104 or coupled to the processor bus 106 (Figure 1), which is the traditional approach to reducing memory read latency, is not well suited to a memory system using memory hubs. The memory hub 140 example shown in Figure 2 provides relatively low memory read latency by including a row cache memory 170 in each of the memory hubs 140. The row cache memory 170 may be similar in design to conventional cache systems including a data memory (not shown), a tag memory (not shown), and conventional address comparison logic (not shown). The row cache memory 170 stores data contained in one or more previously addressed rows of memory cells in one or more of the memory devices 148 in the module 140. The row cache memory 170 receives addresses forming part of a memory request from the link interface 152, which are compared to addresses of cached data. In the event of an address match, which indicates that the data being fetched by the memory request is stored in the row cache memory 170, the memory 170 outputs the requested data and a "ROW HIT" signal indicative of a cache hit. The ROW HIT signal is applied to a multiplexer 176 to cause the data from the cache memory 170 to be coupled to the link interface 152. In the event of a row cache miss, the multiplexer 176 couples data from the memory device interface 166 to the link interface 152. The ROW HIT signal is also applied to the memory sequencer 160 so that the sequencer will not couple the memory request to the memory device interface 166 in the event of a row hit since the data called for by the memory request has already been supplied by the row cache memory 170.

Although the row cache memory 170 may store data only from columns in a row that have been previously accessed, the memory 170 preferably pre-fetches data from many or all of the columns in the cached row when the memory hub 140 is not busy responding to memory requests from the system controller 110. More specifically, the memory sequencer 160 contains conventional circuitry to keep track of

which columns of a row being accessed have had the data stored therein transferred to the row cache memory 170. When the sequencer 160 is not busy servicing memory requests from the link interface 152, the sequencer 160 generates memory requests, which are applied to the memory device interface 166, to cause data stored in the remaining columns of an addressed row to be transferred to the row cache memory 170. As a result, since memory accesses are typically to a series of memory locations in the same row, the row cache memory 170 is likely to be storing data that will be fetched in subsequent memory requests.

The memory hub 140 can process a subsequent memory request directed to a new row of memory cells in one of the memory devices 148 using a variety of procedures. For example, if the row cache memory 170 is capable of storing data from more than one row, the sequencer 160 can simply cause the data stored in the subsequently accessed row to be transferred to the row cache memory 170. If the row cache memory 170 is capable of storing data from only a single row of memory cells, or the cache memory 170 has otherwise reached its storage capacity, the data stored in the newly accessed row of memory cells can simply overwrite the previously stored data.

Although not shown in Figure 2 or discussed above, the memory hub 140 preferably includes circuitry for maintaining cache consistency using conventional memory cache techniques. For example, the hub 140 may employ a "write through" mode of operation or a "write back" mode of operation in the event of a memory request for a write to a location followed by a memory request for a read from that same location.

From the foregoing it will be appreciated that, although specific embodiments of the invention have been described herein for purposes of illustration, various modifications may be made without deviating from the spirit and scope of the invention. Accordingly, the invention is not limited except as by the appended claims.

CLAIMS

1. A memory module, comprising:

a plurality of memory devices; and

a memory hub, comprising:

a link interface receiving memory requests for access to a row of memory cells in at least one of the memory devices;

a memory device interface coupled to the memory devices, the memory device interface being operable to couple memory requests to the memory devices for access to a row of memory cells in at least one of the memory devices and to receive read data responsive to at least some of the memory requests, at least some of the memory requests coupled to the memory devices being responsive to memory requests transferred from the link interface to the memory device interface;

a row cache memory coupled to the memory device interface for receiving and storing read data from a row of memory cells being accessed responsive to at least one of the memory requests being coupled from the memory device interface to the at least one memory device; and

a sequencer coupled to the link interface and the memory device interface and the row cache memory, the sequencer being operable to generate and couple to the memory device interface memory requests to read data from memory cells in row of memory cells being accessed, the read data read from the memory cells in the row of memory cells being accessed being stored in the row cache memory.

