PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04L 12/56, H04Q 11/04

A1

(11) International Publication Number: **WO 99/48257**

(43) International Publication Date: 23 September 1999 (23.09.99)

(21) International Application Number: PCT/US99/05513

(22) International Filing Date: 12 March 1999 (12.03.99)

(30) Priority Data:
09/044,384 18 March 1998 (18.03.98) US

(71) Applicant: CONEXANT SYSTEMS, INC. [US/US]; 4311 Jamboree Road, Newport Beach, CA 92660–3095 (US).

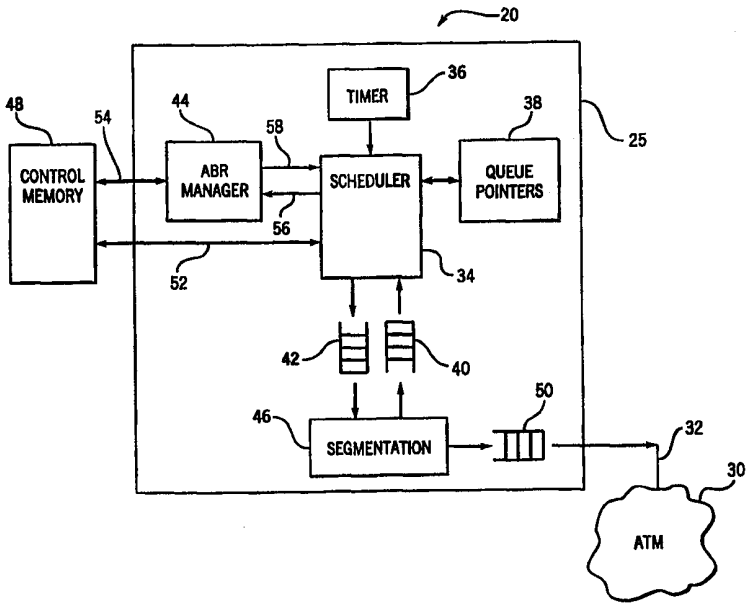
(72) Inventors: GEMAR, Jeff, R.; 2202 Bristol Street, Superior, CO 80027 (US). ANDREWS, Warner, B., Jr.; 754 16th Street, Boulder, CO 80302 (US).

(74) Agents: ZIEBERT, Joseph, N. et al.; Foley & Lardner, 777 East Wisconsin Avenue, Milwaukee, WI 53202–5367 (US).

(81) Designated States: JP, European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published
With international search report.

(54) Title: APPARATUS AND METHOD FOR SCHEDULING MULTIPLE AND SIMULTANEOUS CONNECTIONS IN A COMMUNICATION SYSTEM



(57) Abstract

A traffic manager coupled to a communication system for scheduling transmission of information associated with a plurality of connections in the communication system. The traffic manager includes a priority queue, a connection data structure, and a scheduler. The connection data structure includes a guaranteed bit frame rate field for each connection. The scheduler is coupled to the connection data structure and the priority queue. The scheduler loads information from the data structure to the queue, wherein the scheduler shapes the connection for a minimum cell rate and allows use by the connection of excess bandwidth if the guaranteed bit rate field is set.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	**ES**	Spain	**LS**	Lesotho	**SI**	Slovenia
AM	Armenia	**FI**	Finland	**LT**	Lithuania	**SK**	Slovakia
AT	Austria	**FR**	France	**LU**	Luxembourg	**SN**	Senegal
AU	Australia	**GA**	Gabon	**LV**	Latvia	**SZ**	Swaziland
AZ	Azerbaijan	**GB**	United Kingdom	**MC**	Monaco	**TD**	Chad
BA	Bosnia and Herzegovina	**GE**	Georgia	**MD**	Republic of Moldova	**TG**	Togo
BB	Barbados	**GH**	Ghana	**MG**	Madagascar	**TJ**	Tajikistan
BE	Belgium	**GN**	Guinea	**MK**	The former Yugoslav Republic of Macedonia	**TM**	Turkmenistan
BF	Burkina Faso	**GR**	Greece			**TR**	Turkey
BG	Bulgaria	**HU**	Hungary	**ML**	Mali	**TT**	Trinidad and Tobago
BJ	Benin	**IE**	Ireland	**MN**	Mongolia	**UA**	Ukraine
BR	Brazil	**IL**	Israel	**MR**	Mauritania	**UG**	Uganda
BY	Belarus	**IS**	Iceland	**MW**	Malawi	**US**	United States of America
CA	Canada	**IT**	Italy	**MX**	Mexico	**UZ**	Uzbekistan
CF	Central African Republic	**JP**	Japan	**NE**	Niger	**VN**	Viet Nam
CG	Congo	**KE**	Kenya	**NL**	Netherlands	**YU**	Yugoslavia
CH	Switzerland	**KG**	Kyrgyzstan	**NO**	Norway	**ZW**	Zimbabwe
CI	Côte d'Ivoire	**KP**	Democratic People's Republic of Korea	**NZ**	New Zealand		
CM	Cameroon			**PL**	Poland		
CN	China	**KR**	Republic of Korea	**PT**	Portugal		
CU	Cuba	**KZ**	Kazakstan	**RO**	Romania		
CZ	Czech Republic	**LC**	Saint Lucia	**RU**	Russian Federation		
DE	Germany	**LI**	Liechtenstein	**SD**	Sudan		
DK	Denmark	**LK**	Sri Lanka	**SE**	Sweden		
EE	Estonia	**LR**	Liberia	**SG**	Singapore		

APPARATUS AND METHOD FOR SCHEDULING MULTIPLE AND SIMULTANEOUS CONNECTIONS IN A COMMUNICATION SYSTEM

FIELD OF THE INVENTION

The present invention relates to a method and to an apparatus for transferring information through a communication medium. More particularly, the present invention relates to a method and to an apparatus for scheduling the transmission of information across connections in a communication system.

BACKGROUND OF THE INVENTION

Various systems have been adopted to carry digitally-encoded signals for communication applications, such as, telephone, video, and data services. These systems are often connection-oriented packet mode transmission systems, such as, asynchronous transfer mode (ATM) systems, frame relay systems, X.25 systems, or other transmission systems. Connection-oriented systems (e.g., ATM systems) have been employed in private and public communication systems or networks (e.g., wide area networks (WANs)) to transfer packetized signals (e.g., data cells or protocol data units) across communication lines, such as, telephone lines, cables, optical fibers, air waves, satellite links, or other communication media.

As an example, the ATM system transfers the data cells or units across the ATM system via connections or channels. The data cells can represent voice, sound, video, graphics, data, or combinations thereof for use in computing or communication applications. The connections can be part of a single physical link carrying a number of logical connections or be a single isolated path. The connections are often classified as either virtual channel connections (VCC), or virtual path connections (VPC) and either permanent virtual connections (PVC), or switched virtual connections (SVC). Connections are generally defined by a source and a destination for the data cell.

ATM systems or networks are generally utilized to support the integration of high quality voice, video, and high speed data traffic which can have specific conformance and quality of service (QOS) requirements. A scheduler is employed to properly select connections for transmission in accordance with operational parameters, such as, QOS requirements. The scheduler organizes transmission of data cells associated with the connections to guarantee QOS requirements and protocol conformance at the network ingress point as well as to maximize bandwidth for the communication system.

ATM systems can utilize at least four types of service categories or modes of traffic for transmission of data cells on connections. These four types include the constant bit rate (CBR) mode, the variable bit rate (VBR) mode, the unspecified bit rate (UBR) mode, and the available bit rate (ABR) mode. Each of these modes have somewhat different transmission protocol requirements. For example, the CBR mode requires that a connection be able to send a specific number of cells or bits per second; the CBR mode must have a set end-to-end bandwidth. The CBR mode is often utilized for data sources, such as, voice or video circuits which are susceptible to cell delay variations (CDV). The VBR mode often utilizes a "single leaky bucket" algorithm (e.g., virtual scheduling algorithm) or "dual leaky bucket" algorithm. Although the VBR mode has somewhat more flexible timing requirements than the CBR mode, the VBR mode must meet timing requirements dictated by the single or dual bucket algorithms. The ABR mode requires the sending of data cells in response to feedback from the network.

To appropriately supply and maximize the effectiveness of the transmission of the CBR, UBR, ABR, and VBR modes of traffic on the network, a scheduler controls a segmentation coprocessor (e.g., segmentation and reassembly unit) to direct communication of data cells or data protocol units on the connections. Generally, the scheduler attempts to select particular connections for transmission at particular times to guarantee conformance to traffic contracts and to QOS at network ingress points. This selection process, or scheduling, selects various connections for transmission based upon operational parameters associated with each connection, with feedback from the network, and with traffic mode timing requirements.

Scheduling a large number of connections with particular timing and communication criteria can lead to complex scheduling problems. For example, various connections can need a particular capacity from the network, and the availability of the particular capacity can be impinged by congestion across the network. These scheduling

problems can be compounded by the specific requirements of CBR mode, VBR mode, UBR mode, and ABR modes of traffic.

Traditional network scheduling algorithms, such as, Token Ring and Ethernet utilize fairly simplified scheduling algorithms. The algorithms are typically based upon transmission schemes wherein each member of the network gets a fair share of the network bandwidth. These systems cannot handle QOS requirements or the four categories (e.g., CBR mode, VBR mode, UBR mode, and ABR mode) of traffic for TM systems.

Prior schedulers utilized in ATM systems access a fixed size schedule table to assist time scheduling of connections. The fixed schedule table is comprised of slots, each of which is assigned to particular connections. The scheduler processes each slot one-by-one and transmits the connection assigned to the slot being processed to the network. These prior schedulers are somewhat inflexible and prioritize CBR and VBR modes traffic. Further, the fixed schedule table requires that the CBR modes of traffic be sent a fixed rate or harmonic of a fixed rate. Further still, these prior schedulers often must send idle cells to maintain the drain rate for the connection.

Other prior schedulers utilize a first-in, first-out (FIFO) memory and a timer (e.g., hardware units) to appropriately send connections across the network. However, the number of hardware units is limited by circuit size design criteria and cannot be dynamically allocated as bandwidth becomes available. Each FIFO and timer must be dedicated to a particular rate of transmission. Therefore, the rates of transmission are limited. Additionally, these prior schedulers cannot prioritize VBR and CBR modes of traffic.

U.S. Patent Application Serial No. 08/759,057 filed December 2, 1996, by Lincoln, entitled "Scheduler Utilizing Dynamic Schedule Table" dynamically manages and prioritizes modes of traffic. The communication system can be arranged as an asynchronous transfer mode (ATM) system which can efficiently handle the CBR mode of traffic, the VBR mode of traffic, the UBR mode of traffic, and the ABR mode of traffic. Further, both the CBR mode of traffic and the VBR mode of traffic can be handled on a dynamic basis.

