PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 5 :		(11) International Publication Number:
G06F 11/20, 12/06	A1	WO 92/17842
		(43) International Publication Date: 15 October 1992 (15.10.92)

(21) International Application Number: PCT/GB92/00608

(22) International Filing Date: 3 April 1992 (03.04.92)

(30) Priority data:
1094/91 3 April 1991 (03.04.91) IE

(71) Applicant *(for all designated States except US)*: TOLSYS LIMITED [IE/IE]; Innovation Centre, Trinity College, College Green, Dublin 2 (IE).

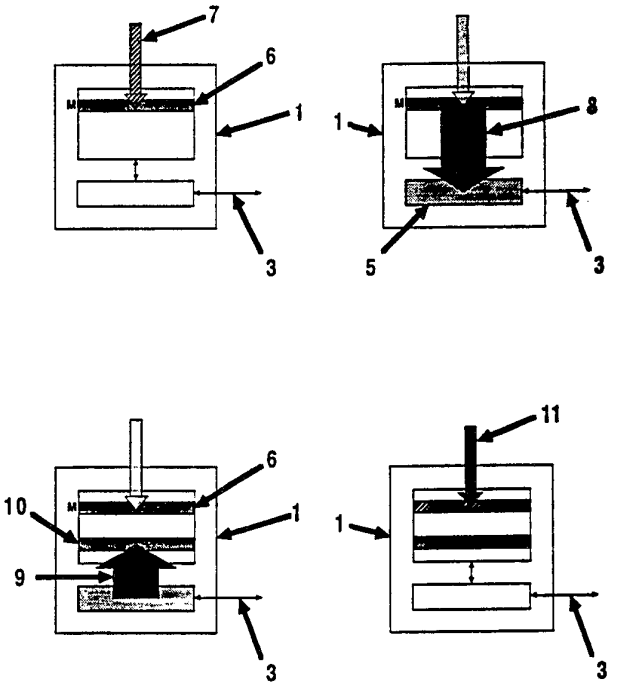
(71)(72) Applicants and Inventors: COGHLAN, Brian, Arthur [AU/IE]; 32 Millview Court, Malahide, Dublin (IE). JONES, Jeremy, Owen [GB/IE]; 303 Clontarf Road, Clontarf, Dublin 3 (IE).

(74) Agent: WALDREN, Robin, Michael; Marks & Clerk, 57-60 Lincoln's Inn Fields, London WC2A 3LS (GB).

(81) Designated States: AT (European patent), AU, BE (European patent), CA, CH (European patent), DE (European patent), DK (European patent), ES (European patent), FR (European patent), GB (European patent), GR (European patent), IT (European patent), JP, LU (European patent), MC (European patent), NL (European patent), SE (European patent), US.

Published
With international search report.

(54) Title: STABLE MEMORY OPERATIONS



(57) Abstract

The invention provides a method of operation of a stable memory circuit having two stable memory banks (bank 0, bank 1). Each bank may comprise one or more VRAMs (1). A stable memory operation involves writing from a source location (6) in a VRAM to a temporary location (5) in the same VRAM and subsequently writing the data from the temporary location (5) to a destination location (10) in the same VRAM. This allows stable memory operations to be carried out very quickly because communication between memory devices is not required.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	**ES**	Spain	**MG**	Madagascar
AU	Australia	**FI**	Finland	**ML**	Mali
BB	Barbados	**FR**	France	**MN**	Mongolia
BE	Belgium	**GA**	Gabon	**MR**	Mauritania
BF	Burkina Faso	**GB**	United Kingdom	**MW**	Malawi
BG	Bulgaria	**GN**	Guinea	**NL**	Netherlands
BJ	Benin	**GR**	Greece	**NO**	Norway
BR	Brazil	**HU**	Hungary	**PL**	Poland
CA	Canada	**IT**	Italy	**RO**	Romania
CF	Central African Republic	**JP**	Japan	**RU**	Russian Federation
CG	Congo	**KP**	Democratic People's Republic of Korea	**SD**	Sudan
CH	Switzerland			**SE**	Sweden
CI	Côte d'Ivoire	**KR**	Republic of Korea	**SN**	Senegal
CM	Cameroon	**LI**	Liechtenstein	**SU**	Soviet Union
CS	Czechoslovakia	**LK**	Sri Lanka	**TD**	Chad
DE	Germany	**LU**	Luxembourg	**TG**	Togo
DK	Denmark	**MC**	Monaco	**US**	United States of America

"Stable Memory Operations"

The invention relates to stable memory operations carried out in a stable memory circuit comprising at least two memory banks which may be interconnected by a stable data path. European Patent Specification No. EP 0418030A2 (TCD, Coghlan, Jones) describes such a circuit.

An object of the invention is to provide stable memory operations which may be carried out much more quickly than heretofore. Another object is to minimise the delay to host processor operations. A still further object is to provide deterministic stable memory operations, (i.e. the duration of which may be predicted in advance), and in which the duration is independent of memory size.

According to the invention, there is provided a stable memory operation carried out in a stable memory circuit comprising at least two memory banks each comprising at least one memory device, the stable memory operation comprising the steps of writing data from a source location of a memory device to a temporary location in the same device, and subsequently writing the data to a destination location in the same device to complete an intra-bank transfer.

In one embodiment, temporary location is in an auxiliary memory area of the device. The auxiliary memory area may be a register.

In another embodiment, the device is a video random access memory, and the temporary location is the serial access register of the video random access memory. In this latter embodiment, the video random access memory has a plurality of serial access registers, one of which is used for the intra-bank transfer, another serial access register being for use in another stable memory operation carried out simultaneously.

In one embodiment, the stable memory operation also includes an inter-bank copy, logical operation time of a host processor being reduced by performing the intra-bank transfer in advance. In this embodiment, the intra-bank transfer may be carried out on a copy-on-write basis.

In another embodiment, the intra-bank transfer is carried out simultaneously in at least two memory devices being replicates of each other. The intra-bank transfer may be carried out simultaneously in all devices, resulting in data in each memory device being checkpointed in a deterministic time, independently of memory size.

The invention will be more clearly understood from the following description of some preferred embodiments thereof,

given by way of example only with reference to the accompanying drawings, in which:-

Fig. 1 is a diagram showing a memory chip for use in a stable memory circuit; and

Figs. 2 to 6 inclusive are diagrams showing stable memory circuits at various stages of stable memory operations being carried out.

Referring to the drawings, and initially to Fig. 1 there is illustrated a memory device, namely, a video random access memory (VRAM) indicated generally by the reference numeral 1 for use in a memory bank of a stable memory circuit. The VRAM 1 comprises a conductor 2 for communication with a host processor and a stable memory manager circuit, and a conductor 3 for a stable data path connecting the bank in which the VRAM 1 is located with another stable memory bank. The VRAM 1 also includes a 512 x 512 x 4 bit memory array 4 and a 512 x 4 bit serial access register 5.

As stated above, the VRAM 1 forms part of a stable memory bank, there being two banks interconnected by a stable data path in a stable memory circuit. Each bank would generally comprise many VRAMs 1, however, it is possible that a bank may have only one VRAM 1.

A stable memory operation may be carried out within the VRAM 1. This operation involves writing data from a source location in the memory array 4 to a temporary location in the serial access register 5. Subsequently, the data is written from the temporary location to a destination location within the memory array 4. This operation is hereinafter referred to as an intra-bank transfer because it occurs within a single memory device of a bank.

In many cases, it would be more practical to carry out the intra-bank transfer for a block (row) of locations of the memory array 4 at once. This is particularly true where the memory device is a VRAM, as the serial access register is wide. Further, in many situations it would be suitable to carry out the intra-bank transfer for a memory address space comprising several blocks.

