



(51) International Patent Classification:
G06F 12/0897 (2016.01)

(21) International Application Number:
PCT/US2018/014666

(22) International Filing Date:
22 January 2018 (22.01.2018)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
15/437,843 21 February 2017 (21.02.2017) US

(71) Applicant: **ADVANCED MICRO DEVICES, INC.**
[US/US]; 2485 Augustine Drive, Santa Clara, California
95054 (US).

(72) Inventor: **CONWAY, Patrick N.**; AMD - US Santa Clara,
2485 Augustine Drive, Santa Clara, California 95054 (US).

(74) Agent: **CAVE, Nicole Teitler** et al.; Zagorin Cave LLP,
4101 Parkstone Heights Drive, Suite 350, Austin, TX 78746
(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, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, 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, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, 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,

(54) Title: ACCELERATION OF CACHE-TO-CACHE DATA TRANSFERS FOR PRODUCER-CONSUMER COMMUNICATION

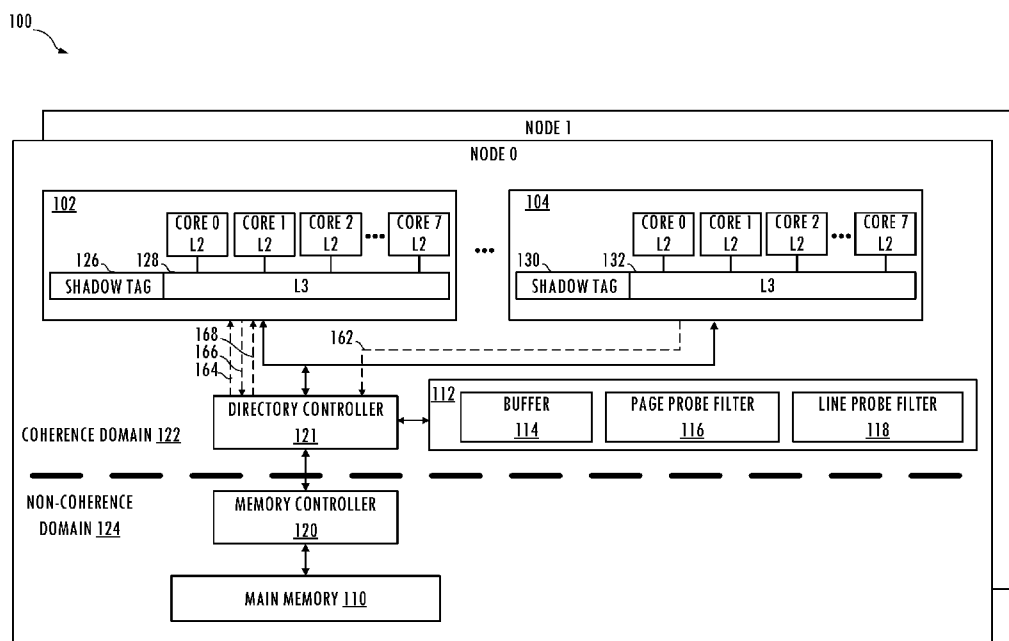


FIG. 2

(57) Abstract: A communication bypass mechanism accelerates cache-to-cache data transfers for communication traffic between caching agents that have separate last-level caches. A method includes bypassing a last-level cache of a first caching agent in response to a cache line having a modified state being evicted from a penultimate-level cache of the first caching agent and a communication attribute of a shadow tag entry associated with the cache line being set. The communication attribute indicates prior communication of the cache line with a second caching agent having a second last-level cache.

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,
KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

- *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*
- *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

Published:

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

- 1 -

ACCELERATION OF CACHE-TO-CACHE DATA TRANSFERS FOR PRODUCER-CONSUMER COMMUNICATION

Patrick N. Conway

BACKGROUND ART

5 In a typical computing system, a memory system is designed with a goal of low latency experienced by a processor when accessing arbitrary units of data. In general, the memory system design takes advantage of memory access properties known as temporal locality and spatial locality. Temporal locality refers to multiple accesses to specific memory locations within a relatively small time period. Spatial
10 locality refers to accesses to memory locations relatively close in the address space within a relatively small time period.

 Typically, temporal locality is evaluated in terms of a granularity smaller than that of a next level in a memory hierarchy. For example, a cache captures a repeated access of blocks of fixed size (i.e., cache lines, e.g., blocks of 64 Bytes (B)), which
15 are smaller than the storage granularity of main memory (e.g., 4 Kilobyte (KB) pages). A cache captures spatial locality by storing locally quantities of sequentially stored data slightly larger than a requested quantity in order to reduce memory access latency in the event of sequential access. For example, a cache is designed to store 64
20 B blocks, although a processor requests one to eight Bytes at a time. Meanwhile, the cache requests blocks of 64 B at a time from a memory, which stores pages of 4 KB contiguous portions.

 In a shared memory multiprocessor system, workloads may cause communications between caching agents (e.g., processors, graphics processing units, processor offload engines, or other processing units that each include a cache in a
25 node, socket, or other multi-processor system). Communications between caching agents that do not share a last-level cache may result in cache-to-cache latency for communication traffic between caching agents. In general, as cache sizes grow larger over time (e.g., using die stacked SRAM), the proportion of communication misses will increase and increase the effects of cache-to-cache latency for communication

- 2 -

traffic between caching agents. Thus, improved techniques for communication between caching agents is desired.

DISCLOSURE OF THE INVENTION

5 In at least one embodiment of the invention, a method includes storing a communication attribute in a shadow tag entry associated with a cache line stored in a penultimate-level cache of a first caching agent having a first last-level cache. The method includes bypassing the first last-level cache in response to the cache line having a modified state, the cache line being evicted from the penultimate-level cache, and the communication attribute having a first state. The first state of the
10 communication attribute indicates prior communication of the cache line with a second caching agent having a second last-level cache. The method may include storing the cache line in the first last-level cache in response to the cache line having the modified state, the cache line being evicted from the penultimate-level cache, and the communication attribute having a second state.

