Memory interface device and method for accessing memories
Speicherschnittstellenvorrichtung und Verfahren zum Speicherzugriff
Dispositif d’interface de mémoire et méthode d’accès aux mémoires
Description

[0001] The present invention relates to a memory interface device, in particular, for memories comprising a plurality of memory regions.

[0002] One type of known memory device is the synchronous dynamic random access memory (SDRAM). A typical example of an SDRAM is shown in Figure 1. The SDRAM 2 comprises two memory banks 4a and 4b. In some known SDRAMs, four memory banks are in fact provided. Each memory bank 4a and 4b contains a plurality of rows R which are sometimes referred to as pages. Each memory bank 4a and 4b also contains a plurality of columns C which intersect the rows R. A memory location is therefore identified by the bank number, the row number and the column number. To access a given memory location (or word) a memory interface unit 6 is provided. The memory interface unit 6 receives an input 7 which provides the address of the word to be accessed. The address identifies the memory bank, row and column of the word to be accessed.

[0003] Based on the address input to the memory interface unit 6, control signals 12 are generated which are output to a respective column control unit 8 and to a respective row control unit 10. Each bank 4a and 4b has its own row and column control units 8 and 10. The row and column control units 8 and 10 are sometimes referred to as row and column decoders respectively. The row control unit 10 will, in accordance with the address input to the memory interface unit, select a row R in the selected memory bank 4a or 4b. Once the row R or page has been selected (or opened), then the appropriate column C is selected by the column control unit 8, again in accordance with the input address.

[0004] The operation to open a page or row R will generally take several cycles. Once a page or row R has been opened, any word in that page or row R can be selected in one cycle. Thus a first word at a first column C location can be accessed in one cycle and a different word in that same row R but in a different column C location can be accessed in the next cycle. Once all the required accesses in a given row R or page have been completed, the open page or row R needs to be closed. This is achieved by the row control unit 10 precharging all the rows R including the selected row in the selected memory bank 4a or 4b to a given voltage. This closing operation must be completed before another page or row in the same bank can be selected or opened. This closing operation also takes several cycles.

[0005] Reference will now be made to Figure 2 which shows a sequence of steps which occurs when eight words from a first selected row R in a first memory bank 4a are read and then eight words from a second selected row in the second memory bank 4b are read. As can be seen, the first six cycles A are used to open the first selected row R and read the first required word in that row of the first memory bank 4a. The next seven cycles B are used to read the remaining required seven words in the opened row R. The next three cycles C are required to close the first selected row R in the first bank 4a. The next six cycles D are used to open the second selected row R in the second memory bank 4b and read the first required word from that row. The next seven cycles E are required to read the other seven required words in the second selected row. The last three cycles F are required to close the second selected row R in the second memory bank 4b. Thus, in order to read eight words from a given row in a memory bank requires 16 cycles even though the reading operation itself only requires 8 cycles. This therefore reduces the efficiency of the memory device and increases the time required in order to complete read and write operations.

[0006] It is therefore an aim of embodiments of the present invention to reduce the number of cycles required to carry out an operation in respect of a memory having a plurality of memory regions.

[0007] US 5,502,835 describes an integrated circuit microprocessor which reads data from an external memory device through early overlapping memory access cycles. A memory device with a slow memory core is therefore able to pipeline a second memory access prior to completion of a first access, thereby increasing system efficiency.

[0008] US 4,924,375 provides a memory organisation scheme. Sequential pages of memory are interleaved between memory banks so that memory accesses which are a page apart will be to two different memory banks. Accesses to bits in the same page as a previous access omit the row pre-charge cycle. Accesses to a separate memory bank from the previous access may be initiated before the previous access cycle is completed. System efficiency is thereby increased.

[0009] US 5,615,355 describes a method and apparatus for data storage and retrieval including first and second banks of refreshable memory. A controller system controls FIFO buffers enabling the continuous writing of user data to one of the memory banks at a given address while simultaneously enabling reading of data at an address from the other bank.

[0010] System efficiency is thereby increased.

[0011] According to one aspect of the present invention, there is provided a memory interface for generating a plurality of commands for controlling a memory having first and second memory regions, only one of said memory regions being accessible at a time, each memory region comprising a plurality of rows, characterised in that said device comprises: a
buffer for storing a plurality of received memory requests for said memory, said memory requests each including information as to the row to be accessed of said respective memory region, and said buffer arranged to provide a respective output for each memory request, each of said outputs indicating said row to be accessed for the respective memory request; a detector arranged to receive said plurality of outputs from said buffer and to detect a next different row in each of said memory regions to be subsequently selected, said detector providing an output signal indicative of said detected next different row for each of said memory regions; and a command provider for providing a sequence of commands in response to said received memory requests and said output signals provided by said detector for controlling said memory, said command sequence being arranged so that a row of one of the first and second memory regions is accessed while said detected next different row of the other of the first and second memory regions is being selected.

Thus, as it is possible for one memory region to be accessed whilst the other memory region is being selected or deselected, the number of cycles required to access a burst may be reduced as compared to the prior art described in relation to Figure 1.

Preferably, the portion of the memory regions which is selected or deselected comprises a row. A row is sometimes referred to as a page in relation to certain memory devices. When a portion of the first or second memory regions is accessed, information may be read from the respective portion. Alternatively, when a portion of the first or second memory regions is accessed, the information is written into the respective portion.

A register for storing a plurality of access requests may be provided, said access requests each including information as to the portion to be accessed and the memory region. This information may comprise address information. The register means may comprise a first-in-first-out register or may be any other suitable buffer. A detector for detecting the portion which is next to be accessed in each of the memory means may be provided. The detector may be arranged to receive from the register information as to the next portion which is to be accessed in each memory region. The detecting means may receive from the register, address information in respect of each request stored in the memory means.

A comparer may be provided for comparing the portion of a memory region which is currently selected with a portion of the memory region which is next to be accessed and outputting a signal based on the comparison.