Conventional schedulers do not support a guaranteed frame rate (GFR) mode of traffic. The GFR mode can also be referred to as UBR plus mode. A scheduler with GFR mode guarantees each session a minimum rate of service while sharing excess bandwidth and providing a global maximum rate. One possible method for achieving a guaranteed frame rate would be to take half of the cells from every frame and pass them

through. This hypothetical method is flawed, through, because although it may result in a satisfactory minimum cell rate (MCR), complete frames (e.g., group of cells) are not passed and proper packet switched communication requires transfer of entire frames, not just parts or cells. The GFR mode ensures MCR for entire frames of data, rather than just cells of data. In this way, the GFR mode ensures that communication time is not wasted because only portion of frames (which have to be discarded) are communicated. Additionally, conventional schedulers cannot efficiently schedule a plurality of tunnel modes.

Thus, there is a need for a scheduler which can support a guaranteed frame rate (GFR). Additionally, there is a need for a scheduler for an asynchronous transfer mode (ATM) system which can support additional service classes, including the CBR mode of traffic, the VBR mode of traffic, the UBR mode of traffic, and the ABR mode of traffic. Further still, there is a need for a scheduler which can handle frame relay and ATM service, along with GFR mode. Even further still, there is a need for a scheduler which can provide a minimum frame rate and provide fair sharing among connections for excess bandwidth. Yet further, there is a need for a scheduler which limits the total GFR connections to a peak cell rate. Still further, there is a need for a scheduler which utilizes VBR shaping and priority queues to obtain a MCR. Yet even further, there is a need for a scheduler which can service two or more tunnel modes so parallel traffic shaping and prioritized tunnel connections, such as tunnel within a tunnel, can be achieved.

SUMMARY OF THE INVENTION

The present invention relates to a traffic manager coupled to a communication system for scheduling transmission of information associated with a plurality of connections in the communication system. The traffic manager includes a priority queue, a connection data structure, and a scheduler. The connection data structure includes a guaranteed bit rate field for each connection. The scheduler is coupled to the connection data structure and the priority queue. The scheduler loads information from the data structure to the queue, wherein the scheduler shapes the connection for a minimum cell rate and allows use by the connection of excess bandwidth if the guaranteed bit rate field is set.

The present invention still further relates to a communication system including a control memory and scheduler. The control memory includes a priority queue and a connection data structure. The priority queue has at least one tunnel entry. The

connection data structure includes a guaranteed bit rate field for each connection. Each connection uses a particular bit rate mode of traffic. Particular bit rate mode parameters for the connection are included in the connection data structure. The scheduler is coupled to the connection data structure. The scheduler schedules a plurality of connections from the at least one tunnel entry to select a connection for eventual transmission. The scheduler also dynamically reschedules the connection in response to a timing parameter, the particular bit rate mode parameters for the connection, and the guaranteed bit rate field.

The present invention further relates to a method of scheduling connections for transmission on a communication network having a first tunnel and a second tunnel. In the method, the scheduler increments a first tunnel counter associated with the first tunnel when a connection is processed. Then, the scheduler sets the first tunnel active when the first tunnel counter is above a first threshold. The scheduler also increments a second tunnel counter associated with the second tunnel when a connection is processed. Then, the scheduler sets the second tunnel active when the second tunnel counter is above a second threshold. Finally, the scheduler determines whether the first and second tunnels are processed.

The present invention still further relates to a traffic manager coupled to a communication system for scheduling transmission of information associated with a plurality of connections in the communication system. The traffic manager includes a schedule table, a priority queue, and a scheduler. The schedule table includes a plurality of slots. The priority queue includes at least one tunnel entry. The scheduler is coupled to the schedule table. The scheduler processes the slots. The scheduler activates the tunnel when a number of slots are processed where the number exceeds a threshold. The scheduler provides a connection associated with the tunnel entry for transmission of information when the tunnel entry is activated.

According to one exemplary aspect of the present invention, a scheduler provides a guaranteed frame rate (GFR). With GFR, each communication session is guaranteed a minimum rate of service while fairly sharing excess bandwidth and providing global maximum rate. Also, the scheduler supports a minimum cell rate (MCR) using priority bumping, where priority levels for communication states are adjusted before scheduling. Parallel traffic shaping is also provided to accommodate simultaneous scheduling to MCR with scheduling of excess bandwidth.

According to yet another exemplary aspect of the present invention, a scheduler provides GFR and supports other service classes, including the CBR mode of

traffic, VBR mode of traffic, the UBR mode of traffic, and the ABR mode of traffic. Scheduling modes of operation can include but are not limited to, guaranteed frame rate (GFR), constant bit rate (CBR). Scheduling modes of operation can include, but are not limited to, guaranteed frame rate (GFR), constant bit rate (CBR). Further, the scheduler provides tunnels for any service class (UBR, VBR single and dual, ABR and GFR). The present invention further relates to a method of scheduling connections for transmission on a communication network having a first tunnel and a second tunnel. In the method a first counter associated with the first tunnel is incremented when a connection is processed. Then, the scheduler sets the first tunnel active when the first tunnel counter is above a first threshold. The scheduler also increments a second tunnel counter associated with the second tunnel when a connection is processed. Then, the scheduler sets the second tunnel active when the second tunnel counter is above a second threshold. Finally, the scheduler determines whether the first and second tunnels are processed.

The present invention still further relates to a traffic manager coupled to a communication system for scheduling transmission of information associated with a plurality of connections in the communication system. The traffic manager includes a schedule table, a priority queue, and a scheduler. The schedule table includes a plurality of slots. The priority queue includes at least one tunnel entry. The scheduler is coupled to the schedule table. The scheduler processes the slots. The scheduler provides a connection associated with the tunnel entry for transmission of information when the tunnel entry is activated.

In the present application, the term communicate refers to any of the following: receive, provide, transmit, obtain, send, input, output, or convey.

BRIEF DESCRIPTION OF THE DRAWINGS

Preferred exemplary embodiments of the invention will hereinafter be described in conjunction with the appended drawings, wherein like numerals denote like elements and:

FIGURE 1 is a schematic block diagram of a traffic class manager system coupled to an asynchronous transfer mode communication system and a control memory in accordance with an exemplary embodiment of the present invention;

FIGURE 2 is an exemplary schematic block diagram of a priority queue for the traffic class system illustrated in FIGURE 1 in accordance with another exemplary embodiment of the present invention;

FIGURE 3 is a schematic block diagram of a pair of linked lists defined by the priority queue illustrated in FIGURE 2;

FIGURE 4 is a schematic block diagram of a connection state data structure in the control memory for use with the traffic class manager system illustrated in FIGURE 1 in accordance with another exemplary embodiment of the present invention;

FIGURE 5 is an exemplary schematic block diagram of a dynamic schedule table in the control memory for use with the traffic class manager system illustrated in FIGURE 1 in accordance with yet another exemplary embodiment of the present invention;

FIGURE 6 is a top level flow chart showing the general operation of the traffic class manager system illustrated in FIGURE 1 in accordance with still another embodiment of the present invention;

FIGURE 7 is a flow chart showing the link-out of-rate operation illustrated in FIGURE 6:

FIGURE 8A-D include a flow chart showing the schedule connection operation illustrated in FIGURE 6;

FIGURE 9 is a flow chart showing the process slot operation illustrated in FIGURE 6;

FIGURE 10 is a flow chart showing the process slot tail queue number operation illustrated in FIGURE 9;

FIGURES 11A-C include a flow chart showing the send cell operation illustrated in FIGURE 6;

FIGURES 12A-C include a flow chart showing the schedule rate operation illustrated in FIGURE 11C; and

FIGURE 13 is a flowchart showing the tunnel activation mechanism for the traffic class system illustrated in FIGURE 1 in accordance with yet another exemplary embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

With reference to FIGURE 1, a communication system 20 includes a traffic manager system or unit 25 coupled to an ATM system or network 30 by a communication line 32. Unit 25 includes a scheduler 34, a timer 36, a global priority queue 38, a start FIFO memory 40, a schedule FIFO memory 42, an available bit rate (ABR) manager 44, a segmentation unit 46, and a transmit FIFO memory 50. Scheduler 34 of unit 25 is coupled to a control memory 48 via bus 52.

Control memory 48 is coupled to ABR manager 44 of unit 25 by a bus 54. Scheduler 34 is coupled to timer 36, global priority queue 38, start FIFO memory 40, schedule FIFO memory 42, and ABR manager 44. Scheduler 34 is coupled to ABR manager 44 by a control bus 56 and by a control bus 58. Segmentation unit 46 is coupled to start FIFO memory 40, schedule FIFO memory 42, and transmit FIFO memory 50. Transmit FIFO memory 50 is coupled to communication line 32.

Control memory 48 can be any memory unit, such as, a semiconductor memory or other storage facility. Queue 38 is preferably a semiconductor memory (e.g., register or static random access memory) device integrated with scheduler 34 for fast transfer of information. Unit 25 is preferably integrated on a single semiconductor substrate, including processing units, state machines, registers, and other circuits for implementing the functions and operations of scheduler 34, timer 36, global priority queue 38, start FIFO memory 40, schedule FIFO memory 42, ABR manager 44, segmentation unit 46, and transmit FIFO memory 50.

Traffic manager unit 25 preferably provides data packets or data cells as protocol data units (PDU) to ATM network 30. In general, scheduler 34 is a state machine which responds to timing signals from timer 36, control signals from ABR manager 44 across control bus 58, connection information and operational parameters for each connection stored in control memory 48, and connection information in start FIFO memory 40 to place connection information in global priority queue 38. Scheduler 34 selects the highest priority connection information in queue 38 and places that information in schedule FIFO memory 42 for eventual transmission through network 30.

Segmentation unit 46 receives the connection information stored in schedule FIFO memory 42. Segmentation unit 46 utilizes the connection information to place data cells associated with the connection information in transmit FIFO memory 50. The data cells are transmitted from FIFO memory 50 to line 32 and then to network 30. Segmentation unit 46 generally organizes the data cells into protocol data units (PDUs) for transmission on the network 30.

Scheduler 34 can also directly place connection information from control memory 48 into schedule FIFO memory 42. For example, scheduler 34 places connection information for the CBR mode of traffic directly from memory 48 to FIFO memory 42 without accessing queue 38. In this way, scheduler 34 ensures that the CBR mode of traffic is sent in accordance with its strict timing requirements.

ABR manager 44 is a state machine which receives feedback from network 30 through control bus 56 from scheduler 34 and requests that scheduler 34 schedule a connection for the ABR mode of traffic in response to the feedback and to the parameters for the particular connection. Manager 44 implements an ABR flow control algorithm to process the feedback based upon user-programmable parameters and connection parameters. In general, ABR manager 44 indicates that a connection in the ABR mode of traffic should be scheduled and the types of cells which should be sent as part of the traffic via control bus 58. For example, ATM network 30 preferably handles three types of ABR cells: data cells, forward resource management (RM) cells, and backward RM cells. ABR manager 44 indicates the type of cell which scheduler 34 should schedule and the mode of traffic (e.g., an out-of-rate transmission or an in-rate transmission).

