(19) NL Octrooicentrum

(11) 2007481

(12) C OCTROOI

(21) Aanvraagnummer: **2007481**

(22) Aanvraag ingediend: **27.09.2011**

(51) Int.Cl.:
G06F 1/32 (2006.01)

(30) Voorrang:
30.09.2010 US 12/894516

(43) Aanvraag gepubliceerd:
11.04.2012

(47) Octrooi verleend:
13.11.2012

(45) Octrooischrift uitgegeven:
21.11.2012

(73) Octrooihouder(s):
Apple Inc. te Cupertino, California, Verenigde Staten van Amerika (US).

(72) Uitvinder(s):
Timothy J. Millet te Cupertino, California (US).
Eric P. Machnicki te Cupertino, California (US).
Deniz Balkan te Cupertino, California (US).
Vijay Gupta te Cupertino, California (US).

(74) Gemachtigde:
Dr. R. Jorritsma c.s. te Den Haag.

(54) **Hardware Dynamic Cache Power Management.**

(57) In an embodiment, a control circuit is configured to transmit operations to a circuit block that is being powered up after being powered down, to reinitialize the circuit block for operation. The operations may be stored in a memory (e.g. a set of registers) to which the control circuit is coupled. In an embodiment, the control circuit may also be configured to transmit other operations from the memory to the circuit block prior to the circuit block being powered down. Accordingly, the circuit block may be powered up or powered down even during times that the processors in the system are powered down (and thus software is not executable at the time), without waking the processors for the power up/power down event. In an embodiment, the circuit block may be a cache coupled to the one or more processors.

NL C 2007481

Hardware Dynamic Cache Power Management

BACKGROUND

Field of the Invention

[0001] This invention is related to the field of digital systems and, more particularly, to power management in digital systems.

Description of the Related Art

[0002] As the number of transistors included on an integrated circuit "chip" continues to increase, power management in the integrated circuits continues to increase in importance. Power management can be critical to integrated circuits that are included in mobile devices such as personal digital assistants (PDAs), cell phones, smart phones, laptop computers, net top computers, etc. These mobile devices often rely on battery power, and reducing power consumption in the integrated circuits can increase the life of the battery. Additionally, reducing power consumption can reduce the heat generated by the integrated circuit, which can reduce cooling requirements in the device that includes the integrated circuit (whether or not it is relying on battery power).

[0003] Clock gating is often used to reduce dynamic power consumption in an integrated circuit, disabling the clock to idle circuitry and thus preventing switching in the idle circuitry. Some integrated circuits have implemented power gating in addition to clock gating. With power gating, the power to ground path of the idle circuitry is interrupted, reducing the leakage current to near zero. When the power is gated to a block and later restored, the block can require reinitialization. The reinitialization is handled by software executed on a processor in the system.

SUMMARY

[0004] In an embodiment, a control circuit is configured to transmit operations to a circuit block that is being powered up after being powered down, to reinitialize the circuit block for operation. The operations may be stored in a memory (e.g. a set of registers) to which the control circuit is coupled, and software executing in the system that includes the control circuit and circuit block may program the memory with the operations at a time prior to the powering down of the circuit block. In an embodiment, the control circuit may also be configured to transmit other operations from the memory to the circuit block prior to the circuit block being powered down. Accordingly, the circuit block may be powered up or powered down even during times that the processors in the system are powered down (and thus software is not executable at the time), without waking the processors for the power up/power down event.

[0005] In an embodiment, the circuit block may be a cache coupled to the one or more processors, and the control circuit may be part of a bridge that couples one or more peripherals and/or peripheral interface controllers to the cache. The cache may be powered down if the processors are powered down and the peripherals are idle (at least with respect to accessing memory) for a period of time. The cache may be powered up for a peripheral memory operation or to power up the processors. In one embodiment, the cache control circuitry may be powered down, but the cache memory may remain powered to retain the cache blocks that are stored in the cache.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] The following detailed description makes reference to the accompanying drawings, which are now briefly described.

[0007] Fig. 1 is a block diagram of one embodiment of a system.

[0008] Fig. 2 is a block diagram of a portion of the system shown in Fig. 1, in greater detail for an embodiment.

[0009] Fig. 3 is a flowchart illustrating operation of one embodiment of a power manager to power down a cache dynamically.

[0010] Fig. 4 is a flowchart illustrating operation of one embodiment of a core interface unit to power down a cache dynamically.

[0011] Fig. 5 is a flowchart illustrating operation of one embodiment of a power manager to power up a cache dynamically.

[0012] Fig. 6 is a flowchart illustrating operation of one embodiment of a core interface unit to power up a cache dynamically.

[0013] Fig. 7 is a timing diagram illustrating dynamic power down and power up of a cache.

[0014] Fig. 8 is a flowchart illustrating operation of one embodiment of cache configuration code.

[0015] Fig. 9 is a block diagram of a computer accessible storage medium.

[0016] Fig. 10 is a block diagram of another embodiment of a system.

[0017] While the invention is susceptible to various modifications and alternative forms, specific embodiments thereof are shown by way of example in the drawings and will herein be described in detail. It should be understood, however, that the drawings and detailed description thereto are not intended to limit the invention to the particular form disclosed, but on the contrary, the intention is to cover all modifications, equivalents and alternatives falling within the spirit and scope of the present invention as defined by the appended claims. The headings used herein are for organizational purposes only and are not meant to be used to limit the scope of the description. As used throughout this application, the word "may" is used in a permissive sense (i.e., meaning having the potential to), rather than the mandatory sense (i.e., meaning must). Similarly, the words "include", "including", and "includes" mean including, but not limited to.

[0018] Various units, circuits, or other components may be described as "configured to" perform a task or tasks. In such contexts, "configured to" is a broad recitation of structure generally meaning "having circuitry that" performs the task or tasks during operation. As such, the unit/circuit/component can be configured to perform the task even when the unit/circuit/component is not currently on. In general, the circuitry that forms the structure corresponding to "configured to" may include hardware circuits and/or memory storing program instructions executable to implement the operation. The memory can include volatile memory such as static or dynamic random access memory and/or nonvolatile memory such as optical or magnetic disk storage, flash memory, programmable read-only memories, etc. Similarly, various units/circuits/components may be described as performing a task or tasks, for convenience in the description. Such descriptions should be interpreted as including the phrase "configured to." Reciting a unit/circuit/component that is configured to perform one or more tasks is expressly intended not to invoke 35 U.S.C. § 112, paragraph six interpretation for that unit/circuit/component.

DETAILED DESCRIPTION OF EMBODIMENTS

[0019] An exemplary system and integrated circuit are described below in which a level 2 (L2) cache may be powered up or down while the processors are powered down, and control circuitry in a bridge may be configured to perform operations to initialize the cache at power up and/or to prepare the cache for power down. However, other embodiments may implement a similar mechanism to power up/power down any circuit block during times that the processors in the system are powered down. The operations may be configuration register write operations, as discussed below for the L2 cache, or may be other types of operations such as register read operations or commands that are interpreted by the circuit block to change the circuit block's state for power up/power down.

[0020] Generally, a circuit block may include a set of related circuits that implement one or more identifiable functions. The related circuits may be referred to as logic circuits or logic circuitry, since the circuits may implement logic operations on inputs to generate outputs. Because the circuits in a given circuit block are related, they may be powered up or powered down as a unit. Each circuit block may generally be treated as a unit during the design of the integrated circuit (e.g. being physically placed within the integrated circuit as a unit). The circuit block may further include memory circuitry (e.g. various static random access memories, or SRAMs) and other storage devices that are part of the logic circuitry. For example, in an integrated circuit that implements a system on a chip (SOC), the components of the SOC may each be a separate circuit block.

Overview

[0021] Turning now to Fig. 1, a block diagram of one embodiment of a system 5 is shown. In the embodiment of Fig. 1, the system 5 includes an integrated circuit (IC) 10 coupled to external memories 12A-12B. In the illustrated embodiment, the integrated circuit 10 includes a central processor unit (CPU) block 14 which includes one or more processors 16 and a level 2 (L2) cache 18. Other embodiments may not include L2 cache 18 and/or may include additional levels of cache. Additionally, embodiments that include more than two processors 16 and that include only one processor 16 are contemplated. The integrated circuit 10 further includes a set of one or more non-real time (NRT) peripherals 20 and a set of one or more real time (RT) peripherals 22. In the illustrated embodiment, the CPU block 14 is coupled to a bridge/direct memory access (DMA) controller 30, which may be coupled to one or more peripheral devices 32A-32C and/or one or more peripheral interface controllers 34. The number of peripheral devices 32 and peripheral interface controllers 34 may vary from zero to any desired number in various embodiments. The system 5 illustrated in Fig. 1 further includes a graphics unit 36 comprising one or more graphics controllers such as G0 38A and G1 38B. The number of graphics controllers per graphics unit and the number of graphics units may vary in other embodiments. As illustrated in Fig. 1, the system 5 includes a memory controller 40 coupled to one or more memory physical interface circuits (PHYs) 42A-42B. The memory PHYs 42A-42B are configured to communicate on pins of the integrated circuit 10 to the memories 12A-12B. The memory controller 40 also includes a set of ports 44A-44E. The ports 44A-44B are coupled to the graphics controllers 38A-38B, respectively. The CPU block 14 is coupled to the port 44C. The NRT peripherals 20 and the RT peripherals 22 are coupled to the ports 44D-44E, respectively. The number of ports included in a memory controller 40 may be varied in other embodiments, as may the number of memory controllers. That is, there may be more or fewer ports than those shown in Fig. 1. The number of memory PHYs 42A-42B and corresponding memories 12A-12B may be one or more than two in other embodiments.

[0022] Generally, a port may be a communication point on the memory controller 40 to communicate with one or more sources. In some cases, the port may be dedicated to a source (e.g. the ports 44A-44B may be dedicated to the graphics controllers 38A-38B, respectively). In other cases, the port may be shared among multiple sources (e.g. the processors 16 may share the CPU port 44C, the NRT peripherals 20 may share the NRT port 44D, and the RT peripherals 22 may share the RT port 44E). Each port 44A-44E is coupled to an interface to communicate with its respective agent. The interface may be any type of communication medium (e.g. a bus, a point-to-point interconnect, etc.) and may implement any protocol. The interconnect between the memory controller and sources may also include any other desired interconnect such as meshes, network on a chip fabrics, shared buses, point-to-point interconnects, etc.

