Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2382927 A1 2001/03/08

(21) 2 382 927

(12) DEMANDE DE BREVET CANADIEN
CANADIAN PATENT APPLICATION

(13) A1

(86) Date de dépôt PCT/PCT Filing Date: 2000/08/31
(87) Date publication PCT/PCT Publication Date: 2001/03/08
(85) Entrée phase nationale/National Entry: 2002/02/25
(86) N° demande PCT/PCT Application No.: US 2000/024217
(87) N° publication PCT/PCT Publication No.: 2001/016741
(30) Priorités/Priorities: 1999/08/31 (60/152,151) US;
2000/07/26 (60/220,748) US; 2000/07/26 (60/220,794) US

(51) Cl.Int.^{7}/Int.Cl.^{7} G06F 9/52, G06F 9/46

(71) Demandeur/Applicant:
TIMES N SYSTEMS, INC., US

(72) Inventeurs/Inventors:
WEST, LYNN PARKER, US;
WEST, KARLON K., US

(74) Agent: GOUDREAU GAGE DUBUC

(54) Titre : GESTION PAR SEMAPHORE DE MEMOIRE PARTAGEE
(54) Title: SEMAPHORE CONTROL OF SHARED-MEMORY

(57) Abrégé/Abstract:
Methods, systems and devices are described for semaphore control of a shared-memory cluster. A method, includes writing at least one pointer to a semaphore region of a shared memory region that is coupled to a plurality of processing nodes. The at least one pointer points to at least one of the plurality of processing nodes, the at least one pointer i) indicating that a portion of the shared memory node is dedicated to reading by the at least one of the plurality of processing nodes and ii) protecting access to the portion of the shared memory node until the portion of the shared memory node has been read by the at least one of the plurality of processing nodes. The methods, systems and devices provide advantages because the speed and scalability of parallel processor systems is enhanced.

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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
8 March 2001 (08.03.2001)

PCT

(10) International Publication Number
WO 01/16741 A3

(51) International Patent Classification[7]: G06F 9/52, 9/46

(21) International Application Number: PCT/US00/24217

(22) International Filing Date: 31 August 2000 (31.08.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

60/152,151	31 August 1999 (31.08.1999)	US
60/220,794	26 July 2000 (26.07.2000)	US
60/220,748	26 July 2000 (26.07.2000)	US

(63) Related by continuation (CON) or continuation-in-part (CIP) to earlier applications:

US	60/152,151 (CON)
Filed on	31 August 1999 (31.08.1999)
US	60/220,794 (CON)
Filed on	26 July 2000 (26.07.2000)
US	60/220,748 (CON)
Filed on	26 July 2000 (26.07.2000)

(71) Applicant *(for all designated States except US)*: **TIMES N SYSTEMS, INC.** [US/US]; Bldg. B, Suite P, 1908 Kramer Lane, Austin, TX 78758 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **WEST, Lynn, Parker** [US/US]; 10201 Pantera Drive, Austin, TX 78759 (US). **WEST, Karlon, K.** [US/US]; 2801 Wells Branch Parkway #2128, Austin, TX 78728 (US).

(74) Agent: **BRUCKNER, John, J.**; Fulbright & Jaworski, L.L.P., 600 Congress Avenue, Suite 2400, Austin, TX 78701 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, 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), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*

(88) Date of publication of the international search report:
20 September 2001

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 01/16741 A3

(54) Title: SEMAPHORE CONTROL OF SHARED-MEMORY

(57) Abstract: Methods, systems and devices are described for semaphore control of a shared-memory cluster. A method, includes writing at least one pointer to a semaphore region of a shared memory region that is coupled to a plurality of processing nodes. The at least one pointer points to at least one of the plurality of processing nodes, the at least one pointer i) indicating that a portion of the shared memory node is dedicated to reading by the at least one of the plurality of processing nodes and ii) protecting access to the portion of the shared memory node until the portion of the shared memory node has been read by the at least one of the plurality of processing nodes. The methods, systems and devices provide advantages because the speed and scalability of parallel processor systems is enhanced.

SEMAPHORE CONTROL OF SHARED-MEMORY

BACKGROUND OF THE INVENTION

1. Field of the Invention

The invention relates generally to the field of computer systems based on multiple processors and shared memory. More particularly, the invention relates to computer systems that utilize semaphore control of a shared-memory cluster.

2. Discussion of the Related Art

The clustering of workstations is a well-known art. In the most common cases, the clustering involves workstations that operate almost totally independently, utilizing the network only to share such services as a printer, license-limited applications, or shared files.

In more-closely-coupled environments, some software packages (such as NQS) allow a cluster of workstations to share work. In such cases the work arrives, typically as batch jobs, at an entry point to the cluster where it is queued and dispatched to the workstations on the basis of load.

In both of these cases, and all other known cases of clustering, the operating system and cluster subsystem are built around the concept of message-passing. The term message-passing means that a given workstation operates on some portion of a job until communications (to send or receive data, typically) with another workstation is necessary. Then, the first workstation prepares and communicates with the other workstation.

Another well-known art is that of clustering processors within a machine, usually called a Massively Parallel Processor or MPP, in which the techniques are essentially identical to those of clustered workstations. Usually, the bandwidth and latency of the interconnect network of an MPP are more highly optimized, but the system operation is the same.

In the general case, the passing of a message is an extremely expensive operation; expensive in the sense that many CPU cycles in the sender and receiver are consumed by the process of sending, receiving, bracketing, verifying, and routing the message, CPU cycles that are therefore not available

for other operations. A highly streamlined message-passing subsystem can typically require 10,000 to 20,000 CPU cycles or more.

There are specific cases wherein the passing of a message requires significantly less overhead. However, none of these specific cases is adaptable to a general-purpose computer system.

Message-passing parallel processor systems have been offered commercially for years but have failed to capture significant market share because of poor performance and difficulty of programming for typical parallel applications. Message-passing parallel processor systems do have some advantages. In particular, because they share no resources, message-passing parallel processor systems are easier to provide with high-availability features. What is needed is a better approach to parallel processor systems.

