

MEMORY DEVICE WITH NETWORK ON CHIP METHODS, APPARATUS, AND SYSTEMS

Related Application

5 This patent application claims priority benefit from U.S. Application No. 12/033,684 filed 19 February 2008 which is incorporated herein by reference.

Background

10 Many electronic devices, such as personal computers, workstations, computer servers, mainframes and other computer related equipment, including printers, scanners and hard disk drives, make use of memory devices that provide a large data storage capability, while attempting to incur low power consumption. One type of memory device that is well-suited for use in the foregoing devices is the dynamic random access memory (DRAM).

15 The demand for larger capacity of memory devices continue to rise and at the same time chip size limitations bound the capacity of these memory devices. The surface area occupied by the components of individual memory cells has been steadily decreased so that the packing density of the memory cells on a semiconductor substrate can be increased along with the gate delays being
20 decreased. Shrinking of the device surface area can result in reducing manufacturing yield, as well as increasing the complexity of interconnects used to connect the numerous banks within the DRAM devices with other devices. Additionally, during miniaturization, interconnect delays do not scale as well as gate delays.

25

Brief Description of the Drawings

 Various embodiments of the present invention are described in detail below with reference to the following drawings.

 FIG. 1 illustrates a block diagram of a memory system, according to
30 various embodiments of the invention.

FIG. 2 illustrates a top view of a memory device, according to some embodiments of the invention.

FIG. 3 illustrates a perspective view of the memory system shown in FIG. 1, according to some embodiments of the invention.

5 FIG. 4 illustrates a cross-sectional view of system similar to that shown in FIG. 1, according to some embodiments of the invention.

FIG. 5A illustrates a data structure of a command packet used in memory system shown in FIG. 1, according to various embodiments of the invention.

10 FIG. 5B illustrates a data structure of a data packet used in memory system shown in FIG. 1, according to various embodiments of the invention.

FIG. 6 illustrates a flow diagram of a method of operation of system shown in FIG. 3, according to some embodiments of the invention.

FIG. 7 illustrates a schematic diagram showing a memory system coupled to a processor module, according to some embodiments of the invention.

15

Detailed Description

Surface area reduction and a consequent increase in the packing density of memories can be achieved by decreasing the horizontal feature size of memory arrays. This can be achieved in various embodiments by forming memory arrays that are significantly three-dimensional, so that the memory arrays extend vertically into and above the substrate, in addition to generally extending across the surface of the substrate.

FIG. 1 illustrates a block diagram of memory system 100 according to various embodiments of the invention. The memory system 100 includes an interface device 150 coupled to memory devices 110, 120 and processor module 160. In some embodiments, interface device 150 includes routing elements 153, 154, 155, 156, 157, 158 and 159 that are interconnected with each other. In some embodiments, interface device 150 also includes a DRAM controller 151 and a Flash controller 152. In some embodiments, DRAM controller 151 includes a non page mode controller. In some embodiments, DRAM controller 151 includes an out of order command queue provided with memory command bus optimization. In some embodiments, the DRAM controller 151 can be programmable and contain Built-in Self Test (BIST) to aid memory testing.

Interface device 150 is coupled to memory devices 110, 120 using interconnections provided through memory Z-vias 135 and 145, respectively. Memory Z-vias are openings provided within the memory device (110, 120) that allows for vertical interconnects to partially or completely pass through the memory device (110, 120) allowing connectivity between memory arrays situated above and below a particular memory array located within the memory device. In some embodiments, interconnects within memory Z-vias 135, 145 include 128-bit data busses. In some embodiments, processing module 160 includes processors 162, 164, 166 and 168 that are coupled to routing elements 153, 155, 156 and 158 using processor Z-vias 172, 174, 176 and 178, respectively. In some embodiments, processing module 160 includes a general-purpose processor or an application specific integrated circuit (ASIC). In some embodiments, processing module 160 can comprise a single-core processor and/or a multiple-core processor.

In some embodiments, memory device 110 includes memory arrays 110-1, 110-2, 110-3 and 110-4, each memory array having memory cells logically arranged in rows and columns. Similarly, in some embodiments, memory device 120 includes memory arrays 120-1, 120-2, 120-3 and 120-4, each memory array having memory cells logically arranged in rows and columns. The interface device 150 provides memory commands to selected memory addresses within the memory devices 110 and 120. In some embodiments, memory devices 110, 120 include Dynamic Random Access Memory (DRAM) devices. In some embodiments, the interface device 150 includes circuitry configured to perform DRAM sequencing.