15 The bypassing may include issuing a victim packet including a communication bypass attribute and the cache line to a directory controller and writing the cache line to a buffer by the directory controller. The bypassing may further include issuing a storing probe packet by the directory controller to the second caching agent in response to receiving the victim packet. The second caching agent
20 may be identified as a previous owner of the cache line by communication history information for the cache line. The bypassing may further include prefetching the cache line into a second last-level cache of the second caching agent, setting to the first state an associated communication attribute in a second shadow tag entry of the second caching agent, and updating the communication history information for the
25 cache line in response to receiving the storing probe packet by the second caching agent. The method may include setting to the first state the communication attribute in response to satisfying a memory request miss of the first last-level cache by a read response from the second caching agent. The method may include storing communication history information for the cache line in a probe filter.

- 3 -

In at least one embodiment of the invention, an apparatus includes a probe filter configured to store communication history information for a cache line stored in a first caching agent having a first last-level cache. The apparatus includes a controller configured to store the cache line in response to the cache line being evicted from a penultimate-level cache of the first caching agent and configured to provide the cache line to a second caching agent having a second last-level cache in response to the communication history information. The communication history information may be set in response to the cache line being provided by the first caching agent to a second caching agent in response to a directed probe. The communication history information may include a previous owner identifier and a communication state.

The first caching agent may include a shadow tag memory associated with the first last-level cache. The shadow tag memory may be configured to store a communication attribute for the cache line. The communication attribute may have a first state in response to a read response from the second caching agent initiated by a miss in the first last-level cache. The first caching agent may be configured to issue a victim packet including a communication bypass attribute to the controller in response to the cache line having a modified state being evicted from a penultimate-level cache of the first caching agent and in response to the communication attribute having the first state. The controller may be configured to issue a storing probe packet to the second caching agent in response to receiving a victim packet from the first caching agent. The probe filter may identify the second caching agent as a previous owner of the cache line in the communication history information. The second caching agent may be configured to prefetch the cache line into the second last-level cache and to set to the first state an associated communication attribute in a second shadow tag of the second caching agent.

In at least one embodiment of the invention, an apparatus includes a shadow tag memory configured to store a communication attribute associated with a cache line stored in a penultimate-level cache of a first caching agent having a first last-level cache. The apparatus includes control logic configured to bypass the first last-level cache in response to the cache line having a modified state, the cache line being evicted from the penultimate-level cache, and the communication attribute

- 4 -

having a first state. The first state indicates prior communication of the cache line with a second caching agent having a second last-level cache. The control logic may be further configured to store the cache line in the first last-level cache in response to the cache line having the modified state, the cache line being evicted from the

5 penultimate-level cache, and the communication attribute having a second state. The apparatus may include a probe filter configured to store status information and communication history information for the cache line. The apparatus may include a directory controller configured to store the cache line to main memory in response to the cache line being evicted from the penultimate-level cache and configured to

10 provide the cache line to the second caching agent responsive to the communication history information.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention may be better understood, and its numerous objects, features, and advantages made apparent to those skilled in the art by referencing the

15 accompanying drawings.

Figure 1 illustrates a functional block diagram of an exemplary processing system and exemplary consumer-producer communications between caching agents consistent with at least one embodiment of the invention.

Figure 2 illustrates a functional block diagram of the exemplary

20 processing system of Fig. 1 and exemplary consumer-producer communications between caching agents consistent with at least one embodiment of the invention.

Figure 3A illustrates conventional cache-to-cache transfers between a producer caching agent and a consumer caching agent.

Figure 3B illustrates exemplary cache-to-cache transfers between a

25 producer caching agent and a consumer caching agent using the communication bypass mechanism of the exemplary processing system of Fig. 1 consistent with at least one embodiment of the invention.

- 5 -

Figure 4 illustrates exemplary data fields of an entry in the line probe filter of the exemplary processing system of Fig. 1 consistent with at least one embodiment of the invention.

The use of the same reference symbols in different drawings indicates
5 similar or identical items.

MODE(S) FOR CARRYING OUT THE INVENTION

A technique for accelerating cache-to-cache transfers between caching agents attempts to prefetch a cache line by a consumer caching agent. Since the consumer caching agent generally does not know when a cache line has been made
10 available by a producer caching agent, the prefetching technique alone is as likely to degrade performance as it is to improve performance depending on the timing of the prefetch by the consumer caching agent.

Another technique for accelerating cache-to-cache transfers introduces a new attribute associated with each cache line stored in a caching agent. The attribute
15 indicates the provenance of an associated cache line. When a cache fill request is serviced by another caching agent and the cache line is dirty, then the cache line is categorized as a communication cache line. When that cache line is eventually cast out of a private cache of the caching agent, the cache line is installed in the last-level cache only if the cache line is not a communication cache line. Otherwise, the cache
20 line is written back to main memory. When the consumer caching agent requests that cache line, it is returned from main memory instead of the cache in the producer caching agent. This approach provides a modest benefit if the latency of accessing the data from main memory is less than a cache-to-cache transfer. However, under some circumstances this technique wastes power unnecessarily by writing a cache line
25 that is involved in producer-consumer communication to main memory and unnecessarily reading that cache line back from main memory.

