



(51) International Patent Classification:

G06F 11/22 (2006.01) G06F 13/38 (2006.01)

G06F 13/14 (2006.01)

(21) International Application Number:

PCT/US201 1/068009

(22) International Filing Date:

30 December 2011 (30.12.2011)

(25) Filing Language:

English

(26) Publication Language:

English

(71) Applicant (for all designated States except US): **INTEL CORPORATION** [US/US]; 2200 Mission College Boulevard, Santa Clara, California 95052 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **KORAH, Shino** [IN/US]; 8004 Levata Drive, Austin, Texas 78739 (US). **DUNGAR, Jeetendrasingh** [IN/US]; 11004 Cherisse Drive, Austin, Texas 78739 (US).

(74) Agent: **RICHARDS, E.E. "Jack"**; Trop, Pruner & Hu, P.C., 1616 S. Voss Rd., Ste. 750, Houston, Texas 77057-2631 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ,

CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

- as to the identity of the inventor (Rule 4.17(i))
- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(H))
- of inventorship (Rule 4.17(iv))

Published:

- with international search report (Art. 21(3))

(54) Title: EARLY FABRIC ERROR FORWARDING

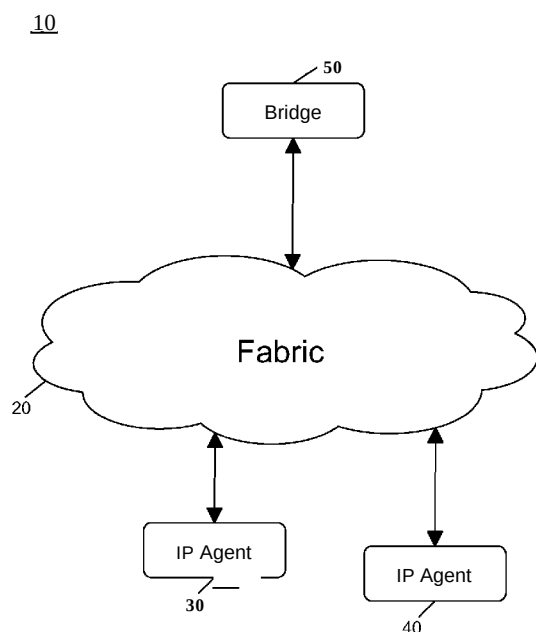


FIG. 1

(57) Abstract: An embodiment includes a processor, included in a first die, detecting a first hardware error (e.g., memory corruption error). The processor formulates a first error message based on the detected first error. The processor may forward the first error message, via a first fabric included on the first die, to a first intellectual property (IP) agent or block, also included in the first die. This may be done using an in-band message, such as one that adheres to the integrated on-chip system fabric (IOSF) specification. The first IP agent may issue an error message that may, for example, alert a user to the error in a manner that is relatively close in time when the error first occurs. This may facilitate better error analysis. The first IP agent may include a processor (e.g., graphics processor), controller (e.g., power management controller), and the like. Other embodiments are described herein.

EARLY FABRIC ERROR FORWARDINGBackground

[0001] Mainstream processor chips, both in high performance and low power segments, are increasingly integrating additional functionality such as graphics, display engines, security engines, PCIe™ ports (i.e., ports in accordance with the Peripheral Component Interconnect Express (PCI Express™ (PCIe™)) Specification Base Specification version 2.0 (published 2007) (hereafter the PCIe™ specification) and other PCIe™ based peripheral devices, while maintaining legacy support for devices compliant with a PCI specification such as the Peripheral Component Interconnect (PCI) Local Bus Specification, version 3.0 (published 2002) (hereafter the PCI specification).

[0002] Such designs are highly segmented due to varying requirements from the server, desktop, mobile, embedded, ultra-mobile and mobile Internet device segments. Different markets seek to use single chip system-on-chip (SoC) solutions that combine at least some of processor cores, memory controllers, input/output controllers and other segment specific acceleration elements onto a single chip. However, designs that accumulate these features are slow to emerge due to the difficulty of integrating different intellectual property (IP) blocks or agents on a single die. This is especially so, as IP blocks can have various requirements and design uniqueness, and can require many specialized wires, communication protocols and so forth to enable their incorporation into an SoC. As a result, each SoC or other advanced semiconductor device that is developed requires a great amount of design complexity and customization to incorporate different IP blocks into a single device. This is so, as a given IP block typically needs to be re-designed to accommodate interface and signaling requirements of a given SoC.

[0003] Even when different IP blocks are successfully incorporated into a SoC, problems persist. For example, when a processor hardware error occurs the error can be slow to be recognized. Since the error is recognized long after it occurs, the error's point of origin can be difficult to ascertain. More specifically, the error may be slow to be recognized because the error must propagate through multiple layers of hardware

- 2 -

and software before becoming "visible" to the user at the system level. This slow error propagation causes error analysis difficulty because the analysis must "back track" the hardware failure back to the time of occurrence.

Brief Description of the Drawings

[0004] Features and advantages of embodiments of the present invention will become apparent from the appended claims, the following detailed description of one or more example embodiments, and the corresponding figures, in which:

[0005] FIG. 1 is a block diagram of a basic interconnect architecture in accordance with an embodiment of the invention.

[0006] FIG. 2 is a block diagram of further details of an interconnect architecture in accordance with an embodiment of the invention.

[0007] FIG. 3 is a high level block diagram of a SoC in accordance with an embodiment of the invention.

[0008] FIG. 4 is a block diagram of a system in accordance with another embodiment of the invention.

[0009] FIG. 5 is a schematic flow diagram for error message processing in an embodiment of the invention.

[0010] FIG. 6 includes a pseudocode example for message processing in an embodiment of the invention.

Detailed Description

[001 1] In the following description, numerous specific details are set forth but embodiments of the invention may be practiced without these specific details. Well-known circuits, structures and techniques have not been shown in detail to avoid obscuring an understanding of this description. "An embodiment", "various embodiments" and the like indicate embodiment(s) so described may include particular features, structures, or characteristics, but not every embodiment necessarily includes the particular features, structures, or characteristics. Some embodiments may have some, all, or none of the features described for other embodiments. "First", "second", "third" and the like describe a common object and

- 3 -

indicate different instances of like objects are being referred to. Such adjectives do not imply objects so described must be in a given sequence, either temporally, spatially, in ranking, or in any other manner. "Connected" may indicate elements are in direct physical or electrical contact with each other and "coupled" may indicate elements co-operate or interact with each other, but they may or may not be in direct physical or electrical contact. Also, while similar or same numbers may be used to designate same or similar parts in different figures, doing so does not mean all figures including similar or same numbers constitute a single or same embodiment.