[0023] The processors 16 may implement any instruction set architecture, and may be configured to execute instructions defined in that instruction set architecture. The processors 16 may employ any microarchitecture, including scalar, superscalar, pipelined, superpipelined, out of order, in order, speculative, non-speculative, etc., or combinations thereof. The processors 16 may include circuitry, and optionally may implement microcoding techniques. The processors 16 may include one or more level 1 caches, and thus the cache 18 is an L2 cache. Other embodiments may include multiple levels of caches in the processors 16, and the cache 18 may be the next level down in the hierarchy. The cache 18 may employ any size and any configuration (set associative, direct mapped, etc.).

[0024] The graphics controllers 38A-38B may be any graphics processing circuitry. Generally, the graphics controllers 38A-38B may be configured to render objects to be displayed into a frame buffer. The graphics controllers 38A-38B may include graphics processors that may execute graphics software to perform a part or all of the graphics operation, and/or hardware acceleration of certain graphics operations. The amount of hardware acceleration and software implementation may vary from embodiment to embodiment.

[0025] The NRT peripherals 20 may include any non-real time peripherals that, for performance and/or bandwidth reasons, are provided independent access to the memory 12A-12B. That is, access by the NRT peripherals 20 is independent of the CPU block 14, and may proceed in parallel with CPU block memory operations. Other peripherals such as the peripherals 32A-32C and/or peripherals coupled to a peripheral interface controlled by the peripheral interface controller 34 may also be non-real time peripherals, but may not require independent access to memory. Various embodiments of the NRT peripherals 20 may include video encoders and decoders, scaler circuitry and image compression and/or decompression circuitry, etc.

[0026] The RT peripherals 22 may include any peripherals that have real time requirements for memory latency. For example, the RT peripherals may include an image processor and one or more display pipes. The display pipes may include circuitry to fetch one or more frames and to blend the frames to create a display image. The display pipes may further include one or more video pipelines. The result of the display pipes may be a stream of pixels to be displayed on the display screen. The pixel values may be transmitted to a display controller for display on the display screen. The image processor may receive camera data and process the data to an image to be stored in memory.

[0027] The bridge/DMA controller 30 may comprise circuitry to bridge the peripheral(s) 32 and the peripheral interface controller(s) 34 to the memory space. In the illustrated embodiment, the bridge/DMA controller 30 may bridge the memory operations from the peripherals/peripheral interface controllers through the CPU block 14 to the memory controller 40. The CPU block 14 may also maintain coherence between the bridged memory operations and memory operations from the processors 16/L2 Cache 18. The L2 cache 18 may also arbitrate the bridged memory operations with memory operations from the processors 16 to be transmitted on the CPU interface to the CPU port 44C. The bridge/DMA controller 30 may also provide DMA operation on behalf of the peripherals 32 and the peripheral interface controllers 34 to transfer blocks of data to and from memory. More particularly, the DMA controller may be configured to perform transfers to and from the memory 12A-12B through the memory controller 40 on behalf of the peripherals 32 and the peripheral interface controllers 34. The DMA controller may be programmable by the processors 16 to perform the DMA operations. For example, the DMA controller may be programmable via descriptors. The descriptors may be data structures stored in the memory 12A-12B that describe DMA transfers (e.g. source and destination addresses, size, etc.). Alternatively, the DMA controller may be programmable via registers in the DMA controller (not shown).

[0028] The peripherals 32A-32C may include any desired input/output devices or other hardware devices that are included on the integrated circuit 10. For example, the peripherals 32A-32C may include networking peripherals such as one or more networking media access controllers (MAC) such as an Ethernet MAC or a wireless fidelity (WiFi) controller. An audio unit including various audio processing devices may be included in the peripherals 32A-32C. One or more digital signal processors may be included in the peripherals 32A-32C. The peripherals 32A-32C may include any other desired functional such as timers, an on-chip secrets memory, an encryption engine, etc., or any combination thereof.

[0029] The peripheral interface controllers 34 may include any controllers for any type of peripheral interface. For example, the peripheral interface controllers may include various interface controllers such as a universal serial bus (USB) controller, a peripheral component interconnect express (PCIe) controller, a flash memory interface, general purpose input/output (I/O) pins, etc.

[0030] The memories 12A-12B may be any type of memory, such as dynamic random access memory (DRAM), synchronous DRAM (SDRAM), double data rate (DDR, DDR2, DDR3, etc.) SDRAM (including mobile versions of the SDRAMs such as mDDR3, etc., and/or low power versions of the SDRAMs such as LPDDR2, etc.), RAMBUS DRAM (RDRAM), static RAM (SRAM), etc. One or more memory devices may be coupled onto a circuit board to form memory modules such as single inline memory modules (SIMMs), dual inline memory modules (DIMMs), etc.

Alternatively, the devices may be mounted with the integrated circuit 10 in a chip-on-chip configuration, a package-on-package configuration, or a multi-chip module configuration.

[0031] The memory PHYs 42A-42B may handle the low-level physical interface to the memory 12A-12B. For example, the memory PHYs 42A-42B may be responsible for the timing of the signals, for proper clocking to synchronous DRAM memory, etc. In one embodiment, the memory PHYs 42A-42B may be configured to lock to a clock supplied within the integrated circuit 10 and may be configured to generate a clock used by the memory 12.

[0032] It is noted that other embodiments may include other combinations of components, including subsets or supersets of the components shown in Fig. 1 and/or other components. While one instance of a given component may be shown in Fig. 1, other embodiments may include one or more instances of the given component. Similarly, throughout this detailed description, one or more instances of a given component may be included even if only one is shown, and/or embodiments that include only one instance may be used even if multiple instances are shown.

L2 Cache Power Up/Power Down

[0033] Turning now to Fig. 2, a block diagram of one embodiment of a portion of the integrated circuit 10 is shown in greater detail. Particularly, the CPU block 14 and the bridge/DMA controller 30 are shown along with a power manager 50. The CPU block 14 includes the processors 16 and the L2 cache 18. In the embodiment of Fig. 2, the L2 cache 18 is illustrated as the L2 cache control 18A and the L2 cache memory 18B. The L2 cache control 18A may include a cache control circuit 52 and a coherence control circuit 54. Each of the cache control circuit 52 and the coherence control circuit 54 may include configuration registers such as configuration registers 56A-56D. The processors 16 are coupled to the L2 cache control 18A, and more particularly to the coherence control circuit 54. The coherence control circuit 54 may be coupled to the cache control circuit 52. The L2 cache control 18A, and more particularly the cache control circuit 52, may be coupled to the L2 cache memory 18B. The L2 cache control 18A may further be coupled the memory controller 40 (e.g. the CPU port 44C in Fig.

1). The power manager 50 may be coupled to the L2 cache control 18A (e.g. the L2 power control signals in Fig. 2) and the processors 16 (e.g. the processor power control signals in Fig. 2).

[0034] The bridge/DMA controller 30 may include a coherent I/O interface unit (CIF) 58, a power up/power down memory 60, and a DMA controller 62. The CIF 58 is coupled to the power up/power down memory 60, to the DMA controller 62, to the L2 cache control 18A (and more particularly to the coherence control circuit 54), to the power manager 50 (e.g. via the I/O idle, PwrUpReq, and PwrUpAck signals in Fig. 2), and to the peripherals 32A-32C and/or the peripheral interface controllers 34. The DMA controller 62 is further coupled to the peripherals 32A-32C and the peripheral interface controllers 34. In an embodiment, the DMA controller 62 and the CIF 58 may be coupled to respective subsets of the peripherals 32A-32C and/or the peripheral interface controllers 34. The subsets may overlap (e.g. some peripherals/peripheral interface controllers may be configured to communicate with memory both through DMA and through direct communications with the CIF 58). Other peripherals/peripheral interface controllers may communicate with memory only through DMA or only through operations directly transmitted to the CIF 58.

[0035] The configuration registers 56A-56D may be programmed by software to control various aspects of the operation of the cache control circuit 52 and the coherence control circuit 54. Generally, circuit blocks may implement configuration registers to permit software to select among various programmable configurations. For example, the size and configuration of the L2 cache 18 may be selectable within certain predefined maximums. The writethrough/writeback operation of the cache may be configured. The coherence mode may be enabled and controlled through configuration registers 56A-56D. In some embodiments, only the cache control circuit 52 may include cache configuration registers 56A-56D or only the coherence control circuit 54 may include cache configuration registers 56A-56D.

[0036] If the L2 cache 18 is powered down, the configuration data stored in at least some of the configuration registers 56A-56D is lost. To restore the configuration after a power down and subsequent power up of the L2 cache 18, the configuration data may be stored in the power up/power down memory 60. For example, when software programs a configuration register 56A-56D with a value that is also to be restored on power up, software may also write the value to the power up/power down memory 60. Similarly, there may be configuration register writes or other register writes to be performed prior to power down. For example, a register may be written with a synchronization command to synchronize the L2 cache 18 (ensuring that any outstanding memory operations or other communications are complete) prior to powering down.

[0037] The CIF 58 may be configured to read the operations from the power up/power down memory 60 during power up or power down events. The power manager 50 may be configured to signal a power up or power down event to the CIF 58, and the CIF 58 may be configured to read the memory 60 and transmit the operations for the corresponding event to the L2 cache 18. Once the operations are complete, the CIF 58 may be configured to communicate the completion to the power manager 50. In response, the power manager 50 may complete the power up/power down event.

[0038] Any communication may be implemented between the power manager 50 and the CIF 58. In the illustrated embodiment, the power manager 50 may signal a power up or power down event using the PwrUpReq signal. More specifically, the power manager 50 may be configured to assert the PwrUpReq signal to indicate that the L2 cache 18 is being powered up, and may be configured to deassert the PwrUpReq signal to indicate that the L2 cache 18 is being powered down. In response to the assertion of the PwrUpReq signal, the CIF 58 may be configured to read any operations in the power up/power down memory 60 that are indicated as power up operations, and may be configured to communicate the operations to the L2 cache 18. The CIF 58 may be configured to determine that the operations are complete (e.g. receiving write completions corresponding to each register write operation), and the CIF 58 may be configured to assert the PwrUpAck signal to acknowledge the power up event. The power manager 50 may be configured to re-enable communication to the L2 cache 18 responsive to the assertion of the PwrUpAck signal.