2. The memory module of claim 1 wherein the memory device interface further comprises a first-in, first-out buffer that is operable to receive and to store memory requests received from the link interface and from the sequencer and to transfer the stored memory requests to the at least one memory device in the order in which they were received.

3. The memory module of claim 1 wherein the link interface comprises a first-in, first-out buffer that is operable to receive and store memory requests and to transfer

the stored memory requests to the memory device interface in the order in which they were received.

4. The memory module of claim 3 wherein the memory device interface further comprises a first-in, first-out buffer that is operable to receive and store memory requests received from the link interface and from the sequencer and to transfer the stored memory requests to the at least one memory device in the order in which they were received.

5. The memory module of claim 1 wherein the link interface comprises an optical input/output port.

6. The memory module of claim 1 wherein the memory devices comprises dynamic random access memory devices.

7. The memory module of claim 1 wherein the sequencer is operable to output an address contained in each read memory request received from the link interface, and wherein the row cache memory is operable to receive the addresses from the sequencer to determine if data called for by the memory request is stored in the row cache memory, the row cache memory outputting the read data and generating a hit signal if the data called for by the memory request is stored in the row cache memory and generating a row miss signal if the data called for by the memory request is not stored in the row cache memory.

8. The memory module of claim 7 further comprising a multiplexer having data inputs coupled to the row cache memory and to the memory device interface, a data output coupled to the link interface and a control input coupled to receive the row cache hit and row cache miss signals from the row cache memory, the multiplexer coupling read data from the memory device interface responsive to the row cache miss signal and coupling read data from the row cache memory responsive to the row cache hit signal.

9. The memory module of claim 1 wherein the sequencer is operable to generate and couple to the memory device interface memory requests to read data from memory cells in row of memory cells being accessed only when memory requests are not being transferred from the link interface to the memory device interface.

10. A memory module, comprising:

a plurality of memory devices; and

a memory hub, comprising:

a link interface receiving memory requests for access to a row of memory cells in at least one of the memory devices;

a memory device interface coupled to the memory devices, the memory device interface being operable to couple memory requests to the memory devices for access to a row of memory cells in at least one of the memory devices and to receive read data responsive to at least some of the memory requests, at least some of the memory requests coupled to the memory devices being responsive to memory requests transferred from the link interface to the memory device interface;

a sequencer coupled to the link interface and the memory device interface and the row cache memory, the sequencer being operable to output an address contained in each read memory request received from the link interface;

a row cache memory coupled to the memory device interface for receiving and storing read data from a row of memory cells being accessed responsive to one of the memory requests being coupled from the memory device interface to the at least one memory device, the row cache memory further being operable to receive the addresses from the sequencer to determine if data called for by the memory request is stored in the row cache memory, the row cache memory outputting the read data and generating a hit signal if the data called for by the memory request is stored in the row cache memory and generating a row miss signal if the data called for by the memory request is not stored in the row cache memory; and

a multiplexer having data inputs coupled to the row cache memory and to the memory device interface, a data output coupled to the link interface and a

control input coupled to receive the row cache hit and row cache miss signals from the row cache memory, the multiplexer coupling read data from the memory device interface responsive to the row cache miss signal and coupling read data from the row cache memory responsive to the row cache hit signal.

11. The memory module of claim 10 wherein the memory device interface further comprises a first-in, first-out buffer that is operable to receive and store memory requests received from the link interface and from the sequencer and to transfer the stored memory requests to the at least one memory device in the order in which they were received.

12. The memory module of claim 10 wherein the link interface comprises a first-in, first-out buffer that is operable to receive and store memory requests and to transfer the stored memory requests to the memory device interface in the order in which they were received.

13. The memory module of claim 12 wherein the memory device interface further comprises a first-in, first-out buffer that is operable to receive and store memory requests received from the link interface and from the sequencer and to transfer the stored memory requests to the at least one memory device in the order in which they were received.

