

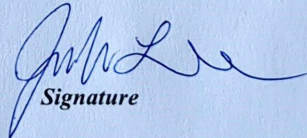


DECLARATION OF TRANSLATOR

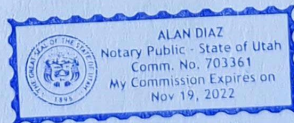
I, Jacob Lee, hereby declare as follows:

1. I personally prepared the attached translation from Japanese into English of the document that is entitled “パケット転送方法、パケット転送システム及びパケット転送装置” in Japanese and “Packet Forwarding Method, Packet Forwarding System, and Packet Forwarding Device” in English (also attached).
2. The attached translation is, to the best of my knowledge, a true, full and accurate translation of the attached Japanese document.
3. I am competent to perform this translation because I have nearly eight years of experience as a professional Japanese-to-English patent translator.
4. I declare under the penalty of perjury under the laws of the United States that the foregoing is true and correct; and further declare that I am aware that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 1001 of Title 18 of the United States Code.

Dated: Sep. 20, 2022


Signature

ACKNOWLEDGEMENT	
State of	<u>Utah</u>
County of	<u>Utah</u>
On	<u>Sep 20, 2022</u> <u>Jacob Lee</u> personally appeared before me <u>Alan Diaz</u>
	whom I know personally
☒	whose identity I verified on the basis of <u>2</u>
☐	whose identity I verified on the oath or affirmation of
	, a credible witness, to be the signer of the above and he/she acknowledged that he/she signed it.
Notary Public	
My Commission Expires:	



15 W. 37th Street 8th Floor
New York, NY 10018
212.581.8870
ParkIP.com

(12) Unexamined Patent Application Publication (A)

(19) Japan Patent Office (JP)

(11) Patent Application Publication No.

JP 2005-86668 A

(P2005-86668A)

(43) Publication Date: Mar. 31, Heisei 17 (2005.3.31)

(51) Int. Cl. ⁷	FI	Theme Code (Reference)
H04L 12/56	H04L 12/56 100Z	5K030

Examination Request: Not Made No. of Claims: 5 OL (20 Pages Total)

(21) Filing No. JP 2003-318516 (P2003-318516)

(22) Filing Date Sep. 10, Heisei 15 (2003.9.10)

(71) Applicant 000005223
Fujitsu Ltd.
4-1-1 Kamikodanaka, Nakahara-ku,
Kawasaki-shi, Kanagawa-ken

(74) Agent 100070150
Tadahiko Itō, Patent Attorney

(72) Inventor Tōru Ishimori
c/o Fujitsu Digital Technology Co. Ltd.
2-3-9 Shin'yokohama, Kōhoku-ku,
Yokohama-shi, Kanagawa-ken

(72) Inventor Takashi Kadosawa
c/o Fujitsu Digital Technology Co. Ltd.
2-3-9 Shin'yokohama, Kōhoku-ku,
Yokohama-shi, Kanagawa-ken

Continued on Final Page

(54) [Title of Invention] Packet Forwarding Method, Packet Forwarding System, and Packet Forwarding Device

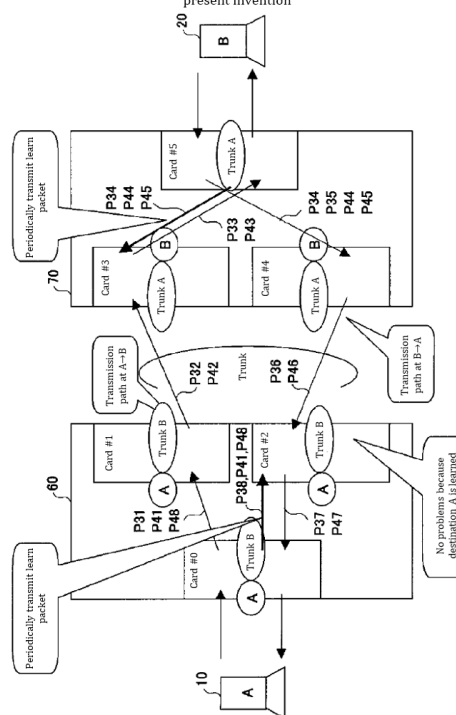
(57) [Abstract]

[Problem] An object is to provide, as a packet forwarding system that uses a path learning function applying a link aggregation function, a system that can prevent repeated flooding even when paths differ between a time of transmission and a time of reception.

[Resolution Means] A configuration is adopted of periodically sending, according to predetermined conditions, a learn packet to all nodes having a trunk.

[Selected FIG.] FIG. 15

Block diagram for describing operations of packet forwarding system of embodiment of present invention



[Scope of Patent Claims]

[Claim 1]

A packet forwarding method for forwarding a receive packet according to a destination address thereof, comprising:

a learning step of learning, at each device performing packet forwarding, a source address of the receive packet in correspondence with a receive path thereof,

each device being configured so for the destination address of the receive packet, when a learning result is had regarding a correspondence relationship between this address and a corresponding path, this receive packet is sent through this path and, meanwhile, when no learning result is had, this receive packet is sent through all paths; and

10 a learn-packet generation step of generating a learn packet for each device at a predetermined timing.

[Claim 2]

The method of claim 1, further comprising: a deletion step of deleting the learning result of the learning step according to a predetermined deletion cycle; wherein

a cycle whereby the learn packet can be generated at the learn-packet generation step is set to be shorter than the predetermined deletion cycle.

[Claim 3]

20 The method of claim 1 or 2, wherein in the learn-packet generation step, for the destination address of the receive packet, the path leading to this destination is a virtual aggregate path provided by applying link aggregation; the learn-packet generation step is configured to be carried out for a situation wherein no learning result is had regarding the correspondence relationship between this address and the corresponding path; and the learn-packet generation step is configured to generate at this time a learn packet for all devices having this virtual aggregate path.

[Claim 4]

A packet forwarding system for forwarding a receive packet according to a destination address thereof, comprising:

a learning means of each device performing packet forwarding learning a source address of the receive packet in correspondence with a receive path thereof; wherein

30 each device is configured so for the destination address of the receive packet, when a learning result is had regarding a correspondence relationship between this address and a corresponding path, this receive packet is sent through this path and, meanwhile, when no learning result is had, this receive packet is sent through all paths, and

each device further has a learn-packet generation means of generating a learn packet for another device at a predetermined timing.

[Claim 5]

A packet forwarding device for forwarding a receive packet according to a destination address thereof, comprising:

40 a learning means of learning a source address of the receive packet in correspondence with a receive path thereof,

a configuration being such that for the destination address of the receive packet, when a learning result is had regarding a correspondence relationship between this address and a corresponding path, this receive packet is sent through this path and, meanwhile, when no learning result is had, this receive packet is sent through all paths; and

a learn-packet generation means of generating a learn packet for another packet forwarding device at a predetermined timing.

[Detailed Description of Invention]

[Field of Art]

[0001]

The present invention relates to a packet forwarding method, a packet forwarding system, and a packet forwarding device. It particularly relates to a packet forwarding method, a packet forwarding system, and a packet forwarding device wherein applied is a system of learning a source address of a receive packet in correspondence with a receive path thereof and, when a packet is received with this source address as a destination address, deciding, based on the learning result, a forwarding path to apply.

[Background Art]

[0002]

10 As a packet forwarding method via a network, there is a method of learning a source MAC address of a receive IP packet together with a receive path thereof and, when an IP packet whose destination address is the same MAC address is received, using the corresponding learning result to decide a transmission path thereof. In this situation, when forwarding, at a node that relays IP packets, an IP packet having an unlearned MAC address as a destination address, a transmission path thereof cannot be specified. As such, in this situation, a broadcast transmission to all nodes (that is, "flooding") is performed. Then, at this time, path learning is carried out by storing the source MAC address (SA) had by this IP packet and a port that received such in a buffer had by each node.

[0003]

20 FIG. 1 and FIG. 2 illustrate this. Note that in the example described below that is illustrated in FIG. 1 and the like, a configuration is supposed wherein each node—50-0, 50-1, 50-2—has one communication card—#0, #1, #2—respectively. When node 50-0 receives an IP packet from terminal A 10 at port #0 of card #0 (steps S1, P1), in a MAC table had by this card, "A", which is a source MAC address (SA) thereof, is learned by being stored in a predetermined buffer in association with information on the receiving card #0 and port #0 at this time (path information) (step S2). Note that in this situation, when a learning result for the same MAC address already exists in the MAC table, this information is overwritten.

[0004]

30 Next, for a destination address (DA; here, MAC address "B") had by this receive IP packet, it is searched whether there is a learning result in the MAC table of the local device (step S3). Then, when the search result is a mishit ("No"), this IP packet is broadcast (flooded) to all nodes (in this example, node 50-1 and node 50-2) (steps S5, P2). Meanwhile, when, as a result of the search at step S3, a learning result relating to the corresponding destination MAC address exists in the buffer ("Yes"), this IP packet is transmitted according to a forwarding path—that is, a card number and port number—included in this learning result (step S4). In this situation, there is no need for flooding.

[0005]

40 Note that the same learning operations are also performed when this IP packet sent from node 50-0 reaches card #2 of node 50-2, which the IP packet passes through on its way to a corresponding destination terminal B 20. As a result, as illustrated in FIG. 1, source path information at this time—that is, the information on card #0, port #0 of node 50-0—is stored as a learning result in association with the MAC address "A" of the source terminal 10 in the buffer (MAC table) of this card #2.

[0006]

50 In this manner, by storing in association with each other (that is, learning) the source MAC address (SA) and the receive path information (card number, port number) once the IP packet is received, the next time an IP packet whose destination (DA) is the same MAC address is received, the forwarding path can be decided by using this learning result. As a result, as long as a learning result relating to the destination MAC address (DA) of the receive packet exists in the buffer, there is no need for flooding and traffic in the connected LAN can be kept to a minimum, since the

forwarding path to apply to the corresponding packet can be decided unambiguously by using this information.

[0007]

That is, in FIG. 3, when an IP packet from terminal B 20 is received at card #2, port #1 of node 50-2 (steps S1, P4), receive path information (card #0, port #0) is already stored in association with the MAC address of terminal A as the learning result of when an IP packet was relayed from terminal A 10 the previous time as above (see FIG. 1) (steps P2, P3). As such, based on this learning result, a hit is obtained for "A", the destination address (DA) of this IP packet ("Yes" at step S3). In this situation, the corresponding path—that is, card #0, port #0—is applied to this IP packet based on this hit result, and this IP packet is forwarded from node 50-2 to card #0 of node 50-0 (step P5). Moreover, in this situation, card #0 of node 50-0 has the learning result relating to the MAC address "A" that is the corresponding destination address at the time of the relaying of steps P1, P2 in this same FIG. 1. As such, port #0 can be specified unambiguously, and the IP packet can be forwarded to terminal A 10 without flooding (step P6).

[0008]

Note that this series of packet forwarding operations described in conjunction with FIG. 1 and FIG. 3 above is also illustrated in terms of the passage of time in FIG. 4. P1 to P6, which are the step numbers given in FIG. 4, respectively correspond to P1 to P6 in FIG. 1 and FIG. 3.

[0009]

Furthermore, in this system, based on an approach of effectively utilizing a limited buffer storage capacity, upon learning, when there is no DA search hit for the same address in a certain period, the learning result relating to this MAC address is deleted. This operation leading to deletion is referred to as "aging". Specifically, as illustrated in FIG. 5, the buffers of the cards of each node have a "hit bit" corresponding to the learned MAC address; when this is learned, the hit bit = "1". Then, at the first aging process, this hit bit is cleared (that is, made to be "0"; steps S11, S12). Here, the learning result is not yet deleted. Then, when the hit bit is still "0" at the next aging process (step S14, step S15, "No" at step S11), the corresponding MAC address is deleted from the buffer (step S13). Meanwhile, when overwriting learning is performed previous to this or when there is a hit at the DA search (that is, the destination search operation of step S3 in FIG. 2 above), the hit bit is again set to "1" (as a result, step S11 "Yes" and step S12). This configuration prevents an address currently being used (that is, being used for communication) from being deleted from the buffer.

[0010]

This aging process is further described in conjunction with FIGS. 6 to 8. Steps P1 to P6 in FIG. 7 are the same steps as steps P1 to P6 in FIG. 4 above. Now, each time an IP packet is forwarded, at steps P1, P2, the MAC address "A" is learned at card #0, and the hit bit thereof is made to be "1". Likewise, at steps P2, P3, the MAC address "A" is learned at card #1 and card #2. Continuing in the same vein, at steps P4, P5 and steps P5, P6, the MAC address "B" is learned at card #2 and card #0.

[0011]

Afterward, as illustrated in the uppermost row in FIG. 8, the first aging process clears the respective hit bits of the MAC addresses "A", "B" at each card #0, #1, #2 (steps S12). It is then supposed that before the second aging process, the next IP packet, likewise addressed to terminal B 20, is sent from terminal A 10 (step P7). At card #0 that receives this, the source MAC address "A" is learned by overwriting. At the same time, in the destination search, the destination MAC address "B" becomes the search target. As such, the MAC addresses "A", "B", which are temporarily cleared to "0" as above, have their hit bits respectively returned to "1".

[0012]

Then, this IP packet is forwarded to card #2 according to the learning result regarding the MAC address "B" that yielded a hit in the destination search (step P8). Then, in card #2 as well, the MAC addresses "A", "B" temporarily cleared to "0" as above have their hit bits respectively returned

to "1" by the same operation as above. It is then supposed that afterward, the next aging-process timing arrives without either of the IP packets of terminals A, B 10, 20 being received. Here, as illustrated in the lowermost row in FIG. 8, at card #0 and card #2, as above, the hit bits of the MAC addresses "A", "B" are each returned to "1" at steps P7, P8. As such, these are again cleared to "0". However, the corresponding learning result is not yet erased at this point (step S12). Meanwhile, for card #1, no corresponding IP packet passes through during this time, and the hit bits of the MAC addresses "A", "B" remain "0". As such, the corresponding learning result is deleted at this point (step S13).

[0013]

10 Next, a link aggregation function is described. This link aggregation function is a function of using a plurality of ports by deeming such to be one virtual high-speed port. This function enables a band improvement effect to be substantially obtained. The virtual port in this situation is referred to as a trunk. In using this trunk (referred to as "trunking"), when, as above, flooding is to be performed when an address is not learned, one representative port is selected from among the large number of ports had by this trunk according to a predetermined computation, and actual packet transmission is performed using this representative port.

[0014]

20 Then, at a node that receives an IP packet via the trunk, when carrying out path learning of a source address of this packet, instead of learning the port number at the time of receipt, information on the trunk including such is stored as the learning result. Afterward, in receiving an IP packet having a destination that passes through this trunk, one path is decided from among the paths constituting the trunk according to predetermined computation methods, independent of each other according to hardware, of these communication cards, and this decided path is applied to the corresponding packet. This configuration prevents the application of a different path in the same trunk to the same packet.

