(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
25 October 2001 (25.10.2001)

PCT

(10) International Publication Number
WO 01/80009 A2

(51) International Patent Classification[7]: G06F 11/16

(21) International Application Number: PCT/US01/12063

(22) International Filing Date: 12 April 2001 (12.04.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/548,528 13 April 2000 (13.04.2000) US

(71) Applicant *(for all designated States except US)*: **STRATUS TECHNOLOGIES INTERNATIONAL, S.A.R.L.** [CH/CH]; Zugerstrasse 76, CH-6340 Baar (CH).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **SOMERS, Jeffrey, S.** [US/US]; 1 Scott Lane, Northboro, MA 01532 (US). **TETREAULT, Mark** [US/US]; 16 Highcrest Park, Webster, MA 01570 (US). **WEGENER, Timothy, M.** [US/US]; 125 West Main Street, Westborough, MA 01581 (US). **HUANG, Wen-Yin** [US/US]; 10 Loosestick Way, Acton, MA 01720 (US).

(74) Agent: **LANZA, John, D.**; Testa, Hurwitz & Thibeault, L.L.P., High Street Tower, 125 High Street, Boston, MA 02110 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

WO 01/80009 A2

(54) Title: FAULT-TOLERANT COMPUTER SYSTEM WITH VOTER DELAY BUFFER

(57) Abstract: A fault-tolerant computer system includes first and second central processing units (CPUs) producing essentially identical data output streams, a voter delay buffer having a first FIFO buffer and a second FIFO buffer, and an I/O module connected to the CPUs. The I/O module includes a comparator for bitwise comparing the CPU data output streams. The first CPU data output stream is transmitted to peripheral devices if both CPU outputs remain substancially identical. Otherwise, if the comparator indicates differences, queued first and second CPU data are routed to the first and second FIFOs respectively , and subsequent data are retained in respective CPU buffers. While the CPUs continue processing, ongoing diagnostic procedures attempt to identify one or the other of the CPUs as malfunctioning and the remaining CPU as correctly-functioning. If the resulting diagnosis is inconclusive, the CPU having the lower rate of error correction is identified as being correctly-functioning. In either case, the buffered output and the subsequently processed data output stream from the correctly-functioning CPU are thereafter transmitted to the peripheral devices.

FAULT-TOLERANT COMPUTER SYSTEM WITH VOTER DELAY BUFFER

Field of the Invention

The present invention is related to fault-tolerant computer systems and, in particular, to a method for efficiently providing reliable operation in a computer system.

Background Information

In most data processing applications, reliable performance of a computer system is critical. To provide for a specified level of reliability, the computer system may include at least one redundant, or backup, central processing unit (CPU), where the CPUs perform the same operations and provide the same data output stream. The input/output (I/O) buses of the CPUs are continually monitored and compared to identify any differences in their respective data streams. If signal differences are detected, a voting device applies predetermined criteria to identify one of the CPUs as malfunctioning. In a redundant computer system having two CPUs, for example, the voting device may identify the CPU having a history of greater cumulative error correction as the malfunctioning CPU. However, experience has shown that this method has an unacceptably low accuracy rate.

The accuracy rate improves with the addition of a second redundant CPU to the computer system. All three CPU outputs are monitored and, when differences are detected, the CPU determined to be malfunctioning is the CPU producing an output not in agreement with the other two CPUs. This approach, however, incurs the additional expense and complexity of integrating the third CPU into the computer system.

Another method used in computer systems having only two redundant CPUs is to have each CPU revert to an idle state and/or lose output data while diagnostic procedures are initiated to determine which CPU is malfunctioning. Based on the results of the diagnostic procedure, one CPU may be identified as malfunctioning. One undesirable side effect of this approach is that the operation of the computer system is impacted and may be severely disrupted while the CPUs are in the idle state.

It is therefore an object of the present invention to provide a computer system achieving a high degree of reliability with a redundant CPU.

It is a further object of the present invention to provide such a computer system in which a malfunctioning CPU can be identified without first placing the CPU into an idle state.

