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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
17 August 2006 (17.08.2006)

PCT

(10) International Publication Number
WO 2006/085272 A2

(51) International Patent Classification: Not classified

(21) International Application Number:
PCT/IB2006/050414

(22) International Filing Date: 8 February 2006 (08.02.2006)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/651,429 8 February 2005 (08.02.2005) US

(71) Applicant *(for all designated States except US)*: **KONINKLIJKE PHILIPS ELECTRONICS N.V.** [NL/NL]; Groenewoudseweg 1, NL-5621 BA Eindhoven (NL).

(71) Applicant *(for AE only)*: **U.S. PHILIPS CORPORATION** [US/US]; 1251 Avenue Of The Americas, New York, New York 10020 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **EE, Lee, Chee** [NL/NL]; Groenewoudseweg 1, NL-5621 BA Eindhoven (NL). **NG, Chee, Yu** [NL/NL]; Groenewoudseweg 1, NL-5621 BA Eindhoven (NL).

(74) Common Representative: **KONINKLIJKE PHILIPS ELECTRONICS N.V.**; C/o Daniel L. Michalek, 1109 Mckay Drive, M/s-41sj, San Jose, California 95131-170 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, LY, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

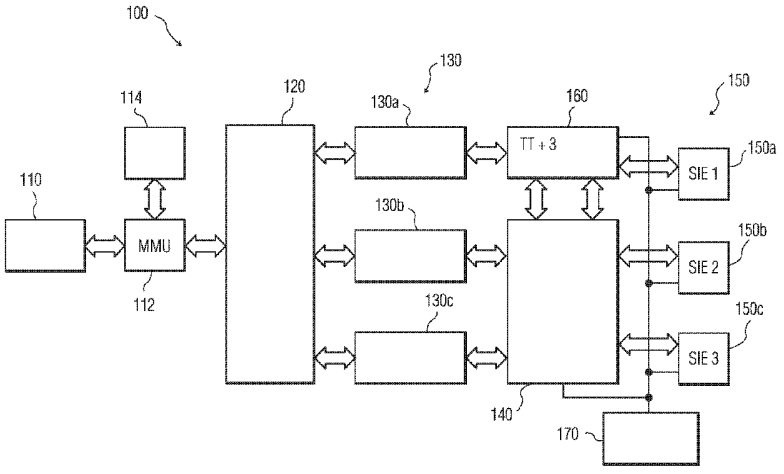
Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

— *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

[Continued on next page]

(54) **Title:** SCALABLE UNIVERSAL SERIAL BUS ARCHITECTURE



(57) **Abstract:** A serial bus interface circuit (100) for use in a computer comprises a master scheduler (120) configured to communicate serial data with the computer. A plurality of transfer controllers (130) are coupled to the master scheduler and configured to manage serial tranactions. A router (140) has a number of ports coupled to the transfer controllers and selectively communicates serial data between selected ports. A plurality of serial interlaces (150) are coupled to the router and adapted for coupling to peripheral devices. In one embodiment, the serial bus interface circuit further comprises a transaction translator (160) coupled to a transfer controller and a serial interface, for normalizing the data speed. In another embodiment, the master scheduler (120) is coupled to a memory management unit (112) and communicates serial data with the memory management unit. Advantages of the invention include the ability to provide high-speed serial communication to multiple ports simultaneously.

WO 2006/085272 A2

Published:

— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

SCALABLE UNIVERSAL SERIAL BUS ARCHITECTURE

The present invention relates to the general field of serial communications, and in particular to a scalable universal serial bus (USB) architecture designed to increase usable bandwidth over a the USB interface.

A conventional universal serial bus (USB) interface employs a single controller over multiple ports, which creates a per-port bandwidth restriction. For example, a typical PCI-based enhanced host controller interface (EHCI) USB host employs a single EHCI controller, providing a total of 480Mbps (60MBytes/s) bandwidth even though there may be up to eight ports available. An exemplary EHCI controller is offered by Philips under the part number ISP1561, which datasheet can be seen at http://www.semiconductors.philips.com/cgi-bin/pldb/pip/isp1561.html. Connecting eight high-speed storage devices (e.g. hard disks) to the eight ports would permit only 60Mbps (7.5MBytes/s) of theoretical maximum bandwidth per device if they are all active. Thus, while the USB interface specification permits high bandwidth, the conventional implementations of the interface do not scale to provide the high bandwidth to the multiple ports simultaneously.

