PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 3: G06F 15/16	A1	(11) International Publication Number: WO 82/01095
		(43) International Publication Date: 1 April 1982 (01.04.82)

(21) International Application Number: PCT/US81/01176

(22) International Filing Date: 2 September 1981 (02.09.81)

(31) Priority Application Number: 190,085

(32) Priority Date: 23 September 1980 (23.09.80)

(33) Priority Country: US

(71) Applicant: WESTERN ELECTRIC COMPANY, INC. [US/US]; 222 Broadway, New York, NY 10038 (US).

(72) Inventor: LARSON, Allen, Leonard; 11277 North Birch Drive, Thornton, CO 80233 (US).

(74) Agents: HIRSCH, A., E., Jr. et al.; Post Office Box 901, Princeton, NJ 08540 (US).

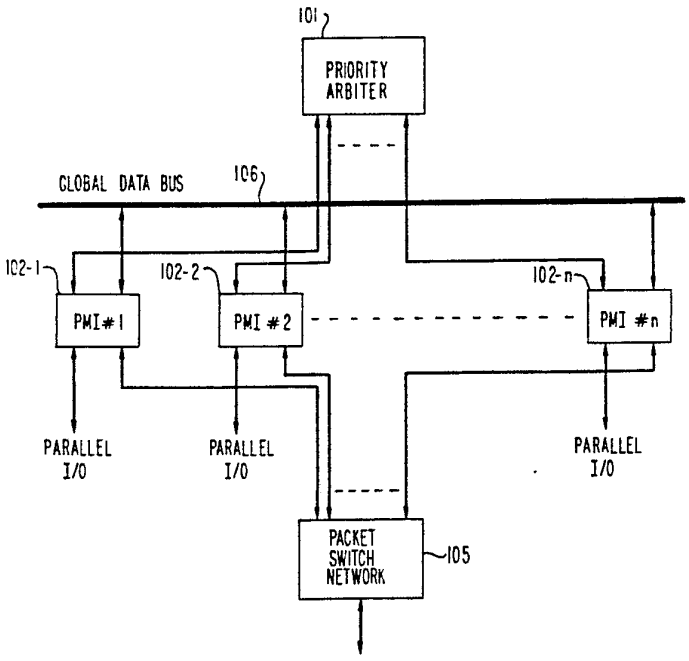
(81) Designated States: DE (European patent), FR (European patent), JP, SE (European patent).

Published
With international search report

(54) Title: PROCESSOR INTERCONNECTION SYSTEM

(57) Abstract

Prior art multiprocessor systems either employ a single set of common resources that are shared by all the processors, or employ a plurality of resources, each of which are dedicated to a particular processor. In this latter situation, serial intercommunication links requiring both the transmitting and the receiving processors to participate are used to transmit data between processors. The subject processor interconnection system interconnects n independent processors (209-1 to 209-n), each of which has associated therewith a plurality of resources (205-i to 207-i), which resources are connected to the processor by a local bus (214-1 to 214-n), such that any processor in the system has direct access to any system resource even though such a resource may be associated with another processor.



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	**KP**	Democratic People's Republic of Korea
AU	Australia	**LI**	Liechtenstein
BR	Brazil	**LU**	Luxembourg
CF	Central African Republic	**MC**	Monaco
CG	Congo	**MG**	Madagascar
CH	Switzerland	**MW**	Malawi
CM	Cameroon	**NL**	Netherlands
DE	Germany, Federal Republic of	**NO**	Norway
DK	Denmark	**RO**	Romania
FI	Finland	**SE**	Sweden
FR	France	**SN**	Senegal
GA	Gabon	**SU**	Soviet Union
GB	United Kingdom	**TD**	Chad
HU	Hungary	**TG**	Togo
JP	Japan	**US**	United States of America

PROCESSOR INTERCONNECTION SYSTEM

Field of the Invention

This invention relates to a multiprocessor computer system comprising a plurality of processors, each of the processors has associated therewith a plurality of resources, a plurality of local bus circuits, each associated on a one-to-one basis with the processors, each of the local bus circuits connects the plurality of resources to the associated processor.

Description of the Prior Art

Prior art multiprocessor systems either employ a single set of common resources that are shared by all the processors, or employ a plurality of resources, each of which are dedicated to a particular processor. In this latter situation, serial intercommunication links are typically used to transmit data between processors. Both the transmitting and the receiving processors are required to participate in this data transfer and this unnecessarily burdens the two processors.

Summary of the Invention

The problem is solved in a multiprocessor computer system in which the computer system further comprises an intercommunication circuit connected to all of the processors, wherein the intercommunication circuit provides any requesting one of the processors with direct access to resources which may be associated with another selected one of the processors. Each processor has a local bus which connects the processor to its associated resources, which resources may be memory devices, I/O ports, serial communication devices, etc. Each processor also has memory management circuitry, interrupt systems and the other standard elements which are normally associated with a processor.