It is a still further object of the present invention to provide such a computer system in which computational data is not lost while the malfunctioning CPU is identified.

It is yet another object of the present invention to provide such a system in which a malfunctioning CPU can be identified with a high degree of reliability. Other objects of the invention will be obvious, in part, and, in part, will become apparent when reading the detailed description to follow.

SUMMARY OF THE INVENTION

The present invention comprises a fault-tolerant computer system which includes a pair of CPUs that produce essentially identical data output streams, a voter delay buffer having first and second FIFO buffers; and an I/O module interconnecting the CPUs and the FIFO buffers. The I/O module compares the data output streams from the two CPUs for differences. If both CPU output streams remain identical, the data output of a selected CPU is transmitted to one or more peripheral devices. Otherwise, if the comparator indicates differences, the data output stream from one CPU is rerouted to the first FIFO, and the data output stream from the other CPU is rerouted to the second FIFO. Meanwhile, the CPUs continue processing operations and ongoing diagnostic procedures identify one of the CPUs as malfunctioning. The FIFOs provide buffering for the data output streams which would otherwise be discarded. Additionally, use of the FIFOs allows the CPUs to continue operation and avoid a disruption to the computer system. If neither CPU is diagnosed as malfunctioning, the I/O module uses data from a priority module to determine which CPU has a higher assigned priority, and identifies the higher-priority CPU as the correctly-functioning CPU. In either case, the computer system then provides the data held in the FIFO associated with the correctly-functioning CPU to the peripheral devices. By thus buffering the data output streams, the present invention allows the computer system to utilize the diagnostic procedures for increasing the probability of correctly identifying a CPU as malfunctioning.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention description below refers to the accompanying drawings, of which:

Fig. 1 is a functional block diagram of a fault-tolerant computer system in accordance with the present invention;

Fig. 2 is a functional block diagram of a CPU in the fault-tolerant computer system of Fig. 1; and

Fig. 3 is a flow diagram illustrating the operation of the fault-tolerant computer system of Fig. 1.

DETAILED DESCRIPTION OF AN ILLUSTRATIVE EMBODIMENT

There is shown in Fig. 1 a fault-tolerant computer system 10 in accordance with the present invention. The computer system 10 includes a first CPU 11 and a second CPU 21. The first CPU 11 and the second CPU 21 are configured to operate in lock-step, or cycle-by-cycle synchronism with one another, as exemplified by a system clock 17. The first CPU 11 includes a maintenance and diagnostic subsystem 15, and the second CPU 21 includes a maintenance and diagnostic subsystem 25. The maintenance and diagnostic subsystems 15 and 25 function to identify and, if possible, correct internal processing errors detected in the operations of the respective CPUs 11 and 21.

The system also includes an I/O module 40 that controls data transfers between the CPUs 11 and 21 and associated peripheral devices (not shown). The first CPU 11 communicates with the I/O module 40 over a first I/O bus 13. Data flowing between the first CPU 11 and the peripheral devices are transmitted over the first I/O bus 13 to the I/O module 40, through the I/O module 40, and to the peripheral devices via a system I/O bus 19. Similarly, data flowing between the peripheral devices and the second CPU 21 is transmitted over a second I/O bus 23 connected to the I/O module 40. It should be understood that the respective data streams on the I/O buses 13 and 23 are essentially identical when both the first CPU 11 and the second CPU 21 are operating error-free.

During normal operation, transient errors may occur within either or both the first CPU 11 and the second CPU 21. Many of the errors are detected and some corrected internally, such as by using error correction logic or parity protocol logic, before transmission over either the first I/O bus 13 or the second I/O bus 23. An ongoing maintenance history as to the occurrence of these transient errors in the first CPU 11 is retained in a first priority register 16. The first priority register 16 is kept updated by the maintenance and diagnostic subsystem 15. Similarly, transient errors occurring in the second CPU 21 are tracked with a second priority register 26 which is kept updated by the maintenance and diagnostic subsystem 25. This maintenance information is made available to a priority module 60 via either a first CPU priority line 61 or a second CPU priority line 62. The priority module 60 includes a software program 63 to assign relative priorities to the two CPUs 11 and 21 based on their relative operational performance parameters. Such statistical data may include, for example, the history of detected transient

errors or the length of time a given CPU has been operating in the computer system 10. These statistical data are used to assign relative priorities to the first CPU 11 and the second CPU 21. These assigned priorities are provided to the I/O module 40.

