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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
8 June 2017 (08.06.2017)

WIPO | PCT

(10) International Publication Number
WO 2017/095554 A1

(51) International Patent Classification:
G06F 9/30 (2006.01) *G06F 11/10* (2006.01)

(21) International Application Number:
PCT/US2016/058660

(22) International Filing Date:
25 October 2016 (25.10.2016)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
14/959,440 4 December 2015 (04.12.2015) US

(71) Applicant: **INTEL CORPORATION** [US/US]; 2200 Mission College Boulevard, Santa Clara, California 95054 (US).

(72) Inventor: **BOOM, Douglas D.**; 5261 NW Wickiup Way, Portland, Oregon 97229 (US).

(74) Agents: **MARLINK, Jeffrey S.** et al.; Schwabe, Williamson & Wyatt, P.C., 1211 SW 5th Avenue, Suite 1500-1900, Portland, Oregon 97204 (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, JP, KE, KG, 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, 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).

Published:

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

— *with amended claims (Art. 19(1))*

(54) Title: PHYSICAL LAYER DEVICE OPERATION SYSTEM AND METHOD

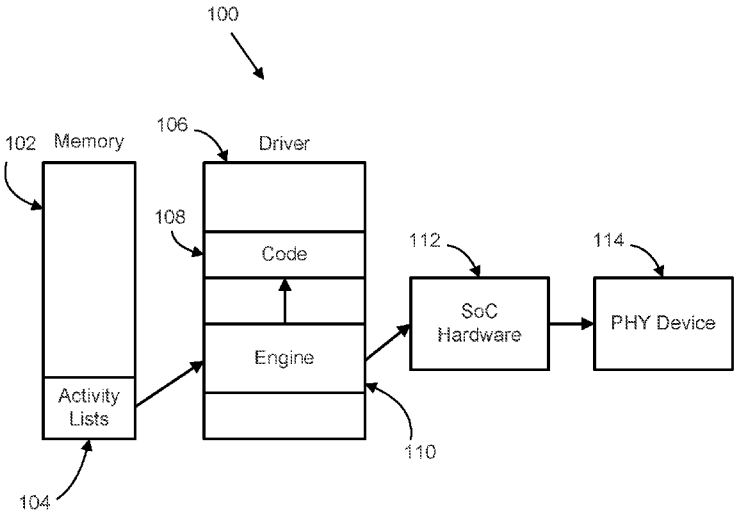


FIGURE 1

(57) Abstract: Apparatuses, systems and methods associated with causing a physical layer (PHY) device are disclosed herein. In embodiments, an apparatus may include a memory device to store one or more activity lists associated with one or more external PHY devices, external to the apparatus, including the first external PHY device. The apparatus may further include a processor, that executes an engine, to receive a request for performance of the operation by the first external PHY device, identify an activity list associated with the first external PHY device from the one or more activity lists, identify an activity to effectuate performance of the operation from the activity list associated with the first external PHY device and cause the first external PHY device to perform the operation in accordance with the activity.

WO 2017/095554 A1

PHYSICAL LAYER DEVICE OPERATION SYSTEM AND METHOD

Related Application

This application claims priority to U.S. Patent Application 14/959,440, entitled "PHYSICAL LAYER DEVICE OPERATION SYSTEM AND METHOD," filed December 4, 2015.

Technical Field

The present disclosure relates to the fields of electronic circuits and computing. More particularly, the present disclosure relates to operation of a physical layer (PHY) device, e.g., within a computing device.

Background

The background description provided herein is for the purpose of generally presenting the context of the disclosure. Unless otherwise indicated herein, the materials described in this section are not prior art to the claims in this application and are not admitted to be prior art by inclusion in this section.

Enablement of a PHY device within a computing device often involves communication between a media access controller (MAC) and the PHY device. In order for the communication to occur between the components, both the MAC and the PHY device must be communicating using the same programming construct. As no standard programming construct has been defined for communication between MACs and PHY devices, multiple different programming constructs may be utilized for communicating between MACs and PHY devices depending on the MACs and/or the PHY devices implemented.

In legacy computing devices, often the MAC and the PHY device were produced by different manufacturers who would develop the components without interaction between the manufacturers. The lack of interaction between the manufacturers would lead to the MAC and the PHY device communicating in a different programming construct from each other. In order to rectify this issue, the MAC manufacturer, the producer of the computing device and/or motherboard, and/or a third party would reprogram binary code within a driver associated with the PHY device to enable communication between the MAC and the PHY device. This approach was complicated, time consuming and costly to perform.

Brief Description of the Drawings

Embodiments will be readily understood by the following detailed description in conjunction with the accompanying drawings. To facilitate this description, like reference numerals designate like structural elements. Embodiments are illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings.

Figure 1 illustrates an example communication flow among components of an example computing device, according to various embodiments.

Figure 2 illustrates an example abstract representation of computing structures utilized for PHY enablement, according to various embodiments.

Figure 3 illustrates an example process of PHY enablement, according to various embodiments.

Figure 4 illustrates an example activity list header structure, according to various embodiments.

Figure 5 illustrates an example table of contents entry structure, according to various embodiments.

Figure 6 illustrates an example table of contents entry header structure, according to various embodiments.

Figure 7 illustrates another example table of contents entry structure, according to various embodiments.

Figure 8 illustrates an example action structure, according to various embodiments.

Figure 9 illustrates an example activity header structure, according to various embodiments.

Figure 10 illustrates an example computing device that may employ the apparatuses and/or methods described herein.

Detailed Description

Apparatuses, methods and storage medium associated with physical layer (PHY) device operation are disclosed herein. In embodiments, an apparatus may include one or more processors, devices, and/or circuitry to identify a PHY device and access an activity list associated with the PHY device. The apparatus may access activities within the activity list that enable communication between a media access controller (MAC) and the PHY device to initiate

PHY-level tasks to be performed by the PHY device. The apparatuses, methods and storage medium disclosed herein may configure and control third party PHY devices without modifications to the driver binary.

In the following detailed description, reference is made to the accompanying drawings which form a part hereof wherein like numerals designate like parts throughout, and in which is shown by way of illustration embodiments that may be practiced. It is to be understood that other embodiments may be utilized and structural or logical changes may be made without departing from the scope of the present disclosure. Therefore, the following detailed description is not to be taken in a limiting sense, and the scope of embodiments is defined by the appended claims and their equivalents.

Aspects of the disclosure are disclosed in the accompanying description. Alternate embodiments of the present disclosure and their equivalents may be devised without parting from the spirit or scope of the present disclosure. It should be noted that like elements disclosed below are indicated by like reference numbers in the drawings.

Various operations may be described as multiple discrete actions or operations in turn, in a manner that is most helpful in understanding the claimed subject matter. However, the order of description should not be construed as to imply that these operations are necessarily order dependent. In particular, these operations may not be performed in the order of presentation. Operations described may be performed in a different order than the described embodiment. Various additional operations may be performed and/or described operations may be omitted in additional embodiments.

For the purposes of the present disclosure, the phrase "A and/or B" means (A), (B), or (A and B). For the purposes of the present disclosure, the phrase "A, B, and/or C" means (A), (B), (C), (A and B), (A and C), (B and C), or (A, B and C).

For the purposes of the present disclosure, the phrase "user" is not limited to an individual. The phrase "user" may be interpreted as an entity, a corporation, a group of individuals, or some combination thereof.

The description may use the phrases "in an embodiment," "in embodiments," or "in some embodiments," which may each refer to one or more of the same or different embodiments. Furthermore, the terms "comprising," "including," "having," and the like, as used with respect to embodiments of the present disclosure, are synonymous.

As used herein, the term "circuitry" may refer to, be part of, or include an Application Specific Integrated Circuit (ASIC), an electronic circuit, a processor (shared, dedicated, or group) and/or memory (shared, dedicated, or group) that execute one or more software or firmware programs, a combinational logic circuit, and/or other suitable components that provide the described functionality.

Referring now to Figure 1, wherein an example communication flow among components of an example computing device 100, according to various embodiments, is shown. As illustrated, an apparatus according to embodiments may include a memory device 102, a system on a chip (SoC) device including a driver 106 and SoC hardware, a PHY device 114, or some combination thereof. The memory device 102, the SoC device and the PHY device 114 may be coupled to each other such that each device is able to communicate with the other devices.

The memory device 102 may include a non-volatile memory device, a volatile memory device, memory residing on the SoC device, a memory device external to the SoC device, a random-access memory (RAM) device, a read-only memory (ROM) device, or some combination thereof. The memory device 102 may store one or more activity lists 104 that enable the SoC device to communicate with the PHY device 114 to cause the PHY device 114 to perform a PHY-level task. Each of the activity lists 104 may be associated with a PHY device, a type of PHY device, a programming construct for communicating with one or more PHY devices, a SoC device, a type of SoC device, a MAC residing on the SoC device, a type of MAC residing on the SoC device, or some combination thereof. Each activity list may include one or more activities, wherein each activity may cause the PHY device to perform a PHY-level task in response to execution of the activity.

A driver 106 may communicate with the memory device 102 and may access the one or more activity lists 104 stored on the memory device 102. The driver 106 may reside on the SoC device. The driver may reside in what would commonly be called firmware and may include a firmware driver. In some embodiments, the driver 106 may reside external to the SoC device and be coupled to the SoC device such that the driver 106 may communicate with the SoC device and/or vice versa.

The driver may be software to enable the SoC device to communicate with the PHY device 114. Execution of a portion of the software may cause the PHY device 114 to perform a PHY-level task. The driver 106 may include a firmware driver, a device driver, or some

combination thereof. The driver 106 may be associated with the SoC device, the PHY device, or some combination thereof.

The driver 106 may include a processing engine 110. The processing engine 110 may have been installed on the apparatus with the driver, may have been added to the driver after installation of the driver, may have replaced and/or altered code within the driver to generate the processing engine 110, or some combination thereof. In some embodiments, the processing engine 110 may be separate from the driver 106 and coupled to the driver 106.

The processing engine 110 may identify the PHY device 114, a programming construct to be utilized for communicating with the PHY device 114, a MAC of the SoC device, or some combination thereof. The identification of the PHY device 114 may include receiving, by the processing engine, a PHY device identifier associated with the PHY device 114. The PHY device identifier may be received in a request for the PHY device to perform some PHY-level task. The processing engine 110 may access the activity lists 104 stored on the memory device 102 and may determine which of the activity lists 104 should be utilized for communicating with the PHY device 114 based the identified PHY device 114, the programming construct, the MAC, or some combination thereof.

The driver 106 may receive a request for performance of a PHY-level task from the SoC device and/or the PHY device 114. The processing engine 110 may utilize the request to determine which PHY-level task for which performance is being requested. The processing engine 110 may utilize the identified PHY-level task to identify one of the activities associated with the PHY-level task within the activity list that should be utilized for communicating with the PHY device 114. The activity associated with the PHY-level task may cause the PHY device 114 to perform the PHY-level task when executed.