A communication bypass mechanism accelerates cache-to-cache data transfers for cache lines communicated between caching agents that have private last-level caches. The acceleration provided by the communication bypass mechanism

- 6 -

increases with increases to the size of the last-level cache. In some applications, since a next access to a cache line is likely to be a read from a consumer caching agent, the communication bypass mechanism uses an eviction of the cache line from the penultimate-level cache (e.g., a level-two cache of a three level cache) of a producer
5 caching agent to trigger a transfer of the cache line to the last-level cache (e.g., a level-three cache of the three level cache) of the consumer caching agent prior to a request for the cache line by the consumer caching agent.

The communication bypass mechanism uses a shadow tag, which is a cache-like hardware structure that is associated with a last-level cache. The shadow
10 tag records any cache lines stored in a penultimate-level cache or higher-level cache of the cores. The shadow tag maintains communication information indicating whether the associated cache line is involved in communication between caching agents. The last-level cache sets the communication information when it receives or provides the associated cache line in response to a directed probe. The caching agent
15 subsequently bypasses a last-level cache install for a communication line when that cache line is evicted from a penultimate-level cache. That victim cache line is associated with state information and a destination that a directory controller uses to inject the data into the last-level cache of a consumer caching agent before the consumer caching agent requests the data. The mechanism allows the consumer
20 caching agent to directly access the data of the victim cache line from its cache and avoid the latency of a cache-to-cache transfer from the last-level cache of the producer caching agent. The directory controller uses the communication history information stored in a probe filter to steer cache lines evicted from a cache of a producer caching agent to a cache of a consumer caching agent.

25 Referring to Fig. 1, processing system 100 (e.g., a server) includes multiple processing nodes (e.g., node 0 and node 1). Each processing node includes multiple caching agents (e.g., processors 102 and 104 coupled to main memory 110). For example, caching agent 102, is a processor including core 0, core 1, core 2, ... core 7 and caching agent 104, is a processor including core 0, core 1, core 2, ... core
30 7) and a memory system. Each of the nodes accesses its own memory within corresponding coherence domain 122 faster than memory in non-coherence domain

- 7 -

124 (e.g., main memory 110) or memory in another node. As referred to herein, a coherence domain refers to a subset of memory (e.g., cache memory of node 0) for which a cache coherence mechanism maintains a coherent view of copies of shared data. A non-coherence domain refers to memory not included in the coherence
5 domain (e.g., main memory 110 or cache memory in another node). Each of the caching agents in a node includes a last-level cache shared by the cores of the caching agent. Each core includes a private penultimate-level cache. For example, caching agent 102 includes last-level cache 128, which is a level-three cache shared by core 0, core 1, core 2, ... core 7, and includes a level-two cache within each of core 0, core 1,
10 core 2, ... core 7. Last-level cache 128 and each level-two cache of caching agent 102 includes storage elements, e.g., storage implemented in fast static Random Access Memory (RAM) or other suitable storage elements. Cache control logic is distributed across last-level cache 128 and each level-two cache of caching agent 102. The caching agents use inter-processor communication via directory controller 121 to
15 maintain coherency of a memory image in main memory 110 when caches of more than one caching agent contain the same cache line (i.e., a copy of contents of the same location of main memory 110) of coherence domain 122. In at least one embodiment of processing system 100, probe filter 112 includes storage for a cache directory used to implement a directory-based cache coherency policy. Probe filter
20 112 is implemented in fast static RAM associated with directory controller 121 or by other suitable storage technique.

Probe filter 112 includes line probe filter 118 and buffer 114. Buffer 114 is used and reused as temporary storage for communications between caching agent 102, caching agent 104, and main memory 110. In at least one embodiment, probe
25 filter 112 includes page probe filter 116, which tracks pages stored in the caches (e.g., 4 KB pages) of coherence domain 122 and line probe filter 118 tracks the caching status for any cache lines shared across caching agents (e.g., written to by a core) in coherence domain 122 and any associated communication history.

Figure 4 illustrates an exemplary line probe filter entry. A portion of the
30 line probe filter entry 300 includes conventional caching status information (e.g., tag, state, owner, etc.) and additional communication history information. The state

- 8 -

information is consistent with a conventional cache coherency protocol (e.g., MOESI protocol) for instruction fetches, data loads, and data store operations. Line probe filter entry 300 optionally includes presence information indicating which caching agents have accessed a line entry in line probe filter such as cluster valid information (CLV), local socket valid information (LSV), remote socket valid information (RSV). In addition, line probe filter entry 300 includes communication history bits for storing previous state information including an indication of a previous owner of the cache line associated with the tag and a communication state for the cache line associated with the tag. The communication state indicates one of the line being invalid, owned by a producer, owned by a consumer, and indication of communication between caching agents. The communication history information not required or used to maintain cache coherence by the conventional cache coherency protocol.

Referring to Figs. 1 and 4, a memory request of caching agent 104 that results in a cache miss (i.e., the cache line does not reside in any level of the cache in caching agent 104) causes caching agent 104 to issue probe 152 to directory controller 121 requesting the cache line from main memory. Directory controller 121 accesses line probe filter 118 and determines whether the target cache line resides in a cache of another caching agent. In response to an indication that the target cache line resides in a cache of caching agent 102, directory controller 121 issues direct probe 154 to caching agent 102 and receives the cache line from caching agent 102 in response 156. Directory controller 121 updates the communication history information in line probe filter entry 300 for the cache line in line probe filter 118 and sends the associated data to caching agent 104 in response 158. For example, directory controller 121 updates a communication state stored in field CSTATE and an identifier of the previous owner stored in field PREV_OWNER of line probe filter entry 300. In addition, caching agent 104 sets a communication attribute in shadow tag 130 indicating that this cache line was already resident in a cache of another caching agent within coherence domain 122 and was communicated to caching agent 104 by that other caching agent instead of being fetched from memory outside coherence domain 122 (e.g., by memory controller 120 from main memory 110).