The I/O module 40 includes a comparator 43 which performs a bit-by-bit cycle compare procedure on the data output streams passing into the I/O module 40 on the I/O buses 13 and 23. The comparator may be, for example, an XOR gate or any other known component for comparing two bit values. If the cycle compare procedure detects a difference between the two data output streams, this may be an indication that one of the CPUs 11 and 21 is malfunctioning. Accordingly, the I/O module 40 responds by issuing a STOP command to both the first CPU 11 and the second CPU 21 over a first command line 41 and a second command line 42 respectively.

When the STOP command is issued, the I/O module 40 stops transmitting output data on the system I/O bus 19 and routes the data output streams on the I/O buses 13 and 23 to a voter delay buffer 50 via a delay buffer line 47. Specifically, the data received from the first CPU 11 is sent to a first FIFO buffer 51, and the data received from the second CPU 21 is sent to a second FIFO buffer 52. This action serves to prevent the peripherals from being sent data which may have been corrupted by the malfunctioning CPU, and also serves to save data which otherwise may have been lost or discarded while the malfunctioning CPU was being identified.

In a preferred embodiment, the maintenance and diagnostic subsystems 15 and 25 continually run their respective diagnostic procedures. It should be understood that, even after the STOP command has been issued to the CPUs 11 and 21, the I/O module 40 continues to forward input data streams sent by the peripheral devices to the CPUs 11 and 21. The CPUs 11 and 21 continue to process the data while running the diagnostic procedures, in accordance with normal operational procedures. The computer system 10 is thus seen by the peripheral devices as functioning normally.

As shown in Fig. 2, the first CPU 11 preferably includes a microprocessor 71, a chipset 73, and a bus interface processor 75. A memory 77 is provided for internal storage of data, as required. The microprocessor 71 receives data from and outputs data to either the memory 77 or the first I/O bus 13 via the chipset 73. Output data to be transmitted by the bus interface processor 75 is held in a buffer 85. When the STOP command is transmitted on the first command line 41 to the bus interface processor 75, the data present in the buffer 85 is retained and not transmitted to the I/O module 40. Because there is finite propagation delay incurred before the STOP signal reaches the bus interface processor 75, some possibly corrupted data may

be sent from the first CPU 11 before the STOP signal is received. This data is sent to the voter delay buffer 50, as described above.

As the first CPU 11 continues its processing and diagnostic operations, output data is retained in the buffer 85. If the buffer 85 becomes full, the bus interface processor 75 sends a BUSY signal to the chipset 73, and further processed data is then stored in a chipset buffer 83. If the chipset buffer 83 becomes full, the data output stream is stored in a microprocessor buffer 81. The output data stored in the buffers 81, 83, and 85 is not output to the peripherals unless the first CPU has been identified as the correctly-functioning CPU, as described in greater detail below. The second CPU 21 (not shown) has an internal configuration similar to that of the first CPU 11, described above, and functions in a similar manner.

Operation of the computer system 10 can best be described with reference to the flow diagram of Fig. 3. The data output streams on the I/O buses 13 and 23 are bit-by-bit compared by the comparator 43, at box 81, to provide a comparative reading from which it can be determined if there are differences between the monitored data output streams. If there are no such differences detected, the comparator 43 continues to monitor the data output streams. If differences are detected, the STOP command is issued, at box 82. Subsequently, the data output streams on the I/O buses 13 and 23 are diverted to the voter delay buffer 50, at step 83.