The invention focuses of a technique to increase the available bandwidth to many ports for high bandwidth data transfer applications. The invention is directed to a scalable-USB Architecture (S-USB) that improves conventional USB host designs with the ability to send multiple data streams on different ports simultaneously. This technique supports an eight-port USB host that can provide full 480Mbps bandwidth to each of the eight connected devices.

In an exemplary embodiment, a serial bus interface circuit for use in a computer comprises a master scheduler configured to communicate serial data with the computer. A plurality of transfer controllers are coupled to the master scheduler and configured to manage serial transactions. A router has a number of ports coupled to the transfer controllers and is configured to selectively communicate serial data between selected ports. A plurality of serial interfaces are coupled to the router and adapted for coupling to peripheral devices.

In one embodiment, the serial bus interface circuit further comprises a transaction translator coupled to a transfer controller and a serial interface, and configured to translate the serial data to a normalized speed.

In another embodiment, the master scheduler is coupled to a memory management unit and configured to communicate serial data with the memory management unit.

Advantages of the invention include the ability to provide high-speed serial communication to multiple ports simultaneously.

The invention is described with reference to the following figures.

Figure 1 depicts a serial bus interface circuit according to an embodiment of the invention.

Figure 2 depicts an exemplary connection data flow according to an embodiment of the invention.

Figure 3 depicts an exemplary connection data flow according to an embodiment of the invention.

Figure 4 depicts an exemplary connection data flow according to an embodiment of the invention.

The invention is described with reference to specific apparatus and embodiments. Those skilled in the art will recognize that the description is for illustration and to provide the best mode of practicing the invention. For example, while reference is made to universal serial bus (USB), the invention is applicable to any communications interface. Likewise, while reference is made to PCI bus architectures, any similar bus architectures and protocols may be employed in the invention. Additionally, while reference is made to hard disks and other typical USB peripheral devices, the invention can be employed with any type of peripheral device that it compatibly designed with the invention and implementation thereof. Further, while the exemplary embodiment depicts three transfer controllers and three serial interface engines (SIE), any number can be used consistent with the description of the invention.

Figure 1 depicts a serial bus interface circuit 100 according to an embodiment of the invention. The interface circuit 100 is designed for use in a computer, including a master scheduler 120 configured to communicate serial data with the computer processor interface and memory, which is demonstrably shown as blocks 110, 112 and 114. CPU in the CPU interface 110 stands for central processing unit, MMU 112 stands for memory management unit, and RAM 114 stands for random access memory.

In one embodiment, the master scheduler 120 is coupled to the memory management unit 112 and configured to communicate serial data with the memory management unit. Depending on the total bandwidth requirement, the master scheduler can

be optimized for power/bandwidth efficiency. For example, it may clock down to provide 600Mbit/s bandwidth for a 3-port system when device connected are not using the maximum possible bandwidth.

A plurality of transfer controllers 130a-130c are coupled to the master scheduler and configured to manage serial transactions. The transfer controllers 130a-130c include circuit logic and RAM for handling a single USB high-speed (HS) transaction without any external intervention.

The master scheduler 120 is a modified version of an enhanced host controller interface (EHCI) with additional features that allow it to schedule simultaneous transaction on multiple ports. An exemplary EHCI controller is offered by Philips under the part number ISP1561, which datasheet can be seen at http://www.semiconductors.philips.com/cgi-bin/pldb/pip/isp1561.html. In one aspect, the EHCI circuit is replicated three times to provide the three exemplary channels, and can be replicated n times to provide n channels. In another aspect, the EHCI is modified by adding additional I/O ports, but the communications circuits share certain resources, for example, the PCI controller and the global control. This aspect is very efficient in terms of space conservation and high performance since it shares certain resources while adding the functionality under the invention. Additional aspects are anticipated by similar modifications. On scheduling a particular transaction to one of the transfer controllers, the master scheduler proceeds to schedule transactions for other transfer controllers, without waiting for the completion of the particular transaction. In this way, three streams of data can be sent out simultaneously.

