PATENT SPECIFICATION (11) 1 595 740

1 595 740

(21) Application No. 22862/78 (22) Filed 25 May 1978

(44) Complete Specification published 19 Aug. 1981

(51) INT. CL.³ G06F 13/00

(52) Index at acceptance

G4A 13E NV

(72) Inventor YUTAKA KATSUMATA

(54) DATA PROCESSING APPARATUS

(71) We, FUJITSU LIMITED, a Japanese Corporation, of 1015, Kamikodanaka, Nakahara-ku, Kawasaki, Japan, 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 data processing apparatus, and more particularly to the provision of a dynamic address translation system for a channel or sub-system adapter in apparatus which employs a main memory of the apparatus in common with a central processing unit of the apparatus.

In one example of dynamic address translation as employed by data channels in a previously proposed data processing apparatus, page fixing processing is previously conducted when making access to a main memory of the apparatus from channels. In order to allow access to this fixed page from channels, a channel invalid bit is made OFF and thereafter an SIO (start input/output) instruction is issued for making access from channels. Such page fixing processes and the channel invalid bit operations are controlled by the operating system of the apparatus. However, the following problems can be encountered in such previously proposed apparatus; the overhead for searching the page to be fixed is applied also on any input/output devices to be connected to channels prior to the execution of channel programs, and the number of pages being fixed increases when starting a number of input/output devices, thereby real memory can run short.

Moreover, in an example of channel dynamic address translation in a previously proposed data processing apparatus, access protection for the main memory has been performed by comparing a storage key in every physical page within the main memory and a channel key to be transferred to channels while the SIO instruction is issued. In a virtual memory system, the access protection should naturally be performed in terms of units of logical data. However, this system has the following problems; the logical access protection data should be mapped to the physical access protection mechanism, and there is an architectural illogicality such that the channel must recognise logical addresses for addressing and physical keys for access protection, and the overhead of mapping is also applied.

According to the present invention there is provided, a dynamic address translation system in an arrangement which includes a central processing unit, a main memory and a channel or sub-system adapter and in which the main memory stores an address translation table for translating logical addresses to a physical addresses wherein said dynamic address translation system is characterized by the provision of a register, in the channel or sub-system adapter, for storing a copy of one entry in the address translation table in the main memory and a bit for indicating validity of the copy, and is operable so that when the central processing unit has issued an instruction for updating an entry within said address translation table or an instruction for advising of alteration of said entry, the bit indicating validity of the copy is set to a value indicating that the copy is not valid in accordance with such instruction.

Further there may be provided counter means for indicating addresses in a memory data unit (e.g. a data block) corresponding to said one entry, and when a data transfer request requesting data transfer in block units is generated by an input/output device or sub-system to be connected to said channel or sub-system adapter,

access is made to said address translation table, the contents of one entry among several included in said address translation table are copied to said register and access is made to the main memory on the basis of the contents of said register only

(1) at the time of starting the data transfer,

(2) when a memory address for making access to the main memory reaches a memory data unit boundary,

(3) when the bit indicating effectiveness of said register is displayed as an invalid bit;

and in other cases, access is not performed to said address conversion table, but to the main memory by utilising the address generated from said register and said counter means.

This invention can realize a dynamic address translation system, for which page fixing must be done previously in the channel or sub-system adapter for an input/output device or sub-system, which is allowed to interrupt during data transfer.

This invention can realize an architecture such that page fixing is closed within the software range; namely such that it is not necessary for hardware to fix the page in the input/output device which requires page fixing, or in the channel or sub-system adapter which supports the sub-system.

This invention can further realize a channel or sub-system connecting unit which executes access protection on the basis of logical access protection data as in the case of a central processing unit (CPU), in a data processing system such that access protection for a page is effected on the basis of the logical access data corresponding to a logical page.

This invention can realize a dynamic address translation system for attaining abovementioned various purposes in which access to the address conversion table need not be made from channels or sub-system adapter each time memory is accessed, with economy in hardware and with high speed operation.

Reference will now be made, by way of example, to the accompanying drawings, in which:–

Figure 1 is a synoptic block diagram of a data processing system in which an embodiment of the present invention may be employed.

Figure 2 is a block diagram illustrating in more detail parts of a system having a general configuration as shown in Figure 1, and in accordance with an embodiment of the present invention,

Figure 3 is a block diagram for use in explanation of operations effected in an embodiment of the present invention, and

Figure 4 is a block diagram for use in explanation of further operations effected in an embodiment of the present invention.