[0039] Powering down the L2 cache 18 may include at least powering down the cache control circuit 52. In some embodiments, the coherence control circuit 54 may also be powered down. The L2 cache memory 18B may remain powered on in some embodiments, retaining cache state in the cache (e.g. various cache blocks from the memory, state of the cache blocks such as tags, validity, and coherence state, etc.). Alternatively, the L2 cache memory 18B may also be powered down as part of powering down the L2 cache 18. Any circuitry/memory that was powered down may be powered up again in response to a power up event.

[0040] The power up/power down memory 60 may be formed from any semiconductor storage. For example, multiple registers may be provided that may be read/written by software. Other embodiments may use other forms of storage (e.g. random access memory (RAM) such as static RAM).

[0041] The power up/power down memory 60 may generally include multiple entries. Two exemplary entries are illustrated in the memory 60 in Fig. 2. In the illustrated embodiment, each entry in the power up/power down memory 60 may include an address and data pair, illustrated as the A field and the Data field in the entries of Fig. 2. The address may identify the configuration register to be written, and the data may be the value to be written to the configuration register. The address may be relative (e.g. the address may be an offset from a base address corresponding to the L2 cache control 18A, or base addresses for the coherence control circuit 54 and/or the cache control circuit 52, more specifically). Alternatively, the address may be the full address that would be transmitted by the processor 16 in a write operation to the corresponding register 56A-56D. Each entry may also include a valid bit (V) indicating whether or not the entry is storing valid information. Additionally, in this embodiment, each entry may include a power down (D) field which indicates whether the configuration register write is performed during power down or during power up. The D bit in the D field may be set to indicate a power down register write, and clear to indicate a power up register write. Other embodiments may use different memories for power down and power up, or may divide the memory in a known fashion, and the D field may not be included in each entry.

[0042] In an embodiment, software may be expected to write the power down address/data pairs in the initial entries of the power up/power down memory 60 and to write the power up address/data pairs in subsequent entries. In such an embodiment, in response to a power down event, the CIF 58 may read operations beginning with the initial entry until an entry having the D bit cleared is encountered. The CIF 58 may retain a pointer to the entry, and may begin reading power up operations from the indicated entry in response to a power up event (after which the pointer may be reset to point to the initial entry again).

[0043] While the power up/power down memory 60 may store configuration register writes, other embodiments may store any type of operations to be performed (e.g. register writes, register reads, commands, etc.). Accordingly, a flexible mechanism for powering up and powering down the L2 cache 18 may be supported. The mechanism may support powering the L2 cache 18 up or down while the processors 16 are powered down (and without waking the processors 16). Additionally, because the operations are programmable in the memory 60, the operations to be performed may be changed and the order of the operations may be changed. Accordingly, the mechanism may be corrected (if operating erroneously) via software changes even though the mechanism itself operates in hardware.

[0044] The cache control circuit 52 may generally be configured to manage access to the L2 cache memory 18B. The cache control circuit 52 may detect hit/miss for cache accesses, initiate cache fills for misses, manage the replacement policy in the L2 cache 18, etc. The coherence control circuit 54 may control cache coherence in the CPU block 14 for processor 16 memory operations and for memory operations from the CIF 58 (e.g. DMA operations from the DMA controller 62 and/or other memory operations received directly from the peripherals 32A-32C and/or the peripheral interface controllers 34). The coherence control 54 may maintain snoop tags for the caches in the processors 16, and may also be configured to generate cache accesses to the cache control circuit 52 to snoop the L2 cache memory 18B for cache coherence purposes.

[0045] The power manager 50 may be configured to monitor the processors 16 and the L2 cache 18, as well as various other activity in the integrated circuit 10 (not shown in

Fig. 2). The power manager 50 may control the power state of the processors 16, including power the processors 16 up or down, via the processor power control signals. The processors 16 may be powered up and down independently or in synchronization in various embodiments.

[0046] The power manager 50 may be configured to power down the L2 cache 18 as well, if the L2 cache 18 is idle. The power manager 50 may detect that the L2 cache 18 is idle in a variety of fashions. For example, the power manager 50 may be aware that the processors 16 are powered down, and thus no memory operations may be expected from the processors 16. Additionally, the power manager 50 may detect that the bridge/DMA controller 30 is idle, at least with respect to memory operations. In the illustrated embodiment, the CIF 58 may generate an I/O idle signal. The CIF 58 may assert the I/O idle signal to indicate that there are no memory operations pending from the peripherals 32A-32C and/or the peripheral interface controllers 34, including no memory operations from the DMA controller 62. In an embodiment, the CIF 58 may detect that there are no memory operations for a programmable number of consecutive clock cycles before asserting the I/O idle signal to the power manager 50. In one embodiment, if the CIF 58 has asserted the I/O idle signal and subsequently receives a memory operation, the CIF 58 may be configured to deassert the I/O idle signal. However, the CIF 58 may await an indication from the power manager 50 that the memory operations can be transmitted. The indication may avoid a race condition in which the power manager 50 has started a power down event prior to the receipt of the memory operation, and thus may prevent the transmission of the memory operation to the L2 cache 18 where it could be lost as part of the power down event. In an embodiment, a ready signal (not shown) may be provided by the power manager 50 to indicate that the L2 cache 18 is ready for the memory operation after the assertion (and deassertion) of the I/O idle signal.

[0047] Turning next to Figs. 3-6, flowcharts are shown illustrating operation of one embodiment of the power manager 50 and the CIF 58 for power up and power down events for the L2 cache 18. While the blocks are shown in a particular order for ease of understanding, other orders may be used. Blocks may be performed in parallel in combinatorial logic circuitry in the power manager 50 and/or the CIF 58. Blocks, combinations of blocks, and/or a flowchart as a whole may be pipelined over multiple clock cycles. The power manager 50 and/or the CIF 58 may be configured to implement the operation illustrated in the flowchart. More particularly, the power manager 50 and/or the CIF 58 may include hardware circuitry that implements the operation illustrated.

[0048] Fig. 3 is a flowchart illustrating operation of one embodiment of the power manager 50 for a power down event. The power manager 50 may determine that a power down event is to occur if the processors 16 are powered down (decision block 70, "yes" leg) and if the CIF 58 has signalled I/O idle (decision block 72, "yes" leg). If so, the power manager 50 may deassert the PwrUpReq signal to the CIF 58 (block 74), initiating the power down event. The power manager 50 may await the acknowledgement from the CIF 58 (decision block 76), and in response to a deassertion of the PwrUpAck (decision block 76, "yes" leg), the power manager 50 may power down the L2 cache (block 78).

[0049] Fig. 4 is a flowchart illustrating operation of one embodiment of the CIF 58 for a power down event. CIF 58 processing of the power down event may begin in response to deassertion of the PwrUpReq signal from the power manager 50 (decision block 80, "yes" leg). The CIF 58 may read an initial entry from the power up/power down memory 60 (block 82), and may determine if the entry is valid and is for a power down event (V and D set, decision block 84). If so (decision block 84, "yes" leg), the CIF 58 may transmit the address/data pair to the L2 control 18A to update the identified configuration register 56A-56D (block 86) and may read the next entry in the memory 60 (block 82). If not (decision block 84, "no" leg), the CIF 58 may determine if the write responses for all of the configuration register writes have been received from the L2 control 18A (decision block 88). If the responses have been received (decision block 88, "yes" leg), the L2 control 18A may be prepared for power down and the CIF 58 may deassert the PwrUpAck signal to acknowledge the power down request (block 90).

[0050] Fig. 5 is a flowchart illustrating operation of one embodiment of the power manager 50 for a power up event. The power manager 50 may determine that a power up event is to occur if the processors 16 are to be powered up, or if a memory operation is received in the CIF 58 (causing the I/O idle to deassert). The power manager 50 may power up the L2 cache control (block 100) and may wait for the power to stabilize. The power manager 50 may assert the PwrUpReq signal (block 102), and may wait for the PwrUpAck signal to be asserted (decision block 104) to determine that the L2 cache 18 is initialized and ready for communication again.

[0051] Fig. 6 is a flowchart illustrating operation of one embodiment of the CIF 58 for a power up event. CIF 58 processing of the power up event may begin in response to assertion of the PwrUpReq signal (decision block 110, "yes" leg). The CIF 58 may read the next entry in the power up/power down memory 60 (block 112). If the entry is valid and a power up operation (V set and D clear, decision block 114, "yes" leg), the CIF 58 may transmit the configuration register write to the L2 cache control 18A (block 116) and may read the next entry in the memory 60 (block 112). If the entry is not valid or is a power down operation (decision block 114, "no" leg), the CIF 58 may determine if the responses for the register writes have been received (decision block 118). If so (decision block 118, "yes" leg), the CIF 58 may assert the PwrUpAck signal (block 120).

[0052] In some embodiments, the power manager 50 may determine that the L2 cache 18 is to power down during the processing of the power up event, or may determine that the L2 cache 18 is to power up during the processing of the power down event. In some implementations, the power manager 50 may be configured to permit the in-progress transition to complete prior to initiating the new transition. In other implementations, the power manager 50 may be configured to signal the new transition upon determination (e.g. by changing the state of the PwrUpReq signal). The CIF 58 may be configured to monitor the PwrUpReq signal to detect the change of state, and may cease processing the in-progress event. The CIF 58 may either be configured to acknowledge the changed state without further processing, or may process the new event (performing the register writes for the new event).

[0053] Fig. 7 is a timing diagram illustrating a power down and power up sequence for the L2 cache 18, for one embodiment. Time may increase from left to right in Fig. 7, in arbitrary units. The L2 cache 18 may be powered up and operating at the beginning of the timing diagram (block 130), and the PwrUpReq and PwrUpAck signals are both asserted. The power manager 50 may determine that the L2 cache is to be powered down, and may deassert the PwrUpReq signal (dotted line 132). The CIF 58 may begin transmitting register writes and collecting responses (block 134). Once the writes are complete and the responses are received, the CIF 58 may deassert the PwrUpAck (dotted line 136) and the L2 cache 18 may be powered down (block 138). At a later point, the power manager 50 may determine that the L2 cache 18 is to be powered up, and may assert the PwrUpReq signal after establishing power to the L2 cache 18 (dotted line 140). The CIF 58 may transmit register writes to initialize the configuration registers (block 142), and may assert the PwrUpAck signal in response to completing the writes and receiving the responses (dotted line 144). The L2 cache 18 may be powered up and operating again at this point (block 146).