A router 140 has a number of ports coupled to the transfer controllers. A plurality of serial interface engines (SIE) 150a-150c are also coupled to the router and adapted for coupling to peripheral devices. The router 140 is configured to selectively communicate serial data between the serial interfaces and the transfer controllers, under control of a port controller 170, which receives information from the SIEs and directs the information as descibed below.

In one embodiment, the serial bus interface further comprises a transaction translator 160 coupled to a transfer controller and a serial interface, and configured to translate the serial data to a normalized speed. In this embodiment, the router is configured to selectively communicate serial data between the serial interfaces, the transfer controllers and the transaction translator.

As shown in the exemplary embodiment, one transaction translator (TT) 160 is shared among the three ports using a HUB and a high-speed/full-speed/low-speed (HS/FS/LS) router. When full-speed/low-speed (FS/LS) devices are connected to SIE2 150b or SIE3 150c, they are routed to a downstream port of the transaction translator 160 and appear as a device connected to SIE1 150a.

A port controller 170 is coupled to the serial interface engines (SIE) 150a-160c and the transaction translator 160. The port controller associates the SIE with the transaction translator based on the type of device is connected, e.g. HS/FS/LS, and whether the corresponding traffic is to be routed to the transaction translator. This function is described below for three examples.

Figures 2, 3 and 4 depict exemplary connection data flow according to embodiments of the invention.

Figure 2 depicts three data streams 210a-210b corresponding with three connected devices 220a-220c, respectively. Device 220a is a high-speed/full-speed/low-speed (HS/FS/LS) device meaning that it operates at any of those speeds. Devices 220b-220c are high-speed (HS) devices meaning that they operate at high-speed.

In this example, transfer controller 130a is controlling port 1 for high-speed, full-speed and low-speed traffic, transfer controller 130b is controlling port 2 for high-speed traffic, and transfer controller 130a is controlling port 3 for high-speed traffic. A table view is provided in Table 1.

TABLE 1

Transfer Controller	Ports and Speed
1 (130a)	Port 1 HS/FS/LS
2 (130b)	Port 2 HS
3 (130c)	Port 3 HS

Figure 3 depicts three data streams 310a-310b corresponding with three connected devices 320a-320c, respectively. Device 320a is a high-speed/full-speed/low-speed (HS/FS/LS) device meaning that it operates at any of those speeds. Device 320b is a full-speed/low-speed (FS/LS) device meaning that it operates at any of those speeds. Device 320c is a high-speed (HS) device meaning that it operates at high-speed.

In this example, transfer controller 130a is controlling port 1 for high-speed, full-speed and low-speed traffic, and port 2 for full-speed and low-speed traffic, transfer controller 130b is idle, and transfer controller 130c is controlling port 3 for high-speed traffic. A table view is provided in Table 2.

TABLE 2

Transfer Controller	Ports and Speed
1 (130a)	Port 1 HS/FS/LS, Port 2 FL/LS
2 (130b)	Idle
3 (130c)	Port 3 HS

Figure 4 depicts three data streams 410a-410b corresponding with three connected devices 420a-420c, respectively. Device 420a is a high-speed/full-speed/low-speed (HS/FS/LS) device meaning that it operates at any of those speeds. Devices 420b-420c are full-speed/low-speed (FS/LS) devices meaning that they operate at any of those speeds.

In this example, transfer controller 130a is controlling port 1 for high-speed, full-speed and low-speed traffic, and ports 2 and 3 for full-speed and low-speed traffic, and transfer controllers 130b and 130c are idle. A table view is provided in Table 3.

TABLE 3

Transfer Controller	Ports and Speed
1 (130a)	Port 1 HS/FS/LS, Port 2 FL/LS, Port 3 FL/LS
2 (130b)	Idle
3 (130c)	Idle

Exemplary applications of the invention's scalable-USB technology include: (a) storage intensive platforms, where multiple USB hard disk are accessible with full 480Mbps bandwidth; (b) multimedia streaming, where streaming data from DVC to hard disk at 60MBytes/s is possible; and (c) software RAID, multiple USB hard disk forms an extremely fast storage device. Additional applications will be apparent to those of skill in the art.