[0015]

30 Applying the link aggregation function is described in conjunction with FIG. 9 to FIG. 11. In this example, node groups 60, 70 are present between terminal A 10 and terminal B 20. Nodes respectively having communication cards #0, #1, #2 are provided in the node group 60, and nodes respectively having communication cards #3, #4, #5 are provided in the node group 70. In FIG. 9 and FIG. 10, a packet received from the terminal 10 and addressed to the terminal 20 is received at card #0 of the nodes 60. Because card #0 has no learning result for the terminal 20, flooding is performed (step P11). Cards #1, #2 that receive the flooded packet have a common trunk. One representative port is selected therefrom according to the computation methods that are independent according to hardware, this representative port is applied, and this packet is further transmitted (step P12).

[0016]

40 The path having this representative port leads to card #3 of the nodes 70. At this card #3 as well, no learning result relating to the terminal 20 that is the destination is had. As such, at card #3 as well, this packet is flooded (step P13). In this manner, this packet reaches the terminal 20 that is the destination through card #5. Note that during the packet forwarding from the terminal 10 to the terminal 20 illustrated in FIG. 9 and FIG. 10, at each communication card #0 to #5 of the node groups 60, 70, as described in conjunction with FIGS. 1 to 4, information relating to the path leading to the local device is learned in association with the source address of the packet.

[0017]

50 Next, as illustrated in FIG. 11 and FIG. 12, it is supposed this time that a packet is sent from the terminal 20 to the terminal 10. At card #5 of the node group 70 that receives this, the path to the terminal 10 that is the destination has been learned at the time of the forwarding operations described in conjunction with FIG. 9 and FIG. 10. As a result of this learning, it is recognized that the path to the terminal 10 has a trunk, and the path to be applied is selected from among the plurality

of paths had by this trunk according to the hardware-specific computation method of the local device. In this situation, it is supposed that the path including card #4 is selected. Note that here, in the packet forwarding operations described in conjunction with FIG. 10, the path going through card #3 instead of card #4 is applied. However, as above, in applying trunking, information on the entire trunk is learned instead of an individual path. Moreover, selection of paths included in the same trunk is decided according to a computation method specific to the source communication card. As such, when the forwarding direction is different, instead of card #3, which constitutes the path of the previous application, another card, #4, in the same trunk may be selected.

[0018]

10 Subsequently, this packet passes through card #3, arrives from card #3 to card #2 of the node group 60 (steps P14, P15), and further reaches the terminal 10 via card #0 in the node group 60.

[Patent Literature 1] JP 2002-247089 A

[Patent Literature 2] JP 2002-252625 A

[Disclosure of Invention]

[Problem to Be Solved by Invention]

[0019]

20 In this manner, in packet forwarding between the terminal 10 and the terminal 20, the hardware (communication cards) selecting the path in the trunk differs. As such, a different path in the same trunk may be selected. In this situation, the communication cards constituting each path only take part in forwarding in one direction. Therefore, at these cards, the learning results thereof are not utilized, and flooding is constantly performed during this packet communication.

[0020]

30 This is described in conjunction with FIG. 12 to FIG. 14. FIG. 13 illustrates operations following steps P11 to P16 illustrated in FIG. 10 and FIG. 11. When a packet addressed to terminal 20 is again received from the terminal 10 at card #0, like the previous time, the path of card #1 in the same trunk is selected and applied (step P17). At card #1 that receives this, from the information relating to this trunk, the packet is forwarded to card #3 (step P18). At card #3 that receives this, a search is made for a learning result for the terminal 20 that is the packet destination. However, as above, the packet heading from the terminal 20 to the terminal 10 does not pass through card #3 and passes through card #4 (step S15). As such, no learning result for the terminal 20 exists in card #3. As such, card #3 again floods this packet (step P19).

[0021]

40 Furthermore, afterward, in FIG. 14, when the packet addressed to the terminal 10 is again received from the terminal 20 at card #5, like the previous time, the path of card #4 in the same trunk is selected and applied (step P20). At card #4 that receives this, from the information relating to this trunk, the packet is forwarded to card #2 (step P21). At card #2 that receives this, a search is made for a learning result for the terminal 10 that is the packet destination. However, as above, the packet heading from the terminal 10 to the terminal 20 does not pass through card #2 and passes through card #1 (step S17). As such, no learning result for the terminal 10 exists in card #2. As such, card #2 again floods this packet (step P22).

[0022]

 Furthermore, at card #2 in the uppermost row in FIG. 14, as above, only forwarding in one direction from the terminal 10 to the terminal 20 occurs. As such, compared to a situation of forwarding in both directions, a forwarding cycle becomes longer, and a possibility of the corresponding learning result aging and being deleted during this time also increases.

[0023]

50 In this manner, nodes that constantly continue to perform flooding during a series of communications arise, and a possibility increases of a situation of increased line traffic.

[0024]

In view of the above problems, the present invention has as an object to provide a packet forwarding system wherein path learning is carried out effectively even in applying trunking by link aggregation and wherein flooding can be kept to a minimum.

[Means for Solving Problem]

[0025]

According to the present invention, each packet forwarding device is configured to generate a learn packet at a predetermined timing under predetermined conditions. That is, even in a situation wherein different paths are selected for a coming direction and a going direction due to trunking or the like and thus only forwarding in one of these directions is performed in a certain device, by generating a learn packet at a predetermined timing, this learn packet can force, for example, the device performing forwarding in only one direction in this manner to also perform packet reception in the other direction. This causes each path to perform packet reception in both directions as appropriate, and path learning in both directions is performed reliably. As a result, performing flooding repeatedly and thus inviting increased line traffic can be prevented.

[0026]

However, because an object of this learn packet is simply for effective path learning to be performed, generating this learn packet with needless frequency actually invites increased line traffic. To prevent such adverse effects, learn-packet generation conditions—that is, a generation frequency, a generation opportunity, and the like—must be appropriately determined.

[0027]

Furthermore, if a learning result were to remain in a buffer indefinitely without being erased despite a series of communications already being ended, a limited buffer storage capacity would not be used effectively. Therefore, there is a need to devise a utilization method of an aging process so communication ending is appropriately determined and, in this situation, a corresponding learning result is swiftly deleted.

[0028]

That is, it is desirable for the learn-packet generation conditions to be a situation wherein the path leading to the destination of the receive packet includes a trunk, the learn packet being generated for the first time not at the time of the first learning but when the learning result becomes a hit in a destination search. That is, a configuration is desirable that generates the learn packet when a packet is received whose destination is the learned source. Note that at this time, it is desirable to transmit the learn packet to all forwarding devices constituting this trunk. Moreover, it is desirable to also generate the learn packet each subsequent time when the learning result again becomes a hit in the destination search.

[0029]

However, during this time as well, it is desirable for the aging process to be carried out periodically. That is, it is desirable for deletion from the buffer to be performed after a predetermined time is elapsed from either among a time of the final learning—that is, the final packet reception from the same source terminal—and the final packet reception for the same destination terminal. Therefore, when a packet is subsequently received again from the same source terminal, this is treated as the first learning.

[Effects of Invention]

[0030]

According to the present invention, effects of path learning can be utilized even when coming and going communication paths are different. As such, flooding can be effectively suppressed, and line traffic can be reduced.

[Preferred Embodiment of Invention]

[0031]

An embodiment of the present invention is described below in conjunction with the drawings.

[0032]

FIG. 15 to FIG. 20 are for describing an embodiment of the present invention. A basic system configuration thereof assumes the system configuration example described in conjunction with FIG. 9 to FIG. 14. Note that each node constituting the system is made to be, for example, L2SW.

[0033]

In FIG. 15, the path to the destination of the receive packet includes a trunk. Moreover, when, at these nodes, a packet from the same address as the corresponding destination address (DA) is already received and the source-address learning result from this time is had—that is, when there is a hit in the destination search (step S3 in FIG. 2)—a learn packet is transmitted to all nodes having the corresponding trunk. That is, in FIG. 15, a packet is first forwarded from the terminal 10 to the terminal 20. When, at each node that this forwarding operation passes through—for example, card #5—a reply packet from the terminal 20 to the terminal 10 is afterward received, at card #5, the learn packet is transmitted to all nodes having the corresponding trunk—that is, cards #3, #4. This causes the path for the corresponding MAC address to be learned at all of these nodes. That is, in this situation, at cards #3, #4, the path having card #5 is learned for the MAC address “B” of the terminal 20.

[0034]

To facilitate description, a situation is then supposed wherein the paths of cards #1, #3 are selected as the representative ports of the trunk in the direction wherein the terminal 20 receives and, in contrast, the paths of cards #4, #2 are selected as the representative ports of the same trunk in the direction of transmitting to the terminal 10. In this situation, when no learn packet is generated, as described in conjunction with FIG. 12 to FIG. 14, flooding occurs repeatedly. That is, when a packet of a transmission direction toward the terminal 20 again reaches card #3, because card #3 has not received a packet of a reception direction from the terminal 20, no corresponding learning result is had, and flooding is again performed. However, according to this embodiment of the present invention, as above, the learn packet is sent from card #5 to cards #3, #4. As such, at both card #3 and card #4, the MAC address “B” of the terminal 20 is already learned. As such, even when a packet of a transmission direction toward the terminal 20 again reaches card #3, card #3 can specify the path to the terminal 20 according to the learning result provided by the learn packet.

[0035]

FIG. 16 illustrates an operation time chart of each card in the present embodiment. In the diagram, (a) is clock pulses for performing the aging process. Note that among these clock pulses, the pulses enclosed in the dashed line are the pulses that start the actual aging process. That is, the aging process is started once every four generations of the clock pulse. (b) illustrates count values of a card LA counter. This counter is updated so as to sequentially adopt the values of 0 to 7 repeatedly as illustrated at each clock pulse (a) above.

[0036]

(c) in FIG. 16 illustrates count values of a content counter. A content counter is provided for each learning result. Moreover, the vertical dashed lines indicate receive-packet reception timings. As illustrated, new learning is performed when a packet of a certain source address (SA) is received. As a result, the value of this content counter is made to be “0”. Then, when a packet for the same MAC address is next received, the value of the card LA counter at this reception timing—here, “4”—is incorporated. Then, in this situation, the learn packet is generated (indicated by the star symbol in the diagram) and transmitted to all nodes of the corresponding trunk.

[0037]

(d) in FIG. 16 illustrates a state of a learn-packet transmitted-state flag. This flag is provided for each learning result and is set to the value “1” when a learn packet is generated for the corresponding MAC address. Then, when a packet of the corresponding destination is again received immediately after the learn-packet generation above, no second learn packet is generated

if this reception is within a period wherein the value of the card LA counter is still "4". That is, the shortest generation cycle of the learn packet is 1/4 of an aging-process starting cycle.

[0038]

Then, afterward, when the aging process is started, the state flag is cleared to "0" as illustrated. Then, as illustrated, when a corresponding receive packet is again received, a learn packet is again generated because the value of the card LA counter was updated after the previous learn-packet generation. Then, as a result, as illustrated, the state flag of (d) again returns to "1".

[0039]

10 Then, afterward, when the next aging process is started without a corresponding receive packet being received, like the hit bits indicated at steps S11, S12 in FIG. 5, first, the state flag is cleared to "0". Then, afterward, when no corresponding receive packet is received until the starting of the next aging process, as illustrated, the learning result for the corresponding MAC address is deleted. Note that in FIG. 16, a state wherein the content counter of (c) includes no number signifies that no learning result is present.

[0040]

20 In this manner, in the configuration illustrated in FIG. 16 according to this embodiment of the present invention, a learn-packet generation interval—that is, sending interval—is set to be shorter than an aging-process starting interval. That is, to prevent learned information from being deleted in the aging process, despite communication being ongoing, due to process timing fluctuations between each node that arise from aging-process timing variation, transmission-process time lags, or the like; packet deletion that arises when a packet flow rate is high; or the like, it is desirable to transmit the learn packet a plurality of times and to set a transmission cycle thereof to be shorter than the aging-process starting interval.

[0041]

In FIG. 15, a situation is supposed wherein an IP packet addressed from the terminal 10 to the terminal 20 is received at card #0 of the node group 60. It is also supposed that at this point, there are no learning results for the MAC address "A" of the terminal 10 and the MAC address "B" of the terminal 20. In this situation, as illustrated in FIG. 17, this packet is flooded from card #0 (step P31).

30 [0042]

Cards #1, #2 that receive this have, as a path leading to the terminal 20, a trunk B that leads to cards #3, #4. Then, this packet is flooded by both card #1 and card #2 (step P32). However, in this situation, for trunk B, card #1 is selected as the representative port. As such, no transmission from card #2 to card #4 in the same trunk B is performed. Moreover, card #3 in the node group 70 that receives this also performs flooding (step P33).

[0043]

40 Cards #4, #5 that receive this perform further flooding (step P34). Note that in this situation, selected as the representative port for trunk B above is not card #4 but card #3, which opposes card #1 selected as the representative port above. As such, no transmission from card #4 to card #2 in the same trunk B is performed. Note that during the above series of forwarding operations, at cards #3, #4, #5, for the MAC address "A" (SA) of the source terminal 10, the same trunk as trunk B above is learned as trunk A.

[0044]

Afterward, card #5 sends, at a predetermined timing, a learn packet to all nodes having trunk A above (step P34). As a result, both card #3 and card #4, which have trunk A, learn the path having card #5 for the MAC address "B" of the terminal 20.

[0045]

50 Afterward, as illustrated in FIG. 18, when card #5 receives an IP packet for the terminal 10 from the terminal 20, because the MAC address "A" of this terminal 10 has been learned when a packet was received at the above step P33, trunk A that is this learning result is applied. At this

time, the path of card #4 is selected from among the paths constituting trunk A according to the hardware-specific computation method of card #5, and this path is applied (step P20). At card #4 that receives this, card #2 is specified as the opposing card from predetermined trunk information, and this card is applied (step P21).

[0046]

At card #2 that receives this, because the MAC address "A" is already learned at the time of the previous packet reception at step P31, card #0 is specified and applied as the corresponding path (step P37). Then, card #0 forwards (and does not flood) the IP packet to the terminal 10 according to the learning result of when a packet was received from the terminal 10 the previous time (uppermost row in FIG. 17). Afterward, card #0 sends a learn packet to all nodes having trunk B above (step P38). As a result, both card #1 and card #2, which have trunk B, perform overwrite learning of the path having card #0 for the MAC address "A" of the terminal 10.

[0047]

Note that when aging is started at this point, "1" is set for both the hit bit and the learn-packet transmitted-state flag above. As such, these are cleared to "0". However, the learning results for the MAC addresses "A", "B" of the corresponding terminal 10 and terminal 20, respectively, are not deleted at this point.

[0048]

Then, afterward, as illustrated in FIG. 19, when a packet for the terminal 20 is again sent from the terminal 10, card #0 that receives this overwrites the learning result for the MAC address "A" of the terminal 10. Moreover, card #0 has learned trunk B as the path for the corresponding MAC address "B" at the time of the previous packet reception from the terminal 20 (step P37 in FIG. 18). As such, the path of card #1 is selected from among the paths constituting trunk B according to the hardware-specific computation method of card #0, and this path is applied (step P41).

