

(51) International Patent Classification:
H04L 12/56 (2006.01)(21) International Application Number:
PCT/CN2012/078855(22) International Filing Date:
19 July 2012 (19.07.2012)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
201110221835.6 3 August 2011 (03.08.2011) CN(71) Applicant (for all designated States except US): **HANG-ZHOU H3C TECHNOLOGIES CO., LTD** [CN/CN]; Huawei Hangzhou Manufacturing base, 310 lihue road, zhijiang hi-tech park hangzhou, hi-tech industry development zone, Hangzhou, Zhejiang 310053 (CN).

(72) Inventor; and

(75) Inventor/Applicant (for US only): **YANG, Yanjun** [CN/CN]; Room 730, Oriental Electronic BLD., No.2

Chuangye Road, Shangdi Information Industry Base, Haidian district, Beijing 100085 (CN).

(74) Agent: **CHINA PATENT AGENT (HK)LTD.**; 22/F., Great Eagle Center, 23 Harbour Road, Wanchai, Hong Kong (CN).

(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, 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, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ,

[Continued on next page]

(54) Title: LOAD BALANCING IN LINK AGGREGATION

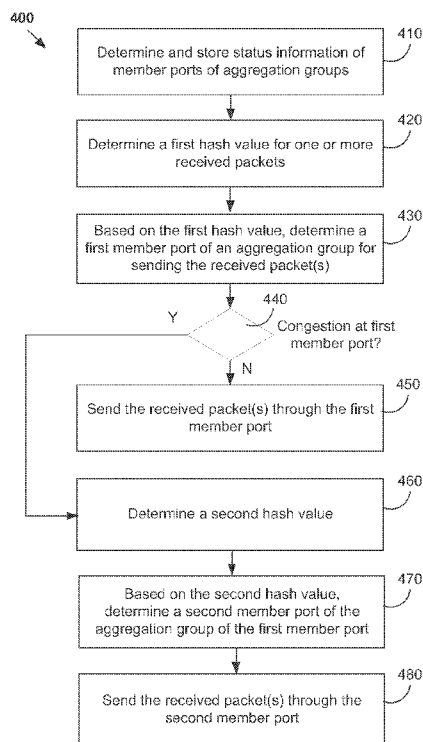


Fig. 4

(57) Abstract: A method for load balancing in a network device supporting link aggregation. A first hash value is determined for a received packet. Based on the first hash value, a first member port of an aggregation group corresponding to a destination address of the packet is determined. If there is congestion at the first member port, a second hash value is determined for the received packet. Based on the second hash value, a second member port in the aggregation group is determined and the received packet is sent through the second member port. A network device and a processor for load balancing are also disclosed.



TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, **Published:**

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

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

Load Balancing in Link Aggregation

Background

The bandwidth of a single port in a network device has increased from 10 Mbps to 10 Gbps in recent years. Despite the rapid increase, the bandwidth of a single port is still inadequate to meet the bandwidth requirements of various network applications. For example, the bandwidth requirement for an uplink port is generally at least 100 Gbps.

Brief Description of Drawings

Non-limiting example(s) will be described with reference to the following drawings, in which:

Fig. 1 is a block diagram of an example network supporting link aggregation;

Fig. 2 is a block diagram of an example network device;

Fig. 3 is a block diagram of the example network device in Fig. 2 during packet transmission;

Fig. 4 is a flowchart of an example method for load balancing in a network device supporting link aggregation;

Fig. 5 is a detailed flowchart of one of the blocks in Fig. 4 for determining and storing status information of ports of a network device;

Fig. 6 is a flowchart of an example structure of a status notification message; and

Fig. 7 is a block diagram of an example structure of a line card of the example network device in Fig. 2 and Fig. 3.

Detailed Description

Fig. 1 shows an example network 100 in which a network device 110 connects source devices 120 to destination devices 130 via multiple network links 140, 150. The source devices 120 are connected to ingress ports 112 of the network device 110, while the destination devices 130 to egress ports 114 of the network device 110. Each physical link 140, 150 terminates at an ingress 112 or egress port 114 of the network device 110.

The network device 110 supports link aggregation, which is also known as port aggregation and trunking. Link aggregation is used to aggregate the network links 140, 150 as logical links 160, 170. Link aggregation increases bandwidth and reliability, and also provides redundancy in case of failure or congestion at one network link or port. The logical links 160, 170 each represent an aggregation group, in which:

- (a) Aggregation groups LAG1 to LAG3 are each formed by network links 140 connecting a source device 120 with the network device 110; and
- (b) Aggregation groups LAG4 to LAG6 are each formed by network links 150 connecting a destination device 130 with the network device 110.

The corresponding ingress 112 or egress 114 ports of each aggregation group are known as their “member ports” or “trunk members”.

- 10 During packet transmission, the network device 110 receives, from a source device 120, packets through one of the member ingress ports 112 of an aggregation group; and determines the egress port 114 through which the packets are sent to the destination device 130. The destination device 130 the packets are sent is not necessarily their final destination.

15

- The network device 110 may be any device for forwarding packets in the network 100, such as a switch, router etc. The source 120 and destination 130 devices may be switches, routers, hosts, computers etc. In one example, the network device 110 may be a core switch connecting multiple access switches (source devices 120) to a router (destination device 140). Although multiple source 120 and destination 130 devices are shown, it will be appreciated that the network device 110 may connect to one source device 120 and/or one destination device 130. The network links 140, 150 may be any suitable links, such as optical fibres, copper wires etc.