There are alternatives to the passing of messages for closely-coupled cluster work. One such alternative is the use of shared memory for inter-processor communication.

Shared-memory systems, have been much more successful at capturing market share than message-passing systems because of the dramatically superior performance of shared-memory systems, up to about four-processor systems. In Search of Clusters, by Gregory F. Pfister 2nd ed. (January 1998) Prentice Hall Computer Books; ISBN: 0138997098 describes a computing system with multiple processing nodes in which each processing node is provided with private, local memory and also has access to a range of memory which is shared with other processing nodes. The disclosure of this publication in its entirety is hereby expressly incorporated herein by reference for the purpose of indicating the background of the invention and illustrating the state of the art.

However, providing high availability for traditional shared-memory systems has proved to be an elusive goal. The nature of these systems, which share all code and all data, including that data which controls the shared operating systems, is incompatible with the separation normally required for high availability. What is needed is an approach to shared-memory systems that improves availability.

Although the use of shared memory for inter-processor communication is a well-known art, prior to the teachings of U.S. Ser. No. 09/273,430, filed March 19, 1999, entitled Shared Memory Apparatus and Method for Multiprocessing Systems, the processors shared a single copy of the operating system. The problem with such systems is that they cannot be efficiently scaled beyond four to eight way systems except in unusual circumstances. All known cases of said unusual circumstances are such that the systems are not good price-performance systems for general-purpose computing.

The entire contents of U.S. Patent Applications 09/273,430, filed March 19, 1999 and PCT/US00/01262, filed January 18, 2000 are hereby expressly incorporated by reference herein for all purposes. U.S. Ser. No. 09/273,430, improved upon the concept of shared memory by teaching the concept which will herein be referred to as a tight cluster. The concept of a tight cluster is that of individual computers, each with its own CPU(s), memory, I/O, and operating system, but for which collection of computers there is a portion of memory which is shared by all the computers and via which they can exchange information. U.S. Ser. No. 09/273,430 describes a system in which each processing node is provided with its own private copy of an operating system and in which the connection to shared memory is via a standard bus. The advantage of a tight cluster in comparison to an SMP is "scalability" which means that a much larger number of computers can be attached together via a tight cluster than an SMP with little loss of processing efficiency.

What is needed are improvements to the concept of the tight cluster. What is also needed is an expansion of the concept of the tight cluster.

Another well-known art is that of a "heartbeat" function. In a system involving multiple independent computers, a function can be provided such that each of said computers occasionally signals to at least a subset of the other processors an indication that status is operational. Failure to signal this heartbeat is a primary indication that the computer has failed, in either hardware or software. Should a companion processor fail to receive the heartbeat within a specified period of time following the previous received heartbeat signal, said companion processor will execute a verification routine. Should the results of

the verification routine indicate computer failure, the system will enter checkpoint restart mode and will restart. The failed computer will be removed from the group upon restart and an operator message will be issued as part of the restart.

In symmetric multiprocessors (SMP's) such a heartbeat function is normally not applicable, as all the processors are using a single copy of the software, and software failure is the most common failure. Also, in event of hardware failure, the cache of the failed processor may contain dirty, required operating system status, so that recovery is often impossible. Since there is no way to determine from other processors whether recovery is possible, there are no known examples of SMP systems which attempt to recover from processor or memory failures.

While the heartbeat functionality can be provided in the context of a message passing system, such systems have performance deficiencies, as discussed above. Therefore, what is also needed in an approach to providing the heartbeat function in the context of a symmetric multiprocessor system.

SUMMARY OF THE INVENTION

A goal of the invention is to simultaneously satisfy the above-discussed requirements of improving and expanding the tight cluster concept which, in the case of the prior art, are not satisfied. The invention can include a system in which accesses to one particular range of addresses is used for inter-processor signaling, semaphores, pointers, and/or to aid high-availability design.

One embodiment of the invention is based on a method, comprising: writing at least one pointer to a semaphore region of a shared memory region that is coupled to a plurality of processing nodes, wherein the at least one pointer points to at least one of said plurality of processing nodes, the at least one pointer i) indicating that a portion of said shared memory node is dedicated to reading by the at least one of said plurality of processing nodes and ii) protecting access to said portion of said shared memory node until said portion of said shared memory node has been read by the at least one of said plurality of processing nodes. Another embodiment of the invention is based on a system,

comprising a multiplicity of processors, each with some private memory and the multiplicity with some shared memory, interconnected and arranged such that memory accesses to a first set of address ranges will be to local, private memory whereas memory accesses to a second set of address ranges will be to shared memory, and arranged such that at least a portion of one special range of memory addresses or I/O addresses are provided for the purpose of signaling from a first processor to a second, said signaling to occur when said first processor reads or writes a location dedicated to the signaling of said second processor, and said location protected by semaphore or other locking mechanism so that said first processor can determine whether said second processor has already been signaled by a third process or processor and can wait using defined procedures for the signaling event to complete. Another embodiment of the invention is based on a system, comprising a multiplicity of processors, each with some private memory and the multiplicity with some shared memory, interconnected and arranged such that memory accesses to a first set of address ranges will be to local, private memory whereas memory accesses to a second set of address ranges will be to shared memory, and arranged such that at least a portion of one special range of memory addresses or I/O addresses are provided for the purpose of semaphore control of the remainder or a portion of the remainder of shared memory so that when a first process enters a critical section it may obtain a semaphore to continue into that section.

These, and other goals and embodiments of the invention will be better appreciated and understood when considered in conjunction with the following description and the accompanying drawings. It should be understood, however, that the following description, while indicating preferred embodiments of the invention and numerous specific details thereof, is given by way of illustration and not of limitation. Many changes and modifications may be made within the scope of the invention without departing from the spirit thereof, and the invention includes all such modifications.