[0049]

Furthermore, in this situation, there is a hit for the corresponding MAC address "B" in the destination search for the terminal 20, and the path leading to the terminal 20 includes a trunk. As such, learn-packet generation conditions are met if 1/4 of the aging-process starting interval is elapsed from the previous learn-packet generation. In this situation, the learn packet is sent to cards #1, #2 (step P41). As a result, both card #1 and card #2 overwrite the learning result for the MAC address "A".

[0050]

Furthermore, when the packet that is sent from the terminal 10 and passes through card #0 reaches card #1, at card #1, card #3 is specified as the opposing card from predetermined trunk information, and this is applied (step P42). As a result, at card #3 that receives the packet, because the MAC address "B" is already learned at the time of the previous learn-packet reception at step P34, card #5 is specified and applied as the corresponding path (step P43).

[0051]

Note that when no learn packet is generated at step S34 in this manner, flooding is performed here. However, even in this situation, as described below, when a packet is afterward sent again from the terminal 20 to the terminal 10 at card #5, the above learn-packet generation conditions come to be met. As such, the learn packet is also sent to card #3, and the terminal 20 is learned (step P45 in FIG. 20). As such, path specification subsequently becomes possible, and flooding becomes unnecessary.

[0052]

Then, card #5 that receives the packet from the terminal 10 in this manner forwards (and does not flood) this packet to the terminal 20 according to the learning result of when a packet was received from the terminal 20 the previous time (uppermost row in FIG. 18). Afterward, card #5 sends a learn packet to all nodes having trunk A (step P44). As a result, both card #3 and card #4,

which have trunk A, perform overwrite learning of the path having card #5 for the MAC address "B" of the terminal 20.

[0053]

Note that when aging is started at this point—that is, the uppermost row in FIG. 20—"1" is set for both the hit bit and the learn-packet transmitted-state flag. As such, these are cleared to "0". However, the learning results for the MAC addresses "A", "B" of the corresponding terminal 10 and terminal 20, respectively, are not deleted.

[0054]

10 Then, afterward, as illustrated in FIG. 20, when a packet for the terminal 10 is once again sent from the terminal 20, card #5 that receives this overwrites the learning result for the MAC address "B" of the terminal 20. Moreover, card #5 has performed overwrite learning of trunk A as the path for the corresponding MAC address "A" at the time of the previous packet reception from the terminal 10 (step P43 in FIG. 19). As such, the path of card #4 is selected from among the paths constituting trunk A according to the hardware-specific computation method of card #5, and this path is applied (step P45).

[0055]

20 Furthermore, in this situation, at card #5, there is a hit for the corresponding MAC address "A" in the destination search for the terminal 10, and the path leading to the terminal 10 includes a trunk. As such, as above, learn-packet generation conditions are met if 1/4 of the aging-process starting interval is elapsed from the previous learn-packet generation. In this situation, the learn packet is sent to cards #3, #4 (step P45). As a result, both card #3 and card #4 overwrite the learning result for the MAC address "B".

[0056]

As above, when the packet that is sent from the above terminal 20 and passes through card #5 reaches card #4, at card #4, card #2 is specified as the opposing card from predetermined trunk information, and this is applied (step P46). At card #2 that receives this, because the MAC address "A" is already learned at the time of the previous learn-packet reception at step P41, card #0 is specified and applied as the corresponding path (step P47).

[0057]

30 Then, card #0 that receives the packet from the terminal 20 in this manner forwards (and does not flood) this packet to the terminal 10 according to the learning result of when a packet was received from the terminal 10 the previous time (uppermost row in FIG. 19). Afterward, card #0 sends a learn packet to all nodes having trunk B (step P48). As a result, both card #1 and card #2, which have trunk B, perform overwrite learning of the path having card #0 for the MAC address "A" of the terminal 10.

[0058]

Note that the present application includes the configurations of the following notes.

(Note 1)

40 A packet forwarding method for forwarding a receive packet according to a destination address thereof, including:

a learning step of learning, at each device, a source address of the receive packet in correspondence with a receive path thereof,

each device being configured so for the destination address of the receive packet, when a learning result is had regarding a correspondence relationship between this address and a corresponding path, this receive packet is sent through this path and, meanwhile, when no learning result is had, this receive packet is sent through all paths; and

a learn-packet generation step of generating a learn packet for each device at a predetermined timing.

(Note 2)

The method of note 1, further including: a deletion step of deleting the learning result of the learning step according to a predetermined deletion cycle; wherein

a cycle whereby the learn packet can be generated at the learn-packet generation step is set to be shorter than the predetermined deletion cycle.

(Note 3)

The method of note 2, wherein each device is configured to have a learn-packet transmitted flag, raise this flag at a time of carrying out the learn-packet generation step, clear this flag if this is raised at a timing of carrying out the deletion step but not delete the learning result, and delete the learning result for the first time when this flag is still cleared at a timing of carrying out the next deletion step.

(Note 4)

The method of any among notes 1 to 3, wherein the learn-packet generation step is configured to be carried out for a situation wherein, for the destination address of the receive packet, no learning result regarding the correspondence relationship between this address and the corresponding path is had.

(Note 5)

The method of any among notes 1 to 4, wherein when applying link aggregation that deems a plurality of paths to be a single, virtual aggregate path, the learning step is configured to learn a virtual aggregate path as the receive path.

(Note 6)

The method of any among notes 1 to 5, wherein in the learn-packet generation step, for the destination address of the receive packet, the path leading to this destination is a virtual aggregate path provided by applying link aggregation; the learn-packet generation step is configured to be carried out for a situation wherein no learning result is had regarding the correspondence relationship between this address and the corresponding path; and the learn-packet generation step is configured to generate at this time a learn packet for all devices having this virtual aggregate path.

(Note 7)

A packet forwarding system for forwarding a receive packet according to a destination address thereof, having:

a learning means of each device performing packet forwarding learning a source address of the receive packet in correspondence with a receive path thereof; wherein

each device is configured so for the destination address of the receive packet, when a learning result is had regarding a correspondence relationship between this address and a corresponding path, this receive packet is sent through this path and, meanwhile, when no learning result is had, this receive packet is sent through all paths, and

each device further has a learn-packet generation means of generating a learn packet for another device at a predetermined timing.

(Note 8)

The packet forwarding system of note 7, further including: a deletion means of deleting the learning result of the learning means according to a predetermined deletion cycle; wherein

a cycle whereby the learn packet can be generated by the learn-packet generation means is set to be shorter than the predetermined deletion cycle.

(Note 9)

The packet forwarding system of note 8, wherein each device is configured to have a learn-packet transmitted flag, raise this flag at a time of learn-packet generation by the learn-packet generation means, clear this flag if this is raised at a timing of learning-result deletion by the deletion means but not delete the learning result, and delete the learning result for the first time when this flag is still cleared at the next deletion timing.

(Note 10)

The packet forwarding system of any among notes 7 to 9, wherein the learn-packet generation means is configured to generate the learn packet for a situation wherein, for the destination address of the receive packet, no learning result regarding the correspondence relationship between this address and the corresponding path is had.

(Note 11)

The packet forwarding system of any among notes 7 to 10, wherein when applying link aggregation that deems a plurality of paths to be a single, virtual aggregate path, the learning means is configured to learn a virtual aggregate path as the receive path.

(Note 12)

10 The packet forwarding system of any among notes 7 to 11, wherein in the learn-packet generation means, for the destination address of the receive packet, the path leading to this destination is a virtual aggregate path provided by applying link aggregation; the learn-packet generation means is configured to generate the learn packet for a situation wherein no learning result is had regarding the correspondence relationship between this address and the corresponding path; and the learn-packet generation means is configured to generate at this time a learn packet for all devices having this virtual aggregate path.

(Note 13)

A packet forwarding device for forwarding a receive packet according to a destination address thereof, comprising:

20 a learning means of learning a source address of the receive packet in correspondence with a receive path thereof,

a configuration being such that for the destination address of the receive packet, when a learning result is had regarding a correspondence relationship between this address and a corresponding path, this receive packet is sent through this path and, meanwhile, when no learning result is had, this receive packet is sent through all paths; and

a learn-packet generation means of generating a learn packet for another packet forwarding device at a predetermined timing.

(Note 14)

30 The packet forwarding device of note 13, further including: a deletion means of deleting the learning result of the learning means according to a predetermined deletion cycle; wherein

a cycle whereby the learn packet can be generated by the learn-packet generation means is set to be shorter than the predetermined deletion cycle.

(Note 15)

The packet forwarding device of note 14, wherein the packet forwarding device is configured to have a learn-packet transmitted flag, raise this flag at a time of learn-packet generation by the learn-packet generation means, clear this flag if this is raised at a timing of learning-result deletion by the deletion means but not delete the learning result, and delete the learning result for the first time when this flag is still cleared at the next deletion timing.

(Note 16)

40 The packet forwarding device of any among notes 13 to 15, wherein the learn-packet generation means is configured to generate the learn packet for a situation wherein, for the destination address of the receive packet, no learning result regarding the correspondence relationship between this address and the corresponding path is had.

(Note 17)

The packet forwarding device of any among notes 13 to 16, wherein when applying link aggregation that deems a plurality of paths to be a single, virtual aggregate path, the learning means is configured to learn a virtual aggregate path as the receive path.

(Note 18)

50 The packet forwarding device of any among notes 13 to 17, wherein in the learn-packet generation means, for the destination address of the receive packet, the path leading to this

destination is a virtual aggregate path provided by applying link aggregation; the learn-packet generation means is configured to generate the learn packet for a situation wherein no learning result is had regarding the correspondence relationship between this address and the corresponding path; and the learn-packet generation means is configured to generate at this time a learn packet for all packet forwarding devices having this virtual aggregate path.

[Brief Description of Drawings]

[0059]

[FIG. 1] A block diagram for describing one example of a conventional path learning function (part 1).

10 [FIG. 2] An operations flowchart for describing one example of a conventional path learning function.

[FIG. 3] A block diagram for describing one example of a conventional path learning function (part 2).

[FIG. 4] An operation transition diagram for describing one example of a conventional path learning function.

[FIG. 5] An operations flowchart for describing one example of a conventional aging-process function.

[FIG. 6] A block diagram for describing one example of a conventional aging-process function.

20 [FIG. 7] An operation transition diagram for describing one example of a conventional aging-process function (part 1).

[FIG. 8] An operation transition diagram for describing one example of a conventional aging-process function (part 2).

[FIG. 9] A block diagram for describing one example of a conventional link aggregation function.

[FIG. 10] An operation transition diagram for describing one example of a conventional link aggregation function (part 1).

[FIG. 11] An operation transition diagram for describing one example of a conventional link aggregation function (part 2).

[FIG. 12] A block diagram for describing problems due to repeated flooding according to one conventional example.

30 [FIG. 13] An operation transition diagram for describing one example of a conventional link aggregation function (part 3).

[FIG. 14] An operation transition diagram for describing one example of a conventional link aggregation function (part 4).

[FIG. 15] A block diagram for describing operations of a packet forwarding system of an embodiment of the present invention.

[FIG. 16] A time-chart diagram for describing operations of a packet forwarding device of an embodiment of the present invention.

[FIG. 17] An operation transition diagram for describing operations of the packet forwarding system of an embodiment of the present invention (part 1).

40 [FIG. 18] An operation transition diagram for describing operations of the packet forwarding system of an embodiment of the present invention (part 2).

[FIG. 19] An operation transition diagram for describing operations of the packet forwarding system of an embodiment of the present invention (part 3).

[FIG. 20] An operation transition diagram for describing operations of the packet forwarding system of an embodiment of the present invention (part 4).

[Description of Reference Signs]

[0060]