Because communication between memory devices is not required for the intra-bank transfer, the transfer may be carried out extremely quickly and results in stable memory operations including this transfer being carried out quickly. In practice, the transfer minimises the delay to host processor operations after it has requested a stable memory operation. Examples are given below.

The VRAM 1 may have more than one serial access register 5, in which case one register may be used for an intra-bank transfer, and another for a different stable memory operation.

Referring now to Fig. 2, a stable memory operation including an intra-bank transfer involving a VRAM 1 is illustrated. A host write access is attempted to a memory block 6 of the VRAM 1 as indicated by the arrow 7 shown in Fig. 2(a). However, if this block is marked (M) for copy-on-write, then before the host write is carried out, a separate stable memory manager circuit connected to the VRAM 1 via the host conductor 2 copies the contents of the memory block 6 temporarily to the serial access register 5 as indicated by the arrow 8 of Fig. 2(b). As indicated by the arrow 9 of Fig. 2(c), the contents are then written from the serial access register 5 to another memory block 10 within the same VRAM 1. As shown in Fig. 2(d), the host memory write may then take place to the memory block 6 as indicated by the arrow 11. This operation is referred to as "copy-on-write" and may be used in memory management, stable memory capability management, transaction processing, or persistent storage management.

Marking for copy-on-write may be carried out either by the host processor or by the stable memory manager circuit. If the host processor does this within its memory management unit, then the host processor's write fault handler would have to request the intra-bank transfers from the manager circuit. If the manager circuit does the marking, then it would have to schedule the intra-bank transfers.

For some stable memory operations, for example, atomic actions, checkpoints or transactions, it is necessary that host writes do not occur to those areas of memory involved until the stable memory operation has been completed. This restriction may be removed by marking those areas of memory for copy-on-write at the start of the stable memory operation. When a host write access is attempted to the relevant memory location, the manager circuit may very rapidly transfer the contents of that source location using an intra-bank transfer as described above, after which the host write may be allowed to the source location. The advantage is that copies are taken only if necessary, and space for copying to is only allocated on demand. Further, because of the speed with which an intra-bank transfer may take place, the delay to the host processor after a stable memory operation has been requested is very short.

Referring now to Fig. 3, there are illustrated two stable memory banks, namely bank 0 and bank 1 interconnected by a stable data path 3, and each bank comprising a single VRAM 1. With reference to Fig. 3, there is now described an atomic commit operation, referred to as a FAST commit. This operation involves the sequence of actions below:-

(a) allocate a stable space S0 in bank 0, and a stable space S1 in bank 1 (Fig. 3(a)),

(b) the host processor carries out work in the space S0 (Fig. 3(b)),

(c) the space S0 is marked (M) for copy-on-write (Fig. 3(c)),

(d) carrying out an inter-bank copy from space S0 to space S1 via the stable data path 3 and continue to work in the space S0 (Fig. 3(d)),

(e) if required, compare the contents of spaces S0 and S1 via the stable data path 3 (Fig. 3(e)),

(f) the host processor continuing work in the space S0 (Fig. 3(f)),

(g) if, as shown in Fig. 3(g), a host write access is attempted to space S0 during the physical commit (steps (d) and (e) above) the manager circuit may allocate an extra space S2 in bank 0 and then very rapidly copy-on-write S0 to S2 using an intra-bank transfer (Figs. 3(h) and 3(i)). After the intra-bank transfer, the host write may be allowed to proceed.

Referring to the above operation, it will be noted that only a single memory space S0 is committed and is thus marked for

copy-on-write at the start of the stable memory operation in step (c). After step (c), the commit is logically complete (but not physically complete), and the host processor may continue to work in the space S0 while the physical commit proceeds. The commit is physically complete after step (e).

If, after the copy-on-write, the attempted host write proceeds to the source space S0 of the intra-bank transfer (Fig. 3(j), then at the end of the stable memory operation, the manager circuit may need to perform post-processing to cater for the copy-on-write. If, however, the host write proceeds to the destination space S2 of the intra-bank transfer (Fig. 3(k)), then no post-processing will be necessary, but after the stable memory operation, the host work will have to continue to this destination space rather than to the source space.

Alternatively, any restrictions on writes to locations involved in stable memory operations may be removed by performing a 3-way stable memory operation, where the primary locations of memory are first copied to secondary locations of memory via intra-bank transfers, then the stable memory operation is conducted between one of these and further tertiary locations of memory. Once the intra-bank transfers are complete, the host processor may continue its operations within the other memory locations (those not involved in the stable memory operation). The advantage here is that the host processor is delayed only for the time taken by the intra-bank

transfer, possibly as short as two manager circuit memory accesses.

The following are some examples of 3-way atomic commit operations described with reference to Figs. 4 and 5. In Figs. 4 and 5, there are again two memory banks, bank 0 and bank 1, each of which, for clarity and simplicity, simply has a single VRAM 1. The following stable memory operation, referred to as a CLASSICAL commit is described with reference to Fig. 4.

(a) The host processor allocates spaces S0 and S1 in bank 0 and bank 1 respectively as being for stable memory operations (Fig. 4(a)).

(b) The host processor carries out work in a memory space A of bank 0 (Fig. 4(b)).

(c) & (d) An intra-bank transfer is carried out between space A and space S0 in bank 0 (Figs. 4(c) and 4(d)).

(e) The processor continues to work in the space A while an inter-bank transfer takes place between the spaces S0 and S1 (Fig. 4(e)).

(f) If required, the contents of spaces S0 and S1 are compared (Fig. 4(f)).

(g) Finally, the host processor continues to work in the space A of bank 0 (Fig. 4(g)).

Here, for simplicity only, a single space A has been committed. After step (b) above, a primary space to be committed (space A) is copied to a secondary area of memory (space S0), via an intra-bank transfer. After step (d), the commit is logically complete (but not physically complete), and the host processor may continue to work in the location A while the physical commit proceeds between the secondary location S0 and a further tertiary location S1 of the stable memory circuit. In this specification, the term "logically complete" is intended to mean that the step is complete to the extent that the host processor may continue.

The commit is physically complete after step (f). If a host processor write access is attempted to location A during the physical commit, there will be no effect upon the physical commit process and no action need be taken. Hence, the host processor is delayed only for the time taken for the intra-bank transfer, but unnecessary transfers may take place, and three locations (A, S0 and S1) will always have to be allocated. It will be appreciated that the commit need not be constrained to a single memory location.

Alternatively, another stable memory commit operation, referred to as an IN-PLACE commit would require the following sequence of actions with reference to Fig. 5.

(a) The host processor allocates stable spaces S0 and S1 in bank 0 and bank 1, respectively (Fig. 5(a)).

(b) The host processor carries out work in the space S0 (Fig. 5(b)).

(c) & (d) An intra-bank transfer is carried out between the spaces S0 and location A, (Figs. 5(c) and 5(d)).

(e) & (f) The host processor continues to work in the location A, while simultaneously an inter-bank copy operation takes place between the space S0 and the space S1 (Fig. 5(e)), and if required the contents of the spaces S0 and S1 are compared (Fig. 5(f)).

(g) The host processor continues to work in the space A, (Fig. 5(g)).

Again, for simplicity only, a single space A is to be committed. After step (b) above, the primary area to be committed (space S0) is copied to a secondary area of memory (space A) via an intra-bank transfer. After step (d) the commit is logically complete (but not physically complete) and

the host may continue to work in the secondary space A while the physical commit proceeds between the primary space S0 and a further tertiary space S1 of memory. The commit is physically complete after step (f). If a host write access is attempted to the space A during the physical commit, there will be no effect upon the physical commit process and no action may be taken. Again, the host processor is delayed only for the time taken by the intra-bank transfer, but unnecessary transfers may take place and three spaces (A, S0 and S1) will always have to be allocated. Again, it will be appreciated that the commit need not be constrained to a single block.