- 25 Referring now to Fig. 2, an example network device 110 in the form of a switch 200 will be explained in more detail. The network device 110 includes multiple line cards, such as three ingress line cards (LC1, LC2 and LC3); and three egress line cards (LC4, LC5 and LC6) in this example. The line cards are generally interconnected via internal forwarding paths such as a switching fabric 210. Throughout this disclosure, the term “line card” is used to generally refer to a network interface, network interface card etc. for transmitting and receiving packets, frames, etc.

- Each line card (LC1-LC6) further includes one or more processors that are connected to the ports (labelled Port 1 ... N) on the network device 110. For simplicity, one processor is provided on each line card (LC1-LC6) in the example in Fig. 2. The processors (P1-P6) are interconnected via internal forwarding paths on the network

device 110, such as the switching fabric 210 in Fig. 2. The processors (P1-P6) are also known as “packet processors” and “forwarding chips”.

Referring also to the block diagram 300 in Fig. 3, there are two aggregation groups in the example in Fig. 2:

- (a) Aggregation group LAG1 whose members are port 1 of processor P4, port 1 of processor P5, and port 1 of processor P6; and
- (b) Aggregation group LAG2 whose members are port 2 of processor P1, and port 1 of processor P2.

The ports connect the switch 110 to one or more other devices (120, 140) in the network 110, as explained using Fig. 1. The ports on ingress line cards LC1 and LC2 are known as ingress ports, and ports on egress line cards LC3 and LC4 are known as egress ports.

In the example in Fig. 3, when packets 310 arrive at an ingress port of a line card (e.g. Port 1 of LC2) of the switch 200, the processor P2 determines the appropriate egress line card and egress port (e.g. Port 1 of LC4) through which the received packets are sent.

The appropriate egress line card and egress port are generally determined based on a hash value computed from the received packet(s). As such, it is possible that packets received at port 2 of processor P1 are also forwarded to port 1 of processor P4, possibly causing load imbalance and congestion at port 1 of processor P4.

Fig. 4 is a flowchart of an example method 400 for load balancing in network device 110 supporting link aggregation:

At block 410, the network device 110 determines and stores status information of member ports of each aggregation group.

At block 420, the network device 110 determines a first hash value for one or more received packets.

At block 430, based on the first hash value, the network device 110 determines a first member port of an aggregation group for sending the received packets.

At block 440, the network device 110 determines whether there is congestion at the first member port. In one example, the determination may be based on the status information stored at block 410.

5

At block 450, if the determination at block 440 is not affirmative (no congestion), the network device 110 sends the received packets through the first member port.

10

At block 460, if the determination at block 430 is affirmative (congested), the network device 110 determines a second hash value.

15

At block 470, based on the second hash value, the network device 110 determines a second member port of the aggregation group through which the received packets are sent.

At block 480, the received packets are sent through the second member port instead of the first member port.

20 Using the example method, if there is congestion at the first member port, traffic or load is diverted from a first member port to a second member port of an aggregation group. This reduces load imbalance among the redundant links, thereby reduces link resource wastage. The example method will be explained in more detail below.

25 Port status information

At block 410 in Fig. 4, the network device 110 determines and stores status information of member ports of each aggregation group configured on the network device 110.

30 An example of this process 410 is shown in Fig. 5, in which each processor (P1-P6) on the network device 110 determines the status of their physical ports and notifies other processors accordingly.

(i) At block 412 in Fig. 5, the processors (P1-P6) on the network device 110 each determine and store status information of the ports on the network device 110. The status information sets out the relationship between member ports and aggregation groups, and the status of each member port.

35

In one implementation, the status information may be in the form of a forwarding table, an example of which is Table 1.

Table 1: Initial port status information

5

Aggregation group ID	Number of member ports	Member port Information	Congestion indicator
1	3	(Device 4, Port 1)	0
		(Device 5, Port 1)	0
		(Device 6, Port 1)	0
2	2	(Device 1, Port 2)	0
		(Device 2, Port 1)	0

The status information includes the identifier of each aggregation group; number of member ports in each aggregation group; the identifier of each member port; and a congestion indicator of each member port. The identifier of each member port of an aggregation group includes an identifier of the processor ("Device") connected to the member port, and an identifier of the port.

In this example, there are three member ports in aggregation group 1: port 1 of processor P4 ("Device 4") or; port 1 of P5 ("Device 5"); and port 1 of P6 ("Device 6"). The congestion indicator, or "congestion factor", is set to an initial value of zero to indicate no congestion.

It will be appreciated that the port status information may be stored in other suitable form(s), such as using an individual forwarding table for each aggregation group.

20

(ii) At block 414 in Fig. 5, the processors (P1-P6) determine the status of their respective ports, and notify other processors of the network device 110 accordingly. The notification may be in the form of a status notification message shown in Fig. 6 that includes:

25

- (a) A 10-bit sender identifier, which identifying the processor that sends the status notification message.
- (b) A 64-bit port status information or port congestion state, where each bit represents the status of a port connected to the processor. Value 0

represents no congestion, whereas value 1 represents the detection of congestion.

(c) A 54-bit reserved value that is currently not in use.

- 5 The status notification message may be broadcasted to other processors on the network device 110 via the switching fabric 210. The status may be determined periodically, and/or when triggered by a trigger signal received from other processors on the network device 110 or other nodes in the network.
- 10 In the example in Fig. 3, processor P4 of line card LC4 determines the status of its ports (I to N) and detects congestion at port I (see label 320). P4 then generates a status notification message with the first bit set to 1, which represents congestion at port I . The status notification message is sent to other processors (i.e. P1-P3 and P5-P6) on the network device 110.
- 15 (iii) At block 416 in Fig. 5, a processor that receives the status notification message reporting congestion at one or more member ports, and updates its port status information accordingly.
- 20 In the example in Fig. 3, processor P2 receives a status notification message from processor P4. From the sender identifier and port status information of the message, processor P2 learns that there is congestion at port 1 of processor P4. Processor P2 then retrieves its copy of the status information of ports on the network device 110 from a memory on line card LC2.
- 25 Processor P2 searches for the entry in Table 1 that is associated with port I of P4, and updates its congestion indicator from a value that indicates no congestion (zero) to a new value that indicates otherwise. The new value may be in any suitable form, such as a random value determined by the processor P2. The updated status information is
- 30 shown in Table 2, in which the congestion indicator of Port 1 of P4 is set to a random value 0.52.