The invention's ability to exceed transfer rates of 480MBytes/s with a plurality of peripheral devices may be beyond the needs of conventional devices using a PCI bus since the PCI interface has a bandwidth of only 133MB/s. However with the introduction of the

PCI-Express interface, with up to 4GBytes/s bandwidth, 640MBytes/s is definitely a goal that should be provided for high performance interface technology.

Exemplary advantages and features of the invention include the following: (a) the invention provides scalable total bandwidth across the USB ports; (b) the invention provides an improved host controller that is able to schedule simultaneous USB transfers; (c) the invention provides a multiple transfer controller unit that manages a single USB transfer; (d) the invention provides a single transaction translator (TT) to control FS/LS USB transfers; (e) the invention provides routing logic to allow a single transaction translator (TT) to provide full-speed/low-speed (FS/LS) for all ports; and (f) the invention provides flexible total bandwidth. Additional advantages will be apparent to those of skill in the art.

As can be seen, the invention provides advantages over simply grouping multiple enhanced host controller interface (EHCI) cores in the same interface circuit. The invention is truly scalable without adding a full complement of resources needed for each EHCI core, thereby saving real estate on devices and inside housings. The invention intelligently combines the sharing of MMU and RAM in an efficient manner. The invention synchronizes the USB ports, while also being able to stream independent data. The invention provides that transactions on different ports can be "time-coupled," which is useful for software RAID features.

Having disclosed exemplary embodiments and the best mode, modifications and variations may be made to the disclosed embodiments while remaining within the subject and spirit of the invention as defined by the following claims.

CLAIMS

What is claimed is:

1. A serial bus interface circuit (100) for use in a computer, comprising: a master scheduler (120) configured to communicate serial data with the computer; a plurality of transfer controllers (130) coupled to the master scheduler and configured to manage serial tranactions; a router (140) having a number of ports coupled to the transfer controllers and configured to selectively communicate serial data between selected ports; and a pluarality of serial interfaces (150) coupled to the router and adapted for coupling to peripheral devices.

2. The serial bus interface circuit of claim 1, further comprising a transaction translator (160) coupled to a transfer controller and a serial interface, and configured to translate the serial data to a normalized speed.

3. The serial bus interface circuit of claim 1, wherein: the router is configured to selectively communicate serial data between the serial interfaces and the transfer controllers.

4. The serial bus interface circuit of claim 2, wherein: the router is configured to selectively communicate serial data between the serial interfaces, the transfer controllers and the transaction translator.

5. The serial bus interface circuit of claim 1, wherein: at least one of the transfer controllers is a low-speed transfer controller and data communication with low-speed peripheral devices is directed to a low-speed transfer controller.

6. The serial bus interface circuit of claim 2, wherein: data communication with low-speed peripheral devices is routed to the transaction translator.

7. The serial bus interface circuit of claim 1, wherein: the master scheduler (120) is coupled to a memory management unit (112) and configured to communicate serial data with the memory management unit.

8. The serial bus interface circuit of claim 2, wherein: the master scheduler (120) is coupled to a memory management unit (112) and configured to communicate serial data with the memory management unit.

9. The serial bus interface circuit of claim 5, wherein: the master scheduler (120) is coupled to a memory management unit (112) and configured to communicate serial data with the memory management unit.

10. The serial bus interface circuit of claim 6, wherein: the master scheduler (120) is coupled to a memory management unit (112) and configured to communicate serial data with the memory management unit.

11. A method of communicating between a serial bus interface and a computer using a master scheduler (120) configured to communicate serial data with the computer; a plurality of transfer controllers (130) coupled to the master scheduler; a router (140) having a number of ports coupled to the transfer controllers; and a pluarality of serial interfaces (150) coupled to the router, comprising the steps of: selectively communicating serial data between the serial interfaces and the transfer controllers via the router.

12. The method of claim 11 further using a transaction translator (160) coupled to a transfer controller and a serial interface, and further comprising the steps of: translating the serial data to a normalized speed.