If the stable memory operation is likely to occur frequently, it may be implemented with a FOUR-WAY operation by declaring two pairs of spaces and alternatively continuing host processor work in one space of one pair, whilst the stable memory operation proceeds between the other pair, and vice versa. For example, a FOUR-WAY variant of the atomic commit operation, here called a FOUR-WAY commit requires the sequence of actions below.

(a) The host processor allocates stable spaces S0, S2 and in bank 0 and spaces S1 and S3 in bank 1.

(b) The host processor carries out work in the space S0.

(c) An intra-bank transfer operation takes place between space S0 and space S2.

(d) The host processor continues work in space S2, while simultaneously an inter-bank copy operation takes place between the spaces S0 and S1 and subsequently if so required the contents of the spaces S0 and S1 are compared.

(e) The host processor continues work in the space S2.

(f) An intra-bank transfer operation takes place between the space S2 and the space S0.

(g) The host processor continues work in the space S0 and simultaneously an inter-bank copy operation takes place between the space S2 and the space S3 and subsequently, if so required, the contents of these spaces are compared.

(h) The host processor continues work in S0.

Again, for simplicity only, a single memory space is committed. After step (b), the primary area to be committed (space S0) is copied to a tertiary area of memory (space S2) via an intra-bank transfer. After step (c) the commit is logically complete and the host may continue to work in the

tertiary space S2 while the physical commit proceeds between the primary space S0 and the secondary space S1. The commit is physically complete after step (d). If a host write access is attempted to the space S2 during the physical commit, there will be no effect on the physical commit process, and no action need by taken. The next commit will take place in the opposite direction, with an intra-bank transfer of space S2 back to space S0 where host work can then continue while the physical commit is conducted between the spaces S2 and S3. An advantage is that the procedure gains some symmetry, and in particular memory allocation is more symmetric. Again, the host processor is delayed only for the time taken by the intra-bank transfer, but unnecessary transfers may still take place and four spaces (S0, S1, S2 and S3) will always have to be allocated.

A further important advantage of the FOUR-WAY commit relates to protection. When using VRAMs, an intra-bank transfer occurs entirely within the VRAMs so that error detection and correction of the intra-bank transfer by an external circuit is not possible. This reduces the level of protection afforded by error detection and correction codes unless a new generation of VRAMs is designed which expressly perform these functions internally. However, protection would be possible using other forms of dual-ported memory, provided that all of the data within the memory location is available externally for simultaneous error detection and correction.

For all of the above commit operations, the error detection and correction occurs on the stable data paths during the physical commit. If an uncorrectable error is found, then all but the FOUR-WAY commit are in error. For the FOUR-WAY commit there is always a full back-up in at least one of the four locations of memory involved.

Fig. 6 shows a stable memory circuit 20 having two banks, namely bank 0 and bank 1 which are not interconnected by a stable data path. Each bank comprises four VRAMs 1 so that the VRAMs in one bank are replicated in the other bank. Accordingly, in the event of a fault within one VRAM, the data may be accessed from a replicate device. Inter-bank copying is not required for a commit operation - data can be committed within the VRAM that holds the data using intra-bank transfers only, provided the operations are performed simultaneously on its replicate devices.

In Fig. 6(a), writes take place to the source block S0 in two of the VRAMs 1, being replicates of each other so that the writes are effectively broadcast. In Figs. 6(b) and 6(c) an intra-bank transfer takes place between the source block S0 and the destination block S1 in each of the VRAMs 1. In Fig. 6(d) reads take place from the source block S0 in two of the VRAMs 1. Any errors are detected using a voting circuit 21 which compares the contents of the block S0 in different VRAMs 1. The important point is that because the operations are

performed simultaneously on replicate VRAMs, inter-bank copying need not take place. In particular, the stable memory operations described above will allow all the memory to be checkpointed extremely rapidly by performing the commits for all VRAMs 1 simultaneously. The checkpointing will take place with a time T_{chk} equal to:-

$$T_{chk} = 2*(\text{VRAM capacity/serial access register capacity})*(\text{transfer cycle time})$$

Thus all of memory may be checkpointed in a deterministic time, independent of the total memory capacity. For modern VRAMs T_{chk} is of the order of 100 microseconds. The significance of this can be seen when used in a real-time system where deadlines must be met:- checkpointing of all of memory can be undertaken regularly, each checkpoint taking just 100 microseconds. On a fault, backward error recovery to the previous checkpoint state can be accomplished (for all of memory) in another 100 microseconds. For the first time, this allows checkpointing with backward error recovery to become the basis for fault-tolerance for both hardware and software in hard-real-time systems. Normally this would be assumed to apply just to hard-real-time recovery from hardware faults.

The invention has been described in use with VRAMs as the memory devices. However, any memory device within which there is an auxiliary memory area which can provide a temporary

location may be used. Registers in devices such as VRAMs have been found to be particularly suitable. With minor modifications the sense amplifiers in DRAMs could also serve this purpose. As shown in Fig. 6, it is not essential that the memory banks be interconnected by a stable data path. The intra-bank transfer may be used with advantage whether or not inter-bank operations also take place.

Although not shown in the specific embodiments, it is envisaged that a processing operation may take place on the data between being written to the temporary location and to the destination location.

The invention is not limited to the embodiments hereinbefore described, but may be varied in construction and detail.

CLAIMS

1. A stable memory operation carried out in a stable memory circuit comprising at least two memory banks each comprising at least one memory device, the stable memory operation comprising the steps of writing data from a source location of a memory device to a temporary location in the same device, and subsequently writing the data to a destination location in the same device to complete an intra-bank transfer.

2. A stable memory operation as claimed in Claim 1, wherein the temporary location is in an auxiliary memory area of the device.

3. A stable memory operation as claimed in Claim 2, wherein the auxiliary memory area is a register.

4. A stable memory operation as claimed in any preceding claim, wherein the device is a video random access memory, and the temporary location is the serial access register of the video random access memory.

5. A stable memory operation as claimed in Claim 4, wherein the video random access memory has a plurality of serial access registers, one of which is used for the intra-bank transfer, another serial access register being for use in

another stable memory operation carried out simultaneously.

6. A stable memory operation as claimed in any preceding claim, wherein the stable memory operation also includes an inter-bank copy, logical operation time of a host processor being reduced by performing the intra-bank transfer in advance.

7. A stable memory operation as claimed in Claim 6 wherein the intra-bank transfer is carried out on a copy-on-write basis.

8. A stable memory operation as claimed in any of Claims 1 to 5, wherein the intra-bank transfer is carried out simultaneously in at least two memory devices being replicates of each other.

9. A stable memory operation as claimed in Claim 8, wherein the intra-bank transfer is carried out simultaneously in all devices, resulting in data in each memory device being checkpointed in a deterministic time, independently of memory size.