The multiprocessor computer system operates in this environment by using a global bus to interconnect all

the processors in the system. Each processor generates a virtual address when requesting access to a resource and the memory management circuit associated with the processor translates this virtual address into an actual hardware resource address. If the resource requested is associated with another processor, the processor interconnection system directly connects the requesting processor, via the global bus, to the local bus to which the resource requested is connected. Thus, each processor can directly access all resources in this multiprocessing environment.

Brief Description of the Drawing

FIG. 1 illustrates the preferred embodiment of the processor interconnection system in block diagram form;

FIG. 2 illustrates the details of a typical Processor-Memory-Interface Module shown in FIG. 1; and

FIGS. 3 and 4 illustrate typical applications of the subject system.

Detailed Description

The disclosed invention comprises a processor interconnection system which can be used to build a multiprocessor computer system from a number of independent or quasi-independent processor-based modules. Each module contains a local data bus that connects the processor to its associated memory and various interface circuits. Each such processor-memory-interface (PMI) module, therefore, will be a small computer in its own right. The subject processor interconnection system serves to combine a multitude of these small computers to form an efficient multiprocessing environment.

Such a system is shown in FIG. 1, and includes a priority arbiter module 101, a packet switch network 105, and a number (n) of PMIs (102-1 to 102-n). A global data bus 106 interconnects the several PMIs. The priority arbiter 101 controls access to global data bus 106 and resolves priority of access to global data bus 106 among the PMIs according to any of several conventional formulae.

The packet switch network 105 serves as an interface between any pair of serial data links including those from outside of the system as well as those within the PMIs. Central to the invention is the PMI module which is shown in detail in FIG. 2.

Each PMI assembly includes a processor 209-1, global bus access circuits 202-1 to 204-1, 208-1, 215-1 accessible local resources 205-1, 206-1, 207-1, and a local data bus 214-1 which connects local resources 205-1 to 207-1 to global bus 213 via bus connect circuit 202-1 or to processor 209-1 via local buffer 216-1 and memory management circuit 208-1. Only a real time clock 212-1 and an interrupt system 210-1 are dedicated solely to processor 209-1 and, therefore, these circuits are not connected to either bus. The global access circuitry within each assembly include a memory management circuit 208-1, bus connection circuit 202-1, and local and global bus arbiters 204-1, 203-1. Other resources contained in each assembly are: one or more memory modules 205-1, one or more parallel input/output ports 206-1, and one or more serial communication interfaces 207-1. Both local and global busses 213 and 214-1 are considered, for illustrative purposes, to carry parallel data.

Bus Selection

Although a given processor 209-1 normally uses the resources 205-1 to 207-1 connected to its local bus 214-1, processor 209-1 does not discriminate between busses. Bus selection is the function of the memory management module 208-1 which works as follows.

When seeking information, processor 209-1 requests the contents of a virtual memory address. The memory management module 208-1 consults an internal address mapping memory which may be realized by either Read Only Memory (ROM) or Random Access Memory (RAM). Whether its contents are fixed or dynamically loaded, however, each mapping memory holds translations of virtual memory addresses to physical memory addresses. Using these data,

the memory management circuit 208-1 can accept a virtual memory address from its dedicated processor 209-1 and thereby access the physical memory address sought.

Each processor 209-1 normally executes software instructions located in its local memory 205-1, but it can access software instructions in any of the memory modules in the system. So data called for by dedicated processor 209-1 outputting a virtual memory address may be either in the local memory 205-1 or in a remote memory 205-i which is its opposite number in another (i^{th}) PMI module 202-i. The first step taken by the local memory management circuitry 208-1 is to find the physical address in its mapping memory that is the translation of the requested virtual memory address.

If the physical address is in the local memory 205-1, memory management 208-1 signals the local/remote arbiter 204-1 via lead SELECT that the processor 209-1 needs access to the local bus 214-1. The local/global selection line, lead SELECT, from memory management 208-1 to the global arbiter 203-1 and the local/remote arbiter 204-1 carries only a single bit, so one or the other is always selected and a change of state calls for a change of bus access. Thus, in the present situation, when the local bus 214-1 is free, the local/remote arbiter 204-1 enables the local buffer 216-1, thereby connecting processor 209-1 to the local bus 214-1 by way of memory management 208-1.