14. The memory module of claim 10 wherein the link interface comprises an optical input/output port.

15. The memory module of claim 10 wherein the memory devices comprises dynamic random access memory devices.

16. A memory hub, comprising:

a link interface receiving memory requests, at least some of the memory requests including a row address;

a memory device interface operable to output memory requests and to receive read data responsive to at least some of the memory requests, at least some of the memory requests output by the memory device interface being responsive to memory requests transferred from the link interface to the memory device interface;

a row cache memory coupled to the memory device interface for receiving and storing read data received from the memory device interface responsive to at least one of the memory requests being output from the memory device interface; and

a sequencer coupled to the link interface and the memory device interface and the row cache memory, the sequencer being operable to generate and couple to the memory device interface memory requests to read data associated with read data received by the memory device interface responsive to memory requests transferred from the link interface to the memory device interface, the read data read received responsive to memory requests from the sequencer being stored in the row cache memory.

17. The memory hub of claim 16 wherein the memory device interface further comprises a first-in, first-out buffer that is operable to receive and store memory requests received from the link interface and from the sequencer and to output the stored memory requests in the order in which they were received.

18. The memory hub of claim 16 wherein the link interface comprises a first-in, first-out buffer that is operable to receive and store memory requests and to transfer the stored memory requests to the memory device interface in the order in which they were received.

19. The memory hub of claim 18 wherein the memory device interface further comprises a first-in, first-out buffer that is operable to receive and store memory requests received from the link interface and from the sequencer and to output the stored memory requests in the order in which they were received.

20. The memory hub of claim 16 wherein the link interface comprises an optical input/output port.

21. The memory hub of claim 16 wherein the sequencer is operable to output an address contained in each read memory request received from the link interface, and wherein the row cache memory is operable to receive the addresses from the sequencer to determine if data called for by the memory request is stored in the row cache memory, the row cache memory outputting the read data and generating a hit signal if the data called for by the memory request is stored in the row cache memory and generating a row miss signal if the data called for by the memory request is not stored in the row cache memory.

22. The memory hub of claim 21 further comprising a multiplexer having data inputs coupled to the row cache memory and to the memory device interface, a data output coupled to the link interface and a control input coupled to receive the row cache hit and row cache miss signals from the row cache memory, the multiplexer coupling read data from the memory device interface responsive to the row cache miss signal and coupling read data from the row cache memory responsive to the row cache hit signal.

23. The memory hub of claim 16 wherein the sequencer is operable to generate and couple to the memory device interface memory requests to read data only when memory requests are not being transferred from the link interface to the memory device interface.

24. A computer system, comprising:

a central processing unit ("CPU");

a system controller coupled to the CPU, the system controller having an input port and an output port;

an input device coupled to the CPU through the system controller;

an output device coupled to the CPU through the system controller;

a storage device coupled to the CPU through the system controller;

a plurality of memory modules, each of the memory modules comprising:

a plurality of memory devices; and

a memory hub, comprising:

a link interface having an input port and an output port, the link interface receiving memory requests through the input port for access to a row of memory cells in at least one of the memory devices and outputting data through the output port responsive to the memory requests;

a memory device interface coupled to the memory devices, the memory device interface being operable to couple memory requests to the memory devices for access to a row of memory cells in at least one of the memory devices and to receive read data responsive to at least some of the memory requests, at least some of the memory requests coupled to the memory devices being responsive to memory requests transferred from the link interface to the memory device interface;

a row cache memory coupled to the memory device interface for receiving and storing read data from a row of memory cells being accessed responsive to at least one of the memory requests being coupled from the memory device interface to the at least one memory device; and

a sequencer coupled to the link interface and the memory device interface and the row cache memory, the sequencer being operable to generate and couple to the memory device interface memory requests to read data from memory cells in row of memory cells being accessed, the read data read from the memory cells in the row of memory cells being accessed being stored in the row cache memory; and

a communications link coupling the output port of the system controller to the input port of the memory hub in each of the memory modules, and coupling the input port of the system controller to the output port of the memory hub in each of the memory modules.