The in-rate transmission is scheduled by scheduler 34 as a "single bucket" VBR mode of traffic transmission, and the out-of-rate transmission is scheduled utilizing a dedicated priority level of queue 38. Generally, out-of-rate transmission is used for both out-of-rate forward RM cells that are requested at a low rate by ABR manager 44 and for out-of-rate backward RM cells that are requested by ABR manager 44 based on received traffic from network 30. A connection can send out-of-rate cells at the same time the connection is using VBR scheduling to send in-rate cells.

ABR manager 44 can also adjust the rate for a connection utilizing the ABR mode of traffic in response to feedback. Manager 44 can write new I and L parameter which are stored in memory 48 to adjust the rate according to a bucket algorithm. In this way, dynamic rate scheduling for ABR modes of traffic with fine granularity can be achieved.

Start FIFO memory 40 holds connection information related to data cells that have been received from a host or data source (not shown). Start FIFO memory 40 is processed by scheduler 34 to add connections to global priority queue 38 for eventual placement in schedule FIFO memory 42 and for eventual transmission on network 30. The connection information is the indication of the connection number of the data received by segmentation unit 46.

With reference to FIGURE 2, global priority queue 38 is comprised of a head pointer stored in a column 60 and a tail pointer stored in a column 62. Queue 38 stores head pointers and tail pointers at priority levels 64A-J. Alternatively, queue 38 can store any number of priority locations. FIGURE 2 shows exemplary priority levels associated with particular modes of traffic and an exemplary table-type data structure. For

example, the out-of-rate UBR traffic mode is given the first priority (e.g. highest) at priority level 64A, and the interval tunnel mode 64J of traffic is given the last priority, level 64J, (e.g., lowest) in queue 38. The head pointer and tail pointer stored in columns 60 and 62, respectively, are simply an address or information which indicates a connection associated with the transmission. The head pointer and the tail pointer generally direct scheduler 34 to memory locations in control memory 48 which are described in more detail below with reference to FIGURE 4. Alternatively, interval tunnel level 64J can be a number of distinct tunnel levels.

To advantageously save memory space in queue 38 and to allow faster data transfer from scheduler 34 to queue 38, only the head pointer and the tail pointer are stored in global priority queue 38 for each of priority levels 64A-J. As an example, global priority queue 38 is shown including ten priority levels 64A-J that can be utilized. The priority levels for ABR, VBR, and UBR modes of traffic can be set by the user.

The head pointers and tail pointers can define a linked list of a number of connections for each of priority levels 64A-J. Exemplary linked lists are shown in FIGURE 3. The linked lists are used to indicate a number of connections at the particular priority level of levels 64A-J in queue 38 without using a large amount of memory. Thus, queue 38 can indicate a large list of connections, even though only a head pointer and tail pointer are stored in queue 38.

Generally, priority queue 38 includes a register (not shown) which indicates when the head pointer in column 60 includes information. In this way, scheduler 34 can simply check the register for each priority level of levels 64A-J rather than addressing priority queue 38 to determine if a head pointer is available in column 60. In this way, scheduler 34 saves time, as only one bit in the register needs to be checked to determine if a head pointer is present.

With reference to FIGURE 3, a linked list 64 of connections for VBR that moves out traffic is defined by a head pointer in column 60 of queue 38 and a tail pointer in column 62 of queue 38. The head pointer provides an address or indication of a connection state 66A in control memory 48 (FIGURE 1). Connection state 66A includes a pointer which points to a next connection state 66B stored in control memory 48. Connection state 66B includes a pointer which points to connection state 66C stored in control memory 48. The tail pointer stored in column 62 of queue 38 also points to connection state 66C. Therefore, for linked list 64, the head pointer in column 60 stores

connection state 66A for the head of list 64 and the tail pointer in column 62 stores the connection state 66C for the end of list 64.

Similarly, a circular linked list 68 is shown in FIGURE 3 which is defined by the head pointer in column 60 and the tail pointer in column 62. The head pointer in column 60 points to a first connection state 70A in circular linked list 68. Similar to linked list 64, state 70A includes a pointer which points to a connection state 70B, and connection state 70B includes a pointer which points to a connection state 70C. Connection states 70A-C are stored in control memory 48. Additionally, connection state 70C includes a pointer which points back to connection state 70A so that a circular buffer or linked list is defined. The tail pointer in column 62 also points to connection state 70C.

Linked list 64 is typically used for VBR modes of traffic, and linked list 68 is typically used for UBR modes of traffic. Circular linked list 68 facilitates round-robin orientation of connections in queue 38 which are utilized by UBR connections. In UBR modes of traffic, the connection is typically not removed after transmission. Since the connection utilizing the VBR modes of traffic are typically removed from queue 38 after transmission, there is no advantage in maintaining a circular linked list such as link 68. The use of the head pointer in column 60 and tail pointer 62 allows easy access to both ends of lists 64 and 68. For example, if connections need to be added to linked list 64, the tail pointer provides directions to connection state 66C which is modified to point to the additional connection state (not shown). The tail pointer is then pointed to the additional connection state. For circular linked list 68, state 70C is modified to point to the additional state, and the additional state includes a pointer to state 70A. The tail pointer is amended to point to the additional state.

With reference to FIGURE 4, control memory 48 includes a data structure or connection state 90 which includes storage locations or fields 92A-T. Control memory 48 includes a connection state 90 for each connection associated with traffic class manager unit 25. Each state 90 represents a connection in network 30. State 90 in FIGURE 4 represents the data structure of states 66A-C and 70A-C in FIGURE 3.

A data available field 92A indicates whether data for the particular connection is available for transmission on network 30. A last CLP field 92B includes a cell loss priority (CLP) parameter which indicates the value of the cell loss priority for the most recently transmitted cell of the connection. A scheduled field 92C indicates that the connection is presently linked to a dynamic schedule table 88 (FIGURE 5) within memory 48 or is part of a linked list, such as, lists 64 and 68 (FIGURE 3) defined in global priority

queue 38. A schedule mode field 92D indicates whether the connection utilizes an UBR, a CBR, a VBR1 (e.g., single bucket VBR), a VBR2 (e.g., dual bucket VBR), a VBRC (dual bucket CLP control), a GFR or an ABR mode of traffic.

A priority field 92E includes a priority parameter which indicates the scheduling priority of the connection. The priority parameter corresponds to priority levels 64A-J of queue 38 (FIGURE 2) and provides an indication as to which priority level of levels 64A-J in priority queue 38 the connection should be stored. For GFR mode, priority field 92E indicates the GFR-VBR priority level 64E which is used to shape the connection to the specified MCR.

A queue pointer field 92F stores a pointer for linking lists defined in dynamic schedule table 88 or in priority queue 38. The queue pointer for the connection is utilized in lists for VBR modes of traffic, such as, list 64 and in lists for the UBR modes of traffic, such as, list 68. A VBR reset field 92G stores a reset parameter for the VBR bucket algorithms (e.g., single bucket, dual bucket, or CLP bucket). The CFR mode of traffic preferably utilizes the single leaky bucket algorithm.

An I1 field 92H includes an I1 parameter for the first VBR bucket algorithm, and an L1 field 921 provides an L1 parameter for the first VBR bucket algorithm. An I2 field 92J provides an I2 parameter for the second VBR bucket algorithm, and an L2 field 92K provides an L2 parameter for the second VBR bucket algorithm.

A present field 92L provides a present parameter indicative of the time of the last VBR mode of traffic transmission for the connection. A delta 1 field 92M provides a delta 1 parameter indicative of the difference between the desired first bucket next transmission time and the last transmission time for VBR modes of traffic. A delta 2 field 92N provides a delta 2 parameter indicative of the difference between the desired second bucket next transmission time and the last transmission time for VBR modes of traffic.

A minimum cell rate (MCR) priority field 920 provides an MCR priority parameter for temporarily increasing the priority of the connection to provide MCR guarantees for ABR and GFR modes of traffic. An MCR limit field 92P provides an MCR limit parameter indicative of a maximum time interval between connection cell transmissions before the MCR priority parameter is set.

An out-of-rate priority field 92Q provides a scheduling priority parameter for out-of-rate cells for ABR modes of traffic. An out-of-rate pointer field 92R provides an out-of-rate pointer to a connection state 90. Similar to the queue pointer in field 92F, the out-of-rate pointer is used to build linked lists, such as, list 68 (FIGURE 3).

A GFR priority field 92S provides a second scheduling priority parameter for UBR portion of GFR service. A GFR queue pointer 92T provides a pointer to link connection to global priority queue used to transmit GFR traffic above the MCR.

With reference to FIGURE 5, dynamic schedule table 88 is located in memory 48 (FIGURE 1). Schedule table 88 is comprised of a number of slots 88A-G (e.g., from slot 1 to M where M is a user-programmable parameter). Slots 88A-G are each comprised of a CBR entry 94, a tunnel entry 95, and slot tail pointer entries 96A-D. Each of slots 88A-G can be configured to include a number of slot tail pointer entries 96A-D from 1 to N. Each of entries 96A-D is associated with a particular VBR priority level of levels 64A-J in queue 38 (FIGURE 2). The VBR priority levels 1-N for entries 96A-D are mapped to global priority levels 64A-J via a fixed offset (although an arbitrary mapping is also possible).

CBR entry 94 stores connection information if a CBR mode of traffic connection should be transmitted. For example, when scheduler 34 processes slot 88C, scheduler 34 reads CBR entry 94, and if the connection information is present, scheduler 34 places the connection information in schedule FIFO memory 42.

Tunnel entry 95 is used to indicate that one of a group of connections should be transmitted in round-robin order in the CBR mode of traffic. In tunnel transmission mode, scheduler 34 reads entry 95 and activates tunnel priority level 64I in queue 38. Queue 38 includes a header or flag bit which is set when tunnel priority level 64I is active. Columns 60 and 62 for the tunnel priority level 64I define a linked list (not shown) similar to linked list 64 or 68 (FIGURE 3).

Scheduler 34 sends the connection at the head of the linked list defined by level 64I of queue 38 to schedule FIFO memory 42 the next time the queue 38 is accessed. Priority level 64I is accessed only when active even if the head pointer in column 60 has a connection entry. When active, the priority level 64I is accessed in the normal priority order. When the next slot of slots 88A-G that has an entry 95 is processed, scheduler 34 sets level 64I active, and the next connection in the circular linked list of the tunnel priority level 64I is selected. Priority level 64I is set inactive after this selection. In this way, connection for the tunnel mode transmission is processed in a round-robin order.

Slot tail pointer entries 96A-D provide slot tail pointers which indicate a connection state at the tail of a linked list (not shown). For example, when slot 88C is processed, scheduler 34 reads the slot tail pointer in entry 96A arid addresses a connection state 96A (having the structure of connection state 90 in FIGURE 4). State 96A includes a

queue pointer field 92F (FIGURE 4) which points to a state 98B (e.g., the head of linked list 98). Similarly, state 98B points to a state 98C which points to state 98A. The head of linked list 98 is reached by following the tail entry 96A to the tail of the list (state 98A) and then following the queue pointer in field 92F of state 98A to the head of list 98 (e.g., state 98B). Entries 96A-D are used for VBR, ABR and GFR modes of traffic.

