

PATENT SPECIFICATION

(11) 1 510 948

1 510 948

- (21) Application No. 34569/76 (22) Filed 19 Aug. 1976
(31) Convention Application No. 26447
(32) Filed 20 Aug. 1975 in
(33) Italy (IT)
(44) Complete Specification published 17 May 1978
(51) INT CL² G11C 11/34
(52) Index at acceptance

G4A 10EX 1C 6B 6MX 9C 9D MH

- (72) Inventors EZIO CISLAGHI
ALLESANDRO SCOTTI and
RENZO PEDERZINI



(54) REFRESHING SEMICONDUCTOR MEMORIES

(71) We, HONEYWELL INFORMATION SYSTEMS ITALIA SPA, an Italian Company of Caluso, Turin, do hereby declare the invention, for which we pray that a patent may be granted to us, and the method by which it is to be performed, to be particularly described in and by the following statement:—

The present invention relates to "refreshing" of the storage elements of semiconductor read-write memories.

In recent times semiconductor dynamic memories have become popular as the main memories in data processing apparatus. However, these memories are intrinsically "volatile": that is, the stored information fades out with the passing of time and may become completely lost. Therefore, these memories must be "refreshed" periodically and it is necessary to make sure that this refreshing operation is carried out within a defined time interval.

It has been proposed to divide a memory of this type into two parts or blocks. When an access (read or write) request is made to the memory, the relevant one of the blocks is accessed, and a refresh operation is carried out at the same time on the other block; thus the refresh is carried out during a period when the block being refreshed would otherwise be idle. Each block has a respective refresh counter, which is used to address each row in turn, and counts up to the maximum number of rows. A timer periodically causes the counts of the counters to be inspected. If the counts are at their maximum, the counters are reset and the timer starts timing again. However, if the counts are not at the maximum, a second timing circuit is energized which causes compulsory refreshing, with the refresh counters being incremented accordingly, at a relatively rapid rate until the counts are both at their maximum.

We have realized that although this known system succeeds in putting many of the refresh cycles in parallel with normal

memory accesses, it is capable of being improved.

Accordingly the present invention provides a rechargeable dynamic semiconductor memory comprising: two similar blocks each comprising a plurality of addressable rows of memory elements; read/write means for accessing a desired row in either block; for each block, a respective refresh counter and associated circuitry for refreshing each row in turn of the block, the accessing of a row in either block by the read/write means being accompanied by the refreshing of a row in the other block; a timing signal generator generating timing signals at a fixed frequency; for each block, a refreshing cycle accumulator which is incremented by the timing signals and decremented by refreshing of a row in that block; and means for generating a low priority refresh interrupt signal when the count in a refreshing cycle accumulator rises to a first predetermined value.

A memory system embodying the invention will now be described, by way of example, with reference to the accompanying drawings, in which:—

Figure 1 is a diagram of a data processing system.

Figure 2 is a simplified diagram of a communication interface among elements of a data processing system.

Figure 3 is a schematic diagram of an integrated semiconductor memory unit.

Figure 4 is a diagram of the storage part of the memory system.

Figure 5 is a diagram of the selection and addressing circuits of the memory system.

Figures 6 and 7 together are a diagram of the memory logic circuitry of the memory system.

Briefly, the present system provides a new refreshing system for a memory characterized by integrated units with a high number of rows, which makes use of refreshing cycles in parallel to normal read-

9 JUL 1981

write memory operation in such a way that the refreshing cycles are not "lost" for normal processing operations. The memory is divided in two parts or blocks, namely A and B. If the central processor unit assigns a cycle for read-write operation to one of the two blocks in parallel the same cycle is utilized to recharge or refresh a row of storage elements of the integrated units which constitute the other block of the memory. Each block has associated with it two counters which manage the recharging operations: a "refreshing counter" which delivers the address of the row of the storage elements to be recharged, and a "refreshing cycle accumulator" which memorizes the recharging state of the block as a function of the time elapsed among subsequent recharges. The refreshing cycle accumulator is incremented at a suitable fixed frequency which corresponds to a pre-established time interval, and is decremented each time that a row of storage elements is recharged or refreshed. An interrupt request with a low priority level for memory refreshing is sent to the central processor unit only when one of the two refreshing cycle accumulators reaches a pre-established count, that is when it has been incremented several times without the occurrence of refreshing cycles decrementing it. An interrupt request with a high priority level (emergency for memory refresh) is sent to the central processor unit if the refresh command corresponding due to the low priority interrupt signal was not received by the memory within a pre-established time interval: in such a case it is mandatory to perform immediately the refreshing operation by blocking all the other running activities. An interrupt request with a high priority level is also generated when a refreshing cycle accumulator reaches a count which indicates that even if the operation of memory refreshing has taken place as a result of more low priority level interrupt requests performed within time intervals each of which is lower than the one which causes the generation of a high priority level interrupt request, nevertheless a total delay has been accumulated such that a time recovery is required and therefore a refresh operation becomes mandatory.

Referring now to the drawings, Figure 1 is a schematic diagram of the organization and of the elements which constitute a data processing system, which consists of a central processor unit 1, a main memory 2 and a plurality of peripheral apparatuses 3, 4, 5, such as printers, readers, card-punch etc. The main memory 2 and the peripherals are connected to the central processor unit by means of sets of wires 6, 7, 8, 9, each of

which sets is called an interface. The information exchange among main memory, peripheral apparatus and central processor unit takes place through the interfaces.