The command provider may be arranged to process received requests in a nonsequential manner if a later request specifies the same memory location of a given memory region as an earlier request with intervening requests for said given memory region being processed after said later request. In this way if, for example, a given page is open, a later request for that same page can be processed before a request for a different page is located. This reduces the number of times that a memory page needs to be opened and closed, thus increasing the cycle time of adjacent channels as small as possible.

For a better understanding of the present invention and as to how the same may be carried into effect, reference will now be made by way of example to the accompanying drawings in which:

- Figure 1 is a schematic diagram of a known SDRAM;
- Figure 2 illustrates the timing of the SDRAM of Figure 1;
- Figure 3 is a SDRAM with circuitry embodying the present invention;
- Figure 4 illustrates the timing of the embodiment shown in Figure 3;
- Figure 5 illustrates a modified version of the circuitry shown in Figure 2;
- Figure 6 schematically shows a conventional integrated circuit device and three data storage devices to which the integrated circuit device may be connected;
- Figure 7 shows an arrangement for connecting a Direct Rambus memory to a conventional integrated circuit;
- Figure 8 shows instantaneous and average data transfer rates for a Direct Rambus memory;
- Figure 9 shows a Direct Rambus connected to an integrated circuit via an interface;
- Figure 10 schematically illustrates how the contents of the interface of Figure 5 vary with time; and
- Figure 11 shows a Direct Rambus connected to a number of integrated circuits via a number of interfaces of the type shown in Figure 9.

Reference will now be made to Figure 3 which illustrates a memory interface 22 embodying the present invention with a SDRAM 20. The memory interface 22 controls the accessing of the SDRAM 20. As with the SDRAM shown in Figure 1, the SDRAM 20 comprises a first and a second memory bank 21a and 22b. Each memory bank 21a and 21b has a plurality of rows and columns which define memory locations. Each memory bank 21a and 21b is also provided with a row control unit 60 and a column control unit 58, similar to those described in relation to Figure 1.

The memory interface 22 comprises a requester 24 which generates or receives requests to access particular locations in the SDRAM 20. In practice the requester 24 is a computer processing unit (CPU). Each request will identify the memory bank of the SDRAM 20, the row (page) and the column which are to be accessed. The requests output from the requester 24 are input to a FIFO (first in first out) buffer 26 where they are stored in the order in which they are received. The requests output by the FIFO 26 in the same order in which they are received.
Each location of the FIFO 26 which stores an address provides an output 28 to a detection circuit 30. In the example shown in Figure 3, the FIFO 26 is able to store nine different requests and accordingly the FIFO has nine outputs 28 which are connected to the detection circuit 30. The detection circuit 30 is arranged to detect which row (page) in the first bank 21a of the SDRAM 20 is next to be accessed as well as the next page which is to be accessed in the second bank 21b of the SDRAM 20. In the embodiment illustrated in Figure 3, page N is the next page to be accessed in the second bank 21b whilst page J is the next page to be accessed in the first bank 21a. The detect circuit 30 will generally be a combinatorial logic circuit but can take any other suitable form.

The detect circuit 30 provides two outputs 32 and 34. The first output 32 contains information as to the next page to be accessed in the first bank 21a. The second output 34 contains information as to the next page to be accessed in second bank 21b. The first and second outputs 32 and 34 of the detect circuit 30 are connected to respective bank managers 36 and 38. The first bank manager 36, which receives the first output 32 from the detect circuit 30 also receives a second input 41 from a memory or buffer 40 which stores the current page which is currently open or selected in the first bank 21a. The first bank manager 36 thus compares the page of the first bank 21a which is currently open with the next page of the first bank 21a which is to be accessed. The first bank manager 36 thus determines the next action for the first bank 21a. If the current page which is open and next page which is to be accessed are the same, first bank manager 36 will indicate that the next action for the first bank 21a will be to access the of the required location on the open page or alternatively provide no output. If the current page which is open and the page which is next to be accessed are not the same, the first bank manager 36 will provide an output 44 which indicates that the next action for the first bank 21a will be to close the page which is currently open and then to open the page which is next to be accessed.

In some circumstances, there will be no page which is currently open. The first bank manager 36 will still output the next action which is required at the first bank 21a. In a preferred embodiment of the present invention the first bank manager 36 is arranged to output one action at a time. For example, the first bank manager 36 is arranged to provide an output which indicates that a page is to be closed. The first bank manager 36 is arranged to provide a subsequent output indicating which page in the first bank 21a is to be opened. This subsequent instruction may, but not necessarily, be provided when the previous instruction has been completed or is likely to be completed shortly. In one embodiment of the present invention each instruction from the first bank manager 36 is output once. In other embodiments of the present invention, each instruction is output until that instruction is acknowledged.

The second bank manager 38 operates in a similar manner to the first bank manager 36. In particular, the second bank manager 38 receives the second output 34 from the detect circuit and an output 41 from a second memory 42 which stores information as to the page which is currently open in the second bank 21b. As with the first bank manager 36, the second bank manager 38 provides an output 46 which determines the next action for the second bank 21b.

The two outputs 44 and 46 from the first and second bank managers 36 and 38 are input to a multiplexer 48 along with the next request which has the next address which is to be accessed. The next request which is to be accessed is output from the FIFO 26 to the multiplexer 48. This next request is the oldest request stored in the FIFO 26. The operation of the multiplexer 48 is controlled by a multiplexer controller 50 which arbitrates between the outputs 44 and 46 of the first and second bank managers 36 and 38 and the request output from the FIFO 26. The multiplexer controller 50 receives an input from the FIFO 26 which provides the next request, an input from the first memory 40 as to which, if any, page is open in the first bank 21a and an input from the second memory 42 as to which, if any, page is currently open in the second bank 21b. Based on this information, the multiplexer controller 50 decides what action should be carried out in the next clock cycle and controls the multiplexer 48. Effectively the multiplexer controller 50 acts as an arbiter and provides a series of commands. Usually, but not necessarily, one command may be provided per clock cycle. An open or close page command may take priority over a read command so that one bank may have a page being opened/closed whilst the other bank is being read. However some commands may include the open and/or close page instruction within a single command with the read/write instructions. In other words, the multiplexer 48 provides a suitable multiplexed output of commands so that, for example, reading of one bank may take place whilst the other bank has a row which is being opened or closed.