When scheduler 34 reads entry 96D, scheduler 34 is directed to a linked list 97. List 97 is comprised of one connection state 97A. The queue pointer in field 92F of state 97A points to itself (connection state 97A). Thus, the tail pointer in entry 96D and the queue pointer in field 92F of state 97A are the same.

Dynamic schedule table 88 can be programmed to be various sizes. The capability allows scheduler 34 to customize table 88 for various CBR modes of traffic. Since the size of table 88 is variable, scheduler 34 is more precisely able to schedule specific rates for CBR modes of traffic.

With reference to FIGURES 6-12, the notation A6B means the field B (fields 92A-T) in connection state 90 indicated by the pointer A. With reference to FIGURE 6, scheduler 34 operates according to a top level flow chart 300. Scheduler 34 determines if an out-of-rate ABR connection must be scheduled at a step 302. At step 302, scheduler 34 determines if a signal from ABR manager 44 has been provided on control bus 58 indicating that an out-of-rate ABR mode of traffic connection must be scheduled.

Connections may need to be scheduled for out-of-rate transmissions of RM cells. Out-of-rate scheduling is used for both out-of-rate forward RM cells that are requested at a low rate by manager 44 and for out-of-rate backward RM cells that are requested by manager 44 based on traffic received from ATM network 30. If scheduler 34 has received a signal from manager 44 that an out-of-rate ABR connection should be scheduled, scheduler 34 performs a flow chart 325 described with reference to FIGURE 7.

If an out-of-rate transmission does not need to be scheduled, scheduler 34 advances from step 302 to a step 304. After flow chart 325 is completed, scheduler 34 advances to step 304.

At step 304, scheduler 34 determines if start FIFO memory 40 is empty. If start FIFO memory 40 is empty, scheduler 34 advances to a step 306, as scheduler 34 does not have to process any connection information stored in FIFO memory 40. If start FIFO memory 40 is not empty (e.g., start FIFO memory 40 has received new connection information), scheduler 34 reads the connection information stored in start FIFO memory 40 at a step 308.

At a step 310, scheduler 34 determines if the connection associated with the connection information read from start FIFO memory 40 has been scheduled by reading scheduled field 92C of connection state 90 (FIGURE 4) associated with the connection. If the connection has been scheduled, scheduler 34 advances to step 306. If the connection has not been scheduled, scheduler 34 performs a schedule connection flow chart 350 described with reference to FIGURE 8A. After schedule connection flow chart 350 is completed, scheduler 34 advances to step 306.

At step 306, scheduler 34 determines if schedule FIFO memory 42 is not full and if a schedule difference parameter is greater than zero. The schedule difference parameter is derived from timing signals provided by timer 36. The schedule difference parameter is indicative of whether scheduler 34 has reached the appropriate slot at the appropriate real time. For example, the schedule difference parameter is representative of the actual slot position minus the desired slot position at a particular time. The schedule difference parameter is incremented whenever the amount of time for the processing of one of slots 88A-G is elapsed and is decremented whenever one of slots 88A-G is processed. Therefore, the schedule difference parameter keeps track of the time for processing slots 88A-G. If the schedule difference parameter is greater than zero (scheduler 34 is late), scheduler 34 should process a slot to bring the schedule difference parameter to zero (scheduler 34 is not late).

If schedule FIFO memory 42 is not full and the schedule difference parameter is greater than zero, scheduler 34 advances to a process slot flow chart 400 discussed with reference to FIGURE 9. After process slot flow chart 400 is completed, scheduler 34 advances to a step 308. At step 306, if either schedule FIFO memory 42 is full or if schedule difference is equal to zero, scheduler 34 advances to step 308. Scheduler 34 should not process a slot if schedule FIFO memory 42 is full because there is no place to store any additional connections in FIFO memory 42. Additionally, if the schedule difference parameter is equal to zero, the timing is not appropriate (e.g. it is too early) to process the slot. More particularly, the appropriate sending parameters for the connections could be violated if slots 88A-G are processed too early (e.g., when the schedule difference parameter is zero).

At a step 308, scheduler 34 determines if the schedule difference parameter is equal to zero and if global priority queue 38 is empty. When schedule difference parameter equals zero, the timing is appropriate for scheduler 34 to schedule a connection on schedule FIFO memory 42. If the schedule difference parameter is greater than zero,

scheduler 34 should not transmit cells until the scheduler 34 catches up (e.g., the schedule difference parameter equals zero). Priority isolation for UBR modes of connections is ensured by only processing queue 38 when the schedule difference parameter is equal to zero. Additionally, if there are no global priority queue entries in levels 64A-J of global priority queue 38, there are no connections available. Therefore, if the schedule difference parameter is greater than zero or there are no global priority queue entries, then scheduler 34 returns to step 302.

If the schedule difference parameter is zero and there is a global priority queue entry in global priority queue 38, scheduler 34 performs a send cell flow chart 500. After flow chart 500 is completed as discussed with reference to FIGURES 11A-C, scheduler 34 returns to step 302.

With reference to FIGURE 7, link out-of-rate flow chart 325 is utilized by scheduler 34 to link a connection to the out-of-rate priority level of global priority queue 38 (FIGURE 2, e.g., priority level 64A). At a step 326, scheduler 34 determines if the head pointer 60 at the out-of-rate priority level 64A of global priority queue 38 is null. If the head pointer is null, scheduler 34 advances to a step 328 and sets the head pointer in column 60 of queue 38 to the connection provided by ABR manager 44. After step 328, scheduler 34 advances to a step 330 and sets the out-of-rate pointer in field 92R in connection state 90 for the connection to the connection provided by ABR manager 44. After step 330, scheduler 34 advances to a step 340.

Steps 326, 328, and 330 build a circular linked list consisting of the connection provided by manager 44. If the head pointer is null, there are no out-of-rate connections stored in global priority queue 38, since out-of-rate connections are stored in circular linked lists, such as, list 68 (FIGURE 3). The UBR modes of traffic, such as, the out-of-rate mode have connections circularly linked to facilitate round-robin orientation of the connection in queue 38. If no other connections are in the out-of-rate queue, the new connection should be linked onto itself.

If the head pointer in column 60 of queue 38 for the out-of-rate priority level 64A is not null, then scheduler 34 advances to a step 332. If the head pointer is not null, a linked list, such as, list 68 (FIGURE 3) already exists for the out-of-rate connections. The connection provided by ABR manager 44 must then be circularly linked to the existing linked list.

At step 332, scheduler 34 reads the schedule mode field 92D of connection state 90 for the connection indicated by the tail pointer at priority level 64A in column 62

of global priority queue 38. Scheduler 34 determines if field 92D indicates that the connection is in the ABR mode. If scheduler 34 determines that the ABR mode has been set in schedule mode field 92D, a circularly linked list must be built utilizing the out-of-rate pointer in field 92R for the connection indicated by the tail pointer in column 62 of queue 38. The out-of-rate pointer in field 92Q is set to the connection provided by manager 44 in a step 334. After step 334, scheduler 34 advances to a step 338.

If in step 332 the schedule mode field 92D is not set to ABR mode, then the connection is an UBR mode of traffic and not an ABR mode of traffic, and scheduler 34 advances to a step 336. Therefore, the queue pointer in field 92F is utilized as opposed to the out-of-rate pointer in field 92R to build the linked list at step 336. In step 336, the queue pointer in field 92F for state 90 indicated by the tail pointer in queue 38 is set to the connection provided by manager 44.

After step 336, a scheduler 34 advances to step 338 where the out-of-rate pointer in field 92R of state 90 for the connection provided by ABR manager 44 is set to the head pointer in column 60 of queue 38. At a step 340, the tail pointer in column 62 is set to the connection provided by manager 44, and flow chart 325 is completed. In this way, a circular queue is built for both ABR linked lists which utilize the out-of-rate pointer in field 92R and for UBR linked lists which utilize the queue pointer in field 92F.

With reference to FIGURE 8A, scheduler 34 performs schedule connection flow chart 350 to add connections from start FIFO memory 40 to global priority queue 38. Typically, scheduler 34 schedules connections by placing the connections on the global priority queue 38 by inserting the connection number or connection information in the head pointer in column 60 and in the tail pointer in column 62.

At a step 352, scheduler 34 determines if connection state 90 of the connection provided by start FIFO memory 40 has a schedule mode field 92D set to a CBR mode of traffic. If so, the connection does not have to be scheduled because it is already scheduled on dynamic schedule table 88 in an entry 94 and will automatically be transferred to schedule FIFO memory 42 when the appropriate slot of slots 88A-G is processed. The nature of CBR mode of traffic is such that the connection must be sent at particular times defined by slots 88A-G to meet strict protocol requirements.

If schedule mode field 92D does not indicate that the connection is a CBR connection, scheduler 34 advances to a step 353 and determines whether connection state 90 of the connection indicated by start FIFO memory 40 has field 92D set to a GFR mode, a field 92S which does not match the currently selected priority, and a field 920 which

indicates MCR priority level. If connection state 90 does not indicate GFR mode field 92S, does the currently selected priority level, or does not indicate the MCR priority, then scheduler 34 advances to a step 354 and determines whether connection state 90 of the connection indicated by start FIFO memory 40 has field 92D set to an ABR mode and a field 920 which indicates MCR priority. If connection state 90 indicates ABR mode and the MCR priority, then scheduler 34 advances to step 356.

At step 356, the priority level, such as, levels 64A-J, for the connection is increased by utilizing the head pointer for a priority level one above the priority level indicated in connection 90 by the priority level parameter in field 92E. The higher priority level is utilized for the remaining steps in flow chart 350. After a step 356, scheduler 34 advances to a step 358. In step 354, if state 90 indicates no ABR mode or no MCR priority for the connection, scheduler 34 advances to step 358.

At step 358, scheduler 34 determines if the head pointer in column 60 of queue 38 for the particular priority level of levels 64A-J is null. If the head pointer for the particular priority level is null, then there are no connections currently scheduled for the priority level, and the head pointer is set to the connection at a step 360. If the head pointer is not null, the connection must be attached to an existing linked list for the priority level stored in priority queue 38 (beginning at a step 362).

In step 362, if scheduler 34 determines that schedule mode field 92D of connection state 90 of the connection from start FIFO memory 40 indicates a UBR mode or if schedule mode field 92D indicates GFR mode and the priority field 92S is set to the current priority level, and schedule field 92D of connection state 90 pointed to by the tail pointer in queue 38 indicates an ABR mode, scheduler 34 advances to a step 364. At step 364, scheduler 34 sets the out-of-rate pointer in field 92R for connection state 90 pointed to by the tail pointer in queue 38 to the connection provided by start FIFO memory 40.