The first CPU 11 continues executing its ongoing diagnostic procedure, at box 84. If the diagnosis indicates that the first CPU 11 is malfunctioning, the first CPU 11 is isolated, at box 85, and operation of the computer system 10 continues with the second CPU 21. The data stored in the second FIFO buffer 52 is output over the system I/O bus 19, at box 86, and thereafter subsequently processed data from the second CPU 21 is output over the system I/O bus 19. Contemporaneously with the ongoing diagnosis procedure in the first CPU 11, at box 84, the second CPU 21 also continues diagnosis, at box 87. If, on the other hand, the resulting diagnosis indicates that the second CPU 21 is malfunctioning, the second CPU 21 is isolated, at box 88, and operation of the computer system 10 continues with the first CPU 11. The data stored in the first FIFO buffer 51 is output over the system I/O bus 19, at box 89, and subsequent processed data from the first CPU 11 is output over the system I/O bus 19.

If the diagnostic procedures fail to detect problems with either the first CPU 11 or the second CPU 21, the relative CPU priorities are used as the determinative factor. The relative priorities are read to establish which of the first CPU 11 or the second CPU 21 has the higher priority, at box 90. As discussed above, the relative priorities of the CPUs have been determined by one or more criteria, such as their operational histories or the comparative cumulative record

of their internal error corrections. If the second CPU 21 has been assigned the higher priority, for example, the computer system 10 selects the first CPU 11 as the malfunctioning CPU and continues to operate with only the second CPU 21, at box 91. Accordingly, the data stored in the second FIFO buffer 52 is output, at box 92, and so forth. On the other hand, if the first CPU 11 has been assigned the higher priority, the computer system 10 selects the second CPU 21 as the malfunctioning CPU and the operation of the computer system 10 continues with the first CPU 11, at box 91. Subsequently, the data stored in the first FIFO buffer 51 is output, at box 92.

While the invention has been described with reference to particular embodiments, it will be understood that the present invention is by no means limited to the particular constructions and methods herein disclosed and/or shown in the drawings, but also comprises any modifications or equivalents within the scope of the claims.

What is claimed is:

CLAIMS

1. A fault-tolerant computer system suitable for exchanging data with peripheral devices, said computer system comprising:

a first central processing unit (CPU) having at least one first CPU buffer;

a second CPU having at least one second CPU buffer, said second CPU being operationally coupled to said first CPU, such that the output of said second CPU is essentially identical to the output of said first CPU;

a voter delay buffer having a first FIFO buffer and a second FIFO buffer;

an I/O module connected to receive data output streams from said first CPU and said second CPU, said I/O module having,

a comparator for comparing said first CPU data output stream to said second CPU data output stream so as to produce a comparative reading;

transmission means responsive to said comparator, for sending said first CPU data output stream to the peripheral devices, if said comparison reading indicates no difference between said first CPU data output stream and said second CPU data output stream;

and

routing means responsive to said comparator, for routing at least a part of said first CPU data output stream to said first FIFO buffer if said comparison reading indicates a difference between said first CPU data output stream and said second CPU data output stream, and for routing at least a part of said second CPU data output stream to said second FIFO buffer if said comparison reading indicates a difference between said first CPU data output stream and said second CPU data output stream.

2. The computer system of claim 1 further comprising a first diagnostic logic resident in said first CPU and a second diagnostic logic in said second CPU.

3. The computer system of claim 2 wherein said I/O module further comprises identification means responsive to said first diagnostic logic and said second diagnostic logic, for identifying one of said first and second CPUs as malfunctioning.

4. The computer system of claim 3 wherein said transmission means is further responsive to said identification means such that

the contents of said first FIFO buffer is transmitted to the peripheral devices if said second CPU is identified as malfunctioning, or

the contents of said second FIFO buffer is transmitted to the peripheral devices if said first CPU is identified as malfunctioning.

5. The computer system of claim 3 wherein said transmission means is further responsive to said identification means such that

the contents of said first CPU buffer is transmitted to the peripheral devices if said second CPU is identified as malfunctioning, or

the contents of said second CPU buffer is transmitted to the peripheral devices if said first CPU is identified as malfunctioning.