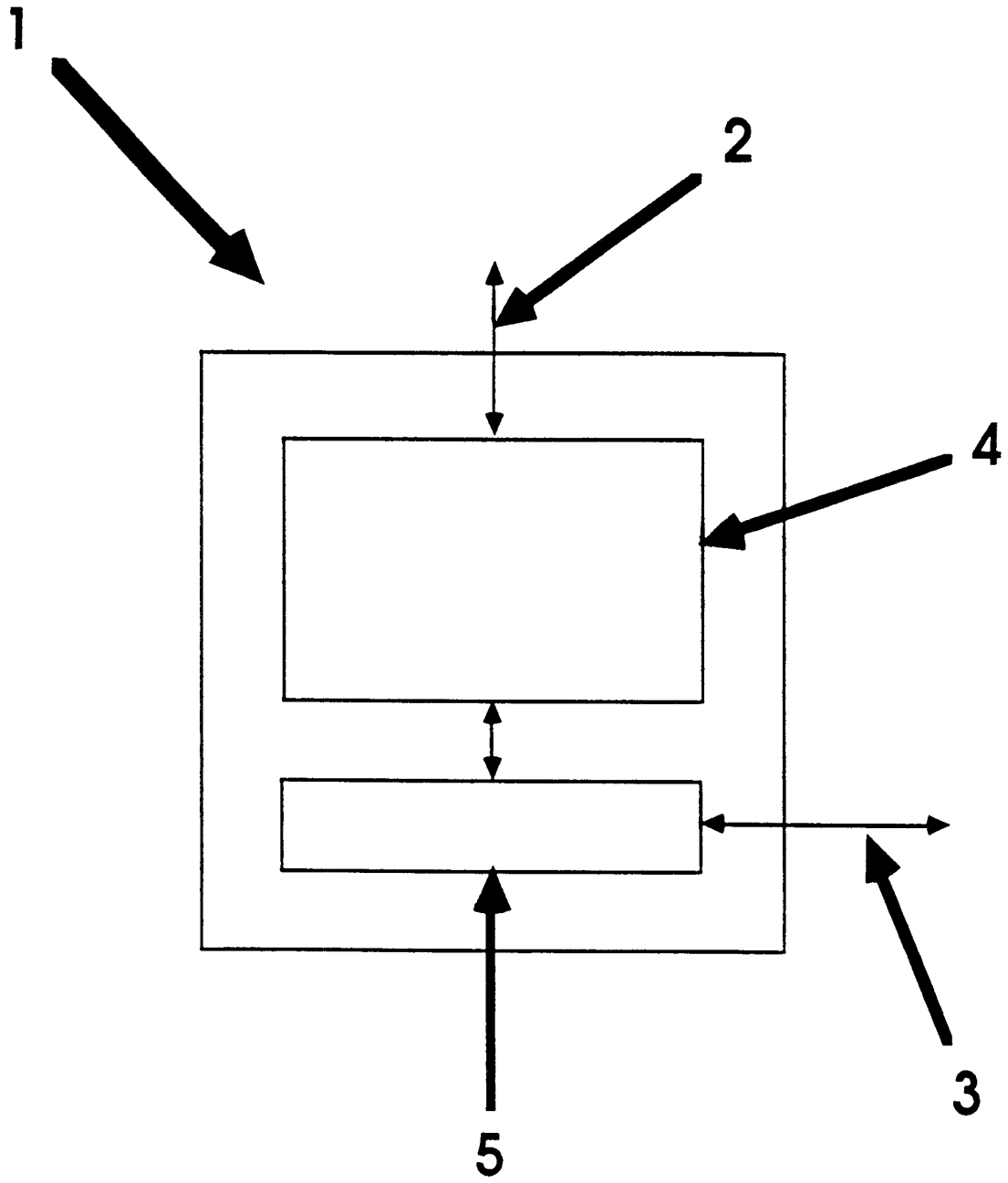


Fig. 1

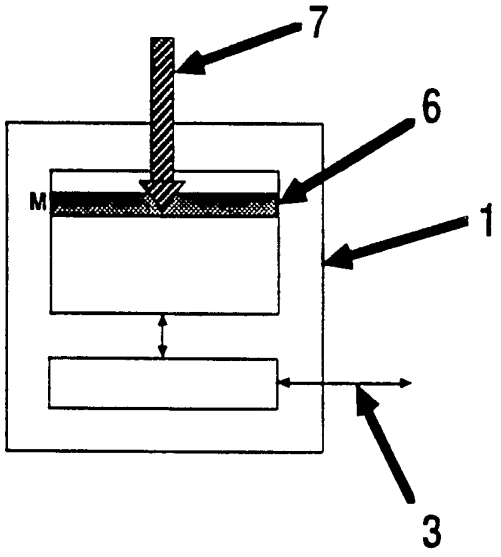


Fig. 2(a)

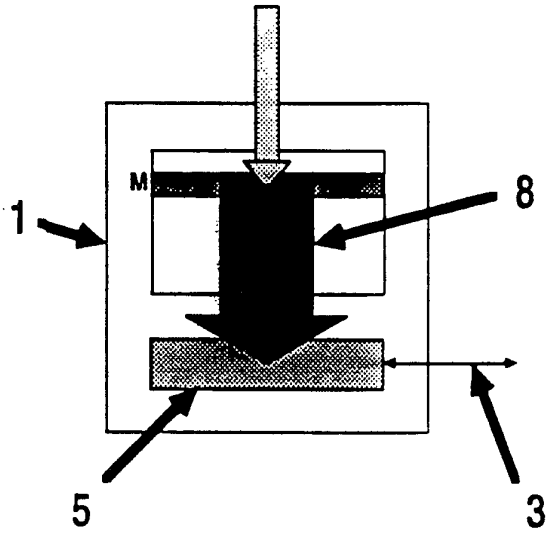


Fig. 2(b)

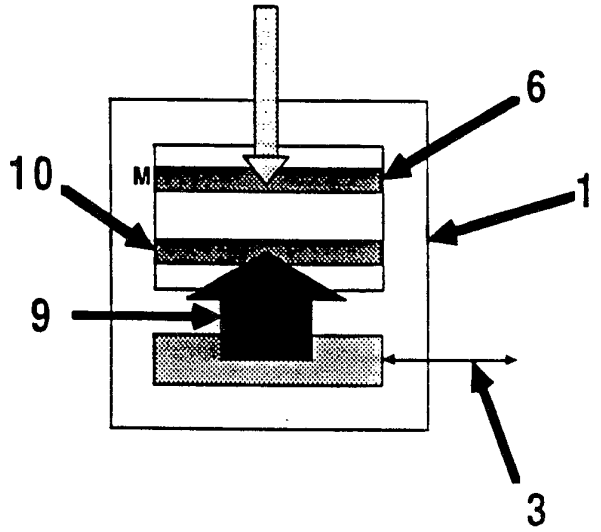


Fig. 2(c)

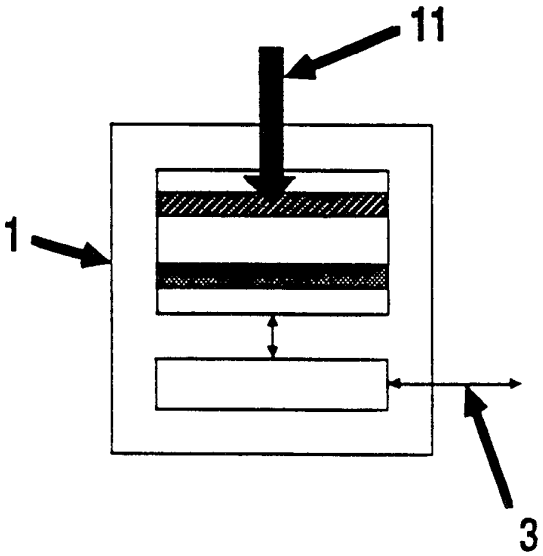


Fig. 2(d)

