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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
10 December 2015 (10.12.2015)

WIPO | PCT

(10) International Publication Number
WO 2015/187810 A1

(51) International Patent Classification:
G06F 13/16 (2006.01) *G11C 7/00* (2006.01)
G06F 3/06 (2006.01)

(21) International Application Number:
PCT/US2015/033953

(22) International Filing Date:
3 June 2015 (03.06.2015)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

62/007,136	3 June 2014 (03.06.2014)	US
14/728,343	2 June 2015 (02.06.2015)	US

(71) Applicant: **QUALCOMM INCORPORATED** [US/US]; ATTN: International IP Administration, 5775 Morehouse Drive, San Diego, California 92121-1714 (US).

(72) Inventors: **SHACHAM, Assaf**; 5775 Morehouse Drive, San Diego, California 92121 (US). **RAVIV, Dolev**; 5775 Morehouse Drive, San Diego, California 92121 (US). **TEB, David**; 5775 Morehouse Drive, San Diego, California 92121 (US).

(74) Agent: **DAVENPORT, Taylor, M.**; Withrow & Terranova, PLLC, 2530 Meridian Parkway, Suite 300, Durham, North Carolina 27713 (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, 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, 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,

[Continued on next page]

(54) Title: AN INPUT/OUTPUT VIRTUALIZATION (IOV) HOST CONTROLLER (HC) (IOV-HC) OF A FLASH-MEMORY-BASED STORAGE DEVICE

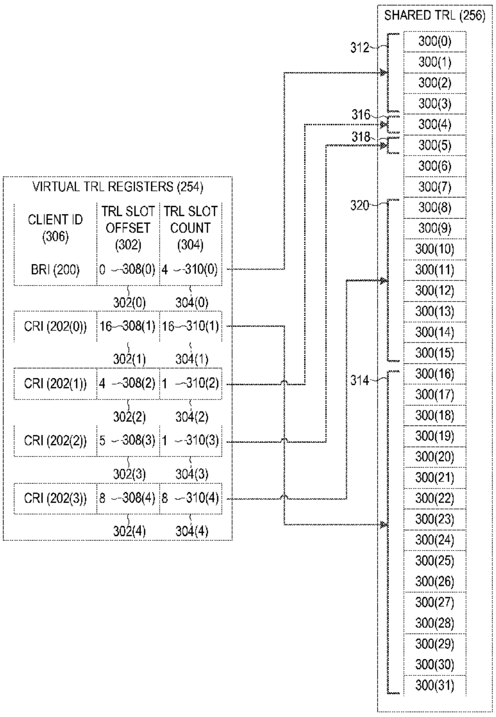


FIG. 3

(57) Abstract: An input/output virtualization (IOV) host controller (HC) (IOV-HC) of a flash-memory-based storage device is disclosed. In one aspect, an IOV-HC is coupled to input/output (I/O) clients via corresponding client register interfaces (CRIs), and is also coupled to a flash-memory-based storage device. The IOV-HC comprises transfer request list (TRL) slot offset registers indicating slots of a shared TRL that are assigned as base slots to each of the CRIs. The IOV-HC further comprises TRL slot count registers indicating how many slots of the shared TRL are assigned to each of the CRIs. When a transfer request (TR) directed to the flash-memory-based storage device is received from a CRI, the IOV-HC is configured to map the TR to a slot of the shared TRL based on a TRL slot offset register and a TRL slot count register of the plurality of TRL slot count registers corresponding to the CRI.

WO 2015/187810 A1

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))*

AN INPUT/OUTPUT VIRTUALIZATION (IOV) HOST CONTROLLER (HC) (IOV-HC) OF A FLASH-MEMORY-BASED STORAGE DEVICE

PRIORITY CLAIM

[0001] The present application claims priority to U.S. Provisional Patent Application Serial No. 62/007,136 filed on June 3, 2014 and entitled "MULTI-HOST UNIVERSAL FLASH STORAGE (UFS)," the contents of which is incorporated herein by reference in its entirety.

[0002] The present application also claims priority to U.S. Patent Application Serial No. 14/728,343 filed on June 2, 2015 and entitled "AN INPUT/OUTPUT VIRTUALIZATION (IOV) HOST CONTROLLER (HC) (IOV-HC) OF A FLASH-MEMORY-BASED STORAGE DEVICE," the contents of which is incorporated herein by reference in its entirety.

BACKGROUND

I. Field of the Disclosure

[0003] The technology of the disclosure relates generally to flash-memory-based storage in mobile computing devices.

II. Background

[0004] Flash memory is a non-volatile data storage medium to which data may be electronically written and erased. Flash memory is presently used in a variety of flash-memory-based storage devices, including memory cards, solid-state drives, and Universal Serial Bus (USB) flash drives. Flash-memory-based storage devices may offer fast read and write times comparable to dynamic Random Access Memory (RAM) while providing higher durability and shock resistance than conventional hard disks.

[0005] To facilitate the widespread use of flash-memory-based storage devices, a number of standards have been developed or are currently under development. One such standard is Universal Flash Storage (UFS), developed by the Joint Electron Device Engineering Council (JEDEC) for flash-memory-based storage in mobile computing devices, such as smart phones and tablet computers. UFS adopts the Small Computer System Interface (SCSI) Architecture Model and command protocols supporting multiple commands with command queuing features, thus enabling a multi-thread programming paradigm. Another standard developed by JEDEC is the Embedded MultiMediaCard (eMMC) standard, which offers a simplified application interface design, small package sizes, and low power consumption. eMMC flash-memory-based storage devices are presently one of the primary forms of storage in mobile devices.

[0006] Conventional flash-memory-based storage device standards, such as UFS and eMMC, are currently designed for management and usage by a single input/output (I/O) client. However, many modern computing devices are capable of supporting multiple I/O clients (e.g., hosts or other processor subsystems) simultaneously using virtualization environments. In such virtualization environments, multiple I/O clients may each need to interact with a single flash-memory-based storage device as if it were the only host of the flash-memory-based storage device. Similarly, the flash-memory-based storage device may need to operate as if it is communicating with only a single I/O client, when, in fact, it is communicating with multiple I/O clients.

[0007] In particular, a conventional host controller (HC) for a flash-memory-based storage device may provide a transfer request list (TRL) made up of multiple "slots" (also referred to under some standards as a task descriptor list (TDL), comprising multiple task descriptors (TDs)). The slots may be used by an I/O client for issuing transfer requests (TRs), such as read/write transactions, to the flash-memory-based storage device. However, to provide access to the flash-memory-based storage device in a multi-host environment, the HC may need to allow multiple I/O clients to access the TRL, as opposed to a single host.

SUMMARY OF THE DISCLOSURE

[0008] Aspects disclosed in the detailed description include an input/output virtualization (IOV) host controller (HC) (IOV-HC) of a flash-memory-based storage device. In this regard, in one aspect, an IOV-HC provides access to a flash-memory-based storage device to multiple input/output (I/O) clients within a single system on a system-on-chip (SoC). In particular, the IOV-HC provides storage access to multiple I/O clients by providing a shared transfer request list (TRL), which comprises physical "slots" for issuing transfer requests (TRs), such as read/write transactions, to the flash-memory-based storage device. The IOV-HC implements a number of client register interfaces (CRIs), each of which is provided with its own virtual TRL that is made up of a subset of the slots of the shared TRL. The virtual TRLs are defined by TRL slot offset registers and TRL slot count registers maintained by the IOV-HC. Using the virtual TRLs, the IOV-HC may efficiently process TRs from multiple I/O devices to the flash-memory-based storage device and responses provided by the flash-memory-based storage device to the multiple I/O clients, in a manner transparent to the multiple I/O clients.

[0009] In another aspect, an IOV-HC is provided. The IOV-HC is communicatively coupled to a plurality of input/output (I/O) clients via a corresponding plurality of client register interfaces (CRIs), and is also communicatively coupled to a flash-memory-based storage device. The IOV-HC comprises a plurality of transfer request list (TRL) slot offset registers, each indicating a slot of a shared TRL that is assigned as a base slot to each CRI of the plurality of CRIs. The IOV-HC further comprises a plurality of TRL slot count registers, each indicating a number of slots of the shared TRL assigned to each CRI of the plurality of CRIs. The IOV-HC is configured to receive a transfer request (TR) directed to the flash-memory-based storage device from a CRI of the plurality of CRIs. The IOV-HC is further configured to map, by a TR fetch circuit of the IOV-HC, the TR to a slot of the shared TRL based on a TRL slot offset register of the plurality of TRL slot offset registers and a TRL slot count register of the plurality of TRL slot count registers, the TRL slot offset register and the TRL slot count register corresponding to the CRI.

[0010] In another aspect, a method for providing virtual TRLs for multiple hosts is provided. The method comprises receiving, by an IOV-HC, a TR directed to a flash-memory-based storage device from a CRI of a plurality of CRIs. The method further comprises mapping, by a TR fetch circuit of the IOV-HC, the TR to a slot of a shared TRL based on a TRL slot offset register of a plurality of TRL slot offset registers and a TRL slot count register of a plurality of TRL slot count registers, the TRL slot offset register and the TRL slot count register corresponding to the CRI. The plurality of TRL slot offset registers each indicates the slot of the shared TRL that is assigned as a base slot to a corresponding CRI of the plurality of CRIs. The plurality of TRL slot count registers each indicates a number of slots of the shared TRL assigned to the corresponding CRI of the plurality of CRIs.

[0011] In another aspect, an IOV-HC is provided. The IOV-HC comprises a means for receiving a TR directed to a flash-memory-based storage device from a CRI of a plurality of CRIs. The IOV-HC further comprises a means for mapping the TR to a slot of a shared TRL based on a TRL slot offset register of a plurality of TRL slot offset registers and a TRL slot count register of a plurality of TRL slot count registers, the TRL slot offset register and the TRL slot count register corresponding to the CRI. The plurality of TRL slot offset registers each indicates the slot of the shared TRL that is assigned as a base slot to a corresponding CRI of the plurality of CRIs. The plurality of TRL slot count registers each indicates a number of slots of the shared TRL assigned to the corresponding CRI of the plurality of CRIs.

BRIEF DESCRIPTION OF THE FIGURES

[0012] Figure 1 is a block diagram of an exemplary flash-memory-based storage device and an exemplary system-on-a-chip (SoC) employing a plurality of hosts and an input/output virtualization (IOV) host controller (HC) (IOV-HC);

[0013] Figure 2 is a block diagram of an exemplary IOV-HC employing virtual transfer request list (TRL) registers;

[0014] Figure 3 is a block diagram illustrating exemplary allocation of slots in a shared TRL to multiple client register interfaces (CRIs) by the IOV-HC of Figure 2 using the virtual TRL registers;

[0015] Figures 4A-4B are block diagrams illustrating exemplary operations of the IOV-HC of Figure 2 for receiving a transfer request (TR), mapping the TR to a slot of a shared TRL using the virtual TRL registers of Figure 2, and returning a response;

[0016] Figure 5 is a block diagram illustrating exemplary allocation of slots in a shared task management request list (TMRL) to multiple CRIs by some aspects of the IOV-HC of Figure 2 using virtual TMRL registers;

[0017] Figure 6 is a block diagram illustrating operations of the IOV-HC of Figure 2 for receiving a task management request (TMR), mapping the TMR to a slot of a shared TMRL using the virtual TMRL registers of Figure 5, and returning a response;

[0018] Figures 7A and 7B are flowcharts illustrating exemplary operations of the IOV-HC of Figure 2 for allocating slots of a shared TRL using virtual TRL registers; and

[0019] Figure 8 is a block diagram of an exemplary computing device that may include an SoC that employs the virtual TRL registers of Figures 2 and 3 within the IOV-HC.

DETAILED DESCRIPTION

[0020] With reference now to the drawing figures, several exemplary aspects of the present disclosure are described. The word "exemplary" is used herein to mean "serving as an example, instance, or illustration." Any aspect described herein as "exemplary" is not necessarily to be construed as preferred or advantageous over other aspects.

[0021] Aspects disclosed in the detailed description include an input/output virtualization (IOV) host controller (HC) (IOV-HC) of a flash-memory-based storage device. In this regard, in one aspect, an IOV-HC provides access to a flash-memory-based storage device to multiple input/output (I/O) clients within a single system on a system-on-chip (SoC). In particular, the IOV-HC provides storage access to multiple I/O clients by providing a shared transfer request list (TRL), which comprises physical "slots" for issuing transfer requests (TRs), such as read/write transactions, to the flash-memory-based storage device. The IOV-HC implements a number of client register interfaces (CRIs), each of which is provided with its own virtual TRL that is made up of a subset of the slots of the shared TRL. The virtual TRLs are defined by TRL slot offset registers and TRL slot count registers maintained by the IOV-HC. Using the virtual TRLs, the IOV-HC may efficiently process TRs from multiple I/O devices to the flash-memory-based storage device and responses provided by the flash-memory-based storage device to the multiple I/O clients, in a manner transparent to the multiple I/O clients.