At step 363, scheduler 34 determines whether schedule mode field 92D of connection state 90 of the connection from start FIFO memory 40 indicates either UBR mode or GFR mode and the priority level matching field 92E, and schedule field 92D of connection state 90 pointed to by the tail pointer in queue 38 indicates a GFR mode. If not, scheduler 34 advances to a step 366 described with reference to FIGURE 8A. If so, then scheduler 34 advances to a step 365. At step 365, scheduler sets a GFR queue pointer in field 92T of the connection state 90 pointed to by the tail pointer to the connection provided by start FIFO memory 40. After step 365, scheduler 34 advances to a step 369 described with reference to FIGURE 8D.

At step 366, scheduler 34 sets the queue pointer in field 92F of connection state 90 pointed to by the tail pointer to the connection provided by start FIFO memory 40.

After step 366, scheduler 34 advances to a step 368 and determines if the schedule mode field 92D of the connection provided by start FIFO memory 40 indicates an UBR mode. If so, a circular buffer must be built, and scheduler 34 advances to a step 370 where the queue pointer in field 92F of connection state 90 for the connection provided by FIFO memory 40 is set to the head pointer of global priority queue 38. If not, scheduler 34 advances to a step 369 described with reference to FIGURE 8D.

In step 369, scheduler 34 determines whether connection state 90 of the connection indicated by start FIFO memory 40 has field 92D set to a GFR mode and a field 92S is set to the current priority. If connection state 90 does not indicate GFR mode or the current priority matches field 92S, scheduler 34 advances to a step 376. If connection state 90 indicates GFR mode and the current priority matches field 92S, scheduler 34 advances to a step 371. At step 371, scheduler 34 sets the GFR queue pointer in field 92T of connection state 90 for the connection provided by FIFO memory 40 to the head pointer of global priority queue 38. After step 371, scheduler 34 advances to step 376 described with reference to FIGURE 8A.

After step 360, scheduler 34 advances to a step 372 and determines if schedule mode field 92D of connection state 90 for the connection provided by FIFO memory 40 indicates an UBR mode. If so, scheduler 34 advances to a step 374 and sets the queue pointer in field 92F of connection state 90 to the connection provided by FIFO memory 42 (e.g., to the head pointer). If not, scheduler 34 advances to a step 373, described with reference to FIGURE 8B, and determines whether connection state 90 of the connection indicated by start FIFO memory 40 has field 92D set to a GFR mode and a field 92S which matches the current priority. If connection state 90 does not indicate GFR mode or field 92S does not match the current priority, then scheduler 34 advances to a step 376. If connection state 90 indicates GFR mode and field 92S matches the current priority, then scheduler 31, advances to a step 375. At step 375, scheduler 34 sets the GFR queue pointer in field 92T of connection state 90 to the connection provided by FIFO memory 42 (e.g., to the head pointer). After step 375, scheduler 34 advances to a step 376 described with reference to FIGURE 8A. After step 374, scheduler 34 advances to step 376. At step 376, scheduler 34 sets the tail pointer in queue 38 to the connection provided by start FIFO memory 40. After step 370, scheduler 34 advances to step 376.

In step 368, if field 92D does not indicate an UBR mode, schedular 34 advances to step 369 described with reference to FIGURE 8D. As discussed above, in step 369, scheduler 34 determines whether connection state 90 of the connection indicated by start FIFO memory 40 has field 92D set to a GFR mode and field 92S which matches the current priority. If connection state 90 does not indicate GFR mode or the field 92S does not match the current priority, scheduler 34 advances to a step 376. If connection state 90 indicates GFR mode and the field 92S matches the current priority, schedular 34 advances to a step 371. At step 371, schedular 34 sets the GFR queue pointer in field 92T of connection state 90 for the connection provided by FIFO memory 40 to the head pointer of global priority queue 38. After step 371, schedular 34 advances to set 376 described with reference to FIGURE 8A.

After step 376, schedular 34 advances to a step 378 and determines whether schedular mode field 92D of connection state 90 indicates a GFR mode and field 92S matches the current priority. If connection state 90 does not indicate GFR mode or field 92S does not match the current priority, then flowchart 350 is completed. If connection state 90 indicates GFR mode and field 92S does not match the current priority, scheduler 34 advances to a step 380. At step 380, priority is set to the GFR priority in field 92S of connection state 90 and scheduler 34 advances to step 352 and repeats flowchart 350. Effectively, step 378 provides for two complete passes of flowchart 350.

Steps 354-380 allow scheduler 34 to determine if the connection in start FIFO memory 40 utilized an UBR, ABR, GFR, or VBR mode of traffic. If the connection utilized an ABR mode or a UBR mode, the connection is circularly linked in priority queue 38. If the connection utilized a VBR mode, a non-circular linked list is built if the connection utilized GFR mode, it is circularly linked to the priority level indicated by field 92S and non-circularly linked to priority field indicated by field 92E.

With reference to FIGURE 9, process slot flowchart 400 is utilized so scheduler 34 can process a slot of slots 88A-G in dynamic schedule table 88. In this example, slot 88C is processed for illustration only. At a step 402, the schedule difference parameter is decremented to accurately represent the temporal relationship between the real time and the position of scheduler 34 with respect to slots 88A-G. At a step 404, scheduler 34 increments the schedule table position to the next position of slots 88A-G to accurately reflect the position on table 88. In this example, the scheduled table position is incremented to point to slot 88C.

At a step 406, scheduler 34 determines if CBR entry 94 of slot 88C includes connection information. If so, scheduler 34 advances to a step 408. At step 408, scheduler 34 determines if connection state 90 indicated by the connection entry 94 of slot 88C has data associated with connection is available.

If the data is available for the connection, scheduler 34 writes the connection in entry 94 to schedule FIFO memory 42 in a step 410. Scheduler 34 writes the CBR connection directly to scheduler 42 as opposed to global priority queue 38 because of the conformance requirements for the CBR mode of traffic. In this way, scheduler 34 guarantees that CBR connection will be transmitted appropriately.

In step 408, if data available field 92A indicates that the data for the connection is not available, scheduler 34 advances to a step 414. In step 406, if entry 94 does not include connection information, scheduler 34 advances to step 414.

At step 414, scheduler 34 determines if tunnel entry 95 slot 88C indicates that a CBR tunnel transmission is scheduled. A tunnel transmission typically consists of several CBR connection distributed in a round-robin order. The circularly linked list of the CBR connections associated with tunnel entry 95 is provided in global priority queue 38.

At a step 416, scheduler 34 sets the tunnel active by providing an appropriate header for priority level 64I associated with tunnel transmission in priority queue 38. After step 416, scheduler 34 advances to a step 418. In step 414, if tunnel entry 95 does not indicate that there is a tunnel connection for slot 88C, scheduler 34 advances to a step 418.

At step 418, scheduler 34 sets the queue number for slot tail pointer entries 96A-D to one indicating entry 96A. After step 418, scheduler 34 advances to a flowchart 425 described with reference to FIGURE 10 and processes the slot tail pointer provided in entry 96A. At a step 422, scheduler 34 determines if all of tail entries 96A-D have been processed. If not, scheduler 34 advances to a step 424 where the queue number is incremented. After step 424, scheduler 34 returns to flowchart 425. To complete processing slot 88C, scheduler 34 would perform flowchart 425, step 422, and step 424. If in step 422 all of the entries 96A-D have been processed, scheduler 34 has completed flowchart 400.

With reference to FIGURE 10, scheduler 34 performs the process slot tail queue flowchart 425 to process the slot tail pointer indicated by entries 94A-D in slot 88C (for example only). The queue number associated with entries 96A-D have a relation ship

to the priority levels 64A-J in priority queue 38. Preferably, there is a fixed offset between the VBR priority levels of entries 96A-D and priority levels 64A-J of queue 38.

In this example, entry 96A is processed. At step 428, scheduler 34 determines if the tail pointer in entry 96A is null. If so, there are no VBR connections in entry 96A and flowchart 425 is completed. If a slot tail pointer is provided in entry 96A, scheduler 34 advances to a step 430.

At step 430, scheduler 34 determines if the head pointer in column 60 of queue 38 for the priority level indicated by slot tail pointer entry 96A is null. If so, scheduler 34 advances to a step 432. At step 432, scheduler 34 sets the head pointer in priority queue 38 to the queue pointer in field 92F of connection state 90 pointed to by the slot tail pointer in entry 96A. After a step 432, advances to a step 434.

If the head pointer is not null in step 430, scheduler 34 advances to a step 436 and adds the connection to an already existing linked list defined in queue 38. At step 436, scheduler 34 sets the queue pointer in field 93F of connection state 90 indicated by the tail pointer in column 62 of global priority queue 38 to the queue pointer stored in field 92F of connection state 90 indicated by the slot tail pointer in entry 96A. For example, connection slate 98B at the head of list 98 is added to the linked list defined in queue 38. In this way, list 98 (FIGURE 5) is added to the list defined in queue 38. After a step 436, scheduler 34 advances to step 434.

At step 434, the tail pointer in column 62 of priority of queue 38 is set to the slot tail pointer in entry 92A. In this example, the slot tail pointer points to state 98A, and all of list 98 is added to the list in queue 38, with only slot tail pointer and a queue pointer being updated. After step 434, scheduler 34 advances to a step 436 and sets the slot tail pointer in entry 96A to null, thereby clearing entry 96A. After step 436, flowchart 425 is completed.

With reference to FIGURES 11A-C, scheduler 34 performs send cell flowchart 500 to sent the highest priority pending connection from global priority queue 38.

At a step 502, scheduler 34 selects the connection stored in the highest pending global priority level of levels 64A-J of global priority queue 38. After step 502, scheduler 34 sets the connection number equal to the connection associated with the head pointer stored in column 60 of queue 38 for the highest pending global priority level of levels 64A-J at a step 504.

After step 504, scheduler 34 determines if data available field 92A for connection state 90 for the connection indicates that data is available at a step 506. If so, scheduler 34 advances to a step 508 (FIGURE 11B), and begins to remove the connection from global priority queue 38. The connection must be removed from the global priority queue to ensure that it is not continually selected.

At step 508, described with reference to FIGURE 8B, scheduler 34 determines if the head pointer is equal to the tail pointer in priority queue 38. If so, scheduler 34 advances to a step 510 and sets the head pointer in priority queue 38 to null. When the head pointer and the tail pointer are equal, scheduler 34 determines that there is only one connection associated with linked list defined by queue 38. Therefore by setting the head pointer in column 60 to null, the linked list is effectively eliminated from queue 38. The tail pointer in column 62 does not have to be set to null because the head pointer in column 60 is always checked first. After step 510, flowchart 500 is completed.

At step 508, if the head pointer is not equal to the tail pointer, scheduler 34 advances to a step 512. At step 512, scheduler 34 determines if connection state 90 for the connection indicated by the head pointer of queue 38 has a schedule mode field 92D which indicates an UBR mode. If so, scheduler 34 must eliminate the connection from the circular linked list associated with the priority level in queue 38 and advances to a step 513. If not, scheduler 34 advances to a step 515 (FIGURE 11B).