[0012] An embodiment includes a processor, included in a first die, detecting a first hardware error (e.g., memory corruption error). The processor formulates a first error message based on the detected first error. The processor (e.g., one or more cores of a multi-processor core) may forward the first error message, via a first fabric included on the first die, to a first IP agent or block, also included in the first die. This may be done using an in-band message, such as one that adheres to the integrated on-chip system fabric (IOSF) specification (discussed below). The first IP agent may issue an error message that may, for example, alert a user to the error in a manner that is relatively close in time when the error first occurs. This may facilitate better error analysis. The first IP agent may include a processor (e.g., graphics processor), controller (e.g., power management controller), and the like.

[0013] Embodiments can be used in many different types of systems. As examples, implementations described herein may be used in connection with semiconductor devices such as processors or other semiconductor devices that can be fabricated on a single semiconductor die. In particular implementations, the device may be a SoC or other advanced processor or chipset that includes various homogeneous and/or heterogeneous processing agents, and additional components such as networking components, e.g., routers, controllers, bridge devices, devices, memories and so forth.

[0014] Some implementations may be used in a semiconductor device that is designed according to a given specification such as an IOSF specification issued by a semiconductor manufacturer to provide a standardized on-die interconnect protocol for attaching IP blocks within a chip, including a SoC. Such IP blocks can be of

- 4 -

varying types, including general-purpose processors such as in-order or out-of-order cores, fixed function units, graphics processors, I/O controllers, display controllers, media processors among many others. By standardizing an interconnect protocol, a framework is thus realized for a broad use of IP agents in different types of chips. Accordingly, not only can the semiconductor manufacturer efficiently design different types of chips across a wide variety of customer segments, it can also, via the specification, enable third parties to design logic such as IP agents to be incorporated in such chips. And furthermore, by providing multiple options for many facets of the interconnect protocol, reuse of designs is efficiently accommodated. Although embodiments are described herein in connection with this IOSF specification, understand the scope of the present invention is not limited in this regard and embodiments can be used in many different types of systems.

[0015] Referring now to FIG. 1, shown is a block diagram of a basic interconnect architecture in accordance with an embodiment of the present invention. As shown in FIG. 1, system 10 may be a portion of a system-on-chip or any other semiconductor device such as a highly integrated processor complex or an integrated I/O hub, and includes a fabric 20 that acts as an interconnect between various components. In the implementation shown, these components include IP agents 30 and 40, which can be independent IP blocks to provide various functionality such as compute capabilities, graphics capabilities, media processing capabilities and so forth. These IP agents are thus IP blocks or logical devices having an interface that is compliant with the IOSF specification, in one embodiment. As further seen, fabric 20 also interfaces to a bridge 50. Although not shown for ease of illustration in the embodiment of FIG. 1, understand that bridge 50 may act as an interface to other system components, e.g., on the same chip or on one or more different chips.

[0016] As will be described further below, each of the elements shown in FIG. 1, namely the fabric, the IP agents, and the bridge may include one or more interfaces to handle communication of various signals. These interfaces may be defined according to the IOSF specification, which defines signals for communication on these interfaces, protocols used for information exchange between agents, arbitration and flow control mechanisms used to initiate and manage information exchange,

- 5 -

supported address decoding and translation capabilities, messaging for in-band or out-of-band communication, power management, test, validation and debug support.

[0017] The IOSF specification includes 3 independent interfaces that can be provided for each agent, namely a primary interface, a sideband message interface and a testability and debug interface (design for test (DFT), design for debug (DFD) interface). According to the IOSF specification, an agent may support any combination of these interfaces. Specifically, an agent can support 0-N primary interfaces, 0-N sideband message interfaces, and optional DFX interfaces.

[0018] Fabric 20 may be a hardware element that moves data between different agents. Note that the topology of fabric 20 will be product specific. As examples, a fabric can be implemented as a bus, a hierarchical bus, a cascaded hub or so forth. Referring now to FIG. 2, shown is a block diagram of further details of an interconnect architecture in accordance with an embodiment of the present invention. As shown in FIG. 2, the IOSF specification defines three distinct fabrics, namely a primary interface fabric 112, a DFX fabric 114, and a sideband fabric 116. Primary interface fabric 112 is used for all in-band communication between agents and memory, e.g., between a host processor such as a central processing unit (CPU) or other processor and an agent. Primary interface fabric 112 may further enable communication of peer transactions between agents and supported fabrics. All transaction types including memory, input output (IO), configuration, and in-band messaging can be delivered via primary interface fabric 112. Thus the primary interface fabric may act as a high performance interface for data transferred between peers and/or communications with upstream components.

[0019] In various implementations, primary interface fabric 112 implements a split transaction protocol to achieve maximum concurrency. That is, this protocol provides for a request phase, a grant phase, and a command and data phase. Primary interface fabric 112 supports three basic request types: posted, non-posted, and completions, in various embodiments. Generally, a posted transaction is a transaction which when sent by a source is considered complete by the source and the source does not receive a completion or other confirmation message regarding the transaction. One such example of a posted transaction may be a write transaction. In

- 6 -

contrast, a non-posted transaction is not considered completed by the source until a return message is received, namely a completion. One example of a non-posted transaction is a read transaction in which the source agent requests a read of data. Accordingly, the completion message provides the requested data.

[0020] In addition, primary interface fabric 112 supports the concept of distinct channels to provide a mechanism for independent data flows throughout the system. As will be described further, primary interface fabric 112 may itself include a master interface that initiates transactions and a target interface that receives transactions. The primary master interface can further be sub-divided into a request interface, a command interface, and a data interface. The request interface can be used to provide control for movement of a transaction's command and data. In various embodiments, primary interface fabric 112 may support PCI ordering rules and enumeration.

[0021] In turn, sideband interface fabric 116 may be a standard mechanism for communicating all out-of-band information. In this way, special-purpose wires designed for a given implementation can be avoided, enhancing the ability of IP reuse across a wide variety of chips. Thus in contrast to an IP block that uses dedicated wires to handle out-of-band communications such as status, interrupt, power management, fuse distribution, configuration shadowing, test modes and so forth, a sideband interface fabric 116 according to the IOSF specification standardizes all out-of-band communication, promoting modularity and reducing validation requirements for IP reuse across different designs. In general, sideband interface fabric 116 may be used to communicate non-performance critical information, rather than for performance critical data transfers, which typically may be communicated via primary interface fabric 112.