[0022] In this regard, Figure 1 is a block diagram of an exemplary system-on-a-chip (SoC) 100 including an IOV-HC 102. In the example of Figure 1, the IOV-HC 102 is provided to enable a plurality of I/O clients 104(0)-104(N) to access a single flash-memory-based storage device 106. In some aspects, the flash-memory-based storage device 106 may comprise a Universal Flash Storage (UFS) device or an Embedded MultiMediaCard (eMMC) device, as non-limiting examples. Notably, while the flash-memory-based storage device 106 is embedded adjacent to the SoC 100 within a mobile device in this aspect, other aspects may achieve similar functionality when employing the flash-memory-based storage device 106 as a removable memory device.

[0023] The SoC 100 also includes a virtual machine manager (VMM) 108, which provides virtualization functionality and management for the I/O clients 104(0)-104(N). The VMM 108 may be a software module that is executed by a processor (not shown) of the SoC 100, and that is resident in the system memory of the SoC 100 or other memory location as needed or desired. Each of the I/O clients 104(0)-104(N) may comprise a host software entity (such as the I/O client 104(0) and the I/O client 104(1) of Figure 1), which is executed on the processor of the SoC 100 and/or on a processor subsystem (not shown) of the SoC 100. The IOV-HC 102 may act as an intermediary between the I/O clients 104(0)-104(N) and the flash-memory-based storage device 106. In this manner, the IOV-HC 102 may control interactions between the I/O clients 104(0)-104(N) and the flash-memory-based storage device 106 such that the I/O clients 104(0)-104(N) each interact with the flash-memory-based storage device 106 as if it were the only I/O client 104(0)-104(N) communicating with the flash-memory-based storage device 106. Similarly, the IOV-HC 102 may be configured to enable the flash-memory-based storage device 106 to operate as if it is communicating with only a single I/O client 104(0)-104(N), when, in fact, it is communicating with multiple I/O clients 104(0)-104(N).

[0024] To better illustrate exemplary constituent elements of the IOV-HC 102 of Figure 1 according to some aspects, Figure 2 is provided. It is to be understood that the IOV-HC 102 may include more or fewer elements than illustrated in Figure 2, according to the needs of the flash-memory-based storage device standard (e.g., UFS or eMMC, as non-limiting examples) in use. As seen in Figure 2, the IOV-HC 102 includes a base register interface (BRI) 200 through which the VMM 108 of Figure 1 may communicate with the IOV-HC 102. The IOV-HC 102 also includes client register interfaces (CRIs) 202(0)-202(N). Each of the CRIs 202(0)-202(N) represents an interface for corresponding ones of the I/O clients 104(0)-104(N) of Figure 1 to transmit a TR (not shown) to the IOV-HC 102. The CRIs 202(0)-202(N) may also be used to transmit other, non-TR requests, such as queries or no-operation (NOP) instructions.

[0025] As seen in Figure 2, the CRIs 202(0)-202(N) are communicatively coupled to a TR fetch circuit (TR FETCH CIRC) 204 of the IOV-HC 102, as indicated by arrows 206, 208, and 210, respectively. In some aspects, the TR fetch circuit 204 may be responsible for fetching a transport protocol descriptor, such as a UFS Transport Protocol (UTP) Transfer Request Descriptor (UTRD) (not shown) from system memory, and decoding the transport protocol descriptor to determine a location of a command descriptor such as a UTP command descriptor (UCD) (not shown). The TR fetch circuit 204 may then fetch a TR section (not shown) of the command descriptor from a system memory 212, as indicated by arrow 214.

[0026] In a conventional host controller, the TR fetch circuit 204 would then extract parameters from the TR, and forward the entire TR to a transport protocol engine 216 (e.g., a UTP engine or an eMMC transport protocol engine) for processing. However, in a multi-client environment, additional inspection and processing of the TR may be required. For instance, it may be desirable to provide custom handling for TRs of different types, or to trap or terminate a TR. Accordingly, the IOV-HC 102 is configured to provide command trapping functionality. The IOV-HC 102 thus includes a TR filter circuit 218 for TR inspection and processing, a TR trap 220 for command trapping, and a response generation circuit 222 for facilitating the transmission of responses resulting from trapped commands to the I/O clients 104(0)-104(N) (not shown). Operations of the TR filter circuit 218, the TR trap 220, and the response generation circuit 222 are discussed in greater detail below.

[0027] With continuing reference to Figure 2, inspection and processing of every TR fetched by the TR fetch circuit 204 is carried out by the TR filter circuit 218 of the IOV-HC 102. The TR filter circuit 218 determines how a TR is to be processed based on its contents. If the contents of the TR indicate that the TR corresponds to a read or write command, the TR may be forwarded to an access control circuit 224 by the TR filter circuit 218, as indicated by arrow 226. The access control circuit 224 provides hardware enforcement of read/write access rules to logical units (not shown). If the TR is validated by the access control circuit 224, the TR is passed on to the transport protocol engine 216, as indicated by arrow 228. TRs that are denied by the access control circuit 224 are trapped for software processing, as indicated by arrow 230. If the contents of the TR indicate that the TR corresponds to a power mode change command, the TR is sent to a multi-host power controller (MHPC) 232 by the TR filter circuit 218, as indicated by arrow 234.

[0028] The MHPC 232 of Figure 2 is a hardware state machine that controls the power modes of the flash-memory-based storage device 106 based on power mode change requests (not shown) from the I/O clients 104(0)-104(N). If the TR is successfully processed by the MHPC 232, the TR may be passed to the transport protocol engine 216, as indicated by arrow 236. The transport protocol engine 216 then communicates power mode change commands and responses with the flash-memory-based storage device 106 via an interconnect 238 and a bus 240. Otherwise, the MHPC 232 may silently discard the TR, or may trap the TR for software processing, as indicated by arrow 242. The MHPC 232 may also generate an error interrupt (not shown) upon notification by the flash-memory-based storage device 106 that a requested power mode change was unsuccessful. If the TR includes other commands, the TR may be intercepted and trapped by the TR filter circuit 218, as indicated by arrow 246.

[0029] As seen in Figure 2, the TR trap 220 is provided by the IOV-HC 102 to store trapped TRs. Within the TR trap 220 are command (CMD) mirrors 248(0)-248(N), each associated with one of the CRIs 202(0)-202(N). A trapped TR is stored in a CMD mirror 248(0)-248(N) of a CRI 202(0)-202(N) from which the TR was received. An interrupt (represented by arrows 250 and 252) is generated to the VMM 108 via the BRI 200 to indicate that the TR is trapped. The VMM 108 may then further process the TR by reading the contents of the appropriate CMD mirror 248(0)-248(N) to extract information, and generating a response to the originating CRI 202(0)-202(N).

[0030] Once a TR is trapped, the TR fetch circuit 204 has completed its processing, and may proceed with fetching the next TR for another CRI 202(0)-202(N). Until the trapped TR is processed by the VMM 108, the CRI 202(0)-202(N) associated with the trapped TR is removed from TR fetch arbitration (not shown). Only when the VMM 108 processes the TR by ordering the IOV-HC 102 to forward or discard the TR may subsequent TRs from the same CRI 202(0)-202(N) be fetched by the TR fetch circuit 204.

[0031] While the aspects described above with respect to Figures 1 and 2 allow many functions to be enabled for use of the flash-memory-based storage device 106 by multiple hosts (e.g., the I/O clients 104(0)-104(N)), further issues remain with respect to processing TRs by the IOV-HC 102. In particular, a conventional host controller (HC) may provide a TRL (not shown) comprising a number of "slots" (in some aspects, up to thirty-two (32) slots, according to the UFS-HCI (UFS Host Controller Interface) standard and the eMMC standard). The TRL slots are used by software processes for issuing TRs to and receiving responses from the flash-memory-based storage device 106. However, in a multiple I/O client environment such as the SoC 100 of Figure 1, the IOV-HC 102 needs to be able to allocate and track slots among the multiple I/O clients 104(0)-104(N). In some aspects, the same issue may arise with respect to allocation of slots of a task management request list (TMRL) (not shown).

[0032] In this regard, the IOV-HC 102 provides virtual TRL registers 254 for allocating slots of a shared TRL 256 as virtual TRLs (not shown) to the CRIs 202(0)-202(N) corresponding to the I/O clients 104(0)-104(N). Some aspects may also provide virtual TMRL registers 258 for allocating slots of a shared TMRL 260 as virtual TMRLs (not shown) to the CRIs 202(0)-202(N) corresponding to the I/O clients 104(0)-104(N). Use of the virtual TRL registers 254 is described in greater detail below with respect to Figures 3, 4A, and 4B, while use of the virtual TMRL registers 258 is described below with respect to Figures 5 and 6.

[0033] Figure 3 provides a more detailed illustration of the shared TRL 256 and the virtual TRL registers 254 of Figure 2. In the example of Figure 3, the shared TRL 256 includes thirty-two (32) slots 300(0)-300(31). It is to be understood that in some aspects, the shared TRL 256 may comprise a number of slots greater than the thirty-two (32) slots 300(0)-300(31) illustrated here. The virtual TRL registers 254 of Figure 3 include a plurality of TRL slot offset registers 302(0)-302(4) and a plurality of TRL slot count registers 304(0)-304(4). Each of the TRL slot offset registers 302(1)-302(4) and the TRL slot count registers 304(1)-304(4) correspond to a CRI 202(0)-202(3), as indicated by a Client ID column 306. For purposes of illustration, the TRL slot offset register 302(0) and the TRL slot count register 304(0) are associated with the BRI 200. In some aspects, the virtual TRL registers 254 may be accessible by the VMM 108 via the BRI 200 of Figure 2. In particular, the size of the shared TRL 256 may be advertised to the VMM 108 via the BRI 200.

[0034] As seen in Figure 3, portions of the shared TRL 256 may be allocated to the BRI 200 and each of the CRIs 202(0)-202(3) by the IOV-HC 102 using the corresponding TRL slot offset registers 302(0)-302(4) and the TRL slot count registers

304(0)-304(4). TRL slot offset register values 308(0)-308(4) of the TRL slot offset registers 302(0)-302(4) each represent an index of a base slot (i.e., a first assigned slot) to the BRI 200 and the CRIs 202(0)-202(3), respectively. Thus, the BRI 200 is assigned slot 300(0) as a base slot, as indicated by the TRL slot offset register value 308(0), while the CRI 202(0) is assigned slot 300(16) as a base slot, as indicated by the TRL slot offset register value 308(1). The CRIs 202(1)-202(3) are assigned slots 300(4), 300(5), and 300(8), respectively, as indicated by the respective TRL slot offset register values 308(2)-308(4). According to some aspects, the TRL slot offset register values 308(0)-308(4) each are an integral power of two (e.g., 1, 2, 4, 8, etc.), and the settings for a particular CRI 202(0)-202(3) must be configured before the corresponding I/O client 104(0)-104(3) is enabled and should not be changed as long as the CRI 202(0)-202(3) is active.

[0035] With continuing reference to Figure 3, the TRL slot count registers 304(0)-304(4) store corresponding TRL slot count register values 310(0)-310(4). The TRL slot count register values 310(0)-310(4) each indicate a number of the slots 300(0)-300(31) of the shared TRL 256 that are assigned to the BRI 200 and the CRIs 202(0)-202(3), respectively. As seen in Figure 3, the BRI 200 is assigned 4 slots, as indicated by the TRL slot count register value 310(0). The BRI 200 is thus allocated a virtual TRL 312 made up of slots 300(0)-300(3) within the shared TRL 256. Similarly, the CRIs 202(0)-202(3) are allocated 16, 1, 1, and 8 slots, respectively, as indicated by corresponding TRL slot count register values 310(1)-310(4). The CRIs 202(0)-202(3) are thus allocated respective virtual TRLs 314, 316, 318, and 320, made up of corresponding slots 300(16)-300(31), 300(4), 300(5), and 300(8)-300(15). In an exemplary aspect, each of the TRL slot count register values 310(0)-310(4) is aligned to the corresponding TRL slot offset register values 308(0)-308(4), and the settings for a particular CRI 202(0)-202(3) must be configured before the corresponding I/O client 104(0)-104(3) is enabled and should not be changed as long as the CRI 202(0)-202(3) is active. These requirements are also optional.