13. The method of claim 11, wherein at least one of the transfer controllers is a low-speed transfer controller, and further comprising the steps of: directing data communication with low-speed peripheral devices to a low-speed transfer controller.

14. The method of claim 12, further comprising the steps of: routing data communication with low-speed peripheral devices to the transaction translator.

15. The method of claim 11, further comprising the steps of: the master scheduler communicating serial data with a memory management unit.

16. The method of claim 12, further comprising the steps of: the master scheduler communicating serial data with a memory management unit.

17. The method of claim 13, further comprising the steps of: the master scheduler communicating serial data with a memory management unit.

18. The method of claim 14, further comprising the steps of: the master scheduler communicating serial data with a memory management unit.

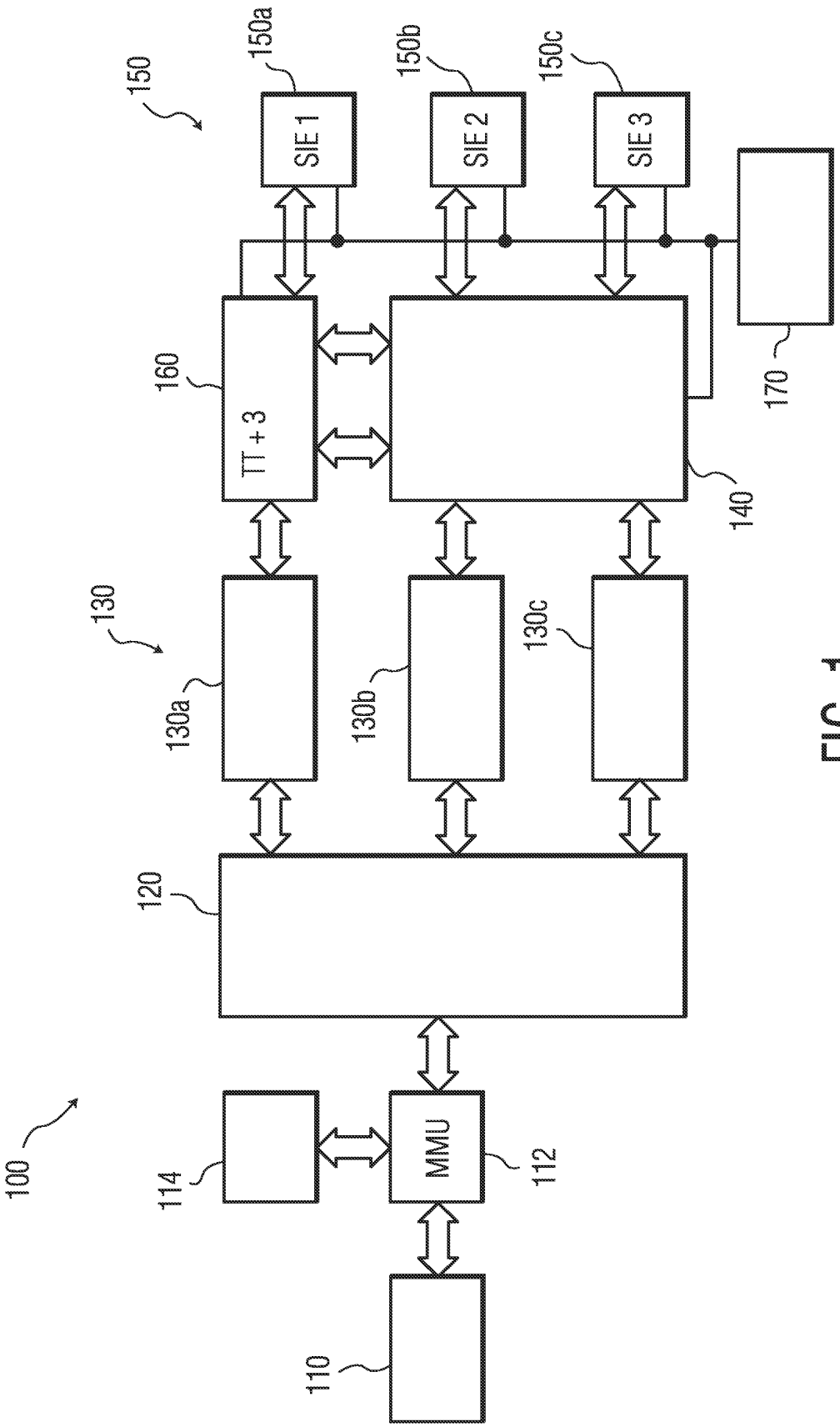

100
110
112
114
120
130
130a
130b
130c
140
150
150a
150b
150c
160
170
MMU
SIE 1
SIE 2
SIE 3
TT + 3