[0022] As further illustrated in FIG. 2, IP agents 130, 140, and 150 may each include a corresponding primary interface, a sideband interface and a DFX interface. However, as discussed above, each agent need not include every one of these interfaces, and a given IP agent may include only a single interface, in some embodiments.

- 7 -

[0023] Using an IOSF specification, various types of chips can be designed having a wide variety of different functionality. Referring now to FIG. 3, shown is a high level block diagram of a SoC in accordance with an embodiment of the present invention. As shown in FIG. 3, SoC 200 may include various components, all of which can be integrated on a single semiconductor die to provide for various processing capabilities at high speeds and low power, consuming a comparatively small amount of real estate. As seen in FIG. 3, SoC 200 includes a plurality of cores 205o - 205n. In various embodiments, cores 205 can be relatively simple in-order cores or more complex out-of-order cores. Or a combination of in-order and out-of-order cores can be present in a single SoC. As seen, cores 205 can be interconnected via a coherent interconnect 215, which further couples to a cache memory 210, e.g., a shared last level cache (LLC). Although the scope of the present invention is not limited in this regard, in one embodiment coherent interconnect 215 may be in accordance with the Quick Path Interconnect (QPI)[™] specification available from Intel Corporation, Santa Clara, California.

[0024] As further seen in FIG. 3, coherent interconnect 215 may communicate via a bridge 220 to a fabric 250, which may be an IOSF fabric. Coherent interconnect 215 may further communicate via an integrated memory controller 225 to an off-chip memory (not shown for ease of illustration the embodiment of FIG. 3), and further through bridge 230 to fabric 250.

[0025] As further seen in FIG. 3, various components can couple to fabric 250 including a content processing module (CPM) 240 which can be used for performing various operations such as security processing, cryptographic functions and so forth. In addition, a display processor 245 can be part of a media processing pipeline that renders video for an associated display.

[0026] As further seen, fabric 250 may further couple to an IP agent 255. Although only a single "IP Agent" is labeled as such, other IP agents are included in the figure and many other such agents are possible in different embodiments. Regarding the embodiment of Figure 3, B unit 276 and H unit 288 are both IP blocks. H unit 278 determines whether a transaction from any of cores (e.g., core 205n) should go to memory (e.g., memory 285) or to a south complex, which may couple to universal

- 8 -

serial bus (USB) controller 265, SATA controller, and the like. B unit 276 forwards core transactions (e.g., core 205n) to memory (e.g., memory 285).

[0027] In addition, to enable communication with other on-chip devices, fabric 250 may further communicate with a PCIe™ controller 260 and USB controller 265, both of which can communicate with various devices according to these protocols. Finally, shown in the embodiment of FIG. 3 is bridge 270, which can be used to communicate with additional components of other protocols, such as an open core protocol (OCP) or an ARM advanced microcontroller bus architecture (AMBA) protocol. Although shown with these particular components in the embodiment of FIG. 3, understand that the scope of the present invention is not limited in this way and in different embodiments additional or different components may be present.

[0028] Furthermore, understand that while shown as a single die SoC implementation in FIG. 3, embodiments can further be implemented in a system in which multiple chips communicate with each other via a non-IOSF interface. Referring now to FIG. 4, shown is a block diagram of a system in accordance with another embodiment of the present invention. As shown in FIG. 4, the system may include SoC 200', which may include many components similar to those discussed above with regard to FIG. 3, and an additional off-die interface 275. Accordingly, SoC 200' can communicate with another chip 280 which may include various functionality to enable communication between these two chips, as well as to various off-chip devices such as different peripherals according to one or more different specifications. Specifically, a second chip 280 is shown to include an off-die interface 282 to enable communication with SoC 200', and which in turn communicates with a fabric 290, which may be an IOSF fabric according to an embodiment of the present invention. As seen, fabric 290 may further be coupled to various controllers in communication with off-chip devices, including a PCIe™ controller 292, a USB controller 294, and a bridge 296.

[0029] One method for failure propagation includes reporting the failure through multiple layers of hardware and software. Returning to Figure 3, core 205n reads location X in memory 285 (e.g., DDR SDRAM). Core 205n may be expecting data Z. However, memory 285 may suffer a failure and may return incorrect memory content,

- 9 -

such as data Y, to core 205. Data Y may be corrupt, be of the wrong data type, and the like. Core 205n may detect a fault (e.g., error condition) and flag this failure. The software layer in an operating system (OS), known as a "fault handler", may then be called. A fault handler includes programming language constructs or computer hardware mechanisms designed to handle the occurrence of exceptions that change the normal flow of program execution. The fault handler may take actions leading to a double or triple fault shutdown when, for example, memory block 285 keeps returning corrupted data to core 205n. This failure may cause a kernel panic, blue screen, or shutdown for the user.

[0030] A double fault exception may occur if the processor encounters a problem while trying to service a pending interrupt or exception. An example situation when a double fault may occur is when an interrupt is triggered but the segment in which the interrupt handler resides is invalid. For example, the interrupt handler may be located in faulty location X of memory 285. If the processor encounters a problem when calling the double fault handler, a triple fault is generated and the processor shuts down. Some processors may save state information to predefined and reserved machine registers. A double fault may then be a situation where another exception happens while the processor is still using the contents of these registers to process the exception.

[0031] A triple fault may be an exception generated by the processor when an exception occurs while the processor is trying to invoke the double fault exception handler, which itself handles exceptions occurring while trying to invoke a regular exception handler. Processors may cause a shutdown cycle to occur when a triple fault is encountered. This may cause hardware to initiate a processor reset which in turn may cause the whole computer system, which includes the processor, to reboot.

[0032] A kernel panic, "blue screen" and the like are various terms for actions taken by various operating systems upon detecting an internal fatal error from which it cannot safely recover. The kernel routines that handle panics are generally designed to output an error message to the console, dump an image of kernel memory to disk for post-mortem debugging, and then either wait for the system to be manually rebooted, or initiate an automatic reboot. Then, to "back track" the failure on the

- 10 -

system level extra hardware may be needed to trace a user visible failure back to the initial hardware error condition.

[0033] In some instances, memory corruption may only get detected when, for example, the caches (e.g., cache 210 of Figure 3) get flushed or a wbinvd instruction (Write Back and Invalidate Cache) is executed.