The processing engine 110 may operate in combination with the code portion 108 of the driver 106. The processing engine 110 may utilize the code portion 108 of the driver 106 to translate the activity into one or more commands in a programming construct that can be read by the PHY device 114. The code portion 108 may provide for translation of the activity into a specific programming construct, such as MDI code and/or I2C code. In some examples, the activity may be translated into PHY commands, MAC commands, or some combination thereof.

The one or more commands produced through the translation may be communicated by SoC hardware 112 to the PHY device 114. The PHY device 114 may be located external to the

SoC device, including the SoC hardware 112. The one or more commands may be communicated from the SoC hardware 112 to the PHY device 114 via one or more busses, including an advanced graphics port, an enhanced integrated drive electronics type bus, an extended industry standard architecture bus, an IEEE 1394 compliant bus, a personal computer memory card international association compliant bus, a peripheral component interconnect bus, a small computer system interface bus, a universal serial bus, a parallel port, a PS/2 port, a serial port, or some combination thereof. The one or more commands may cause the PHY device 114 to perform a PHY-level task associated with the activity. In some examples, the PHY device 114 may transmit a confirmation signal back to the SoC hardware 112 indicating that the PHY-level task has been completed.

Referring now to Figure 2, wherein a representation of computing structures utilized for PHY operation, according to various embodiments, is shown. The computing structures may be stored in a PHY activity lists portion 202 of a memory device, such as the memory device 102 of Figure 1 that stores the one or more activity lists 104 in a portion of the memory device 102.

The PHY activity lists portion 202 may include one or more activity lists, including activity list 204. The one or more activity lists may have similar features to other activity lists described throughout this disclosure, including the activity lists 104 of Figure 1. Each activity list may be associated with a PHY device, a type of PHY device, a programming construct utilized for communicating with a type of PHY device, a SoC device, a MAC device, or some combination thereof. The activity lists may be utilized for communication between a SoC device and a PHY device.

The PHY activity lists portion 202 may further include a table of contents 205 associated with the one or more activity lists. The table of contents 205 may be utilized to determine which one of the activity lists should be utilized for communication between the SoC device and the PHY device based on the PHY device, the programming construct, the SoC device, and/or the MAC. A processing engine, such as processing engine 110 of Figure 1, may access the table of contents 205 and perform the determination. The table of contents 205 may further indicate a storage location of each of the activity lists in the memory device. In other embodiments, another means of determining which one of the activity lists should be utilized may be performed, such as by comparing some identifying factor of the PHY device with data included in the activity list 204.

In some embodiments, in response to the computing device determining which one of the activity lists should be utilized for communication between the SoC device and the PHY device, the activity list, or a copy thereof, may be stored in a different location, from where initially accessed, within the computing device allowing for quicker access to the activity list. For example, when the one or more activity lists are stored on a memory device separate from the SoC device, a copy of the activity list to be utilized may be stored in a memory of the SoC device allowing for quicker access to the activity list by the SoC device.

Expanded activity list representation 206 is a representation of the activity list 204 showing contents of the activity list 204. The expanded activity list representation 206 may be representative of one, all or some portion of the activity lists stored in the PHY activity lists portion 202.

The activity list 204 may include one or more activities, including activity 208, as shown in exampled activity list representation 206. Each of the activities within the activity list 204 may be associated with a PHY-level task to be performed by a PHY device associated with the activity list 204. In some embodiments, the activity list 204 may be limited to activities for making the PHY device operational, while other activities that are not for making the PHY device operational may be excluded from the activity list 204.

The activity list 204 may include a table of contents 209. The table of contents 209 may include data for identifying which activity is associated with which PHY-level task. The processing engine may access the table of contents 209 and may identify an activity from the activity list 204, associated with a desired PHY-level task, to be performed by the PHY device based on the data included in the table of contents 209. The table of contents 209 may further indicate a location of each of the activities within the activity list 204.

Expanded activity representation 210 is a representation of activity 208. The expanded activity representation 210 may be representative of one, all or some portion of the activities within the activity list 204.

The activity 208 may include one or more actions, including action 212, and a header 211. The header 211 may include one or more of the features described in relation to activity header structure 900 below. Each action may include code to cause a PHY device, such as PHY device 114 of Figure 1, to perform a portion of a PHY-level task associated with the activity 208. The actions may cause the PHY device to perform a PHY-level task associated with the activity

208 in response to a SoC device executing the actions. The SoC device may begin execution of the activity by executing the first action within the activity and progressively executing the actions within the activity until the last action within the activity is executed. In some embodiments, the SoC device may begin execution at the first action and may execute a specified number of actions after the first action, regardless of the number of actions within the activity 208.

Further, the activity list 204 may include a table of contents identifying a location of the activities within the corresponding activity list, a number of the activities within the corresponding activity list, a length of the activities within the activity list, a PHY-level task associated with each of the activities, or some combination thereof. In some embodiments, the table of contents may be located in a memory location between a header of the activity list and a first activity within the activity list.

In some embodiments, each activity list of the activity lists 104 may include a checksum value used for performing a checksum operation. The checksum operation may be performed on the checksum value to verify that the corresponding activity list has not been corrupted, has not been altered by an unauthorized user, or some combination thereof. The checksum operation may be performed in response to the computing device determining that the activity list is to be utilized for communication with the PHY device, the computing device accessing the activity list, or some combination thereof.

Referring now to Figure 3, wherein a process 300 of PHY operation, according to various embodiments, is shown. The process 300 may start in block 302, wherein a PHY device identifier may be obtained. The PHY device identifier may identify a PHY device, such as PHY device 114 of Figure 1. The PHY device identifier may be obtained by a SoC device, such as the SoC device of Figure 1 encompassing the driver 106 and the SoC hardware 112, and may be obtained in response to receiving a request for the PHY device to perform a PHY-level task, detecting that the PHY device is newly installed, initialization of the PHY device, the computing device being turned on, initialization of the computing device, driver startup, or some combination thereof.

Once the PHY device identifier has been obtained, the process 300 may proceed to block 304 where the SoC device determines if there is an activity list associated with the PHY device identifier stored in a memory device accessible by the SoC device and which of the activity lists

stored within the memory device is associated with the PHY device. Determining whether there is an activity list and which activity list is associated with the PHY device may be performed in accordance with any of the processes described throughout this disclosure, including the process of the SoC device accessing a table of contents, such as table of contents 205 of Figure 2, to determine which activity list is associated with the PHY device.

In response to the SoC device determining that there is not an activity list associated with the PHY device identifier stored within the memory device, the process 300 may proceed to block 306, where a default action is performed. The default action may include causing the computing device to display on a display device, coupled to the computing device, a message that the PHY-level task cannot be performed. In some embodiments, the default action may include performing a default PHY-level task, including displaying an indication on the PHY device that the PHY-level task cannot be performed, causing the PHY device and/or the SoC device to continue operation with a subsequent PHY-level task rather than waiting for completion of the current PHY-level task, reinitializing the PHY device, proceeding to block 302 to obtain that the PHY device identifier and continue operation from block 302, or some combination thereof.

In response to the SoC device identifying which activity list is associated with the PHY device, the process 300 may proceed to block 308, where the activity list associated with the PHY device is accessed by the SoC device. In some embodiments, accessing the activity list may include generating a copy of the activity list in a local memory of the SoC device for quicker access to the activity list.

After accessing the activity list, the process 300 may proceed to block 310, where a signature of the activity list is checked in order to verify that the activity list is valid. The signature of the activity list may be located in a header of the activity list. The signature may indicate a user who created of the activity list, a user who modified the activity list, a software program that created and/or modified the activity list, an address (such as an internet protocol address) of a user who created and/or modified the activity list, or some combination thereof. The SoC device may compare the signature to a list of trusted signature to verify that the activity list was created and/or modified by a trusted party.

In some embodiments, the signature may include a checksum value. Verification that the activity list is valid may include performing a checksum calculation on the checksum value. If

the result of the checksum calculation is that the checksum value is the expected value, the activity list may be determined to be valid.

In other embodiments, the activity list may not include a signature. In these embodiments, the process may surpass block 310 and proceed directly from block 308 to block 312.

If the activity list is determined to be invalid based on the signature, the process 300 may proceed to block 306, where the default action is performed. If the activity list is determined to be valid, the process 300 may proceed to block 312 where the SoC device locates an activity within the activity list that is associated with the PHY-level task. Location of the activity may involve comparing an identifying feature of the PHY-level task with data within each activity to determine which activity is associated with the PHY-level task.

In some embodiments, the activity list may include a table of contents, such as table of contents 209 of Figure 2. The SoC device may access the table of contents and utilize data stored in the table of contents to determine which activity is associated with the PHY-level task. The table of contents may further include an indication of where each activity is located within the memory device, the local memory of the SoC device, or some combination thereof. The SoC device may utilize the indication of where each activity is located to access activity data included in the activity.

The process 300 may proceed to block 314, where the SoC device determines if it is able to locate the activity within the activity list. If the SoC device is unable to locate the activity within the activity list, the process 300 may proceed to block 306 where the default action is performed.

If the SoC device is able to locate the activity within the activity list, the process 300 may proceed to block 316, where the activity data included in the activity is accessed by the SoC device. The activity data may include an activity header, one or more actions, or some combination thereof. The activity data may include the features described in relation to the expanded activity representation 210 of the activity 208, illustrated in Figure 2.

The SoC device may start accessing the activity by proceeding to a header of the activity. The header of the activity may include data associated with the contents of the activity, including a checksum value.

The process 300 may proceed to block 318, where a checksum operation is performed on

the checksum value included in the header of the activity. The checksum operation may be performed to verify that the activity has not been corrupted. The checksum operation may include a parity byte and/or parity word checksum operation, a modular checksum operation, a position-dependent checksum operation, or some combination thereof. In some embodiments, the checksum operation may include a CRC8 checksum operation. The CRC8 checksum operation may utilize the polynomial expression $X^8 + X^2 + X^1 + X^0$, or 0x07 in normal polynomial representation.

If the checksum operation indicates that the activity has been corrupted (i.e. the checksum value is not an expected value), the process 300 may proceed to block 306 where the default action is performed. If the checksum operation indicates that the activity has not been corrupted (i.e. the checksum value is the expected value), the process 300 may proceed to block 320 where the activity is executed.

In some embodiments, the header may not include a checksum value. Further, in some embodiments, the activity may not include a header. In some embodiments where the activity does not include a header and/or a checksum value, the process 300 may exclude block 318 and may proceed directly from block 316 to block 320. In these embodiments, the SoC device may proceed directly to the first action of the activity list in response to accessing the activity.

In block 320, the SoC device may execute the activity. In response to execution of the activity, the SoC device may cause the PHY device to perform the PHY-level task. Execution of the activity may include any of the processes of executing an activity, including the process described in Figure 1 performed by the driver 106 and the SoC hardware 112. In particular, execution of the activity may involve translating the actions included in the activity list into one or more commands in a programming construct associated with the PHY device, where the PHY device performs the PHY-level task in response to executing the commands. The actions included in the activity may be translated into one or more PHY commands, MAC commands, or some combination thereof.