[0054] Turning now to Fig. 8, a flowchart is shown illustrating one embodiment of software that may update the L2 cache configuration. For example, the software may include L2 configuration code that may execute during boot of the system 5 and/or at other times during operation of the system that the L2 cache configuration is changed. The L2 configuration code may be executed on one of the processors 16 to implement the operation shown in Fig. 8. That is, the L2 configuration code may include instructions which, when executed by one of the processors 16, implements the operation shown in Fig. 8. While the blocks are shown in a particular order for ease of understanding, other orders may be used.

[0055] The L2 configuration code may write one or more configuration registers in the L2 cache 18 (block 150). If one or more of the configuration registers also need to be written during a power down of the L2 cache 18 (decision block 152, "yes" leg), the code may write the address of the configuration register and the corresponding data to the power up/power down memory 60 (block 154). For example, a register write that causes a synchronization of the L2 cache 18 may be included. The code may set the V and D bits in each entry written with a power down write. If one or more of the configuration registers are to be recovered during a power up of the L2 cache 18 (decision block 156, "yes" leg), the code may write the address of the configuration register and the corresponding data to the power up/power down memory 60 and may set the V bit and clear the D bit in each entry (block 158). It is noted that the same configuration register may be included in both the power down writes and the power up writes, in some embodiments.

[0056] In another embodiment, the CIF 58 may be configured to detect writes to configuration registers 56A-56D (or subsets of the configuration registers that are to be restored on power up events and/or written on power down events). The CIF 58 may automatically capture the values written to the registers in the power up/power down memory 60, and thus the L2 configuration code need not perform the writes to the memory 60 explicitly. In some such embodiments, the L2 configuration code may also be able to update the memory 60, in addition to the above-mentioned automatic capture. The L2 configuration code may insert the synchronization command for power down events, for example.

[0057] Turning next to Fig. 9, a block diagram of a computer accessible storage medium 200 is shown. Generally speaking, a computer accessible storage medium may include any storage media accessible by a computer during use to provide instructions and/or data to the computer. For example, a computer accessible storage medium may include storage media such as magnetic or optical media, e.g., disk (fixed or removable), tape, CD-ROM, or DVD-ROM, CD-R, CD-RW, DVD-R, DVD-RW, or Blu-Ray. Storage media may further include volatile or non-volatile memory media such as RAM (e.g. synchronous dynamic RAM (SDRAM), double data rate (DDR, DDR2, DDR3, etc.) SDRAM, low-power DDR (LPDDR2, etc.) SDRAM, Rambus DRAM (RDRAM), static RAM (SRAM), etc.), ROM, Flash memory, non-volatile memory (e.g. Flash memory) accessible via a peripheral interface such as the Universal Serial Bus (USB) interface, etc. Storage media may include microelectromechanical systems (MEMS), as well as storage media accessible via a communication medium such as a network and/or a wireless link. The computer accessible storage medium 200 in Fig. 9 may store L2 configuration code 202, which may implement the flowchart of Fig. 8. Generally, the computer accessible storage medium 200 may store any set of instructions which, when executed, implement a portion or all of the flowchart shown in Fig. 8. A carrier medium may include computer accessible storage media as well as transmission media such as wired or wireless transmission.

[0058] Turning now to Fig. 10, a block diagram of one embodiment of a system 350 is shown. In the illustrated embodiment, the system 350 includes at least one instance of an integrated circuit 10 coupled to an external memory 352. The external memory 352 may form the main memory subsystem discussed above with regard to Fig. 1 (e.g. the external memory 352 may include the memory 12A-12B). The integrated circuit 10 is coupled to one or more peripherals 354 and the external memory 352. A power supply 356 is also provided which supplies the supply voltages to the integrated circuit 358 as well as one or more supply voltages to the memory 352 and/or the peripherals 354. In some embodiments, more than one instance of the integrated circuit 10 may be included (and more than one external memory 352 may be included as well).

[0059] The memory 352 may be any type of memory, such as dynamic random access memory (DRAM), synchronous DRAM (SDRAM), double data rate (DDR, DDR2, DDR3, etc.) SDRAM (including mobile versions of the SDRAMs such as mDDR3, etc., and/or low power versions of the SDRAMs such as LPDDR2, etc.), RAMBUS DRAM (RDRAM), static RAM (SRAM), etc. One or more memory devices may be coupled onto a circuit board to form memory modules such as single inline memory modules (SIMMs), dual inline memory modules (DIMMs), etc. Alternatively, the devices may be mounted with an integrated circuit 10 in a chip-on-chip configuration, a package-on-package configuration, or a multi-chip module configuration.

[0060] The peripherals 354 may include any desired circuitry, depending on the type of system 350. For example, in one embodiment, the system 350 may be a mobile device (e.g. personal digital assistant (PDA), smart phone, etc.) and the peripherals 354 may include devices for various types of wireless communication, such as wifi, Bluetooth, cellular, global positioning system, etc. The peripherals 354 may also include additional storage, including RAM storage, solid state storage, or disk storage. The peripherals 354 may include user interface devices such as a display screen, including touch display screens or multitouch display screens, keyboard or other input devices, microphones, speakers, etc. In other embodiments, the system 350 may be any type of computing system (e.g. desktop personal computer, laptop, workstation, net top etc.).

[0061] Numerous variations and modifications will become apparent to those skilled in the art once the above disclosure is fully appreciated. It is intended that the following claims be interpreted to embrace all such variations and modifications.

Conclusies

1. Inrichting, omvattend:

- een geheugen, geconfigureerd voor het opslaan van data die een eerste stel operaties identificeren welke dienen te worden uitgevoerd voorafgaand aan het uitschakelen van de bedrijfsspanning van een schakelingsblok en een tweede stel operaties welke dienen te worden uitgevoerd na het inschakelen van de bedrijfsspanning van het schakelingsblok;

- een besturingsschakeling, gekoppeld voor het ontvangen van een verzoek om de bedrijfsspanning van het schakelingsblok uit te schakelen gedurende een tijd waarin de bedrijfsspanning van processors in een systeem dat het geheugen en de besturingsschakeling omvat is uitgeschakeld, waarbij de besturingsschakeling is geconfigureerd voor het uitvoeren van het overeenkomstige eerste stel operaties dat is geïdentificeerd in het geheugen in responsie op het verzoek.

2. Inrichting volgens conclusie 1, waarbij het eerste stel operaties een eerste veelheid van registerschrijfoperaties omvat voor een eerste veelheid van registers in het schakelingsblok, en waarbij de data in het geheugen een eerste veelheid van adressen omvatten welke de veelheid van registers en een eerste veelheid van naar de eerste veelheid van registers te schrijven data identificeert, en waarbij het tweede stel operaties een tweede veelheid van registerschrijfoperaties omvat voor een tweede veelheid van registers in het schakelingsblok, en waarbij de data in het geheugen een tweede veelheid van adressen omvatten welke de tweede veelheid van registers en een tweede veelheid van naar de tweede veelheid van registers te schrijven data identificeert.

3. Inrichting volgens conclusie 2, waarbij een eerste register is opgenomen in zowel de eerste veelheid van registers als de tweede veelheid van registers.

4. Inrichting volgens conclusie 1, waarbij het geheugen één of meer registers omvat die programmeerbaar zijn door middel van software met het eerste stel operaties en het tweede stel operaties.

5. Inrichting volgens conclusie 1, waarbij de besturingsschakeling verder is geconfigureerd voor het ontvangen van een verzoek om het bedrijfsvermogen van een schakelingsblok in te schakelen gedurende een tijd waarin het bedrijfsvermogen van de processors in het systeem is uitgeschakeld, en waarbij de besturingsschakeling is geconfigureerd voor het uitvoeren van het overeenkomstige tweede stel operaties dat is geïdentificeerd in het geheugen in responsie op het verzoek om het bedrijfsvermogen van het schakelingsblok in te schakelen.

6. Systeem, omvattend:

- één of meer processors;

- de inrichting volgens conclusie 1, waarbij het schakelingsblok een cache is, welke is gekoppeld aan de één of meer processors; en

- waarbij de besturingsschakeling en het geheugen zijn opgenomen in een bridge die is gekoppeld aan de cache en is geconfigureerd om te zijn gekoppeld aan één of meer perifere inrichtingen, waarbij het geheugen een veelheid van registers omvat welke programmeerbaar zijn met data die het eerste stel operaties en het tweede stel operaties vertegenwoordigen, en waarbij de bridge is geconfigureerd voor het uitvoeren van het eerste stel operaties in responsie op een bedrijfsvermogen-uitschakelgebeurtenis voor de cache en voor het uitvoeren van het tweede stel operaties in responsie op een bedrijfsvermogen-inschakelgebeurtenis voor de cache.

7. Systeem volgens conclusie 6, verder omvattend een vermogenbeheerorgaan, geconfigureerd voor het opwekken van een bedrijfsvermogen-uitschakelgebeurtenis in responsie op detecteren dat het bedrijfsvermogen van de één of meer processors is uitgeschakeld, en verder in responsie op het detecteren dat er geen operaties gaande zijn vanuit de één of meer perifere inrichtingen.

8. Systeem volgens conclusie 7, waarbij de bridge is geconfigureerd teneinde de bedrijfsvermogen-uitschakelgebeurtenis te bevestigen in responsie op het voltooien van de eerste veelheid van operaties.

9. Systeem volgens conclusie 7, waarbij het vermogenbeheerorgaan is

geconfigureerd voor het opwekken van de bedrijfsvermogen-inschakelgebeurtenis, en waarbij de brug is geconfigureerd voor het bevestigen van de bedrijfsvermogen-inschakelgebeurtenis in responsie op het voltooien van de tweede veelheid van operaties.

10. Werkwijze, omvattend:

- het detecteren dat het bedrijfsvermogen van een schakelingsblok dient te worden ingeschakeld in een systeem dat één of meer processors omvat welke zijn gekoppeld aan het schakelingsblok, waarbij het bedrijfsvermogen van de één of meer processors is uitgeschakeld op het moment van het detecteren;

- het afgeven van een verzoek aan een besturingsschakeling welke is gekoppeld aan het schakelingsblok en één of meer perifere inrichtingen; en

- het door de besturingsschakeling reageren op het verzoek door het uitvoeren van een veelheid van operaties welke zijn opgeslagen in de besturingsschakeling teneinde één of meer configuratieregisters in het schakelingsblok te initialiseren.