In some embodiments, interface device 150 is configured to implement a refresh scheme to control error rates based on the type of DRAM used in memory device 110 and 120. An activation pulse for reading and rewriting and a precharge pulse indicating the original state are supplied from a command decoder during an auto refresh sequence acting on a memory address, and a clock signal is applied to the DRAM. In order to avoid a loss of data, the memory cells of DRAMs (dynamic random access memories) have to be regularly read and then have their contents rewritten, which is referred to as a "refresh" of the memory cells. In some embodiments, the interface device 150 is

configured to operate a refresh scheme to control error rates based on the particular signal characteristics of each of memory devices 110 and 120.

In some embodiments, interface device 150 is configured to operate a bad cell recovery scheme on the plurality of memory arrays included within memory devices 110 and 120. In some embodiments, the interface device 150 is programmable and configured to operate based on the type of the memory die disposed adjacent to it. In some embodiments, the interface device 150 is configured to operate a bad cell recovery scheme on the plurality of memory arrays 110-1, 110-2, 110-3 and 110-4. In some embodiments, the interface device 150 includes a pattern generator configured to generate test pattern signals that are used for testing and diagnostic analysis of memory devices 110 and 120.

In some embodiments, interface device 150 acts as an interconnection device and an I/O driver. In some embodiments, the interface device 150 includes traditional functional blocks present within DRAM dice, such as I/O pads, delayed-locked loop (DLL) circuitry, and First In, First Out (FIFO) circuitry that provides read and write pointers, storage, and control logic. Transferring these functional blocks, from the DRAM dice into the interface device 150 allows for the increase in storage area of the DRAM.

In some embodiments, the interface device 150 is coupled to 32 different memory devices (each having several memory arrays; other numbers of devices are possible) with independent interconnects such as the through wafer interconnect (TWI) shown in FIG. 3. In some embodiments, the interface device 150 is coupled to a set of TWI configured to provide different types of interconnection based on the end user application. In some embodiments, the set of TWI 321 passes through the memory Z-vias 322, 323 provided within memory arrays 110-1 and 110-2. In some embodiments, the TWI can be configured to provide connectivity between different types of DRAM and the interface device 150.

In some embodiments, the interface device 150 is configured to transmit and receive test pattern signals between the interface device 150 and the DRAM to calculate the optimum timing level for each interconnect. In some embodiments, the interface device 150 can have any number (e.g., 8, 16, 32, 64,

128, etc.) of I/O pads that provide for external system interconnection. In some embodiments, test pattern signals are transmitted and received between I/O terminals of the interface device 150 and the DRAM, and a training algorithm is executed to calculate an optimum timing for each input/output connection.

5 In some embodiments, the interface device 150 is configured to perform Error Check and Correction (ECC) during data communication between the memory devices 110, 120 and the processor module 160.

 In some embodiments, DRAM controller 151 and Flash controller 152 within interface device 150 are configured to control memory devices 110, 120
10 by providing signals on 128-bit data busses. In some embodiments, the data busses provided within Z-vias 135 and 145 can have widths other than 128 bits. In some embodiments, the memory commands provided to the memory devices 110, 120 include commands to cause a programming operation to write data using the 128-bit data busses passing through Z-vias 135, 145 to the memory
15 cells within memory devices 110, 120, a read operation to read data from the memory cells within memory devices 110, 120, and an erase operation to erase data from all or a portion of memory cells within memory devices 110, 120.

 In some embodiments, memory devices 110, 120 include a flash memory device. In some embodiments, memory cells within memory arrays in memory
20 devices 110, 120 can be arranged in a NAND flash memory arrangement. In some embodiments, memory cells within memory arrays in memory devices 110, 120 can include flash memory cells arranged in a NOR flash memory arrangement.

 In some embodiments, memory devices 110, 120, interface device 150
25 and processor module 160 are included in the same electronic package. In some embodiments, the processor module 160 resides in a different package from that of memory devices 110, 120 and interface device 150.