The output of the multiplexer 48 is input to a memory logic device 56. The output of the multiplexer 48 constitutes the outputs of the memory interface 22. Based on the input, the memory logic device 56 will cause a row to be opened or closed or a location to be accessed (read or written to). The memory logic device 56 controls the row and column control units 58 and 60 in accordance with the output from the memory interface 22. The location can either be read or written to. The memory logic 56 acts as a command decoder and also generates the control logic.

Because of the detect circuit 30, it is possible to ensure that a row in one bank can be opened or closed at the same time that a row in the other bank is being read. This significantly reduces the number of cycles taken to complete operations. The column control units 58 control the accessing of columns of the respective banks. The row control units 60 control the selection and deselection of the rows of the first and second banks 21a and 21b.

The multiplexed output from the multiplexer 48 is received by the memory logic device 56. As discussed hereinbefore, the output of the multiplexer 48 consists of a series of commands which are achieved by the memory logic device 56.
In preferred embodiments of the present invention, the two banks cannot be read from or written to at the same time. However reading or writing in one bank can take place at the same time that the other bank is having a page opened or closed.

Accordingly, when the first burst has been completely accessed, the required row of the second bank 21b can be immediately accessed in order to access the eight words of the second burst, this occurring in the eight cycles marked L. At the same time that the second burst is being accessed, the row of the first bank 21a from which the first word was accessed can be closed in the three cycles marked M. In the six cycles marked N, which occur at the same time as five of the cycles marked L in which the words of the second burst are read, the next required row in the first memory bank 21a is opened and the first word of a third burst is read.

Thus, the first two eight word bursts can be read in 21 cycles. This compares favourably with the 29 cycles required with the known SDRAM described in relation to Figure 1. Additionally, any subsequent burst in this mode will only require eight cycles.

With the conventional SDRAM described in relation to Figure 1, any subsequent burst requires 16 cycles.

Reference is made to Figure 5 which shows a modified version of the present invention. Two requesters 100 and 102 in the form of CPU1 and CPU2 are provided. The requests from the requesters 100 and 102 are input into respective first and second FIFO buffers 104 and 106. These FIFO buffers are the same as FIFO 26 of Figure 3. The output of the first and second FIFO buffers 104 and 106 are input to a common unit 108 which includes the remaining circuitry of the memory interface unit 22 of Figure 3. The bank defect circuit (not shown), will thus look at the contents of both the first and second FIFOs 104 and 106 but will otherwise operate in a similar manner to that of Figure 3. The first and second FIFOs 104 and 106 and the common unit 108 define a memory interface unit 110. The output of the common unit 110 is input to the memory logic for a SDRAM 20.

In preferred embodiments of the present invention, the input is arranged to receive an input from a requester which may be in the form of a computer processing unit. In a preferred embodiment of the present invention, the input is arranged to receive an input from a plurality of requesters. It is preferred that a respective register be provided for storing the requests from each requester.

It should be appreciated that whilst embodiments of the present invention have been described in relation to an SDRAM, an embodiment of the present invention is applicable to any other type of memory which has two separate banks or regions which cannot be accessed at the same time. Embodiments of the present invention can be used with, for example, other types of DRAM.

In the embodiment described hereinbefore, two banks are shown. However, it should be appreciated that any other suitable number of banks may be provided, for example four.

In the illustrated embodiment, the FIFO 26 can have any suitable number of locations.

In one modification to the embodiment of the present invention, the detect circuit 30 may be arranged to check that the FIFO 26 does not contain any more requests for an open page before closing that page. This may involve reordering of the requests and accordingly additional storage capacity may be required in order to ensure that the information read out of the memory banks is ultimately output in the same order in which the requests are made. The additional memory may be required to receive the output of the memory banks.

The SDRAM 20 itself may be of any suitable conventional design or may be specially modified to be used with the accessing device 22. Any other suitable memory device may be used with embodiments of the present invention.

In such systems as described hereinbefore which use integrated circuits, one of the areas which restricts the overall system performance is the rate of data transfer between the memory device and the internal bus of an operational
circuit which accesses that memory. A number of memory devices have been recently introduced which have improved
data transfer rates in comparison to conventional memory devices. For example, conventional SDRAM (synchronous
dynamic random access memory) typically has a data transfer rate of 32 bits at 100MHz. An improvement to this is
double data rate (DDR) SDRAM which is capable of transferring double the data rate than a conventional SDRAM and
hence has a data transfer rate of 32 bits at 200MHz. There are also available memory devices known as Direct Rambus
memories (RDRAM Rambus Dynamic Random Access Memory) which have a transfer rate of 16 bits at 800MHz.
'Rambus' and 'Direct Rambus' are trade marks of Rambus Inc.