At step 513, scheduler 34 sets a next parameter to the queue pointer in field 92F. The next parameter indicates the next connection to which the linked list associated with queue 38 should be set.

At step 515, scheduler 34 determines if field 92D indicates an ABR mode and if the selected priority if equal to the out-of-rate priority parameter in field 92Q. If so, scheduler 34 sets a next parameter to the out-of-rate pointer in field 92R at a step 517 (e.g., chooses the out-of-rate connection). If not, scheduler 34 advances to a step 516 described with reference to FIGURE 11D. At step 516, scheduler 34 determines whether schedule mode field 92D of connection state 90 indicates a GFR mode and field 92S matches the current priority. If connection state 90 does not indicate GFR mode or field 92S does not match the current priority, scheduler 34 advances to a step 519. If connection state 90 indicates GFR mode and field 92S matches the current priority, scheduler 34 advances to a step 521 (FIGURE 11D). At step 521, scheduler 34 sets a next parameter to the GFR queue pointer in field 92T and advances to a step 514. At step 519, scheduler 34

sets a next parameter to the queue pointer in field 92F. After step 517, scheduler 34 advances to step 514. After step 519, scheduler 34 advances to a step 516.

At step 514, scheduler 34 determines if field 92D of connection state 90 pointed by the tail pointer in queue 38 indicates an ABR mode. If so, scheduler 34 advances to a step 518 and sets the out-of-rate pointer field 92R in state 90 for then connection indicated by the tail pointer of queue 38 to the next parameter. After step 518, scheduler 34 advances to step 516.

At step 514, if field 92D does not indicate an ABR mode, scheduler 34 advances to a step 522 described with reference to FIGURE 11E. At step 522, scheduler 34 determines whether field 92D of connection state 90 pointed to by the tail pointer in queue 38 indicates a GFR mode. If not, scheduler 34 advances to a step 520 (FIGURE 11B). If so, scheduler 34 advances to step 524 and sets the GFR queue pointer field 92T in state 90 for the connection indicated by the tail pointer of queue 38 to the next parameter. After step 524, scheduler 34 advances to step 516.

At step 520, scheduler 34 sets the queue pointer in field 92F in connection state 90 pointed to by the tail pointer of queue 38 to the next parameter. After step 520, scheduler 34 advances to step 516 (FIGURE 11D).

At step 516, scheduler 34 sets the head pointer in column 60 of queue 38 to the next parameter. Therefore, scheduler 34 effectively sets the linked list to the next connection indicated by the next parameter. After step 516, flowchart 500 is completed.

At step 506 (FIGURE 11A), if data available field 92A for the connection indicates that data is available, scheduler 34 advances to a step 530. At step 530, scheduler 34 determines if field 92D indicates an ABR mode. If so, scheduler 34 advances to a step 532. At step 532, scheduler 34 receives the cell type from ABR manager 44 on control bus 58 and writes the connection information (e.g., the connection number and the type of connection) to schedule FIFO memory 42. After step 532, scheduler 34 advances to a step 534.

At step 534, scheduler 34 determines if the selected priority level in priority queue 38 is equal to the out-of-rate priority parameter in field 92Q. If so, scheduler 34 advances to a step 558 (FIGURE 11C). If not, scheduler 34 advances to a step 578 (FIGURE 11C).

At step 530, if field 92D indicates that the connection is not an ABR mode, scheduler 34 advances to a step 536 where scheduler 34 writes the connection to schedule FIFO memory 42. At a step 538, scheduler 34 determines whether field 92D indicates

UBR mode. If so, scheduler 34 advances to step 558 (FIGURE 11D). If not, scheduler 34 advances to a step 540. At step 540, scheduler 34 determines whether schedule mode field 92D of connection state 90 indicates a GFR mode and a field 92S matches the current priority. If connection state 90 does not indicate GFR mode or field 92S does not match the current priority, scheduler 34 advances to a step 578 (FIGURE 11C). If connection state 90 indicates GFR mode and field 92S matches the current priority, scheduler 34 advances to step 558 (FIGURE 11C).

If scheduler 34 reaches step 558, the connection utilizing the GFR out-of-rate ABR, or UBR mode of traffic-has been written to schedule FIFO memory 42. At step 558, scheduler 34 determines whether selected priority level of levels 64A-D is indicative of an interval tunnel, represented by field 64J of queue 38 (FIGURE 2). If so, scheduler 34 clears interval tunnel active indication at a step 559. If not, scheduler 34 advances to a step 560. At step 560, scheduler 34 determines if the selected priority level of levels 64A-H is indicative of a tunnel mode represented by field 64I. If so, scheduler 34 clears the tunnel active indication at a step 562 and advances to a step 564. If not, scheduler 34 advances to step 564.

At step 564, scheduler 34 determines if the head pointer in column 60 of queue 38 for the selected priority level is equal to the tail pointer in column 62. If so, flow chart 500 is completed because the circularly linked list consists of only the one connection. If not, scheduler 34 advances to a step 566.

At step 566, scheduler 34 sets the tail pointer in column 62 of queue 38 to the connection (e.g., the head pointer in queue 38) and advances to a step 567. In step 567, scheduler 34 determines if field 92D indicates an ABR mode. If so, scheduler 34 sets the head pointer in queue 38 to the out of rate pointer in field 92R of connection state 90 for the connection at a step 570. If not, scheduler 34 advances to a step 572 and determines whether schedule mode field 92D of connection state 90 indicates a GFR mode. If not, then scheduler 34 advances to a step 568. If so, then scheduler 34 advances to step 574 and sets the head pointer in queue 38 to the GFR queue pointer in field 92T of connection state 90 for the connection. After step 574, flowchart 500 is completed.

At step 568, scheduler 34 sets the head pointer in queue 38 to the queue pointer in field 92F of connection state 90 for the connection. Steps 566, 568, and 570 rotate the circularly linked list associated with GFR, ABR and UBR modes so that the next connection is at the head of the list. After steps 568 and 570, flow chart 500 is completed.

When scheduler 34 reaches step 578, scheduler 34 has determined that a GFR, ABR, or VBR connection has been written to schedule FIFO memory 42. At step 578, scheduler 34 determines whether selected priority level of levels 64A-H is indicative of an interval tunnel connection, represented by field 64J of queue 38 (FIGURE 2). If so, scheduler 34 clears interval tunnel active indication at a step 579. If not, scheduler 34 advances to a step 580. At step 580, scheduler 34 determines if the head pointer in queue 38 is equal to the tail pointer. If so, scheduler 34 sets the head pointer in column 60 of queue 38 to null in a step 582. If not, scheduler 34 advances to a step 584 and sets the head pointer to the queue pointer field 92F of connection state 90 for the connection.

After steps 582 and 584, scheduler 34 advances to schedule rate flow chart 600 described with reference to FIGURES 12A-C. When flow chart 600 is completed, flow chart 500 is completed.

Scheduler 34 utilizes schedule rate flow chart 600 to schedule a connection in the future by linking the connection onto a schedule slot of slots 88A-G in dynamic schedule table 88. In this way, scheduler 34 dynamically allocates connections using ABR and VBR modes to maximize the bandwidth of communication system 20. Scheduler 34 utilizes the single bucket and dual bucket scheduling algorithms to appropriately schedule ASR and VBR modes of traffic in slots 88A-G of table 88. Statistical multiplexing can be achieved for VBR and ABR modes of traffic by bursting connections to the extent allowed by L1 and L2 parameters to recover from periods during which the connections were not allowed to be transmitted.

With reference to FIGURES 12A-C, scheduler 34 sets a now parameter equal to the current time provided by timer 36 at a step 602. At a step 604, scheduler 34 determines if VBR reset field 92G of connection state 90 indicates that the delta 1 and delta 2 parameters for the connection should be reset. Resetting the delta 1 and delta 2 parameters sets VBR buckets to their maximum burst state. If so, scheduler 34 advances to a step 606. If not, scheduler 34 advances to a step 612.

At step 606, the elapsed parameter is set to a maximum value. After step 606, scheduler 34 advances to a step 608 and sets the delta 1 parameter in field 92M and the delta 2 parameter in field 92N to zero. After step 608, scheduler 34 advances to a step 610.

At step 612, scheduler 34 sets the elapsed parameter to the now parameter minus the present parameter in field 92L. The elapsed parameter represents the amount of time since the last VBR transmission for the connection. At step 610, scheduler 34

determines if field 92D of connection state 90 indicates a ARC mode and if CLP field 92B is set. If so, scheduler 34 advances to a step 650 (FIGURE 12B) and schedules the connection in accordance with a VBRC bucket algorithm. If not, scheduler 34 advances to a step 614.

At step 614, a late parameter is set equal to the elapsed parameter minus a delta 1 parameter stored in delta 1 field 92M of connection state 90 for the connection. After step 614, scheduler 34 advances to step 616 and determines if field 92P for the connection is set to an ABR mode or a GFR mode. If not, scheduler 34 advances to a step 618. If so, scheduler 34 advances to a step 624.

At step 618, scheduler 34 determines if the late parameter is greater than the L1 parameter in L1 field 921 of connection state 90 for the connection. If so, scheduler 34 advances to a step 620 and sets the late parameter equal to the L1 parameter in field 92I. If not, scheduler 34 advances to a step 622. After step 620, scheduler 34 advances to step 622.

At step 624, if the late parameter plus the I1 parameter in field 92H for the connection is greater than the MCR limit parameter in the MCR field 92P, scheduler 34 advances to a step 626. Otherwise, scheduler 34 advances to step 618. At step 624, scheduler 34 sets the MCR priority parameter in field 920. Thus, preferred scheduling can be achieved via the MCR priority parameter and the MCR limit parameter.

At step 622, scheduler 34 sets the delta 1 parameter in field 92M equal to the I1 parameter in field 92H minus the late parameter. After step 622, scheduler 34 advances to step 650 (FIGURE 12B). At step 650, scheduler 34 determines if schedule mode field 62C indicates a VBR 1, an ABR mode, or a GFR mode. If so, scheduler 34 advances to a step 652 and sets a delta parameter equal to the delta 1 parameter in field 92M. If not, scheduler 34 advances to a step 654 where the late parameter is set equal to the elapsed parameter minus the delta 2 parameter in field 92N.

After step 654, scheduler 34 advances to a step 656 where the late parameter is compared to the L2 parameter stored in field 92K for the connection. If the late parameter is greater than the L2 parameter, scheduler 34 advances to a step 658 where the late parameter is set equal to the L2 parameter in field 92K. After steps 656 and 658, scheduler 34 advances to a step 660.

At step 660, scheduler 34 sets the delta 2 parameter in delta 2 field 92N equal to the I2 parameter in I2 field 92J minus the late parameter. After step 660, scheduler 34 determines if field 92D indicates a VBRC mode and if the last CLP field 92B

is set at a step 662. If so, scheduler 34 advances to a step 664 where the delta parameter is set to the delta 2 parameter in field 92N for the connection. If not, scheduler 34 advances to a step 666.