25. The computer system of claim 24 wherein the memory device interface further comprises a first-in, first-out buffer that is operable to receive and to store memory

requests received from the link interface and from the sequencer and to transfer the stored memory requests to the at least one memory device in the order in which they were received.

26. The computer system of claim 24 wherein the link interface comprises a first-in, first-out buffer that is operable to receive and store memory requests and to transfer the stored memory requests to the memory device interface in the order in which they were received.

27. The computer system of claim 26 wherein the memory device interface further comprises a first-in, first-out buffer that is operable to receive and store memory requests received from the link interface and from the sequencer and to transfer the stored memory requests to the at least one memory device in the order in which they were received.

28. The computer system of claim 24 wherein the memory devices comprises dynamic random access memory devices.

29. The computer system of claim 24 wherein the sequencer is operable to output an address contained in each read memory request received from the link interface, and wherein the row cache memory is operable to receive the addresses from the sequencer to determine if data called for by the memory request is stored in the row cache memory, the row cache memory outputting the read data and generating a hit signal if the data called for by the memory request is stored in the row cache memory and generating a row miss signal if the data called for by the memory request is not stored in the row cache memory.

30. The computer system of claim 29 further comprising a multiplexer having data inputs coupled to the row cache memory and to the memory device interface, a data output coupled to the link interface and a control input coupled to receive the row cache hit and row cache miss signals from the row cache memory, the multiplexer coupling read data from the memory device interface responsive to the row cache miss signal and coupling read data from the row cache memory responsive to the row cache hit signal.

31. The computer system of claim 24 wherein the sequencer is operable to generate and couple to the memory device interface memory requests to read data from memory cells in row of memory cells being accessed only when memory requests are not being transferred from the link interface to the memory device interface.

32. The computer system of claim 24 wherein the input and output ports of the system controller comprises a combined input/output port coupled to the communications link, and wherein the input and output ports of each of the memory hubs comprises a combined input/output port coupled to the communications link.

33. The computer system of claim 32 wherein the communications link comprises an optical communications link, wherein the input and output ports of the system controller comprises an optical input/output port coupled to the optical communications link and wherein the input and output ports of each of the memory hubs comprises a respective optical input/output port coupled to the optical communications link.

34. A computer system, comprising:

a central processing unit ("CPU");

a system controller coupled to the CPU, the system controller having an input port and an output port;

an input device coupled to the CPU through the system controller;

an output device coupled to the CPU through the system controller;

a storage device coupled to the CPU through the system controller;

a plurality of memory modules, each of the memory modules comprising:

a plurality of memory devices; and

a memory hub, comprising:

a link interface receiving memory requests for access to a row of memory cells in at least one of the memory devices;

a memory device interface coupled to the memory devices, the memory device interface being operable to couple memory requests to the

memory devices for access to a row of memory cells in at least one of the memory devices and to receive read data responsive to at least some of the memory requests, at least some of the memory requests coupled to the memory devices being responsive to memory requests transferred from the link interface to the memory device interface;

a sequencer coupled to the link interface and the memory device interface and the row cache memory, the sequencer being operable to output an address contained in each read memory request received from the link interface;

a row cache memory coupled to the memory device interface for receiving and storing read data from a row of memory cells being accessed responsive to one of the memory requests being coupled from the memory device interface to the at least one memory device, the row cache memory further being operable to receive the addresses from the sequencer to determine if data called for by the memory request is stored in the row cache memory, the row cache memory outputting the read data and generating a hit signal if the data called for by the memory request is stored in the row cache memory and generating a row miss signal if the data called for by the memory request is not stored in the row cache memory; and

a multiplexer having data inputs coupled to the row cache memory and to the memory device interface, a data output coupled to the link interface and a control input coupled to receive the row cache hit and row cache miss signals from the row cache memory, the multiplexer coupling read data from the memory device interface responsive to the row cache miss signal and coupling read data from the row cache memory responsive to the row cache hit signal; and

a communications link coupling the output port of the system controller to the input port of the memory hub in each of the memory modules, and coupling the input port of the system controller to the output port of the memory hub in each of the memory modules.