When the peripherals want to communicate with or to get information from the central processor unit or, through it, they want to access the main memory, they forward "interrupt requests" to the central processor unit. These requests are recognized and honoured according to pre-established priority criteria with regard to the urgency degree which characterize each of them. From this point of view the semiconductor memory also acts as a peripheral. In fact each time that refreshing is required, the memory forwards an interrupt request to the central processor unit in order to obtain from this suitable refreshing instructions.

The refreshing interrupt requests generated by the memory have a pre-established hierarchic level or priority level if compared to the requests generated by the other peripherals. In particular in the refreshing system and the semiconductor memory here described, the memory can generate two different types of interrupt request: a low priority level request that can be temporarily ignored if, at the same time, other peripherals generate interrupt requests with higher priority level (or if some operation is running with the central processor unit in consequence of interrupt requests of high priority level), and a high priority level request which is mandatory and prevails over any other interrupt requests, and interrupts the execution of any operation which is running on the central processor unit at the end of the machine cycle during which the interrupt request occurred.

It is sufficient for present purposes to give an outline about the interface which connects the central processor unit with the semiconductor memory. Figure 2 shows the interface wires which connect the central processor unit with the memory. A set of 18 wires designated DATA BUS is used to transfer in bidirectional mode binary data consisting of 16 bits, that is 2 bytes plus two parity check bits (2 ck). Thus these wires are used to transfer the information to be written into the memory or to be read from the memory.

A set of 17 wires designated ADDR is used to transfer the memory addresses, coded in binary form and consisting of 16 bits plus one parity check bit, from the central processor unit to the semiconductor memory. It is therefore possible to address about 64,000 different memory positions.

A wire designated REFR is used to forward a refreshing command from central

70

75

80

85

90

95

100

105

110

115

120

125

130

processor unit to memory unit. A wire designated READ/WRITE is used to specify, according to the logic level present on it, if the operation to be carried out is a read operation or a write operation. A wire designated RESET is used to put into a known state the logic circuits, the registers, and the counters contained in the memory unit. A wire designated LP INT is used to transfer from the memory to the central processor unit an interrupt request for a low priority level refreshing. A wire designated HP INT is used to transfer from the memory to the central processor unit an interrupt request for a high priority level refreshing. A wire designated START is used to transfer from central processor unit to the memory unit a command signal synchronized with the central processor unit, to initiate a memory cycle as specified by the READ/WRITE wire (or by the REFR. wire).

The physical organization of the memory will not be described in detail. It will merely be noted that the memory consists of modules in order to obtain variable memory capacity, for example from a minimum of 8000 words, 2 bytes each, to a maximum of 64,000 words, 2 bytes each.

Figure 3 is a schematical block diagram of an integrated semi-conductor memory unit, comprising a MOS memory provided with timing, controlling and decoding circuits. The integrated unit consists of a matrix of 64×64 memory cells 10. Each memory cell is addressed by means of a row selection accomplished by a row decoding network 11, provided with 6 input terminals $A_0 \dots A_5$ and 64 outputs, and a column selection accomplished by a column decoding network 12, provided with 6 input terminals $A_6 \dots A_{11}$ and 64 outputs.

The operations performed on the matrix are defined by a logic control block 13, provided with an input terminal D_{IN} to write an information bit on the addressed memory cell, an output terminal D_{OUT} to transfer out, in complemented form, an information bit fetched from the addressed memory cell, an input terminal WE (which corresponds to the interface signal READ/WRITE) to define, according to the logic level present, if the operation is a WRITE operation or a READ operation, and an input selection terminal CS to enable or inhibit both the input terminal D_{IN} and the output terminal D_{OUT} of the integrated circuit. The activation of the integrated unit is controlled by means of a logic timing and controlling block circuit 14, provided with an input terminal CE, to which an enabling and timing signal of the unit is applied.

The main characteristic of the integrated

unit described is that for any operation performed on one memory cell of the unit, said operation causes the recharge of all the memory cells of the unit which belong to the same row. In addition, if no signal is applied to the selection terminal $\overline{CS}$, the activation of the integrated unit (by applying a signal to the input terminal CE) causes the recharge of one row of the unit, namely of the row addressed by the code present on the input terminals $A_0 \dots A_5$, without involving the input terminal D_{IN} or the output terminal D_{OUT} .

Figure 4 shows how the integrated units of Figure 3 are organized. From the functional and circuital point of view the integrated units are organized in rows of 18 units each. For a memory with a capacity equal to 64 K words there are 16 rows. The 16 rows are grouped in two blocks containing the same number of rows. Thus rows 21, 23 ... and all the odd rows constitute a memory block (A) and rows 22 ... 36, that is all the even rows constitute a memory block (B).

An address code of 6 bits on the wires (A) $R_{ADDR} A_0 \dots A_5$ addresses simultaneously the same row of memory cells in all the integrated units which constitute the block (A). Similarly an address code of 6 bits on the wires (B) $R_{ADDR} A_0 \dots A_5$ addresses simultaneously the same row of memory cells in all the integrated units which constitute the block (B). The wires (A) $R_{ADDR} A_0 \dots A_5$ and (B) $R_{ADDR} A_0 \dots A_5$ are connected to the terminals $A_0 \dots A_5$ of all the integrated units which belong respectively to block (A) and block (B). Similarly an address code of 6 bits on the wires $C_{ADDR} A_6 \dots A_{11}$ addresses simultaneously the same column of memory cells in all the integrated units which constitute the memory.