6. The computer system of claim 1 further comprising:

a priority module for receiving first error correction information from said first CPU and second error correction information from said second CPU; and

priority logic for assigning relative priorities to said CPUs, said assigned relative priorities being determined as a function of said first and second error correction information.

7. The computer system of claim 6 wherein said priority logic assigns a higher priority to selected one of said first and second CPUs if the indicated error rate in said correction information corresponding to said selected CPU is less than the indicated error rate in said correction information corresponding to the other one of said first and second CPUs.

8. The computer system of claim 6 wherein said priority logic assigns a higher priority to a selected one of said first and second CPUs if said selected CPU has been operating in said computer system for a greater length of time than the length of time the other one of said first and second CPUs has been operating in said computer system.

9. The computer system of claim 6 wherein said transmission means is further responsive to said priority logic such that

the contents of said first FIFO buffer is transmitted to the peripheral devices if said first CPU has been assigned a higher said relative priority, or

the contents of said second FIFO buffer is transmitted to the peripheral devices if said second CPU has been assigned a higher said relative priority.

10. The computer system of claim 6 wherein said transmission means is further responsive to said priority logic such that

the contents of said first CPU buffer is transmitted to the peripheral devices if said first CPU has been assigned a higher said relative priority, or

the contents of said second CPU buffer is transmitted to the peripheral devices if said second CPU has been assigned a higher said relative priority.

11. A method for reliably exchanging data between peripheral devices and a computer system having a first CPU with a buffer operating in lock-step with a second CPU with a buffer, said method comprising the steps of:

comparing a data output stream from the first CPU with a contemporaneous data output stream from the second CPU to obtain a comparative reading;

transmitting said first CPU data output stream to the peripheral devices if said comparative reading indicates no difference between said first CPU data output stream and said second CPU data output stream; and

transmitting at least a part of said first CPU data output stream to a first FIFO buffer if said comparative reading indicates a difference between said first CPU data output stream and said second CPU data output stream, and

transmitting at least a part of said second CPU data output stream to a second FIFO buffer if said comparative reading indicates a difference between said first CPU data output stream and said second CPU data output stream.

12. The method of claim 11 further comprising the step of executing contemporaneous respective diagnostic procedures in the first CPU and in the second CPU.

13. The method of claim 12 further comprising the steps of:

transmitting the contents of said second FIFO to the peripheral devices if said diagnostic procedures indicate the first CPU to be malfunctioning; and

transmitting the contents of said first FIFO to the peripheral devices if said diagnostic procedures indicate the second CPU to be malfunctioning.

14. The method of claim 13 further comprising the steps of:

isolating the first CPU if said diagnostic procedures indicate the first CPU to be malfunctioning; and

isolating the second CPU if said diagnostic procedures indicate the second CPU to be malfunctioning.

15. The method of claim 11 further comprising the steps of:

accessing a first error correction history for the first CPU;

accessing a first error correction history for the second CPU;

if said error correction histories indicate that the second CPU has a higher error correction rate than the first CPU, assigning a higher priority to the first CPU; and

if said error correction histories indicate that the first CPU has a higher error correction rate than the second CPU, assigning a higher priority to the second CPU.

16. The method of claim 15 further comprising the steps of:

transmitting the contents of said first FIFO to the peripheral devices if the first CPU has been assigned a higher priority; and

transmitting the contents of said second FIFO to the peripheral devices if the second CPU has been assigned a higher priority.

17. The method of claim 12 further comprising the steps of:

retaining at least a second portion of said first CPU data output stream in the first CPU buffer if said diagnostic procedures indicate the first CPU to be malfunctioning; and

retaining at least a second portion of said second CPU data output stream in the second CPU buffer if said diagnostic procedures indicate the second CPU to be malfunctioning.

18. The method of claim 17 further comprising the steps of:

transmitting the contents of said first CPU buffer to the peripheral devices if said diagnostic procedures indicate the second CPU to be malfunctioning; and

transmitting the contents of said second CPU buffer to the peripheral devices if said diagnostic procedures indicate the first CPU to be malfunctioning.