If the physical address generated is of a remote memory 205-i which is the opposite number of 205-1 in module 202-i, memory management 208-1 signals global arbiter 203-1 via lead SELECT that processor 209-1 needs access to global bus 213. The global arbiter 203-1 then notifies the priority arbiter 201 via lead REQUEST-1. The priority arbiter 201 only decides which of several competing PMIs will get the global bus 213 at a given time. When access to the global bus 213 is granted by the priority arbiter 201, priority arbiter 201 transmits a bus

access signal to global arbiter 203-1. In response to this bus access signal, the global arbiter 203-1 enables the global buffer 215 via lead CONTROL G to connect memory management 208-1 to the global bus. Memory management 208-1 then requests the resources of the remote PMI 202-i by placing the hardware address on leads DATA G, which address is placed on global bus 213 by global buffer 215-1.

At the addressed PMI (202-i), the local/remote arbiter 204-1 receives any request for access from the global bus 213. If the addressed local bus 214-i is free, the local/remote arbiter 204-i immediately grants access. If the local bus 214-i is busy, access is denied until it is free. The local/remote arbiter 204-i then grants access to the processor with the highest priority. Any processor can lock the bus it is currently using until completion of a semaphore operation.

Alternate Communication Methods

Besides the global and local busses 213 & 214-1, each processor 209-1 has two other communication means. These are the serial communication interface 207-1 and parallel data input/output port 206-1, means that are known to the art and are included here to complete the illustration of this invention. Each local serial communication interface 207-1 connects to a packet switch network 211 which in turn distributes serial data to external serial links and to other serial communication interface modules opposite numbers of 207-1 systemwide. By way of the parallel input/output ports 206-1, the processors can control widely diverse peripheral devices. Thus, the several serial communications interfaces 207-1 combined with the packet switch network 211 comprise a serial data communication subsystem, and the input/output ports 206-1 combined with the dual bus system comprise a parallel data communication subsystem.

Description of a Typical Application

FIG. 3 shows a system composed of a number (n) of PMIs. The illustrated PMIs are connected to function in several ways. One PMI 301 of FIG. 3 connects the system to any of several kinds of system or network: a processor network 308 is the example given in FIG. 3. Another PMI 302 joins a terminal 305 to the system for the operator's use. The program in the memory of PMI 302 arranges the output of the terminal so that it can communicate in the language of the PMIs, thus creating a virtual terminal for system use. So given a programmable memory, PMI 302 would make it possible for the system to use a wide variety of terminals with no change of hardware. The third PMI 303 of FIG. 3 is shown controlling a small set of telephones 306 and the associated telephone switching network 309. Finally, PMI 30n is depicted as an interface between the system and a mass storage unit 307, such as a disc or tape drive.

Since each local processor has dedicated I/O and memory, each PMI amounts to a separate computer and can execute an operating system program. Each PMI also includes a dedicated real time clock and interrupt system connected to the processor so that PMIs can be operated in logical parallelism. Because of the speed of the subject dual bus system, this distributed computer system compares quite favorably with a centrally operated system.

An operation can be performed by a system such as the subject system regardless of the limitations of any given component processor. Assume, for example, that a function were called for by the operator of terminal 305. PMI 302 would then use the subject dual bus system to call for the required program from mass storage unit 307. According to a protocol given in the program, PMI 302 might send subroutines to the memory modules of several other PMIs according to the computing power available to each and regardless of their nominal functions. Finally, PMI 302 might order subroutines and supply data as needed until the

requested function is performed.

Description of an Alternate Application

FIG. 4 reveals a Private Branch Exchange PBX telephone system organized as taught by the subject invention. This embodiment comprises switching network 404 and port circuits 406, 407 controlled by a network PMI 403 and scanner PMIs 405, 409 supplemented by feature processing PMI 401, console 402, mass storage devices 412, mass storage control PMI 411 and a maintenance PMI 413. Each PMI in this system would be configured as illustrated in FIG. 2 wherein each PMI has a local bus 214-i, processor 209-i, etc.

Operating features available to the system depend on two things: the configuration of the hardware and the contents of the feature PMI's memory. The system operator uses the console 402 to call up routines from the feature PMI 401. Because of the characteristics of the subject system, features can be added or subtracted by programming the feature PMI and, if necessary, adding or removing PMIs.

Port circuits are monitored by the scanner PMI e.g. 405 for such signals as "off-hook or on-hook" from its associated telephones e.g. 408-1 in response to which it can connect or disconnect them from the ports by use of port selection means within the port circuits e.g. 406. A scanner PMI 405 or 409 can then use the dual bus system comprising busses 400 and 214-i (wherein 214-i is the local bus of the scanner) to find from the feature PMI 401 what privileges are assigned to a calling telephone, for example, whether it is allowed to call long distance.

Dialing signals into the switching network 404 from the port circuits 406 or 407 are interpreted by the network PMI 403 and used as data on which to base connection from the calling port to the addressed port. Before connection, the dual bus system may be used by the network PMI 403 to determine from the feature PMI 401 what privileges accrue to the telephone being called.

Requests for data beyond that contained in the memories of the several PMIs are referred to the mass storage device 412 by way of the dual bus system and the mass store control PMI 411.