Presently conventional integrated circuits typically have an internal system bus with a data transfer rate of 32
bits at 100MHz. Figure 6 schematically illustrates a conventional integrated circuit 61 with an internal system bus 65
and three known memory devices, a conventional SDRAM 62, a double data rate SDRAM 63 and a Direct Rambus
memory 64. (In practice only one of the three memory devices is provided). Each of the memory devices 62, 63 and
64 has an output bus which in use is coupled to the internal system bus 65 of the integrated circuit. The output bus 66
of the conventional SDRAM 62 has a data transfer rate of 32 bits at 100MHz and is therefore entirely compatible with
the internal bus 65 of the integrated circuit 61, which as shown, also has a data transfer rate of 32 bits at 100MHz.
However, the output bus 67 of the DDR SDRAM 63 has a data transfer rate of 32 bits at 200MHz and the output bus
68 of the Direct Rambus memory 64 has a data transfer rate of 16 bits at 800MHz. Accordingly the output buses 67 and
68 of the DDR SDRAM 63 and the Direct Rambus memory 64 are not compatible with the internal system bus 65 of the
integrated circuit 61 in terms of data rate. Accordingly, with the existing conventional internal bus system of the integrated
circuit, the higher data transfer rate of the DDR SDRAM and the Direct Rambus cannot be readily used.

To exploit the increased transfer rate of the faster memory devices, the width of the internal bus of the operational
integrated circuit could be increased. For example, for a Direct Rambus memory with a transfer rate of 16 bits at 800MHz,
the internal bus of the operational integrated circuit would have to be increased to a 128 bit bus operating at 100MHz.
As this is four times the present conventional bus width the resulting integrated circuit would be much more complex
and require increased effort in designing the layout of the interconnects within the integrated circuit and would also
consume a much larger area of silicon. This is disadvantageous. Figure 7 illustrates an example of a Direct Rambus 64
connected to an integrated circuit 61, the integrated circuit having an internal system bus 65 with a transfer rate of 128
bits at 100MHz. At the interface between the Direct Rambus memory 64 and the integrated circuit 61, a demultiplexer
70 would be required to spread the short 16 bit words from the Direct Rambus onto the 128 bit wide internal bus of the
integrated circuit. The addition of a demultiplexer 70 further increases the complexity and required silicon area of the
integrated circuit.

The speed of the internal bus of the integrated circuit could be increased to match that of the memory device
connected to it. However, this would require redesigning the integrated circuit and in practice, the internal buses of
integrated circuits which represent the current state of the art already typically operate at a speed close to the current
maximum possible speed.

It would therefore be desirable to provide an improved interface between data storage devices with a relatively
high data transfer rate and the internal bus system of an integrated circuit operating with a relatively low data transfer
which overcomes or at least mitigates against the problems described hereinbefore.

As described hereinafter with reference to Figure 2, in a conventional SDRAM, to retrieve 8 words of data,
it takes 3 clock cycles to close a previous page, 6 cycles to open the next page and retrieve the first-word of data and
a further 7 clock cycles to transfer the requested data from the memory. In a Direct Rambus memory device, the delay
necessary to close a page and open a subsequent page is 9 cycles and the time taken to transfer the remaining data
from the memory is a further 1 cycle.

Because of the delay required to close and open pages, the instantaneous data transfer from the memory
device is not constant. Figure 8 shows the variation of instantaneous data transfer from a Direct Rambus device with
respect to time. The instantaneous data transfer rate is shown by the line I and it can be seen that the memory only in
fact outputs data at a high transfer rate for short periods of time or bursts. One such period is referenced db in Figure
8. The average data transfer rate over a longer period of time is shown by line II and is much lower than the peak data
transfer rate of the memory device.

As stated hereinafter, a Direct Rambus memory device requires a total of 10 cycles to output data from the
memory which has been requested by the integrated circuit (3 cycles to close a page and 6 cycles to open a new page
and retrieve the first data word and 1 cycle to retrieve the remaining data words). As the transfer rate of a Direct Rambus
memory device is 16 bits at 800MHz, the average rate of data transfer is 320 M bytes per second. An internal bus of an
integrated circuit operating at 32 bits at 100MHz is capable of sustaining a data transfer rate of 400 M bytes per second.

Reference is now made to Figure 9 which shows a Direct Rambus 102 connected to an integrated circuit 103
via an interface 101, 106. To take advantage of the higher average data transfer rate of a conventional internal bus of
a integrated circuit in comparison to the average data transfer rate of a Direct Rambus memory device, the interface
101,106 is provided between the Direct Rambus 102 and the integrated circuit 103. This interface 101, 106 is capable
of smoothing out the peaks in the instantaneous data transfer rate of the Direct Rambus and providing an output to the
internal bus of the integrated circuit 103 which operates at least at the average data transfer rate of the Direct Rambus 102. It is preferred that the average data transfer rate of the Direct Rambus 102 be the same as the internal bus of the integrated circuit 103.

In Figure 9, the Direct Rambus memory device 102 is connected to the internal bus of integrated circuit 103 via the interface 101, 106 which consists of a buffer 101 and a controller 106. Connected between the Direct Rambus 102 and the integrated circuit 103 is the buffer 101 and the interface controller 106. The controller 106 may comprise, but not necessarily, the memory interface 22 as described in Figure 3, with buffer 101 replacing FIFO 26 and integrated circuit 103 replacing the requestor 26. The buffer 101 has an input 110 and an output 112. Connected between the input 110 of the buffer 101 and the integrated circuit 103 is a first data bus 105a which has a transfer rate equal to that of the internal bus of the integrated circuit 103, i.e. 32 bits at 100MHz. Also connected between the input 110 of the buffer 101 and the integrated circuit 103 is a first control line 118. Connected from the output 112 of the buffer 101 to the integrated circuit 103 is a second data bus 105b which has a transfer rate equal to the first data bus 105a. It will be appreciated that first and second data buses 105a and 105b in fact comprise the same data bus and are shown separately in Figure 9 merely for the sake of convenience. Also connected between the output 112 of the buffer 101 and the integrated circuit 103 is a second control line 120.

Connected between the Direct Rambus 102 and the interface controller 106 is a third data bus 104. The third data bus 104 has a data transfer rate of 128 bits at 100MHz which is equal to the peak instantaneous data transfer rate of the Direct Rambus. Also connected between the Direct Rambus 102 and the interface controller 106 is a third Direct Rambus control line 122. The interface controller 106 is connected to the buffer 101 by an interface bus 108. The interface bus 108 comprises a plurality of individual data transfer lines 108, 108, ... 108, There are n data transfer lines provided where n is the number of storage locations within buffer 101.