Referring now to Figure 4, wherein an activity list header structure 400, according to various embodiments, is shown. The activity list header structure 400 may be included in an activity list, such as any of the activity lists in the activity lists 104 of Figure 1 and/or the activity list 204 of Figure 2. The activity list header structure 400 may be located at the beginning of an activity list, prior to a table of contents and activities within the activity list, or some combination

thereof. The activity list header structure 400 may be a structure array data type. The activity list header structure 400 may include a unique activity list name 402.

The activity list header structure 400 may include an identifier token 404. The identifier token 404 may be a 16-bit unsigned integer. The identifier token 404 may be associated with a PHY device, a type of PHY device, a programming construct for communicating with one or more PHY devices, a SoC device, a type of SoC device, a MAC residing on the SoC device, a type of MAC residing on the SoC device, or some combination thereof.

In some embodiments, the identifier token 404 may be utilized for determining which activity list should be utilized in communicating with a PHY device. A SoC device may perform a comparison of a PHY device identifier, such as the PHY device identifier described in relation to block 302 of Figure 3, with the identifier token being used to determine if the activity list should be utilized in communicating with the PHY device.

The activity list header structure 400 may include an activity list size indicator 406. The activity list size indicator may be a 16-bit unsigned integer. The activity list size may indicate an amount of memory space occupied by the activity list, an amount of memory space occupied by each activity within the activity list, or some combination thereof.

The activity list header structure 400 may include a number of activities indicator 408. The number of activities indicator 408 may be a 16-bit unsigned integer. The number of activities indicator 408 may indicate a number of activities within the activity list for which the activity list header structure is associated.

The activity list header structure 400 may include a version indicator 410. The version indicator 410 may be a 16-bit unsigned integer. The version indicator 410 may indicate a version of the activity list in which the activity list header structure 400 is included. The version indicator 410 may be incremented and/or updated in another matter in response to the activity list being updated and/or amended. In some embodiments, the version indicator 410 may further indicate the user that generated the updated version of the activity list, a software program that generated the updated version of the activity list, or some combination thereof.

The activity list header structure 400 may include a checksum value 412. A checksum operation may be performed on the checksum value 412 in order to verify that the activity list has not been corrupted. The checksum operation may be performed in accordance with any of the checksum operations described throughout this disclosure, including the checksum operation

described in relation to the activity lists 104 of Figure 1 and/or the checksum operation performed in relation to block 310 of Figure 3.

Referring now to Figure 5, wherein a table of contents entry structure 500, according to various embodiments, is shown. The table of contents entry structure 500 may be included in an activity list, such as any of the activity lists in the activity lists 104 of Figure 1 and/or the activity list 204 of Figure 2. The table of contents entry structure 500 may be located after a header of the activity list and prior to activities within the activity list. The table of contents entry structure 500 may be a structure array data type. The table of contents entry structure 500 may include a unique table of contents entry name 502.

The table of contents entry structure 500 may include an activity name 504. The activity name 504 may be associated with an activity to be performed by a PHY device. The activity name 504 may be a 16-bit unsigned integer.

The table of contents entry structure 500 may include an activity location 506. The activity location 506 may be a 16-bit unsigned integer. The activity location 506 may indicate a location in memory at which the activity is located, a location of the activity relative to beginning and/or end of the activity list, or some combination thereof.

Referring now to Figure 6, wherein a table of contents entry header structure 600, according to various embodiments, is shown. The table of contents entry header structure 600 may be included in an activity list, such as any of the activity lists in the activity lists 104 of Figure 1 and/or the activity list 204 of Figure 2. The table of contents entry header structure 600 may be a structure array data type. The table of contents entry header structure 600 may include a unique header name 602.

The table of contents entry header structure 600 may include a number of activities indication 604. The number of activities indication 604 may be a 16-bit unsigned integer data type. The number of activities indication 604 may indicate a number of the activities within the activity list.

The table of contents entry header structure 600 may include a table of contents item pointer 606. The table of contents item pointer 606 may define a pointer of the type of the table of contents entry structure 500. Accordingly, the table of contents item pointer 606 may point to a table of contents entry within memory, wherein the table of contents entry includes both an activity name, such as the activity name 504 of Figure 5, and an activity location, such as the

activity location 506 of Figure 5. As access to the table of contents by a SoC device progresses through the table of contents, the table of content item pointer 606 may be progressively incremented to a next activity entry in the table of contents until the SoC device identifies the activity associated with a desired PHY-level task to be performed by a PHY device and/or the SoC device has accessed all the activity entries within the table of contents.

Referring now to Figure 7, wherein another table of contents entry structure 700, according to various embodiments, is shown. This table of contents entry structure 700 may be easier to process than table of contents entry structure 500. Accordingly, table of contents entry structure 700 may be utilized for computing devices and/or SoC devices with limited abilities. The table of contents entry structure 700 may be included in an activity list, such as any of the activity lists in the activity lists 104 of Figure 1 and/or the activity list 204 of Figure 2. The table of contents entry structure 700 may be located after a header of the activity list and prior to activities within the activity list. The table of contents entry structure 700 may be a structure array data type. The table of contents entry structure 700 may include a unique table of contents entry name 702.

The table of contents entry structure 700 may include a validity identifier 704. The validity identifier 704 may be a 16-bit unsigned integer. The validity identifier 704 may be utilized to verify that a table of contents entry is a valid entry.

The table of contents entry structure 700 may include an identifier offset 706. The identifier offset 706 may be a 16-bit unsigned integer. The identifier offset 706 may indicate an offset amount that a first activity is after a header of the activity list.

Referring now to Figure 8, wherein an action structure 800, according to various embodiments, is shown. The action structure 800 may define an action, such as action 212 of Figure 2. The action structure 800 may be a structure array data type. The action structure 800 may include a unique action name 802.

The action structure 800 may include a command field 804. The command field 804 may be an 8-bit unsigned integer. The command field 804 may include a command to execute a certain function. The command within the command field 804 may be a MAC command and/or a PHY command. The command field 804 may include a hardware level command, such as a load command, a swap command, a copy command, a test command, a less than comparison command, a greater than comparison command, an equal to comparison command, a jump

command, a nested jump command, an unconditional jump command, a read from multi-media domain (MMD) command, a read from I2C bus command, a read from multiple document interface (MDI) command, a write to MMD command, a write to I2C bus command, a write to MDI command, an increment command, a decrement command, an logic or command, a logic and command, a logical not command, a write to MAC command, a read to MAC command, a mathematical add command, a right shift command, a left shift command, a wait command, a version print command, and/or a RET command (placing a return address in an instruction pointer of the computing device).

The action structure 800 may include a first argument 806 and a second argument 808 associated with the command field 804. The first argument 806 and the second argument 804 may each comprise an 8-bit unsigned integer. The first argument 806 and the second argument 808 may be used to indicate parameter usage for the command field 804. For most actions, arguments may denote source and destination for the operation being performed by the action. For example, the first argument 806 could be "Use Internal Register A as source" and the second argument 808 could be "Use Internal Register B as destination" if the command field 804 is "Copy". In other embodiments, either or both of the first argument 806 and the second argument 808 may be "no argument". The value and use of the first argument 806 and second argument 808 may be dependent of the value and design of the command field 804. Further, some examples of the first argument 806 and/or the second argument 808 may include "Use immediate data" (which may be a reference to data within a data field 810), use Internal Register C or Internal Register D. The command field 804 may determine what the first argument 806 and/or the second argument 808 are in context.

The action structure 800 may include the data field 810. The data field 810 may be a 32-bit unsigned integer. The data within the data field 810 may be used by the command field 804 if directed so by either or both of the first argument 806 and the second argument 808. While other embodiments may use two (or more) data fields, such as data field 810, one is shown here for clarity and, in practice, may be all that is desired. Data fields may not always used, although may be maintained for uniformity of the action structure 800 and/or prospective use at another time.

Referring now to Figure 9, wherein an activity header structure 900, according to various

embodiments, is shown. The activity header structure 900 may be included in an activity, such as activity 208 of Figure 2. The activity header structure 900 may be located at a beginning of the activity, prior to any actions within the activity. The activity header structure 900 may be a structure array data type. The activity header structure 900 may include a unique activity header name 902.

The activity header structure 900 may include an activity name 904. The activity name 904 may be a 16-bit unsigned integer. The activity name 904 may indicate a PHY-level operation to be performed by a PHY device.

The activity header structure 900 may include a size indication 906. The size indication 906 may be a 16-bit unsigned integer. The size indication 906 may indicate a size of the activity. The size indication 906 may indicate an amount of actions (such as action 212 of Figure 2) within the activity, an amount of memory occupied by the activity, an amount of memory occupied by each action within the activity, or some combination thereof.

The activity header structure 900 may include a checksum value 908. The checksum value 908 may be a 16-bit unsigned integer. A checksum operation may be performed with the checksum value 908 to verify that the activity has not been corrupted. The checksum operation may include any of the checksum operations described throughout this disclosure, such as the checksum operation described in block 318 of Figure 3.

The activity header structure 900 may include an action pointer 910. The action pointer 910 may be an action structure as defined by the action structure 800 of Figure 8. The action pointer 910 may be initialized to point to a storage location of a first action within the action list. As the activity is executed, the action pointer 910 may increment to point to a storage location of the next action in response to the current action being completed. Each action within the action list may be the same size, such that each time the action pointer 910 increments to the next action the action pointer 910 increments by a same amount of memory locations for each increment. The action pointer 910 may be continually incremented through the actions within the activity until the last action within the activity is executed, at which point the action pointer 910 may be directed to point to the storage location of the first action within the activity.

In some embodiments, the action pointer 910 may be prevented from pointing to a memory location outside of the current activity. The current activity may be defined to be certain size based on the size indication 906. The current activity may be defined as being a size

indicated by the size indication 906 and may be measured as the indicated size from the first memory location of the activity, from the first memory location of the activity header, from the after the last memory location of the activity header, or some combination thereof.

In response to an action within the activity attempting to address the action pointer 910 to a location outside of the current activity, the SoC device may determine that the activity has been corrupted. The SoC device may prevent performance of any further actions, including the contents of the location outside of the current activity to which the action pointer 910 is attempting to be addressed, in response to the determination that the activity has been corrupted. Further, the SoC device may perform a default action, such as the default action described in relation to block 306 of Figure 3, in response to the determination that the activity has been corrupted.

In other embodiments, the SoC device may ignore any attempts to address the action pointer 910 to a location outside of the current activity and continue incrementing the action pointer 910 to the storage location of the next action within the current activity. The SoC device may continue to perform the actions within the current activity as the action pointer 910 is incremented through the actions within the current activity without addressing the action pointer 910 to the location outside of the current activity and/or performing the activity associated with the location outside of the current activity.

Figure 10 illustrates an example computing device 1000 that may employ the apparatuses and/or methods described herein (e.g., memory 102, driver 106, SoC hardware 112, PHY device 114 and process 300), in accordance with various embodiments. As shown, computing device 1000 may include a number of components, such as one or more processor(s) 1004 (one shown) and at least one communication chip 1006.