16 different wires $CS_1, CS_2 \dots CS_{16}$ for the emission of 16 selection signals, each connected to all the integrated units of a respective row, allow the selection of all the integrated units which belong to the same row. Each of the 16 wires CS_i etc. is also connected to an input terminal of a respective two input OR gate 121, 122, 123, ... 136. To the second input terminal of the OR gates 121, 123 ... 135, that is of all the odd OR gates, there is connected a wire designated REFR. A which transfers a recharge command (REFRA) coming, as we will see, from the control recharge network. To the second input terminal of the OR gates 122, 124, ... 136, that is of all the even OR gates, there is connected a wire designated REFR B which transfers a recharge command (REFR B), coming, as we will see, from the control recharge network.

(16)

4

1,510,948

4

The output terminals of the OR gates 121, 122, ... 136 are connected to one input terminal of as many AND gates 221, 222, ... 236 enabled by a common timing signal CLOCK, applied to the second input terminals of the AND gates. This signal comes from a timing network which will be considered later on. The output terminals of the AND gates are connected respectively to the input terminals of the 16 rows of the memory integrated units and they supply enabling signals to those rows.

The combined use of a selection or addressing code with 12 bits (for the selection of rows and columns), an enabling and timing signal on the wires CE₁ etc, and a selection signal on the wire CS₁ etc allows the addressing of a word of 18 bits, each bit being contained in a well defined memory cell of the integrated units which constitute the selected row. The operation performed on the memory is defined by a read-write command applied to the wire WE which is connected to the terminal WE of all the integrated units which constitute either the block (A) or the block (B). Data to be written into the memory, in a binary code of 18 bits, is applied to the wires D_{IN} (0 ... 17), each of which is connected to all the terminals D_{IN} of the integrated units of the same order belonging to different rows. Similarly the output terminals of each memory unit $\overline{D}_{OUT}$ are grouped for the two blocks (A) and (B) in the collector channel of the 18 wires $\overline{D}_{OUT}$ (0 ... 17). From the collector channel $\overline{D}_{OUT}$ (0 ... 17), the information read out is temporarily memorized in the output register 40, to be then transferred on the interface channel DATA BUS, via a group of control gates 41. In addition, due to the presence of REFR A or REFR B command, all the integrated units of a block are enabled for the execution of a recharge operation.

Figure 5 shows the memory selection circuits for the execution of a read-write and/or memory recharge operations. The START signal, coming from the central processor unit through the interface each time that it is necessary to initiate the memory cycle, activates the timing circuit 45, which may suitably consist of a delay line with intermediate taps T₁ ... T_N, suitably adjusted, in such a way to generate a sequence of timing pulses on the taps. For reason of simplicity, the timing pulses (when not indicated otherwise) will be designated by the same name as the wires which transfer them. With reference to Figure 4, the CLOCK pulse has already been mentioned. The different timing signals are transferred in different locations of the memory where they command the enabling of determined AND gates.

In addition the memory receives from the central processor unit the 17 wires ADDR which carry in coded form the memory addresses. The signals present on these wires are first of all verified by a verifying network and then used for the addressing. Six wires, of the 16 wires really used for the addressing code, are distributed to all the integrated units and they constitute the column addressing wires C_{ADDR} A₆ ... A₁₁, as already seen in Figure 4. Another 6 wires designated R_{ADDR} (A₀ ... A₅) are connected to the input terminals designated "0" of two integrated circuit multiplexors 47 A and 47 B. These multiplexor circuits have a plurality of input groups, an output group and one or more control inputs. Depending on the logic level present at the control inputs, one of the input groups is connected to the outputs so that the signals present on the selected input group are transferred to the outputs. The multiplexor circuits 47 A and 47 B have two input groups designated "0" and "1" and one control input 43 A or 43 B. When the logic level present at the control input is "0", the inputs designated "0" are selected and the signals present thereof are transferred on the outputs. When the logic level present at the control input is "1" the inputs designated "1" are selected and the signals present thereof are transferred on the outputs.

The outputs of the multiplexors 47 A and 47 B designated (A) R_{ADDR} A₀ ... A₅ and (B) R_{ADDR} A₀ ... A₅ provide, as already seen in Figure 4, the address of a well defined row of cells in all the integrated units of each block (A) and (B). Of the 16 addressing wires ADDR, the three wires A₁₂, A₁₃, and A₁₄ are forwarded to two decoding networks 38 A and 38 B, which decode the 3 bit binary code present on those wires, and generate a corresponding signal on only one of the 8 output wires, which are designated CS 1, 3, ... 15 (odd numbers) for decoding network 38 A and CS 2, 4, ... 16 (even numbers) for decoding network 38 B. As already seen in Figure 3 and 4, a signal present on one of these wires selects a row of the memory integrated units and enables the input terminal D_{IN} and the output terminal $\overline{D}_{OUT}$ of those units.

Finally the addressing wire A 15, depending on whether the signal present on it is at logic level "0" or "1", enables multiplexor circuit 47 A or 47 B and the decoding circuit 38 A or 38 B. More precisely, the wire A 15 is connected directly to the enabling input 49 B of the decoding circuit 38 B and in addition it is connected to the selection input 43 A of the multiplexor circuit 47 A through the Z-input OR gate 40 A. Further, it is connected to the enabling input 49 A of the decoding circuit 38 A through the inverter

circuit 41 and, still through the inverter circuit 41 and the OR gate 40 B, is connected to the selection unit 43 B of the multiplexor circuit 47 B. The second input terminals of the two OR circuits 40 A and 40 B are also connected to the wire 45 on which is applied the recharging command REFR coming from the central processor unit.