The system shown in Figure 1 includes a main memory (MM) 1, central processing units (CPU) 2-1 and 2-2, channels 3-1 and 3-2, sub-system adapter 4, input/output control units (IOC) 5-1 and 5-2, input/output devices (IO) 6 and sub-system 7.

Channels 3-1, 3-2 and sub-system adapter 4 shown in Figure 1 employ the main memory 1 in common with central processing units 2-1 and 2-2, and also employ a dynamic address conversion system.

Figure 2 illustrates in relation to main memory 1 a central processing unit 2, and a channel 3, dynamic address conversion functions as performed in an embodiment of the present invention.

In Figure 2 there are illustrated main memory (MM) 1; a central processing unit (CPU) 2; a channel (CH) 3; a segment table 10; a page table 11; a page table entry 12; a page invalid bit (PI) 13; access protection data (ACC) 14; a physical page address (PPA) 15; a control register (CR) 16; a CCW (channel control word) address storage register (CCAW) 17; a main memory logical address storage register (CLAR) 18; a CCW storage register (CCCW) 19; a page table entry data storage register (CTLB) 20; a control register (CCR) 21 within the channel 3, a counter 22; a main memory physical address storage register (CPAR) 23; an invalid bit (I) 24; a comparator circuit 25; and a unit control word (UCW) memory 26.

Each CPU and each channel in Figure 1 may comprise the relevant items shown in Figure 2.

The main memory 1 includes the segment table 10 and page table 11, which are used for dynamic address translation by the central processing unit (CPU) 2. Page table entry 12, within the page table 11, includes page invalid bit (PI) 13 the value of which indicates the effectiveness (validity) of the relevant page table entry, access protection data (ACC) 14 for the logical page corresponding to the page table entry, and physical page address (PPA) 15 in which the relevant logical page actually exists. In each central processing unit (CPU) 2, control register (CR) 16 is provided for holding contents which indicate the heading address of a segment table 10 which relates to the logical address space of the main memory to be recognised by a program to be executed. In each channel CCW address storage register (CCAW) 17 is provided for holding the logical address and the channel access key of a channel control word (CCW) which is to be executed. The main memory logical address storage register (CLAR) 18 is provided for holding a main memory address (logical address) which is to be accessed, the CCW storage register (CCCW) 19 is provided for holding a CCW currently being executed, the page table entry data storage register (CTLB) 20 is provided for holding a copy of a page table entry relating to the logical page to which an access is currently to be made and an invalid bit (I bit) relating to the validity of the copy, and the control register (CCR) 21 in the channel is provided for storing the heading address of the segment table which relates to the space to which an access is to be made. The contents which the main memory logical address storage register (CLAR) 18 within each

channel (CH) 3 is provided to hold comprise a logical page address part and an address part indicating a location within the logical page, and contents of the address part indicating a location within a page can be altered by means of the counter 22. An access to the main memory from a channel (CH) is performed by means of the main memory physical address storage register (CPAR) 23 which always holds contents combining the physical page address (PPA) in the page table entry data storage register (CTLB) 20 and the in-page address in the main memory logical address storage register (CLAR) 18. An invalid bit (I) 24 in the page table entry data storage register (CTLB) 20 is synchronously set to "1" when any central processing unit (CPU) 2 issues a PURGE TLB instruction as described later. Moreover, each channel (CH) 3 is provided with a comparator circuit 25 which compares the access protection data (ACC) in the page table entry data storage register (CTLB) 20 and a channel access key in the CCW address storage register (CCAW) 17. In addition each channel (CH) 3 is also provided with a variety of other components, for example, the unit control word (UCW) memory 26 having a storage area (sub-channel) which holds information relating to the executable condition of channel programs, as in the case of previously proposed channels.