In various embodiments, the one or more processor(s) 1004 each may include one or more processor cores. In various embodiments, the at least one communication chip 1006 may be physically and electrically coupled to the one or more processor(s) 1004. In further implementations, the communication chip 1006 may be part of the one or more processor(s) 1004. In various embodiments, computing device 1000 may include printed circuit board (PCB) 1002. For these embodiments, the one or more processor(s) 1004 and communication chip 1006 may be disposed thereon. In alternate embodiments, the various components may be coupled without the employment of PCB 1002.

Depending on its applications, computing device 1000 may include other components that may or may not be physically and electrically coupled to the PCB 1002. These other components include, but are not limited to, memory controller 1005, volatile memory (e.g., dynamic random access memory (DRAM) 1008), non-volatile memory such as read only memory (ROM) 1010, flash memory 1012, storage device 1011 (e.g., a hard-disk drive (HDD)), an I/O controller 1014, a digital signal processor (not shown), a crypto processor (not shown), a graphics processor 1016, one or more antenna 1018, a display (not shown), a touch screen display 1020, a touch screen controller 1022, a battery 1024, an audio codec (not shown), a video codec (not shown), a global positioning system (GPS) device 1028, a compass 1030, an accelerometer (not shown), a gyroscope (not shown), a speaker 1032, a camera 1034, and a mass storage device (such as hard disk drive, a solid state drive, compact disk (CD), digital versatile disk (DVD)) (not shown), and so forth.

In some embodiments, the one or more processor(s) 1004, flash memory 1012, and/or storage device 1011 may include associated firmware (not shown) storing programming instructions configured to enable computing device 1000, in response to execution of the programming instructions by one or more processor(s) 1004, to practice all or selected aspects of the methods described herein. In various embodiments, these aspects may additionally or alternatively be implemented using hardware separate from the one or more processor(s) 1004, flash memory 1012, or storage device 1011.

In various embodiments, one or more components of the computing device 1000 may include the memory device 102, the driver 106, the SoC hardware 112 and/or the PHY device 114 described herein. For example, the memory device 102, the driver 106, the SoC hardware 112 and/or the PHY device 114 may be included in I/O controller 1014, processor 1004, memory controller 1005, and/or another component of computing device 1000. In some embodiments, I/O controller 1014 may interface with one or more external devices to receive a data signal using the memory device 102, the driver 106, the SoC hardware 112 and/or the PHY device 114. Additionally, or alternatively, the memory device 102, the driver 106, the SoC hardware 112 and/or the PHY device 114 may be used to receive a data signal transmitted between two components of the computing device 1000.

The communication chips 1006 may enable wired and/or wireless communications for the transfer of data to and from the computing device 1000. The term "wireless" and its

derivatives may be used to describe circuits, devices, systems, methods, techniques, communications channels, etc., that may communicate data through the use of modulated electromagnetic radiation through a non-solid medium. The term does not imply that the associated devices do not contain any wires, although in some embodiments they might not. The communication chip 1006 may implement any of a number of wireless standards or protocols, including but not limited to IEEE 702.20, Long Term Evolution (LTE), LTE Advanced (LTE-A), General Packet Radio Service (GPRS), Evolution Data Optimized (Ev-DO), Evolved High Speed Packet Access (HSPA+), Evolved High Speed Downlink Packet Access (HSDPA+), Evolved High Speed Uplink Packet Access (HSUPA+), Global System for Mobile Communications (GSM), Enhanced Data rates for GSM Evolution (EDGE), Code Division Multiple Access (CDMA), Time Division Multiple Access (TDMA), Digital Enhanced Cordless Telecommunications (DECT), Worldwide Interoperability for Microwave Access (WiMAX), Bluetooth, derivatives thereof, as well as any other wireless protocols that are designated as 3G, 4G, 5G, and beyond. The computing device 1000 may include a plurality of communication chips 1006. For instance, a first communication chip 1006 may be dedicated to shorter range wireless communications such as Wi-Fi and Bluetooth, and a second communication chip 1006 may be dedicated to longer range wireless communications such as GPS, EDGE, GPRS, CDMA, WiMAX, LTE, Ev-DO, and others.

In various implementations, the computing device 1000 may be a laptop, a netbook, a notebook, an ultrabook, a smartphone, a computing tablet, a personal digital assistant (PDA), an ultra-mobile PC, a mobile phone, a desktop computer, a server, a printer, a scanner, a monitor, a set-top box, an entertainment control unit (e.g., a gaming console or automotive entertainment unit), a digital camera, an appliance, a portable music player, or a digital video recorder. In further implementations, the computing device 1000 may be any other electronic device that processes data.

It will be apparent to those skilled in the art that various modifications and variations can be made in the disclosed embodiments of the disclosed device and associated methods without departing from the spirit or scope of the disclosure. Thus, it is intended that the present disclosure covers the modifications and variations of the embodiments disclosed above provided that the modifications and variations come within the scope of any claims and their equivalents.

Example 1 may include an apparatus to cause a first external physical layer (PHY)

device, external to the apparatus, to perform an operation. The apparatus may include a memory device to store one or more activity lists associated with one or more external PHY devices including the first external PHY device. The apparatus may further includea processor, that executes an engine, to receive a request for performance of the operation by the first external PHY device, identify an activity list associated with the first external PHY device from the one or more activity lists, identify an activity to effectuate performance of the operation from the activity list associated with the first external PHY device, and cause the first external PHY device to perform the operation in accordance with the activity.

Example 2 may include the apparatus of example 1, wherein the apparatus is a system on chip (SoC) device, and the engine is located in a firmware driver of the SoC device.

Example 3 may include the apparatus of any of the examples 1 and 2, wherein the processor is to further perform a checksum operation on data included in a header of the activity list to verify the activity list has not been corrupted, wherein causation of performance of the operation in accordance with the activity occurs in response to verification that the activity list has not been corrupted.

Example 4 may include the apparatus of example 3, wherein the checksum operation includes a CRC8 checksum operation, and wherein the data included in the header of the activity list includes a checksum value on which the CRC8 checksum operation is performed.

Example 5 may include the apparatus of any of the examples 1 and 2, wherein causation of performance of operation in accordance with the activity includes translation of one or more actions included in the activity into one or more PHY commands that cause the first external PHY device to perform the operation.

Example 6 may include the apparatus of any of the examples 1 and 2, wherein causation of performance of operation in accordance with the activity includes translation of one or more actions included in the activity into one or more media access control commands that cause the first external PHY device to perform the operation.

Example 7 may include the apparatus of any of the examples 1 and 2, wherein the memory device is further to store a table of contents associated with the activity list and the processor is further to access the table of contents, wherein identification of the activity is based on data contained in the table of contents.

Example 8 may include the apparatus of any of the examples 1 and 2, wherein a header of

the activity indicates storage locations of one or more actions included in the activity, wherein the processor is further to retrieve the one or more actions from the storage locations, and wherein causation of performance of the operation in accordance with the activity includes execution of the one or more actions retrieved from the storage locations.

Example 9 may include the apparatus of example 8, wherein the processor is further to identify a first action of the one or more actions, wherein the first action includes an action call to a second action not located in the storage locations of the one or more actions included in the activity and prevent execution of the first action in response to identification of the first action that includes the action call to the second action not located in the storage locations.

Example 10 may include the apparatus of any of the examples 1 and 2, wherein the request for performance of the operation includes a PHY device indicator associated with the first external PHY device, and wherein the activity list associated with the first external PHY device is identified based, at least in part, on the PHY device indicator.

Example 11 may include one or more computer-readable media having instructions stored thereon, wherein the instructions, in response to execution by a device, cause the device to retrieve an activity, which includes one or more actions to effectuate performance of a PHY-level task by an external physical layer (PHY) device located external to the device, from an activity list stored on a memory device, the activity list associated with the external PHY device and cause the external PHY device to perform the PHY-level task in accordance with the activity.

Example 12 may include the one or more computer-readable media of example 11, wherein the instructions further cause the device to receive a request for the external PHY device to perform the PHY-level task, wherein the activity is retrieved in response to reception of the request.

Example 13 may include the one or more computer-readable media of any of the examples 11 and 12, wherein the instructions further cause the device to access a table of contents associated with the activity list, the table of contents stored on the memory device and identify the activity based on data contained in the table of contents.

Example 14 may include the one or more computer-readable media of any of the examples 11 and 12, wherein the instructions further cause the device to perform a checksum operation on data included in a header of the activity to verify that the activity has not been

corrupted, wherein causation of performance of the PHY-level task in accordance with the activity occurs in response to verifying the activity has not been corrupted.

Example 15 may include the one or more computer-readable media of any of the examples 11 and 12, wherein causation of performance of the PHY-level task includes translation of the one or more actions into one or more PHY commands that effectuate performance of the PHY-level task by the external PHY device.

Example 16 may include the one or more computer-readable media of any of the examples 11 and 12, wherein causation of performance of the PHY-level task includes translation of the one or more actions into one or more media access control commands that effectuate performance of the PHY-level task by the external PHY device.

Example 17 may include the one or more computer-readable media of any of the examples 11 and 12, wherein a header of the activity indicates storage locations of the one or more actions, wherein the instructions further cause the device to retrieve the one or more actions from the storage locations, and wherein causation of performance of the PHY-level task in accordance with the activity includes execution of the one or more actions.

Example 18 may include the one or more computer-readable media of example 17, wherein instructions further cause the device to identify a first action of the one or more actions, wherein the first action includes an action call to a second action not located in the storage locations of the one or more actions and prevent execution of the first action in response to identification of the first action that includes the action call to the second action not located in the storage locations.

Example 19 may include the one or more computer-readable media of example 12, wherein the request includes a PHY device identifier associated with the external PHY device, and wherein the instructions further cause the device to identify the activity list associated with the external PHY device based on the PHY device identifier.

Example 20 may include a method to cause a physical layer (PHY) device to perform a first PHY-level task, comprising receiving, by a system on chip (SoC) device, a request to cause the PHY device to perform the first PHY-level task, the PHY device being external to the SoC device, retrieving, by the SoC device, an activity associated with the PHY-level task from an activity list associated with the PHY device, the activity list being stored in memory and including one or more activities associated with one or more PHY-level tasks

including the first PHY-level task, and causing, by the SoC device, the PHY device to perform the first PHY-level task in accordance the activity.

Example 21 may include the method of example 20, further comprising performing, by the SoC device, a checksum operation on data included in a header of the activity to verify the activity has not been corrupted, wherein causation of the performance of the first PHY-level task occurs in response to verification that the activity has not been corrupted.

Example 22 may include the method of any of the examples 20 and 21, wherein causation of the performance of the first PHY-level task includes translating one or more actions included in the activity into one or more PHY commands that cause the external PHY device to perform the first PHY-level task.

Example 23 may include the method of any of the examples 20 and 21, wherein causation of the performance of the first PHY-level task includes translating one or more actions included in the activity into one or more media access control commands that cause the external PHY device to perform the first PHY-level task.