To complete the description of Figure 5, it is noted that the two input groups designated "1" in the multiplexor circuits 47 A and 47 B are connected to the output terminals of the counters 46 A and 46 B respectively. The counters 46 A and 46 B, consisting of six stages, are able to count from 0 up to 63 and they provide at their output terminals a 6 bit row address which is used to address a row of each one of the memory integrated units, in the block (A) or in the block (B) as an alternative to the $R_{ADDR} (A_0 \dots A_5)$ code received through the interface. The counters 46 A and 46 B are of the cyclic type, that is, once they reach the count of 63, they restart to count from 0; their count is increased by 1 each time they receive a command pulse INCR A and INCR B at the input terminals 48 A and 48 B respectively. Said command pulses are generated by the recharging control network which will be shown later.

The memory selection circuits forward recharging signals to the control network through the wires 147 A and 147 B. These wires are derived respectively from the OR gates 40 A and 40 B. Thus on these wires there is present a signal at logic level "1".

a) on both wires when the central processor unit emits a REFRESH command.

b) on wire 147 A when the central processor unit emits a memory selection code relating to the block B and such that the logic level present on the wire A 15 is "1".

c) on wire 147 B when the central processor unit emits a memory selection code relating to the block A and such that the logic level present on the wire A 15 is "0".

Figures 6 and 7 show jointly the memory logic that provides to the control of the recharging operations. In particular Figure 6 shows that part of the logic which can be defined as internal memory clock. In fact it has to be remarked that, to prevent the loss of information, each integrated memory unit has to be recharged completely within a pre-established time interval, for instance 2 ms. For safety reasons the average recharge time assume is much smaller, for instance 1 ms. Since the recharge takes place row by row it is necessary in a 64 row integrated memory unit for the recharge to be performed each 16 μ s beginning from

the instant at which the memory starts to operate, independently of the timing and from the operations carried out by the central processor unit.

An internal clock takes care of the passed time and of the consequent necessity of recharging the memory. This clock is activated by power supplying the memory and it runs in full autonomous way with respect to the central processor unit operation.

Referring now to Figure 6, an oscillator 50 generates a 5 MHz signal and feeds a waveshaper and band suppressor 51, which in turn feeds a binary counter of two 4-bit integrated counter circuits 52 and 53. The output of the pulse shaper 51 is connected to the input 54 of the counter 52 and it triggers the counter with a sequence of pulses each having a pulse duration of 100 ns and with a period of 200 ns. The counter 52 is initially reset by a RESET signal (shown in Figure 2) and then it counts at each pulse and presents, at its output terminals 55, 56, 57, and 58 a count which goes from 0000 to 1111. After the first 15 pulses, at the occurrence of the 16th pulse, that is after 3.2 μ s from the beginning of operation, the counter goes back to the count 0000 and it starts to count again. At the same time a carry pulse is transferred from the carry output CARRY to the input 59 of the counter 53, which is similar to the counter 52. The counter 53 (also initially in the reset state) counts due to said pulse and to subsequent carry pulses, so that its outputs 60, 61, 62, and 63 have a count that advances starting from 0000.

The outputs 60 and 62 are connected to the inputs of an AND circuit 67 whose output is connected to a reset input of the counter 53. Therefore when at the outputs 60, 61, 62, 63 the count reaches 0101 (corresponding to 5), the counter 53 is reset. The counter 53 thus operates as a cyclic counter with period equal to 5. The outputs 55, 56, 57, 58, and 62 together with the output of the pulse shaper 51 are connected directly to the inputs of an AND gate 64, the output of which goes to logic level "1" each time that at its inputs is present the configuration "all one", that is every 16 μ s (and after 15.2 μ s from the starting instant). The pulse duration at the output of the AND gate 64 is obviously 100 μ s. This pulse is designated CLUP0 (CLOCK-UP), since, as we will see, it must activate the forward counting of two counters, which indicate the recharging state, being contained in the recharge logic.

The outputs 55, 56, 58, 60, and 61 and the output of the pulse shaper 51 are also connected directly to an AND gate 65, and the output 57 is connected to the AND gate

65 through an inverter circuit. The output of the AND gate 65 goes therefore to a logic level "1", with a duration of 100 ns, 4 μ s in advance of the output pulses of the AND gate 64. The output pulses of the AND gate 65 are designated CLOAP and they are used, as we will see, for the timing of the generation of the high priority level interrupt request for a memory refresh.

A group of flip-flops 66, 67, and 68 is used for suitably synchronizing with the internal memory clock signals asynchronous with respect to a clock. The START command (interface signal, Figure 2) is asynchronous with respect to the internal memory clock and depends only on the central processor unit timing: it appears each time that the central processor unit is willing to initiate a read-write operation or a recharge operation. Also the signals on the wires 147 A and 147 B (Figure 5) which represent recharge commands relative to block A or block B of the memory are asynchronous following addressing or REFRESH signals.