 In some embodiments, memory devices 110, 120 are stacked on one side of the interface device 150 and processor module 160 is positioned on the other
30 side of the interface device 150 in a three-dimensional manner. In some embodiments, the memory arrays can interface with interface device 150 with a standard, variable width connection. This would allow different types/sizes of memory arrays to be connected to the interface device 150. In some

embodiments, each side of the interface device can have a regular array of vias. In some embodiments, a group of vias can contain a memory channel that can be used to couple to the processor module 160. In some embodiments, the user can device as to how many memory channels are desired for a given processor
5 module.

FIG. 2 illustrates a top view of a memory system 200, in accordance with some embodiments of the invention. Memory system 200, which can be similar to or identical to the memory system 100 of FIG. 1, includes a substrate 210 having memory devices 220, 230, 240 and 250 disposed thereon. In some
10 embodiments, each of memory devices 220, 230, 240 and 250 can include multiple memory arrays 260. In some embodiments, each of memory devices 220, 230, 240 and 250 includes a volatile memory device, a non-volatile memory device, or a combination of both. For example, memory devices 220, 230, 240 and 250 can include a DRAM device, a static random access memory (SRAM)
15 device, a flash memory device, or a combination of these memory devices. In some embodiments, localized vias 270 are provided around memory devices 220, 230, 240 and 250 that provides a path for through wafer interconnects to couple other devices (such as interface device 150, processor module 160 shown in Fig. 1) that may be disposed above and below substrate 260.

FIG. 3 illustrates a perspective view of a memory system 300 similar to that shown in FIG. 1, according to some embodiments of the invention. Memory system 300 includes a substrate 310 having a matrix of solder balls 344, an interface device 150, a first memory array 110-1, a second memory array 110-2, a third memory array 120-1, and a fourth memory array 120-2. In some
25 embodiments, the first memory array 110-1 is disposed on the second memory array 110-2 and the second memory array 110-2 is disposed on the interface device 150. In some embodiments, the third memory array 120-1 is disposed on the fourth memory array 120-2 and the fourth memory array 120-2 is disposed on the interface device 150. The first memory array 110-1 is coupled to Through
30 Wafer Interconnects (TWI) 321, which in turn are coupled to the interface device 150. In some embodiments, TWI 321 passes through a set of vias 322 within second memory array 110-2 to connect with interface device 150. In some embodiments, TWI 321 passes through a set of vias 323 within interface device

150 to connect to devices within the substrate 310. In some embodiments, the second memory array 110-2 is coupled to interface device 150 using connection pins 326. In some embodiments, connection pin 330 communicatively couples memory array 120-2 with interface device 150 and TWI 325 communicatively couples memory array 120-1 with interface device 150. In some embodiments, connection pins 332 allow for communication between routing elements 152-158 embedded within interface device 150 and other devices (such as processors 162-168) embedded within substrate 160. In some embodiments, substrate 1600 can include a circuit board having circuits communicatively coupled to the interface device 150.

In some embodiments, memory system 300 can be included in computers (e.g., desktops, laptops, hand-held devices, servers, Web appliances, routing elements, etc.), wireless communication devices (e.g., cellular phones, cordless phones, pagers, personal digital assistants, etc.), computer-related peripherals (e.g., printers, scanners, monitors, etc.), entertainment devices (e.g., televisions, radios, stereos, tape players, compact disc players, DVD players, video cassette recorders, DVD recorders, camcorders, digital cameras, MP3 (Motion Picture Experts Group, Audio Layer 3) players, video games, watches, etc., and the like.

FIG. 4 illustrates a cross-sectional view of system 400 similar to that shown in FIG. 1, according to some embodiments of the invention. System 400 includes a package 402 having an interface device 150 within the package 402, a processor module 406 attached to the package 402 using a matrix of solder balls 408, and a memory device 410. In some embodiments, memory device 410 includes memory arrays 410-1, 410-2, a memory buffer 430 and TWI 412-1, 412-2 that connect memory arrays 410-1, 410-2 to memory buffer 430. In some embodiments, processor module 406 includes processors 162-168. In some embodiments, memory device 410 includes memory arrays similar to 110-1, 110-2, 110-3 and 110-4. In some embodiments, memory device 410 includes memory arrays similar to 120-1, 120-2, 120-3 and 120-4.