Example 24 may include the method of any of the examples 20 and 21, further comprising accessing, by the SoC device, a table of contents stored in the memory in response to reception of the request and identifying, by the SoC device, the activity based on data included in the table of contents.

Example 25 may include the method of any of the examples 20 and 21, further comprising identifying, by the SoC device, storage locations of one or more actions included in the activity based on data included in a header of the activity and retrieving, by the SoC device, the one or more actions from the storage locations, wherein causation of the performance of the first PHY-level task includes executing, by the SoC device, the one or more actions retrieved from the storage locations.

Example 26 may include the method of example 25, further comprising identifying, by the SoC device, a first action of the one or more actions, wherein the first action includes an action call to a second action not located in the storage locations of the one or more actions and preventing, by the SoC device, execution of the first action in response to identification of the first action that includes the action call to the second action not located in the storage locations.

Example 27 may include the method of example 20, wherein the request includes a PHY device identifier associated with the PHY device, and wherein the method further comprises

identifying, by the SOC device, the activity list associated with the PHY device based on the PHY device identifier.

Example 28 may include an apparatus to cause a first external physical layer (PHY) device to perform an operation, comprising means for receiving a request to cause the PHY device to perform the first PHY-level task, the PHY device being external to the SoC device, means for retrieving an activity associated with the PHY-level task from an activity list associated with the PHY device, the activity list being stored in memory and including one or more activities associated with one or more PHY-level tasks including the first PHY-level task, and means for causing the PHY device to perform the first PHY-level task, through a firmware driver, in accordance the activity.

Example 29 may include the apparatus of example 28, further comprising means for performing a checksum operation on data included in a header of the activity to verify the activity has not been corrupted, wherein causation of the performance of the first PHY-level task occurs in response to verification that the activity has not been corrupted.

Example 30 may include the apparatus of any of the examples 28 and 29, wherein the means for causing of the performance of the first PHY-level task includes means for translating one or more actions included in the activity into one or more PHY commands that cause the external PHY device to perform the first PHY-level task.

Example 31 may include the apparatus of any of the examples 28 and 29, wherein the means for causing of the performance of the first PHY-level task includes means for translating one or more actions included in the activity into one or more media access control commands that cause the external PHY device to perform the first PHY-level task.

Example 32 may include the apparatus of any of the examples 28 and 29, further comprising means for accessing a table of contents stored in the memory in response to reception of the request and means for identifying the activity based on data included in the table of contents.

Example 33 may include the apparatus of any of the examples 28 and 29, further comprising means for identifying storage locations of one or more actions included in the activity based on data included in a header of the activity and means for retrieving the one or more actions from the storage locations, wherein causation of the performance of the first PHY-level task includes executing, by the SoC device, the one or more actions retrieved from the

storage locations.

Example 34 may include the apparatus of example 33, further comprising means for identifying a first action of the one or more actions, wherein the first action includes an action call to a second action not located in the storage locations of the one or more actions and means for preventing execution of the first action in response to identification of the first action that includes the action call to the second action not located in the storage locations.

Claims

What is claimed is:

1. An apparatus to cause a first external physical layer (PHY) device, external to the apparatus, to perform an operation, comprising:

a memory device to store one or more activity lists associated with one or more external PHY devices including the first external PHY device; and

a processor, that executes an engine, to:

receive a request for performance of the operation by the first external PHY device;

identify an activity list associated with the first external PHY device from the one or more activity lists;

identify an activity to effectuate performance of the operation from the activity list associated with the first external PHY device; and

cause the first external PHY device to perform the operation in accordance with the activity.

2. The apparatus of claim 1, wherein the apparatus is a system on chip (SoC) device, and the engine is located in a firmware driver of the SoC device.

3. The apparatus of any of the claims 1 and 2, wherein the processor is to further perform a checksum operation on data included in a header of the activity list to verify the activity list has not been corrupted, wherein causation of performance of the operation in accordance with the activity occurs in response to verification that the activity list has not been corrupted.

4. The apparatus of claim 3, wherein the checksum operation includes a CRC8 checksum operation, and wherein the data included in the header of the activity list includes a checksum value on which the CRC8 checksum operation is performed.

5. The apparatus of any of the claims 1 and 2, wherein causation of performance of operation in accordance with the activity includes translation of one or more actions included in

the activity into one or more PHY commands that cause the first external PHY device to perform the operation.

6. The apparatus of any of the claims 1 and 2, wherein causation of performance of operation in accordance with the activity includes translation of one or more actions included in the activity into one or more media access control commands that cause the first external PHY device to perform the operation.

7. The apparatus of any of the claims 1 and 2, wherein:

the memory device is further to store a table of contents associated with the activity list; and

the processor is further to access the table of contents, wherein identification of the activity is based on data contained in the table of contents.

8. The apparatus of any of the claims 1 and 2, wherein a header of the activity indicates storage locations of one or more actions included in the activity, wherein the processor is further to retrieve the one or more actions from the storage locations, and wherein causation of performance of the operation in accordance with the activity includes execution of the one or more actions retrieved from the storage locations.

9. The apparatus of claim 8, wherein the processor is further to:

identify a first action of the one or more actions, wherein the first action includes an action call to a second action not located in the storage locations of the one or more actions included in the activity; and

prevent execution of the first action in response to identification of the first action that includes the action call to the second action not located in the storage locations.

10. The apparatus of any of the claims 1 and 2, wherein the request for performance of the operation includes a PHY device indicator associated with the first external PHY device, and wherein the activity list associated with the first external PHY device is identified based, at least in part, on the PHY device indicator.

11. A method to cause a physical layer (PHY) device to perform a first PHY-level task, comprising:

receiving, by a system on chip (SoC) device, a request to cause the PHY device to perform the first PHY-level task, the PHY device being external to the SoC device;

retrieving, by the SoC device, an activity associated with the PHY-level task from an activity list associated with the PHY device, the activity list being stored in memory and including one or more activities associated with one or more PHY-level tasks including the first PHY-level task; and

causing, by the SoC device, the PHY device to perform the first PHY-level task in accordance the activity.

12. The method of claim 11, further comprising performing, by the SoC device, a checksum operation on data included in a header of the activity to verify the activity has not been corrupted, wherein causation of the performance of the first PHY-level task occurs in response to verification that the activity has not been corrupted.

13. The method of any of the claims 11 and 12, wherein causation of the performance of the first PHY-level task includes translating one or more actions included in the activity into one or more PHY commands that cause the external PHY device to perform the first PHY-level task.

14. The method of any of the claims 11 and 12, wherein causation of the performance of the first PHY-level task includes translating one or more actions included in the activity into one or more media access control commands that cause the external PHY device to perform the first PHY-level task.

15. The method of any of the claims 11 and 12, further comprising:

accessing, by the SoC device, a table of contents stored in the memory in response to reception of the request; and

identifying, by the SoC device, the activity based on data included in the table of contents.

16. The method of any of the claims 11 and 12, further comprising:

identifying, by the SoC device, storage locations of one or more actions included in the activity based on data included in a header of the activity; and

retrieving, by the SoC device, the one or more actions from the storage locations, wherein causation of the performance of the first PHY-level task includes executing, by the SoC device, the one or more actions retrieved from the storage locations.

17. The method of claim 16, further comprising:

identifying, by the SoC device, a first action of the one or more actions, wherein the first action includes an action call to a second action not located in the storage locations of the one or more actions; and

preventing, by the SoC device, execution of the first action in response to identification of the first action that includes the action call to the second action not located in the storage locations.

18. The method of claim 11, wherein the request includes a PHY device identifier associated with the PHY device, and wherein the method further comprises identifying, by the SOC device, the activity list associated with the PHY device based on the PHY device identifier.

19. One or more computer-readable media having instructions stored thereon, wherein the instructions, in response to execution by a device, cause the device to perform the methods of any of the claims 11-18.

20. An apparatus to cause a first external physical layer (PHY) device to perform an operation, comprising:

means for receiving a request to cause the PHY device to perform the first PHY-level task, the PHY device being external to the SoC device;

means for retrieving an activity associated with the PHY-level task from an activity list associated with the PHY device, the activity list being stored in memory and including one or

more activities associated with one or more PHY-level tasks including the first PHY-level task; and

means for causing the PHY device to perform the first PHY-level task, through a firmware driver, in accordance the activity.

21. The apparatus of claim 20, further comprising means for performing a checksum operation on data included in a header of the activity to verify the activity has not been corrupted, wherein causation of the performance of the first PHY-level task occurs in response to verification that the activity has not been corrupted.

22. The apparatus of any of the claims 20 and 21, wherein the means for causing of the performance of the first PHY-level task includes means for translating one or more actions included in the activity into one or more PHY commands that cause the external PHY device to perform the first PHY-level task.

23. The apparatus of any of the claims 20 and 21, wherein the means for causing of the performance of the first PHY-level task includes means for translating one or more actions included in the activity into one or more media access control commands that cause the external PHY device to perform the first PHY-level task.

24. The apparatus of any of the claims 20 and 21, further comprising:

means for accessing a table of contents stored in the memory in response to reception of the request; and

means for identifying the activity based on data included in the table of contents.

25. The apparatus of any of the claims 20 and 21, further comprising:

means for identifying storage locations of one or more actions included in the activity based on data included in a header of the activity; and

means for retrieving the one or more actions from the storage locations, wherein causation of the performance of the first PHY-level task includes executing, by the SoC device, the one or more actions retrieved from the storage locations.

AMENDED CLAIMS
received by the International Bureau on 4 April 2017 (04.04.2017)

What is claimed is:

1. An apparatus to cause a first external physical layer (PHY) device, external to the apparatus, to perform an operation, comprising:

a memory device to store one or more activity lists associated with one or more external PHY devices including the first external PHY device; and

a processor, that executes an engine, to:

receive a request for performance of the operation by the first external PHY device;

identify an activity list associated with the first external PHY device from the one or more activity lists;

identify an activity to effectuate performance of the operation from the activity list associated with the first external PHY device; and

cause the first external PHY device to perform the operation in accordance with the activity.

2. The apparatus of claim 1, wherein the apparatus is a system on chip (SoC) device, and the engine is located in a firmware driver of the SoC device.

3. The apparatus of any of the claims 1 and 2, wherein the processor is to further perform a checksum operation on data included in a header of the activity list to verify the activity list has not been corrupted, wherein causation of performance of the operation in accordance with the activity occurs in response to verification that the activity list has not been corrupted.

4. The apparatus of claim 3, wherein the checksum operation includes a CRC8 checksum operation, and wherein the data included in the header of the activity list includes a checksum value on which the CRC8 checksum operation is performed.

5. The apparatus of any of the claims 1 and 2, wherein causation of performance of operation in accordance with the activity includes translation of one or more actions included in the activity into one or more PHY commands that cause the first external PHY device to perform the operation.

6. The apparatus of any of the claims 1 and 2, wherein causation of performance of operation in accordance with the activity includes translation of one or more actions included in the activity into one or more media access control commands that cause the first external PHY device to perform the operation.

7. The apparatus of any of the claims 1 and 2, wherein:

the memory device is further to store a table of contents associated with the activity list; and

the processor is further to access the table of contents, wherein identification of the activity is based on data contained in the table of contents.