[0036] In some aspects, upon initialization of the IOV-HC 102, the IOV-HC 102 may receive the TRL slot offset register values 308(0)-308(4) and the TRL slot count register values 310(0)-308(4) from the VMM 108, which may be responsible for allocation of the slots 300(0)-300(31). Some aspects may provide that the slots 300(0)-

300(31) are allocated based on a static configuration (not shown) or on an outcome of a heuristic algorithm (not shown). According to some aspects, the IOV-HC 102 may receive a TRL slot offset register value, such as the TRL slot offset register value 308(4), and a TRL slot count register value, such as the TRL slot count register value 310(4), from the VMM 108 when a new CRI, such as the CRI 202(3), is created by the VMM 108. Some aspects may provide that the VMM 108 controls the parameters of the virtual TRLs 312, 314, 316, 318, 320 presented to the I/O clients 104(0)-104(3). In some aspects, the number and location of allocated slots 300(0)-300(31) may be written in a capability register (not shown) in the IOV-HC 102, and, in particular, in a register accessed by the VMM 108 through the BRI 200.

[0037] In conventional controllers based on UFS-HCI, capability fields, implemented as hardware constants and referred to as NUTRS and/or NUTMRS, are used to indicate a size of a TRL and/or a TMRL, respectively. In some aspects of the present disclosure, these fields may be incorporated into each of the CRIs 202(0)-202(N). In such aspects, the VMM 108 may write to these fields upon initialization or virtual machine creation to notify each of the I/O clients 104(0)-104(3) how many of the slots 300(0)-300(31) are allocated.

[0038] To illustrate exemplary operations of the IOV-HC 102 of Figure 2 for receiving a TR, mapping the TR to a slot of the shared TRL 256 using the virtual TRL registers 254 of Figures 2 and 3, and returning a response, Figures 4A and 4B are provided. For the sake of clarity, elements of Figures 1-3 are referenced in describing Figures 4A and 4B. In Figure 4A, the IOV-HC 102 receives a TR 400 from the CRI 202(3). The TR 400 includes a slot identifier 402, which in this example has a value of one (1) and represents an index to the second slot in the virtual TRL 320 of Figure 3 allocated to the CRI 202(3). The IOV-HC 102 (in particular, the TR fetch circuit 204 of the IOV-HC 102) then maps the TR 400 to the slot 300(9) of the shared TRL 256 using the TRL slot offset register 302(4) and the TRL slot count register 304(4) corresponding to the CRI 202(3). For example, mapping may be accomplished by summing the slot identifier 402 with the TRL slot offset register value 308(4) corresponding to the CRI 202(3). As seen in Figure 3, the TRL slot offset register value 308(4) is eight (8), which, when summed with the slot identifier 402 having a value of one (1), results in a slot index of nine (9). The recalculated slot index is then encoded by the IOV-HC 102 within the TR 400 to be transmitted to the flash-memory-based storage device 106. As non-limiting examples, in aspects based on UFS, the recalculated slot index may be encoded in a Task Tag field of the TR 400, while in aspects based on eMMC, the recalculated slot index may be encoded in a Task ID field of the TR 400. The IOV-HC 102 thus directs the TR 400 to the slot 300(9) of the shared TRL 256, and from there the TR 400 is processed serially and passed to the flash-memory-based storage device 106. Some aspects may provide that the TRL slot count register 304(4) is used for bounds-checking the slot identifier 402 to ensure that the TR 400 is directed to a valid slot 300(0)-300(31) of the virtual TRL 320. In such aspects, if the TR 400 is directed to a slot 300(0)-300(31) outside the range assigned to the corresponding CRI 202(0)-202(3), the TR 400 is trapped.

[0039] In Figure 4B, the IOV-HC 102 receives a response 404 from the flash-memory-based storage device 106. The response 404 is directed to the same slot 300(9) from which the TR 400 was sent. Accordingly, the IOV-HC 102 routes the response 404 to the CRI 202(3) using the TRL slot offset register 302(4) and the TRL slot count register 304(4) corresponding to the CRI 202(3). As a non-limiting example, the IOV-HC 102 may subtract the TRL slot offset register value 308(4) (i.e., eight (8)) corresponding to the CRI 202(3) from a slot identifier 406 (nine (9)) of the response 404. As non-limiting examples, in aspects based on UFS, the slot identifier 406 may be retrieved from the Task Tag field of the response 404, while in aspects based on eMMC, the slot identifier 406 may be retrieved from the Task ID field of the response 404. The response 404 is then provided to the CRI 202(3) with a slot identifier 408 of one (1).

[0040] Some aspects of the present disclosure may further provide that the allocation of slots in a TMRL for multiple I/O clients 104(0)-104(N) is managed by the IOV-HC 102 in a manner similar to the allocation of the slots 300(0)-300(31) in the shared TRL 256 as described above. Accordingly, Figure 5 is provided to illustrate exemplary aspects of the virtual TMRL registers 258 and the shared TMRL 260 of Figure 2. In the example of Figure 5, the shared TMRL 260 includes eight (8) slots 500(0)-500(7), although some aspects may provide more or fewer slots than illustrated here. The virtual TMRL registers 258 of Figure 5 provide a plurality of TMRL slot offset registers 502(0)-502(4) and a plurality of TMRL slot count registers 504(0)-504(4). Each of the TMRL slot offset registers 502(1)-502(4) and the TMRL slot count registers 504(1)-504(4) correspond to a CRI 202(0)-202(3), as indicated by a Client ID column 506. For purposes of illustration, the TMRL slot offset register 502(0) and the TMRL slot count register 504(0) are associated with the BRI 200.

[0041] As seen in Figure 5, the IOV-HC 102 may allocate portions of the shared TMRL 260 to the BRI 200 and each of the CRIs 202(0)-202(3) using the corresponding TMRL slot offset registers 502(0)-502(4) and the TMRL slot count registers 504(0)-504(4). TMRL slot offset register values 508(0)-508(4) of the TMRL slot offset registers 502(0)-502(4) each represent an index of a base slot (i.e., a first assigned slot) to the BRI 200 and the CRIs 202(0)-202(3), respectively. The BRI 200 thus is assigned slot 500(0) as a base slot, as indicated by the TMRL slot offset register value 508(0), while CRI 202(0) is assigned slot 500(3) as a base slot, as indicated by the TMRL slot offset register value 508(1). CRIs 202(1)-202(3) are assigned slots 500(1), 500(2), and 500(5), respectively, as indicated by the respective TMRL slot offset register values 508(2)-508(4). According to some aspects, the TMRL slot offset register values 508(0)-508(4) each may be an integral power of two (e.g., 1, 2, 4, 8, etc.), and the settings for a particular CRI 202(0)-202(3) must be configured before the corresponding I/O client 104(0)-104(3) is enabled and should not be changed as long as the CRI 202(0)-202(3) is active.

[0042] Continuing to refer to Figure 5, the TMRL slot count registers 504(0)-504(4) store corresponding TMRL slot count register values 510(0)-510(4). The TMRL slot count register values 510(0)-510(4) each indicate a number of the slots 500(0)-500(31) of the shared TMRL 260 that are assigned to the BRI 200 and the CRIs 202(0)-202(3), respectively. As seen in Figure 5, the BRI 200 is assigned one (1) slots, as indicated by the TMRL slot count register value 510(0). The BRI 200 is thus allocated a virtual TMRL 512 made up of slot 500(0) within the shared TMRL 260. Similarly, CRIs 202(0)-202(3) are allocated 2, 1, 1, and 3 slots, respectively, as indicated by corresponding TMRL slot count register values 510(1)-510(4). CRIs 202(0)-202(3) are thus allocated respective virtual TMRLs 514, 516, 518, and 520, made up of slots 500(6)-500(7), 500(1), 500(2), and 500(3)-500(4) respectively. In an exemplary aspect, each of the TMRL slot count register values 510(0)-510(4) should be aligned to the corresponding TMRL slot offset register values 508(0)-508(4), and the settings for a particular CRI 202(0)-202(3) must be configured before the corresponding I/O client

104(0)-104(3) is enabled and should not be changed as long as the CRI 202(0)-202(3) is active. These requirements are also optional.

[0043] To illustrate exemplary operations of the IOV-HC 102 of Figure 2 for receiving a TMR and mapping the TMR to a slot of the shared TMRL 260 using the virtual TMRL registers 258 of Figures 2 and 5, Figure 6 is provided. For the sake of clarity, elements of Figures 1, 2, and 5 are referenced in describing Figure 6. In Figure 6, the IOV-HC 102 receives a TMR 600 from the CRI 202(3). The TMR 600 includes a slot identifier 602, which in this example has a value of one (1) and represents an index to the second slot in the virtual TMRL 520 of Figure 5 allocated to the CRI 202(3). The IOV-HC 102 (in particular, the TR fetch circuit 204 of the IOV-HC 102) then maps the TMR 600 to the slot 500(6) of the shared TMRL 260 using the TMRL slot offset register 502(4) and the TMRL slot count register 504(4) corresponding to the CRI 202(3). For example, mapping may be accomplished by summing the slot identifier 602 with the TMRL slot offset register value 508(4) corresponding to the CRI 202(3). As seen in Figure 5, the TMRL slot offset register value 508(4) is five (5), which, when summed with the slot identifier 602 having a value of one (1), results in a slot index of six(6). The recalculated slot index is then encoded by the IOV-HC 102 within the TMR 600 to be transmitted to the flash-memory-based storage device 106. As non-limiting examples, in aspects based on UFS, the recalculated slot index may be encoded in a Task Tag field of the TMR 600, while in aspects based on eMMC, the recalculated slot index may be encoded in a Task ID field of the TMR 600. The IOV-HC 102 thus directs the TMR 600 to the slot 500(6) of the shared TMRL 260, and from there the TMR 600 is processed serially and passed to the flash-memory-based storage device 106. In some aspects, the TMRL slot count register 504(4) may be used for bounds-checking the slot identifier 602 to ensure that the TMR 600 is directed to a valid slot 500(1)-500(31) of the virtual TMRL 520.

[0044] Figures 7A and 7B are provided to illustrate exemplary operations of the IOV-HC 102 of Figure 1 for allocating slots of the shared TRL 256 using the virtual TRL registers 254 of Figure 2. Elements of Figures 1-6 are referenced in describing Figures 7A and 7B for the sake of clarity. In Figure 7A, operations according to some aspects may begin with the IOV-HC 102 receiving, from the VMM 108, a TRL slot offset register value (such as the TRL slot offset register value 308(1)) and a TRL slot count register value (such as the TRL slot count register value 310(1)) for each CRI 202(1) of the plurality of CRIs 202(0)-202(N), responsive to initialization of the IOV-HC 102 (block 700). The IOV-HC 102 thus may be referred to herein as a "means for receiving, from a VMM, a TRL slot offset register value and a TRL slot count register value for each CRI of the plurality of CRIs, responsive to initialization of the IOV-HC." The IOV-HC 102 then receives a TR 400 directed to the flash-memory-based storage device 106 from the CRI 202(3) of the plurality of CRIs 202(0)-202(N) (block 702). In this regard, the IOV-HC 102 may be referred to herein as a "means for receiving a TR directed to a flash-memory-based storage device from a CRI."

[0045] The IOV-HC 102 (and in particular, the TR fetch circuit 204 of the IOV-HC 102) next maps the TR 400 to a slot 300(9) of a shared TRL 256 based on a TRL slot offset register 302(4) of a plurality of TRL slot offset registers 302(0)-302(4) and a TRL slot count register 304(4) of a plurality of TRL slot count registers 304(0)-304(4) (block 704). Accordingly, the TR fetch circuit 204 may be referred to herein as a "means for mapping the TR to a slot of a shared TRL based on a TRL slot offset register of a plurality of TRL slot offset registers and a TRL slot count register of a plurality of TRL slot count registers." The TRL slot offset register 302(4) and the TRL slot count register 304(4) both correspond to the CRI 202(3). In some aspects, the operations of block 704 for mapping the TR 400 to the slot 300(9) may be based on a sum of a slot identifier 402 of the TR 400 and a TRL slot offset register value 308(4) of the TRL slot offset register 302(4) (block 706). Processing then continues at block 708 of Figure 7B.