10, 20 Terminal

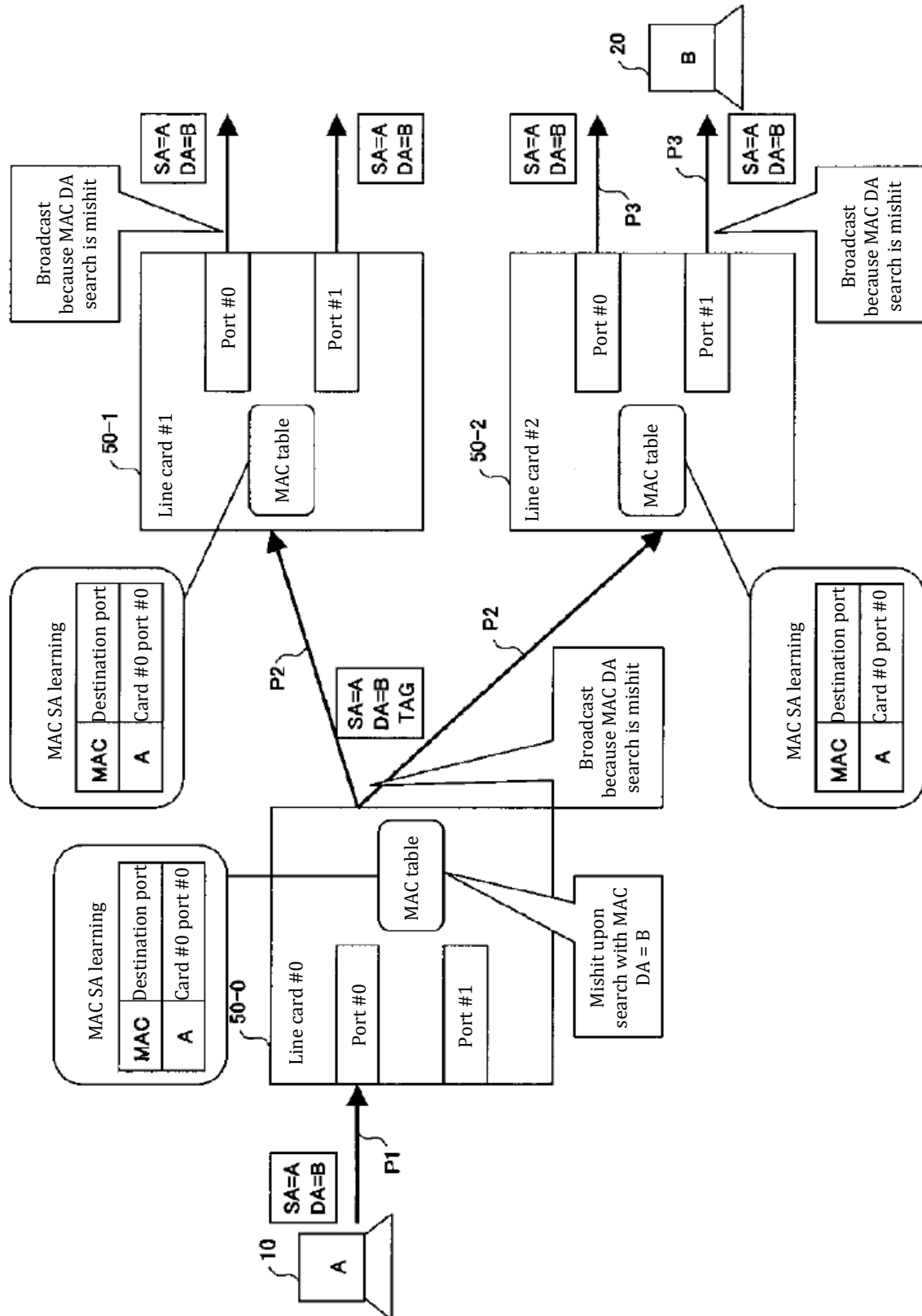
50-0, 50-1, 50-2 Node

50 #0, #1, #2, #3, #4, #5 Line card or node

60, 70 Node group

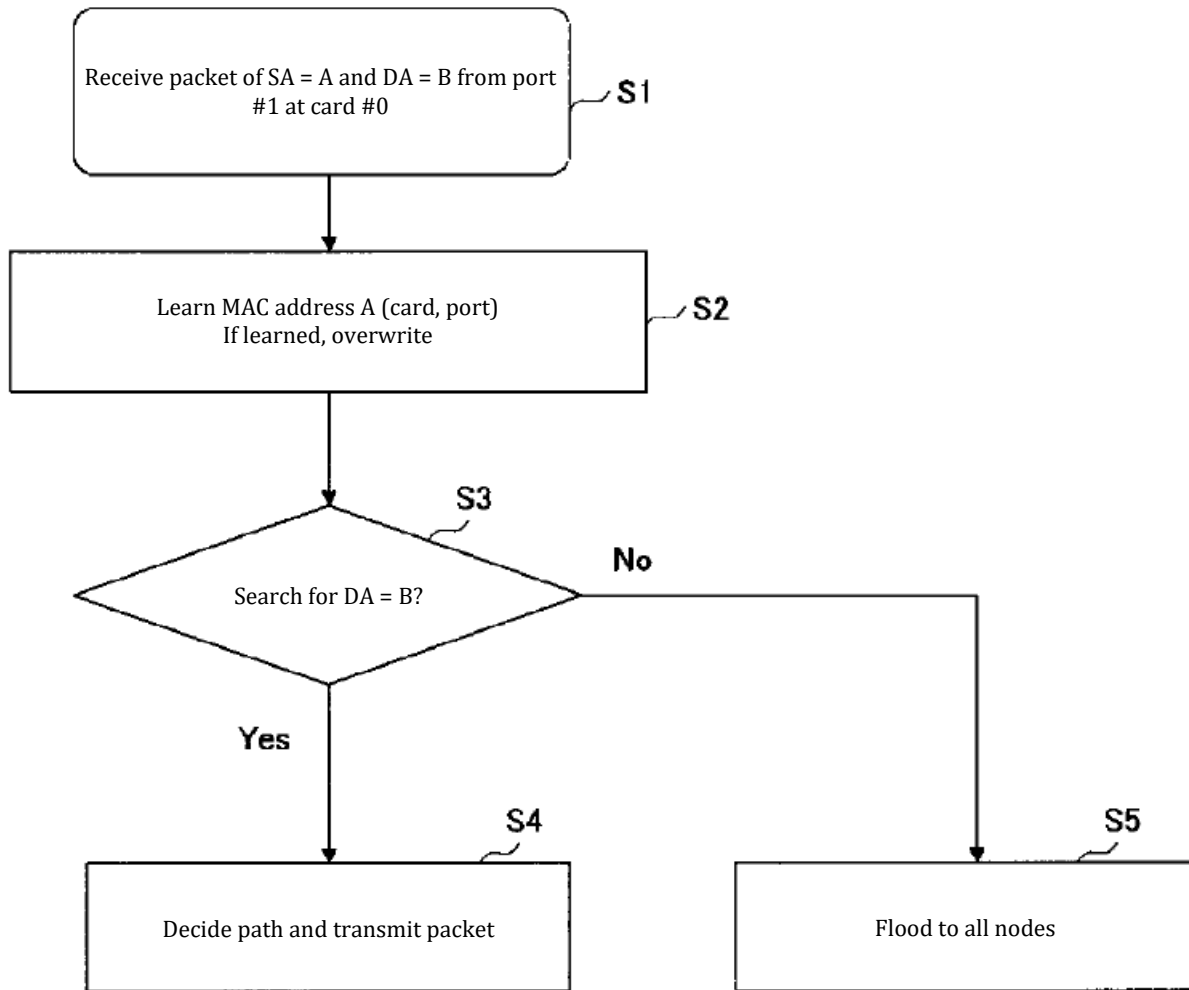
[FIG. 1]

Block diagram for describing one example of conventional path learning function (part 1)

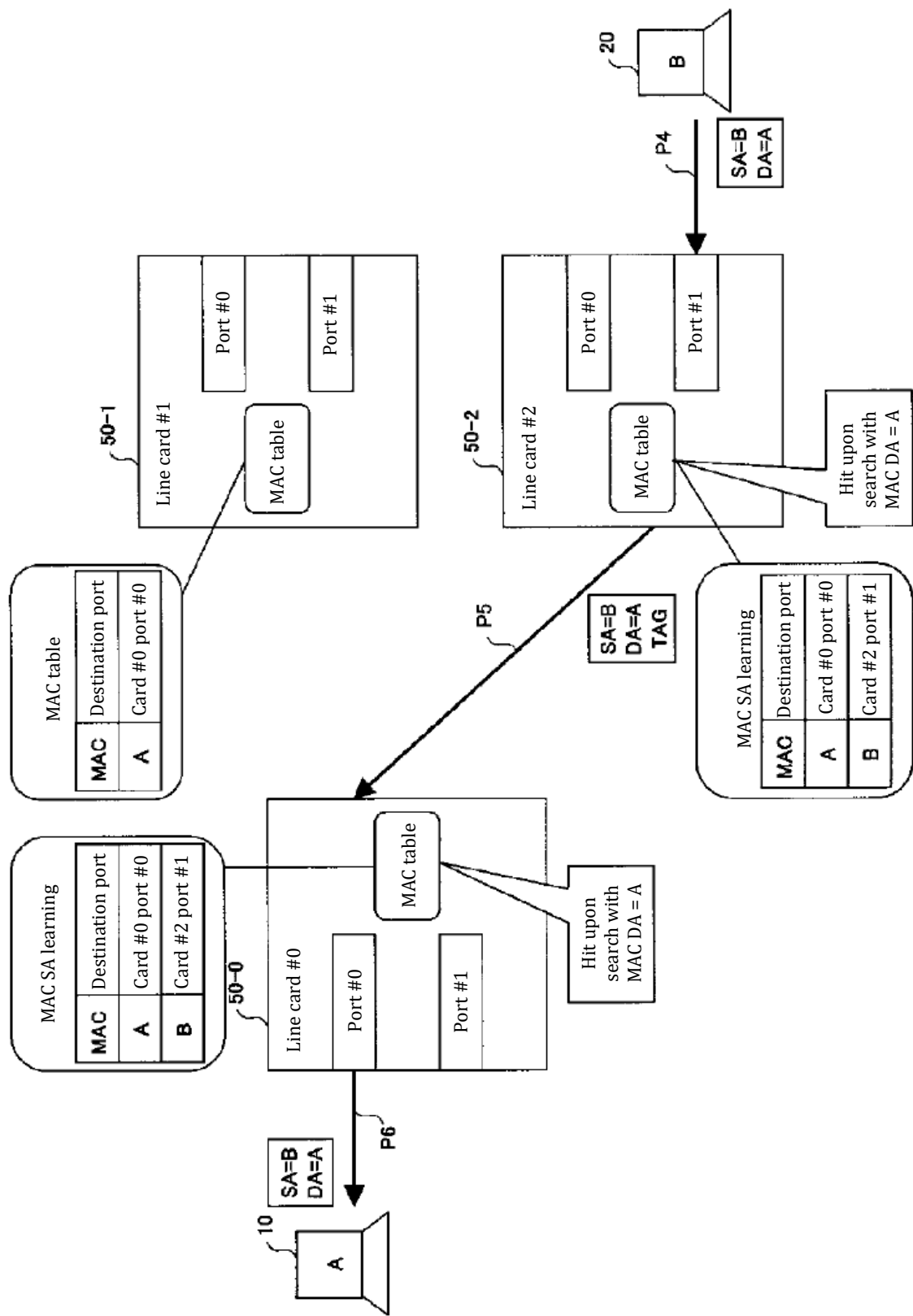


[FIG. 2]

Operations flowchart for describing one example of conventional path learning function

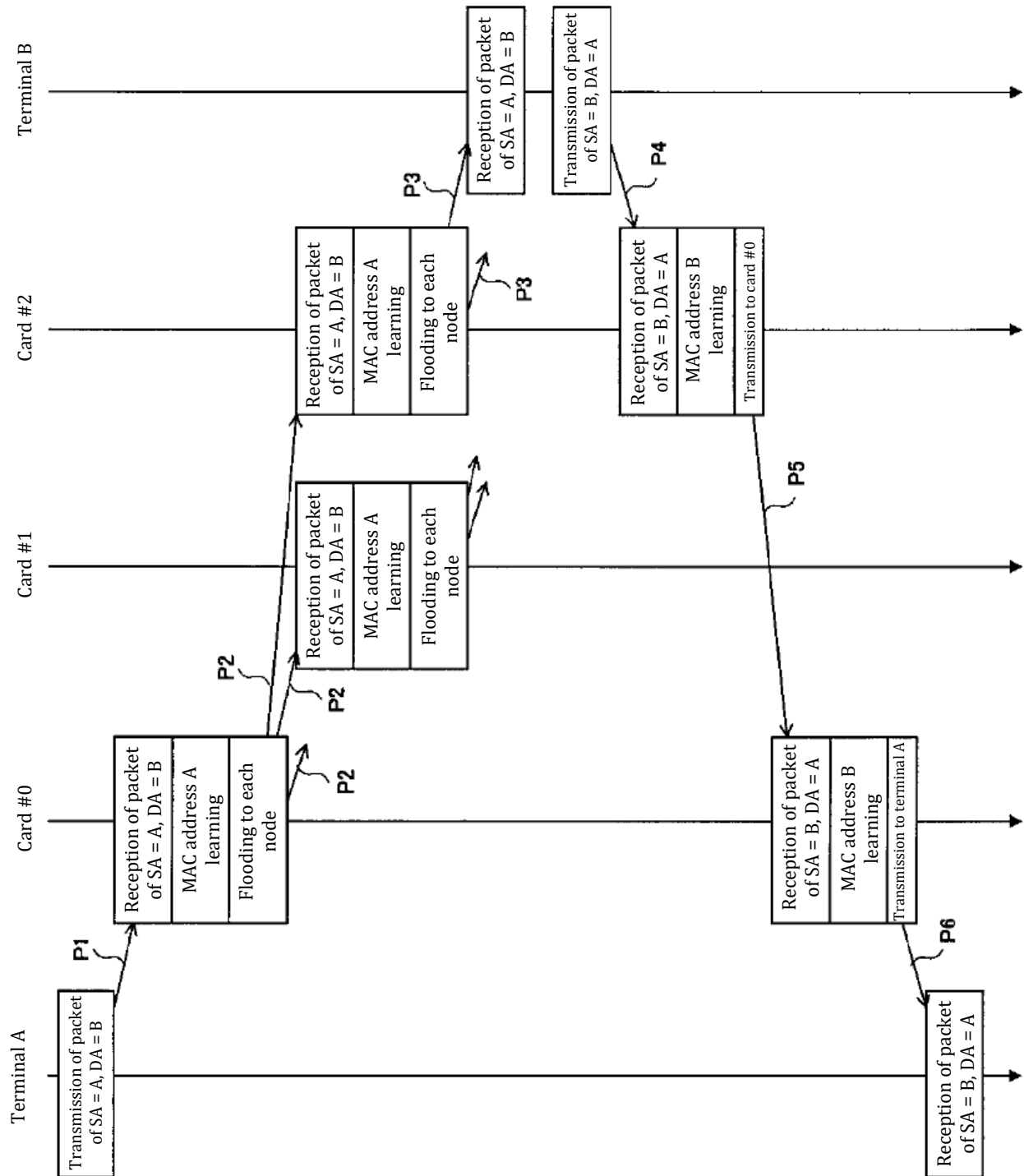


[FIG. 3]
Block diagram for describing one example of conventional path learning function (part 2)



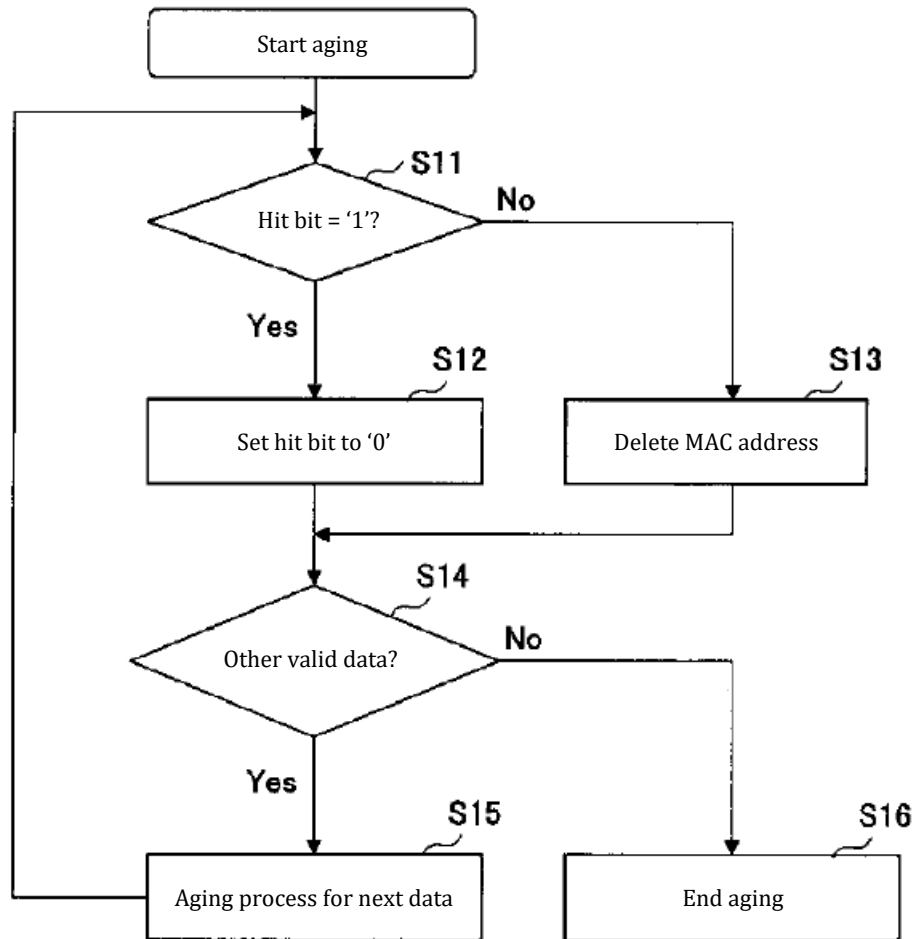
[FIG. 4]