BRIEF DESCRIPTION OF THE DRAWINGS

A clear conception of the advantages and features constituting the invention, and of the components and operation of model systems provided with the invention, will become more readily apparent by referring to the exemplary, and therefore nonlimiting, embodiments illustrated in the drawings accompanying and forming a part of this specification, wherein like reference characters designate the same parts. It should be noted that the features illustrated in the drawings are not necessarily drawn to scale.

FIG. 1 illustrates a block diagram of a system including a shared memory node and a plurality of processing nodes, representing an embodiment of the invention.

FIG. 2 illustrates a block diagram of a shared memory node having a shared memory region and a semaphore region, representing an embodiment of the invention.

FIG. 3 illustrates a block diagram of a system including a shared memory region and a semaphore region, showing pointers from the semaphore region to each of six processing nodes, representing an embodiment of the invention.

FIG. 4 illustrates a block diagram of a system showing a node P5 signaling to a node P2 via a semaphore region, representing an embodiment of the invention.

FIG. 5 illustrates a semaphore format, representing an embodiment of the invention.

DESCRIPTION OF PREFERRED EMBODIMENTS

The invention and the various features and advantageous details thereof are explained more fully with reference to the nonlimiting embodiments that are illustrated in the accompanying drawings and detailed in the following description of preferred embodiments. Descriptions of well known components and processing techniques are omitted so as not to unnecessarily obscure the invention in detail.

The teachings of U.S. Ser. No. 09/273,430 include a system which is a single entity; one large supercomputer. The invention is also applicable to a cluster of workstations, or even a network.

The invention is applicable to systems of the type of Pfister or the type of U.S. Ser. No. 09/273,430 in which each processing node has its own copy of an operating system. The invention is also applicable to other types of multiple processing node systems.

The context of the invention can include a tight cluster as described in U.S. Ser. No. 09/273,430. A tight cluster is defined as a cluster of workstations or an arrangement within a single, multiple-processor machine in which the processors are connected by a high-speed, low-latency interconnection, and in which some but not all memory is shared among the processors. Within the scope of a given processor, accesses to a first set of ranges of memory addresses will be to local, private memory but accesses to a second set of memory address ranges will be to shared memory. The significant advantage to a tight cluster in comparison to a message-passing cluster is that, assuming the environment has been appropriately established, the exchange of information involves a single STORE instruction by the sending processor and a subsequent single LOAD instruction by the receiving processor.

The establishment of the environment, taught by U.S. Ser. No. 09/273,430 and more fully by companion disclosures (U.S. Provisional Application Ser. No. 60/220,794, filed July 26, 2000; U.S. Provisional Application Ser. No. 60/220,748, filed July 26, 2000; WSGR 15245-711; WSGR 15245-712; WSGR 15245-713; WSGR 15245-715; WSGR 15245-716; WSGR 15245-718; WSGR 15245-719; and WSGR 15245-720, the entire contents of all which are hereby expressly incorporated herein by reference for all purposes) can be performed in such a way as to require relatively little system overhead, and to be done once for many, many information exchanges. Therefore, a comparison of 10,000 instructions for message-passing to a pair of instructions for tight-clustering, is valid.

The present invention contemplates such an environment in which each standing computer is provided with very high-speed, low-latency

communication means to shared memory, and which shared-memory includes at least one address range usable for each of the following: (1) the signaling of one processor by another; (2) semaphores; (3) pointer passing; and (4) failure recovery. The low-latency communication means can include a communication link based on traces, cables and/or optical fiber(s). The low-latency communication means can include hardware (e.g., a circuit), firmware (e.g., flash memory) and/or software (e.g., a program). The communications means can include on-chip traces and/or waveguides. Further, each of these communication means can be duplicated.

Any or several of the above functions could be provided in a separate address range, or several address ranges, or a single dedicated address range. Any or several of the above functions could be provided in an I/O address range, or several I/O address ranges, or a single dedicated I/O address range.

Within the stated address range, signaling of one processor by another can be accomplished by reading or writing a particular address, an address dedicated to the destination processor. Semaphore protecting the access to the particular address will assure that after a first processor signals the target processor, a second processor can determine whether a signaling event to the target processor is under way.

Referring to FIG. 1, a system 100 includes a shared memory node 110 and a plurality of processing nodes 120, 130, 140. Processing node 120 is coupled to the shared memory node 110 with a communications link 150. Processing node 130 is coupled to the shared memory node 110 with a communications link 160. Processing node 140 is coupled to the shared memory node 110 with a communications link 170.

Referring to FIG. 2, the shared memory node 110 includes a shared memory region 210. The shared memory region 210 includes a semaphore region 220.

Referring to FIG. 3, another embodiment is shown. This embodiment includes six processing nodes. The six processing nodes include a first processing node 310. The six processing nodes include a second processing node 320. The six processing nodes include a third processing node 330. The

six processing nodes include a fourth processing node 340. The six processing nodes include a fifth processing node 350. The six processing nodes include a sixth processing node 360.

Still referring to FIG. 3, a shared memory node 370 includes a semaphore region 380. The semaphore region includes six pointers P0, P1, P2, P3, P4, P5, P6, each of which points to one of the six processing nodes. The pointing feature can be implements with data contained within the pointer itself as discussed below in more detail.

By convention the signaled processor and the signaling processor use a separate mechanism to indicate acknowledgment of the signaling event. However, acknowledgment is not required.

The signaling event is initiated by reading or writing a particular address. The value written to that or to a companion address can be a simple command or vector address to a location providing more complex command or sequence of commands.

Within the range, or in a separate range, a sub-range of addresses can be used to semaphore protect all of memory. Semaphore protection should be on a word, a cache line, or other meaningful element. All of protectable memory will be hashed, on an entity-granularity basis, to the semaphore sub-range. In the preferred embodiment, the means of such hashing is to address the sub-range modulo the lowest unique memory address bits. Of course, other hashing means can be derived.

When a given process encounters a critical section that requires semaphore protection, that process uses the atomic operation provided for the particular processor architecture to write the semaphore protecting that address range. If the semaphore has already been written, the process enters a waiting routine.