[0046] Referring now to Figure 7B, the IOV-HC 102 according to some aspects may receive, from the flash-memory-based storage device 106, a response 404 to the TR 400 (block 708). The IOV-HC 102 thus may be referred to herein as a "means for receiving, from the flash-memory-based storage device, a response to the TR." Based on the TRL slot offset register 302(4) and the TRL slot count register 304(4) corresponding to the CRI 202(3), the IOV-HC 102 may route the response 404 to the CRI 202(3) (block 710). In this regard, the IOV-HC 102 may be referred to herein as a "means for routing the response to the CRI based on the TRL slot offset register and the TRL slot count register corresponding to the CRI." In some aspects, the IOV-HC 102 may receive, from the VMM 108, a TRL slot offset register value (such as the TRL slot offset register value 308(2)) and a TRL slot count register value (such as the TRL slot count register value 310(2)) for a new CRI 202(1) corresponding to a new I/O client 104(1), responsive to creation of the new I/O client 104(1) by the VMM 108 (block 712). Accordingly, the IOV-HC 102 may be referred to herein as a "means for receiving, from a VMM, a TRL slot offset register value and a TRL slot count register value for a new CRI corresponding to a I/O client."

[0047] As noted above, in some aspects, allocation of slots 500(0)-500(31) for task management for multiple I/O clients 104(0)-104(N) may be managed by the IOV-HC 102 in a manner similar to the allocation of slots 300(0)-300(31) in the shared TRL 256 as described above. Accordingly, in such aspects, the IOV-HC 102 may receive a TMR 600 from a CRI 202(3) of the plurality of CRIs 202(0)-202(N) (block 714). The IOV-HC 102 may thus be referred to herein as a "means for receiving a TMR from a CRI of the plurality of CRIs." The IOV-HC 102 (in particular, the TR fetch circuit 204 of the IOV-HC 102) may then map the TMR 600 to a slot 500(9) of a shared TMRL 260 based on a TMRL slot offset register 502(4) of a plurality of TMRL slot offset registers 502(0)-502(4) and a TMRL slot count register 504(4) of a plurality of TMRL slot count registers 504(0)-504(4), the TMRL slot offset register 502(4) and the TMRL slot count register 504(4) corresponding to the CRI 202(3) (block 716). In this regard, the TR fetch circuit 204 may be referred to herein as a "means for mapping the TMR to a slot of a shared TMRL based on a TMRL slot offset register of a plurality of TMRL slot offset registers and a TMRL slot count register of a plurality of TMRL slot count registers."

[0048] The IOV-HC of the flash-memory-based storage device according to aspects disclosed herein may be provided in or integrated into any processor-based device. Examples, without limitation, include a set top box, an entertainment unit, a navigation device, a communications device, a fixed location data unit, a mobile location data unit, a mobile phone, a cellular phone, a computer, a portable computer, a desktop computer, a personal digital assistant (PDA), a monitor, a computer monitor, a television, a tuner, a radio, a satellite radio, a music player, a digital music player, a portable music player, a digital video player, a video player, a digital video disc (DVD) player, and a portable digital video player.

[0049] In this regard, Figure 8 illustrates an example of a processor-based system 800 that can employ the SoC 100 of Figure 2 with the IOV-HC 102 illustrated in Figure

2. In this example, the processor-based system 800 includes one or more central processing units (CPUs) 802, each including one or more processors 804. The CPU(s) 802 may have cache memory 806 coupled to the processor(s) 804 for rapid access to temporarily stored data. The CPU(s) 802 is coupled to a system bus 808 and can intercouple devices included in the processor-based system 800. As is well known, the CPU(s) 802 communicates with these other devices by exchanging address, control, and data information over the system bus 808. For example, the CPU(s) 802 can communicate bus transaction requests to a memory controller 810 as an example of a slave device. Although not illustrated in Figure 8, multiple system buses 808 could be provided.

[0050] Other devices can be connected to the system bus 808. As illustrated in Figure 8, these devices can include a memory system 812, one or more input devices 814, one or more output devices 816, one or more network interface devices 818, and one or more display controllers 820, as examples. The input device(s) 814 can include any type of input device, including but not limited to input keys, switches, voice processors, etc. The output device(s) 816 can include any type of output device, including but not limited to audio, video, other visual indicators, etc. The network interface device(s) 818 can be any devices configured to allow exchange of data to and from a network 822. The network 822 can be any type of network, including but not limited to a wired or wireless network, a private or public network, a local area network (LAN), a wide local area network, wireless local area network, BLUETOOTH (BT), and the Internet. The network interface device(s) 818 can be configured to support any type of communications protocol desired. The memory system 812 can include one or more memory units 824(0)-824(N).

[0051] The CPU(s) 802 may also be configured to access the display controller(s) 820 over the system bus 808 to control information sent to one or more displays 826. The display controller(s) 820 sends information to the display(s) 826 to be displayed via one or more video processors 828, which process the information to be displayed into a format suitable for the display(s) 826. The display(s) 826 can include any type of display, including but not limited to a cathode ray tube (CRT), a liquid crystal display (LCD), a light emitting diode (LED) display, a plasma display, etc.

[0052] Those of skill in the art will further appreciate that the various illustrative logical blocks, modules, circuits, and algorithms described in connection with the aspects disclosed herein may be implemented as electronic hardware, instructions stored in memory or in another computer-readable medium and executed by a processor or other processing device, or combinations of both. The devices described herein may be employed in any circuit, hardware component, integrated circuit (IC), or IC chip, as examples. Memory disclosed herein may be any type and size of memory and may be configured to store any type of information desired. To clearly illustrate this interchangeability, various illustrative components, blocks, modules, circuits, and steps have been described above generally in terms of their functionality. How such functionality is implemented depends upon the particular application, design choices, and/or design constraints imposed on the overall system. Skilled artisans may implement the described functionality in varying ways for each particular application, but such implementation decisions should not be interpreted as causing a departure from the scope of the present disclosure.

[0053] The various illustrative logical blocks, modules, and circuits described in connection with the aspects disclosed herein may be implemented or performed with a processor, a Digital Signal Processor (DSP), an Application Specific Integrated Circuit (ASIC), a Field Programmable Gate Array (FPGA) or other programmable logic device, discrete gate or transistor logic, discrete hardware components, or any combination thereof designed to perform the functions described herein. A processor may be a microprocessor, but in the alternative, the processor may be any conventional processor, controller, microcontroller, or state machine. A processor may also be implemented as a combination of computing devices (e.g., a combination of a DSP and a microprocessor, a plurality of microprocessors, one or more microprocessors in conjunction with a DSP core, or any other such configuration).

[0054] The aspects disclosed herein may be embodied in hardware and in instructions that are stored in hardware, and may reside, for example, in Random Access Memory (RAM), flash memory, Read Only Memory (ROM), Electrically Programmable ROM (EPROM), Electrically Erasable Programmable ROM (EEPROM), registers, a hard disk, a removable disk, a CD-ROM, or any other form of computer readable medium known in the art. An exemplary storage medium is coupled to the processor such that the processor can read information from, and write information to, the storage medium. In the alternative, the storage medium may be integral to the processor. The processor and the storage medium may reside in an ASIC. The ASIC may reside in a remote station. In the alternative, the processor and the storage medium may reside as discrete components in a remote station, base station, or server.

[0055] It is also noted that the operational steps described in any of the exemplary aspects herein are described to provide examples and discussion. The operations described may be performed in numerous different sequences other than the illustrated sequences. Furthermore, operations described in a single operational step may actually be performed in a number of different steps. Additionally, one or more operational steps discussed in the exemplary aspects may be combined. It is to be understood that the operational steps illustrated in the flow chart diagrams may be subject to numerous different modifications as will be readily apparent to one of skill in the art. Those of skill in the art will also understand that information and signals may be represented using any of a variety of different technologies and techniques. For example, data, instructions, commands, information, signals, bits, symbols, and chips that may be referenced throughout the above description may be represented by voltages, currents, electromagnetic waves, magnetic fields or particles, optical fields or particles, or any combination thereof.

[0056] The previous description of the disclosure is provided to enable any person skilled in the art to make or use the disclosure. Various modifications to the disclosure will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other variations without departing from the spirit or scope of the disclosure. Thus, the disclosure is not intended to be limited to the examples and designs described herein, but is to be accorded the widest scope consistent with the principles and novel features disclosed herein.

What is claimed is:

1. An input/output virtualization (IOV) host controller (HC) (IOV-HC) communicatively coupled to a plurality of input/output (I/O) clients via a corresponding plurality of client register interfaces (CRIs), and to a flash-memory-based storage device;

the IOV-HC comprising:

a plurality of transfer request list (TRL) slot offset registers, each indicating a slot of a shared TRL that is assigned as a base slot to each CRI of the plurality of CRIs; and

a plurality of TRL slot count registers, each indicating a number of slots of the shared TRL assigned to each CRI of the plurality of CRIs; and

the IOV-HC configured to:

receive a transfer request (TR) directed to the flash-memory-based storage device from a CRI of the plurality of CRIs; and

map, by a TR fetch circuit of the IOV-HC, the TR to a slot of the shared TRL based on a TRL slot offset register of the plurality of TRL slot offset registers and a TRL slot count register of the plurality of TRL slot count registers, the TRL slot offset register and the TRL slot count register corresponding to the CRI.

2. The IOV-HC of claim 1, configured to map the TR to the slot of the shared TRL based on a sum of a slot identifier of the TR and a TRL slot offset register value of the TRL slot offset register.

3. The IOV-HC of claim 1, further configured to:

receive, from the flash-memory-based storage device, a response to the TR; and

route, by the IOV-HC, the response to the CRI based on the TRL slot offset register and the TRL slot count register corresponding to the CRI.

4. The IOV-HC of claim 1, further configured to receive, from a virtual machine manager (VMM), a TRL slot offset register value and a TRL slot count register value for each CRI of the plurality of CRIs, responsive to initialization of the IOV-HC.

5. The IOV-HC of claim 1, further configured to receive, from a VMM, a TRL slot offset register value and a TRL slot count register value for a new CRI corresponding to a new I/O client, responsive to creation of the new I/O client by the VMM.

6. The IOV-HC of claim 1, further comprising:

a plurality of task management request list (TMRL) slot offset registers, each indicating a slot of a shared TMRL that is assigned as a base slot to each CRI of the plurality of CRIs;

a plurality of TMRL slot count registers, each indicating a number of slots of the shared TMRL assigned to each CRI of the plurality of CRIs; and

the IOV-HC further configured to:

receive a task management request (TMR) from a CRI of the plurality of CRIs; and

map, by the TR fetch circuit of the IOV-HC, the TMR to a slot of the shared TMRL based on a TMRL slot offset register of the plurality of TMRL slot offset registers and a TMRL slot count register of the plurality of TMRL slot count registers, the TMRL slot offset register and the TMRL slot count register corresponding to the CRI.

7. The IOV-HC of claim 1, communicatively coupled to the flash-memory-based storage device according to an Embedded MultiMedia Card (eMMC) standard.

8. The IOV-HC of claim 1, communicatively coupled to the flash-memory-based storage device according to a Universal Flash Storage (UFS) standard.

9. The IOV-HC of claim 1 integrated into an integrated circuit (IC).

10. The IOV-HC of claim 1 integrated into a device selected from the group consisting of: a set top box; an entertainment unit; a navigation device; a communications device; a fixed location data unit; a mobile location data unit; a mobile phone; a cellular phone; a computer; a portable computer; a desktop computer; a personal digital assistant (PDA); a monitor; a computer monitor; a television; a tuner; a radio; a satellite radio; a music player; a digital music player; a portable music player; a digital video player; a video player; a digital video disc (DVD) player; and a portable digital video player.

11. A method for providing virtual transfer request lists (TRLs) for multiple hosts, comprising:

receiving, by an input/output virtualization (IOV) host controller (HC) (IOV-HC), a transfer request (TR) directed to a flash-memory-based storage device from a client register interface (CRI) of a plurality of CRIs; and

mapping, by a TR fetch circuit of the IOV-HC, the TR to a slot of a shared TRL based on a TRL slot offset register of a plurality of TRL slot offset registers and a TRL slot count register of a plurality of TRL slot count registers, the TRL slot offset register and the TRL slot count register corresponding to the CRI;

wherein:

the plurality of TRL slot offset registers each indicates a slot of the shared TRL that is assigned as a base slot to a corresponding CRI of the plurality of CRIs; and

the plurality of TRL slot count registers each indicates a number of slots of the shared TRL assigned to the corresponding CRI of the plurality of CRIs.

12. The method of claim 11, wherein mapping the TR to the slot of the shared TRL is based on a sum of a slot identifier of the TR and a TRL slot offset register value of the TRL slot offset register.

13. The method of claim 11, further comprising:

receiving, from the flash-memory-based storage device, a response to the TR; and

routing the response to the CRI based on the TRL slot offset register and the TRL slot count register corresponding to the CRI.

14. The method of claim 11, further comprising receiving, from a virtual machine manager (VMM), a TRL slot offset register value and a TRL slot count register value for each CRI of the plurality of CRIs, responsive to initialization of the IOV-HC.