Operation transition diagram for describing one example of conventional path learning function



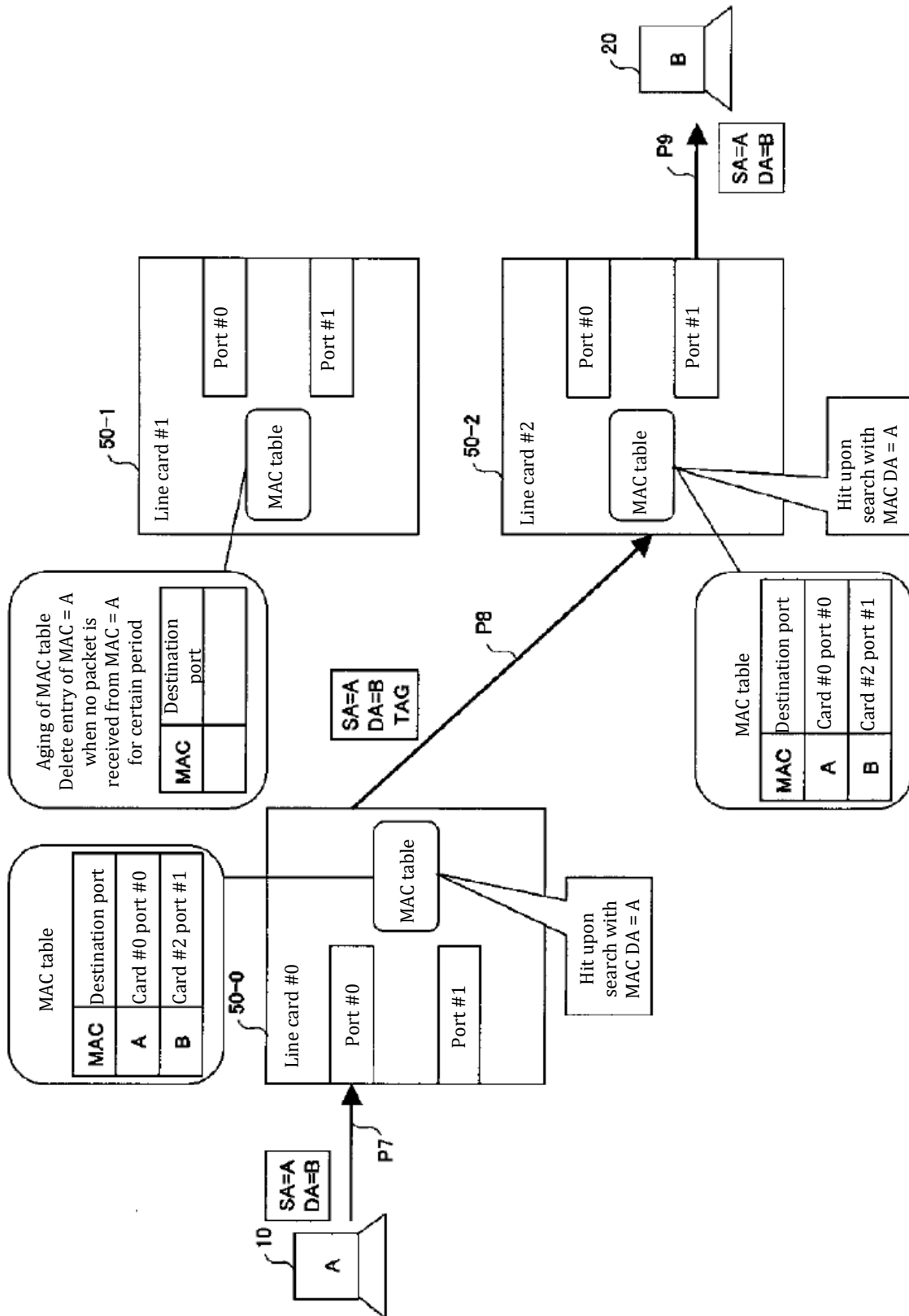
[FIG. 5]

Operations flowchart for describing one example of conventional aging-process function



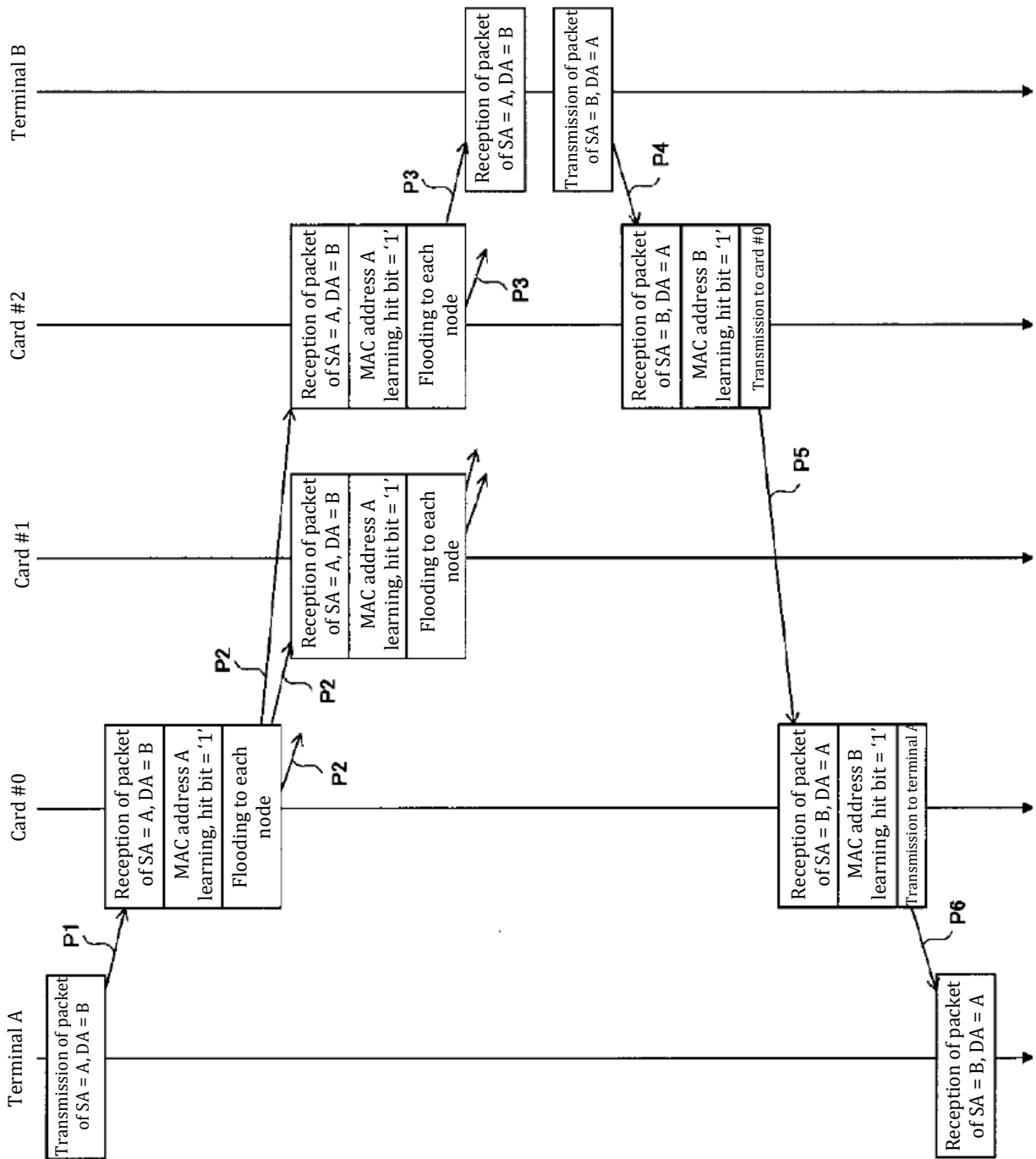
[FIG. 6]

Block diagram for describing one example of conventional aging-process function



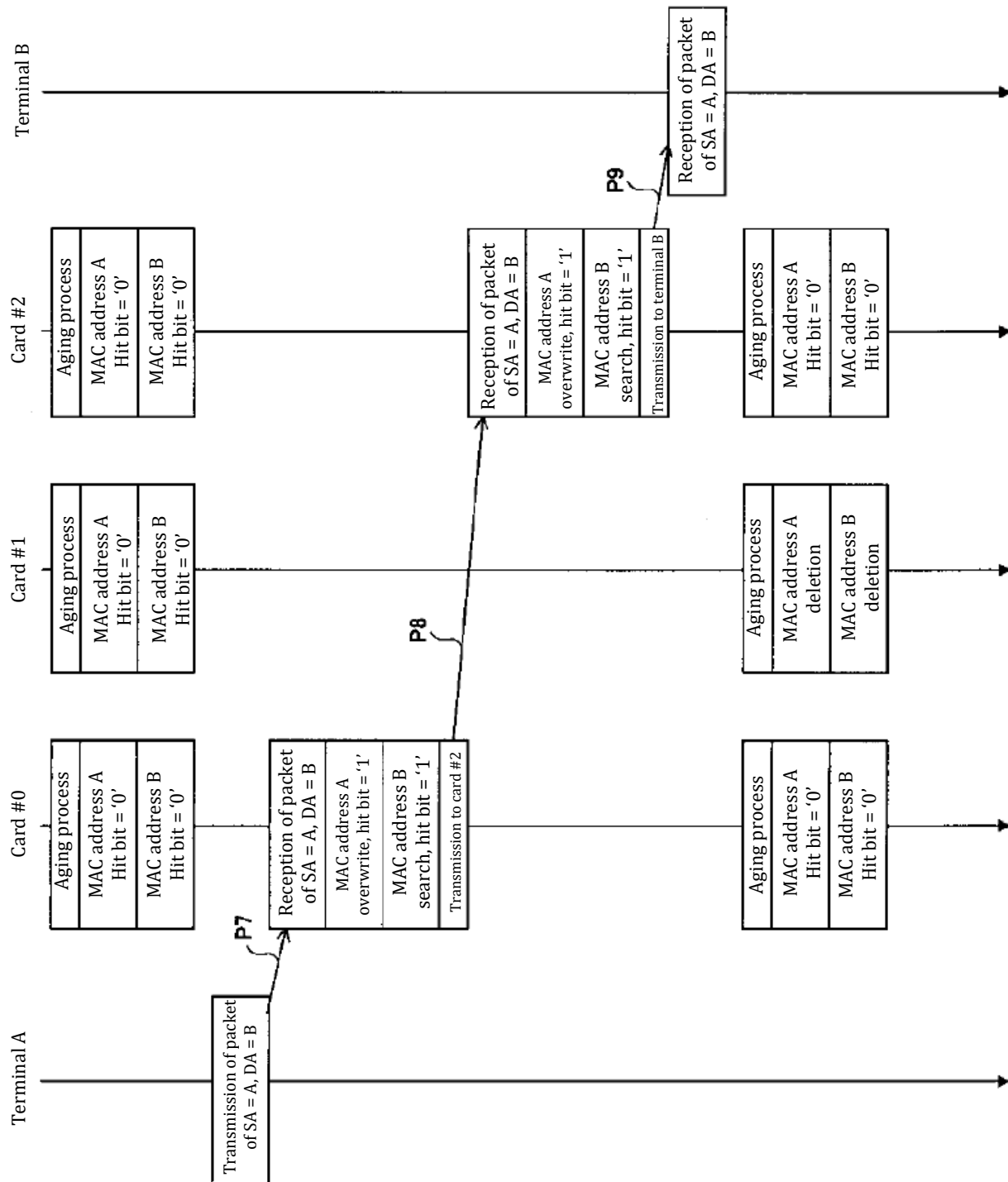
[FIG. 7]

Operation transition diagram for describing one example of conventional aging-process function
(part 1)



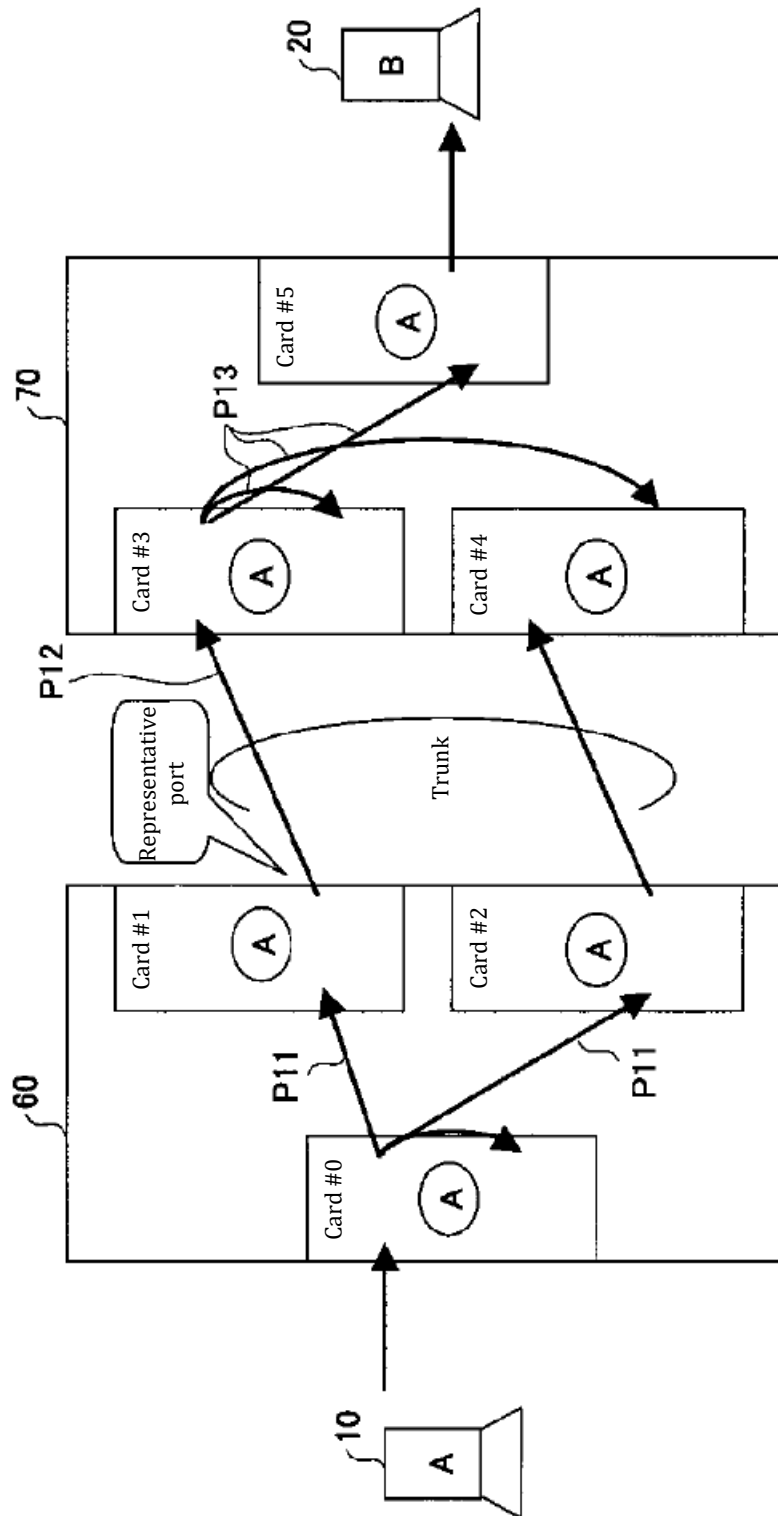
[FIG. 8]