The START signal is applied to the set input of the flip-flop latch 66 putting it in the set state. The direct output 69 of the flip-flop 66 is connected to the enabling inputs of the two AND gates 70 and 71, the second inputs of which are connected to the wires 147 A and 147 B respectively. On the occurrence of a REFRESH command or a read-write command, either or both the wires convey a signal at logic level "1" which is transferred to the output. The outputs of the gate circuits 70 and 71 are connected respectively to the J and K inputs of the two J-K flip-flops 67 and 68. The clock inputs 71 A and 72 of these flip-flops are connected to the output of the pulse shaper 51. As soon as a pulse is received on the clock input, if the output of the AND gate 70 is at logic level "1", the flip-flop 67 is put in the set state; in the opposite case it is maintained in or reset to the reset state. Similarly, as soon as a pulse is received on the clock input, if the output of the AND gate 71 is at logic level "1", the flip-flop 68 is put on the set state; in the opposite case it is maintained in or reset to the reset state.

The direct outputs of the flip-flops 67 and 68 are connected to one input of each of the AND circuits 73 and 74, whose second inputs are fed with output 55 of the counter 52 and whose third inputs are fed with the output of the pulse shaper 51 through an inverter circuit 75. It is clear that the outputs of the AND circuits 73 and 74 go respectively to logic level "1" only if the flip-flops 67 and 68 respectively are in the set state, the output of the pulse shaper 51 is at logic level "0", and the output 55 at logic level "1", that is for a duration of 100 ns. The output pulses of the AND gates 73 and

74 are therefore out of phase and not superimposed with the output pulses of the AND gates 64 and 65. The output pulses of the AND circuits 73 and 74 are named respectively CLDA and CLDB (CLOCK DOWN A and CLOCK DOWN B), since, as we will see, their function is to activate the decrement of two counters which indicate the recharge state and are contained in the recharge logic.

The outputs of the AND circuits 73 and 74 are connected to the inputs of an OR gate 76, the output of which is connected to the reset input of flip-flop 66. Therefore as soon as a pulse CLDA or CLDB is emitted, the flip-flop 66 is put in the reset state and also the flip-flop 67 and 68 are put in the reset state by the next clock pulse of the pulse shaper 51, and the generation of additional pulses CLDA or CLDB is inhibited until to the occurrence of new START pulse.

Figure 7 shows the recharge logic circuitry, which consists mainly of two counters or recharge buffers 80 and 81, the function of which is to memorize the recharge state of the two memory blocks A and B. These recharge buffers each consist of two reversible binary counters of four stages.

The counter 80 has an incrementing input 82, a decrementing input 83, and suitable preset inputs and a preset command not shown. It has four outputs with increasing binary weights, 84, 85, 86, and 87. The CLDA pulses coming from the internal clock (Figure 6) through the four input AND circuit 88 are applied to the decrementing input 83. In order to transfer the CLDA pulses to the input 83 the following conditions must be verified:

- 1) the counter 80 must not be in the configuration 0000—Such a condition is detected by connecting the outputs 84, 85, 86, and 87 to an OR circuit 89 and by connecting the output of that OR gate to an input of the AND circuit 88.
- 2) When an interrupt request for the refreshing operation is running, recharge operations on a memory block, if any, following read-write operations on the other block, must not modify the state of the counter; that is, they have to be ignored. In other words the criteria is that an interrupt request must be specifically satisfied by the central processor unit with a refreshing command. This condition is satisfied by connecting, to one input of the AND gate 88, the complemented output of a flip-flop 90 used, as we will see, to generate the low priority level signal LPINT and its complemented $\overline{\text{LPINT}}$. In this way,

when the flip-flop 90 is in the set state, the AND circuit 88 is inhibited.

- 3) The CLDA pulses, following recharge operations caused by recharge interrupts, must not modify the counter state and must be ignored. This condition is satisfied by connecting, to one input of the AND circuit 88, the complemented output of the flip-flop 91 used, as we will see, to generate an inhibiting signal BLCD that originates with an interrupt request and keeps up to the falling of the CLDA pulse following the refreshing operation caused by said interrupt request.

On the other hand the pulses CLUPO coming from the internal clock (Figure 6) are applied to the incrementing input 82 through the OR circuit 92 and the AND circuit 93. In order to transfer the CLUPO pulses through the and circuit 93, the condition that the output 86 or the output 87 are not at logic level "1" must be checked. In other words the counter 80 can be incremented up to reach the output logic configuration 0100, but when this configuration or a following one is reached, the CLUPO pulses do not have any more effect. To ensure this, the outputs of the counter with higher weight 86 and 87 are connected to the two inputs of the NOR circuit 94, the output of which is connected to the enabling input of the AND circuit 93.

Further incrementing of the counter are made possible by the one-shot 95, the output of which is connected to a second input of the OR circuit 92. This one-shot is activated by the falling edge of the signal PRIN generated by a flip-flop 96, the function of which will be seen later on.

The structure of the counter 81 and of the circuits associated with it is quite similar, and will not be described in detail.

There will now be described the part of the refresh logic circuitry which is intended for the generation of the refresh interrupt signals. The two outputs with higher weight of the two counters 80 and 81, that is the outputs 86, 87 and 102, 103 are connected to the input of the OR circuit 111, whose output is connected to the inputs J and K of the flip-flop J-K 90 already mentioned. The pulses CLUPO are applied to the clock input C of this flip-flop. Therefore at the fall of the pulses CLUPO, the flip-flop 90 is set or reset depending on whether the output of the OR gate 111 is at logic level "1" or "0". The flip-flop 90 is also fed with a direct reset input R, connected to the output of the AND circuit 112, to the inputs of which the signals REFR (Refresh Command) and START are applied. Therefore when the central processor unit

commands a refreshing operation, the flip-flop 90 is reset.