Dynamic address translating operations are effected in each channel as explained below. The central processing unit (CPU) 2 sets, at first, the heading address of channel program and a channel access key into a channel address word (CAW) (not illustrated) when an I/O request is first issued, and then issues an SIO instruction. The channel (CH) 3 designated in the operand of the SIO instruction sets the channel address word (CAW) and the content (heading address of segment table 10) of control register (CR) 16 into the CCW address storage register (CCAW) 17 and control register (CCR) 21 within the channel, and an access to a segment table entry is made on the basis of an address which is obtained from the heading address for the segment table 10 designated by the contents of the control register (CCR) 21 in the channel and the segment address part of the channel program heading address which is set in the main memory logical address storage register (CLAR) 18 from CCAW 17. Then, an access is made to a page table entry 12 on the basis of an address which is obtained from the page address of the page table heading addresss included in the accessed segment table entry and the channel program heading address in the main memory logical address storage register (CLAR) 18, and thereafter the content (13, 14, 15) of the accessed page table entry is copied to the page table entry data storage register (CTLB) 20 in the channel (CH) 3, with the invalid bit (I) 24 in the page table entry data storage register (CTLB) 20 being made "0". The above-described processes are known as dynamic address translation (DAT).

The channel operates to couple the physical page address (PPA, 15) in the page table entry data storage register (CTLB) 20 (copied from the page table entry 12) and the in-page address in the main memory logical address storage register (CLAR) 18, and generates the real address for access to the main memory (MM) 1 in the main memory physical address storage register (CPAR) 23, makes access to the main memory (MM) 1 and then fetches the accessed channel control word (CCW). The channel (CH) 3 decodes this channel control word (CCW) and starts an input/output device (IO) in dependence upon the results of the decoding. The fetched channel control word is set in CCCW 19. When the execution of this channel control word (CCW) involves an access to the main memory, the channel (CH) 3 sets the logical address in the channel control word (CCW) into the main memory logical address storage register (CLAR) 18 from CCCW 19 to undergo dynamic address translation (DAT) as explained above, and then sets the derived physical address to the main memory physical address storage register (CPAR) 23.

When execution of channel control word (CCW) involves accessing a main memory area comprising a number of continuous or consecutive bytes, after a value has first been set up in the main memory physical address storage register (CPAR) 23, the channel (CH) 3 makes access, until following "Conditions" are satisfied, to the main memory in accordance with physical addresses generated by the physical page address in the page table entry data storage register (CTLB) 20 and in-page addresses in the main memory logical address storage register (CLAR) 18 which are counted up by means of the counter 22.

"Conditions"

(1) Overflow occurs in the in-page address as a result of the counting up performed on the in-page address part in the main memory logical address storage register (CLAR) 18.

(2) The invalid bit (I) 24 in the page table entry data storage register (CTLB) 20 becomes "1".

In a case in which the above conditions are set up, the channel (CH) 3 carries out dynamic address translation (DAT) operations as explained above and makes an access to the main memory (MM) 1 by set-

ting a physical address in the main memory physical address storage register (CPAR) 23.

Thereafter, the channel (CH) 3 executes sequentially the channel control words (CCW) in the channel programs as in the case of previously proposed channels, and whenever an execution start for a new channel control word is accompanied by an access to the main memory, it repeats the above-mentioned operations.

In addition, whenever the content of the page table entry data storage register (CTLB) 20 is rewritten in the course of dynamic address translation (DAT), a check is always effected on the page invalid bit (PI) and access protection data (ACC). When the page invalid bit (PI) is "1", the channel generates a page fault exception, and when the result of comparison between the access protection data (ACC) and channel access key in the CCW address storage register (CCAW) 17 indicates an access inhibit condition, the comparator circuit 25 generates a protection exception.

When these exceptions are generated the channel (CH) 3 generates an interruption request which is delivered to the central processing unit (CPU) 2 and issues a command to cause a ceasing of operations to the relevant input/output device (IO) in order to stop the execution of channel programs.

There will now be explained operations for setting the invalid bit (I) 24 in the page table entry data storage register (CTLB) 20 to "1" by means of the PURGE TLB instruction.

Figure 3 shows a configuration of circuitry directly related to the execution of a PURGE TLB instruction in the central processing unit (CPU) 2 and the channel (CH) 3.

In Figure 3 there are shown main memory (MM)1; a central processing unit (CPU)2; a channel (CH) 3; page table entry data storage register (CTLB) 20; invalid bit (I) 24; unit control word (UCW) memory 26; instruction register 30; decoder (DEC) 31; PURGE TLB instruction control circuit 32; table conversion index buffer (TLB) 33; memory access control circuit 34; channel PURGE TLB control circuit 35; data transfer control circuit 36; and I/O interface control circuit 37.