35. The computer system of claim 34 wherein the memory device interface further comprises a first-in, first-out buffer that is operable to receive and store memory requests received from the link interface and from the sequencer and to transfer the stored memory requests to the at least one memory device in the order in which they were received.

36. The computer system of claim 34 wherein the link interface comprises a first-in, first-out buffer that is operable to receive and store memory requests and to transfer the stored memory requests to the memory device interface in the order in which they were received.

37. The computer system of claim 36 wherein the memory device interface further comprises a first-in, first-out buffer that is operable to receive and store memory requests received from the link interface and from the sequencer and to transfer the stored memory requests to the at least one memory device in the order in which they were received.

38. The computer system of claim 34 wherein the link interface comprises an optical input/output port.

39. The computer system of claim 34 wherein the memory devices comprises dynamic random access memory devices.

40. The computer system of claim 34 wherein the input and output ports of the system controller comprises a combined input/output port coupled to the communications link, and wherein the input and output ports of each of the memory hubs comprises a combined input/output port coupled to the communications link.

41. The computer system of claim 40 wherein the communications link comprises an optical communications link, wherein the input and output ports of the system controller comprises an optical input/output port coupled to the optical communications link

and wherein the input and output ports of each of the memory hubs comprises a respective optical input/output port coupled to the optical communications link.

42. A method of reading data in each of a plurality of memory modules using a controller coupled to the memory modules, the method comprising:

receiving memory requests from the controller by a first one of the memory modules, at least one of the memory requests being a request to access memory cells in a row of memory cells of at least one memory device in a plurality of memory devices included in the first memory module;

coupling the received memory requests to the at least one memory device in the first memory module;

generating requests to read data from memory cells in the row of memory cells being accessed responsive to the request to access memory cells in a row of memory cells of the at least one memory device, the requests being generated when memory requests from the controller are not being coupled to the at least one memory device;

coupling the generated memory requests to the at least one memory device; and

storing in cache memory in the first memory module read data responsive to received memory requests and the generated memory requests.

43. The method of claim 42, further comprising, in response to a memory request from the controller to read data from the first memory module:

determining if the read data are stored in the cache memory;

if the read data are stored in the cache memory, transferring the requested read data from the cache memory to the controller; and

if the read data are not stored in the cache memory, transferring the requested read data from the at least one memory device to the controller.

44. The method of claim 42 wherein the act of receiving memory requests from the controller by a first one of the memory modules comprises receiving optical signals corresponding to the received memory requests.

45. The method of claim 42 wherein the memory devices included in the first memory module comprises dynamic random access memory devices.