11. Werkwijze volgens conclusie 10, waarbij iedere operatie is gerepresenteerd door een adres van het configuratieregister en een naar het configuratieregister te schrijven waarde.

12. Werkwijze volgens conclusie 10, verder omvattend:

- het detecteren dat het bedrijfsvermogen van het schakelingsblok dient te worden uitgeschakeld;

- het afgeven van een tweede verzoek aan de besturingsschakeling; en

- het door de besturingsschakeling reageren op het tweede verzoek door het uitvoeren van één of meer tweede operaties welke zijn opgeslagen in de besturingsschakeling.

13. Werkwijze volgens conclusie 12, waarbij de één of meer tweede operaties een synchronisatie-operatie in het schakelingsblok doen plaatsvinden.

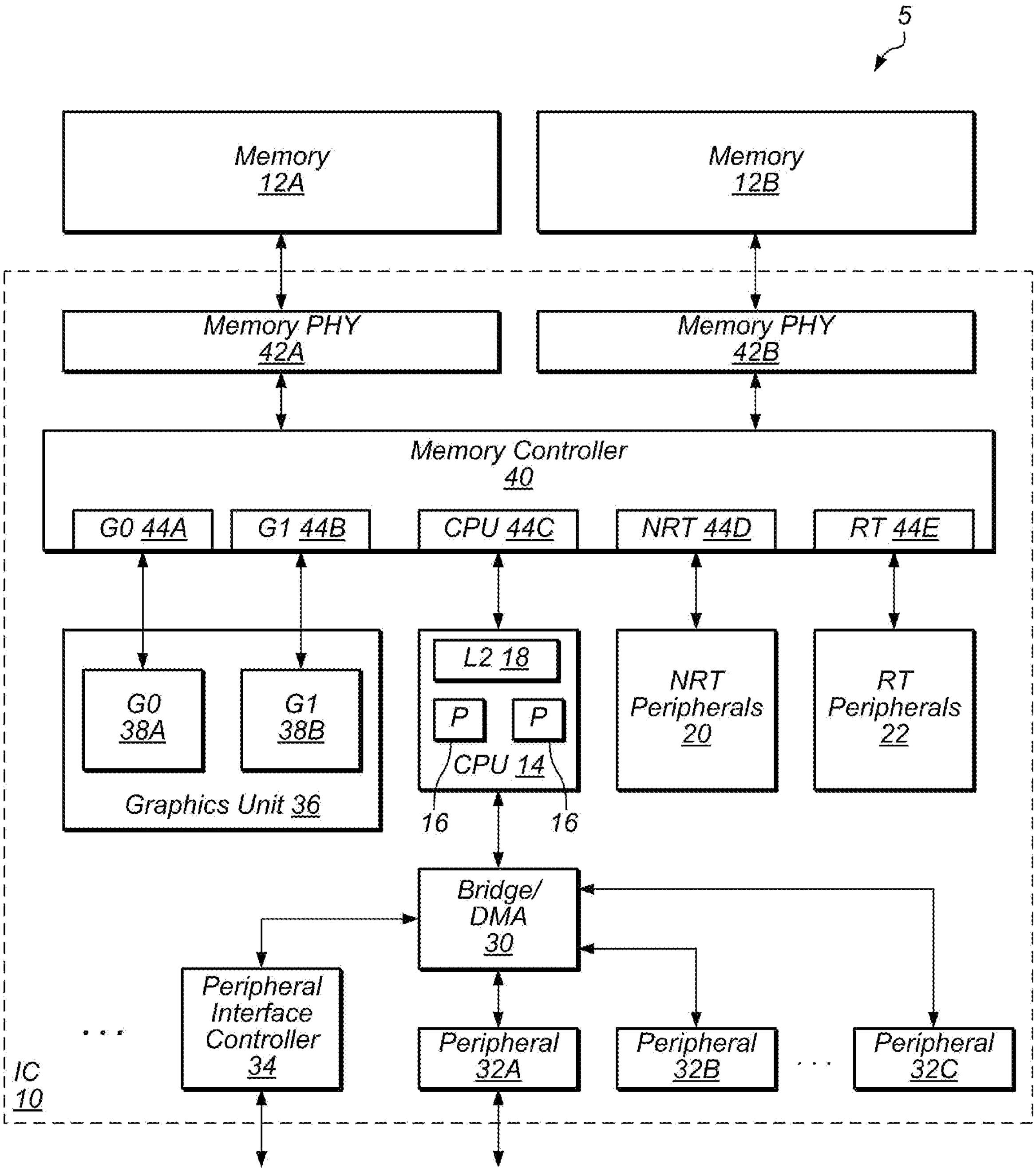
14. Werkwijze volgens conclusie 12, verder omvattend:

- het door de besturingsschakeling vaststellen dat de één of meer tweede operaties volledig zijn;

- het door de besturingsschakeling bevestigen van het tweede verzoek in responsie op de vaststelling; en

- het uitschakelen van het bedrijfsvermogen van het schakelingsblok in responsie op bevestiging door de bridge.

Fig. 1



5
Memory 12A
Memory 12B
Memory PHY 42A
Memory PHY 42B
Memory Controller 40
G0 44A
G1 44B
CPU 44C
NRT 44D
RT 44E
G0 38A
G1 38B
Graphics Unit 36
L2 18
P
P
CPU 14
16
16
NRT Peripherals 20
RT Peripherals 22
Bridge/ DMA 30
Peripheral Interface Controller 34
Peripheral 32A
Peripheral 32B
Peripheral 32C
IC 10

Fig. 2

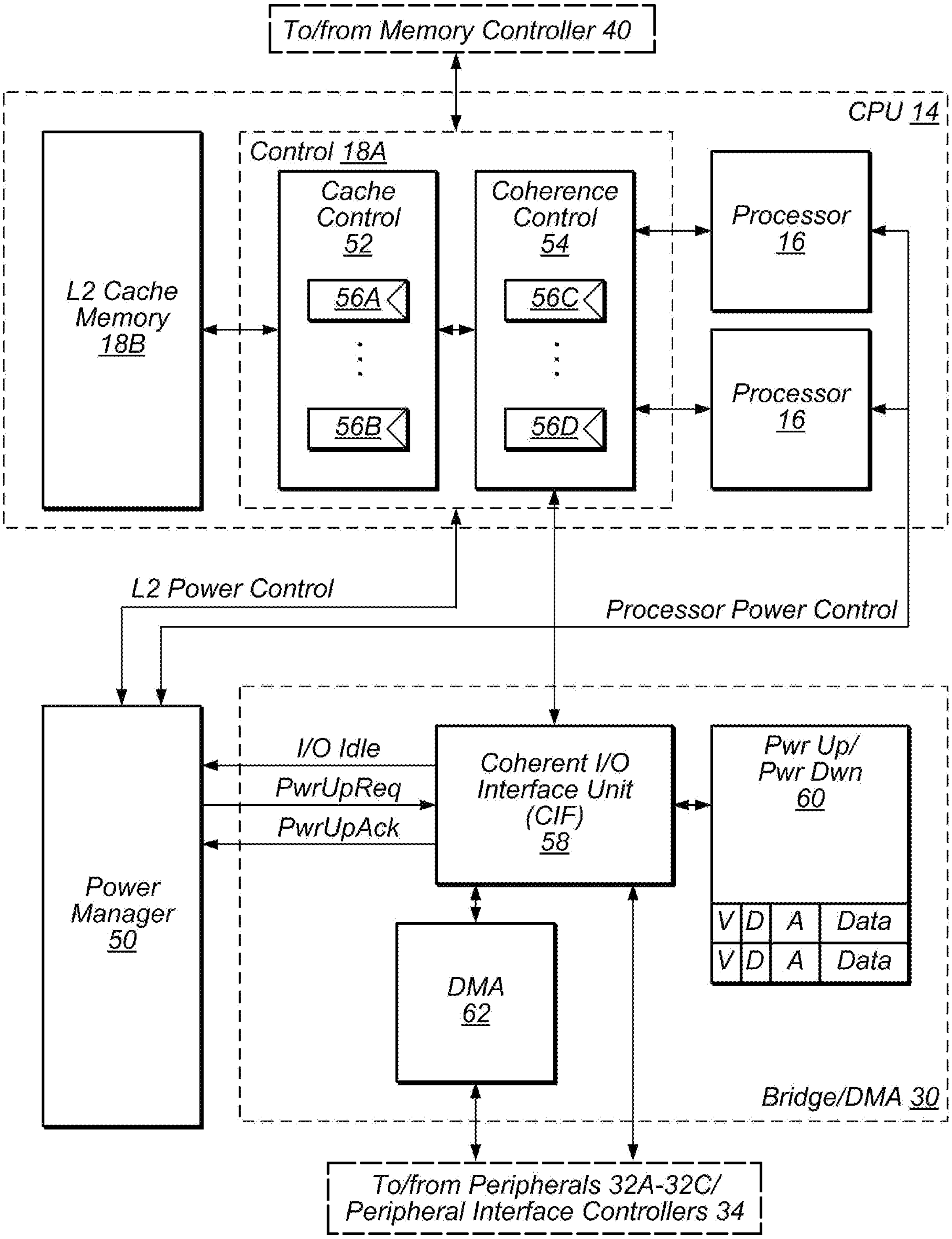
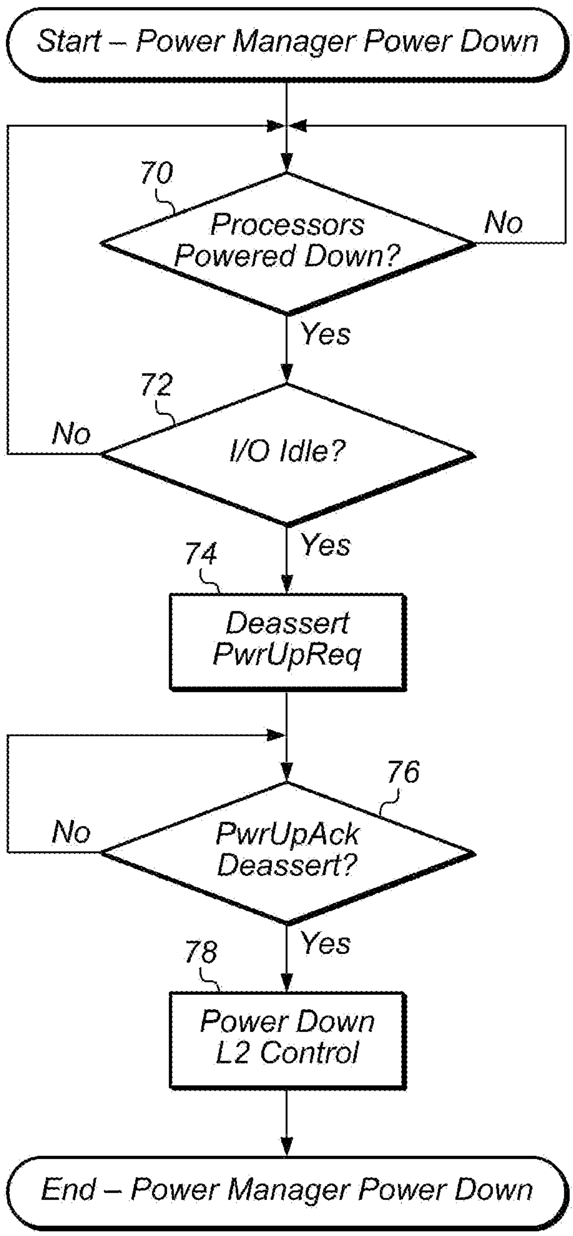
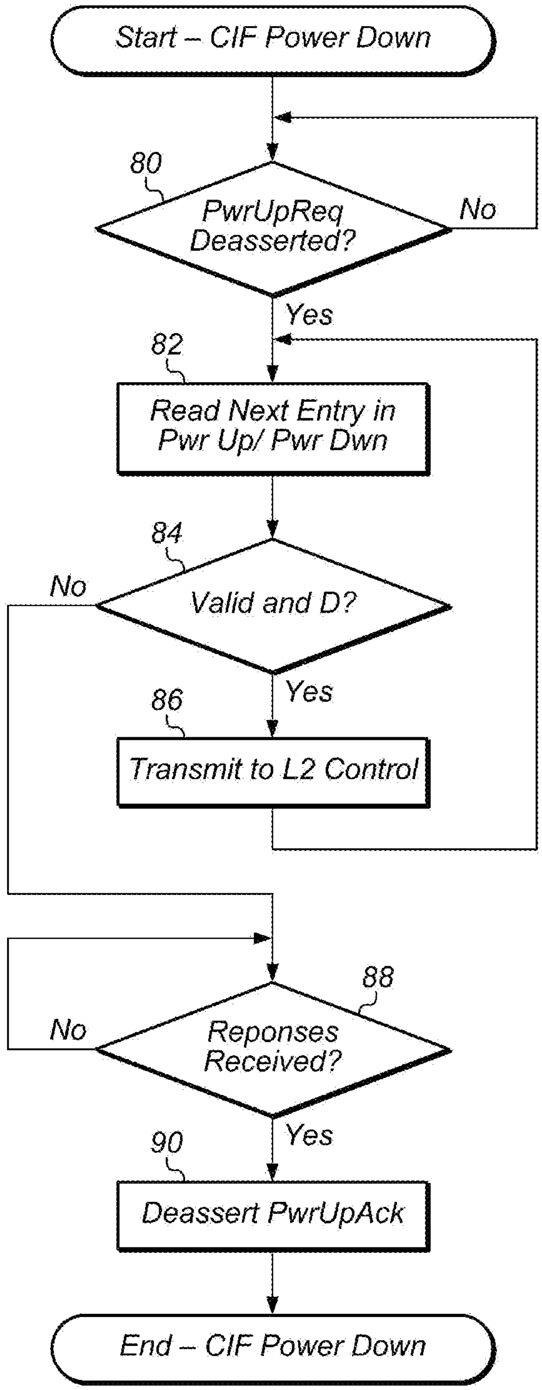