The PURGE TLB instruction is an instruction which is used to advise of updating of an entry in the address translation table or of alteration of said entry. The operations effected in the execution of such an instruction will be explained with reference to Figure 3.

The central processing unit (CPU) 2 decodes an instruction code (OP) in the instruction register 30 by means of the decoder (DEC) 31 and starts the PURGE TLB instruction control circuit 32 when a PURGE TLB instruction is present in the instruction register. The PURGE TLB instruction control circuit 32 informs channel 3 of the issuance of the PURGE TLB instruction, as indicated by line ① in Figure 3.

The channel PURGE TLB control circuit 35, upon receiving the information indicating the issuance of the PURGE TLB instruction, sends an interruption command to the data transfer control circuit 36, when the channel is presently in a data transfer condition, in order to interrupt data transfer operations at the most efficacious moment, as illustrated by line ② in Figure 3. Then, the channel PURGE TLB control circuit 35 informs the PURGE TLB instruction control circuit 32 of the central processing unit (CPU) 2 that interruption of the data transfer has been effected as indicated by line ③. Then, the PURGE TLB instruction control circuit 32 rewrites the invalid bit (I) in every entry in the table conversion index buffer (TLB) 33 in the central processing unit (CPU) 2 to "1" and then orders the channel (CH) 3 also to alter the invalid bit (I) 24 of the page table entry data storage register (CTLB) 20 to "1". This information causes the channel PURGE TLB control circuit 35 to rewrite the invalid bit (I) 24 in the page table entry data storage register (CTLB) 20 to "1". These operations are indicated by lines ④ in Figure 3. The channel PURGE TLB control circuit 35 informs the central processing unit (CPU) 2 when invalidation of the page table entry data storage register (CTLB) 20 has been completed as indicated by line ⑤. Thereby, execution of the PURGE TLB instruction is complete.

There will now be explained operations involved in informing a central processing unit (CPU) and an input/output device of a page fault exception and a protection exception.

Figure 4 shows a configuration for circuitry directly related to operations involved in page fault exception and protection exception in the channels.

In Figure 4, there are shown central processing units (CPU) 2-1 and 2-2; a channel (CH) 3; CCW address storage register (CCAW) 17; main memory logical address storage register (CLAR) 18; page table entry data storage register (CTLB) 20; comparator circuit 25; unit control word (UCW) memory 26; data transfer control circuit 36; I/O interface control circuit 37; selector circuit 40; interrupted CPU determination circuit 41; page fault address send gate 42; page fault detection line 43; protection exception detection line 44; page fault interruption request line 45; protection exception interruption request line 46; page fault address send designation signal line 47;

interruption end information line 48; and page fault address line 49.

Operations involved in informing a central processing unit (CPU) and an input/output device of a page fault exception are performed in the manner described below.

In Figure 4 the data transfer control circuit 36 is informed of the fact that the page invalid bit (PI) in the page table entry data storage register (CTLB) 20 is "1" by means of the page fault detection line 43. The data transfer control circuit 36 informs the interrupted CPU determination circuit 41 that there is a page fault interruption request via the page fault interruption request line 45. The interrupted CPU determination circuit 41 determines the CPU to be interrupted as in the case of a channel cross-call and indicates a data route in the selector circuit 40. The data transfer control circuit 36 issues an interruption request and simultaneously opens the page fault address send gate 42 via the page fault address send designation signal line 47 and puts the fault address sent from the main memory logical address storage register (CLAR) 18 on the page fault address line 49. Thus, the page fault interruption request and page fault address are sent to the designated central processing unit (CPU) from the channel (CH)3via the page fault interruption request line 45 and page fault address line 49. The central processing unit (CPU) performs operations similar to those performed when a page fault is generated in the central processing unit (CPU) in order to generate conversion exceptions and simultaneously stores the address wherein page fault is generated in the channel in a fixed address of a prefixed area of the central processing unit (CPU). Here, the storage area used may be the same as that for conversion exception address storage in the central processing unit (CPU). The data transfer control circuit 36 in the channel (CH) 3 issues an interruption request to the central processing unit side, and simultaneously requests the I/O interface control circuit 37 to execute a completion sequence designated on the side of the channel. The I/O interface control circuit 37 executes a completion sequence such as in a case where an error is generated in a memory access from the channel on the I/O interface. When interruption end information is returned from the central processing unit (CPU) 2 via the interruption end information line 48 and when the completion sequence end information is sent from the I/O interface control circuit 37, the data transfer control circuit 36 completes a series of the page fault processing.