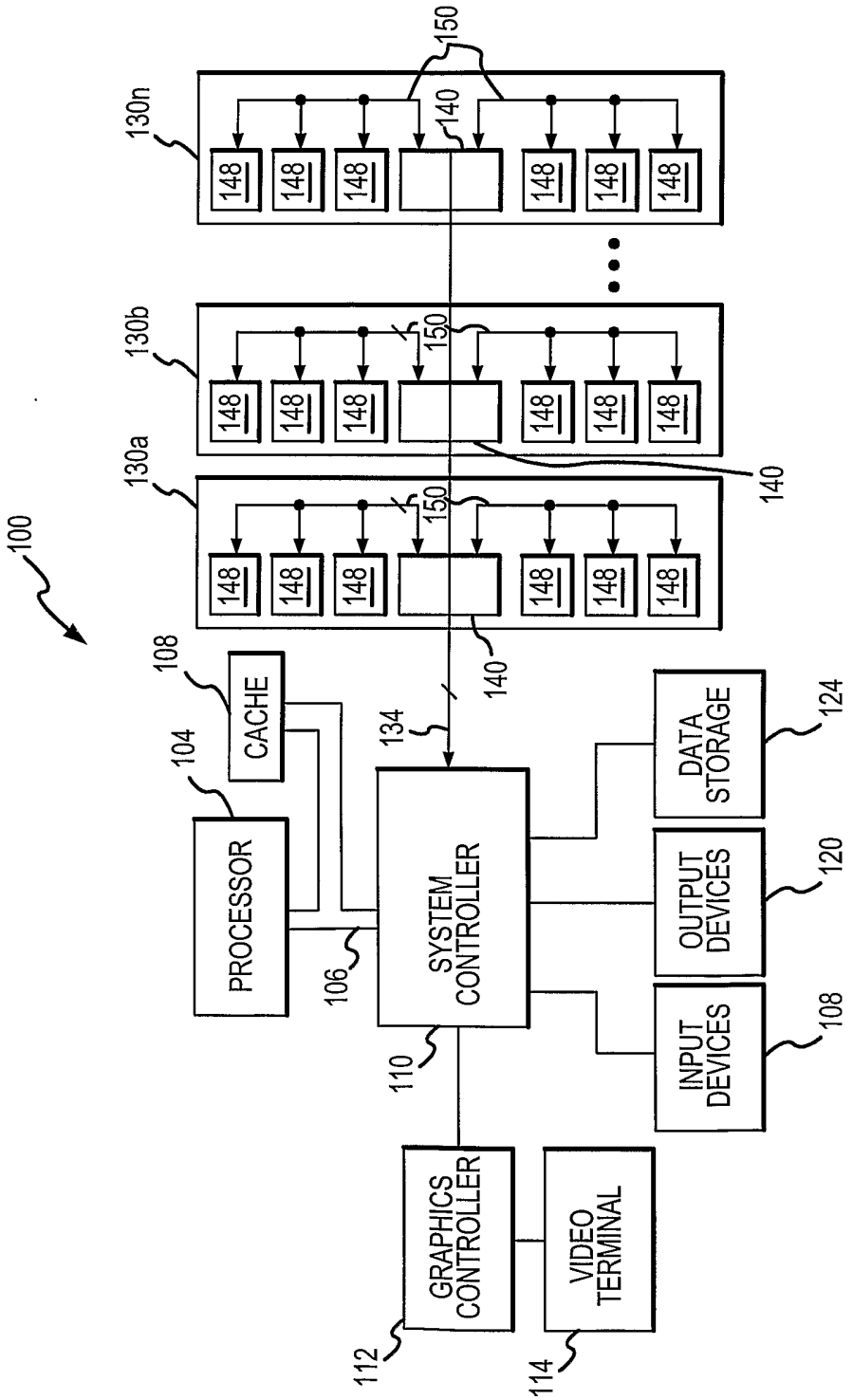

100
130n
130b
130a
148
150
140
108
CACHE
104
PROCESSOR
106
134
SYSTEM CONTROLLER
110
DATA STORAGE
124
OUTPUT DEVICES
120
INPUT DEVICES
108
GRAPHICS CONTROLLER
112
VIDEO TERMINAL
114