Because it interacts with all system components, a description of maintenance PMI 413 operation will be used to illustrate how the subject embodiment works. In the course of ordinary system operation, maintenance PMI 413 constantly monitors the global bus 213 for commands and responses among the system PMIs. Within PMI 413, these messages are routed through bus connect 202-1 (see FIG. 2) and memory management 208-1 to the processor 209-1. Processor 209-1 then compares these command/response sets read from the global bus to master patterns in memory 205-1. If the messages from the global bus match these ideal patterns, the system is known to be working. Besides these online test routines, the maintenance PMI 413 can interrupt system operation for offline routines.

According to a programmed schedule, for example, it can transmit commands on the global bus 213 to the other PMIs. These commands are transmitted by the means described in the paragraphs on bus selection. Then maintenance PMI 413 can compare the response to these commands to master patterns stored in its memory 205-1. It can also perform any or all of these offline system tests upon orders entered by an operator on system console 402.

In case of an apparent malfunction, therefore, the operator can invoke the resources of maintenance PMI 413. Such a command is routed from the operator's console 402 through feature PMI 401 and the dual bus structure to maintenance PMI 413. In response to the said command, maintenance PMI 413 selects appropriate tests according to a test procedure in its internal memory and applies them to the system as described above. Having received suspect responses from the system, the maintenance processor 209-1 analyzes the errata to determine the source of the malfunction. It then searches mass storage device

412 by way of mass store control PMI 411 and the dual bus structure for a program designed to deal with the bad component.

Assume, for the sake of illustration, that the processor has failed in PMI 409 and consequently that no connection can be made to half the telephones 410-1 to 410-n in the system. Using the repair program mentioned in the last paragraph, maintenance PMI 413 might isolate defective processor 209-i (see FIG. 2) by instructing memory management module 208-i to pass no data from defective processor 209-i to either bus. Maintenance PMI 413 might then distribute scanning duties among the remaining PMIs according to their capacities. The substitute PMIs would use the global data bus 213 both to access data from scanner 409's memory 205-i and to command the port circuits 407 via scanner 409's internal input/output port 206-i. Thus, the system would be made temporarily functional although at a slower speed than before. Because of the high speed of the subject dual bus system, however, the repaired system would run much faster than a similarly repaired distributed system of the previous art. Having confirmed proper operation of the system, the maintenance PMI 413 would send repair instructions to the operator via the dual bus structure and modules 401 and 402. So programmable maps, privileges, and priorities will have made the temporary repair possible without hardware changes.

If troubleshooting problems cannot be handled within the system, they can be referred to a remote expert. This troubleshooter can connect a terminal or computer to the resources of maintenance module 413 by means of a telephone line to the remote maintenance link 414. The remote maintenance link 414 connects to the system by means of the input/output port 206-1 in the maintenance PMI 413. Thus, all tests and results available to module 413 are available for outside use and module 413 can be programmed from anywhere.

While a specific embodiment of the invention has been disclosed, variations in structural detail, within the scope of the appended claims, are possible and are contemplated. There is no intention of limitation to what is contained in the abstract or the exact disclosure as herein presented. The above-described arrangements are only illustrative of the application of the principles of the invention. Normally, other arrangements may be devised by those skilled in the art without departing from the spirit and the scope of the invention.

Claims

1. A multiprocessor computer system comprising:

a plurality of processors (209-1 to 209-n), each of the processors (209-1 to 209-n) has associated therewith a plurality of resources (205-i to 207-i),

a plurality of local bus circuits (214-1 to 214-n), each associated on a one-to-one basis with the processors (209-1 to 209-n), each of the local bus circuits connects the plurality of resources (205-i to 207-i) to the associated processor (209-i); and

CHARACTERIZED IN THAT

the computer system further comprises:

an intercommunication circuit (200) connected to all of the processors (209-1 to 209-n), wherein the intercommunication circuit (200) provides any requesting one of the processors (209-1 to 209-i) with direct access to resources (205-i) which may be associated with another selected one of the processors (209-i).

2. A multiprocessor computer system in accordance with claim 1

CHARACTERIZED IN THAT

the intercommunication circuit (200) comprises:

a global bus circuit (213) connected to all of the processors (209-1 to 209-n) for carrying data signals therebetween;

a priority arbiter (201) connected to all of the processors (209-1 to 209-n) and responsive to the busy/idle status of the global bus circuit (213) for regulating the access of the requesting processor (209-1) to the global bus circuit (213); and

a plurality of memory management circuits (208-1 to 208-n) associated on a one-to-one basis with each of the processors (209-1 to 209-n), wherein each of the memory management circuits (208-1) interconnects the associated processor (209-1) with the global bus circuit (213) when the associated processor (209-1) requests access to the resources (205-i) associated with another selected one of

the processors (209-i).