- 9 -

Referring to Figs. 2 and 4, when caching agent 104 evicts a cache line from the penultimate-level cache (e.g., level-two cache of core 0 in caching agent 104), if the victim cache line has a modified or dirty state, caching agent 104 determines a status of an associated communication attribute in shadow tag 130. If
5 the associated communication attribute in shadow tag 130 is set to a state indicating prior communication of the cache line with another caching agent having a last-level cache, then cache control logic in caching agent 104 bypasses the last-level cache (e.g., level-three cache 132) and writes the victim cache line back to main memory
10 110 rather than installing the cache line in last-level cache 132. For example, caching agent 104 sends victim packet 162 to directory controller 121, which stores the victim cache line in buffer 114 temporarily as part of a process of writing the modified data back to main memory 110 using memory controller 120.

In at least one embodiment, victim packet 162 carries a communication bypass attribute that causes directory controller 121 to send storing probe 164 to the
15 previous owner indicated in the communication history field of associated line probe filter entry 300 in line probe filter 118. Storing probe 164 serves as a hint to caching agent 102 that the cache line is being written back to main memory 110 and then cache-to-cache data transfers may be accelerated. If the previous owner is caching agent 102, in response to storing probe 164, caching agent 102 sends prefetch 166. In
20 response to prefetch 166, directory controller 121 sends response 168, which causes cache control logic to install the cache line into the last-level cache of caching agent 102 (e.g., level-three cache 128) and sets an associated communication attribute for the cache line in shadow tag 126. Since at least some of the cache line resides in buffer 114, the installation of the cache line in caching agent 102 is further
25 accelerated by the prefetch, as compared to a later fetch of the cache line from main memory 110. If the communication attribute associated with the victim cache line has a value indicating that the cache line was not communicated to caching agent 104 by another caching agent, then caching agent 104 installs the data associated with that cache line in the last-level cache, as is typical of eviction from the penultimate-level
30 cache. Note that the probe messages and responses of Figs. 1 and 2 are exemplary only and other probe messages and responses between caching agent 102 or caching agent 104, and directory controller 121 may be used.

- 10 -

Referring to Figs. 2 and 3A, since the communication bypass mechanism targets a data communication buffer, and not a synchronization flag, the communication bypass mechanism takes advantage of communication patterns and reuse of buffer 114, which is shared by caching agents 102 and 104. The communication bypass mechanism accelerates cache-to-cache data transfers by preloading a cache line into a destination last-level cache of a consumer caching agent and/or writing a victim cache line to main memory before the consumer agent uses the victim data. In a conventional system of Fig. 3A, the communication bypass mechanism is not implemented. In response to a producer caching agent evicting cache line 224 (e.g., N lines cache lines), the memory controller writes those victim cache lines to main memory. Upon completion of the write to main memory, the producer caching agent writes flag 220 indicating that the data communication buffer contains valid data. Meanwhile, in response to a consumer caching agent requesting the cache line, the consumer caching agent polls buffer 114 until flag 220 is visible. In response to detecting flag 220, the consumer caching agent reads the cache line from main memory.

A system implementing the communication bypass mechanism of Figs. 1, 2, and 4, installs at least portion 226 of cache line 224 into the last-level cache of the consumer caching agent before the producer caching agent finishes writing the cache line to main memory and writing flag 220, as illustrated in Fig. 3B. For example, in response to a prefetch initiated by eviction of a cache line from a penultimate-level cache, portion 226 of cache line 224 is deposited in the last-level cache of the consumer caching agent, from buffer 114, and prior to directory controller 121 writing flag 220. When the consumer caching agent prefetches that victim cache line, the consumer caching agent reads portion 226 from buffer 114 to level-three cache 128 and polls flag 220. In response to flag 220 being set, the consumer caching agent obtains remaining portion 228 from main memory using conventional memory transactions. A subsequent access to the prefetched cache line is serviced from the cache of the consumer caching agent instead of from another caching agent or main memory. Thus, the communication bypass mechanism reduces latency of cache-to-cache communications between caching agents.

- 11 -

Thus, a communication bypass mechanism that reduces the latency of cache-to-cache transfers for producer-consumer communication between caching agents in a shared memory multiprocessor system has been described. The reduced latency increases throughput and reduces response times for workloads that have a high level of producer-consumer communication (e.g., inter-task communications). Bus trace analysis for benchmark simulations indicates that several exemplary workloads involve only a small number of cache lines in communication and heavily access those cache lines. Thus, buffer 114 need not be large to capture the set of cache lines involved in producer-consumer communication. Performance for cache-to-cache transfers of a probe filter-based (i.e., cache directory-based) system approaches the performance for cache-to-cache transfers in systems including a large center last-level cache that is shared by all cores in a node.

While circuits and physical structures have been generally presumed in describing embodiments of the invention, it is well recognized that in modern semiconductor design and fabrication, physical structures and circuits may be embodied in computer-readable descriptive form suitable for use in subsequent design, simulation, test or fabrication stages. Structures and functionality presented as discrete components in the exemplary configurations may be implemented as a combined structure or component. Various embodiments of the invention are contemplated to include circuits, systems of circuits, related methods, and tangible computer-readable medium having encodings thereon (e.g., VHSIC Hardware Description Language (VHDL), Verilog, GDSII data, Electronic Design Interchange Format (EDIF), and/or Gerber file) of such circuits, systems, and methods, all as described herein, and as defined in the appended claims. In addition, the computer-readable media may store instructions as well as data that can be used to implement the invention. The instructions/data may be related to hardware, software, firmware or combinations thereof.