The operation of the circuit shown in Figure 9 will now be described. Beginning from the initial conditions of the buffer 101 being empty and the Direct Rambus 102 having all its pages closed, the integrated circuit 103 loads a memory request, MEM-REQ, into the buffer 110 via the first data bus 105a. The memory request M-REQ may be a request to access (read) data stored in the Direct Rambus memory 102 or it may be a request to write data to the Direct Rambus memory 102. If the memory request is a request to write data to the Direct Rambus memory 102, the data to be written is also loaded into the buffer 101 via the data bus 105a. Control signals for controlling the operation of the buffer 101 are also output from the integrated circuit 103 via the first control line 118.

The interface controller 106 scans the storage locations of the buffer 101 in turn, starting from the nth data storage location and when a data storage location is scanned which contains a memory request the memory request is output from the buffer 101 via the corresponding data transfer line to the interface controller 106. The interface controller 106 scans a number of data storage locations within buffer 101 and multiplexes the memory request signals and any corresponding data onto the third data bus 104 such that the memory requests are input to the Direct Rambus 102. The Direct Rambus 102 now begins the action of opening a page in the memory array in order to supply the requested data or to write the supplied data in the relevant memory location. As previously discussed, there is a delay of 6 cycles while the page is opened before any data can be output from the Direct Rambus 102. During this delay, the integrated circuit 103 may be outputting further memory requests to the buffer 101. These further memory requests are stored in the buffer 101 during the delay period which occurs while the Direct Rambus 102 is opening the page associated with the first memory request.

When the Direct Rambus 102 has opened the page associated with the first memory request, if that memory request is a request to access data from the Direct Rambus, the requested data is then transferred from the Direct Rambus 102 via the third data bus 104 to the interface controller 106. Control signals associated with controlling the input and output from the Direct Rambus 102 are passed between the Direct Rambus 102 and the interface controller 106 by the third control line 122. The interface controller 106 demultiplexes the data received from the Direct Rambus 102 and inputs it via data transfer lines 108, to empty data storage locations within buffer 101. The interface controller 106 then scans the data storage locations within the buffer 101 for the next memory requests which are to be served. The data accessed from the Direct Rambus 102 in response to a memory request received from the integrated circuit 103 via the interface controller 106 and buffer 101, is passed through the buffer 101 and output at output 112 to the integrated circuit 103 via the second data bus 105b, together with associated control signals via the second control line 120.

The buffer 101 serves two functions. Firstly, it is able to buffer the memory requests from the integrated circuit 103 to the Direct Rambus 102, allowing the integrated circuit to output a number of memory requests without having to wait for each of those requests to be served by the Direct Rambus 102 before outputing subsequent requests. Secondly, the buffer 101 is able to buffer the data supplied from the Direct Rambus 102 before it is transmitted to the integrated circuit 103. Preferably, the buffer 101 should always have space available to store the accessed data from the Direct Rambus 102 thus enabling the Direct Rambus 102 to always output data at its maximum speed of 16 bits at 800MHz.

To calculate the required size of the buffer it is assumed that the integrated circuit 103 will output ICY consecutive memory requests. The Direct Rambus 102 will see these as MemY accesses, as MemY=ICY (integrated circuit internal
bus width = Direct Rambus internal bus width). It is assumed that all of the memory requests are in the same page in the Direct Rambus 102. The Direct Rambus 102 can process MemY-1 memory requests in MemY-1 cycles. During the same time, the integrated circuit 103 can issue X memory requests, where

\[
X = \text{MemY}-1 \cdot (\text{Direct Rambus bus width} + \text{ic bus width})
\]

Approximating MemY-1 to MemY we have buffer size (in memory word units) = MemY-X

\[
= \text{MemY} \cdot (1 - (\text{icclk} \times \text{icwd}))
\]

\[
(\text{memclk} \times \text{memwd})
\]

where iclk = integrated circuit clock speed, mem clk = Direct Rambus clock speed, icwd = integrated circuit internal bus width, and memwd = Direct Rambus internal bus width.

[0057] Figure 10 demonstrates the behaviour of the buffer when the size of the buffer has been correctly chosen. Line A represents the number of memory requests stored in the buffer which are yet to be processed by the memory and line B represents the total amount of memory requests which have been processed by the memory with the associated data being stored in the buffer. The distance between lines A and B represents the total amount of data stored in the buffer. Initially line A rises sharply over the time period tsub as the buffer stores an increasing number of memory requests from the integrated circuit. This initial sharp rise occurs during the delay caused by the memory opening the required page of the first memory request. Once the required page has been opened, the memory is able to output the requested data to the buffer and begin to process the next memory request stored in the buffer. If subsequent memory requests require the same page which is currently open, a number of bursts of requested data may be output to the buffer. This line A begins to fall as the number of memory requests in the buffer falls, and line B begins to fall, representing an increase in the amount of data stored in the buffer. This is indicated on figure 6 by the period tserv.

[0058] At a subsequent point in time, point c, it will be necessary to close the currently open page and open a new page in the memory, causing the delay in outputting data from the memory as previously discussed. The memory is said, at this point, to be 'stalled'. Whilst the memory is stalled lines A and B rise again as the number of memory requests stored in the buffer once again rises and the amount of data stored in the buffer from the memory decreases. This is shown by the period tsra. This action continues over time with lines A and B rising and falling together. It can be seen that the distance between lines A and B which represents the total amount of data contained within the buffer remains approximately constant.