8. The apparatus of any of the claims 1 and 2, wherein a header of the activity indicates storage locations of one or more actions included in the activity, wherein the processor is further to retrieve the one or more actions from the storage locations, and wherein causation of performance of the operation in accordance with the activity includes execution of the one or more actions retrieved from the storage locations.

9. The apparatus of claim 8, wherein the processor is further to:

identify a first action of the one or more actions, wherein the first action includes an action call to a second action not located in the storage locations of the one or more actions included in the activity; and

prevent execution of the first action in response to identification of the first action that includes the action call to the second action not located in the storage locations.

10. The apparatus of any of the claims 1 and 2, wherein the request for performance of the operation includes a PHY device indicator associated with the first external PHY device, and wherein the activity list associated with the first external PHY device is identified based, at least in part, on the PHY device indicator.

11. A method to cause a physical layer (PHY) device to perform a first PHY-level task, comprising:

receiving, by a system on chip (SoC) device, a request to cause the PHY device to perform the first PHY-level task, the PHY device being external to the SoC device;

retrieving, by the SoC device, an activity associated with the first PHY-level task from an activity list associated with the PHY device, the activity list being stored in memory and including one or more activities associated with one or more PHY-level tasks including the first PHY-level task; and

causing, by the SoC device, the PHY device to perform the first PHY-level task in accordance the activity.

12. The method of claim 11, further comprising performing, by the SoC device, a checksum operation on data included in a header of the activity to verify the activity has not been corrupted, wherein causation of the performance of the first PHY-level task occurs in response to verification that the activity has not been corrupted.

13. The method of any of the claims 11 and 12, wherein causation of the performance of the first PHY-level task includes translating one or more actions included in the activity into one or more PHY commands that cause the external PHY device to perform the first PHY-level task.

14. The method of any of the claims 11 and 12, wherein causation of the performance of the first PHY-level task includes translating one or more actions included in the activity into one or more media access control commands that cause the external PHY device to perform the first PHY-level task.

15. The method of any of the claims 11 and 12, further comprising:

accessing, by the SoC device, a table of contents stored in the memory in response to reception of the request; and

identifying, by the SoC device, the activity based on data included in the table of contents.

16. The method of any of the claims 11 and 12, further comprising:

identifying, by the SoC device, storage locations of one or more actions included in the activity based on data included in a header of the activity; and

retrieving, by the SoC device, the one or more actions from the storage locations, wherein causation of the performance of the first PHY-level task includes executing, by the SoC device, the one or more actions retrieved from the storage locations.

17. The method of claim 16, further comprising:

identifying, by the SoC device, a first action of the one or more actions, wherein the first action includes an action call to a second action not located in the storage locations of the one or more actions; and

preventing, by the SoC device, execution of the first action in response to identification of the first action that includes the action call to the second action not located in the storage locations.

18. The method of claim 11, wherein the request includes a PHY device identifier associated with the PHY device, and wherein the method further comprises identifying, by the SOC device, the activity list associated with the PHY device based on the PHY device identifier.

19. One or more computer-readable media having instructions stored thereon, wherein the instructions, in response to execution by a device, cause the device to perform the methods of any of the claims 11 and 12.

20. An apparatus to cause a first external physical layer (PHY) device to perform an operation, comprising:

means for receiving a request to cause the PHY device to perform the first PHY-level task, the PHY device being external to the SoC device;

means for retrieving an activity associated with the first PHY-level task from an activity list associated with the PHY device, the activity list being stored in memory and including one or more activities associated with one or more PHY-level tasks including the first PHY-level task; and

means for causing the PHY device to perform the first PHY-level task, through a firmware driver, in accordance the activity.

21. The apparatus of claim 20, further comprising means for performing a checksum operation on data included in a header of the activity to verify the activity has not been corrupted, wherein causation of the performance of the first PHY-level task occurs in response to verification that the activity has not been corrupted.

22. The apparatus of any of the claims 20 and 21, wherein the means for causing of the performance of the first PHY-level task includes means for translating one or more actions included in the activity into one or more PHY commands that cause the external PHY device to perform the first PHY-level task.

23. The apparatus of any of the claims 20 and 21, wherein the means for causing of the performance of the first PHY-level task includes means for translating one or more actions included in the activity into one or more media access control commands that cause the external PHY device to perform the first PHY-level task.

24. The apparatus of any of the claims 20 and 21, further comprising:

means for accessing a table of contents stored in the memory in response to reception of the request; and

means for identifying the activity based on data included in the table of contents.

25. The apparatus of any of the claims 20 and 21, further comprising:

means for identifying storage locations of one or more actions included in the activity based on data included in a header of the activity; and

means for retrieving the one or more actions from the storage locations, wherein causation of the performance of the first PHY-level task includes executing, by the SoC device, the one or more actions retrieved from the storage locations.

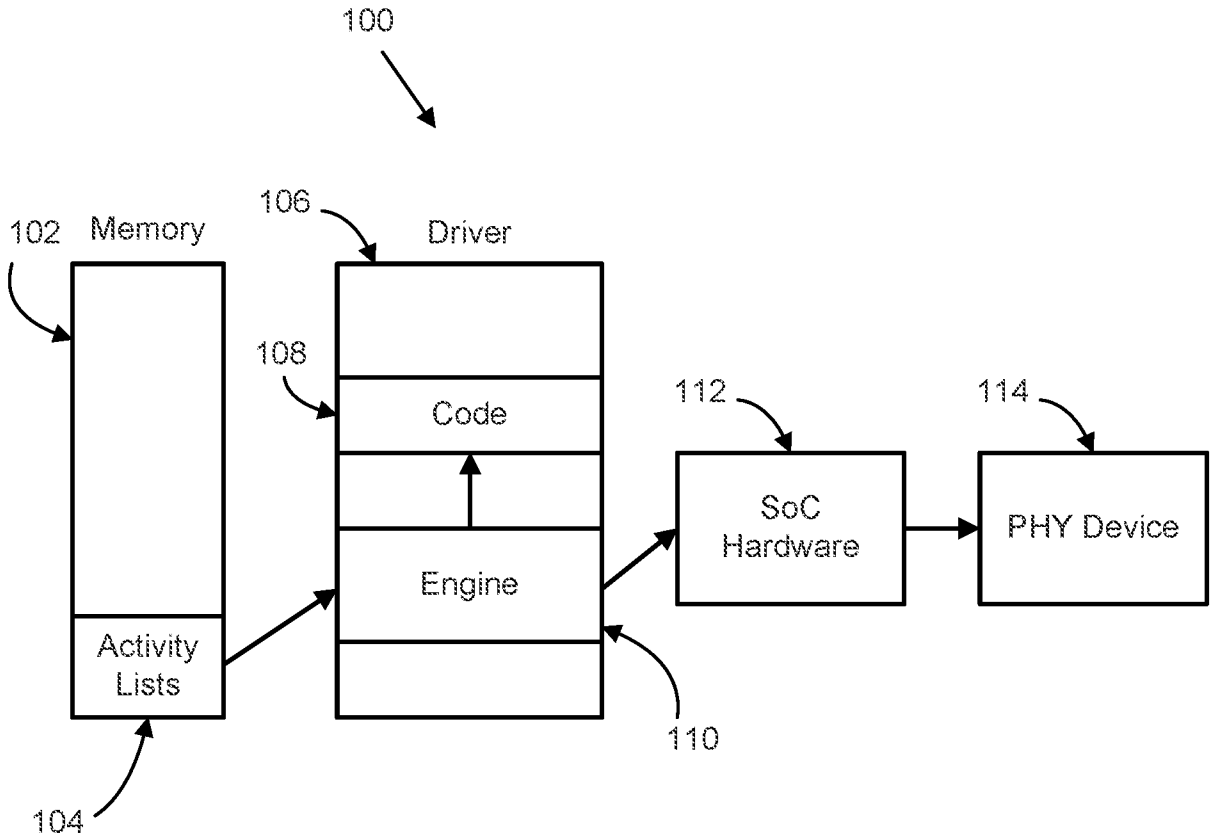

100
102
Memory
106
Driver
108
Code
112
SoC
Hardware
114
PHY Device
Engine
Activity
Lists
110
104

FIGURE 1

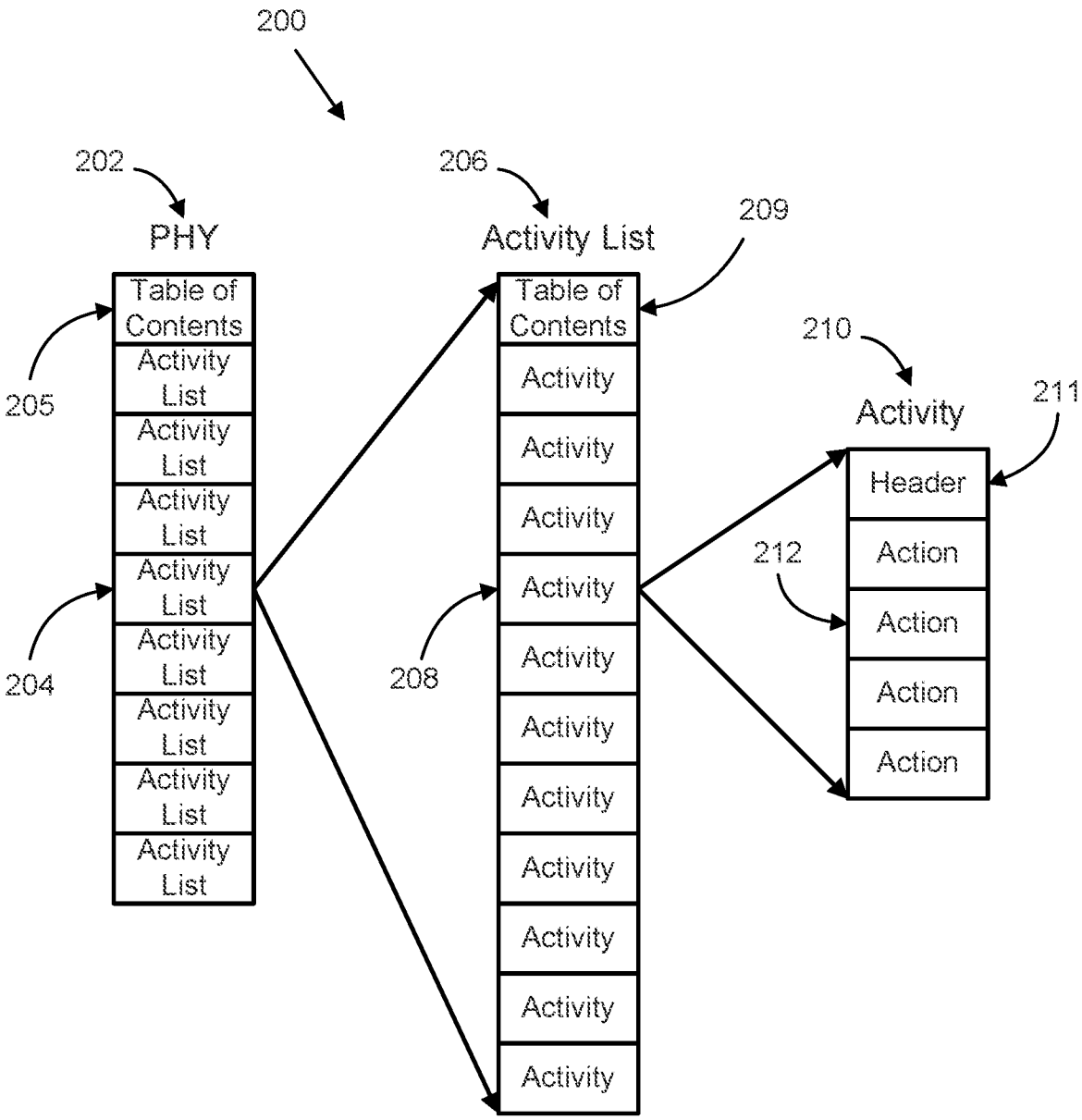

200
202
PHY
206
Activity List
209
Table of Contents
Activity List
Activity List
Activity List
Activity List
Activity List
Activity List
Activity List
205
204
Table of Contents
Activity
Activity
Activity
Activity
Activity
Activity
Activity
Activity
Activity
Activity
208
210
Activity
211
Header
Action
Action
Action
Action
212