In some embodiments, the interface device 150 is disposed on the processor 406 and the memory device 410 is disposed on the interface device 150. In some embodiments, package 402 may include a circuit board 403 and interface device 150 is disposed on the circuit board 403 and the memory device

410 is disposed on the interface device 150. In some embodiments, memory device 410 is communicatively coupled to the processor 406 using a narrow high speed bus 420. In some embodiments the narrow high speed bus is configured to communicate data at 128 GB/s. In some embodiments, the narrow high speed
5 bus 420 provides for full duplex communication having 64 GB/s for read data and 64 GB/s for write data.

In some embodiments, package 402 is attached to a matrix of solder balls 404 that enables system 400 to be mounted on a circuit board having other devices. In some embodiments, memory device 410 is attached to a matrix of
10 solder balls 414 used to communicatively couple the memory device 410 to package 402.

In some embodiments, interface device 150 acts as an interconnection device and an I/O driver. In some embodiments, the interface device 150 includes traditional functional blocks present within DRAM dice, such as I/O
15 pads, delayed-locked loop (DLL) circuitry, and First In, First Out (FIFO) circuitry that provides read and write pointers, storage, and control logic. Transferring these functional blocks, which are well-known to those of ordinary skill in the art, from the DRAM dice into the interface device 150 can allow for the increase in storage area of the DRAM dice.

In some embodiments, the interface device 150 is coupled to 32 different
20 memory devices (each having several memory arrays; other numbers of devices are possible) with independent interconnects such as the TWI shown in FIG. 4. In some embodiments, the interface device 150 is coupled to a set of TWI that are configured to provide different types of interconnection based on the end
25 user application. In some embodiments, the TWI can be configured to provide connectivity between independent groups of DRAM and the interface device 150.

In some embodiments, the interface device 150 is configured to transmit and receive test pattern signals between the interface device 150 and the DRAM
30 to calculate the optimum timing level for each interconnect. In some embodiments, the interface device 150 can have any number (e.g., 8, 16, 32, 64, 128, etc.) of I/O pads that provide for external system interconnection. In some embodiments, test pattern signals are transmitted and received between I/O

terminals of the interface device 150 and the DRAM, and a training algorithm is executed to calculate an optimum timing for each input/output connection.

In some embodiments, the interface device 150 is configured to perform power management within system 400, wherein the interface device 150 is operated at a voltage sufficient to prevent the generation of error bits in the communication between the interface device 150 and memory device 410.

In some embodiments, the interface device 150 is configured to perform error detection and/or correction during data communication between the memory device 410 and the processor module 406. In some embodiments, the interface device 150 is configured to perform power management of system 400 by operating at some voltage that is determined beforehand such that no error bits are generated at that voltage.

In some embodiments, the interface device 150 includes a diagnostic and Built in Self-Test (BIST) module 152. In some embodiments, the BIST module is coupled to a maintenance bus 154 that is coupled between the interface device 150 and memory device 410. In some embodiments, the BIST module is configured to send command signals and data to the memory device 410 through the maintenance bus 154 that are received from a user. In some embodiments, the maintenance bus 154 is also used to receive the results of the diagnostic testing. In some embodiments, the diagnostic and BIST module 152 generates control signals and forwards the user supplied command and data signals to carry out the commands of the user. For example, the diagnostic and BIST module 152 may invoke a pattern generator program or hardware module to begin generating a test pattern in accordance with the user's commands and data, and also forward the user provided memory commands to a sequencer 156 for translation into control signals that will be applied to the memory device 410 for carrying out diagnostic operations of memory device 410.