Operations involved in informing the central processing unit (CPU) and input/output device of a protection exception are conducted as explained below.

In Figure 4, the data transfer control circuit 36 is informed of the generation of a protection exception via the protection exception detection line 44 extending from the comparator circuit 25. The data transfer control circuit 36 informs the interrupted CPU determination circuit 41 of the existence of a protection exception interruption request via the protection exception interruption request line 46. The interrupted CPU determination circuit 41 determines the CPU to be interrupted as in the case of the page fault interruption and then indicates a data route in the selector circuit 40. Thereby, the protection exception interruption request is issued to the central processing unit (CPU) designated by the channel (CH)3 through the protection exception interruption request line 46. Operations differ from page fault interruption operations in that no address is transmitted. The data transfer control circuit 36 issues the interruption request to the central processing unit and simultaneously requests the I/O interface control circuit 37 to execute a completion sequence designated by the channel.

The I/O interface control circuit 37 executes the completion sequence which is similar to that to be executed when an error is generated in a memory access from the channel on the I/O interface. Thus, when the interruption end information is returned from the central processing unit (CPU) 2 via the interruption end information line 48 and when completion sequence end information is sent from the I/O interface control circuit 37, the data transfer control circuit 36 completes a series of protection exception processings.

In the above explanation, a channel is taken as an example of a unit which uses a main memory in common with a central processing unit and performs dynamic address translation. However, embodiments of the present invention are not limited only to channel applications and it will be apparent that embodiments of the present invention can be adopted in a unit such as a sub-system adapter, as shown in Figure 1. In this case, the sub-system corresponds to an input/output device.

For an I/O device or sub-system which does not substantially require page fixing, embodiments of this invention can have the following advantages; a channel or sub-system connection unit which operates at a high speed can be provided with economy in hardware, and the following problems can be avoided:

(1) an overhead due to the processing for searching a page to be fixed prior to the execution of the channel programs;

(2) when starting a number of I/O devices,

the number of fixed pages increases and the real memory runs short.

Moreover embodiments of this invention can have the following additional advantages; that page fixing can be executed entirely by means of software not only for an I/O device which does not require page fixing but also for an I/O device which requires page fixing, since the channel invalid bit in the page table entry which is used in the existing case becomes unnecessary, and the overhead for the software processing required for the operation of the channel invalid bit can be eliminated.

Moreover, application of embodiments of this invention in connection with a channel or sub-system adapter of a data processing system in which logical access protection data related to a page corresponding to an entry in the address translation table is provided in the entry in the address translation table allows establishment of a data processing system in which problems present if the channel recognizes the logical address for addressing as in the existing case but must recognise the physical key for the access protection, can be avoided.

Furthermore, embodiments of this invention can provide for a high speed access with less hardware added for the dynamic address translation since it is not required to make an access to the address translation table every time an access is made to the main memory from the channel or sub-system adapter.

Thus, in embodiments of the present invention there is provided a dynamic address translation system in a channel or a sub-system adapter, for which the main memory is used in common with the central processing unit.

In one example of dynamic address translation provided in a channel in a previously proposed data processing apparatus, it is a precondition to fix a page for the access to the main memory from channels.

In a dynamic address translation system in embodiments of the present invention there is provided a register for storing a copy of an entry within an address translation table in the main memory and a bit which indicates effectiveness of such copy, and when the central processing unit has issued an instruction for updating an entry within said address translation table or an instruction for informing alteration of an entry, the bit which indicates effectiveness of said register based thereon is indicated as an invalid bit. In embodiments of this invention, it is not necessary to fix the page in the main memory prior to execution of channel programs.

WHAT WE CLAIM IS:-

1. A dynamic address translation system in an arrangement which includes a central processing unit, a main memory and a channel or sub-system adapter and in which the main memory stores an address translation table for translating logical addresses to physical addresses, wherein said dynamic address translation system is characterized by the provision of a register, in the channel or sub-system adapter, for storing a copy of one entry in the address translation table in the main memory and a bit for indicating validity of the copy, and is operable so that when the central processing unit has issued an instruction for updating an entry within said address translation table or an instruction for advising of alteration of said entry, the bit indicating validity of the copy is set to a value indicating that the copy is not valid, in accordance with such instruction.