The direct output D of the flip-flop 90 sends, on the wire 113 and through the memory interface, a low priority level interrupt signal LPINT to the central processor unit, in order to request a recharge operation. In addition it takes care, through a direct set input, of setting the flip-flop 91. The complemented output of flip-flop 90, when it is at logic level "0" inhibits the AND circuits 88 and 98. The direct output D of the flip-flop 90 is also connected to the inputs J and K of the J-K flip-flop 96, to the clock input C of which the pulses CLUPO are applied. Therefore at the fall of the pulses CLUPO, the flip-flop 96 is set or reset depending on the state of the flip-flop 90. The flip-flop 96 is provided with a direct reset input to which is connected the output of the AND circuit 104, which is fed with the pulses CLDA and CLDB. Therefore when a refreshing operation is commanded by the central processor unit and consequently the pulses CLDA and CLDB are generated (with a certain delay), the flip-flop 96 is reset.

The direct output of the flip-flop 96 sends to the OR circuit 114 a signal PRIN for preparing the high priority level interrupt request. The complemented output I of the flip-flop 96 sends to the clock input of the one-shot 95 the signal PRIN. As soon as the flip-flop 96 is set, the falling edge of the signal PRIN activates the one-shot 95. The OR circuit 114 receives at its input, in addition to the signal PRIN from the flip-flop 96, the signals present at the outputs with higher weight 87 and 103 of the two counters 80 and 81. Therefore the output of the OR circuit 114 goes to logic level "1", when any one of the following conditions is satisfied:

- 1) the flip-flop 96 is in the set state;
- 2) the counter 80 reaches the configuration 1000;
- 3) the counter 81 reaches the configuration 1000.

The output of the OR circuit 114 is connected to the inputs J and K of a J-K flip-flop 115, which is activated through the clock input C by the pulse CLOAP received from the internal clock. Therefore if the output of the OR gate 114 is at logic level "1" at the falling of the pulse CLOAP, the flip-flop 115 is set and it generates a high priority level interrupt signal HPINT which is transferred, through the interface, to the central processor unit. Flip-flop 115 is provided with a direct reset input to which is connected the output of the AND circuit 112. Therefore the flip-flop 115 goes to the reset state when both the REFR command

and the START pulse are simultaneously present.

Taking again into consideration the flip-flop 91, it can be remarked that its clock input C is connected to the output of the AND circuit 104, while the input J and K are always connected to the logic level "0". Therefore with the falling of the pulses CLDA and CLBD applied to the inputs of the AND circuit 104, the flip-flop 91 goes into the reset state.

The recharge logic is completed by some auxiliary circuits: the output of the OR circuit 89 is connected (besides the AND gate 88) to the inputs J and K of a J-K flip-flop 116, which is set, through the clock input C, by a timing pulse START emitted by the timing network (Figure 5) and is reset by a timing pulse T3 (emitted by the network 45). The direct output of this flip-flop enables, for a suitable time interval, the AND circuit 117, an input of which is connected to the wire 147 A (Figure 5). The output of the AND circuit 117 consequently sends a refresh command REFRA to the block A only if the counter 80 is not in the reset state and can be decremented. Such a recharge command is shown in Figure 4. The output of the AND circuit 117 is also connected to the inverting input of a one-shot 118. Therefore the falling edge of the REFRA signal (which occurs at the end of the recharge operation on the block A) activates the one-shot 118 that generates at its output a command pulse INCRA. This pulse is applied, as already seen, to the input of the counter 64 A and it increments its contents by 1, prearranging it to the recharge address to be used in the next recharge operation.

Similarly the output of the circuit OR 99 is connected to the inputs J and K of the flip-flop 119, which feeds an AND gate 120 and a one-shot 121 to generate the signals REFRB and INCRB.

Now that the structure of the memory and of the control logic related to the refreshing operations has been described, its operation will now be considered. Initially, when power is applied to the central processor unit and the memory, all the registers, the flip-flop and the counters are reset and the oscillator 50 begins to run. Supposing that no read-write or refreshing operation takes place, the behaviour of the logic circuitry is as follows.

The counters 52 and 53 count up. After 12 μ s from the start, the pulse CLOAP is generated. This pulse does not modify the state of the flip-flop 115 since the output of the OR circuit 114 is at logic level "0". After another 4 μ s (that is after 16 μ s from the start) the pulse CLUPO is generated. Since the counter 80 was in the

configuration 0000, it is incremented by one unit and it goes in the 0001 configuration. The same happens for the counter 81. On the contrary the flip-flop 90 remains in the reset state, since the output of the OR circuit 111 was at logic level "0", and the flip-flop 96 remains in reset state too, since the flip-flop 90 was and remains in the reset state.

After a further 12 μ s (that is 28 μ s from the start) a new pulse CLOAP is generated which, as previously, does not have any effect. Another 4 μ s later (that is 32 μ s from start) a new pulse CLUPO is generated that increments by another 1 the content of the counters 80 and 81, bringing them to the 0010 configuration. Taking as a "cycle" the interval of 16 μ s existing between two successive CLUPO pulses, it is clear that after 4 cycles, the counters 80 and 81 go in the 0100 configuration. At this point additional increments of the two counters are inhibited, but the output of the OR circuit 111 goes to logic level "1". At the end of the fourth cycle, the falling of the pulse CLUPO, applied to the flip-flop 90, puts it in set state. Therefore a low priority level interrupt request LP.INT is generated and the decrementing input of the two counters 80 and 81 is inhibited. At the same time the flip-flop 91 is set. The pulses CLOAP still do not have any effect.