Table 2: Updated status information

Aggregation group ID	Number of member ports	Member port Information	Congestion indicator
----------------------	------------------------	-------------------------	----------------------

1	3	(Device 4, Port 1)	0.52
		(Device 5, Port 1)	0
		(Device 6, Port 1)	0
2	2	(Device 1, Port 2)	0
		(Device 2, Port 1)	0

Once port 1 of processor P4 recovers from congestion, processor P4 similarly sends a status notification message to other processors on the network device 110. This time, the bit that represents port 1 of processor P4 in the 64-bit port status information is set to zero instead of a one.

Upon receiving the status notification message from processor P4, processor P2 update the status information of port 1 of processor P4, from the random value to a value that represents no congestion, i.e. zero in this example.

First hash value

At block 420 in Fig. 4, a processor determines a first hash value for a packet received through an ingress port of the network device 110. The first hash value may also be determined for a packet flow, but for simplicity, a packet is used here.

In the example in Fig. 3, processor P2 receives a packet 310 through ingress port 1. After ingress processing, since the destination address of the received packet corresponds to an aggregation group (or aggregated links), processor P2 performs hash computation on the received packet to determine the first hash value.

First member port

At block 430 in Fig. 4, based on the first hash value, the processor determines a first member port of an aggregation group for sending the received packet.

In the example in Fig. 3, processor P2 determines the number of member ports of the aggregation group, and determine the first member port based on the hash value and number of member ports:

$$\text{Index1} = \text{First Hash Value} \% \text{Number of Member Ports},$$

where Index1 (or Trunk Select Index1) is the index of the first member port in the aggregation group; First Hash Value (or Trunk Hash Value) is the hash value of the

received packet; and Number of Member Ports (or Number of Trunk Members) is the number of members in the aggregation group.

Congestion determination

- 5 At block 440 in Fig. 4, the processor determines whether there is congestion at the first member port.

In one example, the determination is based on the status information stored at block 410 in Fig. 4. If the value of the congestion indicator of the first member port is zero,
10 this indicates no congestion at the first member port. But otherwise, if the congestion indicator has a random value, this indicates that there is congestion at the first member port.

At block 450 in Fig. 4, if there is no congestion at the first member port, the received
15 packet is sent through the first member port.

Second hash value

At block 460 in Fig. 4, if there is congestion at the first member port, the processor determines a second hash value. In one example implementation, the second hash
20 value is determined based on the first hash value and an “additional value”.

The congestion indicator of the first member port, which has a random value if there is congestion, may be used as the “additional value”. It will be appreciated that adding the additional value to the first hash value allows the same hash computation algorithm
25 to be used, causing less disturbance compared to changing the algorithm.

In the example in Fig. 3, processor P2 retrieves Table 2 from its memory to determine whether there is congestion at the first member port (i.e. port 1 of processor P4). Processor P2 searches for the entry that corresponds to the first member port and its
30 aggregation group.

In Table 2, the first member port is port 1 of processor P4 of aggregation group ID 1, its congestion indicator has a random value of 0.52. This value is used to determine the second hash value, such as by adding the random value to the first hash value, as
35 follows:

$$\text{Second Hash Value} = \text{First Hash Value} + \text{Random Value}$$

Second member port

At block 470 in Fig. 4, based on the second hash value, the processor determines a second member port of the same aggregation group through which the packet is sent
5 from the network device 110.

In the example in Fig. 3, processor P2 determines the second member port as follows:

Index2 = (First Hash Value + Random Factor) % Number of Member Ports,
10

where Index2 (or Trunk Select Index2) is the index of the second member port in the aggregation group of the first member port; First Hash Value (or Trunk Hash Value) is the hash value of the received packet; Random Factor is a random value; and Number of Member Ports (or Number of Trunk Members) is the number of members in the
15 aggregation group.

At block 480 in Fig. 4, the received packet is sent through the second member port instead of the first member port, thereby diverting the traffic away from the first member port to reduce the likelihood of further aggravating the congestion.
20

It will be appreciated that blocks 460 and 470 may be repeated, if the second member port is also congested based on the status information of the second member port. In this case, a third hash value is calculated to determine a third member port through which the received packet is sent, and so on.
25

Network Device 110

Fig. 7 shows a block diagram of an example line card 700 in the example network device 110 in Fig. 2 and Fig. 3.

30 The line card 700 includes one or more processors 710 (labelled P1_a to P1_n) that are each connected to a subset of ports 720 (also labelled *a* to *n*). The processors 710 are interconnected to each other via internal paths 750, and connected to a central processing unit (CPU) 730 and memory 740.

35 Each processor 710 may be connected to any number of ports 320, and this number may vary from one processor 310 to another. An aggregation group may include ports

720 from the same processor 310 or different ones, as well as ports on the same line card 700 or different line cards of the network device 110.

The CPU 730 programs the processors 710 with machine-readable instructions 742 to
5 analyse a packet received at the line card 300, and determines an egress port of the network device 110 to forward the packet flow. The machine-readable instructions 742 are stored in the memory 740. Other information required for load balancing, such as the status information in Table 1 and Table 2, is also stored in the memory 740.