Operation transition diagram for describing one example of conventional aging-process function
(part 2)

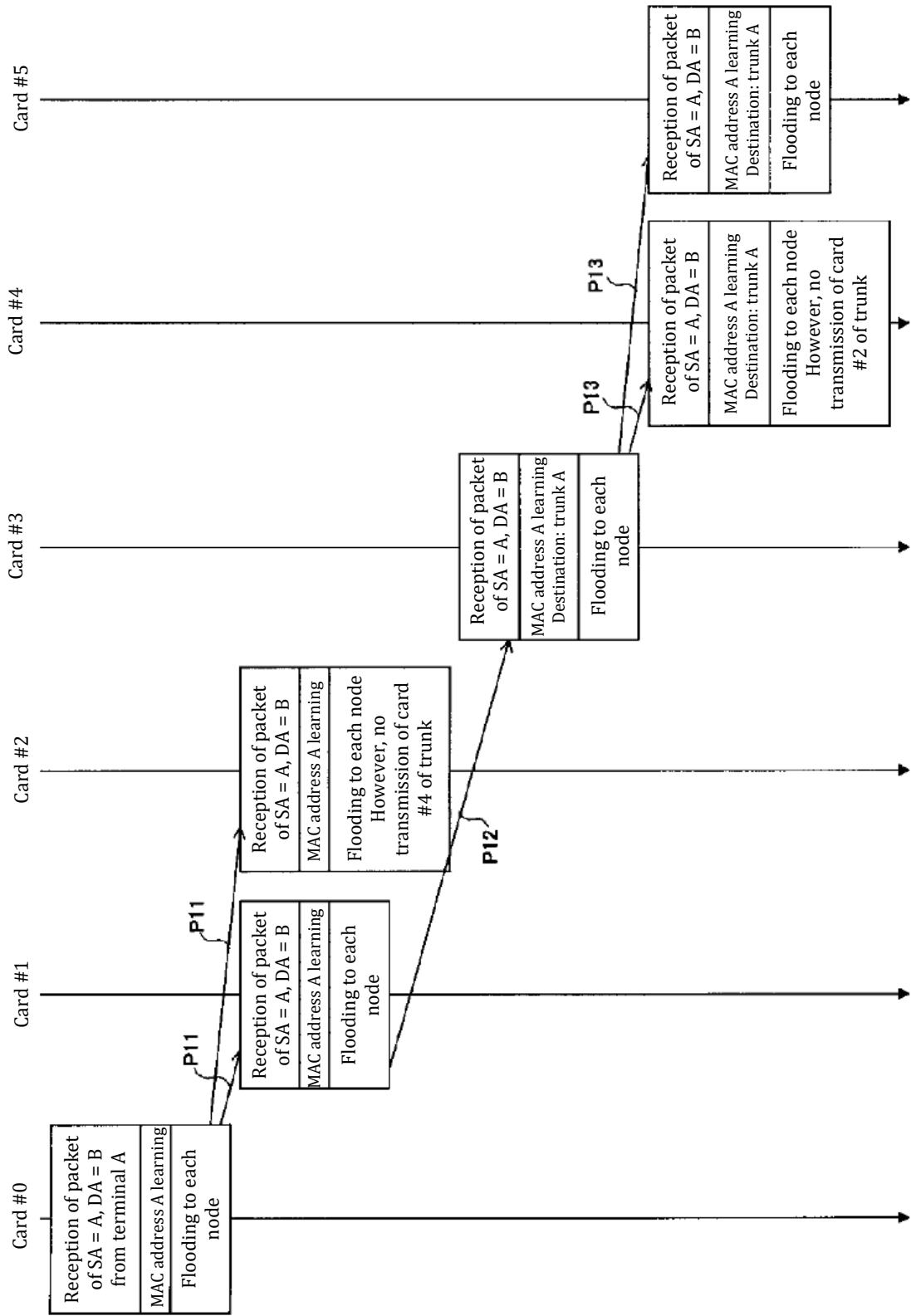


[FIG. 9]

Block diagram for describing one example of conventional link aggregation function

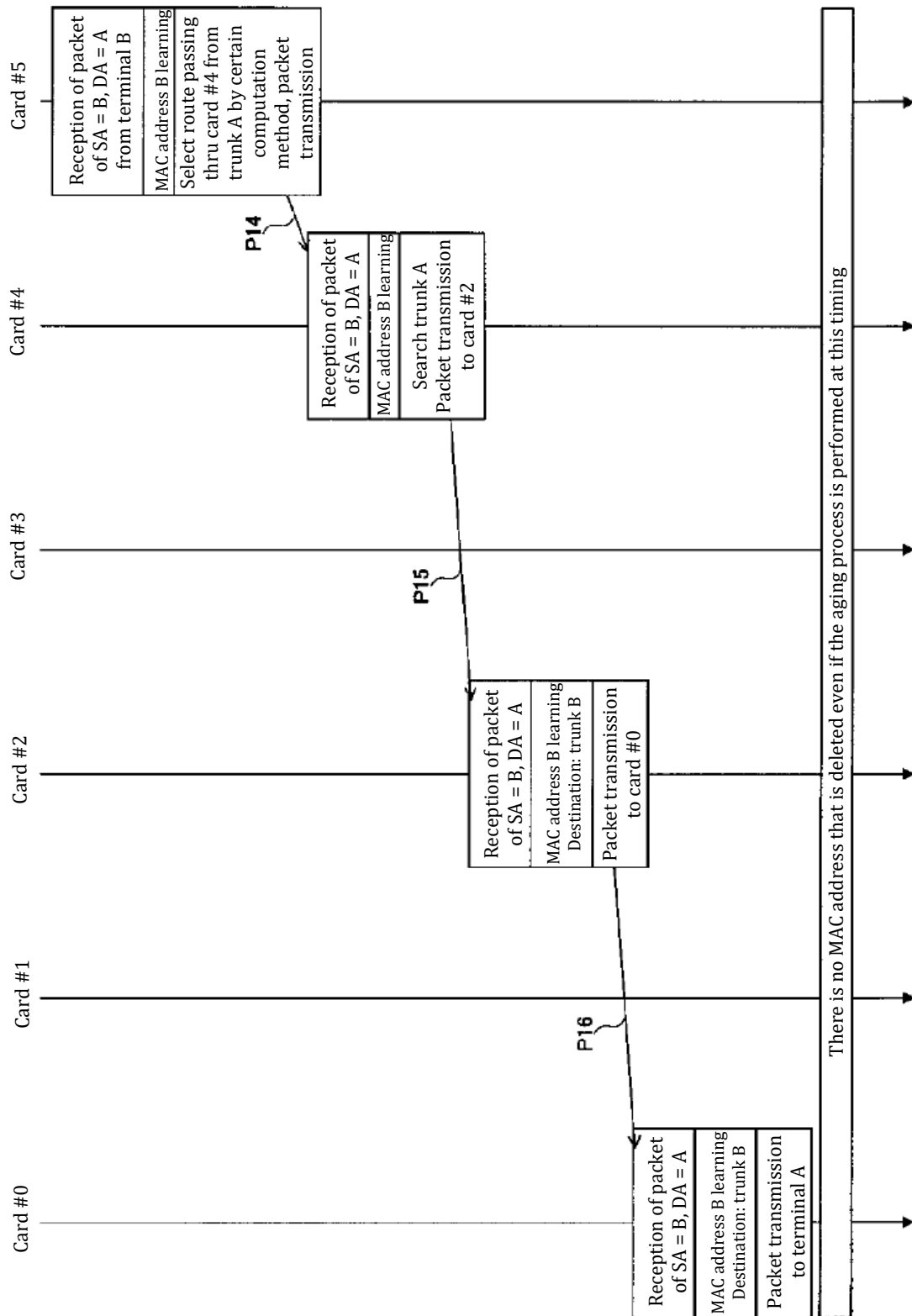


[FIG. 10]
Operation transition diagram for describing one example of conventional link aggregation function
(part 1)



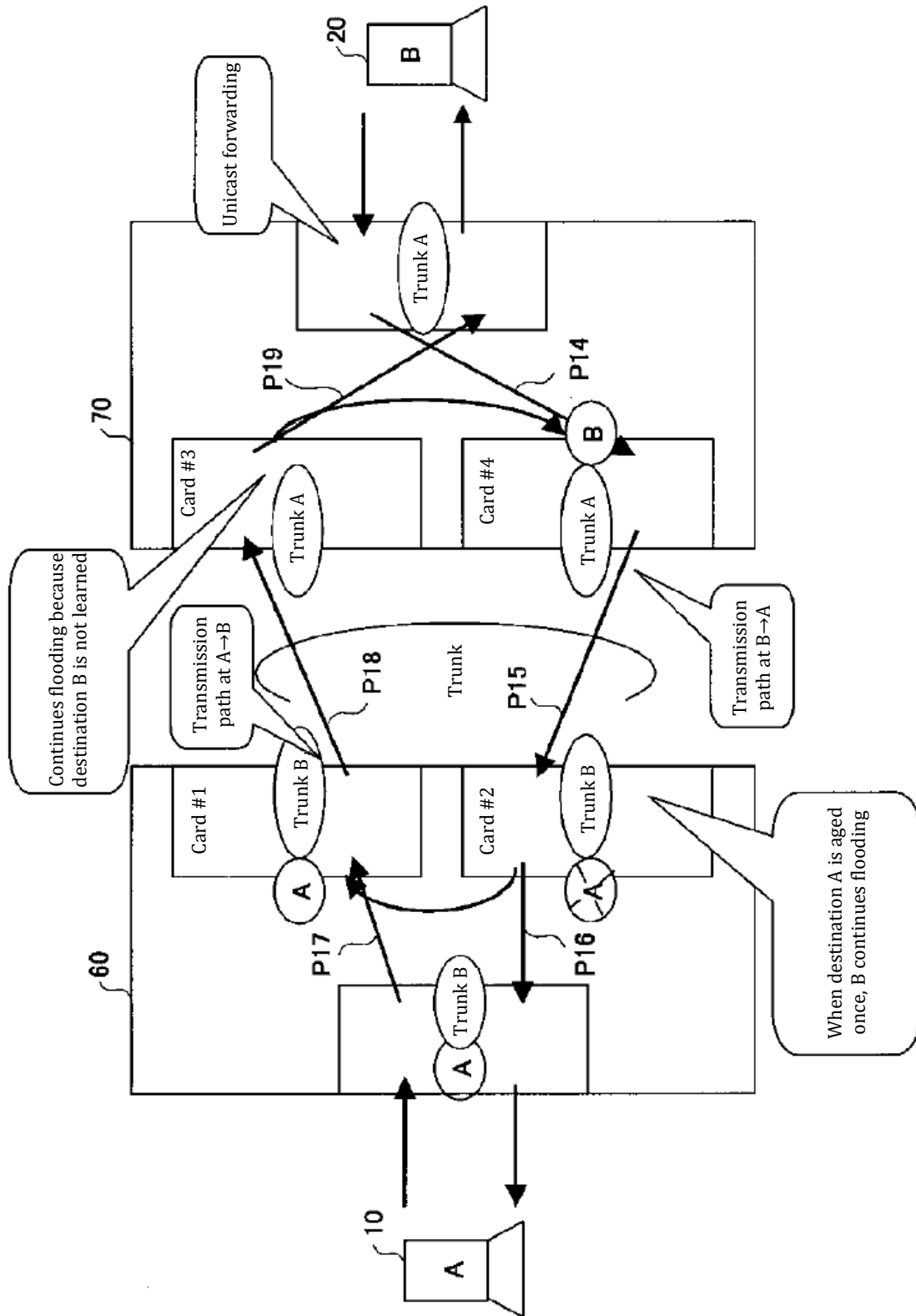
[FIG. 11]

Operation transition diagram for describing one example of conventional link aggregation function
(part 2)

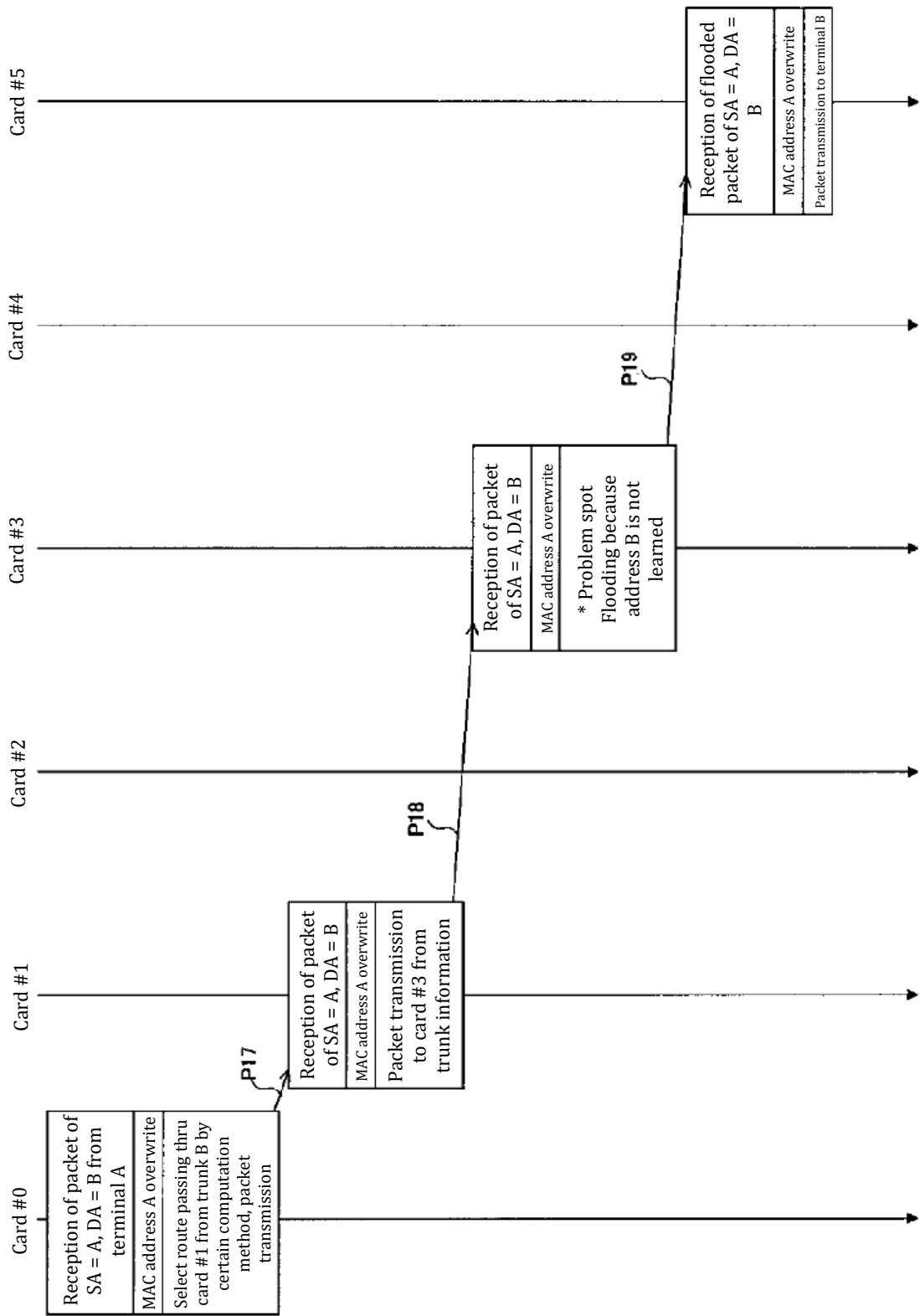


[FIG. 12]