To/from Memory Controller 40
CPU 14
Control 18A
L2 Cache Memory 18B
Cache Control 52
56A
56B
Coherence Control 54
56C
56D
Processor 16
Processor 16
L2 Power Control
Processor Power Control
Power Manager 50
I/O Idle
PwrUpReq
PwrUpAck
Coherent I/O Interface Unit (CIF) 58
Pwr Up/ Pwr Dwn 60
V
D
A
Data
V
D
A
Data
DMA 62
Bridge/DMA 30
To/from Peripherals 32A-32C/ Peripheral Interface Controllers 34

Fig. 3



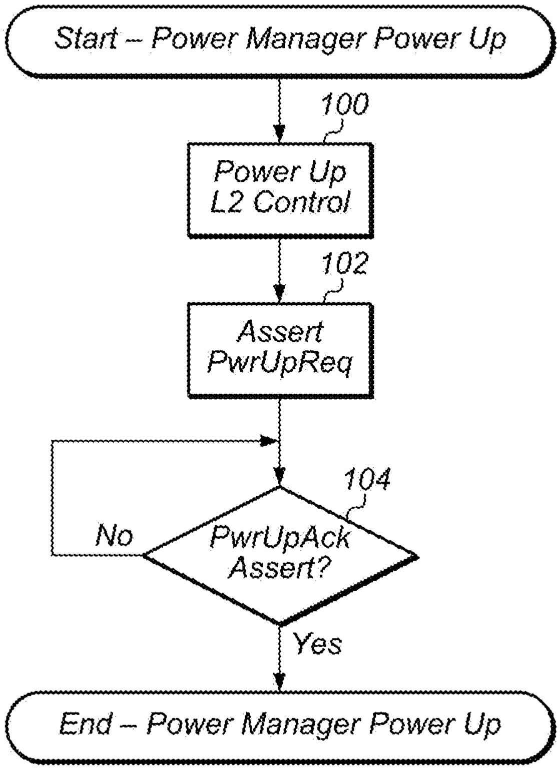
Start – Power Manager Power Down
70
Processors Powered Down?
No
Yes
72
No
I/O Idle?
Yes
74
Deassert PwrUpReq
76
No
PwrUpAck Deassert?
Yes
78
Power Down L2 Control
End – Power Manager Power Down

Fig. 4



Start – CIF Power Down
80
PwrUpReq Deasserted?
No
Yes
82
Read Next Entry in Pwr Up/ Pwr Dwn
84
No
Valid and D?
Yes
86
Transmit to L2 Control
88
No
Reponses Received?
Yes
90
Deassert PwrUpAck
End – CIF Power Down

Fig. 5



Start – Power Manager Power Up
100
Power Up
L2 Control
102
Assert
PwrUpReq
104
No
PwrUpAck
Assert?
Yes
End – Power Manager Power Up

Fig. 6

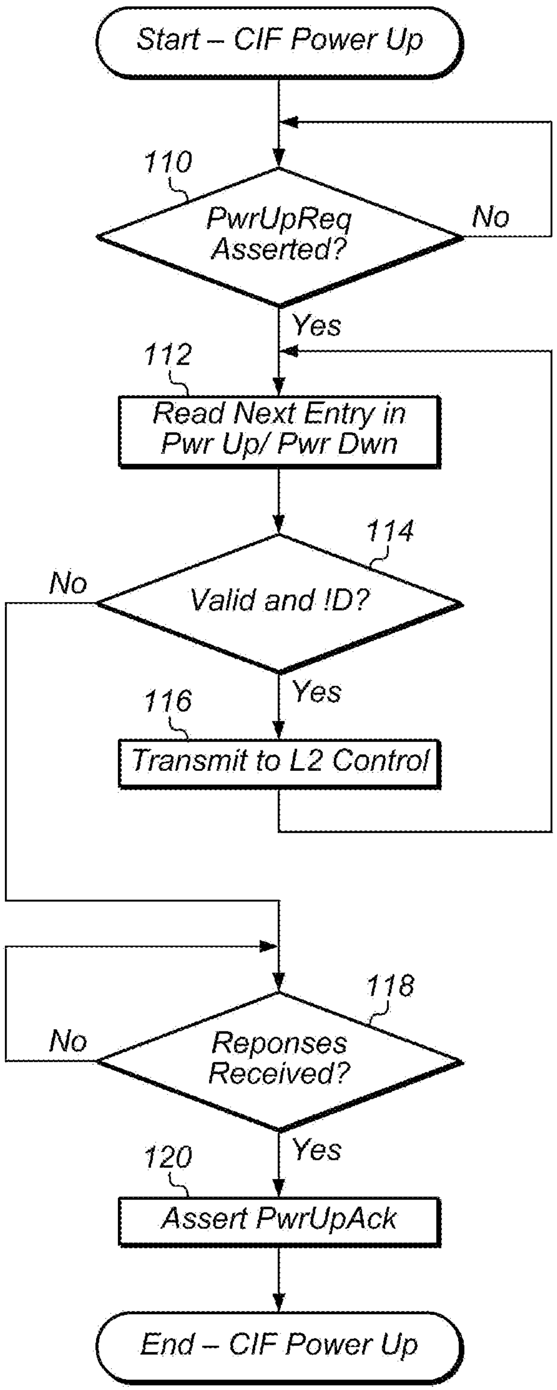

Start – CIF Power Up
110
PwrUpReq Asserted?
No
Yes
112
Read Next Entry in Pwr Up/ Pwr Dwn
114
No
Valid and !D?
Yes
116
Transmit to L2 Control
118
No
Reponses Received?
Yes
120
Assert PwrUpAck
End – CIF Power Up