10 The methods, processes and functional units described herein may be implemented by hardware (including hardware logic circuitry), software or firmware or a combination thereof. The term 'processor' is to be interpreted broadly to include a processing unit, ASIC, logic unit, or programmable gate array etc. The processes, methods and functional units may all be performed by the one or more processors 710; reference in
15 this disclosure or the claims to a 'processor' should thus be interpreted to mean 'one or more processors'.

Further, the processes, methods and functional units described in this disclosure may be implemented in the form of a computer software product. The computer software
20 product is stored in a storage medium and comprises a plurality of instructions for making a processor to implement the methods recited in the examples of the present disclosure.

The figures are only illustrations of an example, wherein the units or procedure shown
25 in the figures are not necessarily essential for implementing the present disclosure. Those skilled in the art will understand that the units in the device in the example can be arranged in the device in the examples as described, or can be alternatively located in one or more devices different from that in the examples. The units in the examples described can be combined into one module or further divided into a plurality of sub-
30 units.

Although the flowcharts described show a specific order of execution, the order of execution may differ from that which is depicted. For example, the order of execution of two or more blocks may be changed relative to the order shown. Also, two or more
35 blocks shown in succession may be executed concurrently or with partial concurrence. All such variations are within the scope of the present disclosure.

It will be appreciated that numerous variations and/or modifications may be made to the processes, methods and functional units as shown in the examples without departing from the scope of the disclosure as broadly described. The examples are, therefore, to
5 be considered in all respects as illustrative and not restrictive.

Claims

1. A method for load balancing in a network device supporting link aggregation, the method comprising a first processor of the network device:
 - 5 determining a first hash value for a received packet;
based on the first hash value, determining a first member port of an aggregation group corresponding to a destination address of the packet;
if there is congestion at the first member port, determining a second hash value for the received packet; and
 - 10 based on the second hash value, determining a second member port in the aggregation group and sending the received packet through the second member port.
2. The method of claim 1, wherein the second hash value for the received packet is determined based on the first hash value and an additional value.
- 15 3. The method of claim 2, wherein the additional value is a congestion indicator of the first member port, the congestion indicator having a random value if there is congestion at the first member port.
- 20 4. The method of claim 3, further comprising:
receiving, from a second processor connected to the first member port of the aggregation group, a status notification message reporting congestion at the first member port; and
updating the congestion indicator of the first member port from a value that
25 represents no congestion to the random value.
5. The method of claim 3 or 4, further comprising:
receiving, from the second processor connected to the first member port of the aggregation group, a status notification message reporting recovery of the first member
30 port from congestion; and
updating the congestion indicator of the first member port from the random value to a value that represents no congestion.
6. The method of claim 4 or 5, wherein the status notification message includes
35 information identifying the second processor, and status of each port connected to the second processor.

7. The method of any one of the preceding claims, wherein the second port member is determined based on the second hash value and number of members in the aggregation group.
- 5 8. A network device for load balancing, the network device supporting link aggregation and comprising a first processor to:
- determine a first hash value for a received packet;
 - based on the first hash value, determine a first member port of an aggregation group corresponding to a destination address of the packet;
 - 10 if there is congestion at the first member port, determine a second hash value for the received packet; and
 - based on the second hash value, determine a second member port in the aggregation group and send the received packet through the second member port.
- 15 9. The network device of claim 8, wherein the second hash value for the received packet is determined based on the first hash value and an additional value.
10. The network device of claim 8 or 9, wherein the additional value is a congestion indicator of the first member port, the congestion indicator having a random value if
- 20 there is congestion at the first member port.
11. The network device of claim 10, wherein the first processor is further to:
- receive, from a second processor connected to the first member port of the aggregation group, a status notification message reporting congestion at the first
 - 25 member port; and
 - update the congestion indicator of the first member port from a value that represents no congestion to the random value.
12. The network device of claim 10 or 11, wherein the first processor is further to:
- 30 receive, from a second processor connected to the first member port of the aggregation group, a status notification message reporting recovery of the first member port from congestion; and
- update the congestion indicator in the status information of the first member port from the random value to a value that represents no congestion.
- 35

13. The network device of claim 11 or 12, wherein the status notification message includes information identifying the second processor, and status of each port connected to the second processor.
- 5 14. The network device of any one of claims 8 to 13, wherein the second port member is determined based on the second hash value and number of members in the aggregation group.
- 10 15. A processor for load balancing in a network device supporting link aggregation, the processor is to:
- determine a first hash value for a received packet;
 - based on the first hash value, determine a first member port of an aggregation group corresponding to a destination address of the packet;
 - if there is congestion at the first member port, determine a second hash value for
- 15 the received packet; and
- based on the second hash value, determine a second member port in the aggregation group and send the received packet through the second member port.