At step 666, scheduler 34 determines if the delta 1 parameter in field 92M is less than the delta 2 parameter in field 92N. If so, scheduler 34 advances to step 664. If not, scheduler 34 advances to step 652.

After steps 652 and 664, scheduler 34 advances to a step 668 where the present parameter in field 92L for the connection is set equal to the now parameter.

After step 668, scheduler 34 determines if the delta parameter is less than one at a step 669 (FIGURE 12C). If the delta parameter is less than one, scheduler 34 advances to a step 670 and sets the delta parameter equal to one. After step 670, scheduler 34 advances to a step 672.

If the delta parameter is not less than one at step 669, scheduler 34 advances to step 672. At step 672, scheduler 34 calculates a new schedule position parameter equal to a current position plus the ceiling (delta parameter). The function ceiling (delta parameter) is the lowest integer which is greater than or equal to the delta parameter. The new schedule position refers to a new slot of slots 88A-G.

After step 672, scheduler 34 determines if the new schedule position parameter is greater than or equal to the schedule table size parameter at a step 674. The schedule table size parameter is a user-set parameter which indicates the number of slots 88A-G for dynamic schedule table 88. If so, the new schedule position parameter will not fit within one cycle of the dynamic schedule table 88, and scheduler 34 subtracts the schedule table size parameter from the new schedule position parameter at a step 676. After step 676, scheduler 34 advances to a step 678.

If the new schedule position parameter is less than the schedule table size parameter, scheduler 34 advances from step 674 to step 678. At step 678, scheduler 34 selects a slot tail pointer entry of entries 96A-D in dynamic schedule table 88 for one of slot 88A-G. The priority level of the chosen entry of entries 96A-D is equal to the priority parameter in field 92E minus the VBR priority offset minus the MCR priority parameter in field 920. Therefore, the priority can be increased in the new schedule slot so that the connection is more likely to be transmitted after the new schedule position has been processed.

After step 678, scheduler 34 determines if the slot tail pointer entry of entries 96A-D is null at a step 680. If so, scheduler 34 sets the queue pointer in field 92F equal to the connection at a step 682. After step 682, scheduler 34 advances to a step 684.

If the slot tail pointer in the selected entry is not null, scheduler 34 advances from step 680 to a step 686 and sets the queue pointer in field 92F equal to the queue pointer in field 92F of connection state 90 pointed to by the slot tail pointer in the selected entry of entries 96A-D. After step 686, scheduler 34 sets the queue pointer in field 92F of connection state 90 for the connection pointed to by the slot tail pointer of the selected entry to the connection. After step 688, scheduler 34 advances to a step 684. At step 684, scheduler 34 sets the slot tail pointer for the selected entry to the connection. After step 684, flow chart 600 is completed.

Alternatively, scheduler 34 can be modified to handle packets of data which are not of fixed size. In flow chart 600, the calculation of the delta parameter can be according to the following equation to handle variable size data packets: delta parameter = (I parameter! next packet size) - Late. This equation is the same as the equation in steps 622 and 660 where the next packet size parameter is equal to one in the fixed packet size case.

With reference to FIGURE 13, scheduler 34 performs a tunnel activation mechanism flow chart 700 to schedule connections for transmission on a communication network having a plurality of tunnels. At a step 702, scheduler 34 determines whether a slot interval timer is incremented. If it is, scheduler 34 advances to a step 704; otherwise, scheduler 34 remains at step 702. At step 704, scheduler 34 increments an interval tunnel counter and advances to a step 706. At step 706, scheduler 34 compares the interval tunnel counter to a tunnel interval trigger value. If the values are equal, scheduler 34 advances to a step 708 where the interval tunnel counter is set to zero and a step 710 where the interval tunnel (e.g., level 64J or a level 64J associated with the counter) is set active. After step 710, scheduler 34 advances to a step 712. If the values are not equal in step 706, scheduler advances to step 712. At step 712, scheduler 34 determines whether all interval tunnels are processed. If not, scheduler 34 increments an interval tunnel value 'n' in a step 714 and returns to step 704. If so, scheduler 34 advances to a step 716, sets the interval tunnel value 'n' to zero, and returns to step 702. As such, scheduler 34 follows flow chart 700 continually, throughout its operation and in parallel with operation 300 Figure 6. Thus, the tunnel mechanism embodiment advantageously allows scheduler 34 to utilize more than one tunnel. Also scheduler 34 does not need to reserve slots 88A-G for tunnel connections,

thereby reducing hardware and timing overhead. Such a scheme also allows total tunnel bandwidth to exceed line rate bandwidth as tunnel connections are scheduled according to different priority levels.

It is understood that, while the detailed drawings, specific examples, and particular memory structures given describe a preferred exemplary embodiment of the present invention, they are for the purpose of illustration only. The apparatus and method of the invention is not limited to the precise details and conditions disclosed. For example, although the scheduler is an ATM scheduler, it may be utilized with other communication systems. Also, although particular timing parameters are described, other methods of tracking scheduling times may be utilized. Further still, single lines in the various drawings may represent multiple conductors. Various changes may be made to the details disclosed without departing from the spirit of the invention which is defined by the following claims:

CLAIMS

What is claimed is:

1. A traffic manager coupled to a communication system for scheduling transmission of information associated with a plurality of connections in the communication system, the traffic manager comprising:

a priority queue;

a connection data structure including a guaranteed frame rate field for each connection;

a scheduler coupled to the connection data structure and the priority queue, the scheduler loading information from the data structure to the queue, wherein the scheduler shapes the connection for a minimum cell rate and allows use by the connection of excess bandwidth if the guaranteed frame rate field is set.

2. The traffic manager of Claim 1, wherein the shaping is performed by using a VBR service.

3. The traffic manager of Claim 1, wherein the excess bandwidth is utilized with a UBR service.

4. The traffic manager of Claim 1, wherein the connection data structure and the scheduler are located in the same unit.

5. The traffic manager of Claim 1 wherein the communication system is an asynchronous transfer mode (ATM) system.

6. A communication system comprising:

a control memory including a priority queue and a connection data structure, the priority queue having at least one tunnel entry, the connection data structure including a guaranteed bit rate field for each connection and, the connection using a particular bit rate mode of traffic, the connection data structure including particular bit rate mode parameters for the connection;

a scheduler coupled to the connection data structure, the scheduler scheduling a plurality of connections from the at least one tunnel

entry to select a connection for eventual transmission, the scheduler dynamically rescheduling the connection in response to a timing parameter, the particular bit rate mode parameters for the connection, and the guaranteed bit rate field.

7. The communication system of Claim 6, wherein the at least one tunnel entry includes a XBR tunnel for transmission of a variety of traffic modes, including ABR and VBR.

8. The communication system of Claim 6, wherein the at least one tunnel entry includes a UBR tunnel for transmission of traffic modes such as UBR which are associated with circularly linked lists.

9. The communication system of Claim 6, wherein the particular bit rate mode of traffic is variable bit rate (VBR).

10. The communication system of Claim 6, wherein the particular bit rate mode of traffic is unspecified bit rate (UBR).

11. The communication system of Claim 6, wherein the scheduler further shapes the connection for a minimum cell rate and allows use by the connection of excess bandwidth if the guaranteed bit rate field is set.

12. The communication system of Claim 11, wherein the shaping is performed by using a VBR service.

13. The communication system of Claim 11, wherein the excess bandwidth is utilized with a UBR service.

14. The communication system of Claim 9, wherein the scheduler dynamically reschedules in accordance with a single leaky bucket algorithm.

15. The communication system of Claim 9, wherein the scheduler dynamically reschedules in accordance with a dual leaky bucket algorithm.

16. The communication system of Claim 15, wherein the variable bit rate mode parameters include delta 1, delta 2, L1, L2, I1, and I2 parameters.

17. A traffic manager coupled to a communication system for scheduling transmission of information associated with a plurality of connections in the communication system, the traffic manager comprising:

a schedule table having a plurality of slots;

a GFR field:

a scheduler coupled to the schedule table, the scheduler processing the slots, the scheduler servicing a connection and scheduling the connection in accordance with GFR parameters when the GFR field is set.

18. The traffic manager of Claim 17, wherein the schedule table and a priority queue are located in the same memory unit.

19. The traffic manager of Claim 18, wherein the scheduler shapes the connection for a minimum cell rate and allows use by the connection of excess bandwidth if the GFR field is set.

20. The traffic manager of Claim 19, wherein the shaping is performed by using a VBR service and wherein the excess bandwidth is utilized with a UBR service.

21. A method of scheduling connections for transmission on a communication network having a first tunnel and a second tunnel, the method comprising:

incrementing a first tunnel counter associated with the first tunnel when a connection is processed;

setting the first tunnel active when the first tunnel counter is above a first threshold;

incrementing a second tunnel counter associated with the second tunnel when a connection is processed;

setting the second tunnel active when the second tunnel counter is above a second threshold; and determining whether the first and second tunnels are processed.

22. The method of Claim 21, wherein the first tunnel communicates any of a plurality of service classes.

23. The method of Claim 22, wherein the plurality of service classes include single and dual variable bit rate (VBR) modes.

24. The method of Claim 22, wherein the plurality of service classes includes unspecified bit rate (UBR) mode.

25. A scheduler, comprising:

a priority queue including at least one tunnel level;

a tunnel activation mechanism, the tunnel activation mechanism activating a tunnel level within the priority queue m response to a number of processing operations being completed.

26. A traffic manager coupled to a communication system for scheduling transmission of information associated with a plurality of connections in the communication system, the traffic manager comprising:

a schedule table including a plurality of slots, each of the slots having at least one tunnel entry;

a global priority queue having a tunnel level associated with the tunnel entry, the tunnel level defining at least one connection from one of a plurality of service classes; and

a scheduler couples to the schedule table and the global priority queue, the scheduleer loading information from the schedule table to the global priority queue, wherein the scheduler shapes the connection for a minimum cell rate and allows use by the connection of excess bandwidth if the guaranteed bit rate field is set, the scheduler further selecting at least one connection from the tunnel level of the global priority queue when the tunnel level is active, whereby the selection of connections from the plurality of service classes allows per connection shaping in the tunnel mode to maximize communication bandwidth.