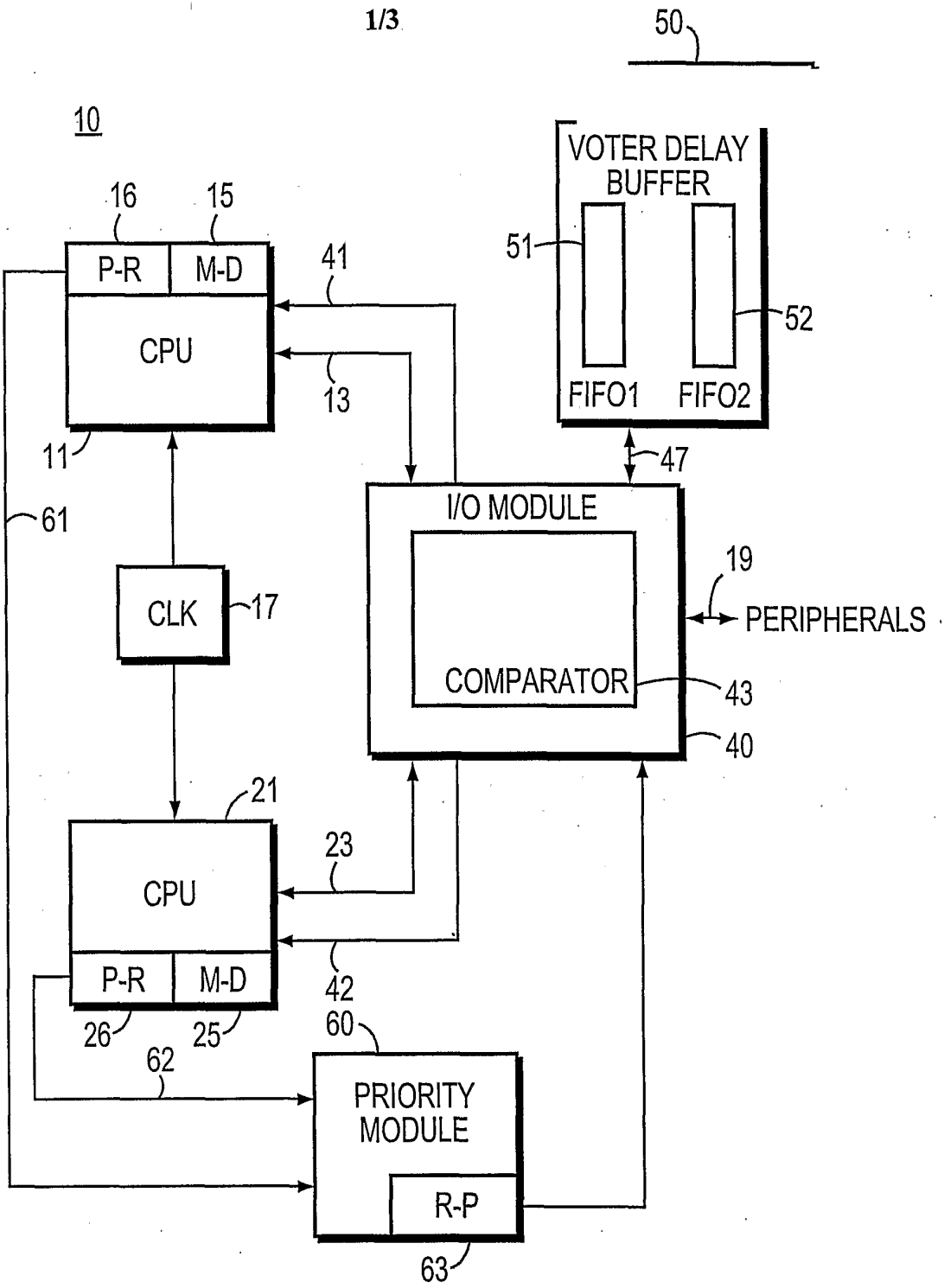
50
10
VOTER DELAY BUFFER
16
15
P-R
M-D
41
51
52
CPU
13
FIFO1
FIFO2
11
47
61
I/O MODULE
19
CLK
17
PERIPHERALS
COMPARATOR
43
40
21
23
CPU
P-R
M-D
42
26
25
60
62
PRIORITY MODULE
R-P
63