Given the hashing of memory for semaphores, there is a non-zero probability that a lock on one critical section can interfere with entry to a different critical section. Based on empirical tests, if at least 2048 semaphores are provided, such interference did not occur. In the actual running of a system

over time, such interference will occur, but is only a performance issue so that if it occurs very rarely, no measurable loss of performance will occur.

While not being limited to any particular performance indicator or diagnostic identifier, preferred embodiments of the invention can be identified one at a time by testing for the presence of measurable loss of performance. Preferred embodiments will demonstrate substantially no measurable loss of performance. The test for the presence of measurable loss of performance can be carried out without undue experimentation by the use of a simple and conventional benchmark (speed) experiment.

The semaphore range can be used as an aid in failure recovery when accompanied by "heartbeat" mechanisms. When capturing a semaphore, this invention teaches the concept that the owning processor write its identification to the semaphore location. When subsequent heartbeat mechanisms indicate that a processor has failed, the processor detecting the heartbeat failure will search and release semaphores owned by the failed processor.

The semaphore region can be duplicated in a mirrored-memory region. This increases the reliability, and therefore, the availability of the system.

The term substantially, as used herein, is defined as at least approaching a given state (e.g., preferably within 10% of, more preferably within 1% of, and most preferably within 0.1% of). The term coupled, as used herein, is defined as connected, although not necessarily directly, and not necessarily mechanically. The term means, as used herein, is defined as hardware, firmware and/or software for achieving a result. The term program or phrase computer program, as used herein, is defined as a sequence of instructions designed for execution on a computer system. A program may include a subroutine, a function, a procedure, an object method, an object implementation, an executable application, an applet, a servlet, a source code, an object code, and/or other sequence of instructions designed for execution on a computer system.

Example

A specific embodiment of the invention will now be further described by the following, nonlimiting example which will serve to illustrate in some detail various features of significance. The example is intended merely to facilitate an

understanding of ways in which the invention may be practiced and to further enable those of skill in the art to practice the invention. Accordingly, the example should not be construed as limiting the scope of the invention.

Referring to FIG. 4, a specific implementation of the invention is shown. A shared memory node 410 includes a semaphore region 420. A first node P5 can signal to a second node P2 with a pointer P2 that is located within the semaphore region 420. The existence of the pointer P2 alerts other nodes to the fact that a portion of the shared region is dedicated to reading by node P2.

Referring to FIG. 5, a semaphore can be more informationally rich than a simple flag. A semaphore format 500 can include a count of messages in the shared region for a node 510 (in this example for node P2). The semaphore format 500 can include a pointer to a shared region dedicated to reading by the node 520 (in this example node P2). The semaphore 500 can include a pointer to the node 530 (in this example node P). Thus, the semaphore can be self identifying and does not have to be written to a node specific memory region. Finally, the semaphore 500 can include a lock bit 540.

Practical Applications of the Invention

A practical application of the invention that has value within the technological arts is waveform transformation. Further, the invention is useful in conjunction with data input and transformation (such as are used for the purpose of speech recognition), or in conjunction with transforming the appearance of a display (such as are used for the purpose of video games), or the like. There are virtually innumerable uses for the invention, all of which need not be detailed here.

Advantages of the Invention

A system, representing an embodiment of the invention, can be cost effective and advantageous for at least the following reasons. The invention improves the speed of parallel computing systems. The invention improves the scalability of parallel computing systems.

All the disclosed embodiments of the invention described herein can be realized and practiced without undue experimentation. Although the best mode of carrying out the invention contemplated by the inventors is disclosed above,

practice of the invention is not limited thereto. Accordingly, it will be appreciated by those skilled in the art that the invention may be practiced otherwise than as specifically described herein.

For example, although the shared memory node described herein can be a separate module, it will be manifest that the shared memory node may be integrated into the system with which it is associated. Furthermore, all the disclosed elements and features of each disclosed embodiment can be combined with, or substituted for, the disclosed elements and features of every other disclosed embodiment except where such elements or features are mutually exclusive.

It will be manifest that various additions, modifications and rearrangements of the features of the invention may be made without deviating from the spirit and scope of the underlying inventive concept. It is intended that the scope of the invention as defined by the appended claims and their equivalents cover all such additions, modifications, and rearrangements.

The appended claims are not to be interpreted as including means-plus-function limitations, unless such a limitation is explicitly recited in a given claim using the phrase "means for." Expedient embodiments of the invention are differentiated by the appended subclaims.

CLAIMS

What is claimed is:

1. A method, comprising:

writing at least one pointer to a semaphore region of a shared memory region that is coupled to a plurality of processing nodes,

wherein the at least one pointer points to at least one of said plurality of processing nodes, the at least one pointer i) indicating that a portion of said shared memory node is dedicated to reading by the at least one of said plurality of processing nodes and ii) protecting access to said portion of said shared memory node until said portion of said shared memory node has been read by the at least one of said plurality of processing nodes.

2. The method of claim 1, wherein writing the at least one pointer includes writing data that represents a count of messages in said portion of said shared memory node to be read the at least one of said plurality of processing nodes.

3. The method of claim 1, wherein writing the at least one pointer includes writing data that identifies said portion of said shared memory node.

4. The method of claim 1, wherein writing the at least one pointer includes writing data that points to one of said plurality of processing nodes.

5. The method of claim 1, wherein writing the at least one pointer includes writing a lock bit.

6. The method of claim 1, further comprising determining whether a signaling event to the at least one of said plurality of processor nodes is under way.

7. The method of claim 1, further comprising acknowledging completion of a signaling event.

8. The method of claim 1, further comprising erasing the at least one pointer from said semaphore region.

9. The method of claim 1, further comprising writing another pointer to said semaphore region.

10. An apparatus, comprising:

a shared memory node; and

a plurality of processing nodes coupled to said shared memory node,

wherein said shared memory node includes at least one semaphore region having at least one pointer that points to at least one of said plurality of processing nodes, the at least one pointer i) indicating that a portion of said shared memory node is dedicated to reading by the at least one of said plurality of processing nodes and ii) protecting access to said portion of said shared memory node until said portion of said shared memory node has been read by the at least one of said plurality of processing nodes.