FIGURE 2

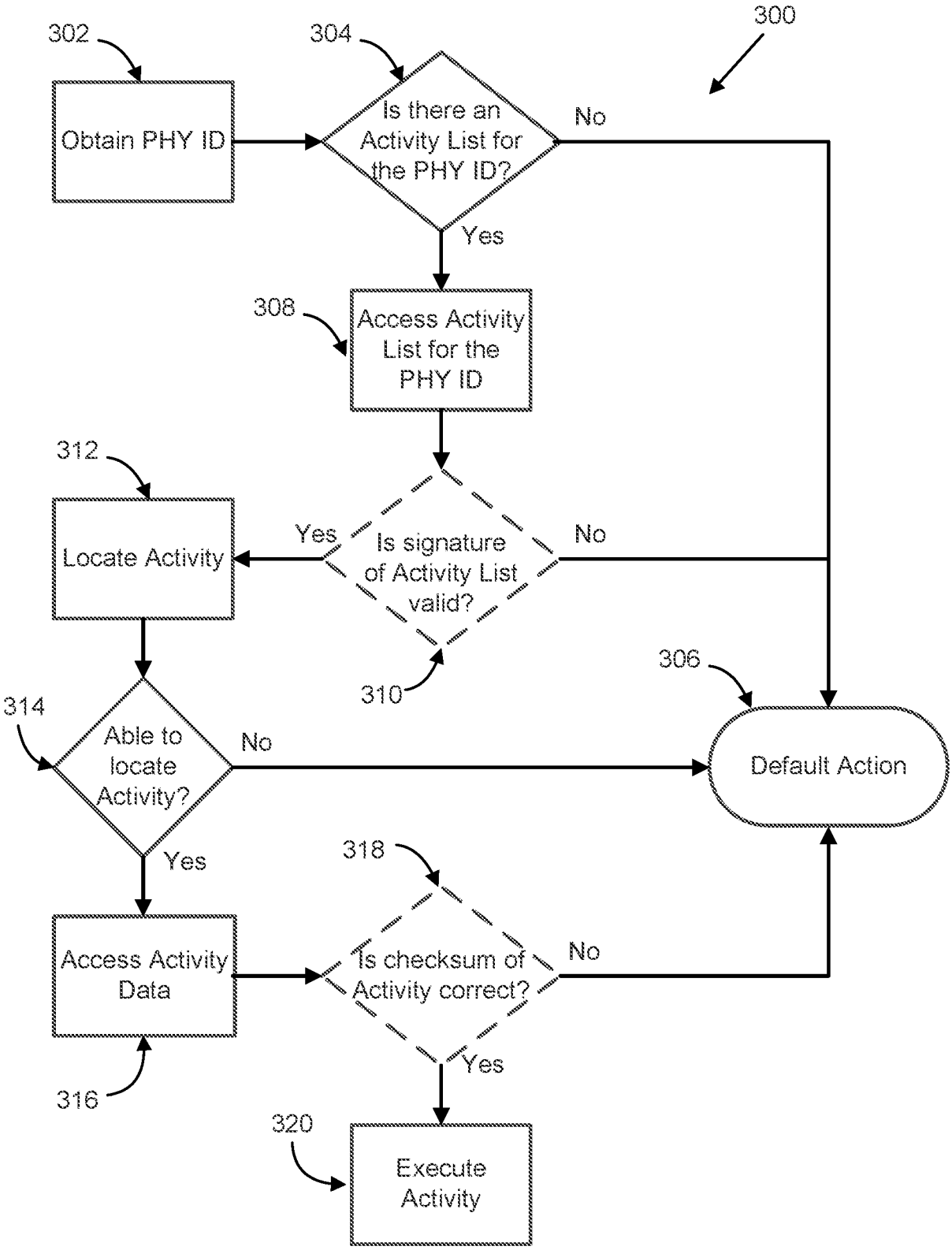

300
302
Obtain PHY ID
304
Is there an Activity List for the PHY ID?
No
Yes
308
Access Activity List for the PHY ID
312
Locate Activity
Yes
Is signature of Activity List valid?
No
310
306
Default Action
314
Able to locate Activity?
No
Yes
318
Access Activity Data
Is checksum of Activity correct?
No
Yes
316
320
Execute Activity