[0059] The circuitry described hereinbefore operates particularly advantageously when memory requests are issued by the integrated circuit 103 in short bursts. When this occurs data stored in the buffer in response to memory requests can always be output to the integrated circuit 103 at the same time that the Direct Rambus 102 is stalled during the opening of a further page in the memory. If the number of memory requests issued by the integrated circuit 103 at any one time is too large then when those requests are served by the Direct Rambus 102 the average rate of data transfer will increase to a value which is in excess of the transfer rate of the internal bus within the integrated circuit 103 and the Direct Rambus 102 will be forced to wait for previously served memory requests to be delivered to the integrated circuit before the Direct Rambus can output any further data. However, this occasional loss in performance may not be fatal to the operation of the integrated circuit, it will be more cost effective in terms of design effort and silicon area consumed to use the buffering arrangement of embodiments of the present invention.

[0060] A further example of improved interface described hereinbefore is shown in Figure 11. In this example a number of buffers 201a, 201b, 201c are connected via a demultiplexer 130 to the output of a Direct Rambus 202. Each of the buffers 201a, 201b and 201c are of the same type as the buffer 101 shown in Figure 9 and described hereinbefore. Each of the integrated circuits is of the same type as integrated circuit 103 shown in Figure 9. Each buffer is connected to the internal bus of an integrated circuit 203a, 203b, 203c, each of the internal buses having a lower peak data transfer rate than that of the Direct Rambus 202. Each of the integrated circuits may have a different function from each other and may thus make different memory requests to the Direct Rambus 202 at different times. The Direct Rambus 202 provides the data in response to these requests and outputs the data to the multiplexer 130 which is arranged to route the data to whichever of the buffers 201a, 201b or 201c issued the memory requests. In this example, by providing a number of buffers connected to the Direct Rambus 202 a higher maximum average data transfer rate from the Direct Rambus can be achieved as the data output from the memory is stored in a number of different buffers. If, as shown, the number of buffers provided is 3, then this allows 3 times the maximum average data transfer rate from the Direct Rambus before the Direct Rambus is forced to wait for the slower internal buses of the integrated circuit to retrieve the stored memory data from the relevant buffers.

[0061] It should be appreciated that in embodiments of the present invention, it is not necessary that when one memory
region is being accessed that the other memory region have a portion thereof being selected or deselected. Rather, embodiments of the present invention are particularly applicable to situations where the instructions occur in an order such that it is possible that one memory region can be accessed whilst a portion of the other memory region is being selected or deselected.

Claims

1. A memory interface device (22, 106) for generating a plurality of commands for controlling a memory having first and second memory regions (21a, 21b), only one of said memory regions being accessible at a time, each memory region comprising a plurality of rows, characterised in that said device comprises:

   - a buffer (26; 101) for storing a plurality of received memory requests for said memory, said memory requests each including information as to the row to be accessed of said respective memory region (21a, 21b), and said buffer arranged to provide a respective output (28) for each memory request, each of said outputs indicating said row to be accessed for the respective memory request;
   - a detector (30) arranged to receive said plurality of outputs (28) from said buffer (26; 101) and to detect a next different row in each of said memory regions (21a, 21b) to be subsequently selected, said detector (30) providing an output signal indicative of said detected next different row for each of said memory regions; and
   - a command provider (50) for providing a sequence of commands in response to said received memory requests and said output signals provided by said detector (30) for controlling said memory, said command sequence being arranged so that a row of one of the first and second memory regions (21a, 21b) is accessed while said detected next different row of the other of the first and second memory regions is being selected.

2. A device as claimed in claim 1, wherein said detector (30) receives at least two said outputs (28) from said buffer (26; 101), said at least two outputs corresponding to at least one of said memory regions (21a, 21b).

3. A device as claimed in claim 1 or 2, wherein said outputs (28) provided from said buffer (26; 101) are equal in number to said plurality of memory requests stored in said buffer (26; 101).

4. A device as claimed in any preceding claim, wherein when said row of the first or the second memory regions (21a, 21b) is accessed, information is read from said respective row.

5. A device as claimed in any preceding claim, wherein when said row of the first or second memory regions (21a, 21b) is accessed, the information is written into said respective row.

6. A device as claimed in any preceding claim, wherein said buffer (26; 101) comprises a first in first out register.

7. A device as claimed in claim 6, wherein said register (101) also stores data from said memory, said register comprising output means (112) for outputting request signals and said stored data from the memory to a requesting device.

8. A device as claimed in claim 7, wherein said requesting device is an operational circuit (103).

9. A device as claimed in claim 8, wherein a first data bus (105a) is provided between the operational circuit (103) and the register (101) and a second data bus (104) is provided between the memory interface device (106) and the memory, the maximum data transfer rate of the first bus (105a) being different to that of the second bus (104).

10. A device as claimed in any preceding claim, wherein a comparer (36, 38) is provided for comparing the row of a memory region which is currently selected with the row of that memory region which is next to be accessed and outputting a signal based on said comparison.

11. A device as claimed in claim 10, wherein a comparer (36, 38) is provided for each memory region.

12. A device as claimed in any preceding claim, wherein an output is provided for outputting said commands to said memory.

13. A device as claimed in any one of the preceding claims, wherein said buffer (26; 101) is arranged to receive said memory requests from a requester (24, 103).
14. A device as claimed in claim 13, wherein said requester (24, 103) is a computer processing unit.

15. A device as claimed in claim 13 or 14, wherein said buffer (26; 101) is arranged to receive said memory requests from a plurality of requesters.

16. A device as claimed in claim 15, wherein a respective register is provided for storing requests for each requester.

17. A device as claimed in any preceding claim, wherein said command provider (50) is arranged to process received requests in a nonsequential manner if a later request specifies the same memory location of a given memory region as an earlier request, with intervening requests for said given memory region being processed after said later request.

18. In combination, a device as claimed in any preceding claim and a memory comprising a plurality of memory regions.

19. A combination as claimed in claim 18, wherein said memory is a dynamic random access memory.

20. A combination as claimed in claim 18, wherein said memory comprises a synchronous dynamic random access memory.

21. A device as claimed in claim 14, 15 or 16, wherein said memory regions are arranged so that row n of the first memory region contains address x to x+y where y+1 is the number of locations in said row and row n of the second memory region contains address x+y+1 to x+2y+1.