FIG.1

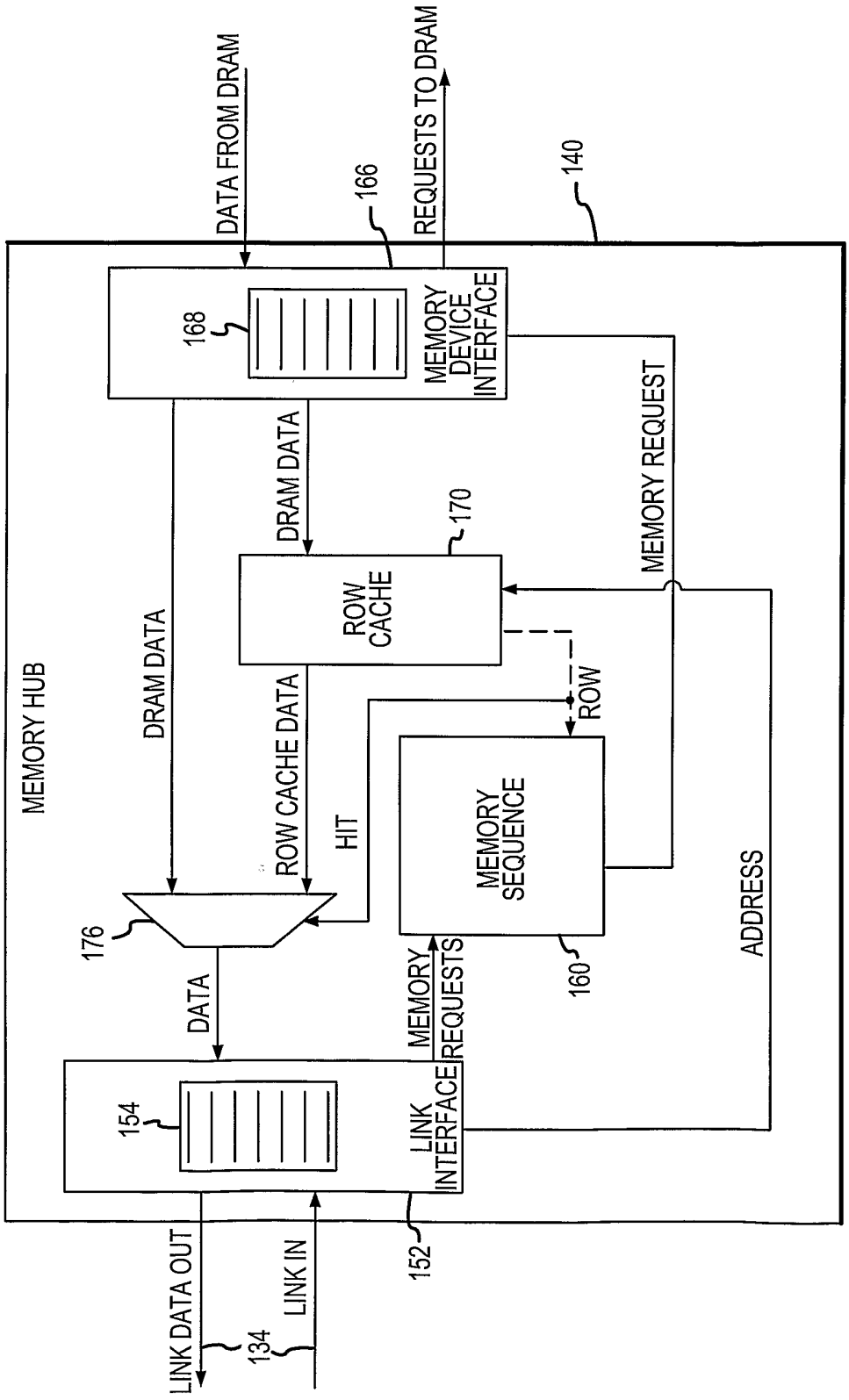

DATA FROM DRAM
166
REQUESTS TO DRAM
140
168
MEMORY DEVICE INTERFACE
DRAM DATA
170
ROW CACHE
MEMORY REQUEST
DRAM DATA
ROW CACHE DATA
HIT
ROW
MEMORY HUB
MEMORY SEQUENCE
176
DATA
MEMORY REQUESTS
ADDRESS
160
154
LINK INTERFACE
LINK DATA OUT
LINK IN
134
152

FIG.2