FIG. 1

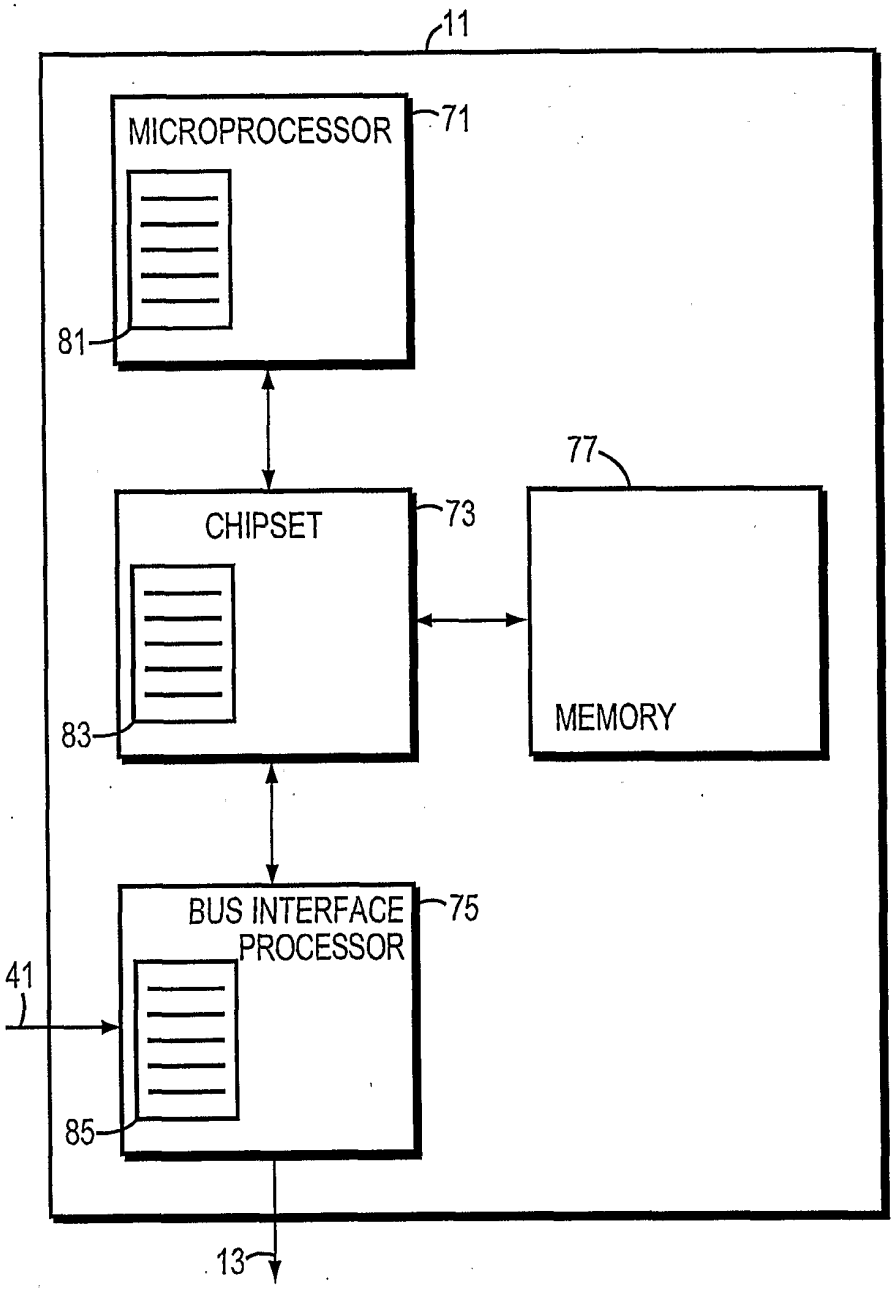
11
MICROPROCESSOR
71
81
77
CHIPSET
73
MEMORY
83
BUS INTERFACE
PROCESSOR
75
41
85
13