[0034] However, in one embodiment error detection is sped up. This may be based at least in part on fabric (e.g., IOSF) error forwarding. This may occur in a multi-clock domain SOC. The embodiment may detect and flag a processor error (e.g., hardware error or failure) relatively close in time to when the failure occurs on the fabric based SOC system.

[0035] Returning again to Figure 3, core 205n may read location X in memory 285. Core 205n may be expecting data Z. However, memory 285 may suffer a failure and may return incorrect memory content, such as data Y, to core 205. Data Y may be corrupt, be of the wrong data type, and the like.

[0036] Core 205n may detect a fault (e.g., error condition) and flag this failure. Core 205 may have hardware to detect the fault/exceptions. For example, with an Intel® Atom™ processor the "Front-End Cluster", "Integer Execution Cluster", and/or "Memory Execution Cluster" may detect various types of instruction-based or external event-based errors.

[0037] However, upon detection of a fault, instead of just proceeding to call a fault handler, an embodiment may log the error (e.g., using a core) and microcode (e.g., processor microcode coupled to core 205n) may issue a related fabric based message. In one embodiment the fabric message consists of a destination ID, source ID, message payload, and data payload. Source ID may relate to processor 205n and destination ID may relate to any IP agent, such as agent 255. The message may include a hexadecimal opcode such as 0xF2, 0xF4, and the like and the data payload may include, for example, 0x0.

[0038] Core 205n may forward the fabric based message through system fabric 250, included in SOC system 200, to IP agent 255. IP blocks are interconnected together using, for example, the IOSF protocol. Such IP blocks can be of varying types,

- 11 -

including general-purpose processors such as in-order or out-of-order cores, power management controller (PMC)/power management unit (PMU), fixed function units, graphics processors, IO controllers, display controllers, media processors, power management controllers among many others.

[0039] The PMC (also called PMU) may include a microcontroller that governs power functions of digital platforms (e.g., system 200). This microcontroller may include firmware, software, memory, a processor or controller, input/output functions, timers to measure intervals of time, as well as analog to digital converters to measure the voltages of the main battery or power source of the computer. The PMU may control power-consuming functions in system 200, run diagnostics on various power-related operations, and check those diagnostics against energy settings to allow the PMC to actively manage power consumption.

[0040] Based on receiving the message, the IP agents (e.g., PMC of IP block 255) can trigger actions such as, for example, toggling on/off chip pins such as a General Purpose Input/Output (GPIO) pin (which may be connected to a light emitting diode (LED), thereby causing a message visible to a user), stopping the clock in a multiclock domain, forwarding an interrupt to processor 205n, generating an interrupt based on screen message, and/or various user defined actions that may alert the user to error and the source of the error.

[0041] While core 205n is used to, for example, detect an error, create a message, and forward a message (see above), please note these tasks may be performed by other cores (e.g., core 205₀) or by a combination of cores (e.g., 205₀ and 205n) and still be said to be performed by a processor (albeit one with several cores).

[0042] Figure 6 includes a pseudocode example for message processing by a processor as described immediately above.

[0043] Figure 5 is a schematic flow diagram for error message processing in an embodiment of the invention. Such a process allows for detecting the failure relatively near the time of occurrence and does not depend on the system to shutdown for user detection (or other user visible symptoms, such as "blue screen" or kernel panic, where the elapsed time from failure to user detection is very high). Such a process

- 12 -

captures the failure at nearly the time of occurrence. In an embodiment, the process uses processor microcode to issue an IOSF side band message to trigger system failure.

[0044] Block 505 of process 500 includes a processor (cores 205₀, 205_n), included in a first die, detecting a first error. Hardware such as the Atom™ processor's "Front-End Cluster", "Integer Execution Cluster", and/or "Memory Execution Cluster" may facilitate such detection. The error may include, for example, a memory corruption error.

[0045] In block 510, the processor formulates a first error message based on the detected first error. As noted above, the first error message may include, for example, destination ID, source ID, message payload, and data payload.

[0046] In block 515, the processor may forward the first error message, via a first fabric (e.g., fabric 250) included on the first die, to a first IP agent (i.e., IP block 255). The first IP agent may include one of, for example, a graphics processor, IO controller, display controller, media processor, and PMC. The "forwarding" may be done via a side band message. Further, the first IP agent may (in some embodiments) be included in the same die as the processor. The message may be forwarded to the IP agent via a first side band interface (e.g., sideband interface fabric 116 of Figure 2) included in the processor and a second side band interface (e.g., sideband interface 144 of IP agent 140 in Figure 2) included in the first IP agent. The side band interfaces, as well as the message itself, may adhere to fabric specification (e.g., IOSF).

[0047] Also, this forwarding may comprise forwarding the first error message directly from the fabric to the first IP agent. In non-fabric based systems (e.g., conventional main stream CPU design) there may be no direct way of communicating to, for example, the south complex from the CPU. However, in a fabric based system fabric side band message communication and fabric error forwarding can be used to trigger error detection. This reduces the circuitry needed to "trace back" an error.

[0048] In block 520 the first IP agent issues a second error message based on the first error message. This second message may be forwarded independently of all

- 13 -

operating system fault handler messages. For example, the second message may be created and forwarded regardless of whether a fault handler is called. The second error message may, directly or indirectly, perform some user defined action to detect failure such as, for example, manipulate a GPIO pin coupled to a LED to alert a user to the fault. As another example, a PMU included in IP block 255 may be programmed to respond to a side band message from core 205n to log and detect any error at the system level.

[0049] In an embodiment, in block 530 (depicted in dashed lines to connote the block as an alternative or additional route to the route depicted in solid lines) the processor, having detected another error or possibly the same error of block 505, forms a third error message. The third error message may have a makeup similar to the message of block 510. In block 535 the processor forwards the third error message, via the first fabric included on the first die, to a second fabric included in a second die. The second fabric and the first fabric may both adhere to an IOSF standard. For example, see fabrics 250 and 290 of Figure 4.

[0050] As used herein a processor or controller may include control logic intended to represent any of a wide variety of control logic known in the art and, as such, may be implemented as a microprocessor, a micro-controller, a field-programmable gate array (FPGA), application specific integrated circuit (ASIC), programmable logic device (PLD) and the like. A processor may refer to any device or portion of a device that processes electronic data from registers and/or memory to transform that electronic data into other electronic data that may be stored in registers and/or memory. A processor may be distributed, may include different cores, and the like. For example, the various portions of Figure 5 may be conducted by various cores of a processor.