15. The method of claim 11, further comprising receiving, from a VMM, a TRL slot offset register value and a TRL slot count register value for a new CRI corresponding to a new input/output (I/O) client, responsive to creation of the new I/O client by the VMM.

16. The method of claim 11, further comprising:

receiving, by the IOV-HC, a task management request (TMR) from a CRI of the plurality of CRIs; and

mapping, by the TR fetch circuit of the IOV-HC, the TMR to a slot of a shared task management request list (TMRL) based on a TMRL slot offset register of a plurality of TMRL slot offset registers and a TMRL slot count register of a plurality of TMRL slot count registers, the TMRL slot offset register and the TMRL slot count register corresponding to the CRI;

wherein:

the plurality of TMRL slot offset registers each indicates a slot of the shared TMRL that is assigned as a base slot to the corresponding CRI of the plurality of CRIs; and

the plurality of TMRL slot count registers each indicates a number of slots of the shared TMRL assigned to the corresponding CRI of the plurality of CRIs.

17. The method of claim 11, wherein the flash-memory-based storage device comprises an Embedded MultiMedia Card (eMMC) device.

18. The method of claim 11, wherein the flash-memory-based storage device comprises a Universal Flash Storage (UFS) device.

19. An input/output virtualization (IOV) host controller (HC) (IOV-HC), comprising:

a means for receiving a transfer request (TR) directed to a flash-memory-based storage device from a client register interface (CRI) of a plurality of CRIs; and

a means for mapping the TR to a slot of a shared transfer request list (TRL) based on a TRL slot offset register of a plurality of TRL slot offset registers and a TRL slot count register of a plurality of TRL slot count registers, the TRL slot offset register and the TRL slot count register corresponding to the CRI;

wherein:

the plurality of TRL slot offset registers each indicates a slot of the shared TRL that is assigned as a base slot to a corresponding CRI of the plurality of CRIs; and

the plurality of TRL slot count registers each indicates a number of slots of the shared TRL assigned to the corresponding CRI of the plurality of CRIs.

20. The IOV-HC of claim 19, further comprising:

a means for receiving, from the flash-memory-based storage device, a response to the TR; and

a means for routing the response to the CRI based on the TRL slot offset register and the TRL slot count register corresponding to the CRI.

21. The IOV-HC of claim 19, further comprising a means for receiving, from a virtual machine manager (VMM), a TRL slot offset register value and a TRL slot count register value for each CRI of the plurality of CRIs, responsive to initialization of the IOV-HC.

22. The IOV-HC of claim 19, further comprising a means for receiving, from a VMM, a TRL slot offset register value and a TRL slot count register value for a new CRI corresponding to a new input/output (I/O) client, responsive to creation of the new I/O client by the VMM.

23. The IOV-HC of claim 19, further comprising:

a means for receiving a task management request (TMR) from a CRI of the plurality of CRIs; and

a means for mapping the TMR to a slot of a shared task management request list (TMRL) based on a TMRL slot offset register of a plurality of TMRL slot offset registers and a TMRL slot count register of a plurality of TMRL slot count registers, the TMRL slot offset register and the TMRL slot count register corresponding to the CRI;

wherein:

the plurality of TMRL slot offset registers each indicates a slot of the shared TMRL that is assigned as a base slot to a corresponding CRI of the plurality of CRIs; and

the plurality of TMRL slot count registers each indicates a number of slots of the shared TMRL assigned to the corresponding CRI of the plurality of CRIs.

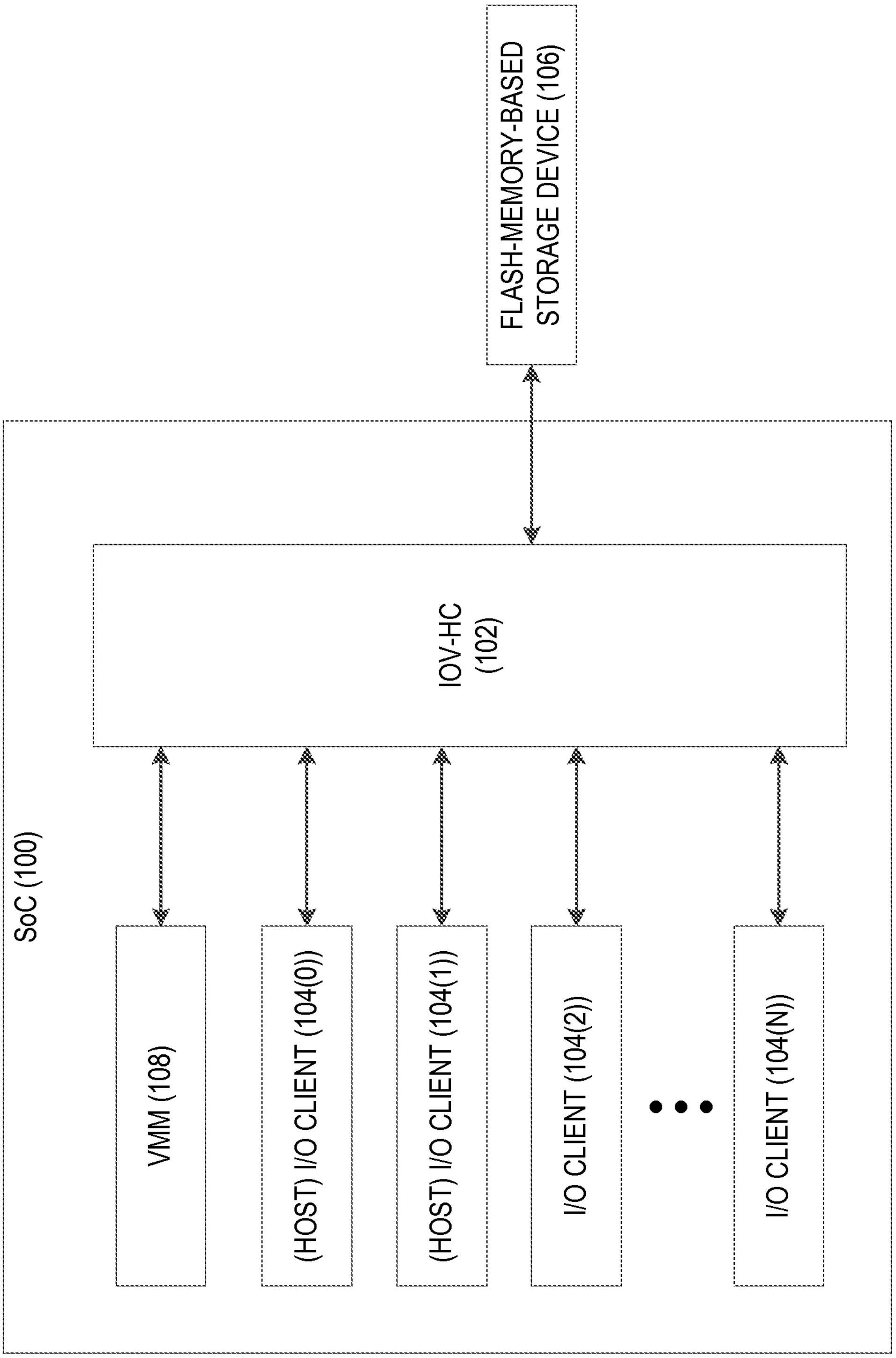

FLASH-MEMORY-BASED STORAGE DEVICE (106)
SoC (100)
IOV-HC (102)
VMM (108)
(HOST) I/O CLIENT (104(0))
(HOST) I/O CLIENT (104(1))
I/O CLIENT (104(2))
I/O CLIENT (104(N))

FIG. 1

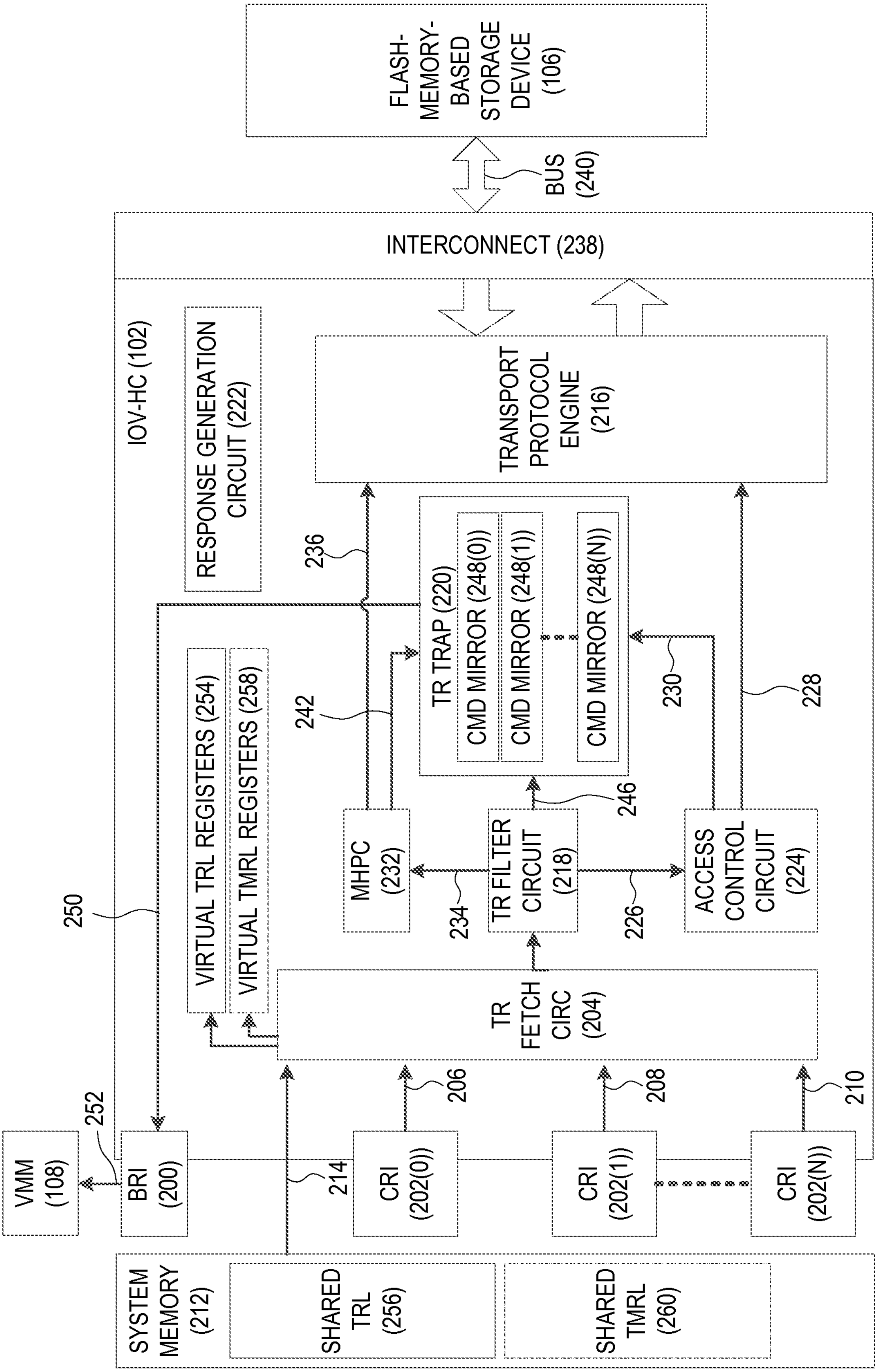

FLASH-MEMORY-BASED STORAGE DEVICE (106)
BUS (240)
INTERCONNECT (238)
IOV-HC (102)
RESPONSE GENERATION CIRCUIT (222)
TRANSPORT PROTOCOL ENGINE (216)
236
TR TRAP (220)
CMD MIRROR (248(0))
CMD MIRROR (248(1))
CMD MIRROR (248(N))
242
230
228
VIRTUAL TRL REGISTERS (254)
VIRTUAL TMRL REGISTERS (258)
250
MHPC (232)
TR FILTER CIRCUIT (218)
246
ACCESS CONTROL CIRCUIT (224)
234
226
TR FETCH CIRC (204)
252
206
208
210
VMM (108)
BRI (200)
214
CRI (202(0))
CRI (202(1))
CRI (202(N))
SYSTEM MEMORY (212)
SHARED TRL (256)
SHARED TMRL (260)