FIG. 1

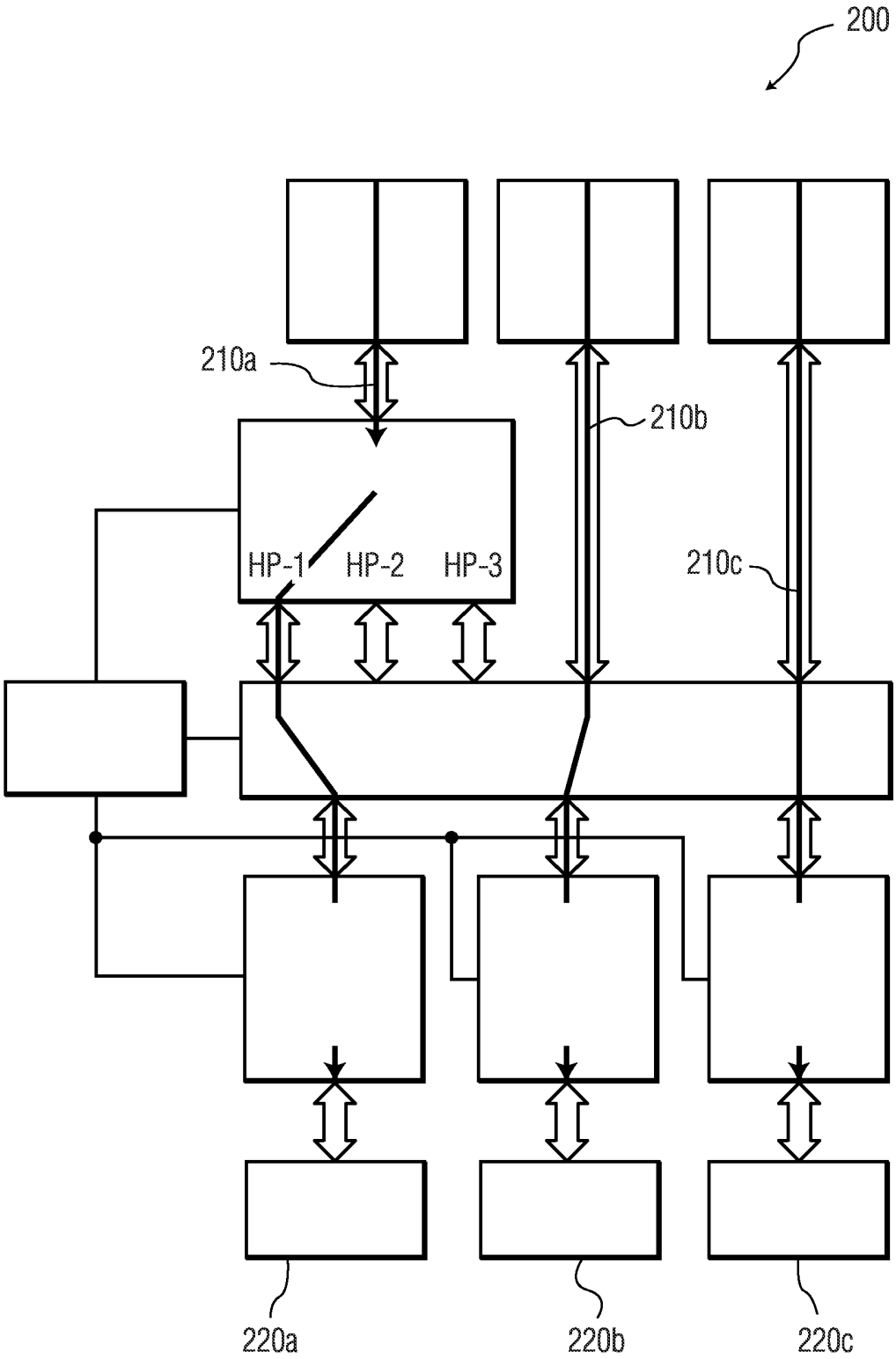