[0051] Thus, one embodiment includes a SoC based on a fabric (e.g., IOSF fabric) as the interface between on die blocks. The aforementioned embodiments may allow a system to detect processor (e.g., CPU) triggered hardware errors by utilizing fabric based error forwarding.

[0052] Embodiments may be implemented in code and may be stored on a storage medium having stored thereon instructions which can be used to program a system to perform the instructions. The storage medium may include, but is not limited to, any

- 14 -

type of disk including floppy disks, optical disks, solid state drives (SSDs), compact disk read-only memories (CD-ROMs), compact disk rewritables (CD-RWs), and magneto-optical disks, semiconductor devices such as read-only memories (ROMs), random access memories (RAMs) such as dynamic random access memories (DRAMs), static random access memories (SRAMs), erasable programmable read-only memories (EPROMs), flash memories, electrically erasable programmable read-only memories (EEPROMs), magnetic or optical cards, or any other type of media suitable for storing electronic instructions.

[0053] Embodiments of the invention may be described herein with reference to data such as instructions, functions, procedures, data structures, application programs, configuration settings, code, etc. When the data is accessed by a machine, the machine may respond by performing tasks, defining abstract data types, establishing low-level hardware contexts, and/or performing other operations, as described in greater detail herein. The data may be stored in volatile and/or non-volatile data storage. For purposes of this disclosure, the terms "code" or "program" cover a broad range of components and constructs, including applications, drivers, processes, routines, methods, modules, and subprograms. Thus, the terms "code" or "program" may be used to refer to any collection of instructions which, when executed by a processing system, performs a desired operation or operations. In addition, alternative embodiments may include processes that use fewer than all of the disclosed operations, processes that use additional operations, processes that use the same operations in a different sequence, and processes in which the individual operations disclosed herein are combined, subdivided, or otherwise altered.

[0054] While the present invention has been described with respect to a limited number of embodiments, those skilled in the art will appreciate numerous modifications and variations therefrom. It is intended that the appended claims cover all such modifications and variations as fall within the true spirit and scope of this present invention.

- 15 -

- 1 1. A method comprising:
2 a processor, included in a first die, detecting a first error;
3 the processor formulating a first error message based on the detected first
4 error;
5 the processor forwarding the first error message, via a first fabric included in
6 the first die, to a first intellectual property (IP) agent; and
7 the first IP agent issuing a second error message based on the first error
8 message.
- 1 2. The method of claim 1, wherein the first IP agent includes one of a graphics
2 processor, input/output (IO) controller, display controller, media processor, and
3 power management controller.
- 1 3. The method of claim 1 comprising the processor forwarding the first error
2 message to the first IP agent, which is included in the first die, via a first side band
3 message.
- 1 4. The method of claim 3, wherein the first error includes a memory corruption
2 error.
- 1 5. The method of claim 3 comprising the processor forwarding the first error
2 message to the first IP agent via a first side band interface included in the processor
3 and a second side band interface included in the first IP agent.
- 1 6. The method of claim 5, wherein the first error message and the first and
2 second side band interfaces adhere to an integrated on-chip system fabric (IOSF)
3 specification.

- 16 -

1 7. The method of claim 3 comprising forwarding the first error message directly
2 from the fabric to the first IP agent with no intervening components between the
3 fabric and the first IP agent.

1 8. The method of claim 3, wherein the second error message performs one of
2 manipulating a General Purpose Input/Output (GPIO) pin and performing a user
3 defined action that alerts a user to the first error.

1 9. The method of claim 3 comprising the first IP agent issuing the second error
2 message independently of all operating system fault handler messages and all
3 cache flushes of any cache coupled to the processor.

1 10. The method of claim 1 comprising the processor forwarding a third error
2 message, via a first side band message and the first fabric, to a second fabric
3 included in a second die.

1 11. A set of instructions residing in one or more storage mediums, the set of
2 instructions to be executed by at least one processor to implement the method of
3 claims 1 to 10.

1 12. An apparatus comprising:
2 a processor included in a first die;
3 a first fabric, coupled to the processor, included in the first die; and
4 a first intellectual property (IP) agent, coupled to the fabric, included in the first
5 die;
6 wherein the processor is to formulate a first error message based on the detected
7 first error and forward the first error message, via the first fabric, to the first IP agent;
8 and the first IP agent is to issue a second error message based on the first error
9 message.

- 17 -

1 13. The apparatus of claim 12, wherein the first IP agent includes one of a
2 graphics processor, input/output (IO) controller, display controller, media processor,
3 and power management controller.

1 14. The apparatus of claim 12, wherein the processor is to forward the first
2 error message to the first IP agent, which is included in the first die, via a first side
3 band message.

1 15. The apparatus of claim 14, wherein the first error includes a memory
2 corruption error.

1 16. The apparatus of claim 14, wherein the processor is to forward the first error
2 message to the first IP agent via a first side band interface included in the processor
3 and a second side band interface included in the first IP agent.

1 17. The apparatus of claim 16, wherein the first error message and the first and
2 second side band interfaces adhere to an integrated on-chip system fabric (IOSF)
3 specification.

1 18. The apparatus of claim 14, wherein the fabric is to forward the first error
2 message directly from the fabric to the first IP agent.

1 19. The apparatus of claim 14, wherein the processor is to forward a third error
2 message, via the first fabric included on the first die, to a second fabric included in a
3 second die.

1 20. A computer system including the first and second dies of claim 19.