25. Speicherschnittstellenvorrichtung (22, 106) zum Erzeugen einer Mehrzahl von Befehlen zum Steuern eines Speichers mit einem ersten und einem zweiten Speicherbereich (21a, 21b), wobei jeweils nur auf einen der Speicherbereiche zugegriffen werden kann und jeder Speicherbereich eine Mehrzahl von Reihen aufweist, dadurch gekennzeichnet, dass die Vorrichtung aufweist:

1. einen Puffer (26; 101) zum Speichern einer Mehrzahl von empfangenen Speicheranfragen für den Speicher, wobei jede der Speicheranfragen Information bezüglich der Reihe des entsprechenden Speicherbereichs (21a, 21b) enthält, auf die zugegriffen werden soll, und wobei der Puffer so eingerichtet ist, dass er eine entsprechende Ausgabe (28) für jede Speicheranfrage liefert, wobei jede der Ausgaben die Reihe anzeigt, auf die die entsprechende Speicheranfrage zugegriffen werden soll;

2. einen Detektor (30), der so eingerichtet ist, dass er die Mehrzahl von Ausgaben (28) von dem Puffer (26; 101) empfängt und eine nächste andere Reihe in jedem der Speicherbereiche (21a, 21b) detektiert, die anschließend ausgewählt werden soll, wobei der Detektor (30) ein Ausgangssignal liefert, das indikativ für die detektierte nächste andere Reihe für jeden der Speicherbereiche ist; und

3. eine BefehlslieferEinrichtung (50) zum Liefern einer Sequenz von Befehlen in Reaktion auf die empfangenen Speicheranfragen und die von dem Detektor (30) gelieferten Ausgabesignale zum Steuern des Speichers, wobei die Befehlssequenz so angeordnet ist, dass auf eine Reihe von entweder dem ersten oder dem zweiten Speicherbereich (21a, 21b) zugegriffen wird, während die detektierte nächste andere Reihe des anderen von dem ersten und dem zweiten Speicherbereich ausgewählt wird.

2. Vorrichtung nach Anspruch 1, wobei der Detektor (30) mindestens zwei Ausgaben (28) von dem Puffer (26; 101) empfängt und die mindestens zwei Ausgaben mindestens einem der Speicherbereiche (21a, 21b) entsprechen.

3. Vorrichtung nach Anspruch 1 oder 2, wobei die Anzahl der von dem Puffer (26; 101) gelieferten Ausgaben (28) gleich der Anzahl der in dem Puffer (26; 101) gespeicherten Speicheranfragen ist.

4. Vorrichtung nach einem vorhergehenden Anspruch, wobei, wenn auf die Reihe des ersten oder des zweiten Speicherbereichs (21a, 21b) zugegriffen wird, Information aus der jeweiligen Reihe gelesen wird.

5. Vorrichtung nach einem vorhergehenden Anspruch, wobei, wenn auf die Reihe des ersten oder des zweiten Speicherbereichs (21a, 21b) zugegriffen wird, die Information in die jeweilige Reihe geschrieben wird.

6. Vorrichtung nach einem vorhergehenden Anspruch, wobei der Puffer (26; 101) ein FIFO-Register aufweist.
7. Vorrichtung nach Anspruch 6, wobei das Register (101) auch Daten aus dem Speicher speichert und das Register eine Ausgabeereinrichtung (112) zum Ausgeben von Anfragesignalen und der gespeicherten Daten aus dem Speicher an eine Anfragevorrichtung aufweist.

8. Vorrichtung nach Anspruch 7, wobei die Anfragevorrichtung eine Betriebsschaltung (103) ist.

9. Vorrichtung nach Anspruch 8, wobei ein erster Datenbus (105a) zwischen der Betriebsschaltung (103) und dem Register (101) vorgesehen ist und ein zweiter Datenbus (104) zwischen der Speicherschnittstellenvorrichtung (106) und dem Speicher vorgesehen ist, wobei sich die maximale Datentransferrate des ersten Buffers (105a) von der des zweiten Buffers (104) unterscheidet.

10. Vorrichtung nach einem vorhergehenden Anspruch, wobei eine Vergleichseinrichtung (36, 38) zum Vergleichen der gegenwärtig ausgewählten Reihe eines Speicherbereichs mit der Reihe dieses Speicherbereichs, auf die als nächstes zugegriffen werden soll, und zum Ausgeben eines Signals auf der Basis dieses Vergleichs vorgesehen ist.

11. Vorrichtung nach Anspruch 10, wobei eine Vergleichseinrichtung (36, 38) für jeden Speicherbereich vorgesehen ist.

12. Vorrichtung nach einem vorhergehenden Anspruch, wobei ein Ausgang zum Ausgeben der Befehle an den Speicher vorgesehen ist.

13. Vorrichtung nach einem der vorhergehenden Ansprüche, wobei der Puffer (26; 101) so eingerichtet ist, dass er die Speicheranfragen von einer Anfrageeinrichtung (24, 103) empfängt.

14. Vorrichtung nach Anspruch 13, wobei die Anfrageeinrichtung (24, 103) eine Computerverarbeitungseinheit ist.

15. Vorrichtung nach Anspruch 13 oder 14, wobei der Puffer (26; 101) so eingerichtet ist, dass er die Speicheranfragen von einer Mehrzahl von Anfrageeinrichtungen empfängt.

16. Vorrichtung nach Anspruch 15, wobei ein entsprechendes Register zum Speichern von Anfragen für jede Anfrageeinrichtung vorgesehen ist.