FIGURE 3

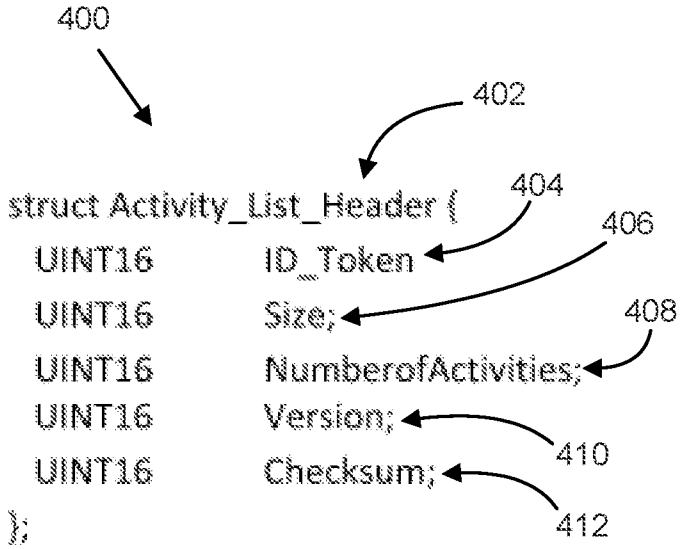


FIGURE 4

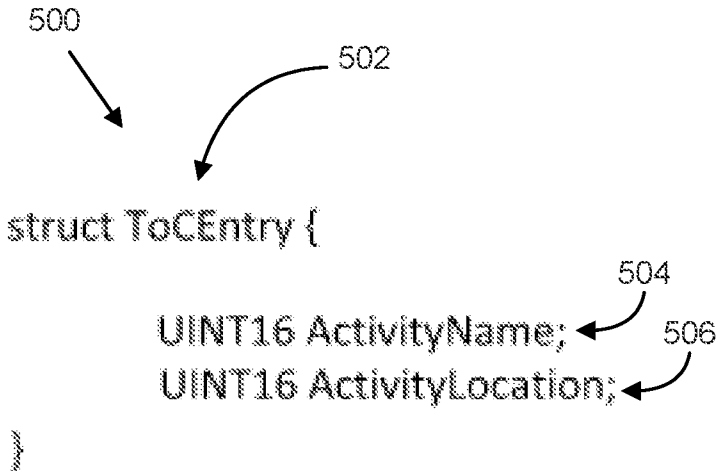


FIGURE 5

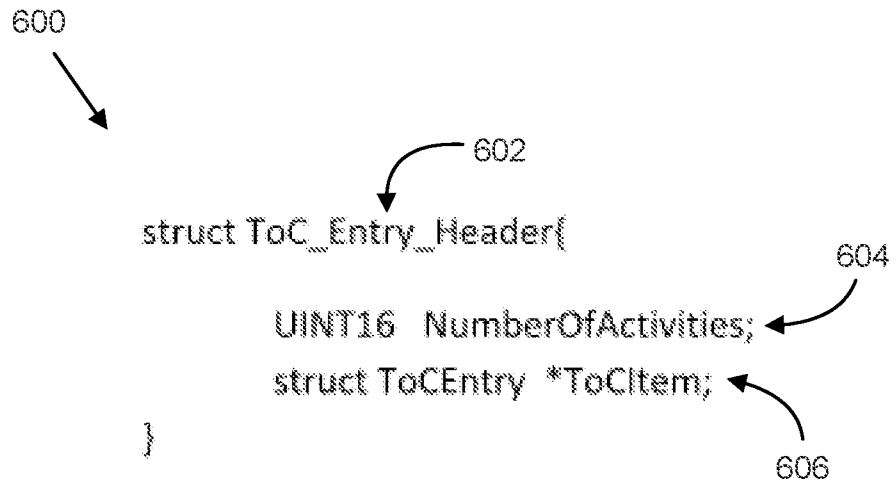


FIGURE 6

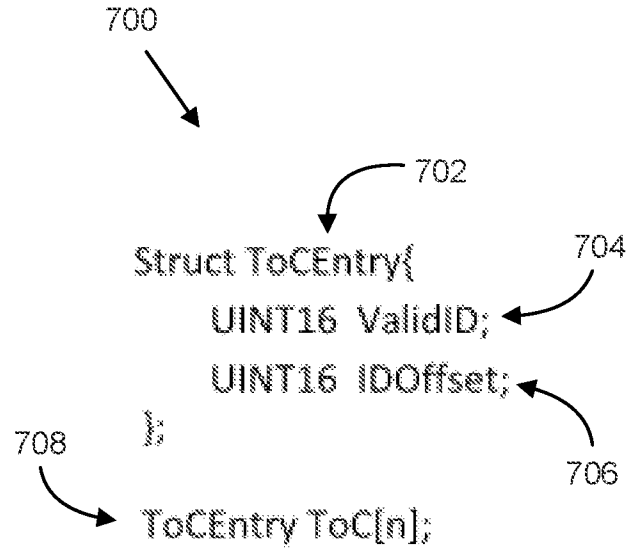


FIGURE 7

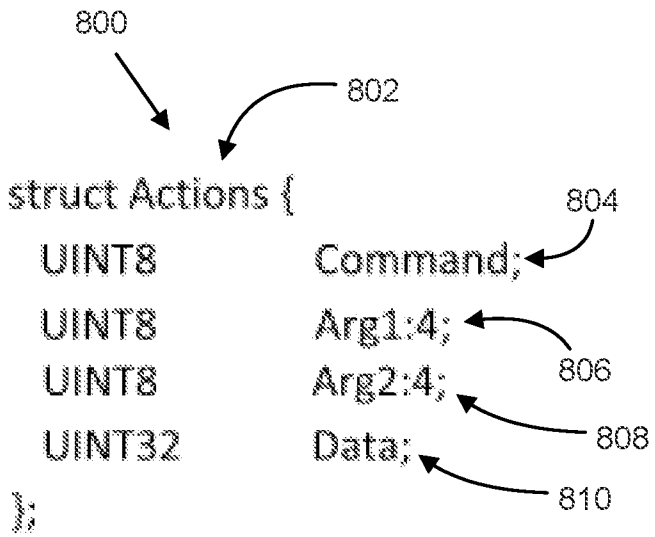


FIGURE 8

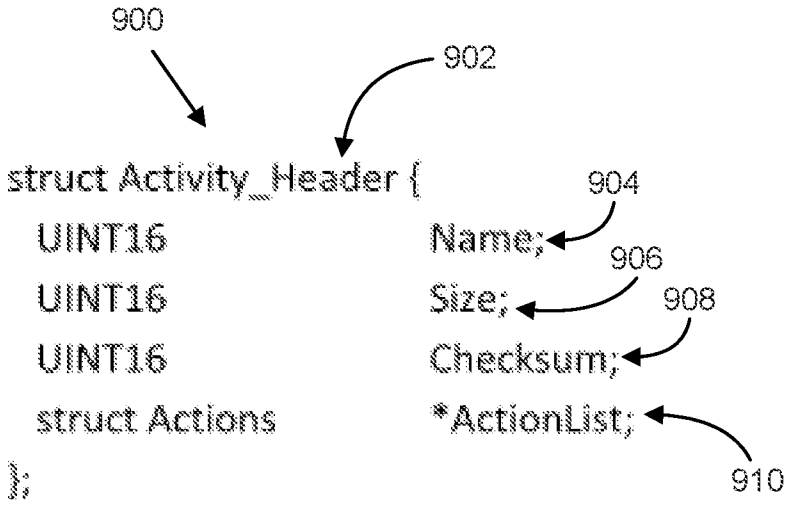


FIGURE 9

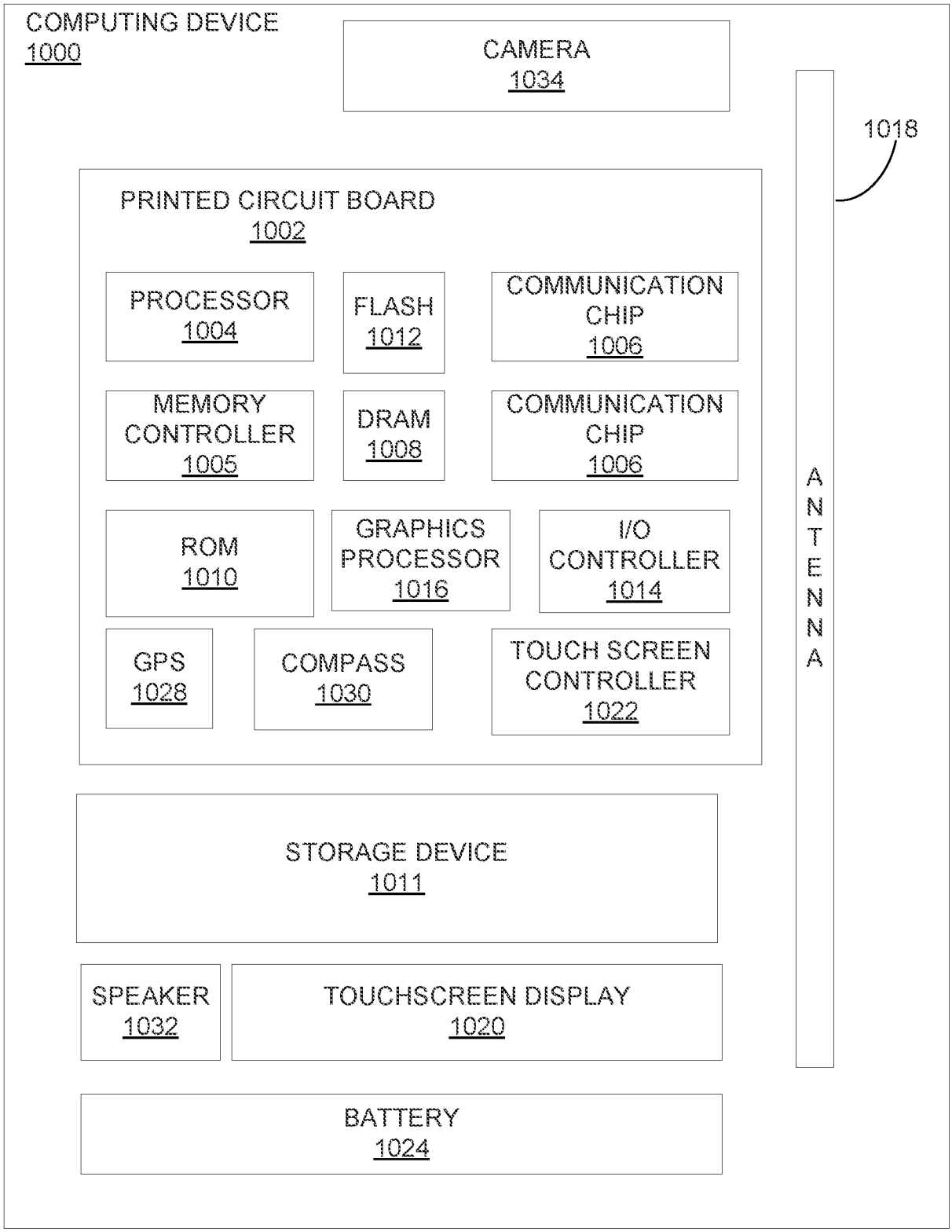

COMPUTING DEVICE
1000
CAMERA
1034
1018
PRINTED CIRCUIT BOARD
1002
PROCESSOR
1004
FLASH
1012
COMMUNICATION CHIP
1006
MEMORY CONTROLLER
1005
DRAM
1008
COMMUNICATION CHIP
1006
ROM
1010
GRAPHICS PROCESSOR
1016
I/O CONTROLLER
1014
GPS
1028
COMPASS
1030
TOUCH SCREEN CONTROLLER
1022
ANTENNA
STORAGE DEVICE
1011
SPEAKER
1032
TOUCHSCREEN DISPLAY
1020
BATTERY
1024

FIGURE 10

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2016/058660

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/30(2006.01)i, G06F 11/10(2006.01)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 9/30; H04L 29/02; G06F 15/16; G06F 12/00; G06F 13/28; H04L 12/56; G06F 11/10

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: physical layer device, action, table, communication, signature

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2014-158171 A1 (HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.) 02 October 2014 See paragraphs [0016]-[0060]; and claims 1, 3, 5, 9.	1-25
A	US 2012-0246400 A1 (SANDEEP BHADRA et al.) 27 September 2012 See paragraphs [0038]-[0041]; and figures 5-11.	1-25
A	US 2005-0111403 A1 (MARIAN RUDOLF et al.) 26 May 2005 See paragraphs [0036]-[0053]; and figures 4-6.	1-25
A	US 2006-0034269 A1 (JAMES LARRY TAYLOR et al.) 16 February 2006 See paragraphs [0015]-[0017]; and figures 1-3.	1-25
A	US 2008-0104264 A1 (VICKY P. DUERK et al.) 01 May 2008 See paragraphs [0040]-[0041].	1-25

☐ 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	Date of mailing of the international search report
24 January 2017 (24.01.2017)	**26 January 2017 (26.01.2017)**

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
CHIN, Sang Bum
Telephone No. +82-42-481-8398

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.
PCT/US2016/058660

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2014-158171 A1	02/10/2014	US 2016-0043949 A1	11/02/2016
US 2012-0246400 A1	27/09/2012	JP 05921666 B2	24/05/2016
		JP 2014-512121 A	19/05/2014
		US 8874876 B2	28/10/2014
		WO 2012-129432 A2	27/09/2012
		WO 2012-129432 A3	06/12/2012
US 2005-0111403 A1	26/05/2005	AR 046392 A1	07/12/2005
		AU 2004-301610 A1	03/02/2005
		AU 2004-301610 B2	20/09/2007
		AU 2004-301610 C1	15/05/2008
		BR PI0412170 A	22/08/2006
		CA 2532662 A1	03/02/2005
		CN 104955083 A	30/09/2015
		CN 104955084 A	30/09/2015
		DE 202004011065 U1	23/12/2004
		EP 1649709 A2	26/04/2006
		EP 1649709 A4	30/08/2006
		EP 1649709 B1	20/06/2012
		ES 2388603 T3	16/10/2012
		GE P20105122B	25/11/2010
		IL 172672 A	31/12/2013
		IL 172672 D0	10/04/2006
		JP 2007-532020 A	08/11/2007
		JP 2008-017533 A	24/01/2008
		JP 2011-155670 A	11/08/2011
		JP 2013-110762 A	06/06/2013
		JP 2014-171257 A	18/09/2014
		JP 2015-167418 A	24/09/2015
		JP 5379178 B2	25/12/2013
		JP 5669870 B2	18/02/2015
		JP 5832590 B2	16/12/2015
		KR 10-0684562 B1	12/03/2007
		KR 10-0967156 B1	30/06/2010
		KR 10-1204572 B1	26/11/2012
		KR 10-1204608 B1	23/11/2012
		KR 10-1357385 B1	03/02/2014
		KR 10-1475586 B1	22/12/2014
		KR 10-2005-0009236 A	24/01/2005
		KR 10-2005-0090354 A	13/09/2005
		KR 10-2006-0037356 A	03/05/2006
		KR 10-2009-0099592 A	22/09/2009
		KR 10-2012-0081250 A	18/07/2012
		KR 10-2013-0009886 A	23/01/2013
		KR 10-2013-0093154 A	21/08/2013
		KR 10-2014-0010993 A	27/01/2014
		KR 20-0364023 Y1	08/10/2004
		MX PA06000603 A	11/04/2006

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2016/058660

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		NO 20060746 A	18/04/2006
		NO 330909 B1	15/08/2011
		SG 129441 A1	26/02/2007
		TW 200824362 A	01/06/2008
		TW 201316726 A	16/04/2013
		TW 201421938 A	01/06/2014
		TW 259680 B	01/08/2006
		TW 261923 U	11/04/2005
		TW I259680 B	01/08/2006
		TW I390898B	21/03/2013
		TW I526022 B	11/03/2016
		TW M261923 U	11/04/2005
		US 2012-140756 A1	07/06/2012
		US 8121098 B2	21/02/2012
		WO 2005-011312 A2	03/02/2005
		WO 2005-011312 A3	31/03/2005
US 2006-0034269 A1	16/02/2006	US 7428242 B2	23/09/2008
US 2008-0104264 A1	01/05/2008	TW 200836072 A	01/09/2008
		TW I358026 B	11/02/2012
		US 8447872 B2	21/05/2013