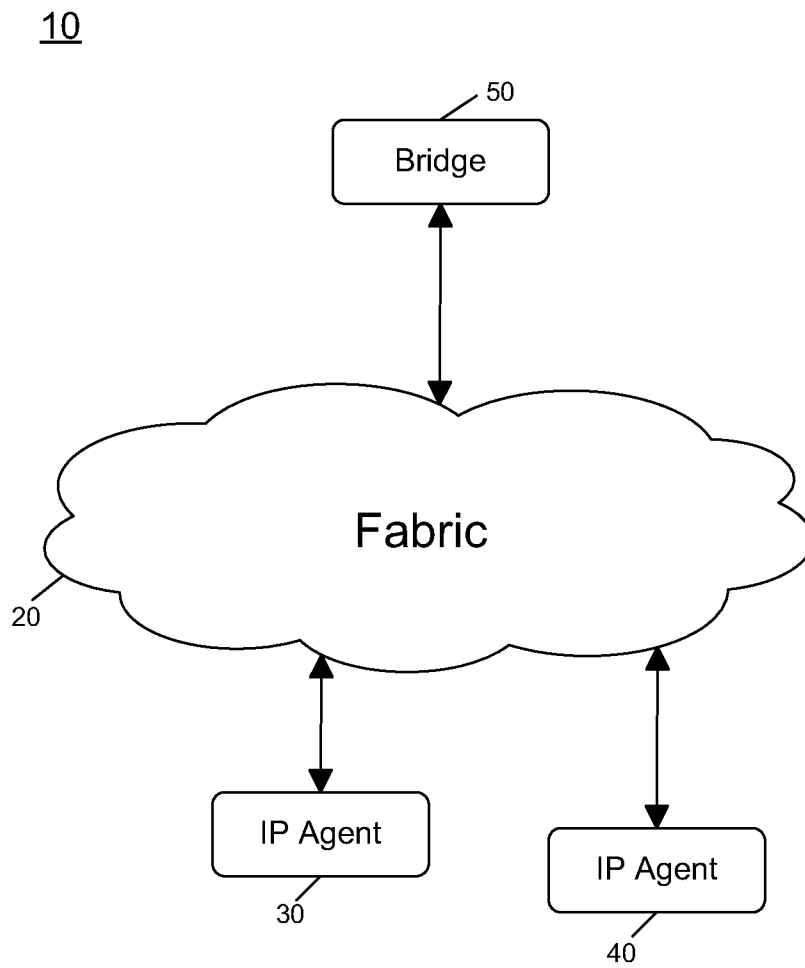


FIG. 1

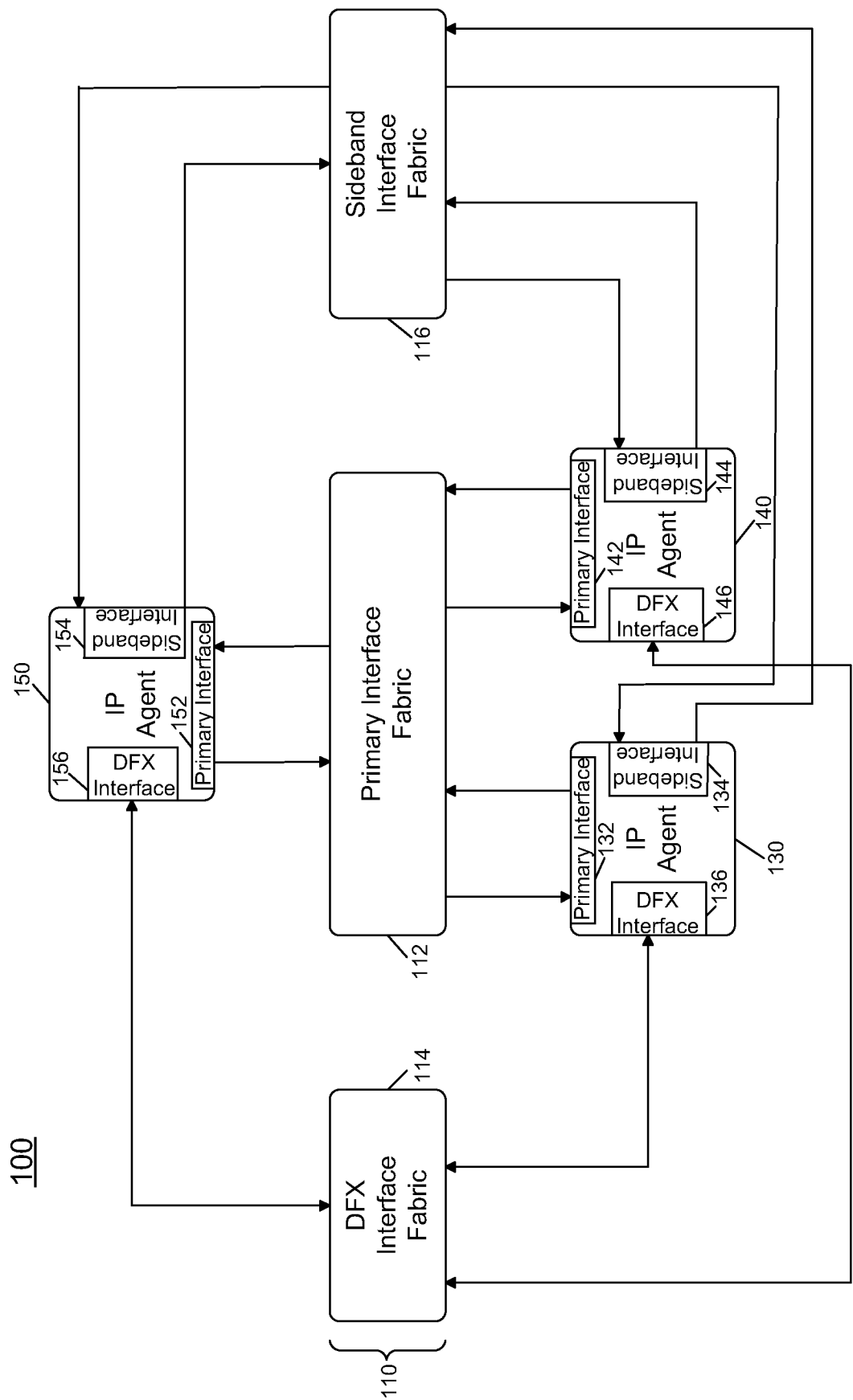


FIG. 2

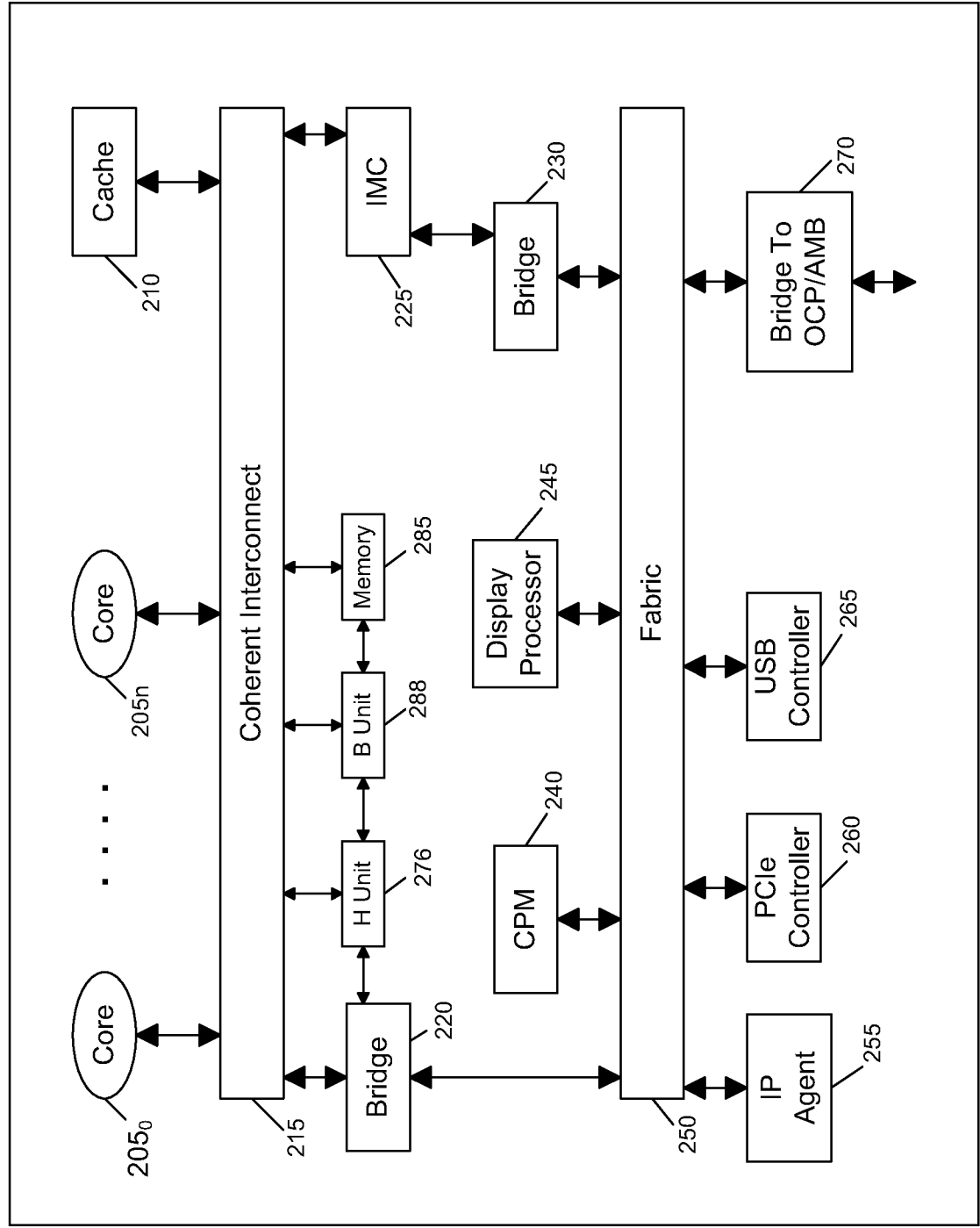


FIG. 3

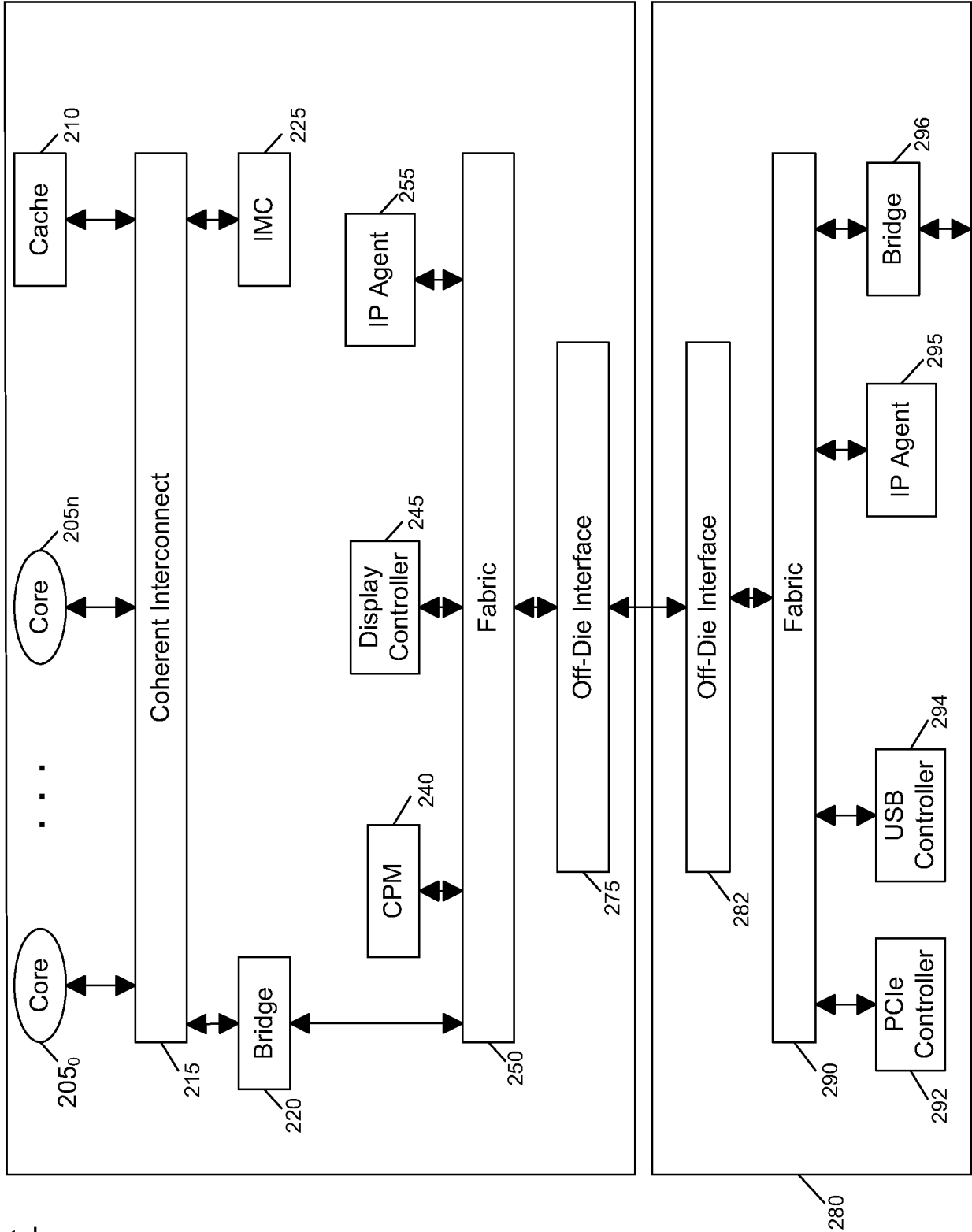


FIG. 4

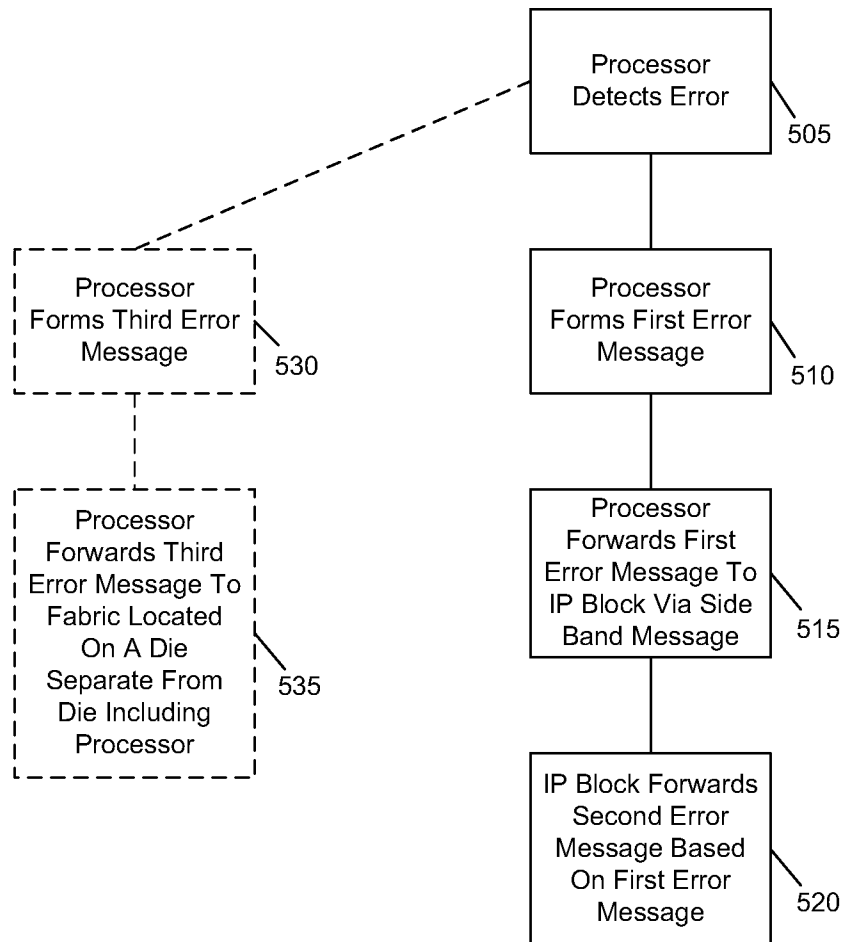
500

FIG. 5

```
While(1){  
  Read fault/exception status register  
  If (register bit set){  
    Generate packet for fabric message with  
      Destination ID = PMC (power management controller)  
      Source ID = processor  
      Opcode message = 0xf3  
    Send_fabric_message()  
  }  
  Else{  
    Do_nothing();  
  }  
}
```

FIG. 6

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US201 1/068009**A. CLASSIFICATION OF SUBJECT MATTER****G06F 11/22(2006.01)i, G06F 13/14(2006.01)1, G06F 13/38(2006.01)1**

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F 11/22; G05F 1/63; G06F 13/16; H04L 12/56; G06F 11/00

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility modelsElectronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & Keywords: "Error, forward, intellectual, property, processor"**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2008-0297129 A1 (NAGASAKI OSAMU et al.) 04 December 2008 See abstract ; paragraphs [0039] -[0053] and figure 2 .	1-10 , 12-20