17. Vorrichtung nach einem vorhergehenden Anspruch, wobei die Befehlslieferbereitstellung (50) so eingerichtet ist, dass sie empfangene Anfragen auf eine nicht sequentielle Weise verarbeitet, wenn eine später Anfrage die gleiche Speicherstelle eines gegebenen Speicherbereichs wie eine frühere Anfrage festlegt, wobei intervenierende Anfragen für den gegebenen Speicherbereich nach der späteren Anfrage verarbeitet werden.


19. Kombination nach Anspruch 18, wobei der Speicher ein dynamischer Speicher mit wahlfreiem Zugriff ist.

20. Kombination nach Anspruch 18, wobei der Speicher einen synchronen dynamischen Speicher mit wahlfreiem Zugriff aufweist.

21. Vorrichtung nach Anspruch 14, 15 oder 16, wobei die Speicherbereiche so angeordnet sind, dass Reihe n des ersten Speicherbereichs Adresse x bis x+y enthält, wobei y+1 die Anzahl von Stellen in der Reihe ist, und Reihe n des zweiten Speicherbereichs Adresse x+y+1 bis x+2y+1 enthält.

Revendications

1. Dispositif d’interface mémoire (22, 106) destiné à produire une pluralité de commandes pour commander une mémoire comportant des première et deuxième régions mémoire (21a, 21b), une seule des régions mémoire étant accessible à la fois, chaque région mémoire comprenant une pluralité de rangées, caractérisé en ce que le dispositif comprend :

   un tampon (26 ; 101) pour mémoriser une pluralité de requêtes mémoire reçues pour la mémoire, chacune des requêtes mémoire comprenant des informations quant à la rangée de la région mémoire respective (21a, 21b)
à laquelle accéder, et le tampon étant agencé pour fournir une sortie respective (28) pour chaque requête mémoire, chacune de ces sorties indiquant la rangée à laquelle accéder pour la requête mémoire respective ; un détecteur (30) agencé pour recevoir la pluralité de sorties (28) provenant du tampon (26 ; 101) et pour déterminer une rangée différente suivante dans chacune des régions mémoire (21a, 21b) à sélectionner ensuite, le détecteur (30) fournissant un signal de sortie indicatif de la rangée différente suivante détectée pour chacune des régions mémoire ; et un dispositif de fourniture de commandes (50) pour fournir une séquence de commandes en réponse aux requêtes mémoire reçues et aux signaux de sortie fournis par le détecteur (30) pour commander la mémoire, la séquence de commandes étant agencée de sorte qu’on accède à une rangée d’une des première et deuxième régions mémoire (21a, 21b) tandis que la rangée différente suivante de l’autre des première et deuxième régions mémoire est sélectionnée.

2. Dispositif selon la revendication 1, dans lequel le détecteur (30) reçoit au moins deux des sorties (28) du tampon (26 ; 101), lesdites au moins deux sorties correspondant à au moins une des régions mémoire (21a, 21b).

3. Dispositif selon la revendication 1 ou 2, dans lequel les sorties (28) fournies par le tampon (26 ; 101) sont en nombre égal à la pluralité de requêtes mémoire mémorisées dans le tampon (26 ; 101).

4. Dispositif selon l’une quelconque des revendications précédentes, dans lequel lorsqu’on accède à la rangée de la première ou de la deuxième région mémoire (21a, 21b), des informations sont lues dans la rangée respective.

5. Dispositif selon l’une quelconque des revendications précédentes, dans lequel lorsqu’on accède à la rangée de la première ou de la deuxième région mémoire (21a, 21b), les informations sont écrites dans la rangée respective.

6. Dispositif selon l’une quelconque des revendications précédentes, dans lequel le tampon (26 ; 101) comprend un registre de type premier-entré/premier-sorti.

7. Dispositif selon la revendication 6, dans lequel le registre (101) mémorise aussi des données provenant de la mémoire, le registre comprenant des moyens de sortie (112) pour fournir des signaux de requête et les données mémorisées de la mémoire à un dispositif demandeur.

8. Dispositif selon la revendication 7, dans lequel le dispositif demandeur est un circuit fonctionnel (103).

9. Dispositif selon la revendication 8, dans lequel un premier bus de données (105a) est prévu entre le circuit fonctionnel (103) et le registre (101) et un deuxième bus de données (104) est prévu entre le dispositif d’interface mémoire (106) et la mémoire, le débit de transfert de données maximum du premier bus (105a) étant différent de celui du deuxième bus (104).
17. Dispositif selon l'une quelconque des revendications précédentes, dans lequel le dispositif de fourniture de commandes (50) est agencé pour traiter des requêtes reçues d'une manière non séquentielle si une requête ultérieure spécifie le même emplacement mémoire d'une région mémoire donnée qu'une requête précédente, les requêtes intermédiaires concernant la région mémoire donnée étant traitées après la requête ultérieure.

18. En combinaison, un dispositif selon l'une quelconque des revendications précédentes et une mémoire comprenant une pluralité de régions mémoire.

19. Combinaison selon la revendication 18, dans laquelle la mémoire est une mémoire dynamique à accès aléatoire.

20. Combinaison selon la revendication 18, dans laquelle la mémoire comprend une mémoire dynamique à accès aléatoire synchrone.

21. Dispositif selon la revendication 14, 15 ou 16, dans lequel les régions mémoire sont agencées de sorte que la rangée n de la première région mémoire contient les adresses x à x+y, où y+1 est le nombre d’emplacements dans la rangée et la rangée n de la deuxième région mémoire contient les adresses x+y+1 à x+2y+1.
FIG. 5

CONVENTIONAL SDRAM 62

32 BIT @100 MHz

DDR SDRAM 63

32 BIT @200 MHz

DIRECT RAMBUS RDRAM 64

16 BIT @800 MHz

INTEGRATED CIRCUIT

32 BIT @100 MHz

FIG. 6
PRIOR ART
REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader’s convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- US 5502835 A [0007]
- US 4924375 A [0008]
- US 5615355 A [0009]