Block diagram for describing problems due to repeated flooding according to one conventional example

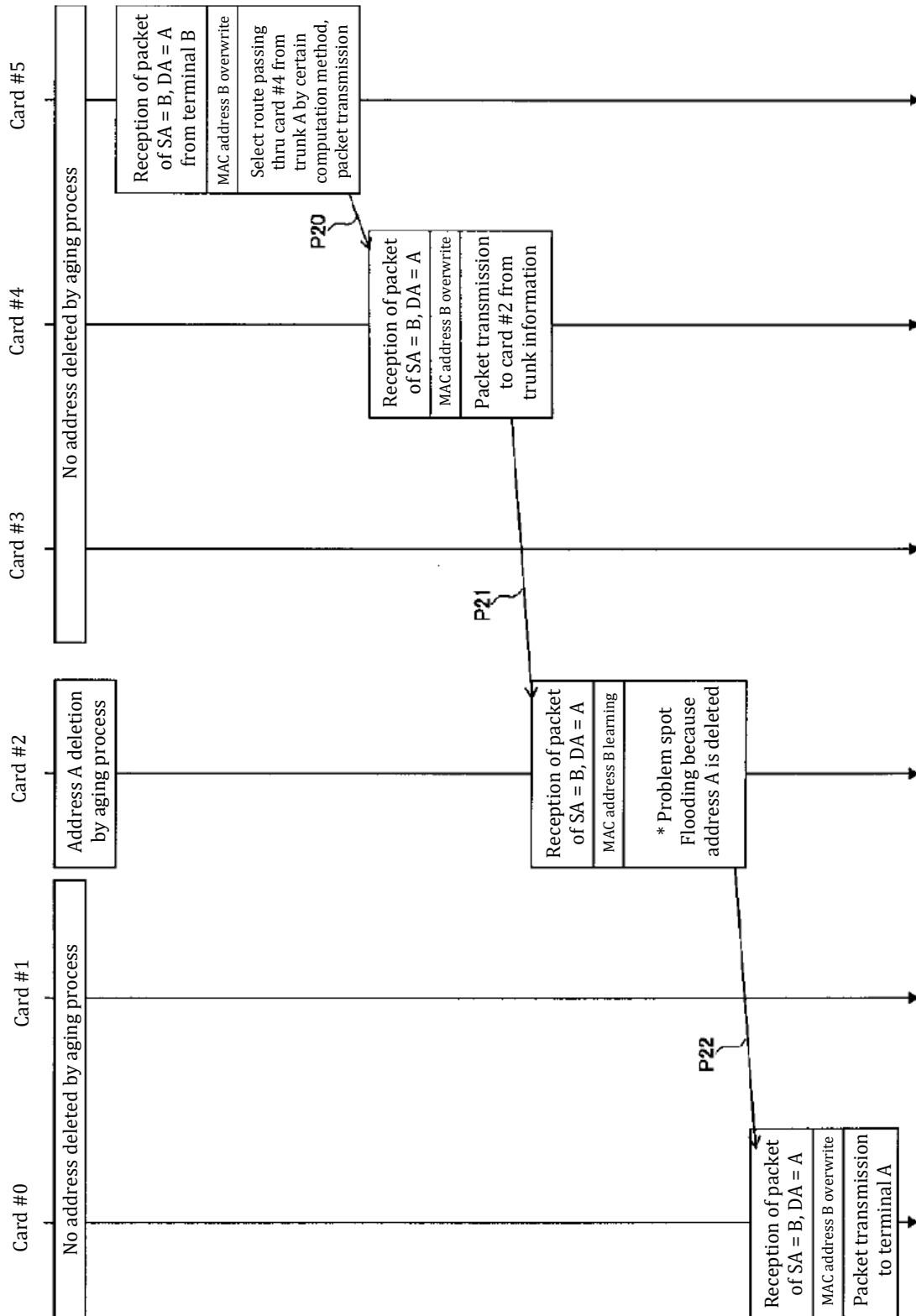


[FIG. 13]
Operation transition diagram for describing one example of conventional link aggregation function
(part 3)



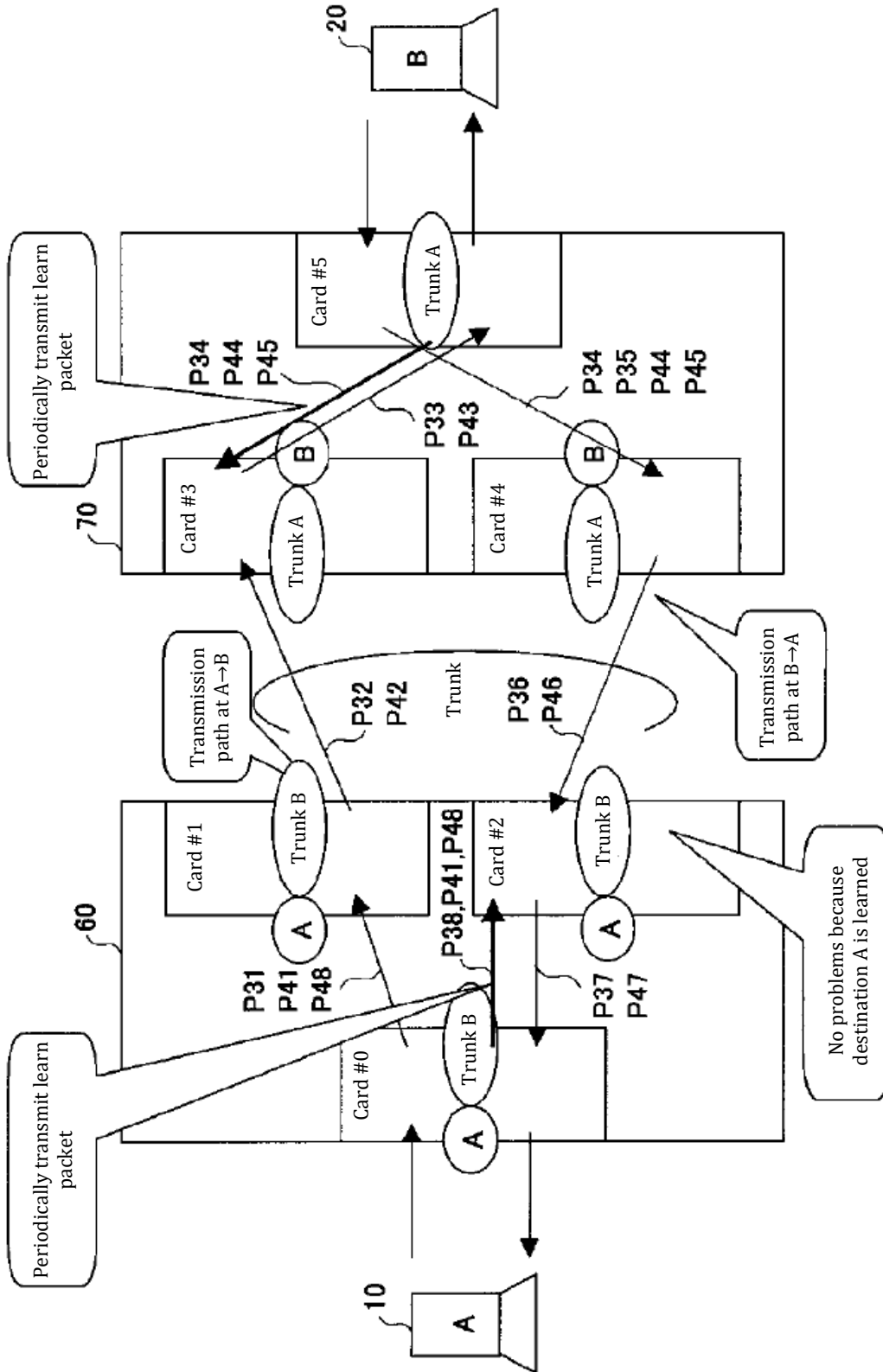
[FIG. 14]

Operation transition diagram for describing one example of conventional link aggregation function (part 4)



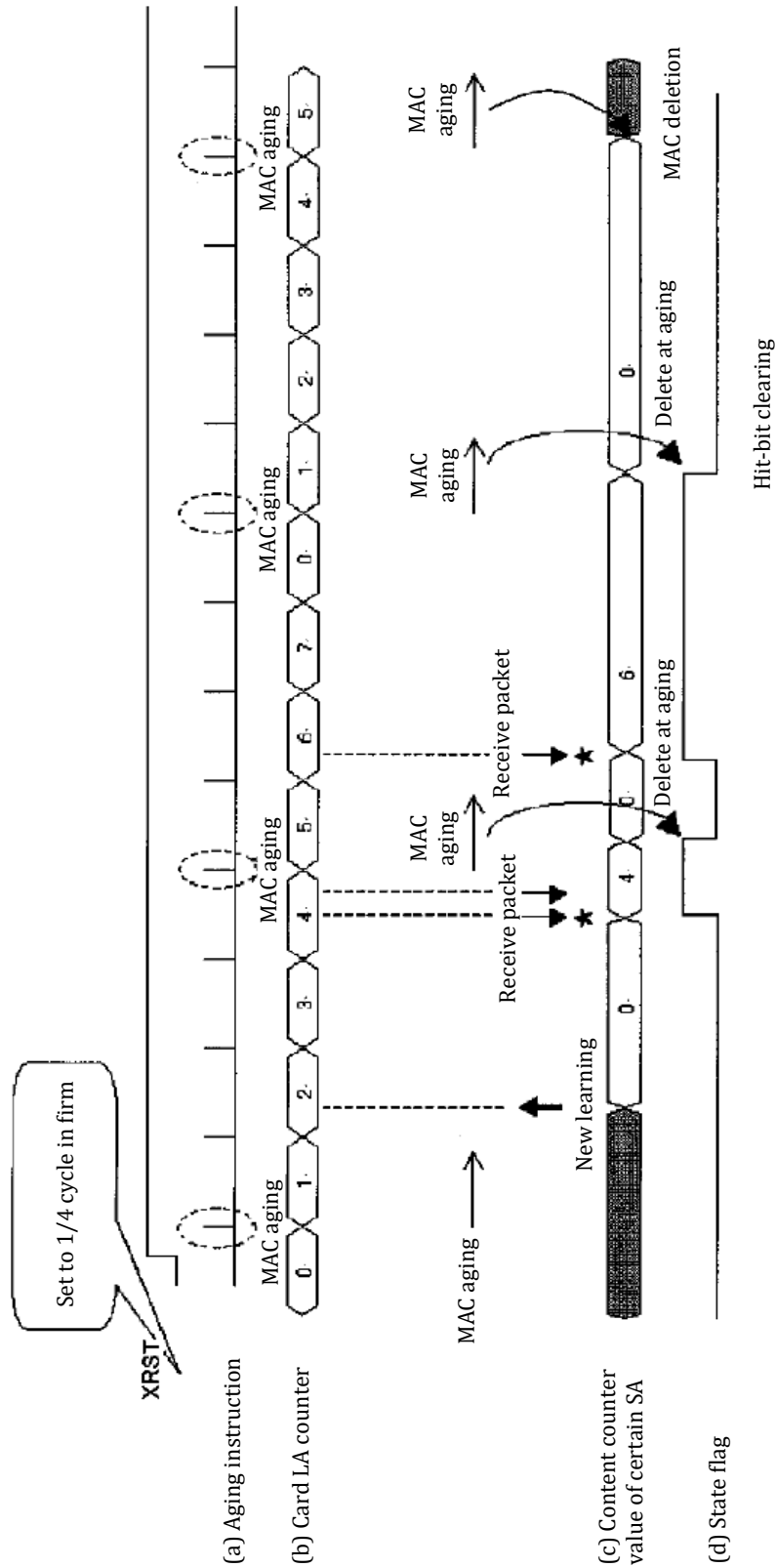
[FIG. 15]

Block diagram for describing operations of packet forwarding system of embodiment of present invention



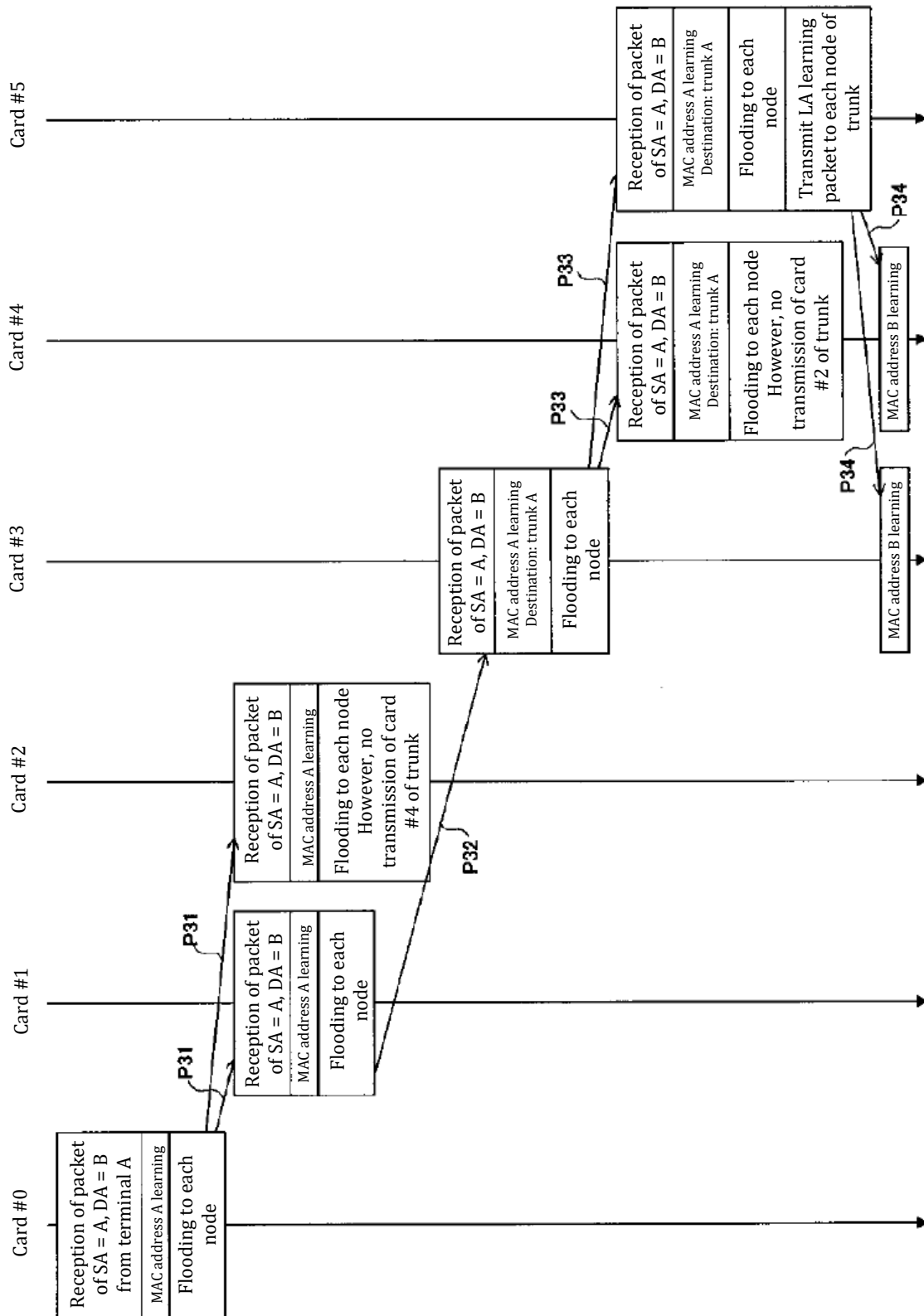
[FIG. 16]