1/7

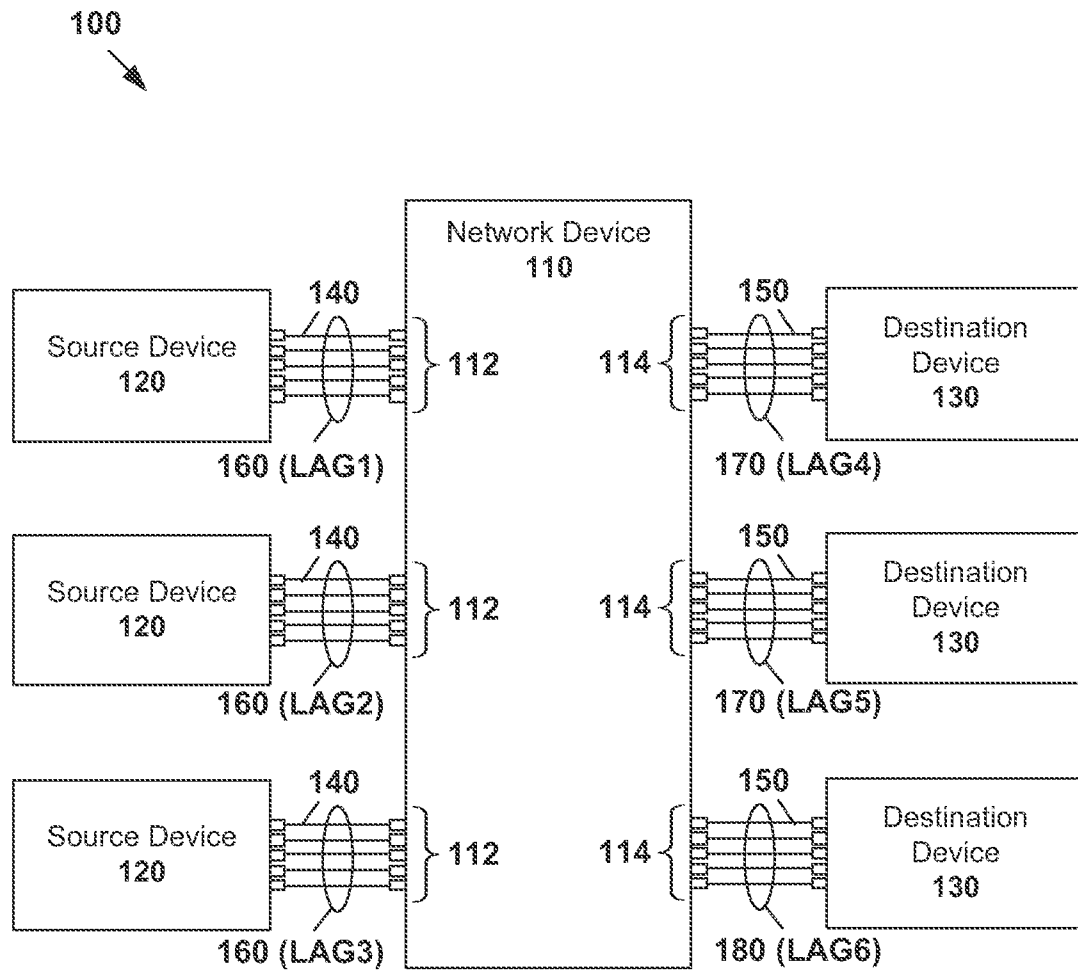


Fig. 1

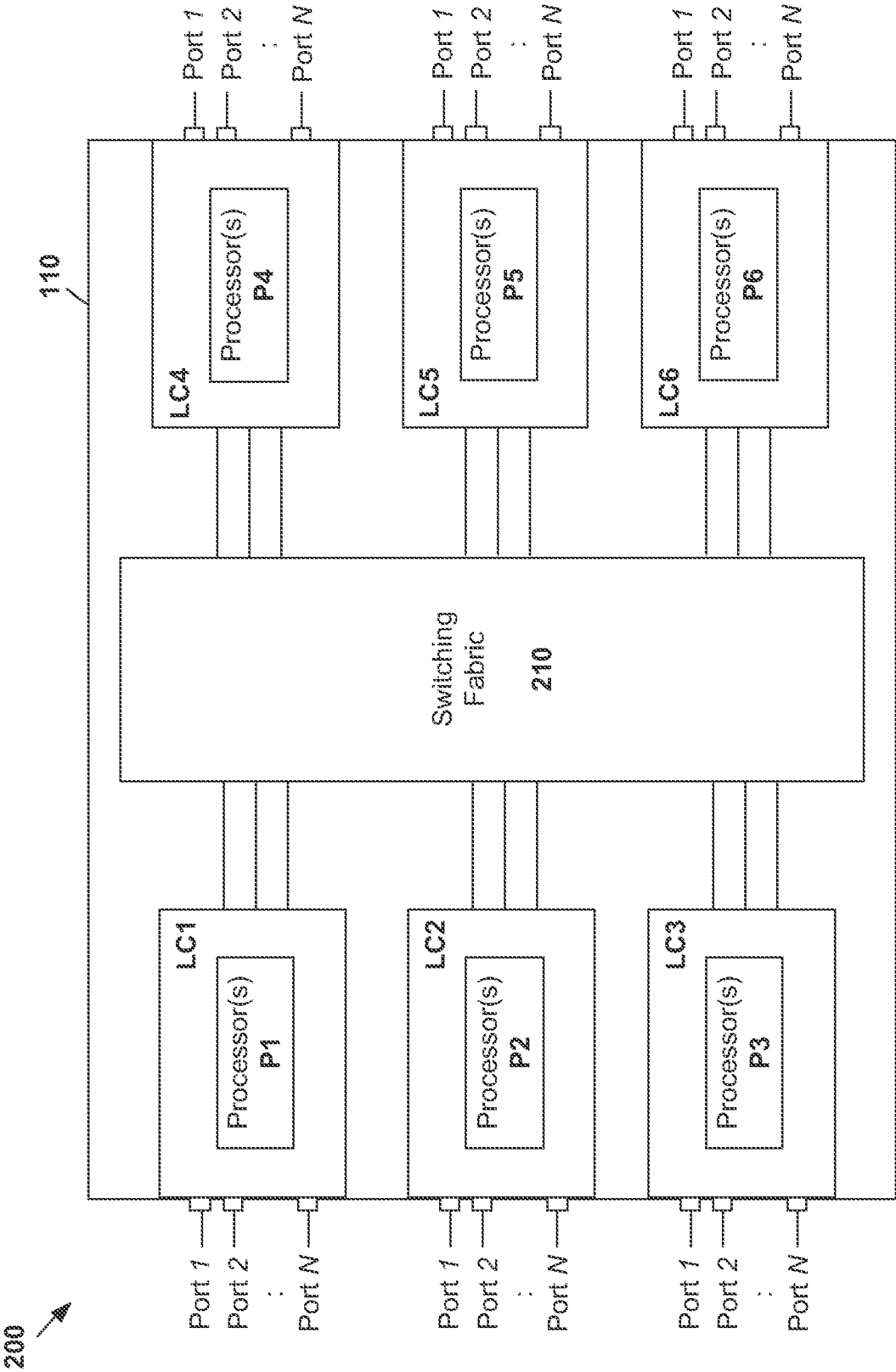


Fig. 2

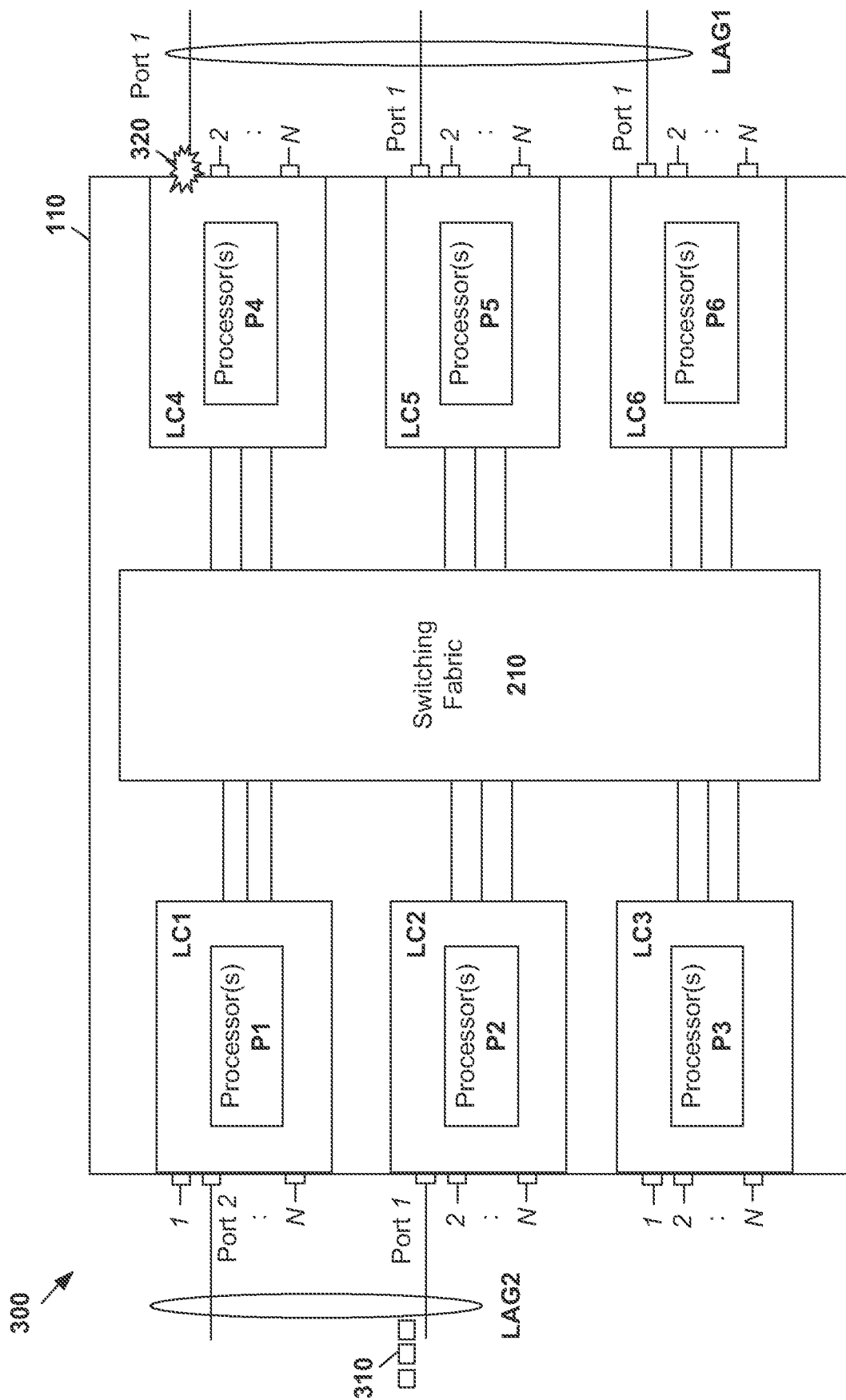


Fig. 3

4/7

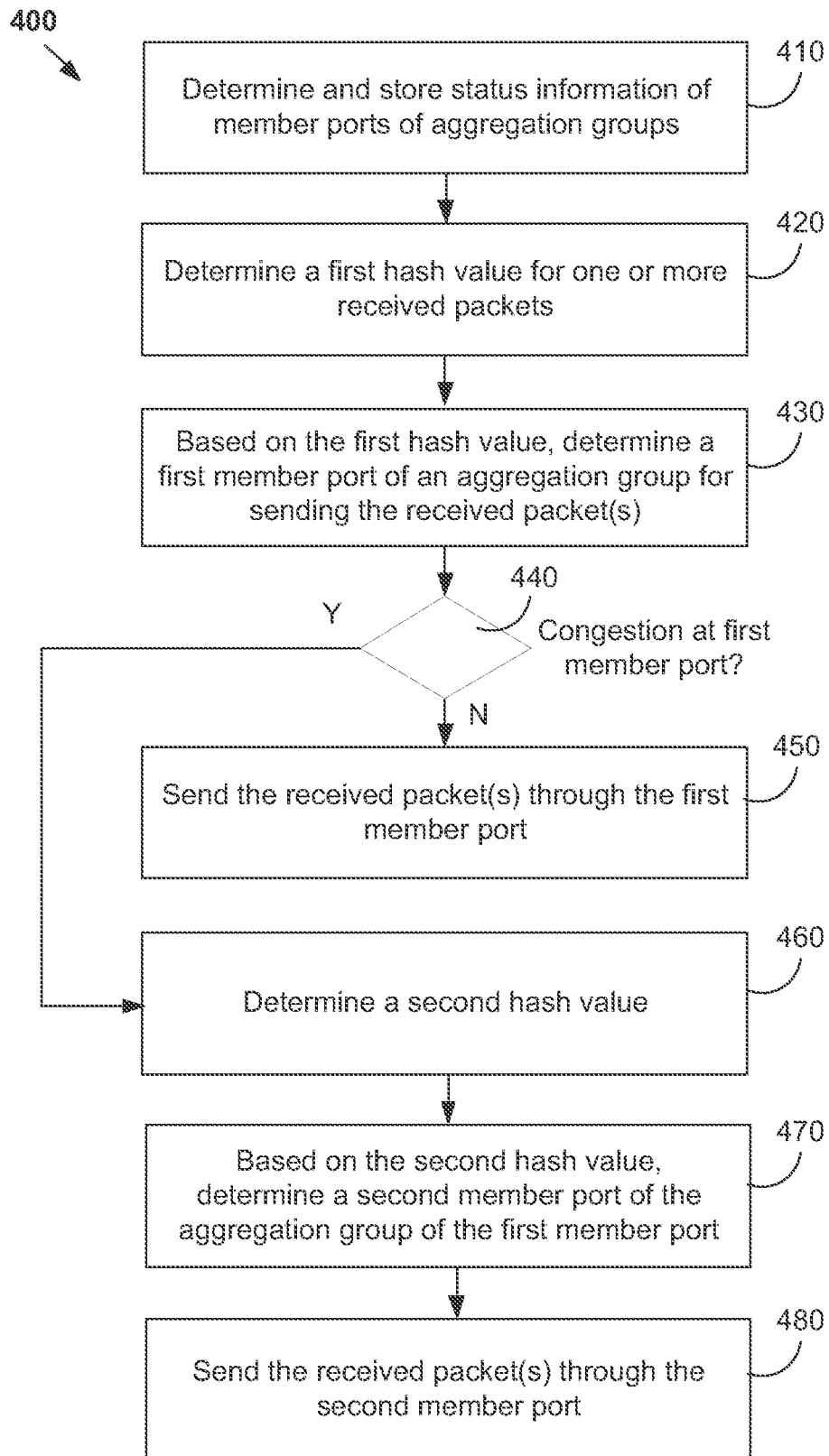


Fig. 4

5/7

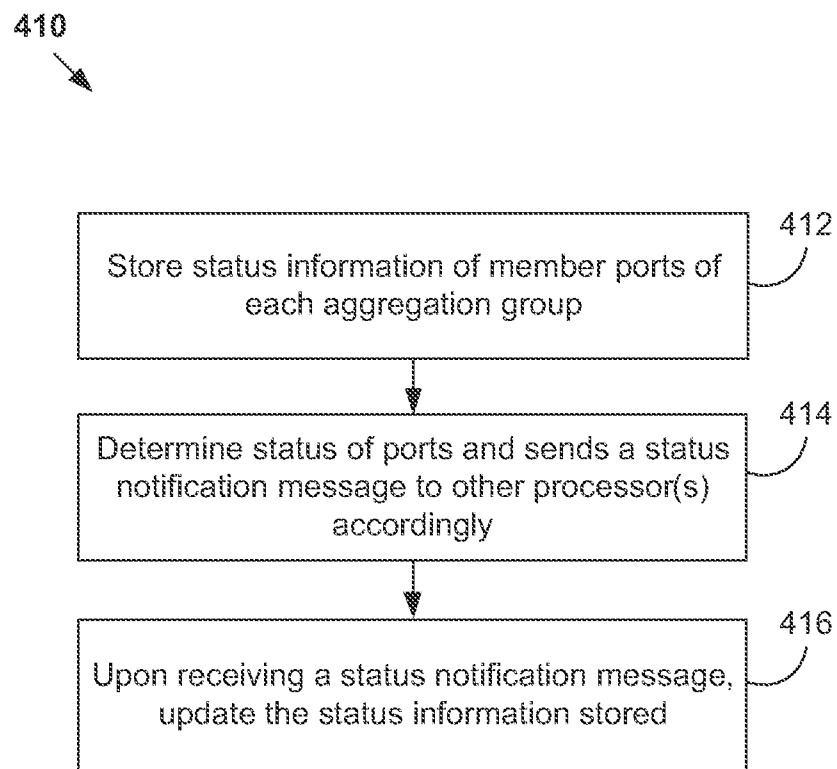


Fig. 5

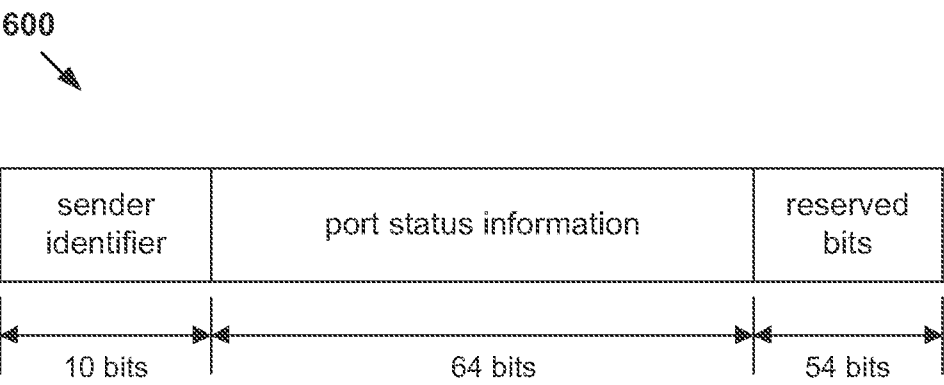


Fig. 6

7/7

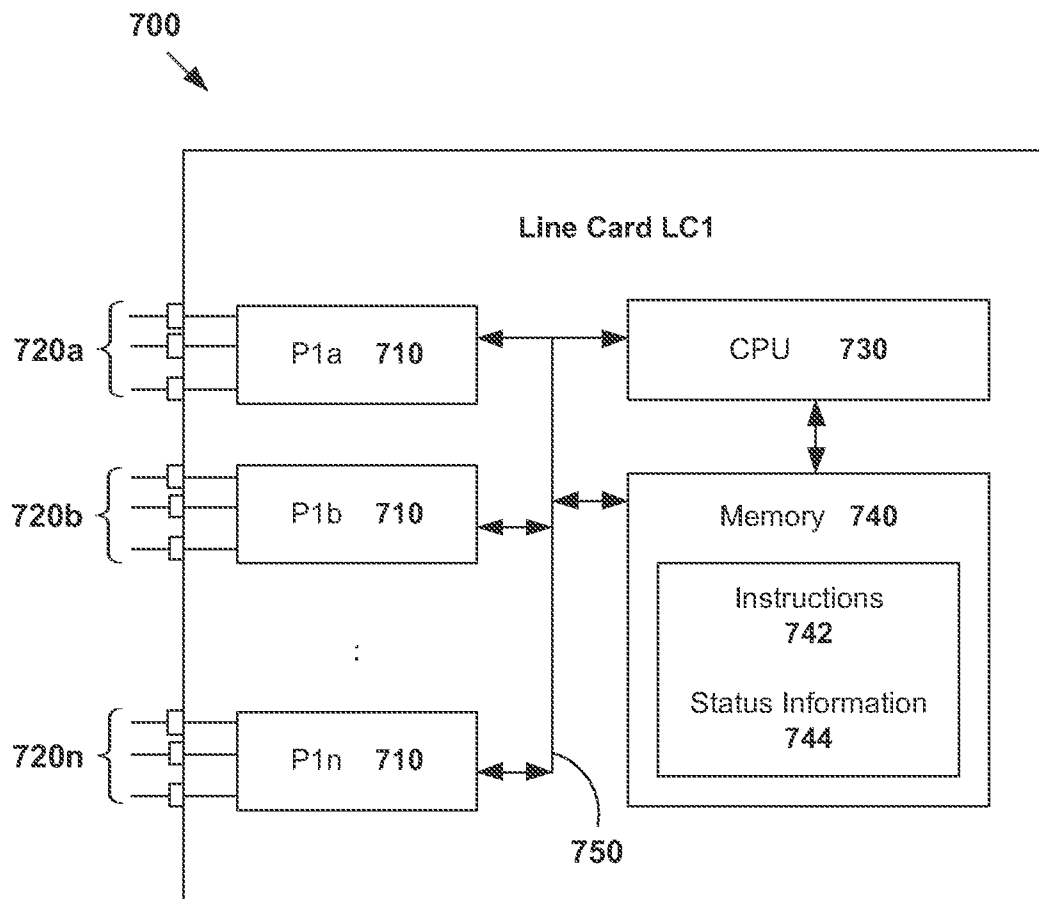


Fig. 7

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2012/078855

A. CLASSIFICATION OF SUBJECT MATTER

H04L12/56(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC:H04L

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)