The description of the invention set forth herein is illustrative, and is not intended to limit the scope of the invention as set forth in the following claims. For example, while the invention has been described in an embodiment in which a MOESI cache coherence protocol is used and three levels of cache are used, one of

- 12 -

skill in the art will appreciate that the teachings herein can be utilized with other cache coherence protocols and caches having other numbers of levels. In addition, while the invention has been described in embodiments in which the caching agents are multi-core processors, one of skill in the art will appreciate that the teachings

5 herein can be utilized to accelerate producer-consumer communication between any pair of caching agents (e.g., processor core-to-processor core, processor core-to-GPU, processor core-to-offload engine, etc.). Variations and modifications of the embodiments disclosed herein, may be made based on the description set forth herein, without departing from the scope of the invention as set forth in the following claims.

- 13 -

WHAT IS CLAIMED IS:

1. A method comprising:
storing a communication attribute in a shadow tag entry associated with a
cache line stored in a penultimate-level cache of a first caching agent
having a first last-level cache; and
5 bypassing the first last-level cache in response to the cache line having a
modified state, the cache line being evicted from the penultimate-level
cache, and the communication attribute having a first state,
wherein the first state indicates prior communication of the cache line with a
second caching agent having a second last-level cache.
- 10 2. The method, as recited in claim 1, further comprising:
storing the cache line in the first last-level cache in response to the cache line
having the modified state, the cache line being evicted from the
penultimate-level cache, and the communication attribute having a
second state.
- 15 3. The method, as recited in claim 1 or 2, wherein the bypassing comprises:
issuing a victim packet including a communication bypass attribute and the
cache line to a directory controller; and
writing the cache line to a buffer by the directory controller.
- 20 4. The method, as recited in claim 3, wherein the bypassing further comprises:
issuing a storing probe packet by the directory controller to the second caching
agent in response to receiving the victim packet, the second caching
agent being identified as a previous owner of the cache line by
communication history information for the cache line.

- 14 -

5. The method, as recited in claim 4, wherein the bypassing further comprises:
prefetching the cache line into the second last-level cache, setting to the first
state an associated communication attribute in a second shadow tag
entry of the second caching agent, and updating the communication
5 history information for the cache line in response to receiving the
storing probe packet by the second caching agent.

6. The method, as recited in claim 1 or 2, further comprising:
setting to the first state the communication attribute in response to satisfying a
memory request miss of the first last-level cache by a read response
10 from the second caching agent.

7. The method, as recited in claim 1 or 2, further comprising:
storing communication history information for the cache line in a probe filter.

8. The method, as recited in claim 7, wherein the communication history
information includes a previous owner identifier and a communication state.

9. The method, as recited in claim 8, wherein the communication state
15 indicates one of the cache line being invalid, the cache line being owned by a
producer, the cache line being owned by a consumer, and the cache line being
communicated between the first caching agent and the second caching agent in
response to a directed probe.

10. An apparatus comprising:
20 a probe filter configured to store communication history information for a
cache line stored in a first caching agent having a first last-level cache;
and
a controller configured to store the cache line in response to the cache line
25 being evicted from a penultimate-level cache of the first caching agent
and configured to provide the cache line to a second caching agent
having a second last-level cache in response to the communication
history information.

- 15 -

11. The apparatus, as recited in claim 10, wherein the communication history information is set in response to the cache line being provided by the first caching agent to a second caching agent in response to a directed probe.

12. The apparatus, as recited in claim 10, wherein the communication history
5 information includes a previous owner identifier and a communication state.

13. The apparatus, as recited in claim 12, wherein the communication state indicates one of the cache line being invalid, the cache line being previously owned by a producer, the cache line being previously owned by a consumer, and the cache line being communicated between the first caching agent and the second caching
10 agent in response to a directed probe.

14. The apparatus, as recited in claim 10, 11, or 12 further comprising:
the first caching agent comprising:

a shadow tag memory associated with the first last-level cache, the
shadow tag memory being configured to store a communication
15 attribute for the cache line, the communication attribute having
a first state in response to a read response from the second
caching agent initiated by a miss in the first last-level cache,
wherein the first state indicates prior communication of the cache line
with a second caching agent having a second last-level cache.

20 15. The apparatus, as recited in claim 14,
wherein the first caching agent is configured to issue a victim packet including
a communication bypass attribute to the controller in response to the
cache line having a modified state and in response to the
communication attribute having the first state.

25 16. The apparatus, as recited in claim 14, wherein the controller is configured
to issue a storing probe packet to the second caching agent in response to receiving a
victim packet from the first caching agent, the probe filter identifying the second

- 16 -

caching agent as a previous owner of the cache line in the communication history information.

17. The apparatus, as recited in claim 14, wherein the second caching agent is configured to prefetch the cache line into the second last-level cache and to set to the
5 first state an associated communication attribute in a second shadow tag of the second caching agent.

18. An apparatus comprising:
a shadow tag memory configured to store a communication attribute
associated with a cache line stored in a penultimate-level cache of a
10 first caching agent having a first last-level cache; and
control logic configured to bypass the first last-level cache in response to the
cache line having a modified state, the cache line being evicted from
the penultimate-level cache, and the communication attribute having a
first state,
15 wherein the first state indicates prior communication of the cache line with a
second caching agent having a second last-level cache.

19. The apparatus, as recited in claim 18, wherein the control logic is further
configured to store the cache line in the first last-level cache in response to the cache
line having the modified state, the cache line being evicted from the penultimate-level
20 cache, and the communication attribute having a second state.

20. The apparatus, as recited in claim 18 or 19, further comprising:
a probe filter configured to store status information and communication
history information for the cache line; and
a directory controller configured to store the cache line to main memory in
25 response to the cache line being evicted from the penultimate-level
cache and configured to provide the cache line to the second caching
agent responsive to the communication history information.