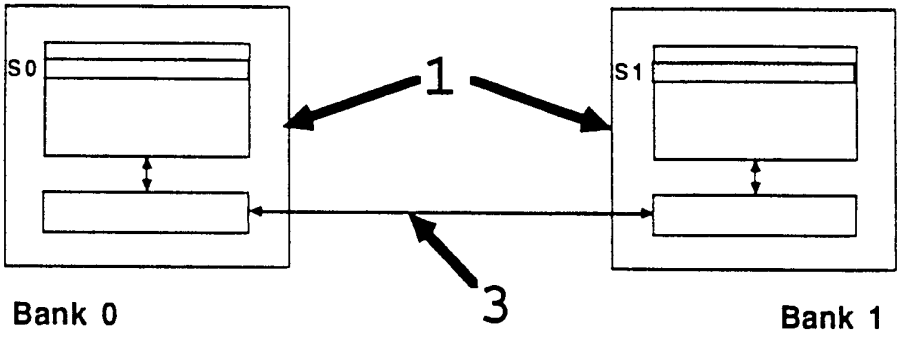


Fig.3(a)

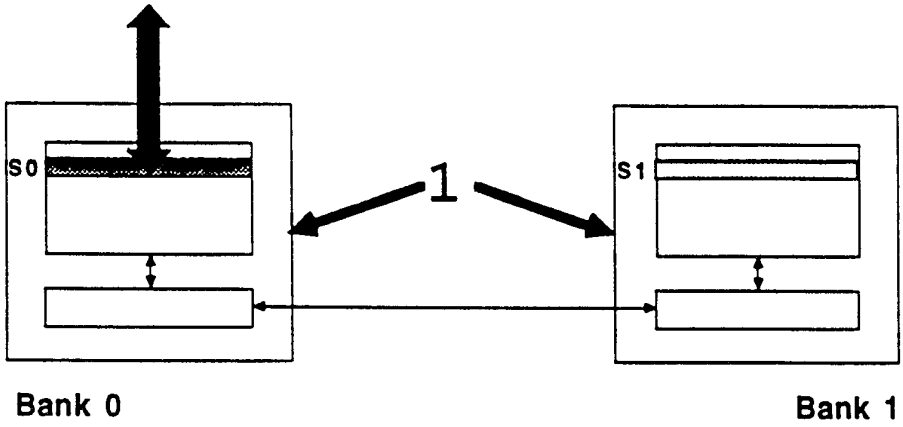


Fig.3(b)

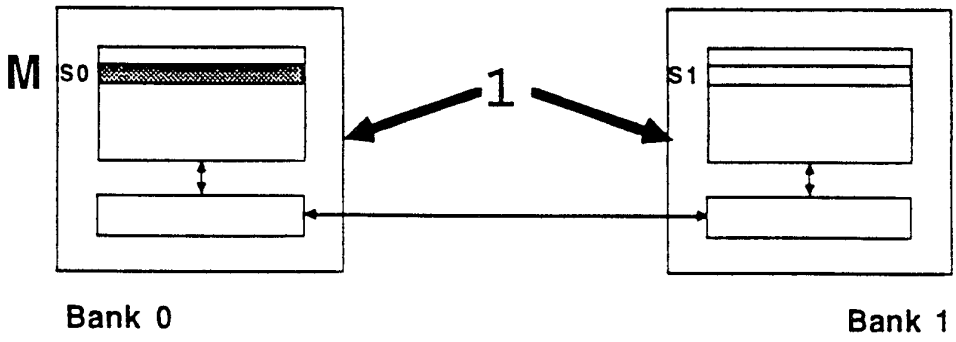


Fig.3(c)

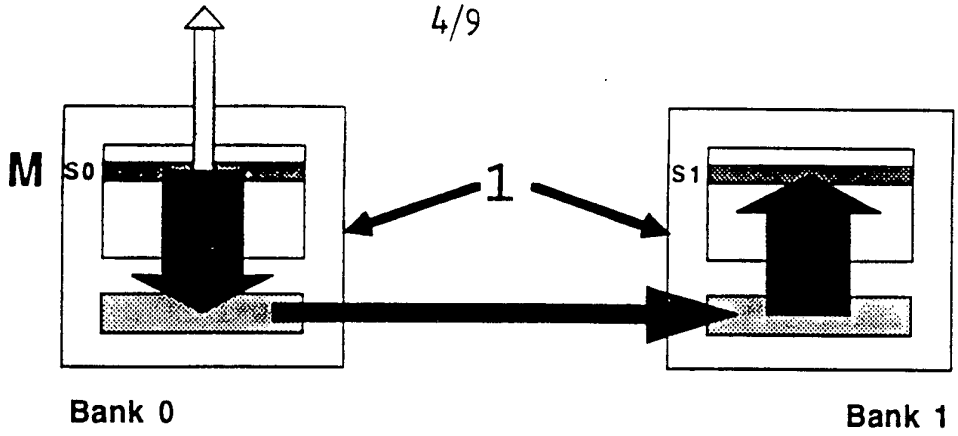


Fig.3(d)

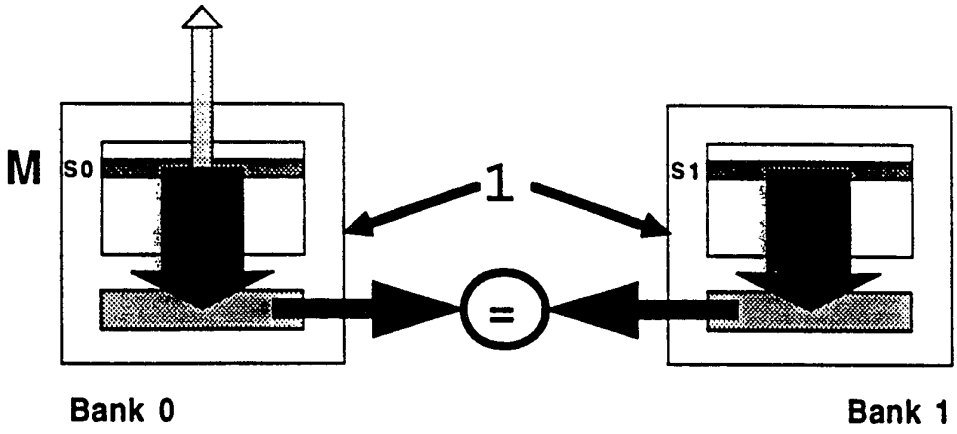


Fig.3(e)

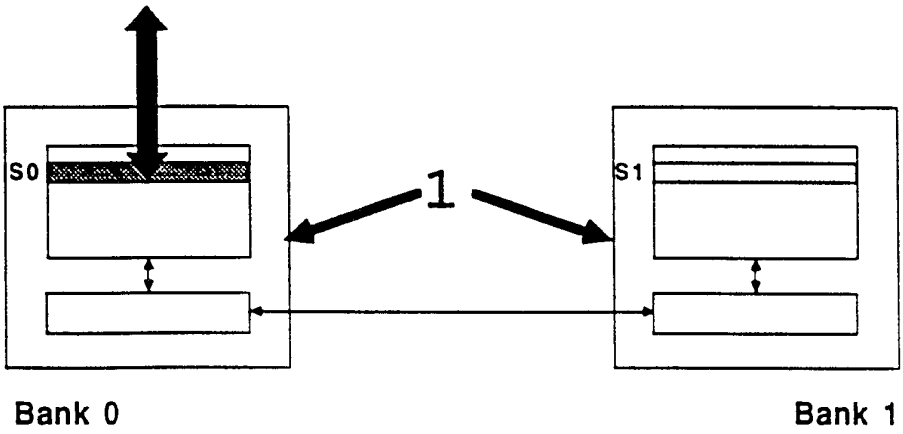


Fig.3(f)