Fig. 7

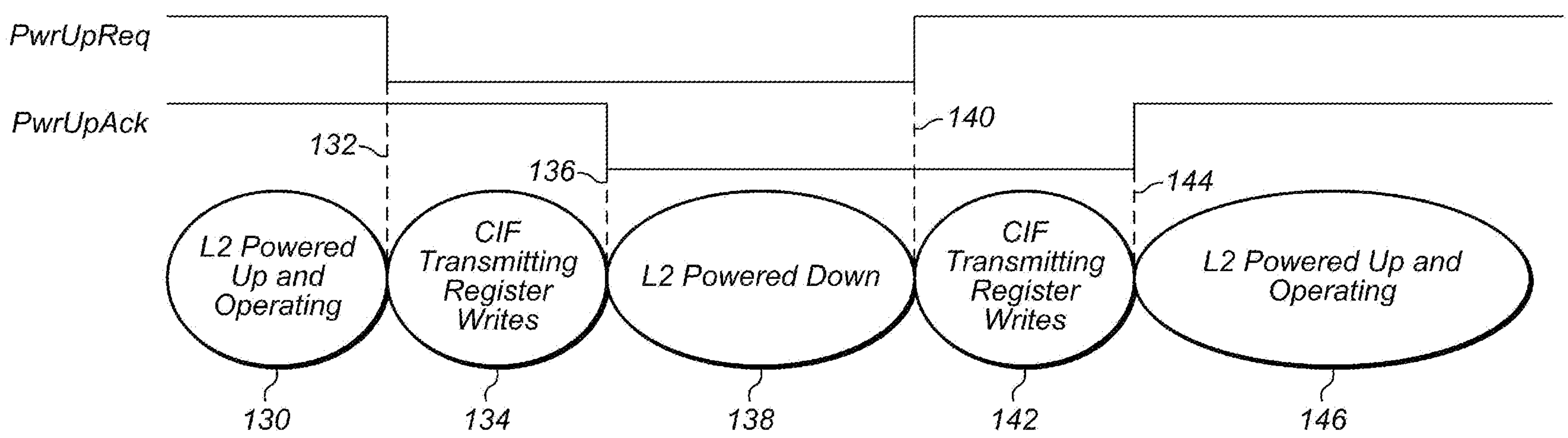

PwrUpReq
PwrUpAck
132
136
140
144
L2 Powered Up and Operating
CIF Transmitting Register Writes
L2 Powered Down
CIF Transmitting Register Writes
L2 Powered Up and Operating
130
134
138
142
146

Fig. 8

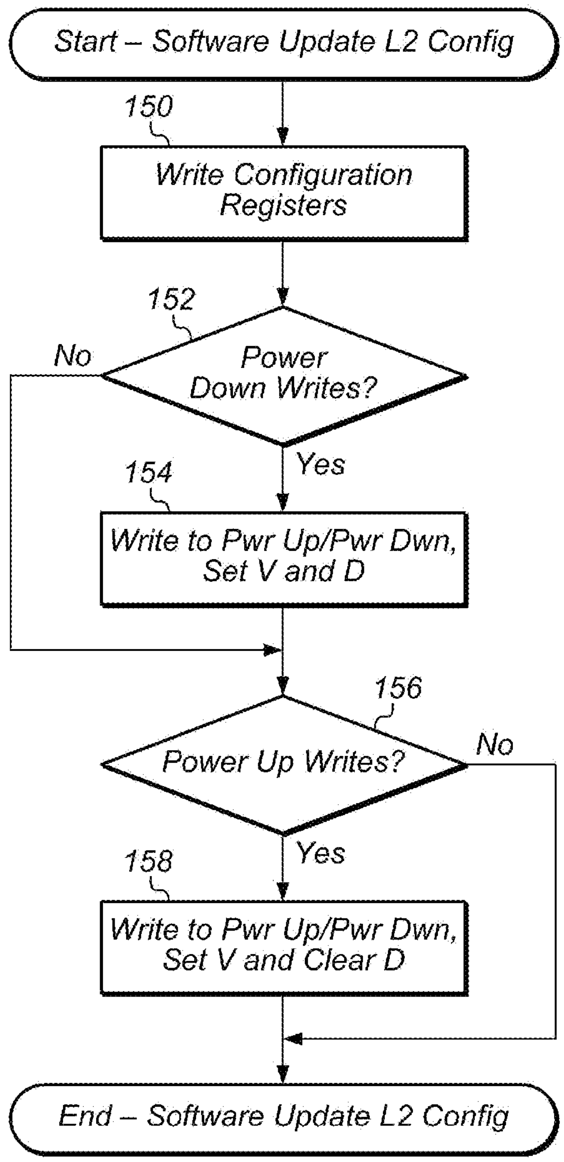

Start – Software Update L2 Config
150
Write Configuration Registers
152
Power Down Writes?
No
Yes
154
Write to Pwr Up/Pwr Dwn, Set V and D
156
Power Up Writes?
No
Yes
158
Write to Pwr Up/Pwr Dwn, Set V and Clear D
End – Software Update L2 Config

Fig. 9

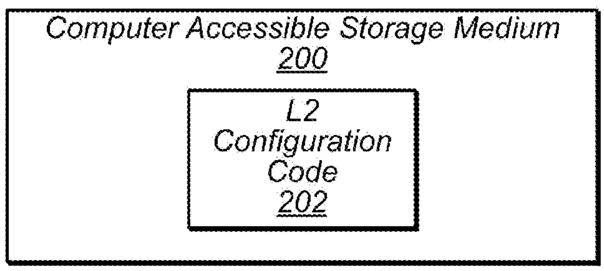
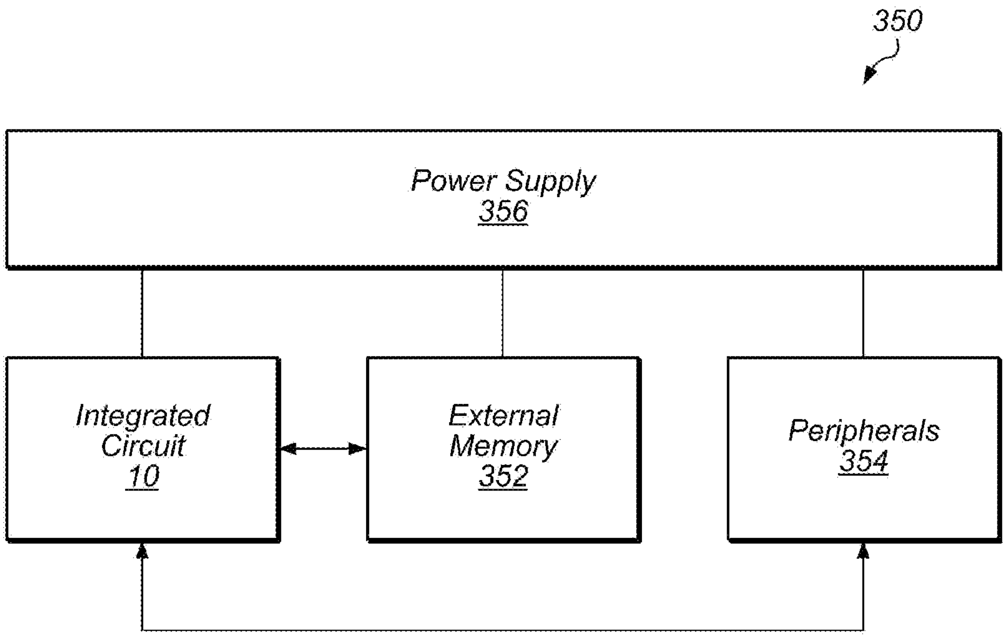

Computer Accessible Storage Medium
200
L2 Configuration Code
202

Fig. 10


350
Power Supply
356
Integrated
Circuit
10
External
Memory
352
Peripherals
354

Agentschap NL
Ministerie van Economische Zaken, Landbouw en Innovatie

OCTROOIAANVRAAG NR.:
NO 137888
NL 2007481

ONDERZOEKSRAPPORT

BETREFFENDE HET RESULTAAT VAN HET ONDERZOEK NAAR DE STAND VAN DE TECHNIEK

RELEVANTE LITERATUUR

Categorie[1]	Literatuur met, voor zover nodig, aanduiding van speciaal van belang zijnde tekstgedeelten of figuren.	Van belang voor conclusie(s) nr:	Classificatie (IPC)
X	US 2007/113057 A1 (KNOTH MATTHIAS [US]) 17 mei 2007 (2007-05-17) * samenvatting * * alinea [0018] * * alinea [0024] * * alinea [0036] - alinea [0037] *	1-14	INV. G06F1/32
X	EP 1 653 331 A2 (ST MICROELECTRONICS PVT LTD [IN] ST ERICSSON SA [CH]) 3 mei 2006 (2006-05-03)	1-5, 10-14	
A	* samenvatting * * alinea [0007] - alinea [0008] *	6-9	
X	WO 01/33322 A2 (INTEL CORP [US]; JAIN SATCHIT [US]; CHO SUNG SOO [US]) 10 mei 2001 (2001-05-10)	1-5, 10-14	
A	* samenvatting * * bladzijde 2, regel 4 - regel 21 * * bladzijde 8, regel 4 - bladzijde 9, regel 28 *	6-9	
	-/--		Onderzochte gebieden van de techniek: G06F

Indien gewijzigde conclusies zijn ingediend, heeft dit rapport betrekking op de conclusies ingediend op:

Plaats van onderzoek:	Datum waarop het onderzoek werd voltooid:	Bevoegd ambtenaar:
's-Gravenhage	5 september 2012	Archontopoulos, E

[1] CATEGORIE VAN DE VERMELDE LITERATUUR

X: de conclusie wordt als niet nieuw of niet inventief beschouwd ten opzichte van deze literatuur
Y: de conclusie wordt als niet inventief beschouwd ten opzichte van de combinatie van deze literatuur met andere geciteerde literatuur van dezelfde categorie, waarbij de combinatie voor de vakman voor de hand liggend wordt geacht
A: niet tot de categorie X of Y behorende literatuur die de stand van de techniek beschrijft
O: niet-schriftelijke stand van de techniek
P: tussen de voorrangsdatum en de indieningsdatum gepubliceerde literatuur
T: na de indieningsdatum of de voorrangsdatum gepubliceerde literatuur die niet bezwarend is voor de octrooiaanvrage, maar wordt vermeld ter verheldering van de theorie of het principe dat ten grondslag ligt aan de uitvinding
E: eerdere octrooi(aanvrage), gepubliceerd op of na de indieningsdatum, waarin dezelfde uitvinding wordt beschreven
D: in de octrooiaanvrage vermeld
L: om andere redenen vermelde literatuur
&: lid van dezelfde octrooifamilie of overeenkomstige octrooipublicatie

3

EOB FORM 02.83 (P0414B)