Time-chart diagram for describing operations of packet forwarding device of embodiment of present invention

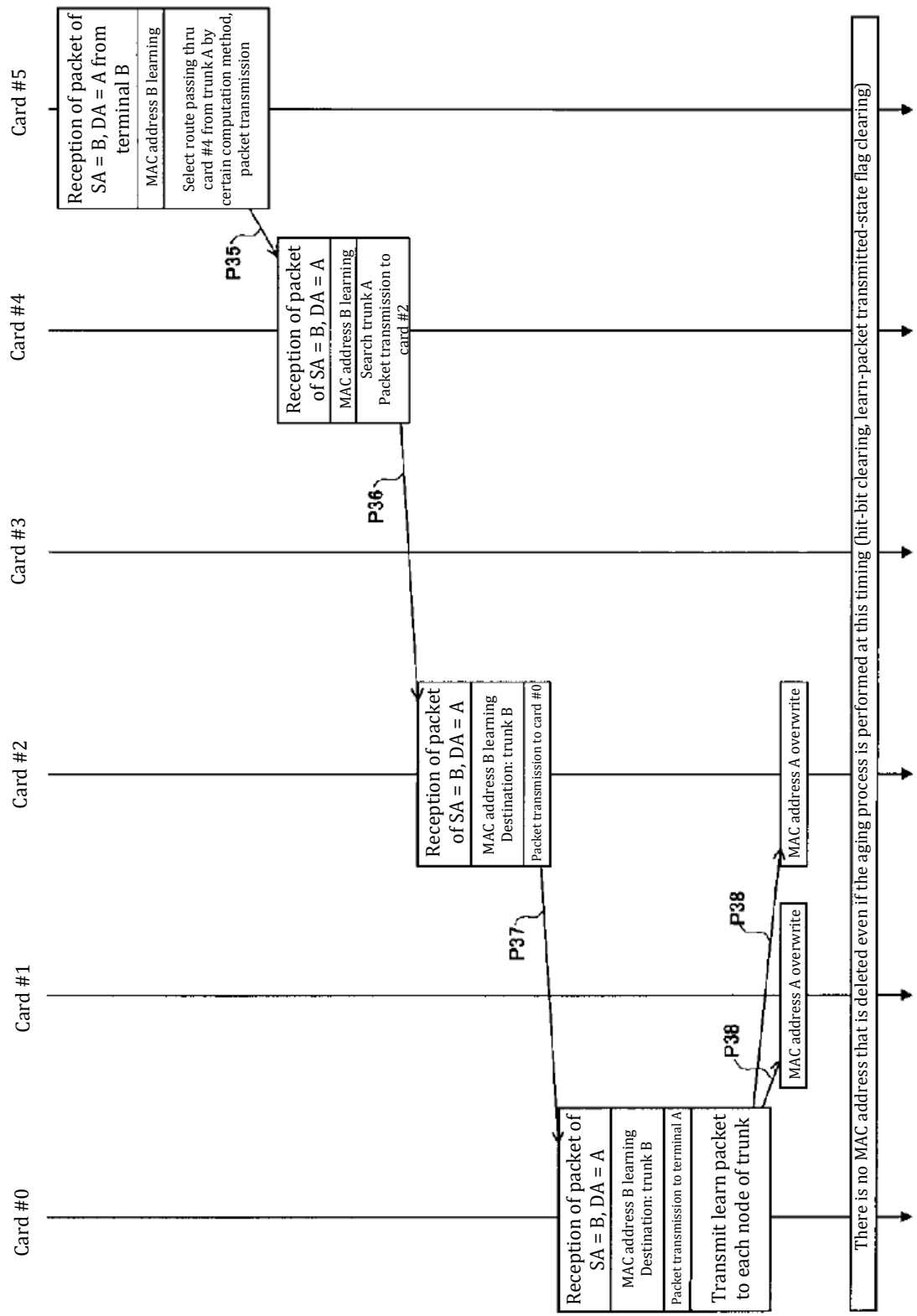


[FIG. 17]

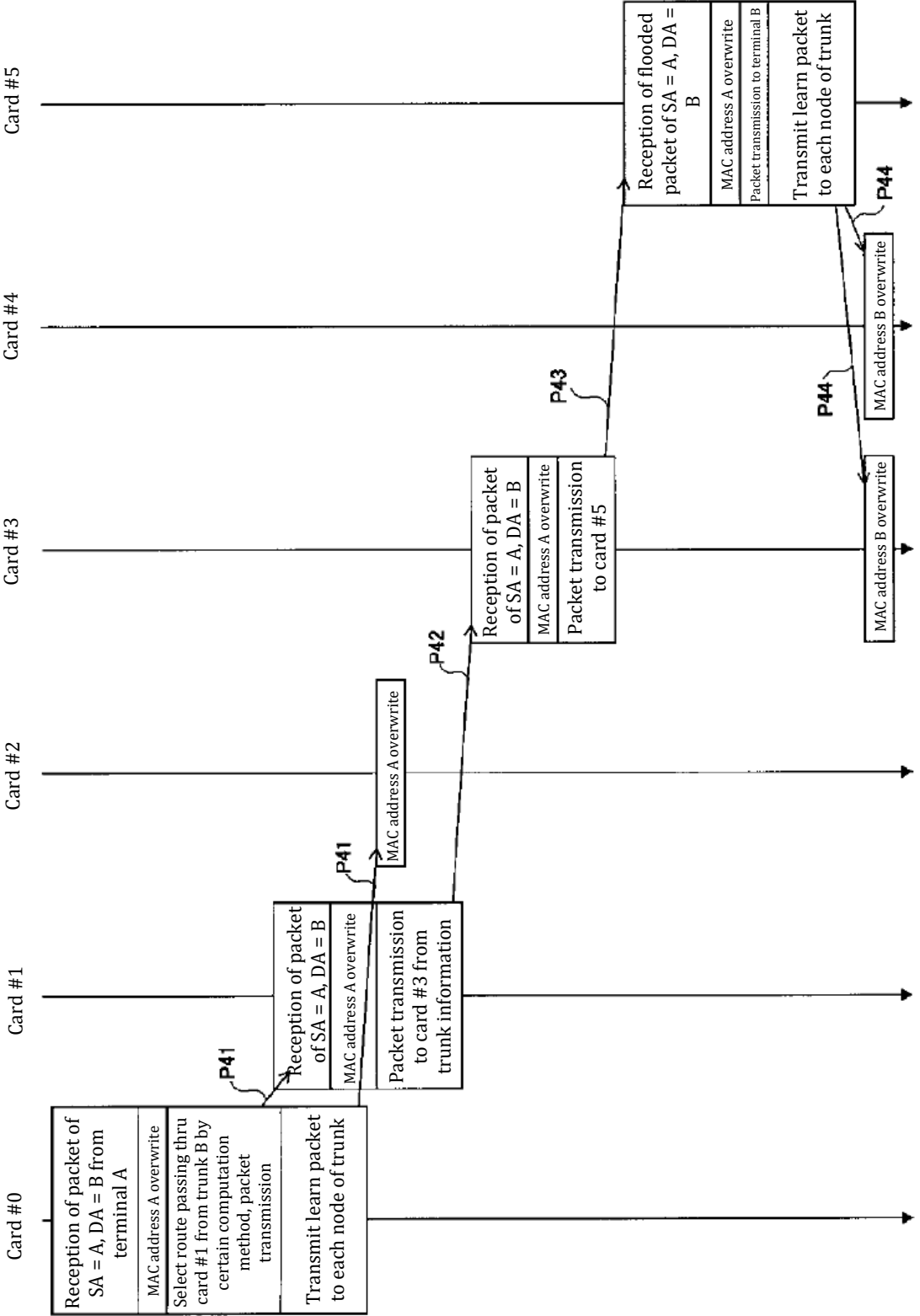
Operation transition diagram for describing operations of packet forwarding system of embodiment of present invention (part 1)



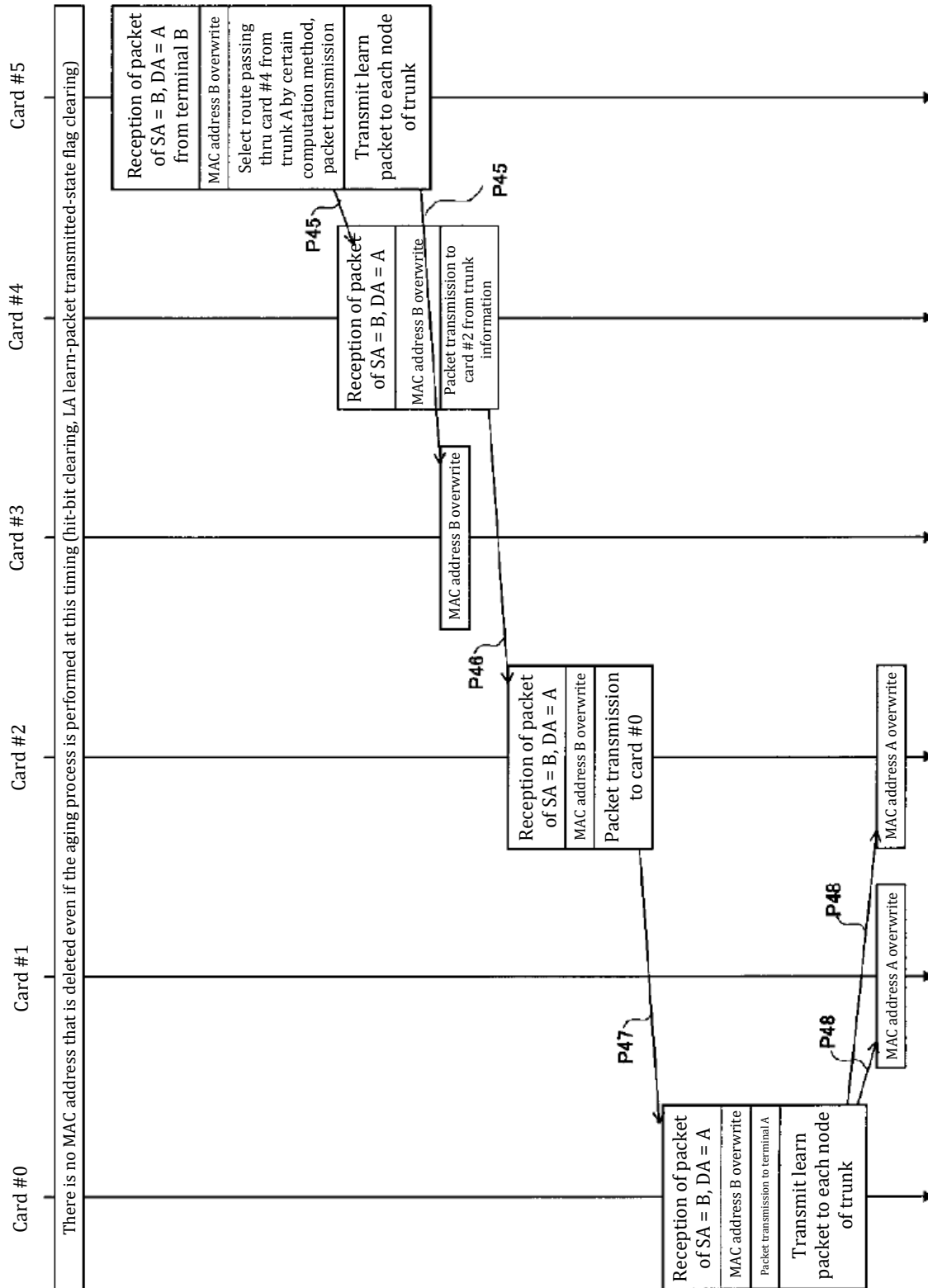
[FIG. 18]
Operation transition diagram for describing operations of packet forwarding system of
embodiment of present invention (part 2)



[FIG. 19]
Operation transition diagram for describing operations of packet forwarding system of
embodiment of present invention (part 3)



Operation transition diagram for describing operations of packet forwarding system of embodiment of present invention (part 4)



Continued from Front Page

(72) Inventor	Yoshimasa Itsuki c/o Fujitsu Digital Technology Co. Ltd. 2-3-9 Shin'yokahama, Kōhoku-ku, Yokohama-shi, Kanagawa-ken
(72) Inventor	Hidenori Nakaishi c/o Fujitsu Digital Technology Co. Ltd. 2-3-9 Shin'yokahama, Kōhoku-ku, Yokohama-shi, Kanagawa-ken
(72) Inventor	Masayo Terada c/o Fujitsu Digital Technology Co. Ltd. 2-3-9 Shin'yokahama, Kōhoku-ku, Yokohama-shi, Kanagawa-ken
(72) Inventor	Masami Yokoyama c/o Fujitsu Digital Technology Co. Ltd. 2-3-9 Shin'yokahama, Kōhoku-ku, Yokohama-shi, Kanagawa-ken
(72) Inventor	Toshinori Koyanagi c/o Fujitsu Ltd. 4-1-1 Kamikodanaka, Nakahara-ku, Kawasaki-shi, Kanagawa-ken
F Terms (Reference)	5K030 GA11 HA08 HD10 KA05 LB05