FIG. 2

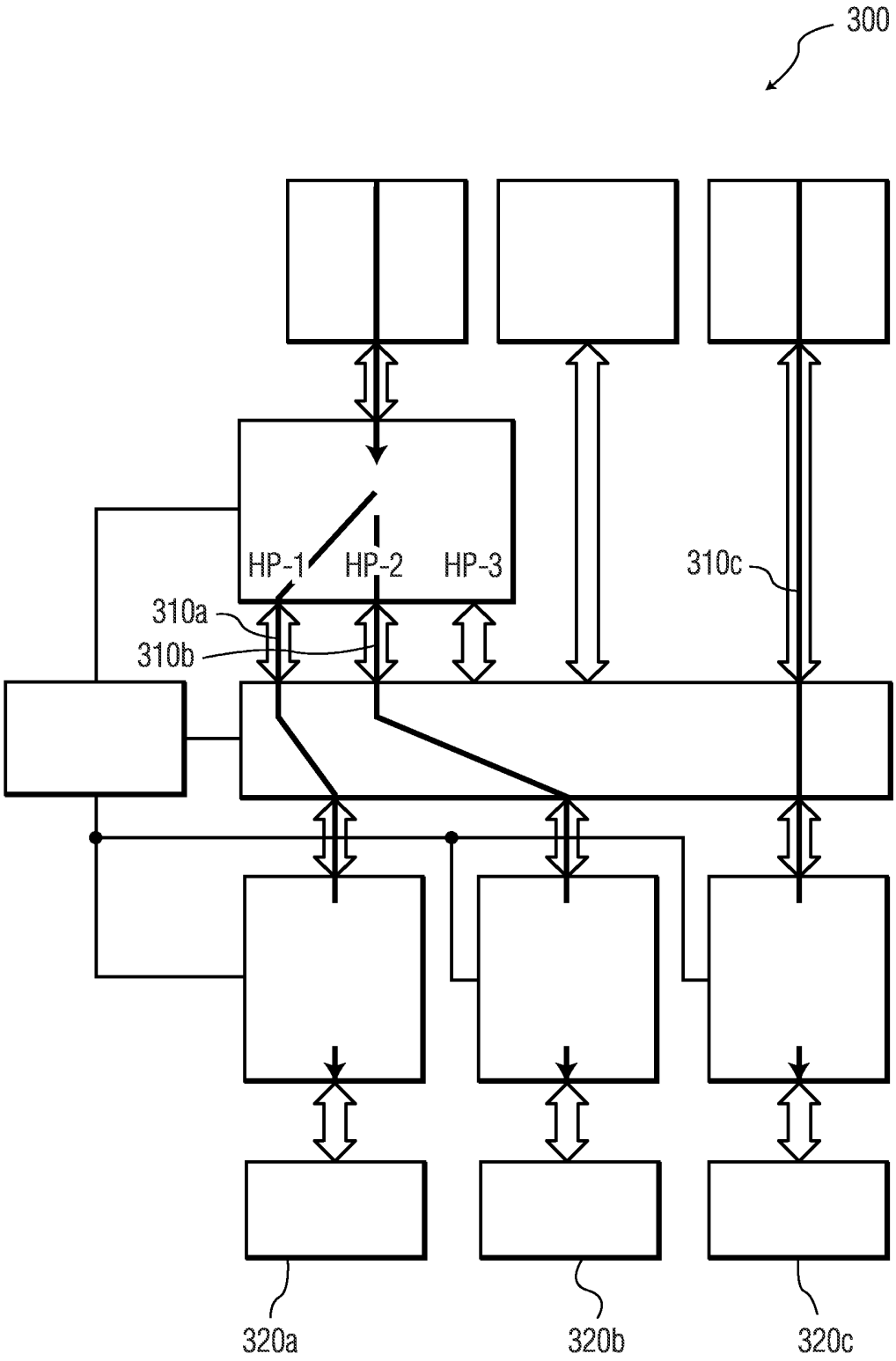