11. The apparatus of claim 10, wherein the at least one pointer includes data that represents a count of messages in said portion of said shared memory node to be read by the at least one of said plurality of processing nodes.

12. The apparatus of claim 10, wherein the at least one pointer includes data that identifies said portion of said shared memory node.

13. The apparatus of claim 10, wherein the at least one pointer includes data that points to one of said plurality of processing nodes.

14. The apparatus of claim 10, wherein the at least one pointer includes a lock bit.

15. The apparatus of claim 10, further comprising a plurality of links coupled between said shared memory node and said plurality of processing nodes.

16. A computer system comprising the apparatus of claim 10.

17. An electronic media, comprising: a computer program adapted to write at least one pointer to a semaphore region of a shared memory region that is coupled to a plurality of processing nodes,

wherein the at least one pointer points to at least one of said plurality of processing nodes, the at least one pointer i) indicating that a portion of said shared memory node is dedicated to reading by the at least one of said plurality of processing nodes and ii) protecting access to said portion of said shared memory node until said portion of said shared memory node has been read by the at least one of said plurality of processing nodes.

18. A computer program comprising computer program means adapted to perform the step of writing at least one pointer to a semaphore region of a shared memory region that is coupled to a plurality of processing nodes when said computer program is run on a computer,

wherein the at least one pointer points to at least one of said plurality of processing nodes, the at least one pointer i) indicating that a portion of said shared memory node is dedicated to reading by the at least one of said plurality of processing nodes and ii) protecting access to said portion of said shared memory node until said portion of said shared memory node has been read by the at least one of said plurality of processing nodes.

19. A computer program as claimed in claim 18, embodied on a computer-readable medium.

20. A system, comprising:

a multiplicity of processors, each with private memory; and

a shared memory,

wherein said multiplicity of processors and said shared memory are interconnected and arranged such that

memory accesses to a first set of address ranges will be to private memory whereas memory accesses to a second set of address ranges will be to said shared memory and

at least a portion of one range of memory addresses or I/O addresses are provided for the purpose of signaling from a first processor to a second processor, said signaling to occur when said first processor reads or writes a location dedicated to said signaling, said location protected by a semaphore or other locking mechanism so that said first processor can determine whether said second processor has already been signaled by a third process or processor and, if said second processor has already been signaled by a third process or processor, said first processor can wait using defined procedures for signaling by said third process or processor to be complete.

21. The system of claim 20, further comprising storing in a signal address location or a companion location a simple command or vector address to a location providing a complex command or sequence of commands.

22. The system of claim 20, wherein said at least said portion of one range of memory is a binary bit.

23. A system, comprising:

a multiplicity of processors, each with private memory; and

a shared memory,

wherein said multiplicity of processors and said shared memory are interconnected and arranged such that

memory accesses to a first set of address ranges will be to local, private memory whereas memory accesses to a second set of address ranges will be to shared memory, and

at least a portion of one range of memory addresses or I/O addresses are provided for the purpose of semaphore control of at least a portion of a remainder of said shared memory so that when a first process enters a critical section it may obtain a semaphore to continue into that section.

24. The system of claim 23, wherein said semaphore is to be obtained by reading a semaphore location to which an addressable element address hashes, said obtaining of said semaphore to be protected by the hardware atomic operation provided for the system.

25. The system of claim 24, wherein if a semaphore is already locked when said first process attempts to obtain said semaphore, said semaphore is not obtained by said first process and said first process waits for the release of said semaphore.

26. The system of claim 23, wherein obtaining said semaphore is accompanied by

registration of an obtaining processor's identification and

determining at a first processor when a second processor has stopped responding using an implementation of a heartbeat function,

wherein release of said semaphore held by said second processor in response to such determination on a repeated basis is an indication that operations are normal.

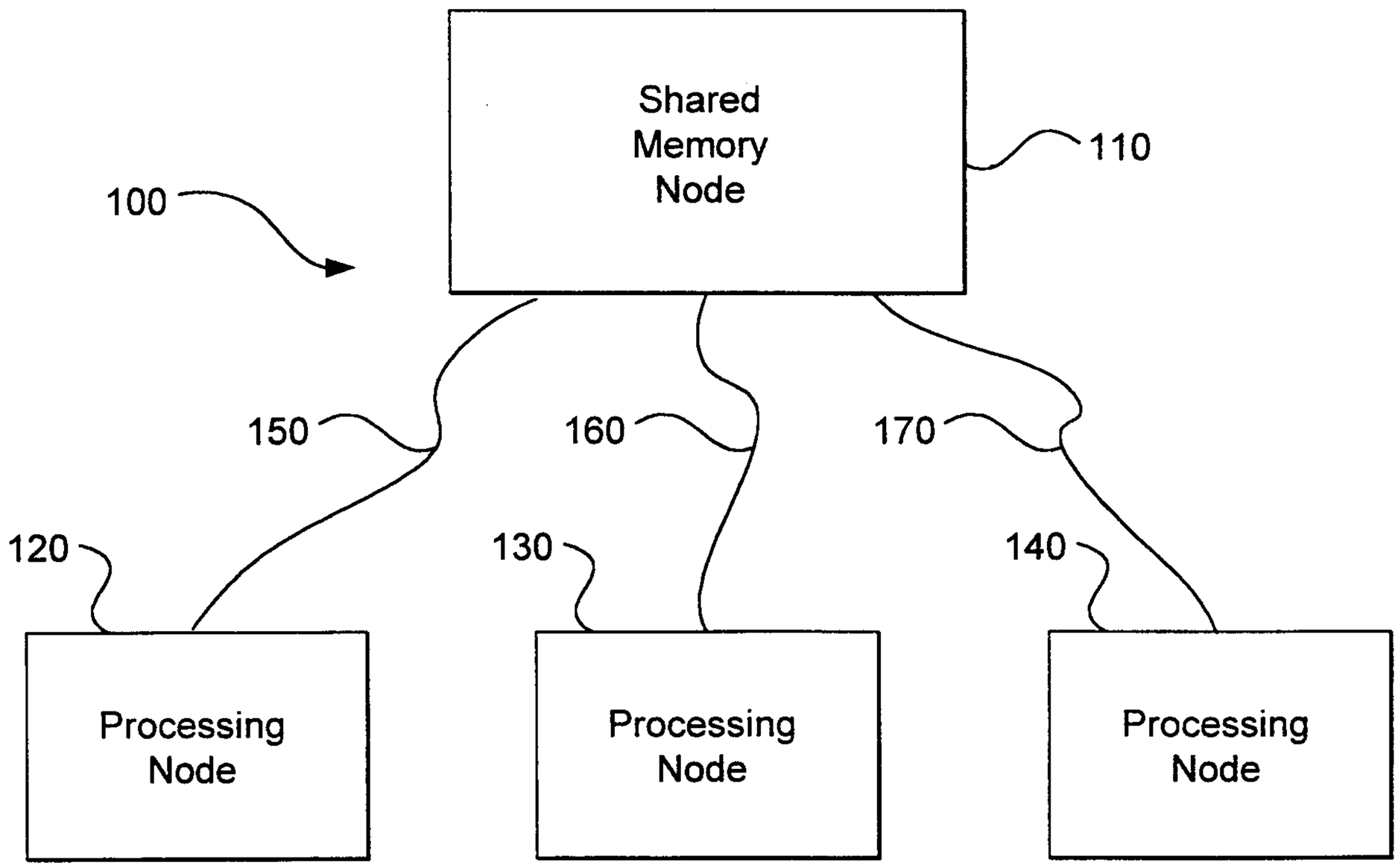
Shared
Memory
Node
110
100
150
160
170
120
130
140
Processing
Node
Processing
Node
Processing
Node