FIG. 2

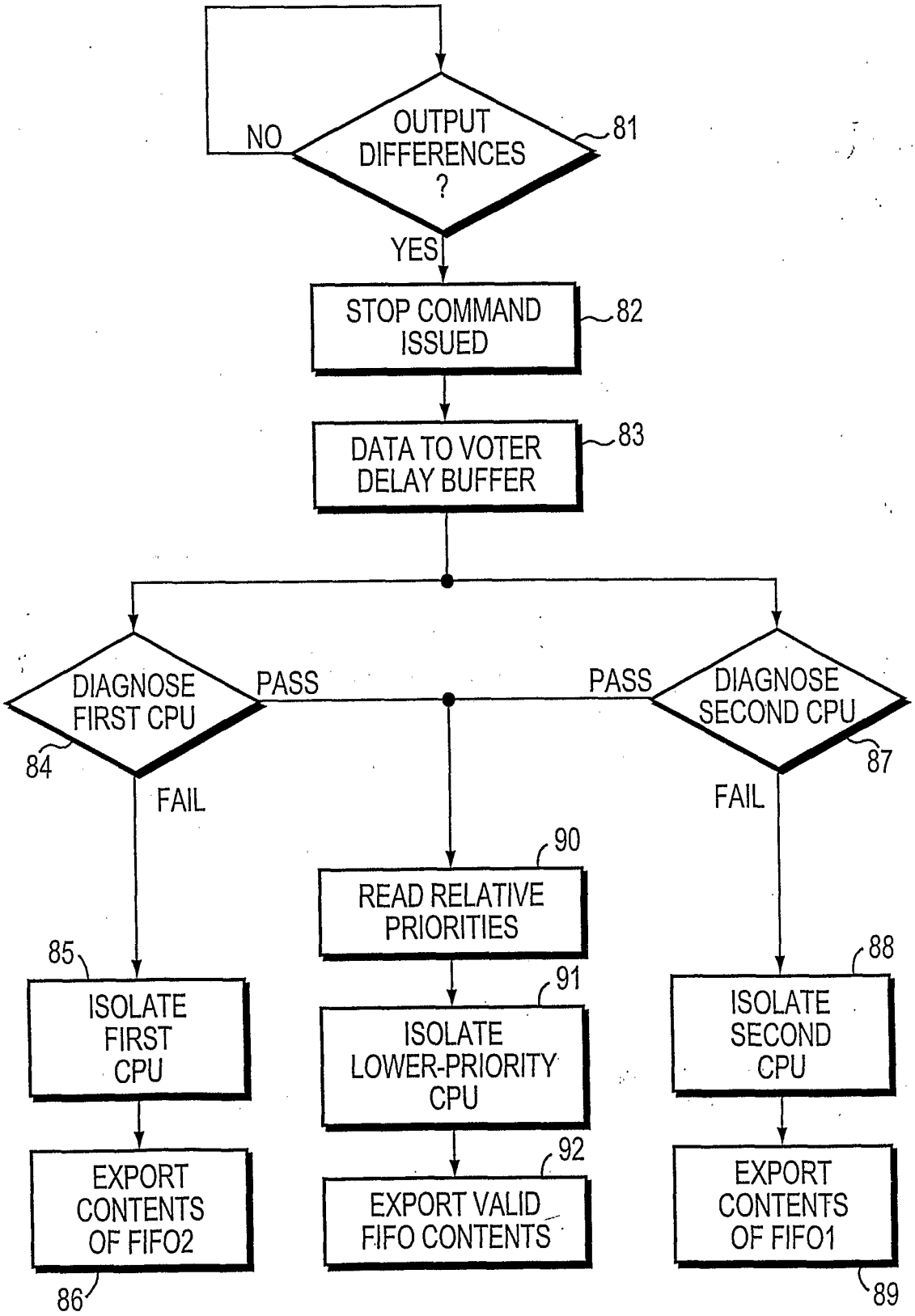

FIG. 3