CNABS,CNTEXT,CNKI,VEN: link w aggregation, load 1w balance+, flow, hash, congest+, link, aggregate+, allocat+, select+, port?

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
PX	CN102263697A(HANGZHOU H3C TECHNOLOGIES CO) 30 Nov. 2011(30.11.2011) the description paragraph [0026]-[0069], claims 1-12 and figures 1-3	1-15
	CN1622532A (HUAWEI TECHNOLOGIES CO LTD) 01 Jun. 2005(01.06.2005)	
Y	the claims 1-2	1,8,15
A	the whole document	2-7,9-14
	CN1571354A(HUAWEI TECHNOLOGIES CO LTD) 26 Jan. 2005(26.01.2005)	
Y	the description page 1 line 19 to page 2 line 7	1,8,15
A	the whole document	2-7,9-14
A	CN1809021A(HUAWEI TECHNOLOGIES CO LTD) 26 Jul. 2006(26.07.2006)	1-15
	the whole document	

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:	“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
“A” document defining the general state of the art which is not considered to be of particular relevance	“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
“E” earlier application or patent but published on or after the international filing date	“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
“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)	“&” document member of the same patent family
“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	

Date of the actual completion of the international search 26 Sept. 2012(26.09.2012)	Date of mailing of the international search report 18 Oct. 2012 (18.10.2012)
Name and mailing address of the ISA/CN The State Intellectual Property Office, the P.R.China 6 Xitucheng Rd., Jimen Bridge, Haidian District, Beijing, China 100088 Facsimile No. 86-10-62019451	Authorized officer XUE, Yu Telephone No. (86-10)62412008

International application No.
PCT/CN2012/078855

Form PCT/ISA /210 (patent family annex) (July 2009)