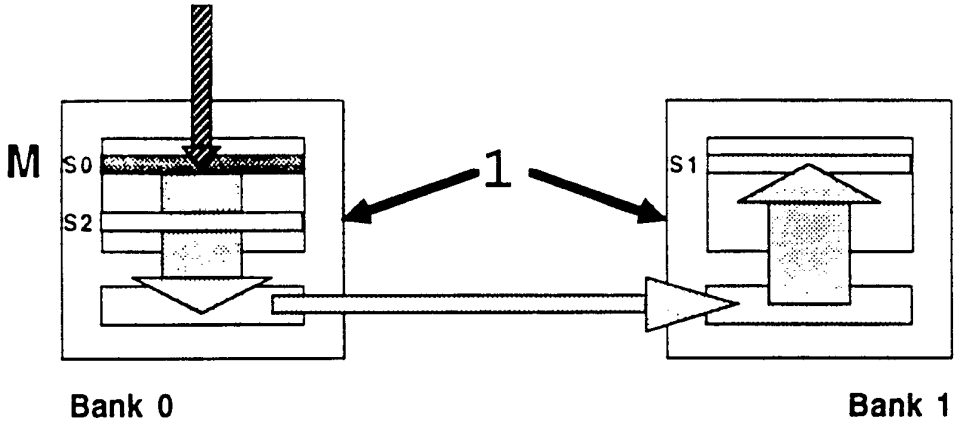


Fig.3(g)

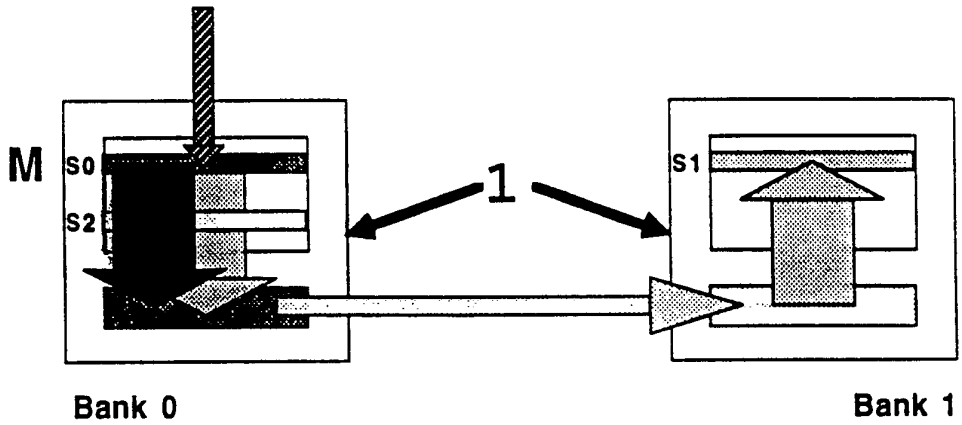


Fig.3(h)

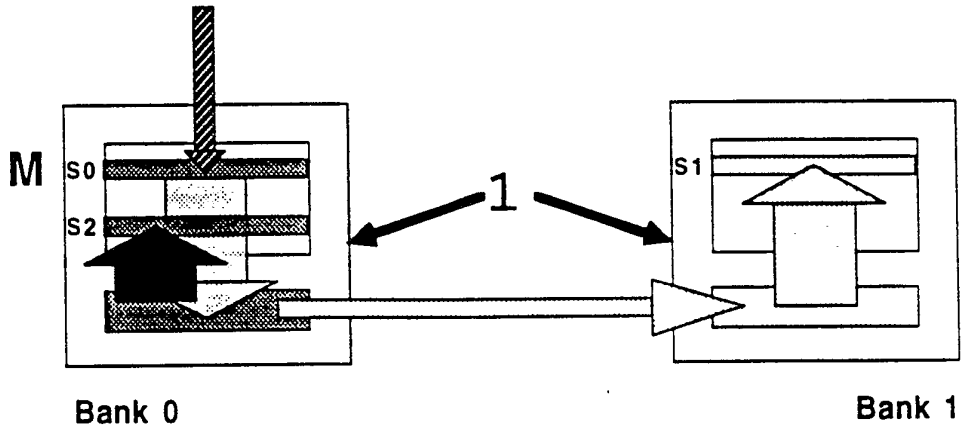


Fig.3(i)

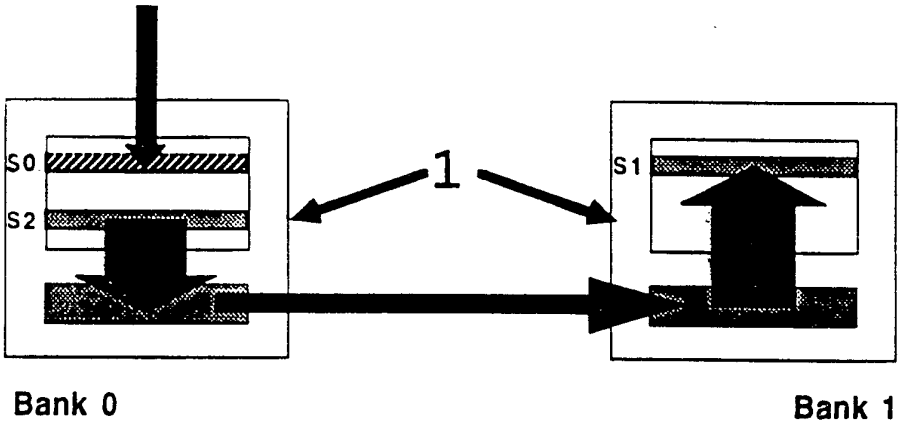


Fig.3(j)

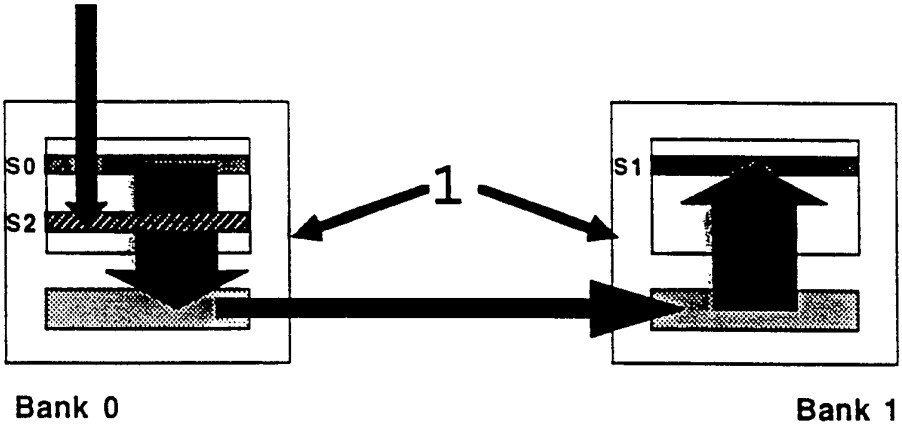


Fig.3(k)

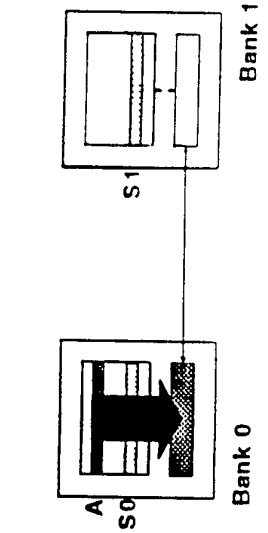


Fig.4(c)

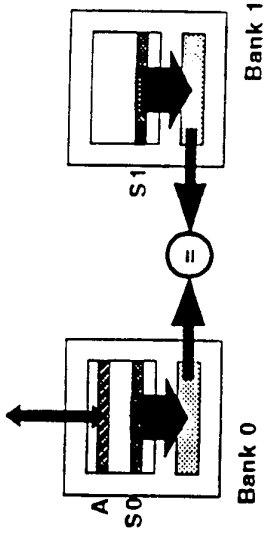


Fig.4(f)