FIG. 2

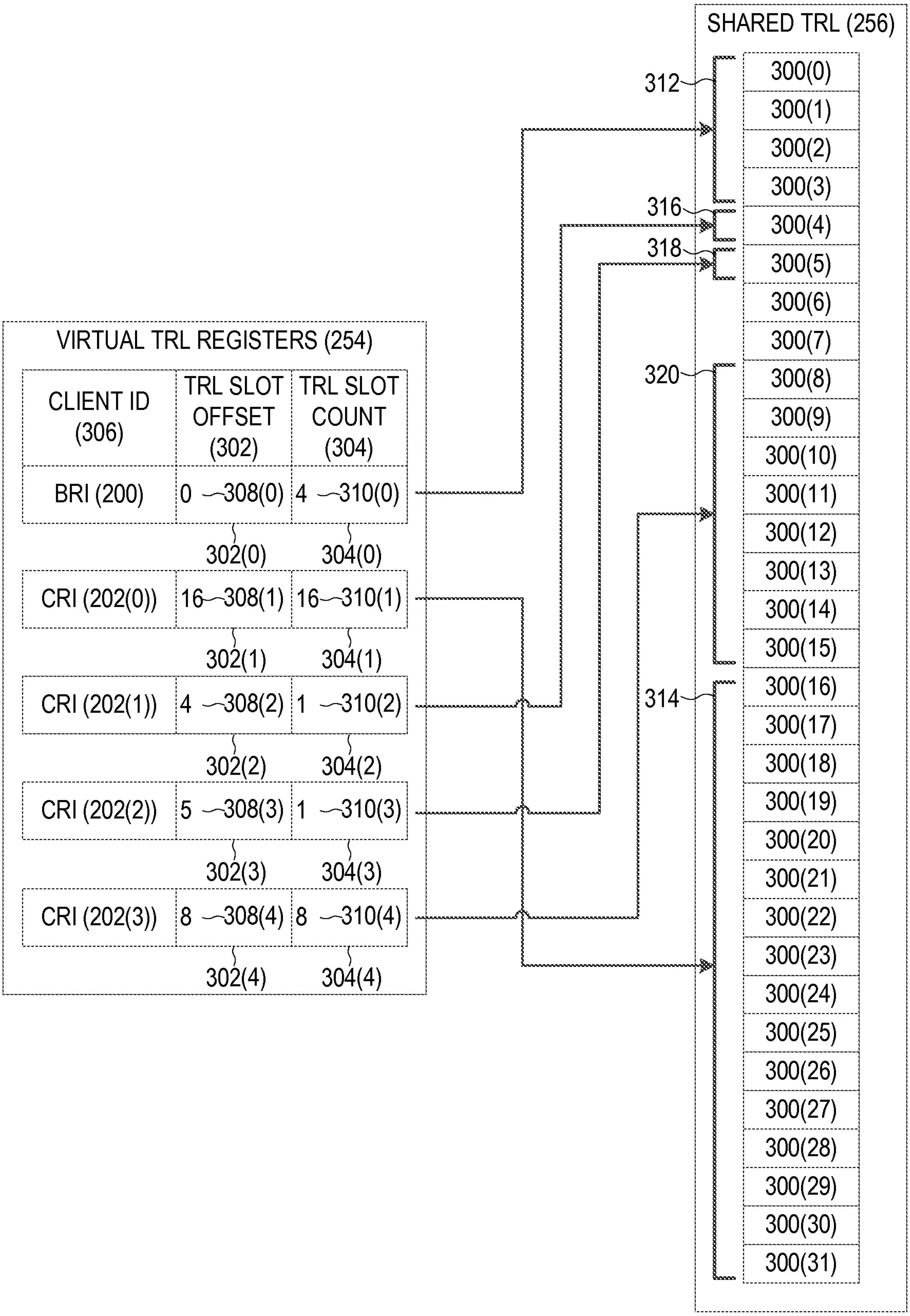

SHARED TRL (256)
312
316
318
320
314
300(0)
300(1)
300(2)
300(3)
300(4)
300(5)
300(6)
300(7)
300(8)
300(9)
300(10)
300(11)
300(12)
300(13)
300(14)
300(15)
300(16)
300(17)
300(18)
300(19)
300(20)
300(21)
300(22)
300(23)
300(24)
300(25)
300(26)
300(27)
300(28)
300(29)
300(30)
300(31)
VIRTUAL TRL REGISTERS (254)
CLIENT ID (306)
TRL SLOT OFFSET (302)
TRL SLOT COUNT (304)
BRI (200)
0 308(0)
4 310(0)
302(0)
304(0)
CRI (202(0))
16 308(1)
16 310(1)
302(1)
304(1)
CRI (202(1))
4 308(2)
1 310(2)
302(2)
304(2)
CRI (202(2))
5 308(3)
1 310(3)
302(3)
304(3)
CRI (202(3))
8 308(4)
8 310(4)
302(4)
304(4)

FIG. 3

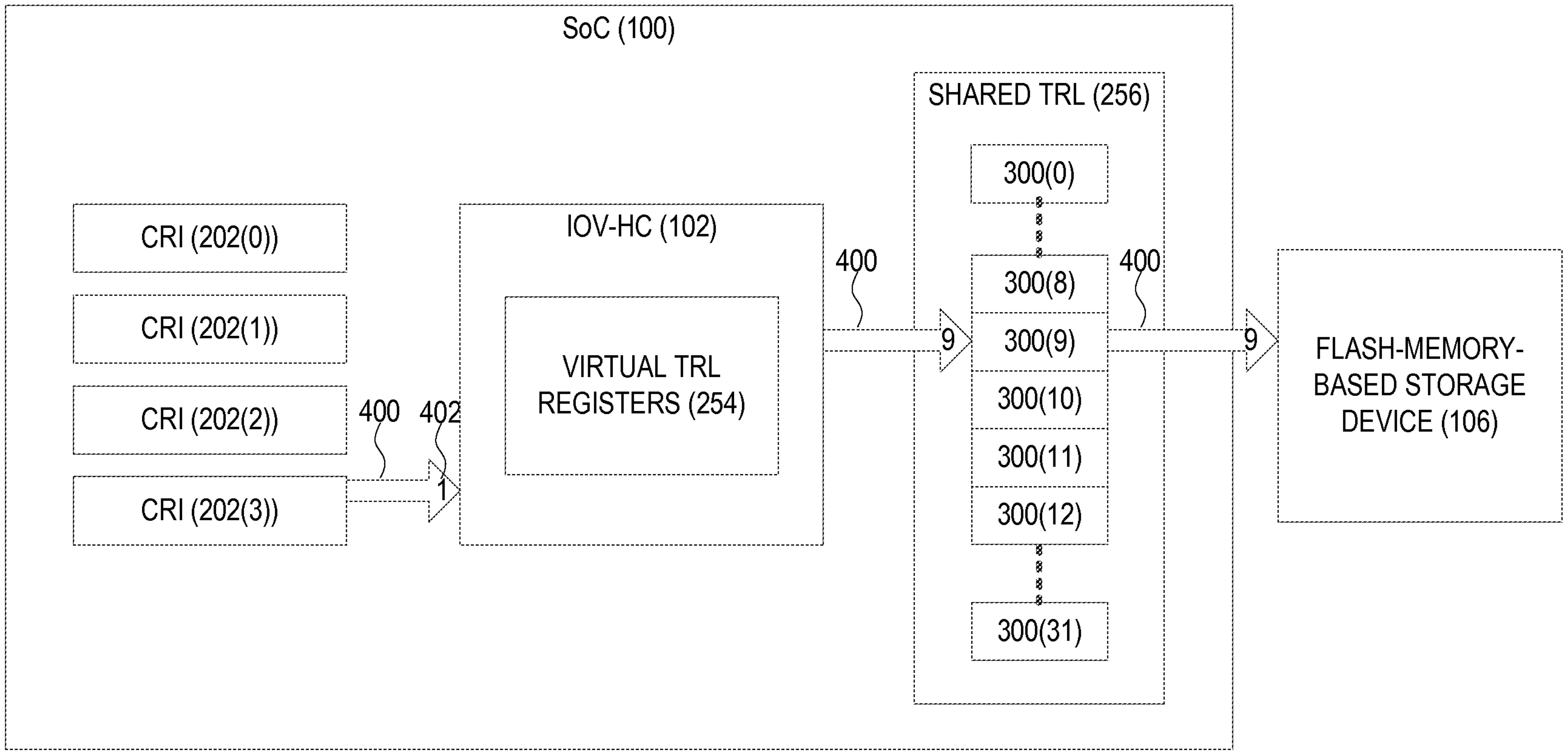

SoC (100)
CRI (202(0))
CRI (202(1))
CRI (202(2))
CRI (202(3))
400
402
1
IOV-HC (102)
VIRTUAL TRL
REGISTERS (254)
400
9
SHARED TRL (256)
300(0)
300(8)
300(9)
300(10)
300(11)
300(12)
300(31)
400
9
FLASH-MEMORY-
BASED STORAGE
DEVICE (106)

FIG. 4A

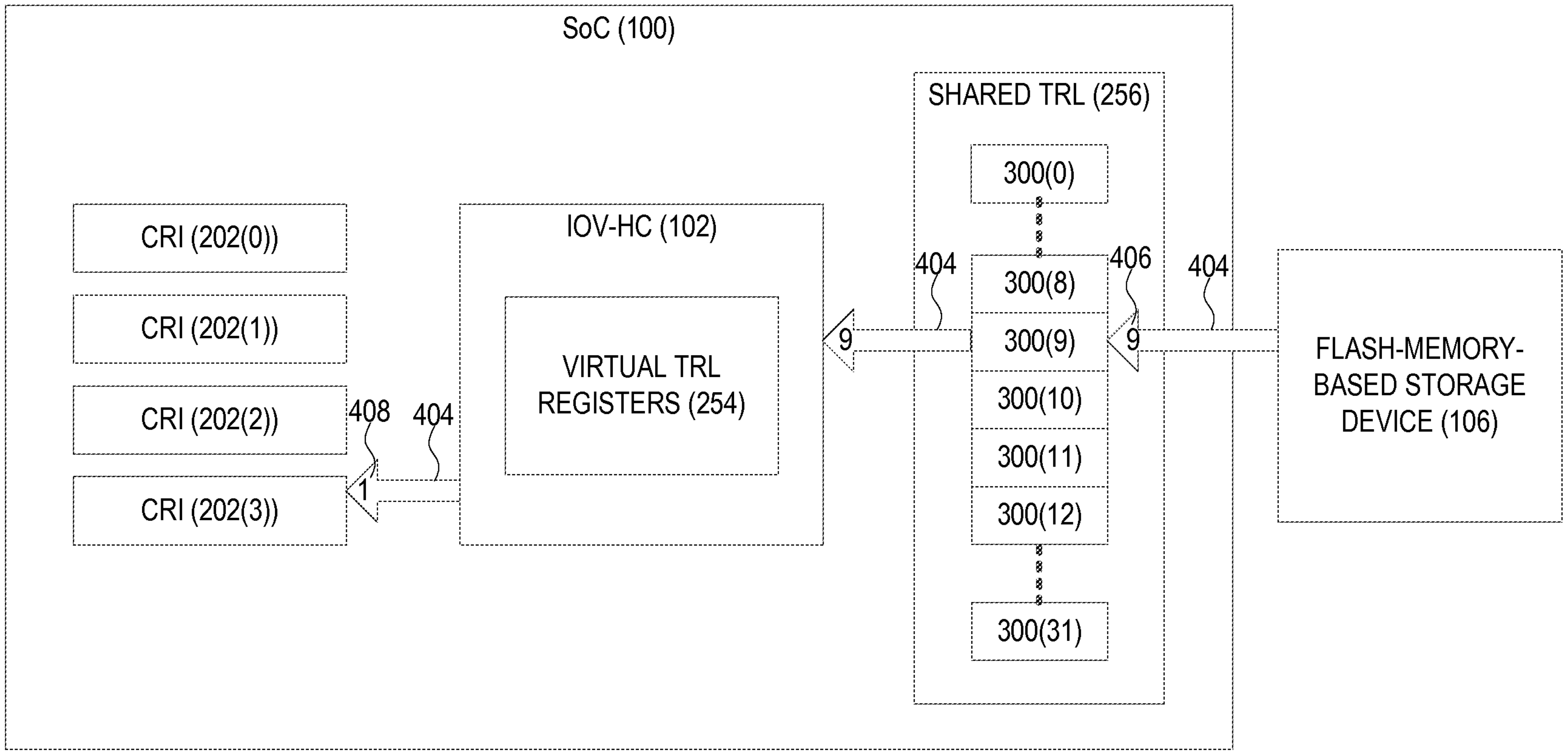

SoC (100)
SHARED TRL (256)
300(0)
300(8)
300(9)
300(10)
300(11)
300(12)
300(31)
404
406
404
9
9
FLASH-MEMORY-BASED STORAGE DEVICE (106)
IOV-HC (102)
VIRTUAL TRL REGISTERS (254)
CRI (202(0))
CRI (202(1))
CRI (202(2))
CRI (202(3))
408
404
1