FIG. 5A illustrates a data structure 500 of a command packet used in memory system shown in FIG. 1, according to various embodiments of the invention. In some embodiments, the user can interface to memory systems 110 and 120 using "read" and "write" packets. In other words, the actual operation of the memory device (for e.g. a DRAM) is virtualized behind interface device 150. In some embodiments, memory commands and data are sent using separate

read and write channels (uplinks and downlinks). In some embodiments, data structure 500 contains varies fields that includes request ID digits 502, valid indicator digits 504, header digits 506, tail digits 508, command digits 510, address digits 512 and CRC (cyclic redundancy code)/ECC (error correction code) digits 514. In some embodiments, request ID digits 502 are used to identify the address of the sender sending the command digits 510. In some embodiments, the request ID digits are used to re-order the data that is returned to the sender. In some embodiments, valid indicator digits 504 are used to indicate the validity of the data being transferred. In some embodiments, header digits 506 are used to identify the start of a data block transfer and the tail digits 508 are used to identify the end of a data block transfer. In some embodiments, the command bits are used to read, write or configure a memory device 110, 120. In some embodiments, the address digits 512 contain X, Y address for routing data to memory locations within the memory device 110, 120, wherein, the X would indicate the sending entities address and Y would indicate the receiving entities address. In some embodiments, CRC/ECC digits are used to provide error detection and correction.

FIG. 5B illustrates a data structure 520 of a data packet used in memory system shown in FIG. 1, according to various embodiments of the invention. In some embodiments, data structure 520 contains varies fields that includes request ID digits 522, valid indicator digits 524, header digits 526, tail digits 528, data digits 530 and CRC (cyclic redundancy code)/ECC (error correction code) digits 532. In some embodiments, request ID digits 522 are used to identify the address of the sender sending the data digits 530. In some embodiments, valid indicator digits 504 are used to indicate the validity of the data digits 530 being transferred. In some embodiments, header digits 526 are used to identify the start of a data block transfer and the tail digits 528 are used to identify the end of a data block transfer. In some embodiments, the data width represented by data digits field 530 can be programmable to any of 32, 64, 128 digits. In some embodiments, CRC/ECC digits are used to provide error detection and correction.

In some embodiments, the router elements can be programmed based on memory ranges of the memory devices to which they are communicatively

coupled. In some embodiments, the first packet establishes a connection between the memory system and interface device until the “end/tail” packet arrives.

FIG. 6 illustrates a flow chart of a method 600 of operation of system shown in FIG. 1 and FIG. 4, according to some embodiments of the invention. At 602, method 600 includes sending and receiving data between a plurality of processors in a processor module 160 and a plurality of routing elements in an interface device 150. At 604, method 600 includes routing data to a plurality of memory arrays within at least one memory device using the plurality of routing elements in the interface device 150. At 606, method 600 includes storing data in the plurality of memory arrays. In some embodiments, at 606, the method includes storing the data in a DRAM array. In some embodiments, at 606, the method includes storing the data in a NAND flash array. In some embodiments, at 606, the method includes storing the data in a NOR flash array. At 608, method 600 includes retrieving data from the plurality of memory arrays using a plurality of through wafer interconnects provided in vias of an interface device and/or at least one memory device, the interface device being above or below and being coupled to at least one memory device. At 610, method 600 includes routing the retrieved data to the plurality of processors in the processor module using the plurality of routing elements and interconnects. In some embodiments, the plurality of processors communicates with the memory devices 110, 120 over a high speed chip-to-chip interconnect 420. In some embodiments, memory devices 110, 120 can also be coupled to each other using a similar high speed chip-to-chip interconnect. In some embodiments the high speed chip-to-chip interconnect is configured to communicate data at 128 GB/s. In some embodiments, the high speed chip-to-chip interconnect 420 provides for full duplex communication having 64 GB/s for read data and 64 GB/s for write data.

In some embodiments, the interface device 150 is configured to communicatively couple the plurality of memory arrays within memory devices 110, 120 to the plurality of processors 162-168 using the plurality of routing elements 151-159 based on commands stored within packet data exchanged between them.

FIG. 7 illustrates a schematic diagram 700 showing a memory system coupled to a processor module, according to some embodiments of the invention. FIG. 7 includes a system 710 including a plurality of processor modules 406 and a plurality of memory device 410, similar to that shown in FIG. 7. In some
5 embodiments, the memory devices 410 adjacent to the processor modules 406 are coupled to the processor modules 406 closest to it using a narrow high speed bus 420. In some embodiments, the memory devices 410 shown in FIG. 7 can be cubical in structure and coupled to the four adjacent cubical memory devices using a narrow high speed bus 420. In some embodiments, the narrow high
10 speed bus 420 is configured to communicate data at 128 GB/s. In some embodiments, the narrow high speed bus 420 provides for full duplex communication having 64 GB/s for read data and 64 GB/s for write data.