OCTROOIAANVRAAG NR.:
NO 137888
NL 2007481

RELEVANTE LITERATUUR		
Categorie[1]	Literatuur met, voor zover nodig, aanduiding van speciaal van belang zijnde tekstgedeelten of figuren.	Van belang voor conclusie(s) nr:
A	LEA HWANG LEE ET AL: "Instruction fetch energy reduction using loop caches for embedded applications with small tight loops", PROCEEDINGS 1999 INTERNATIONAL SYMPOSIUM ON LOW POWER ELECTRONICS AND DESIGN. (ISLPED). SAN DIEGO, CA, AUG. 16 - 17, 1999; [INTERNATIONAL SYMPOSIUM ON LOW POWER ELECTRONICS AND DESIGN], NEW YORK, NY : ACM, US, 16 augustus 1999 (1999-08-16), bladzijden 267-269, XP010355969, DOI: 10.1145/313817.313944 ISBN: 978-1-58113-133-8 * het gehele document * -----	1,6,10

3

EOB FORM 02.83 (P0414C)

[1] CATEGORIE VAN DE VERMELDE LITERATUUR

X: de conclusie wordt als niet nieuw of niet inventief beschouwd ten opzichte van deze literatuur
Y: de conclusie wordt als niet inventief beschouwd ten opzichte van de combinatie van deze literatuur met andere geciteerde literatuur van dezelfde categorie, waarbij de combinatie voor de vakman voor de hand liggend wordt geacht
A: niet tot de categorie X of Y behorende literatuur die de stand van de techniek beschrijft
O: niet-schriftelijke stand van de techniek
P: tussen de voorrangsdatum en de indieningsdatum gepubliceerde literatuur
T: na de indieningsdatum of de voorrangsdatum gepubliceerde literatuur die niet bezwarend is voor de octrooiaanvrage, maar wordt vermeld ter verheldering van de theorie of het principe dat ten grondslag ligt aan de uitvinding
E: eerdere octrooi(aanvrage), gepubliceerd op of na de indieningsdatum, waarin dezelfde uitvinding wordt beschreven
D: in de octrooiaanvrage vermeld
L: om andere redenen vermelde literatuur
&: lid van dezelfde octrooifamilie of overeenkomstige octrooipublicatie

AANHANGSEL BEHORENDE BIJ HET RAPPORT BETREFFENDE HET ONDERZOEK NAAR DE STAND VAN DE TECHNIEK, UITGEVOERD IN DE OCTROOIAANVRAGE NR.

NO 137888
NL 2007481

Het aanhangsel bevat een opgave van elders gepubliceerde octrooiaanvragen of octrooien (zogenaamde leden van dezelfde octrooifamilie), die overeenkomen met octrooischriften genoemd in het rapport.
De opgave is samengesteld aan de hand van gegevens uit het computerbestand van het Europees Octrooibureau per
De juistheid en volledigheid van deze opgave wordt noch door het Europees Octrooibureau, noch door het Bureau voor de Industriële eigendom gegarandeerd; de gegevens worden verstrekt voor informatiedoeleinden.

05-09-2012

In het rapport genoemd octrooigeschrift	Datum van publicatie	Overeenkomend(e) geschrift(en)	Datum van publicatie
US 2007113057 A1	17-05-2007	CN 101375244 A	25-02-2009
		US 2007113057 A1	17-05-2007
		WO 2007059214 A1	24-05-2007
EP 1653331 A2	03-05-2006	EP 1653331 A2	03-05-2006
		US 2006253716 A1	09-11-2006
WO 0133322 A2	10-05-2001	AT 510250 T	15-06-2011
		AU 4609401 A	14-05-2001
		CN 1415085 A	30-04-2003
		EP 1228413 A2	07-08-2002
		HK 1052569 A1	23-04-2010
		JP 3701910 B2	05-10-2005
		JP 2003519830 A	24-06-2003
		SG 122817 A1	29-06-2006
		TW 498194 B	11-08-2002
		US 6571333 B1	27-05-2003
		US 2003172313 A1	11-09-2003
		US 2004225907 A1	11-11-2004
		WO 0133322 A2	10-05-2001

Algemene informatie over dit aanhangsel is gepubliceerd in de 'Official Journal' van het Europees Octrooibureau nr 12/82 blz 448 ev

Agentschap NL
Ministerie van Economische Zaken, Landbouw en Innovatie

SCHRIFTELIJKE OPINIE

DOSSIER NUMMER	INDIENINGSDATUM	VOORRANGSDATUM	AANVRAAGNUMMER
NO137888	27.09.2011	30.09.2010	NL2007481

CLASSIFICATIE
INV. G06F1/32

AANVRAGER
Apple Inc.

Deze schriftelijke opinie bevat een toelichting op de volgende onderdelen:

- ☒ Onderdeel I Basis van de schriftelijke opinie
- ☐ Onderdeel II Voorrang
- ☐ Onderdeel III Vaststelling nieuwheid, inventiviteit en industriële toepasbaarheid niet mogelijk
- ☐ Onderdeel IV De aanvraag heeft betrekking op meer dan één uitvinding
- ☒ Onderdeel V Gemotiveerde verklaring ten aanzien van nieuwheid, inventiviteit en industriële toepasbaarheid
- ☐ Onderdeel VI Andere geciteerde documenten
- ☐ Onderdeel VII Overige gebreken
- ☒ Onderdeel VIII Overige opmerkingen

	DE BEVOEGDE AMBTENAAR
	Archontopoulos, E

Form NL237A (Dekblad) (July 2006)

Aanvraag nr.:

SCHRIFTELIJKE OPINIE

NL2007481

Onderdeel I Basis van de Schriftelijke Opinie

1. Deze schriftelijke opinie is opgesteld op basis van de meest recente conclusies ingediend voor aanvang van het onderzoek.

2. Met betrekking tot **nucleotide en/of aminozuur sequenties** die genoemd worden in de aanvraag en relevant zijn voor de uitvinding zoals beschreven in de conclusies, is dit onderzoek gedaan op basis van:

   a. type materiaal:

   ☐ sequentie opsomming

   ☐ tabel met betrekking tot de sequentie lijst

   b. vorm van het materiaal:

   ☐ op papier

   ☐ in elektronische vorm

   c. moment van indiening/aanlevering:

   ☐ opgenomen in de aanvraag zoals ingediend

   ☐ samen met de aanvraag elektronisch ingediend

   ☐ later aangeleverd voor het onderzoek

3. ☐ In geval er meer dan één versie of kopie van een sequentie opsomming of tabel met betrekking op een sequentie is ingediend of aangeleverd, zijn de benodigde verklaringen ingediend dat de informatie in de latere of additionele kopieën identiek is aan de aanvraag zoals ingediend of niet meer informatie bevatten dan de aanvraag zoals oorspronkelijk werd ingediend.

4. Overige opmerkingen:

SCHRIFTELIJKE OPINIE

Aanvraag nr.:
NL2007481

Onderdeel V Gemotiveerde verklaring ten aanzien van nieuwheid, inventiviteit en industriële toepasbaarheid

1. Verklaring

Nieuwheid	Ja:	Conclusies 1-14
	Nee:	Conclusies
Inventiviteit	Ja:	Conclusies
	Nee:	Conclusies 1-14
Industriële toepasbaarheid	Ja:	Conclusies 1-14
	Nee:	Conclusies

2. Citaties en toelichting:

Zie aparte bladzijde

Onderdeel VIII Overige opmerkingen

De volgende opmerkingen met betrekking tot de duidelijkheid van de conclusies, beschrijving, en figuren, of met betrekking tot de vraag of de conclusies nawerkbaar zijn, worden gemaakt:

Zie aparte bladzijde

Re Item V

Reasoned statement with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1 Reference is made to the following document.

D1: US 2007/113057 A1, 17 May 2007.

2 The present application does not meet the criteria of patentability, because the subject-matter of all claims does not involve an inventive step.

3 D1 is regarded as being the prior art closest to the subject-matter of claim 1 and discloses (paragraphs [0024], [0036]-[0037]):

- A memory configured to store a first set of operations to be performed prior to powering down a circuit block: instruction fetch unit 120 includes a fetch controller 200 and a multiplexer 208 which selects between an instruction provided on bus 202 by instruction cache 102 and an instruction provided on bus 204 by loop buffer 104 and provides the selected instruction to execution unit 106 on bus 206. The selected instruction is also provided to loop buffer 104 on bus 210 so that the selected instruction may be stored in loop buffer 104.
- A memory configured to store a second set of operations to be performed subsequent to powering up the circuit block: By disabling the instruction cache only after a designated number of branch not taken instructions are stored in the instruction cache, instructions may be provided to the execution unit more quickly when the loop exits at the branch instruction.
- A control circuit configured to perform the corresponding first set of operations when before the actual power down of the circuit block: In step 306, once a loop is stored in the loop buffer, the instruction cache is disabled to reduce power consumption. When a branch instruction forms the loop stored in the loop buffer, the instruction cache is disabled only after a designated number of instructions following the branch not taken path is made available in the instruction cache.

4 The subject-matter of claim 1 only differs from what is disclosed in D1 in that the instructions performed before the powering down are not directly executed in response to the request to power down the circuit block.

5 This is however a mere straightforward choice and, moreover, it is noted that the net technical effect of D1 and that of claim 1 is the same in this respect.

6 The subject-matter of claim 1 is therefore not inventive, as not inventive is that of the corresponding dual claim 10.

7 D1 already discloses that the block powered down is a cache; D1 thus anticipates the feature of claim 6 of the circuit block being, indeed, a cache.

8 All other features defined in claim 6 constituting obvious design and implementation choices, the subject-matter of claim 6 is not inventive, either.

9 Dependent claims 2-5, 7-9 and 11-14 merely define straightforward implementation choices void of any non-obvious contribution that the skilled person in the filed of low-level power saving would effortlessly consider and apply to the system disclosed in D1 and so arrive at the teaching of said claims without resorting to any inventive activity.

10 The subject-matter of claims 2-5, 7-9 and 11-14 is in consequence not inventive.

Re Item VIII

Certain observations on the application

11 The vague and imprecise statements in paragraph 17 of the description implies that the subject-matter for which protection is sought may be different to that defined by the claims, thereby resulting in lack of clarity when used to interpret them.

12 Analogously, reference to 35 U.S.C. § 112 in paragraph 18 should be removed.