A	US 2011-0080920 A1 (KLINGLESMTIH MICHAEL et al.) 07 April 2011 See abstract ; paragraphs [0025] -[0034] and figures 1 , 2 .	1-10 , 12-20
A	EP 2189911 A1 (FUJITSU LIMITED) 26 May 2010 See abstract ; paragraphs [0001] -[0063] and figure 9 .	1-10 , 12-20
A	US 2008-0126852 A1 (BRANDYBERRY MARK A. et al.) 29 May 2008 See abstract ; paragraphs [0035] -[0077] and figures 1 , 2 .	1-10 , 12-20



Further documents are listed in the continuation of Box C .



See patent family annex.

* Special categories of cited documents:

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

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search

27 SEPTEMBER 2012 (27.09.2012)

Date of mailing of the international search report

28 SEPTEMBER 2012 (28.09.2012)

Name and mailing address of the ISA/KR

Korean Intellectual Property Office
189 Cheongsu-ro, Seo-gu, Daejeon Metropolitan
City, 302-701, Republic of Korea

Facsimile No. 82-42-472-7140

Authorized officer

Lee Byung So

Telephone No. 82-42 481 5697



INTERNATIONAL SEARCH REPORT

International application No.

PCT/US201 1/068009

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☒ Claims Nos.: II
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