The apparatus, systems, and methods disclosed herein can provide for increased speed and throughput while accessing memory arrays in addition to
15 achieving a higher density of memory arrays compared to conventional designs. In some embodiments, as a result, the DRAM die size is also reduced. Additionally, use of through wafer interconnects allows for a larger number of interconnects traversing shorter distances and consequently improving the speed of each connection established by the interconnects. Moreover, the apparatus,
20 systems and methods disclosed herein provides for processors coupled to memory arrays that are capable of handling more bandwidth with reduced latency in designs having improved packing density.

The accompanying drawings that form a part hereof show, by way of illustration and not of limitation, specific embodiments in which the subject
25 matter can be practiced. The embodiments illustrated are described in sufficient detail to enable those skilled in the art to practice the teachings disclosed herein. Other embodiments can be used and derived therefrom, such that structural and logical substitutions and changes can be made without departing from the scope of this disclosure. This Detailed Description, therefore, is not to be taken in a
30 limiting sense, and the scope of various embodiments is defined only by the appended claims and the full range of equivalents to which such claims are entitled.

Although specific embodiments have been illustrated and described herein, any arrangement calculated to achieve the same purpose can be substituted for the specific embodiments shown. This disclosure is intended to cover any and all adaptations or variations of various embodiments.

5 Combinations of the above embodiments and other embodiments not specifically described herein will be apparent to those of skill in the art upon studying the above description.

The Abstract of the Disclosure is provided to comply with 37 C.F.R. §1.72(b) requiring an abstract that will allow the reader to quickly ascertain the
10 nature of the technical disclosure. It is submitted with the understanding that it will not be used to interpret or limit the scope or meaning of the claims. In the foregoing Detailed Description, various features are grouped together in a single embodiment for the purpose of streamlining the disclosure. This method of disclosure is not to be interpreted to require more features than are expressly
15 recited in each claim. Rather, inventive subject matter can be found in less than all features of a single disclosed embodiment. Thus the following claims are hereby incorporated into the Detailed Description, with each claim standing on its own as a separate embodiment.

Methods, apparatus, and systems have disclosed memory device
20 connection schemes that can increase memory capacity within a given horizontal space. Various embodiments include a substrate, an interface device disposed on the substrate, a first memory die having a plurality of memory arrays disposed on the interface device, with the first memory die coupled to a plurality of through wafer interconnects (TWI). Various embodiments include a second
25 memory die having a plurality of memory arrays disposed on the first memory die, the second memory die including a plurality of vias, wherein the plurality of vias are configured to allow the plurality of TWI to pass through the second memory die. The second memory die, in turn, can be coupled to a second plurality of TWI, and the interface device can be used to communicatively
30 couple the first memory die and the second memory die using the first and second plurality of TWI. Additionally, the interface device includes a plurality of routing elements that communicatively couples the first memory die and

second memory die with a plurality of processors using the plurality of through wafer interconnects.

One or more embodiments provide an improved mechanism for interconnecting memory devices. Furthermore, various embodiments described
5 herein can improve the density of some memory arrays within memory devices and, as a result, reduce the size of memory devices.

CLAIMS

What is claimed is:

- 5 1. An apparatus comprising:
 an interface device disposed above or below a processor module, the
interface device including a plurality of routing elements;
 at least one memory device disposed above or below the interface device
and including a first plurality of memory arrays, the first plurality of memory
10 arrays coupled to the interface device using a plurality of interconnects provided
in vias in at least one of the interface device and the memory device; and
 wherein the interface device is configured to communicatively couple the
first plurality of memory arrays to the processor module using the plurality of
routing elements and the interconnects.
15
2. The apparatus of claim 1, wherein the interface device is configured to
provide memory commands to the memory device.
3. The apparatus of claim 1, further comprising a second memory device
20 including a second plurality of memory arrays.
4. The apparatus of claim 3, wherein the first memory device further
comprises another plurality of interconnects provided in vias to couple the
second plurality of memory arrays to the plurality of routing elements in the
25 interface device.
5. The apparatus of claim 4, wherein the interface device includes circuitry
configured to perform sequencing of at least one DRAM array.
- 30 6. The apparatus of claim 1, wherein the interface device is configured to
perform power management of the apparatus, and wherein the interface device is
configured to operate at a voltage sufficient to preclude error bit generation.