Supposing that the interrupt request is not honoured by the central processor unit within the occurrence of the subsequent CLUPO pulse, that is within the end of the fifth cycle, the flip-flop 90 is kept in the set state and the flip-flop 96 is put into the set state. Therefore the signal PRIN is generated and the signal PRIM disappears.

The falling of $\overline{\text{PRIM}}$ activates the one-shot 95 and causes an incrementing of the two counters 80 and 81 which go into the 0101 configuration. Supposing that also subsequently the interrupt request is not honoured within 12 μ s, the occurrence of the pulse CLOAP sets the flip-flop 115 and generates the high priority level interrupt signal HP INT. The central processor unit honours such an interrupt request almost immediately, that is in the machine cycle subsequent to that in which the interrupt request has occurred (or within a predetermined number of cycles, if it is designed that way). Within a reduced time interval and in any case less than 4 μ s, a REFR command and a START command are sent to the memory. These signals cause the reset of flip-flops 115 and 90, therefore the interrupt requests fall to logic level "0". In addition through the addressing logic (Figure 5) and the auxiliary circuits shown in Figure 7, they generate the commands REFR A and REFR B which cause the

refreshing operation. By acting on the internal clock (Figure 6) they cause the generation of CLDA and CLDB. However these pulses do not cause a decrement of the counters 80 and 81, since the two AND circuits 88 and 98 are inhibited by the absence of the signal BLCD. Only with the falling of the pulses CLDA and CLDB and with the consequent reset of the flip-flop 91 are the two AND gates 88 and 98 again enabled. The pulses CLDA and CLDB also reset the flip-flop 96.

At the end of the refreshing operation, which is completed in a machine cycle, that is within $2 \mu s$, a pulse T3 from the timing network 45 (Figure 5) resets the flip-flops 116 and 119 with the consequent activation of the one-shots 118 and 121 and the generation of the pulses INCR A and INCR B. These pulses increment by 1 the addressing counters 46 A and 46 B devoted to the refreshing operation, setting them for a new refreshing operation. The two counters 80 and 81 remain in the 0101 configuration. Therefore at the end of the sixth cycle, the pulse CLUPO which occurs does not modify the content of the two counters 80 and 81 (the AND gates 93 and 107 are inhibited by the output configuration of the counters), but sets again the flip-flop 90 generating a new low priority level interrupt request LPINT. If the request is not honoured within the seventh cycle, at the end of this cycle the pulse CLUPO of that cycle sets the flip-flop 96 and through this activates the one-shot 95 in such a way that the two counters 80 and 81 are incremented by 1 and assume the configuration 0110. Supposing that also in the first $12 \mu s$ of the eighth cycle the low priority interrupt request is not honoured, the pulse CLOAP sets the flip-flop 115 and generates a high priority level interrupt request HPINT. This interrupt is necessarily honoured but, as already seen, the refreshing operation which follows does not modify the content of the counters 80 and 81.

Therefore at the end of the sixth cycle, the pulse CLUPO which occurs does not modify the content of the two counters 80 and 81 (the AND gates 93 and 107 are inhibited by the output configuration of the counters), but sets again the flip-flop 90 generating a new low priority level interrupt request LPINT. If the request is not honoured within the seventh cycle, at the end of this cycle the pulse CLUPO which occurs sets the flip-flop 96 and through this activates the one-shot 95 in such a way that the two counters 80 and 81 are incremented by 1 and assume the configuration 0110.

Supposing that also in the first $12 \mu s$ of the eighth cycle the low priority interrupt

request is not honoured, the pulse CLOAP sets the flip-flop 115 and generates a high priority level interrupt request HPINT. This interrupt is necessarily honoured but, as already seen, the refreshing operation which follows does not modify the content of the counters 80 and 81. Therefore at the end of the eighth cycle a new low priority level interrupt request is generated. If also this request is not honoured within the ninth cycle, the new CLUPO pulse which occurs at the end of the ninth cycle causes the set of the flip-flop 96 and the activation of the one-shot 95 and therefore the incrementing of the counters 80 and 81 which assume the configuration 0111. Again, if the interrupt request is not honoured in the subsequent $12 \mu s$ the pulse CLOAP sets the flip-flop 115 and a high priority level interrupt request is generated. This request is necessarily honoured within the tenth cycle at the end of which a new low priority level interrupt request is generated. If the request is not honoured within the eleventh cycle, at the end of this cycle the pulse CLUPO sets the flip-flop 96, activates the one-shot 95, puts the flip-flop 115 in the set state and increments the content of the counters 80 and 81, which assume the configuration 1000. If the request is not honoured in the first $12 \mu s$ of the twelfth cycle, a high priority level interrupt request is generated which is necessarily honoured within the twelfth cycle and does not modify the content of the counters 80 and 81. At the end of this cycle a new low priority level interrupt request is generated.

However at this point the content of the counters is 1000. Therefore if this request is not honoured within the first $12 \mu s$ of the thirteenth cycle, the pulse CLOAP sets the flip-flop 115 since the output of OR circuit 114 is at logic level 1. Therefore a high priority level interrupt request is generated that is necessarily honoured within the end of the thirteenth cycle and does not modify the content of the counter. Again at the end of the thirteenth cycle a low priority level interrupt request is generated and if this is not honoured in the subsequent $12 \mu s$, a high priority level interrupt request is generated. In fact the content of the counter has not been modified. Taking into consideration the number of cycles past and the number of refreshing operations carried out, this represents a worst case, since the central processor units honours only the high priority level interrupt requests.