2. A dynamic address translation system claimed in the claim 1, wherein the channel or sub-system adapter is provided with a counter for indicating addresses within a memory data unit to which the said one entry corresponds and wherein the system is operable so that when a data transfer request requesting data transfer in data block units is generated from an input/output device or a sub-system connected to said channel or sub-system adapter,

(1) in the case when data transfer is to be started,

(2) in the case when a memory address for making access to the main memory reaches the boundary of said memory data unit, and

(3) in the case when the bit indicating validity of the copy is set to a value indicating that the copy is not valid,

an access is made to said address translation table, the content of one entry among several included in the said address translation table is copied into the register and an access is made to the main memory based on the content of the register, and in other cases no access is made to said address translation table, but an access is made to the main memory by using an address generated from the copy in the register and the counter.

3. A dynamic address translation system claimed in the claim 1, wherein the channel or sub-system adapter is provided with a means for informing the central processing unit when the bit indicating validity of the copy in the register is set to a value indicating that the copy is not valid.

4. A dynamic address translation system

claimed in the claim 1, wherein the channel or sub-system adapter is provided with a means for comparing access protection data included in said one entry and an access key held in said channel or sub-system adapter, and a means for informing the central processing unit when the result of a comparison by the comparing means indicates impossibility of access.

5. A dynamic address translation system substantially as hereinbefore described with reference to the accompanying drawings.

HASELTINE, LAKE & CO.,
Chartered Patent Agents,
Hazlitt House,
28, Southampton Buildings,
Chancery Lane,
London WC2A 1AT
– Also –
Temple Gate House, Temple Gate,
Bristol BS1 6PT
– Also –
9, Park Square, Leeds LS1 2LH,
Yorks.

Printed for Her Majesty's Stationery Office by The Tweeddale Press Ltd., Berwick-upon-Tweed, 1981. Published at the Patent Office, 25 Southampton Buildings, London, WC2A 1AY, from which copies may be obtained.

1595740 COMPLETE SPECIFICATION
3 SHEETS *This drawing is a reproduction of the Original on a reduced scale*
Sheet 1

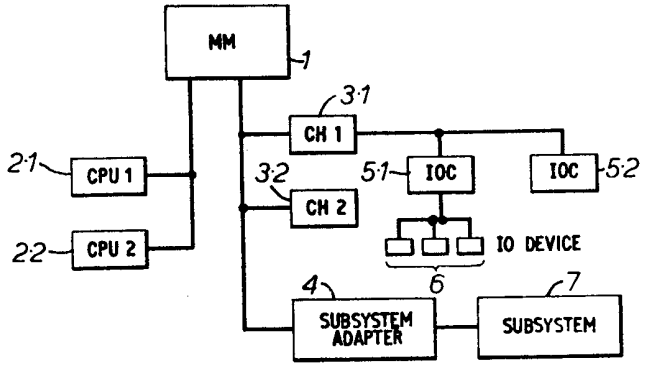

FIG. 1.

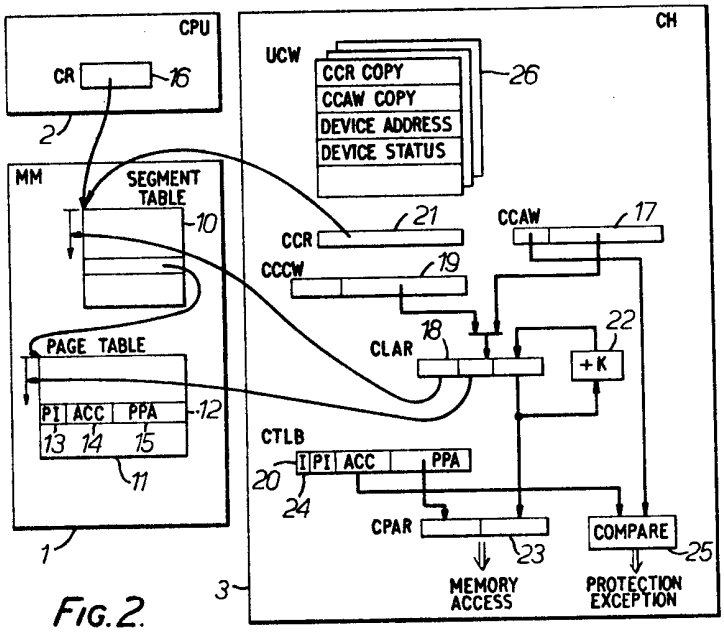

FIG. 2.