FIG. 1

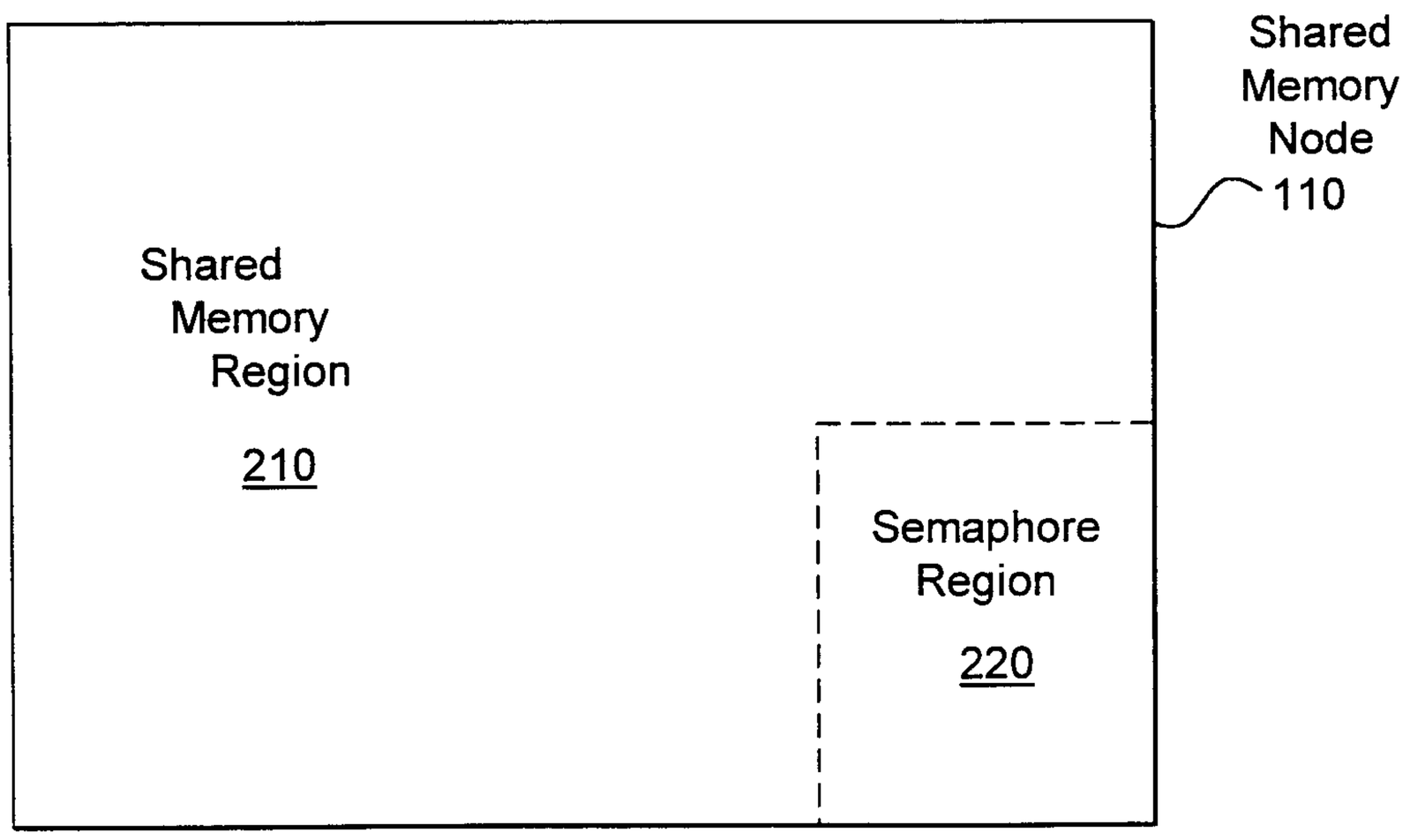
Shared
Memory
Node
110
Shared
Memory
Region
210
Semaphore
Region
220