1. ☒ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2. ☐ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.
3. ☒ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

- ☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.
- ☒ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.
- ☒ No protest accompanied the payment of additional search fees.

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US20 11/068009

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2008-0297129 A1	04.12.2008	JP 2009-171824 A	30.07.2009
US 2011-0080920 A1	07.04.2011	CN 101873339 A	27.10.2010
		CN 102326362 A	18.01.2012
		US 2010-0246594 A1	30.09.2010
		US 2011-0131362 A1	02.06.2011
		US 2011-0131456 A1	02.06.2011
		US 7873068 B2	18.01.2011
		WO 2010-117524 A2	14.10.2010
		WO 2010-117524 A3	13.01.2011
EP 2189911 A1	26.05.2010	AT 522869 T	15.09.2011
		CN 101739341 A	16.06.2010
		EP 2189911 B1	31.08.2011
		JP 2010-118020 A	27.05.2010
		KR 10-1087177 B1	25.11.2011
		KR 10-2010-0054734 A	25.05.2010
		US 2010-0125687 A1	20.05.2010
		US 8185668 B2	22.05.2012
US 2008-0126852 A1	29.05.2008	CN 101126995 A	20.02.2008
		CN 101126995 B	29.09.2010
		CN 101126995 C0	20.02.2008
		US 7594144 B2	22.09.2009