3. A multiprocessor computer system in accordance with claim 2

CHARACTERIZED IN THAT

each of the memory management circuits (208-1) is responsive to a virtual memory address generated by the associated processor (209-1) for mapping the virtual memory address into a resource address; and

wherein each of the memory management circuits (208-1) is responsive to a resource address which identified a resource (205-i) not connected to the associated local bus (214-1) for generating a global bus circuit request and for applying the global bus circuit request to the priority arbiter (201).

4. A multiprocessor computer system in accordance with claim 3

CHARACTERIZED IN THAT

the priority arbiter (201) is responsive to the global bus circuit request for generating a bus access signal when the global bus circuit (213) is idle and for applying the bus access signal to the memory management circuit (208-1) generating the global bus circuit request;

wherein the last mentioned memory management circuit (208-1) is responsive to the bus access signal for applying the resource address to the global bus circuit (213) thereby signalling the selected processor (209-i) of the request.

5. A multiprocessor computer system in accordance with claim 4

CHARACTERIZED IN THAT

the intercommunication circuit (200) comprises:

a plurality of bus connect circuits (202-1 to 209-n) associated on a one-to-one basis with the processors (209-1 to 209-n) for directly connecting the global bus circuit (213) with the local bus circuit (214-i) associated with the selected processor (209-i) for providing access from the global bus circuit (213) to the resources (205-i

to 207-i) associated with the selected processor (209-i) via the associated local bus circuit (214-i).

6. A multiprocessor computer system in accordance with claim 5

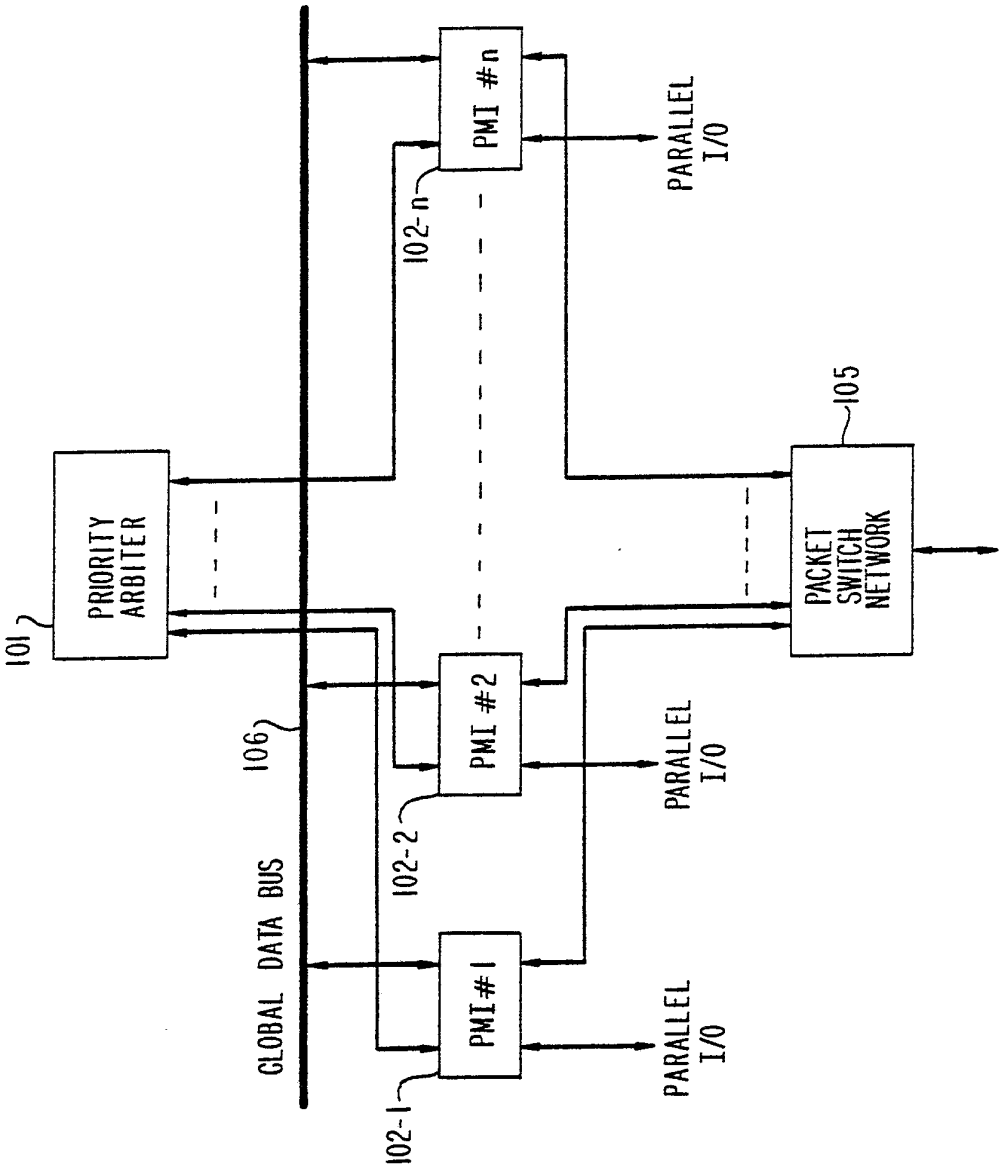
CHARACTERIZED IN THAT

the intercommunication circuit (200) comprises:

a plurality of local arbiter circuits (204-1 to 204-n) associated on a one-to-one basis with the processors (209-1 to 209-n), wherein each of the local arbiter circuits (204-i) is responsive to the application of the resource address to the global bus circuit (213) by the requesting memory management circuit (208-1) for comparing the resource address to the addresses of the resources (205-i to 207-i) connected to the associated local bus circuit (214-i) and for generating a receive enable signal when the resource address matches the address of one of the plurality of resources (205-i) connected to the local bus circuit (214-i); and

wherein the bus connect circuit (202-i) is responsive to the receive enable signal for directly connecting the global bus circuit (213) to the associated local bus circuit (214-i).

FIG. 1



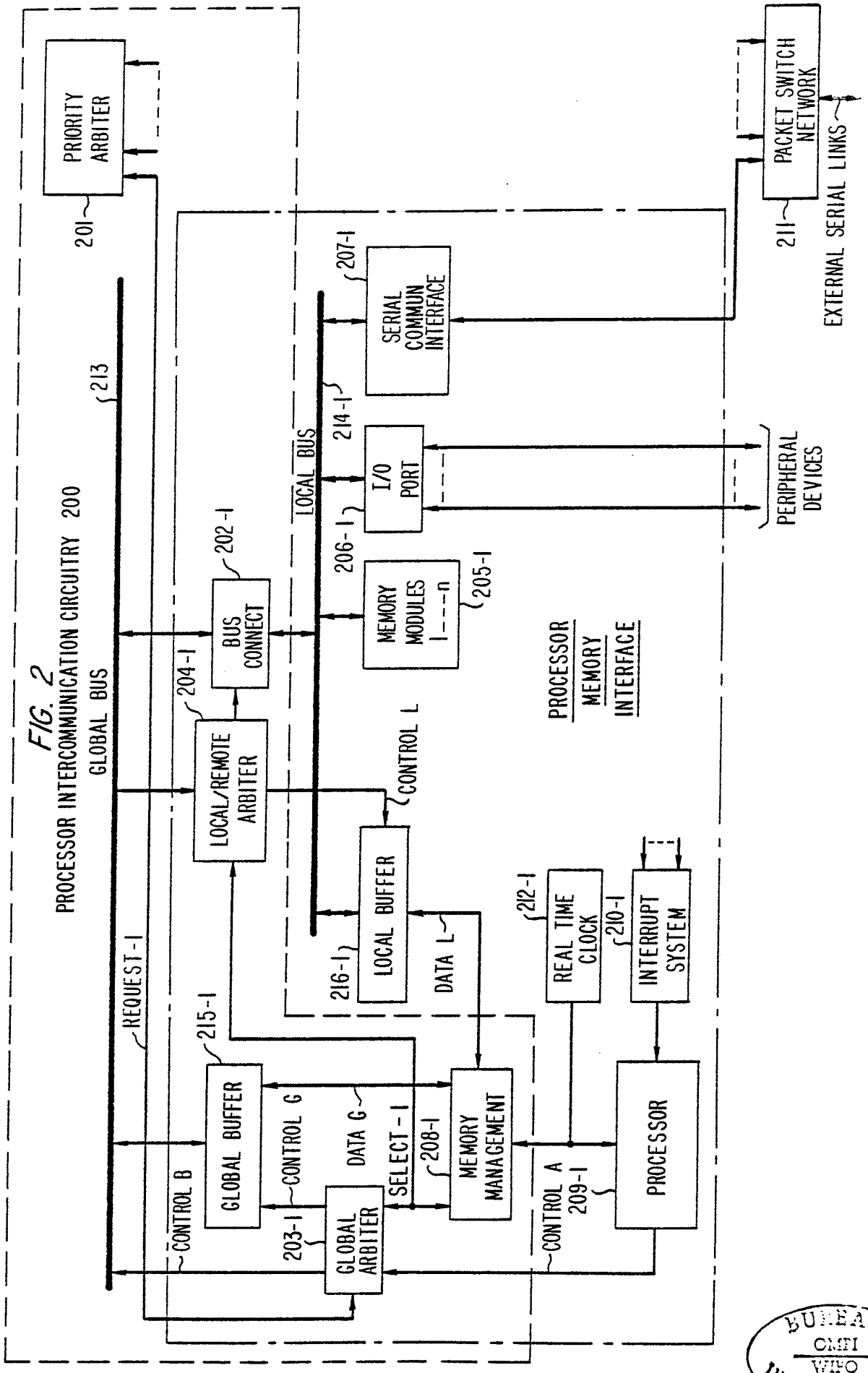
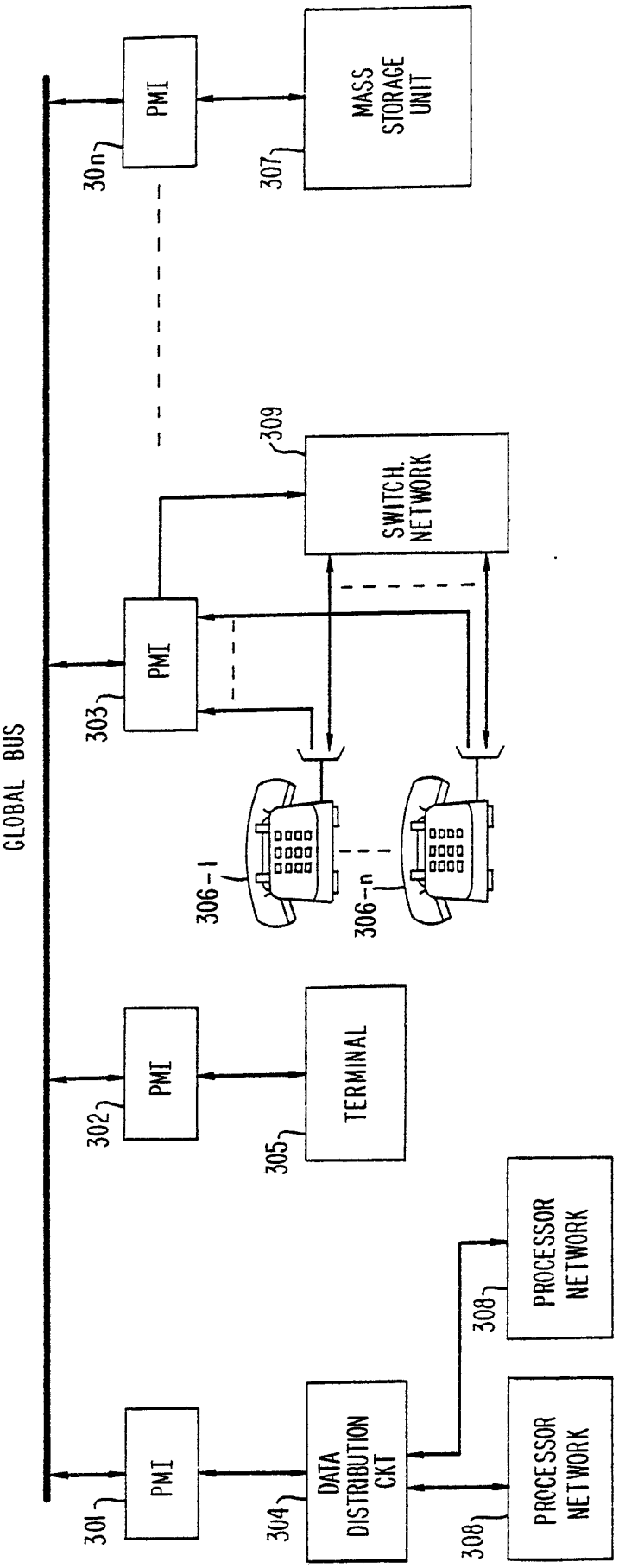

PROCESSOR INTERCOMMUNICATION CIRCUITRY 200
GLOBAL BUS
213
PRIORITY ARBITER
201
PACKET SWITCH NETWORK
211
EXTERNAL SERIAL LINKS
SERIAL COMMUN INTERFACE
207-1
LOCAL BUS
214-1
I/O PORT
206-1
PERIPHERAL DEVICES
MEMORY MODULES 1---n
205-1
BUS CONNECT
202-1
204-1
LOCAL/REMOTE ARBITER
CONTROL L
PROCESSOR MEMORY INTERFACE
LOCAL BUFFER
216-1
DATA L
REAL TIME CLOCK
212-1
INTERRUPT SYSTEM
210-1
REQUEST-1
215-1
GLOBAL BUFFER
CONTROL G
DATA G
SELECT-1
208-1
MEMORY MANAGEMENT
CONTROL B
CONTROL A
209-1
PROCESSOR
203-1
GLOBAL ARBITER