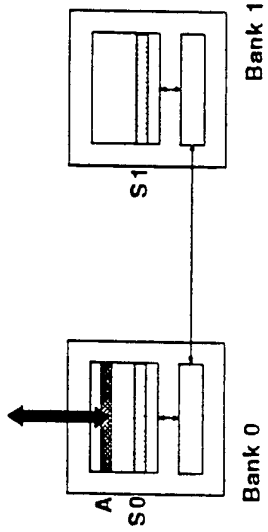


Fig.4(b)

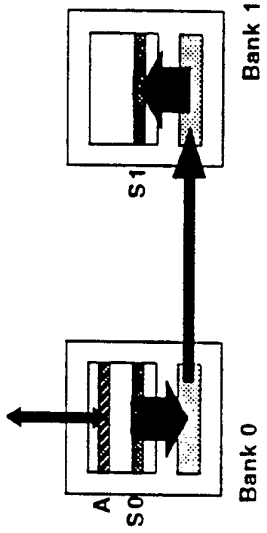


Fig.4(e)

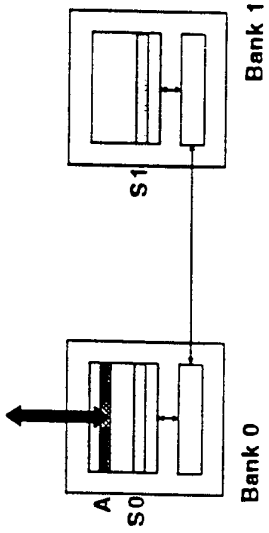


Fig.4(g)

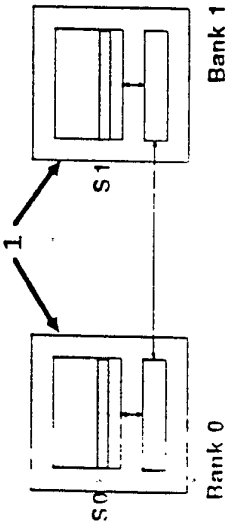


Fig.4(a)

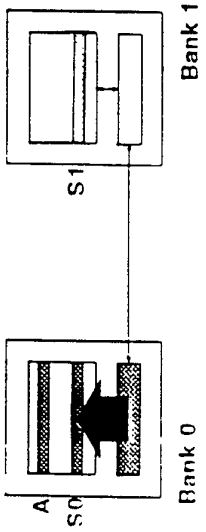


Fig.4(d)

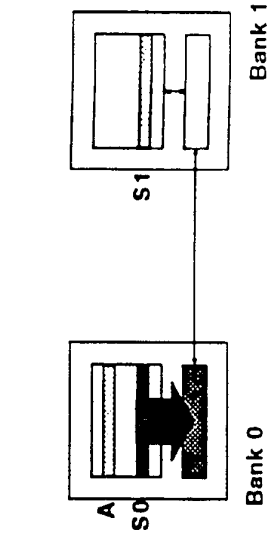


Fig.5(c)

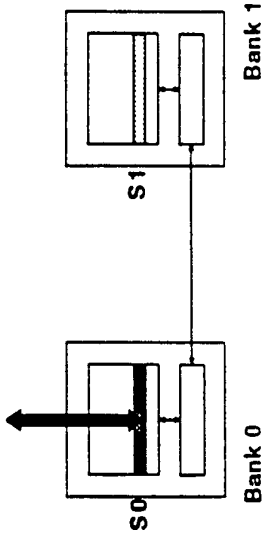


Fig.5(b)

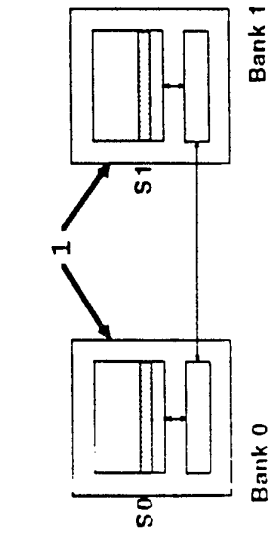


Fig.5(a)

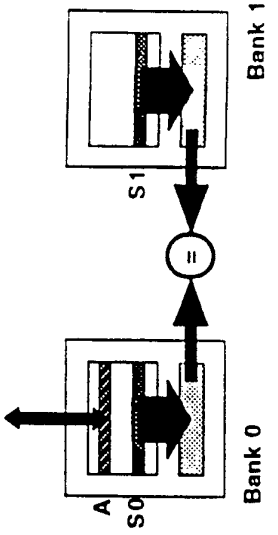


Fig.5(f)

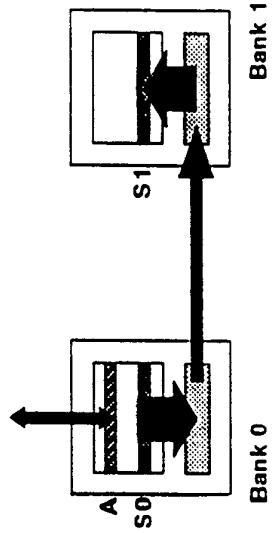


Fig.5(e)

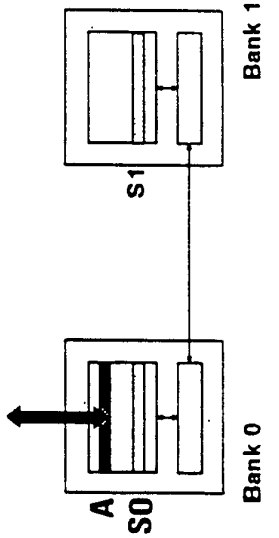


Fig.5(g)

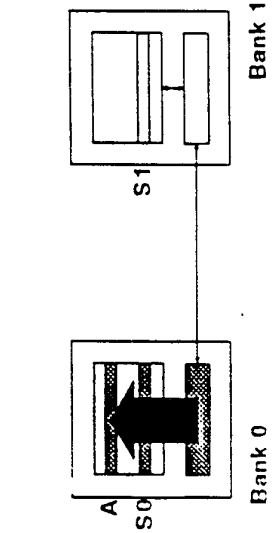


Fig.5(d)

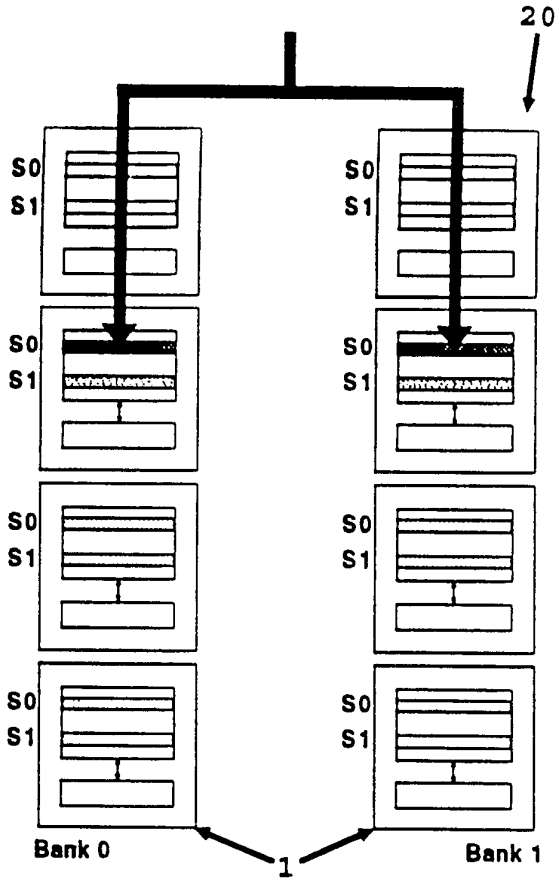


Fig.6(a)