7. The apparatus of claim 1, wherein the interface device further comprises an input/output driver circuit.
8. The apparatus of claim 1, wherein the interface device is configured to
5 implement a refresh scheme to control error rates based on the type of array used in the memory device.
9. The apparatus of claim 1, wherein the interface device is configured to
10 implement a refresh scheme to control error rates based on the signal characteristics of the memory device.
10. The apparatus of claim 3, wherein the interface device is configured to perform Error Check and Correction (ECC) during data communication between the at least one memory device and the processor module.
15
11. The apparatus of claim 3, wherein the second plurality of memory arrays comprises a NAND flash memory array.
12. The apparatus of claim 3, wherein the second plurality of memory arrays
20 comprises a NOR flash memory array.
13. The apparatus of claim 5, wherein the first and second plurality of through wafer interconnects extend in a vertical direction with respect to a plane including the first memory device.
25
14. The apparatus of claim 3, wherein the interconnects comprises through wafer interconnects that form a data bus coupling the plurality of first and second memory arrays and the interface device.
- 30 15. A system comprising:
a processor module including a plurality of processors;

an interface device disposed on the processing module, the interface device including a plurality of routing elements and a first plurality of vias adapted to pass a plurality of interconnects;

5 a first memory device disposed above or below the interface device, the first memory device including a plurality of DRAM memory arrays and a second plurality of vias adapted to pass at least one of the plurality of interconnects to couple the plurality of DRAM memory arrays to the interface device;

10 a second memory device disposed above or below the interface device, the second memory device including a plurality of FLASH memory arrays and a third plurality of vias adapted to pass at least one of the plurality of interconnects to couple the plurality of FLASH memory arrays to the interface device; and

15 wherein the plurality of routing elements in interface device is configured to communicatively couple the plurality of DRAM memory arrays and the plurality of FLASH memory arrays to the plurality of processors using the plurality of routing elements.

16. The system of claim 15, wherein the interface device is disposed above a circuit board within a package.

20 17. The system of claim 15, wherein the first and second memory device is coupled to the processor module using a portion of the plurality of interconnects.

18. The system of claim 15, wherein the pluralities of routing elements are coupled to the plurality of processors using the plurality of interconnects.

25

19. The system of claim 15, wherein the pluralities of routing elements are coupled to the plurality of processors using connection pins.

20. The system of claim 15, wherein the processor module comprises a
30 substrate with embedded processors.

21. The system of claim 15, wherein the processor module comprises an ASIC.

22. The system of claim 15, wherein the processor module includes a single-core processor.

5 23. The system of claim 15, wherein the processor module includes a multi-core processor.

24. A method comprising:

receiving data from a processor module at an interface device disposed
10 above or below the processor module;
routing the received data to a plurality of memory arrays within at least one memory device disposed above or below the interface device using a plurality of routing elements in the interface device and a plurality of interconnects provided in vias of the interface device and/or at least one memory
15 device;

retrieving data from a plurality of memory arrays of at least one memory device using interconnects provided in vias of the interface device and/or the at least one memory device, the interface device being disposed above or below and being coupled to the at least one memory device; and

20 routing retrieved data to the processor module using the plurality of routing elements and the interconnects.

25. An apparatus comprising:

an interface device coupled to a processor module including a plurality of
25 processors, the interface device including a plurality of routing elements and the interface device configured to send and receive packet data to and from at least one of the plurality of processors;

at least one memory device including a plurality of memory arrays coupled to the interface device using a plurality of interconnects provided in vias
30 in at least one of the interface device and the memory device, the memory device configured to send and receive data to and from the interface device; and

wherein the interface device is configured to communicatively couple the plurality of memory arrays to the plurality of processors using the plurality of routing elements based on commands stored in the packet data.

5 26. A method comprising:

receiving packet data from a processor module at an interface device in an electronic package;

routing the received packet data using a plurality of routing elements in the interface device based on the address provided within the packet data;

10 retrieving stored data from a plurality of memory arrays of at least one memory device based on memory commands provided within the packet data; and

routing retrieved stored data to the processor module using the plurality of routing elements.

15