FIG. 2

FIG. 3



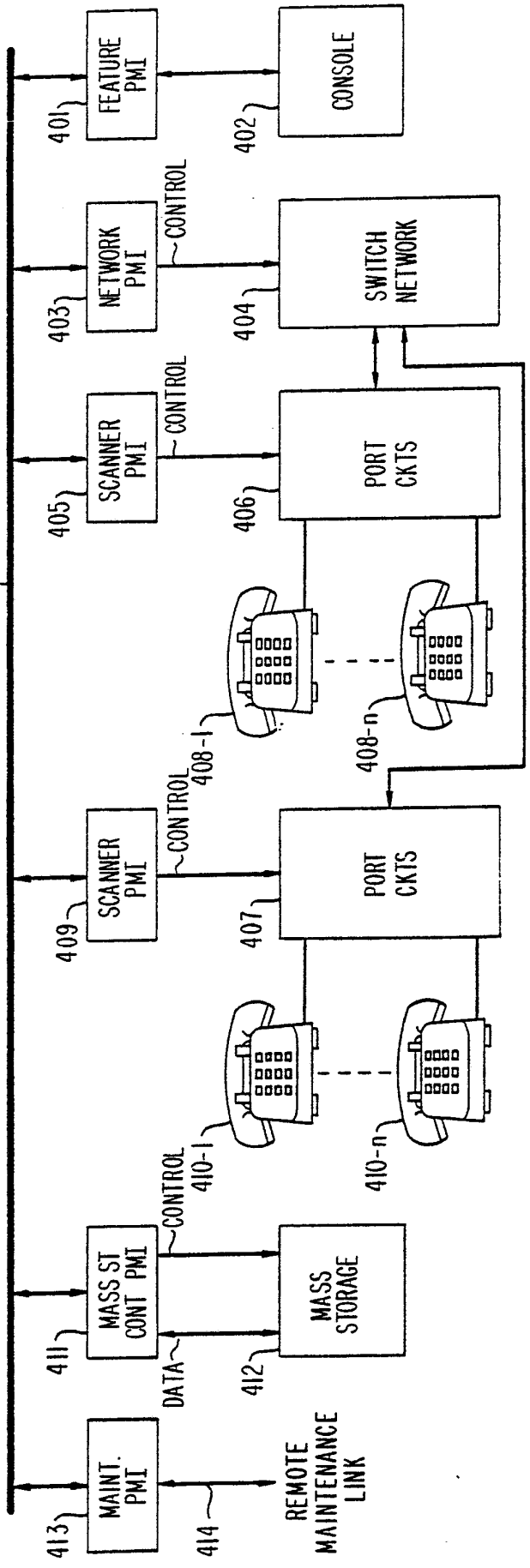

GLOBAL BUS
400
FEATURE PMI
401
CONSOLE
402
NETWORK PMI
403
CONTROL
SWITCH NETWORK
404
SCANNER PMI
405
CONTROL
PORT CKTS
406
408-1
408-n
SCANNER PMI
409
CONTROL
PORT CKTS
407
410-1
410-n
CONTROL
MASS ST CONT PMI
411
DATA
MASS STORAGE
412
MAINT. PMI
413
414
REMOTE MAINTENANCE LINK

FIG. 4

INTERNATIONAL SEARCH REPORT

International Application No PCT/US81/01176

I. CLASSIFICATION OF SUBJECT MATTER (if several classification symbols apply, indicate all) [3]

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

U.S. CL. 364/200

INT. CL.[3] G06F 15/16

II. FIELDS SEARCHED

Minimum Documentation Searched [4]

Classification System	Classification Symbols
U.S.	364/200

Documentation Searched other than Minimum Documentation to the Extent that such Documents are Included in the Fields Searched [5]

III. DOCUMENTS CONSIDERED TO BE RELEVANT [14]

Category *	Citation of Document, [16] with indication, where appropriate, of the relevant passages [17]	Relevant to Claim No. [18]
A	US, A, 3,934,232 Published 20 January 1976, Curley et al.	1
A	US, A, 4,123,794 Published 31 October 1978, Matsumoto.	1
A,P	US, A, 4,253,144 Published 24 February 1981, Bellamy et al.	1
A,P	US, A, 4,264,954 Published 28 April 1981, Briggs et al.	1
A,P	US, A, 4,276,594 Published 30 June 1981, Morley.	1

* Special categories of cited documents: [15]

"A" document defining the general state of the art

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

"L" document cited for special reason other than those referred to in the other categories

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

"P" document published prior to the international filing date but on or after the priority date claimed

"T" later document published on or 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

IV. CERTIFICATION

Date of the Actual Completion of the International Search [2]	Date of Mailing of this International Search Report [2]
23 November 1981	02 DEC 1981
International Searching Authority [1]	**Signature of Authorized Officer [20]**
ISA/US	Thomas M. Heckler

Form PCT/ISA/210 (second sheet) (October 1977)