The operation of refreshing control network can be summarized as follows:

- 1) The control network does not generate any refreshing request if the delay accumulated in the refreshing is smaller than 4 cycles.

16

10

1,510,948

10

2) It generates a low priority level interrupt signal when the delay accumulated reaches or exceeds the 4 cycles.

5 3) It generates a high priority level interrupt signal if the low priority interrupt requests are not honoured within 28 μ s.

10 4) In any case it generates a high priority interrupt signal when the delay accumulated reaches the 8 cycles (plus 12 μ s which represents the delay of the pulse CLOAP with respect to the cycle end).

15 However it is clear that for any configuration assumed by the counters 80 and 81 in the time intervals in which there is no refreshing interrupt request pending (both low and high priority), a read-write operation on a memory block causes a refreshing operation on the other block and consequently a decrement of the content of the counter relating to that block. Therefore it follows that if in a time interval of 64 μ s at least four read-write operations occur on the block (A) and four read-write operations on the block (B), the refreshing operations which result therefrom maintain the counter configuration in a state that does not give rise to refreshing interrupt requests.

20 For a data processing system with a machine cycle of the order of 1 μ s, the probability that read-write operations will occur with a frequency higher than that indicated is so high that in reality the refreshing operations are superimposed on the read-write operations and only exceptionally will interrupt requests occur, for instance when the central processor unit is busy with the execution of very arduous internal computations.

30 It has also to be remarked that while in the previous description it was supposed that the counters 80 and 81 had the same counts, in reality the two counters could assume different counts. In such a case, the counter which controls the generation of the interrupt requests is that one with the greater count.

35 In order to obtain the maximum efficiency of the system, the two memory blocks include respectively all the memory positions having even addresses and all the memory positions having odd addresses. In this way since, in fetching the information from the memory or by writing the information into the memory, the order followed is the sequential one, on average, the refreshing operations of the block A are equal to those of block B in the short term period.

40 It is interesting to consider the case where the two counters 80 and 81 are set to the configuration 0011 at the time in which

the memory starts to operate and through the use of the present input terminals. In such a case if during the first cycle no refreshing operation takes place, at the end of said cycle a low priority level interrupt signal is generated. On the contrary, if during the first cycle some refreshing operation is carried out, the configuration 0011 is decremented accordingly. In other words it is possible to say that the counters 80 and 81 contain the most recent history of the refreshing operations and they give a correlation between the number of refreshing operations carried out and the number of part cycles. If the number of refreshing operations is higher than the number of part cycles, the counters memorize the excess up to a maximum of 4 and inhibit the generation of recharge signals until the excess is absorbed. When this excess is absorbed, each 16 μ s low priority level interrupt requests are generated, and if these are not honoured within a pre-established time interval (28 μ s), a high priority level interrupt request is generated.

70 Similarly, the counters memorize also the failing number or deficiency of refresh operations with respect to the number of past cycles. Supposing that, on average, a recharge operation must occur in correspondence to each past cycle, if the failing number is 1 a low priority level interrupt is generated and this is followed by a high priority level interrupt if the low priority level request is not honoured within a pre-established time interval. If the low priority level interrupt is honoured within the due time, this however involves the loss of time of one refresh operation, which loss is not recovered by the fact that said interrupt has been honoured. Therefore if recharge operations as a consequence of low priority level interrupt signals occur, for each interrupt a one unit shortcoming relative to the number of recharge operations with respect to the past cycles is accumulated. When this shortcoming reaches four units, a high priority level interrupt signal is generated.

75 Many variations of the arrangement are of course possible. For example, instead of inhibiting the refreshing operations following read-write operations which could be carried out when interrupt requests are pending, it is possible to perform these operations and to cancel the interrupt requests emitted, if the central processor unit allows this kind of procedure. Further it is possible to manage independently the refreshing operations of the two blocks, following an interrupt request, for instance by splitting some circuit element such as the flip-flops 91, 96 and the one-shot 95.

70

75

80

85

90

95

100

105

110

115

120

125

130

WHAT WE CLAIM IS:—

1. A rechargeable dynamic semiconductor memory comprising:
two similar blocks each comprising a plurality of addressable rows of memory elements;
5 read/write means for accessing a desired row in either block;
for each block a respective refresh counter and associated circuitry for refreshing each row in turn of the block, the accessing of a row in either block by the read/write means being accompanied by the refreshing of a row in the other block;
10 a timing signal generator generating timing signals at a fixed frequency;
for each block, a refreshing cycle accumulator which is incremented by the timing signals and decremented by refreshing of a row in that block; and
20 means for generating a low priority refresh interrupt signal when the count in a

refreshing cycle accumulator rises to a first predetermined value.

2. A memory according to Claim 1 wherein a high priority refresh interrupt signal is generated when the count in a refreshing cycle accumulator reaches a second predetermined value greater than the first. 25 30

3. A memory according to either previous Claim wherein a high priority refresh interrupt signal is generated if a low priority refresh interrupt signal is not honoured within a predetermined period. 35

4. A rechargeable dynamic semiconductor memory substantially as herein described with reference to Figures 3 to 7 of the accompanying drawings.

M. G. HARMAN.
Agent for the Applicant.

Printed for Her Majesty's Stationery Office, by the Courier Press, Leamington Spa, 1978
Published by The Patent Office, 25 Southampton Buildings, London, WC2A 1AY, from which copies may be obtained.



* C L M S *