FIG. 3

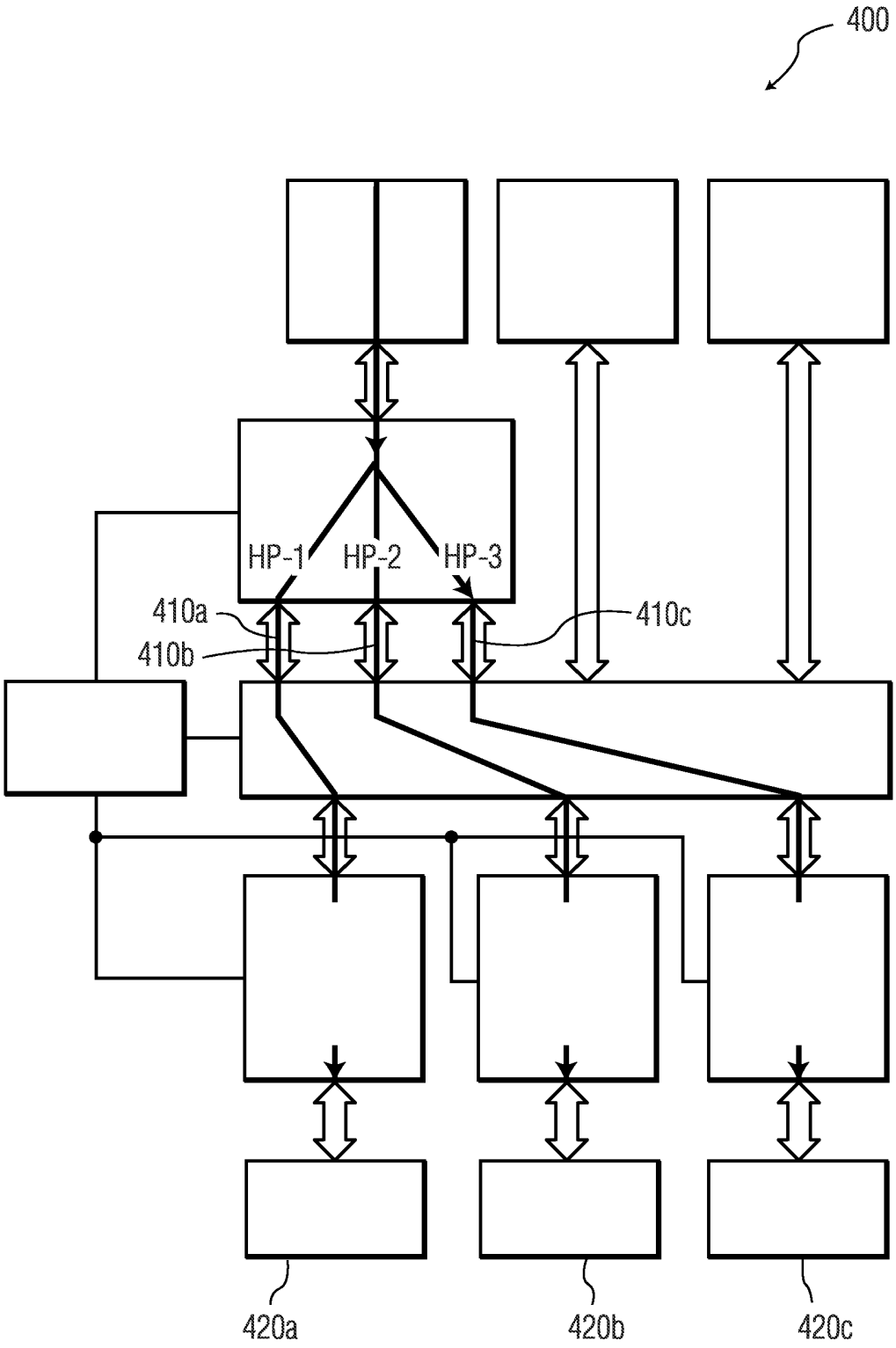


FIG. 4