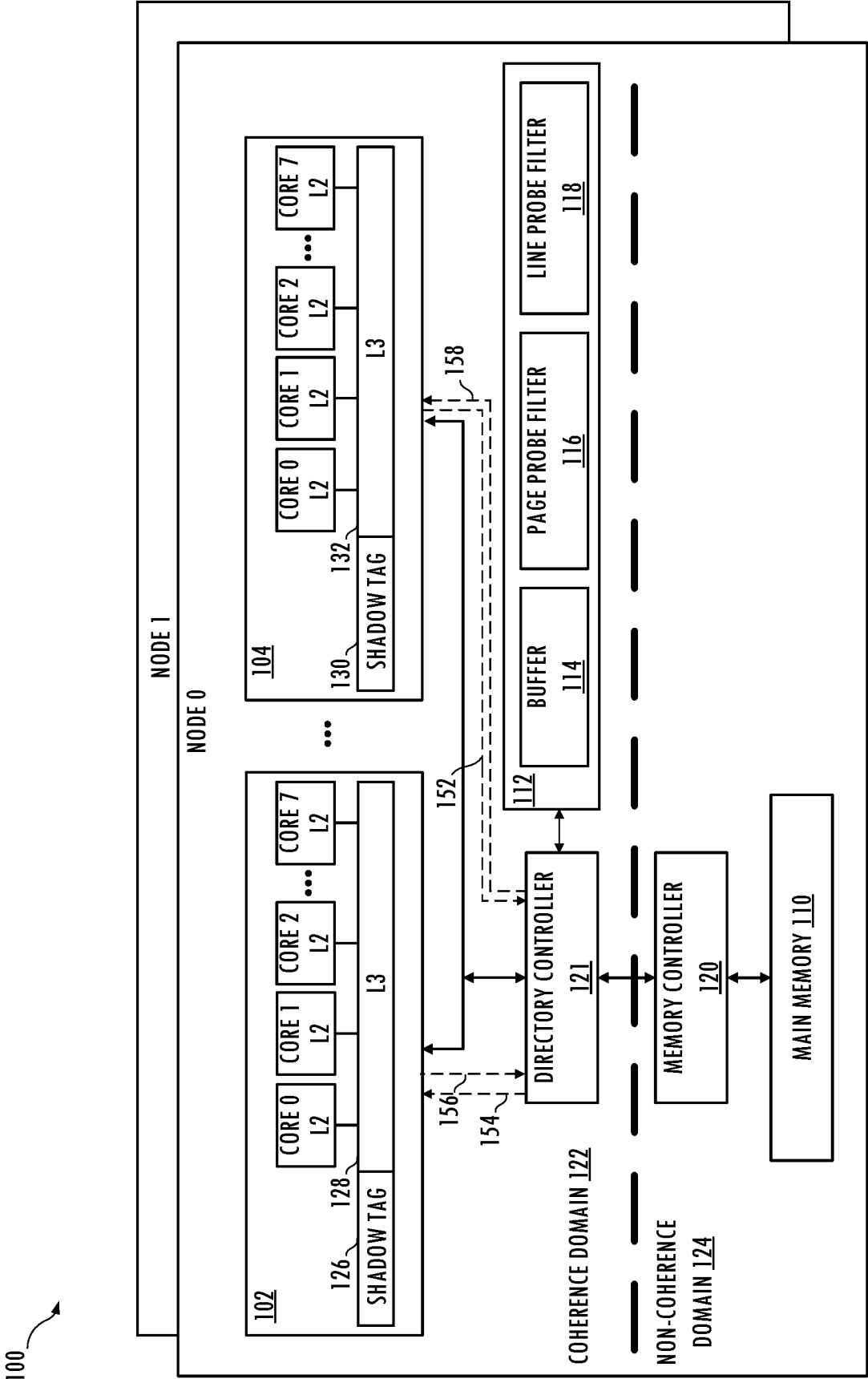


FIG. 1

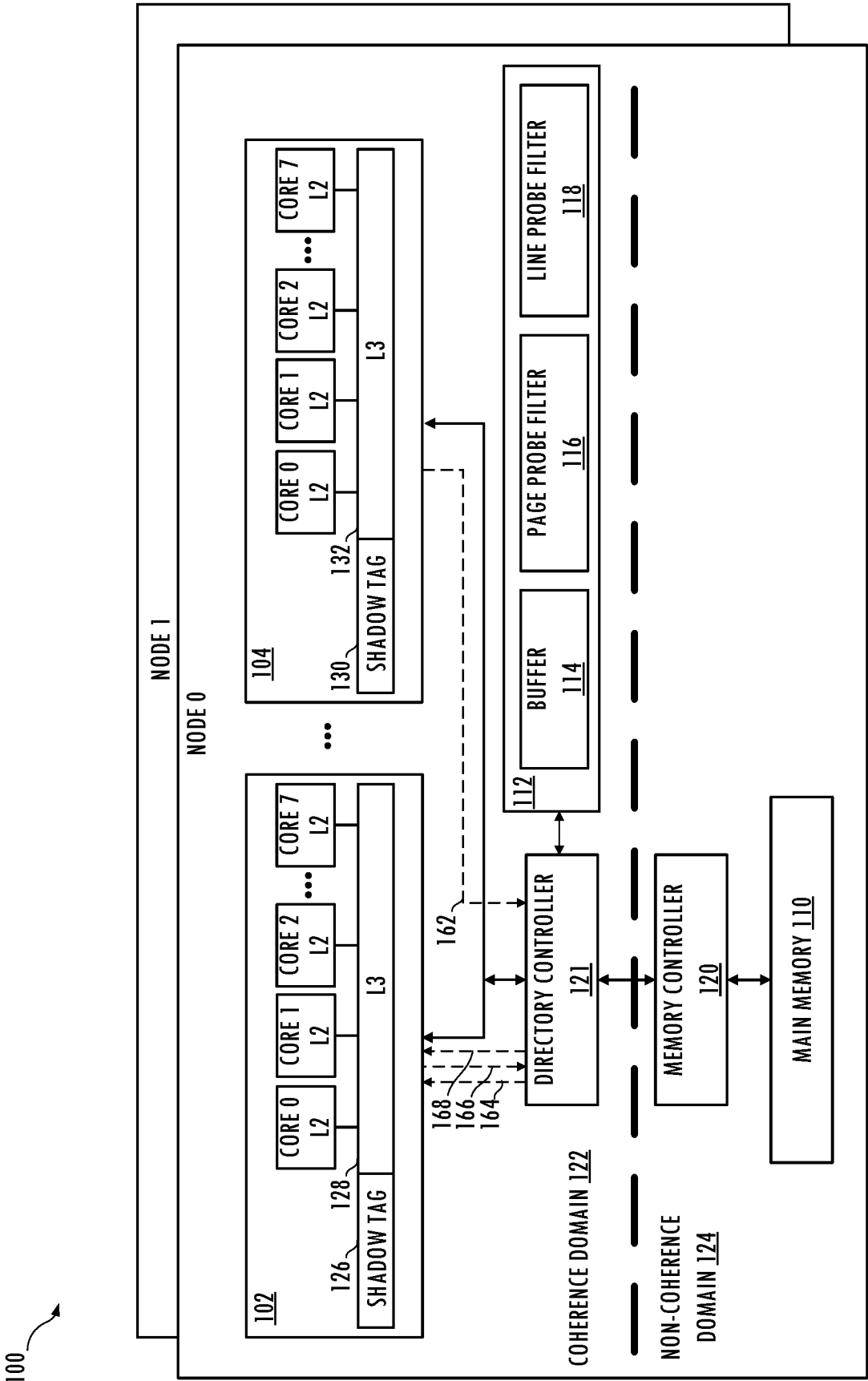


FIG. 2

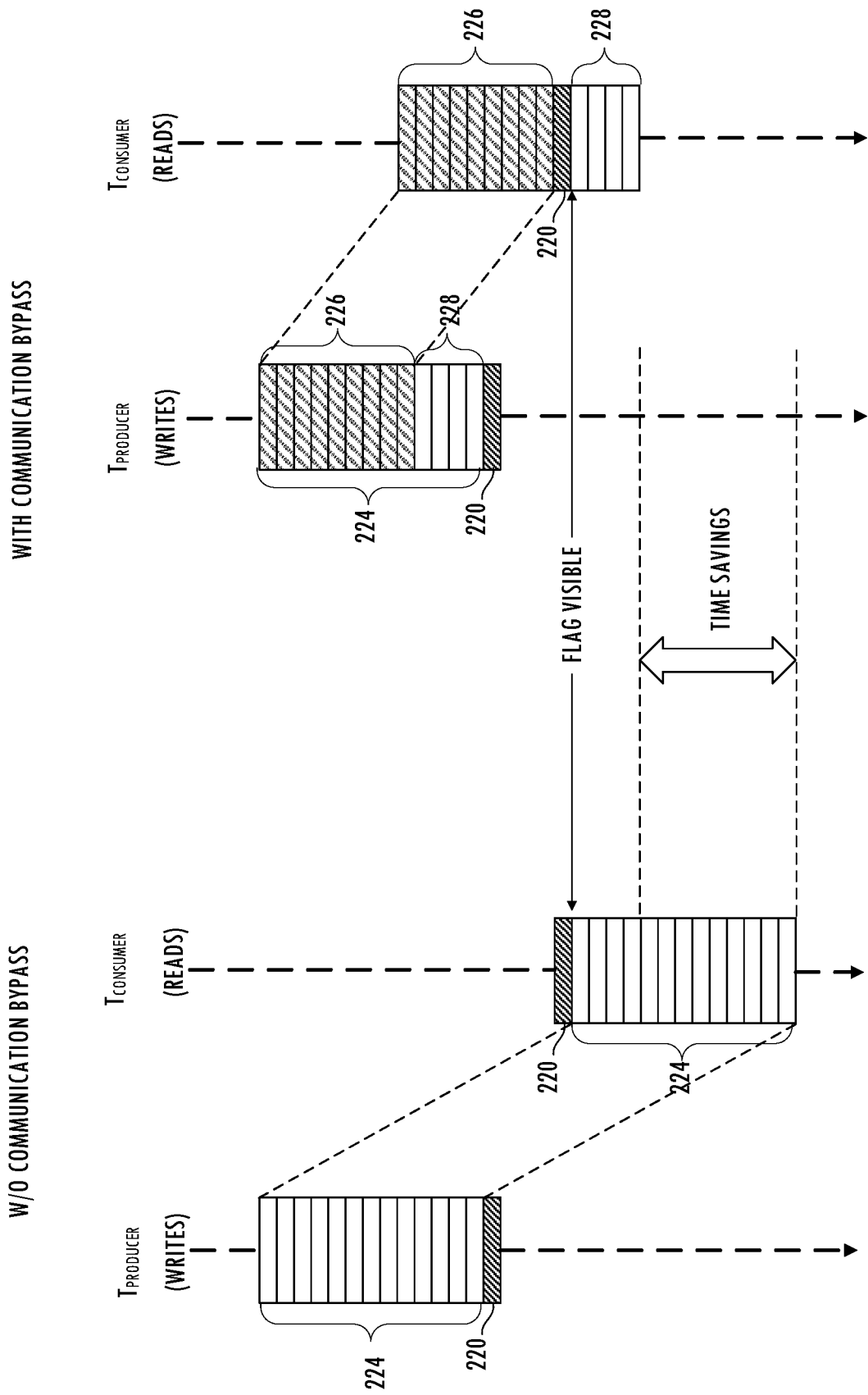


FIG. 3B

FIG. 3A

300



TAG	STATE	CLV	LSV	RSV	OWNER	PREV_OWNER	CSTATE
-----	-------	-----	-----	-----	-------	------------	--------

FIG. 4

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2018/014666**A. CLASSIFICATION OF SUBJECT MATTER****G06F 12/0897(2016.01)i**