FIG. 2

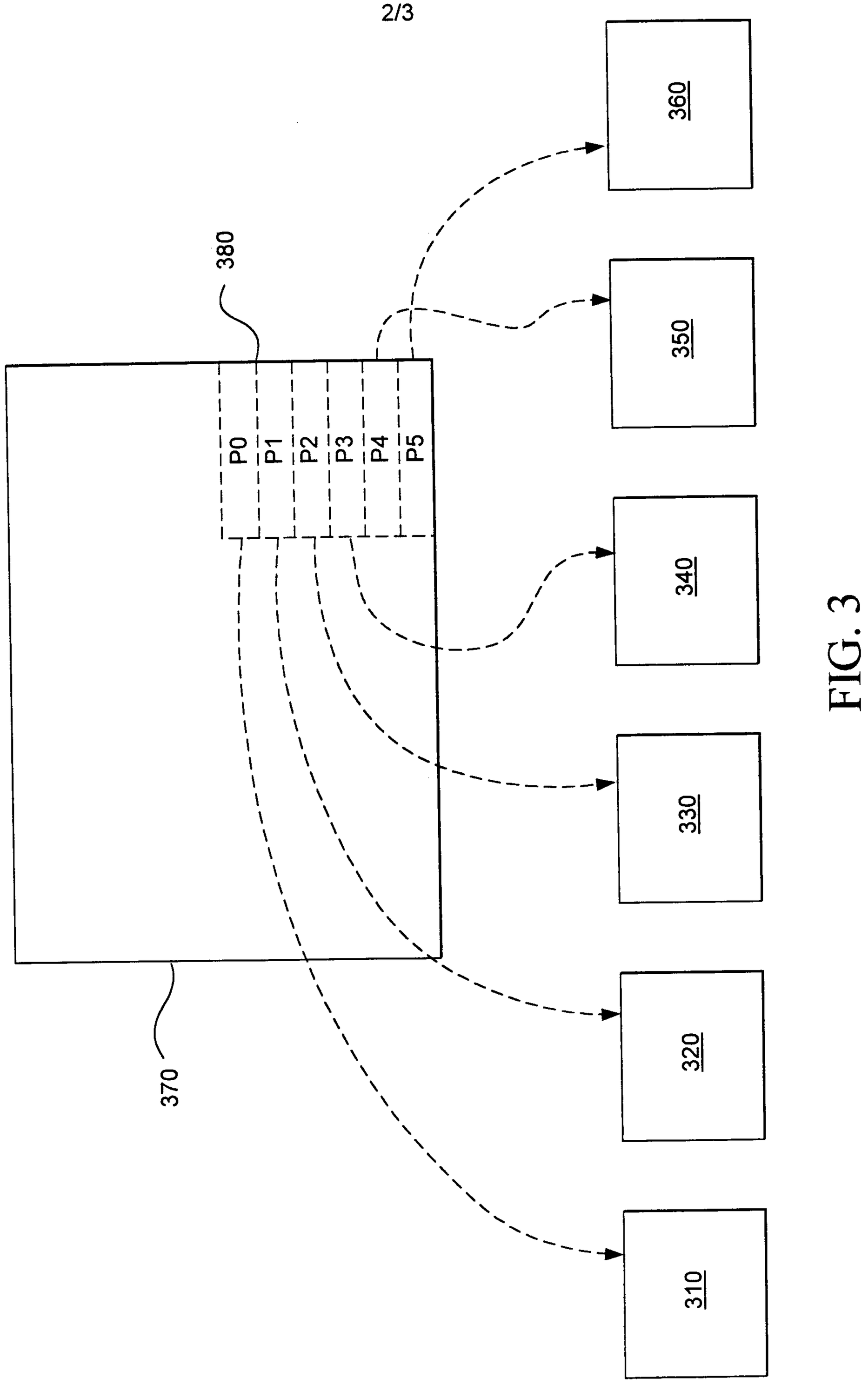

FIG. 3

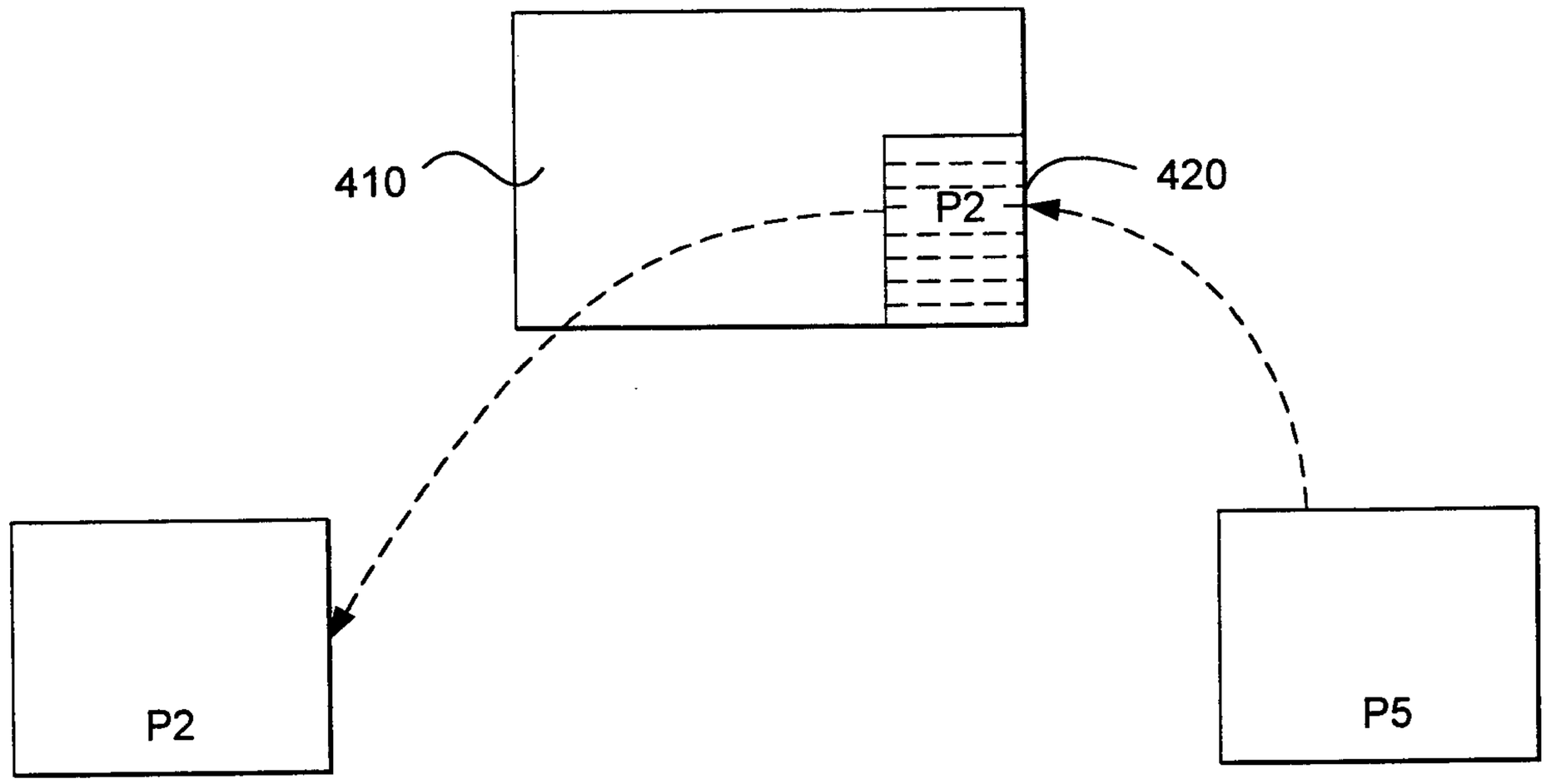
410
P2
420
P2
P5

FIG. 4

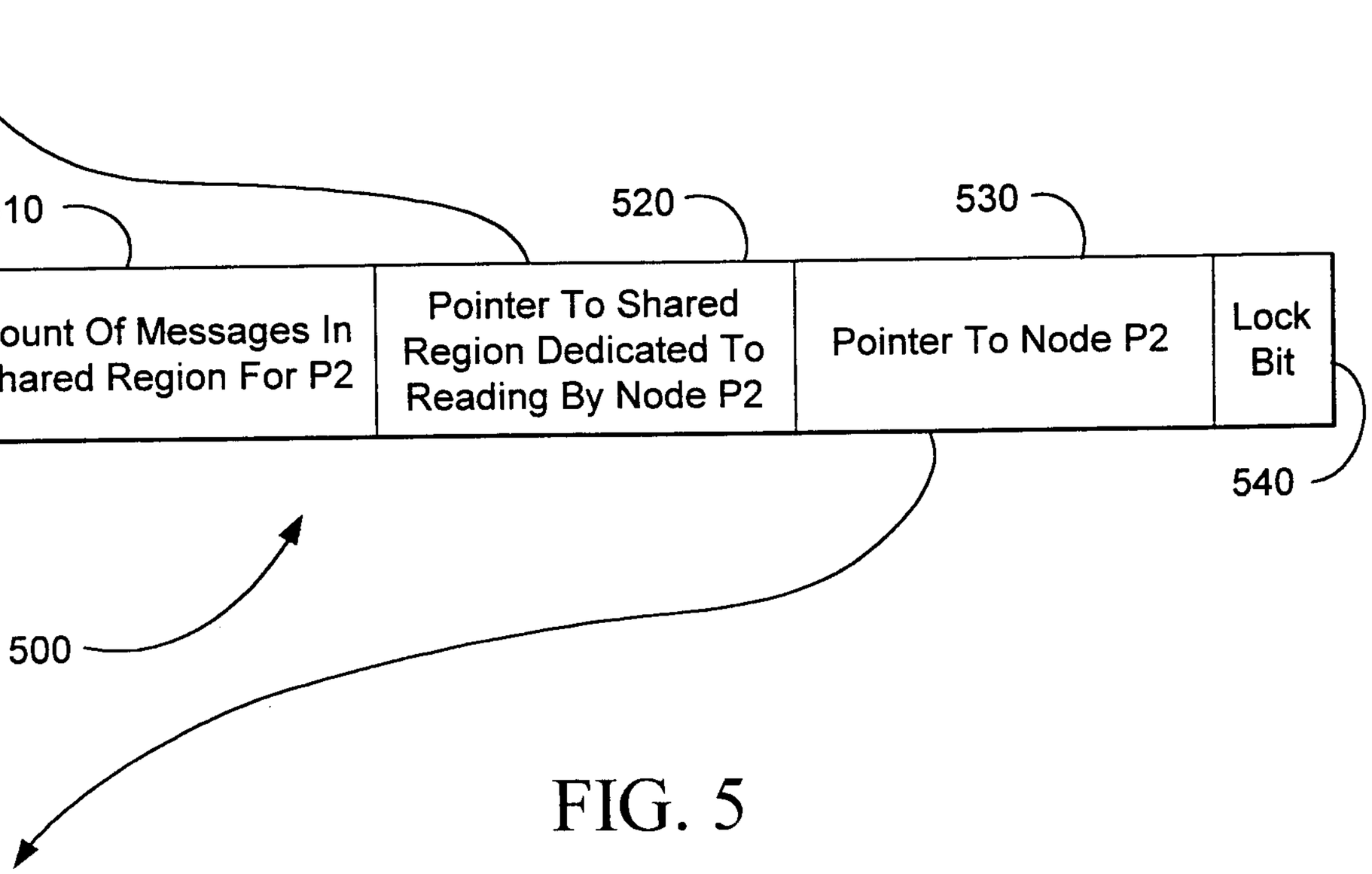
510
520
530
Count Of Messages In Shared Region For P2
Pointer To Shared Region Dedicated To Reading By Node P2
Pointer To Node P2
Lock Bit
540
500

FIG. 5