FIG. 4B

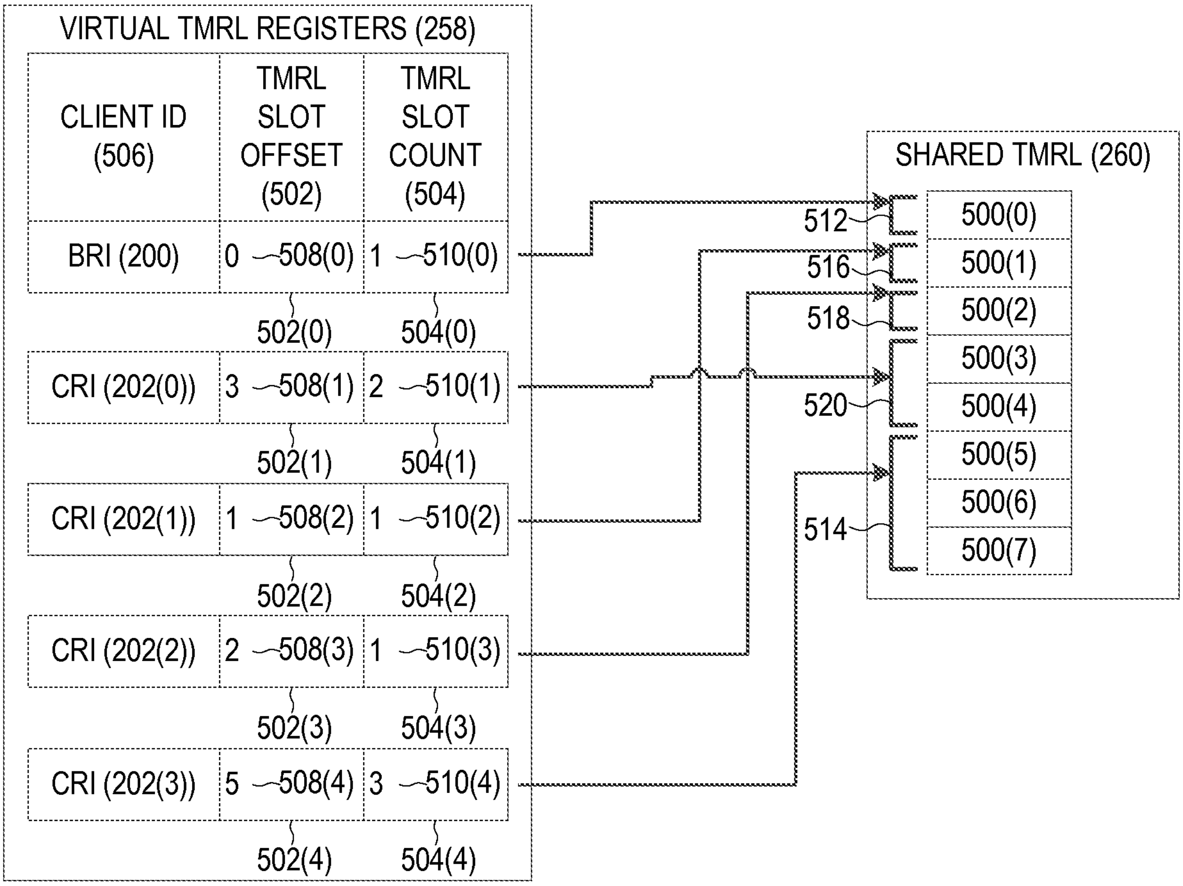

VIRTUAL TMRL REGISTERS (258)
CLIENT ID (506)
TMRL SLOT OFFSET (502)
TMRL SLOT COUNT (504)
BRI (200)
0
508(0)
1
510(0)
502(0)
504(0)
CRI (202(0))
3
508(1)
2
510(1)
502(1)
504(1)
CRI (202(1))
1
508(2)
1
510(2)
502(2)
504(2)
CRI (202(2))
2
508(3)
1
510(3)
502(3)
504(3)
CRI (202(3))
5
508(4)
3
510(4)
502(4)
504(4)
SHARED TMRL (260)
512
516
518
520
514
500(0)
500(1)
500(2)
500(3)
500(4)
500(5)
500(6)
500(7)

FIG. 5

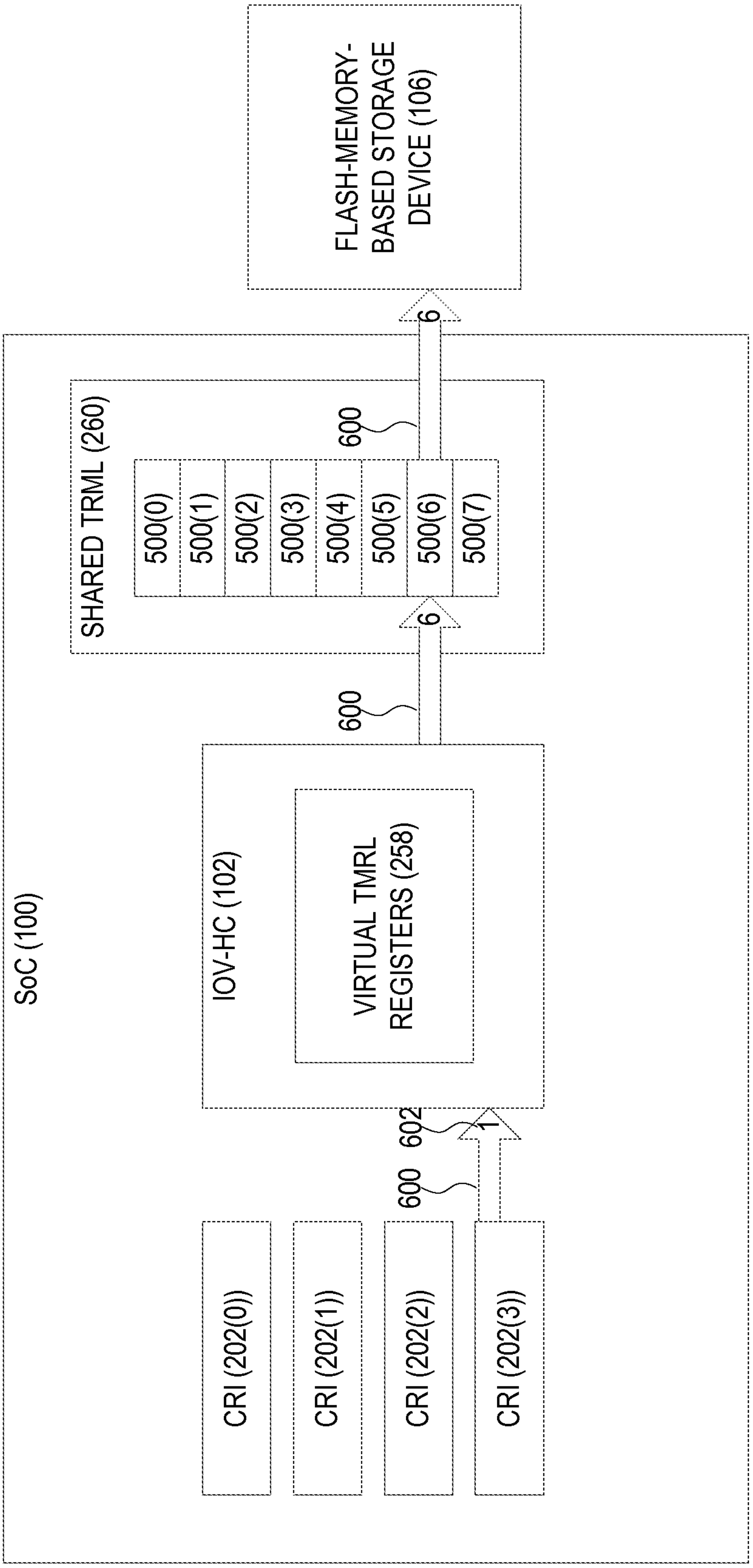

SoC (100)
CRI (202(0))
CRI (202(1))
CRI (202(2))
CRI (202(3))
600
602
1
IOV-HC (102)
VIRTUAL TMRL REGISTERS (258)
600
6
SHARED TRML (260)
500(0)
500(1)
500(2)
500(3)
500(4)
500(5)
500(6)
500(7)
600
6
FLASH-MEMORY-BASED STORAGE DEVICE (106)

FIG. 6

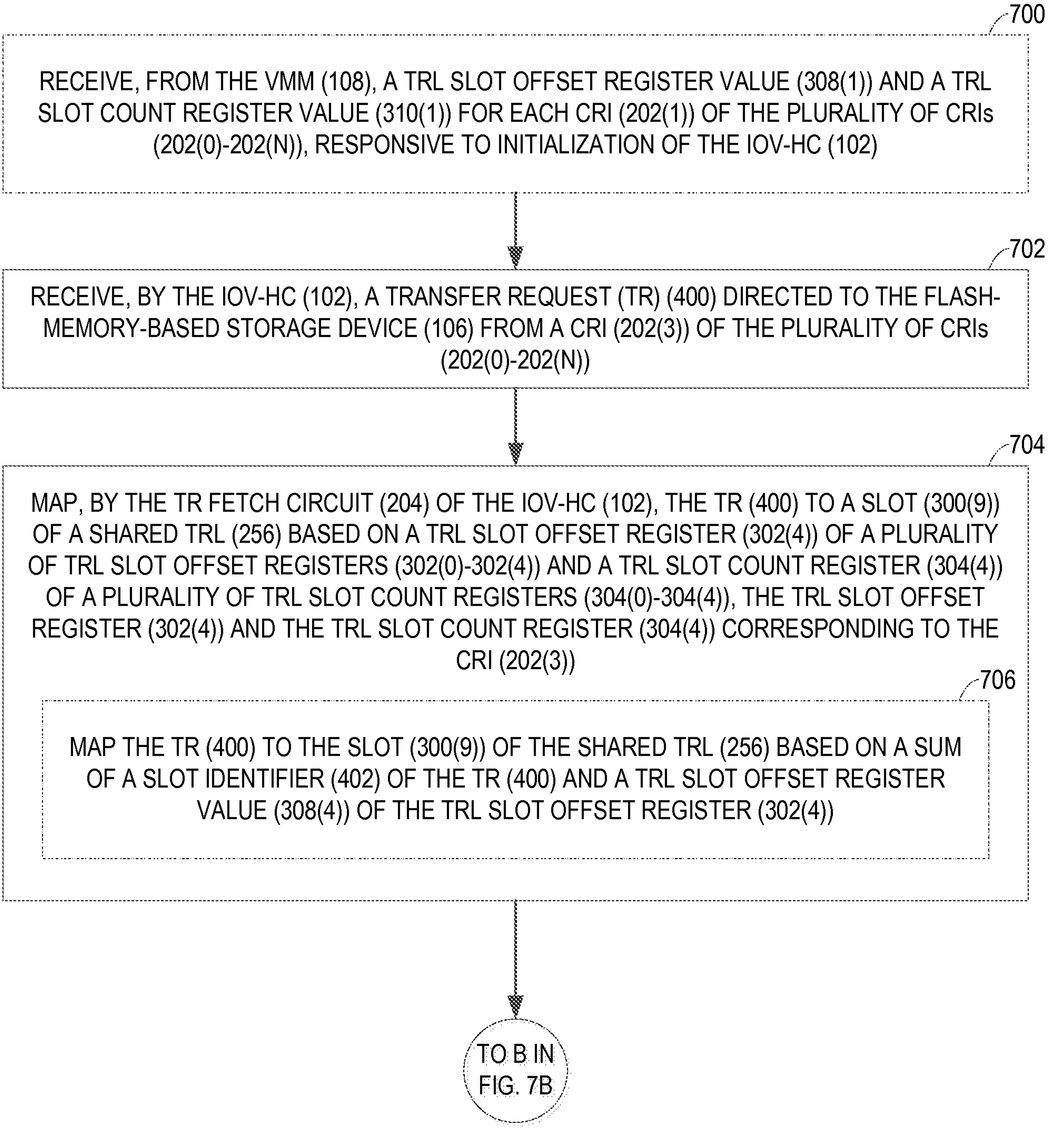
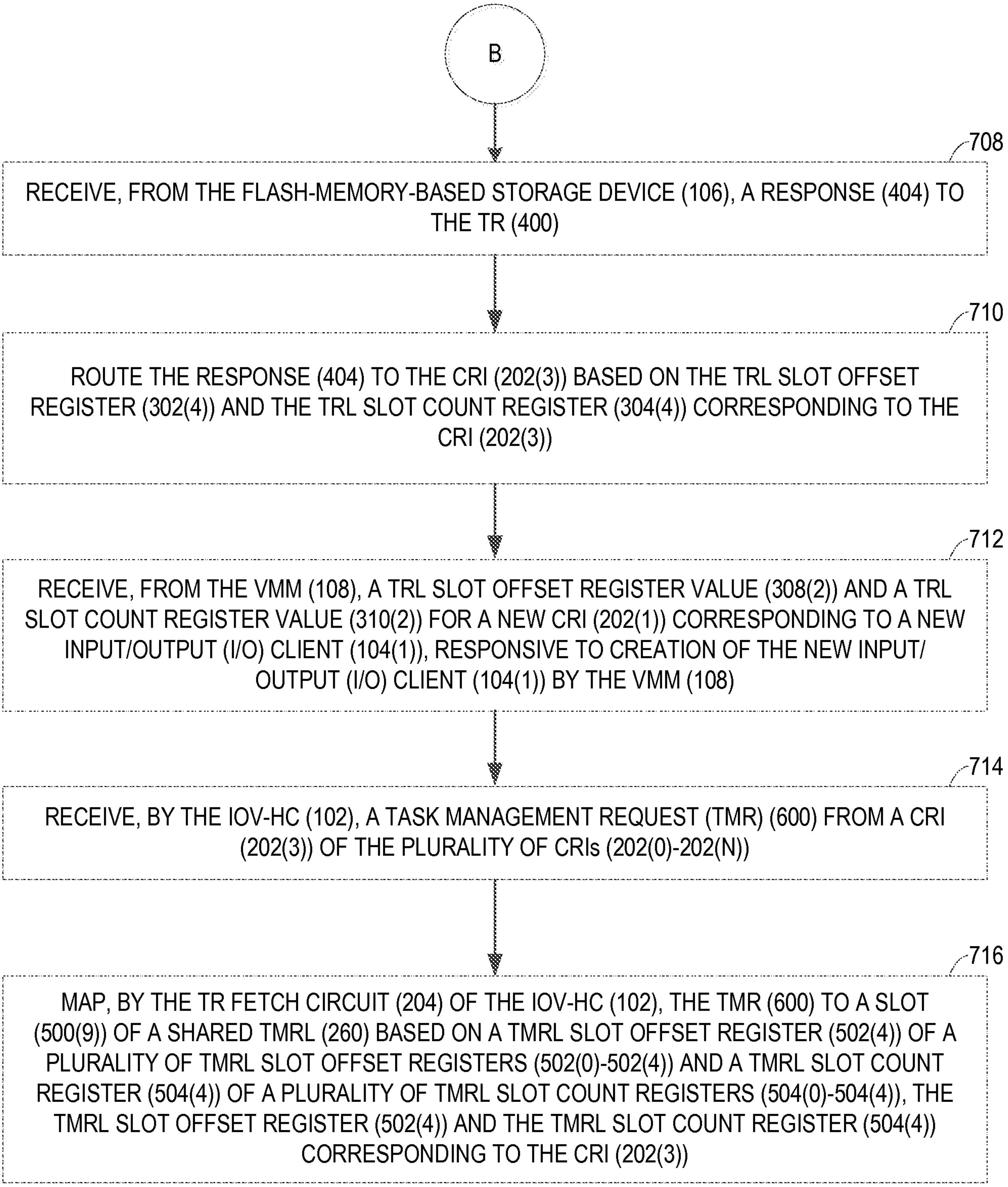