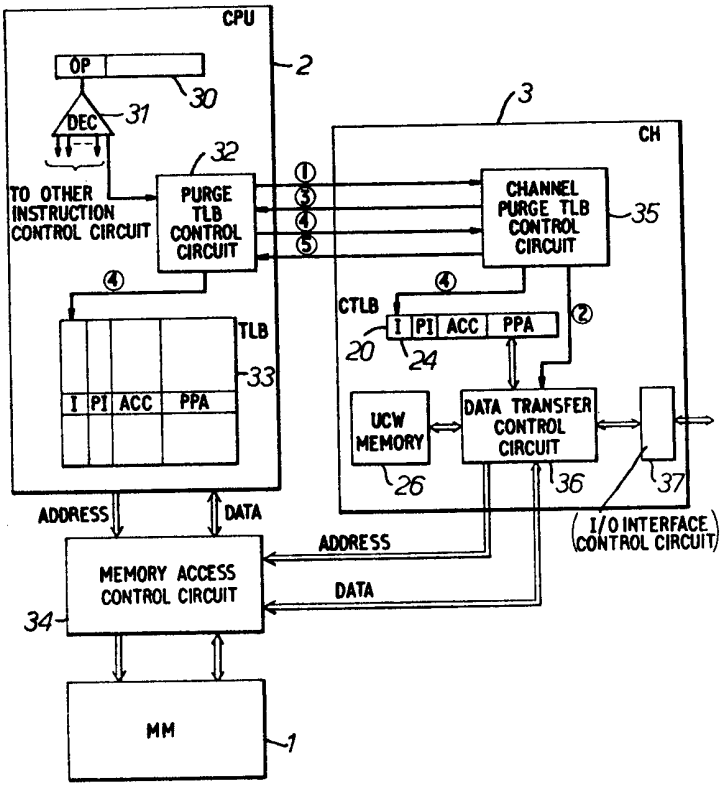
CPU
OP
2
30
DEC
31
3
CH
32
TO OTHER INSTRUCTION CONTROL CIRCUIT
PURGE TLB CONTROL CIRCUIT
CHANNEL PURGE TLB CONTROL CIRCUIT
35
TLB
33
I PI ACC PPA
CTLB
20
24
UCW MEMORY
26
DATA TRANSFER CONTROL CIRCUIT
36
37
I/O INTERFACE CONTROL CIRCUIT
ADDRESS
DATA
MEMORY ACCESS CONTROL CIRCUIT
34
MM
1

FIG. 3.

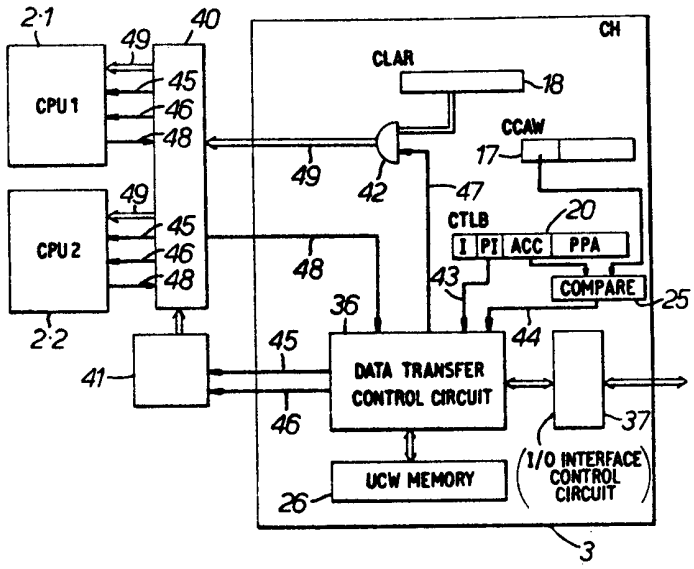
2·1
40
49
CPU1
45
46
48
CPU2
2·2
41
CH
CLAR
18
CCAW
17
49
42
47
CTLB
20
I
PI
ACC
PPA
COMPARE
25
43
36
48
44
45
DATA TRANSFER CONTROL CIRCUIT
46
37
I/O INTERFACE CONTROL CIRCUIT
UCW MEMORY
26
3

FIG. 4.