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 12/0897; G06F 12/12; G06F 12/10; G06F 12/00; G06F 12/08

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 models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKOMPASS(KIPO internal) & keywords: caching agent, penultimate-level cache, last-level cache, shadow tag entry, bypass, probe filter, communication history**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2015-0269090 A1 (TEXAS INSTRUMENTS INCORPORATED) 24 September 2015 See paragraphs [0049]-[0060]; claim 5; and figures 7-8.	1-20
A	US 2004-0059877 A1 (JEFFREY DOUGLAS BROWN et al.) 25 March 2004 See paragraphs [0027]-[0033]; claim 1; and figures 2-5.	1-20
A	US 2014-0189254 A1 (ILAN PARDO et al.) 03 July 2014 See paragraphs [0009]-[0017]; and figure 1.	1-20
A	US 2016-0203079 A1 (INTERNATIONAL BUSINESS MACHINES CORPORATION) 14 July 2016 See paragraphs [0015]-[0030]; claim 1; and figure 1.	1-20
A	US 2004-0268054 A1 (JIH-KWON PEIR et al.) 30 December 2004 See paragraphs [0027]-[0034]; claim 1; and figure 3.	1-20



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 another 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

"T" 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

10 May 2018 (10.05.2018)

Date of mailing of the international search report

10 May 2018 (10.05.2018)

Name and mailing address of the ISA/KR

International Application Division

Korean Intellectual Property Office

189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea

Facsimile No. +82-42-481-8578

Authorized officer

NHO, Ji Myong

Telephone No. +82-42-481-8528



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/014666

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2015-0269090 A1	24/09/2015	US 2012-0075005 A1	29/03/2012
		US 2012-0079102 A1	29/03/2012
		US 2012-0079155 A1	29/03/2012
		US 2012-0079203 A1	29/03/2012
		US 2012-0079204 A1	29/03/2012
		US 2012-0079247 A1	29/03/2012
		US 2012-0191913 A1	26/07/2012
		US 2012-0191914 A1	26/07/2012
		US 2012-0191915 A1	26/07/2012
		US 2012-0191916 A1	26/07/2012
		US 2012-0192027 A1	26/07/2012
		US 2012-0197954 A1	02/08/2012
		US 2012-0198160 A1	02/08/2012
		US 2012-0198161 A1	02/08/2012
		US 2012-0198162 A1	02/08/2012
		US 2012-0198163 A1	02/08/2012
		US 2012-0198164 A1	02/08/2012
		US 2012-0198165 A1	02/08/2012
		US 2012-0198166 A1	02/08/2012
		US 2012-0198171 A1	02/08/2012
		US 2012-0198192 A1	02/08/2012
		US 2012-0198272 A1	02/08/2012
		US 2012-0198310 A1	02/08/2012
		US 2012-0260031 A1	11/10/2012
		US 2012-0290755 A1	15/11/2012
		US 2012-0290756 A1	15/11/2012
		US 2012-0314833 A1	13/12/2012
		US 2013-0243148 A1	19/09/2013
		US 2015-0178221 A1	25/06/2015
		US 2016-0034396 A1	04/02/2016
		US 8532247 B2	10/09/2013
		US 8547164 B2	01/10/2013
		US 8560896 B2	15/10/2013
		US 8598932 B2	03/12/2013
		US 8607000 B2	10/12/2013
		US 8656105 B2	18/02/2014
		US 8661199 B2	25/02/2014
		US 8683115 B2	25/03/2014
		US 8683137 B2	25/03/2014
		US 8707127 B2	22/04/2014
		US 8732398 B2	20/05/2014
		US 8732416 B2	20/05/2014
		US 8832166 B2	09/09/2014
		US 8856446 B2	07/10/2014
		US 8880855 B2	04/11/2014
		US 8904110 B2	02/12/2014
		US 8904115 B2	02/12/2014
		US 8904260 B2	02/12/2014
		US 9003122 B2	07/04/2015

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2018/014666

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		US 9009408 B2	14/04/2015
		US 9075743 B2	07/07/2015
		US 9075744 B2	07/07/2015
		US 9183084 B2	10/11/2015
		US 9189331 B2	17/11/2015
		US 9195610 B2	24/11/2015
		US 9268708 B2	23/02/2016
		US 9298643 B2	29/03/2016
		US 9575901 B2	21/02/2017
US 2004-0059877 A1	25/03/2004	US 7194586 B2	20/03/2007
US 2014-0189254 A1	03/07/2014	US 9268697 B2	23/02/2016
US 2016-0203079 A1	14/07/2016	US 2014-0237186 A1	21/08/2014
		US 9323675 B2	26/04/2016
		US 9645931 B2	09/05/2017
US 2004-0268054 A1	30/12/2004	AU 2001-66815 A1	08/01/2002
		AU 6681501 A	08/01/2002
		CN 1307559 C	28/03/2007
		CN 1447940 A	08/10/2003
		EP 1314088 A2	28/05/2003
		TW 548545 B	21/08/2003
		US 6725341 B1	20/04/2004
		US 7076613 B2	11/07/2006
		WO 02-01366 A2	03/01/2002
		WO 02-01366 A3	04/07/2002