700
RECEIVE, FROM THE VMM (108), A TRL SLOT OFFSET REGISTER VALUE (308(1)) AND A TRL SLOT COUNT REGISTER VALUE (310(1)) FOR EACH CRI (202(1)) OF THE PLURALITY OF CRIs (202(0)-202(N)), RESPONSIVE TO INITIALIZATION OF THE IOV-HC (102)
702
RECEIVE, BY THE IOV-HC (102), A TRANSFER REQUEST (TR) (400) DIRECTED TO THE FLASH-MEMORY-BASED STORAGE DEVICE (106) FROM A CRI (202(3)) OF THE PLURALITY OF CRIs (202(0)-202(N))
704
MAP, BY THE TR FETCH CIRCUIT (204) OF THE IOV-HC (102), THE TR (400) TO A SLOT (300(9)) OF A SHARED TRL (256) BASED ON A TRL SLOT OFFSET REGISTER (302(4)) OF A PLURALITY OF TRL SLOT OFFSET REGISTERS (302(0)-302(4)) AND A TRL SLOT COUNT REGISTER (304(4)) OF A PLURALITY OF TRL SLOT COUNT REGISTERS (304(0)-304(4)), THE TRL SLOT OFFSET REGISTER (302(4)) AND THE TRL SLOT COUNT REGISTER (304(4)) CORRESPONDING TO THE CRI (202(3))
706
MAP THE TR (400) TO THE SLOT (300(9)) OF THE SHARED TRL (256) BASED ON A SUM OF A SLOT IDENTIFIER (402) OF THE TR (400) AND A TRL SLOT OFFSET REGISTER VALUE (308(4)) OF THE TRL SLOT OFFSET REGISTER (302(4))
TO B IN FIG. 7B

FIG. 7A



RECEIVE, FROM THE FLASH-MEMORY-BASED STORAGE DEVICE (106), A RESPONSE (404) TO THE TR (400)

ROUTE THE RESPONSE (404) TO THE CRI (202(3)) BASED ON THE TRL SLOT OFFSET REGISTER (302(4)) AND THE TRL SLOT COUNT REGISTER (304(4)) CORRESPONDING TO THE CRI (202(3))

RECEIVE, FROM THE VMM (108), A TRL SLOT OFFSET REGISTER VALUE (308(2)) AND A TRL SLOT COUNT REGISTER VALUE (310(2)) FOR A NEW CRI (202(1)) CORRESPONDING TO A NEW INPUT/OUTPUT (I/O) CLIENT (104(1)), RESPONSIVE TO CREATION OF THE NEW INPUT/ OUTPUT (I/O) CLIENT (104(1)) BY THE VMM (108)

RECEIVE, BY THE IOV-HC (102), A TASK MANAGEMENT REQUEST (TMR) (600) FROM A CRI (202(3)) OF THE PLURALITY OF CRIs (202(0)-202(N))

MAP, BY THE TR FETCH CIRCUIT (204) OF THE IOV-HC (102), THE TMR (600) TO A SLOT (500(9)) OF A SHARED TMRL (260) BASED ON A TMRL SLOT OFFSET REGISTER (502(4)) OF A PLURALITY OF TMRL SLOT OFFSET REGISTERS (502(0)-502(4)) AND A TMRL SLOT COUNT REGISTER (504(4)) OF A PLURALITY OF TMRL SLOT COUNT REGISTERS (504(0)-504(4)), THE TMRL SLOT OFFSET REGISTER (502(4)) AND THE TMRL SLOT COUNT REGISTER (504(4)) CORRESPONDING TO THE CRI (202(3))

FIG. 7B

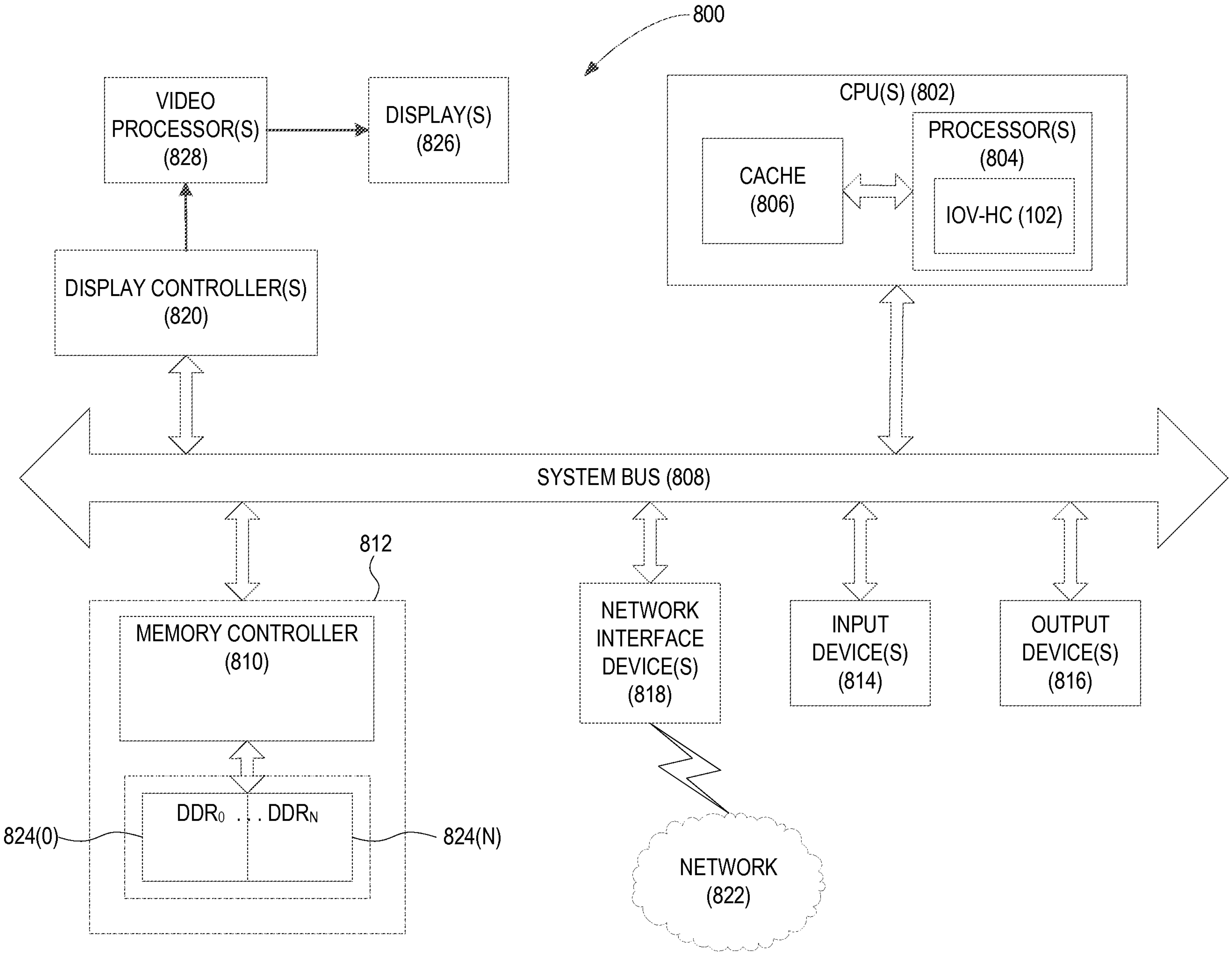

800
VIDEO PROCESSOR(S) (828)
DISPLAY(S) (826)
CPU(S) (802)
CACHE (806)
PROCESSOR(S) (804)
IOV-HC (102)
DISPLAY CONTROLLER(S) (820)
SYSTEM BUS (808)
812
MEMORY CONTROLLER (810)
DDR_0 . . . DDR_N
824(0)
824(N)
NETWORK INTERFACE DEVICE(S) (818)
INPUT DEVICE(S) (814)
OUTPUT DEVICE(S) (816)
NETWORK (822)

FIG. 8

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2015/033953

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F13/16 G06F3/06 G11C7/00
ADD.

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 G11C

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

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

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP 2 492 916 A1 (HUAWEI TECH CO LTD [CN]) 29 August 2012 (2012-08-29) figures 2,3,6 -----	1-23
A	US 2012/030408 A1 (FLYNN DAVID [US] ET AL) 2 February 2012 (2012-02-02) paragraphs [0127] - [0133]; figure 4 -----	1-23
A	US 2013/275656 A1 (TALAGALA NISHA [US] ET AL) 17 October 2013 (2013-10-17) figure 1B -----	1-23

□ 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 30 July 2015	Date of mailing of the international search report 11/08/2015
Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Authorized officer Albert, Jozsef

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2015/033953

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 2492916 A1	29-08-2012	CN 101833989 A	15-09-2010
		EP 2492916 A1	29-08-2012
		KR 20120087980 A	07-08-2012
		RU 2012128534 A	20-01-2014
		US 2012233386 A1	13-09-2012
		WO 2011147324 A1	01-12-2011
US 2012030408 A1	02-02-2012	CN 103098034 A	08-05-2013
		EP 2598996 A2	05-06-2013
		US 2012030408 A1	02-02-2012
		US 2013205097 A1	08-08-2013
		WO 2012016089 A2	02-02-2012
US 2013275656 A1	17-10-2013	NONE	