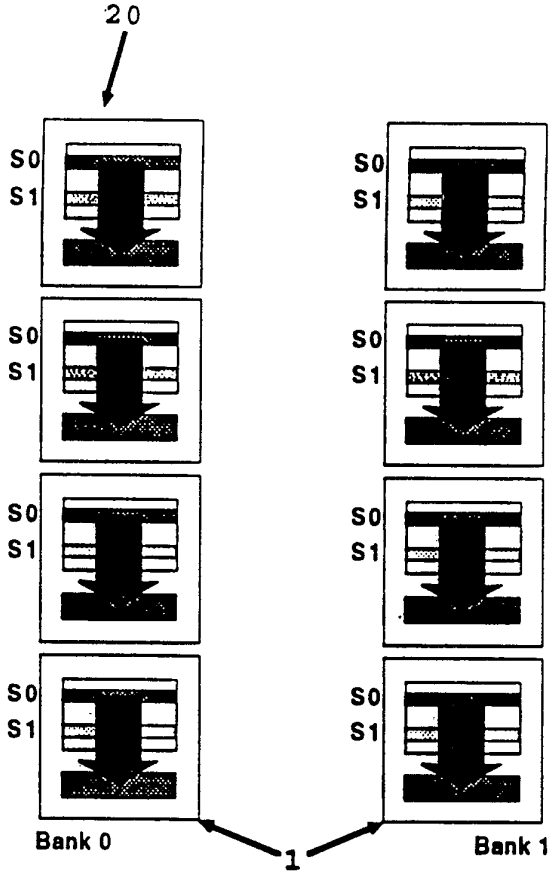


Fig.6(b)

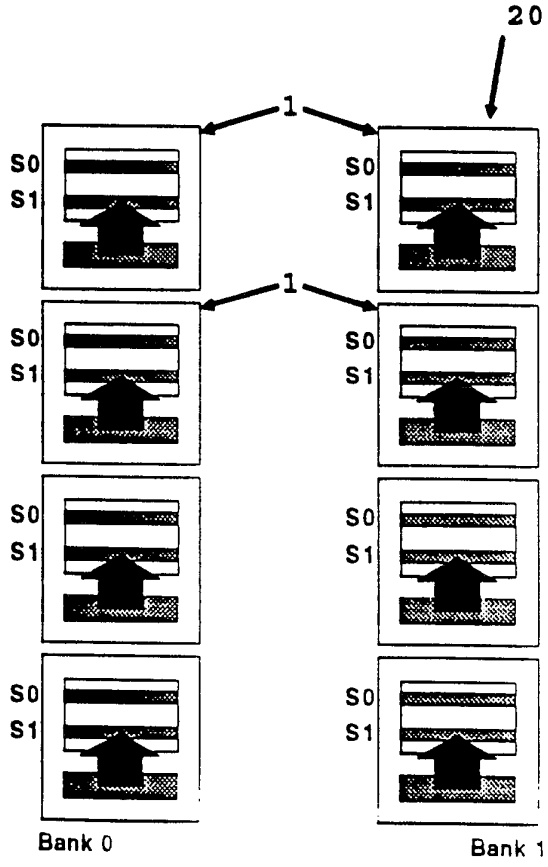


Fig.6(c)

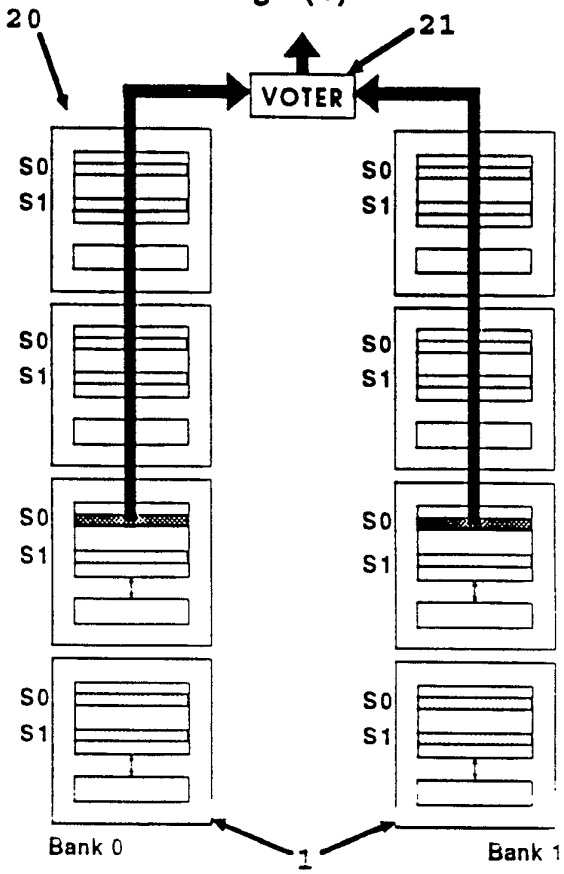


Fig.6(d)

INTERNATIONAL SEARCH REPORT

International Application No PCT/GB 92/00608

I. CLASSIFICATION OF SUBJECT MATTER (if several classification symbols apply, indicate all)[6]

According to International Patent Classification (IPC) or to both National Classification and IPC

Int.Cl. 5 G06F11/20; G06F12/06

II. FIELDS SEARCHED

Minimum Documentation Searched[7]

Classification System	Classification Symbols
Int.Cl. 5	G06F

Documentation Searched other than Minimum Documentation to the Extent that such Documents are Included in the Fields Searched[8]

III. DOCUMENTS CONSIDERED TO BE RELEVANT[9]

Category °	Citation of Document,[11] with indication, where appropriate, of the relevant passages[12]	Relevant to Claim No.[13]
A	US,A,4 799 186 (PLOYETTE) 17 January 1989 see abstract; figures 1-2 see column 5, line 3 - line 12 see column 6, line 31 - column 7, line 32 see claim 1 ---	1,6,8,9
A	US,A,3 866 182 (YAMADA ET AL.) 11 February 1975 see abstract; figure 1 see column 1, line 52 - line 68 see column 4, line 16 - line 46 see claim 1 ---	1,6,8,9

-/--

° Special categories of cited documents :[10]

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

IV. CERTIFICATION

Date of the Actual Completion of the International Search	Date of Mailing of this International Search Report
01 JULY 1992	13 JUIL. 1992
International Searching Authority	**Signature of Authorized Officer**
EUROPEAN PATENT OFFICE	SARASUA GARCIA

III. DOCUMENTS CONSIDERED TO BE RELEVANT (CONTINUED FROM THE SECOND SHEET)

Category °	Citation of Document, with indication, where appropriate, of the relevant passages	Relevant to Claim No.
A	EP,A,0 418 030 (THE PROVOST, FELLOWS AND SCHOLARS OF, TRINITY COLLEGE DUBLIN) 20 March 1991 cited in the application see abstract; figure 1 see column 4, line 20 - line 48 see claim 1 ---	1,4

ANNEX TO THE INTERNATIONAL SEARCH REPORT ON INTERNATIONAL PATENT APPLICATION NO. GB 9200608 SA 58031

This annex lists the patent family members relating to the patent documents cited in the above-mentioned international search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information. 01/07/92

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US-A-4799186	17-01-89	FR-A- 2591775	19-06-87
		CA-A- 1254666	23-05-89
		EP-A,B 0228329	08-07-87
		JP-A- 62217345	24-09-87
US-A-3866182	11-02-75	None	
EP-A-0418030	20-03-91	AU-A- 6235190	14-03-91
		GB-A- 2237666	08-05-91