27. The traffic manager of Claim 26, wherein the schedule table and priority queue are located in the same unit.

28. The traffic manager of Claim 26, wherein the service classes include a VBR mode or an ABR mode.

29. The traffic manager of Claim 26, wherein the service classes include UBR mode.

30. The traffic manager of Claim 26, wherein the communication system is an asynchronous transfer mode (ATM) system.

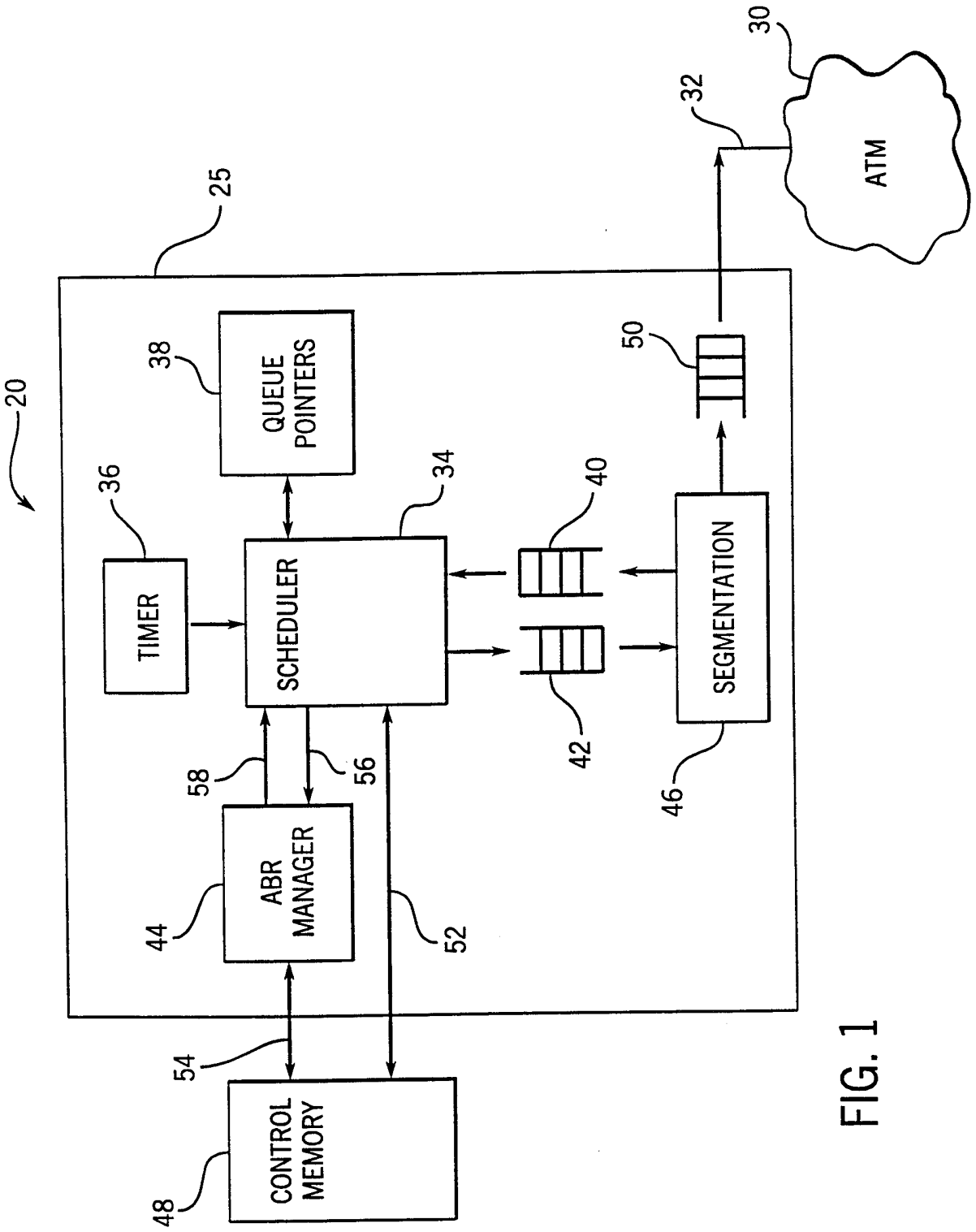

20
25
30
32
ATM
38
QUEUE POINTERS
36
TIMER
34
SCHEDULER
40
42
50
SEGMENTATION
46
58
56
44
ABR MANAGER
52
54
48
CONTROL MEMORY

FIG. 1

38

PRIORITY	60 HEAD POINTER	62 TAIL POINTER	
1 UBR1	HEAD POINTER 1	TAIL POINTER 1	64A
2 VBR1	HEAD POINTER 2	TAIL POINTER 2	64B
3 VBR2	HEAD POINTER 3	TAIL POINTER 3	64C
4 VBR3	HEAD POINTER 4	TAIL POINTER 4	64D
5 GFR-VBR	HEAD POINTER 5	TAIL POINTER 5	64E
6 ABR	HEAD POINTER 6	TAIL POINTER 6	64F
7 GFR-UBR	HEAD POINTER 7	TAIL POINTER 7	64G
8 UBR2	HEAD POINTER 8	TAIL POINTER 8	64H
9 TUNNEL	HEAD POINTER 9	TAIL POINTER 9	64I
10 INTERVAL TUNNEL	HEAD POINTER 10	TAIL POINTER 10	64J

FIG. 2

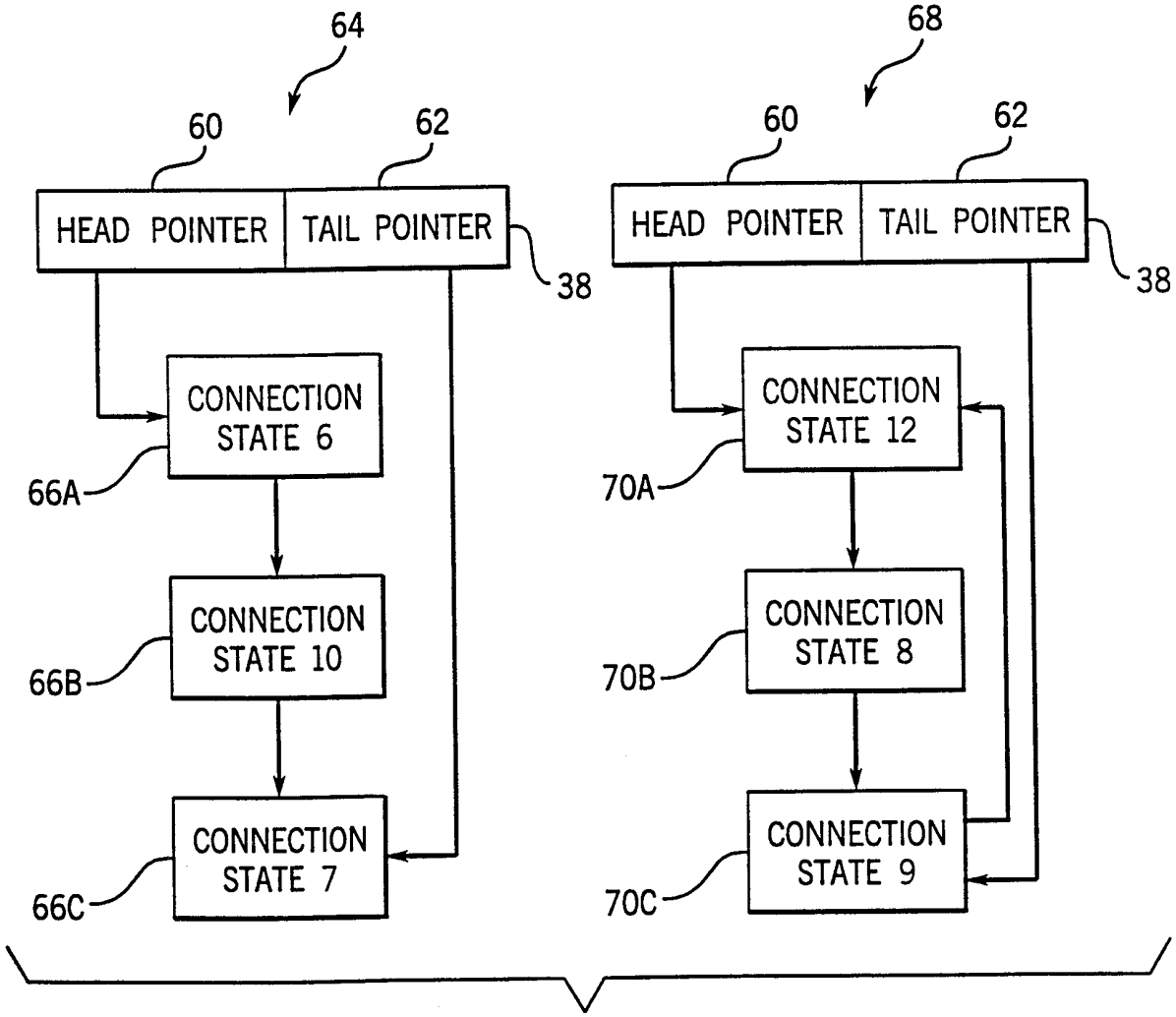


FIG. 3

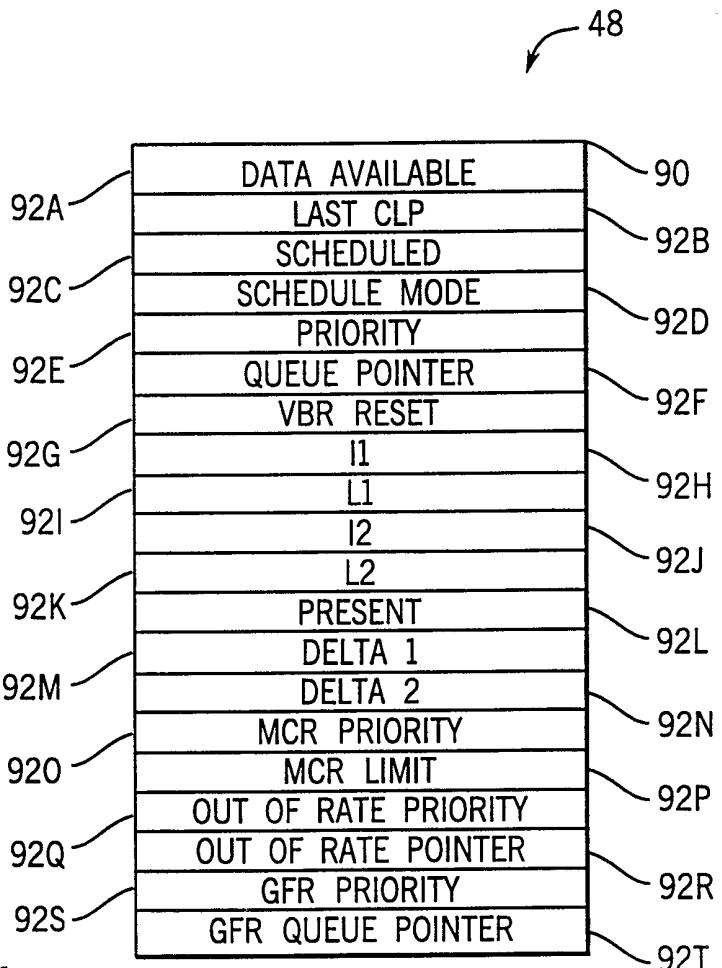

48
DATA AVAILABLE
LAST CLP
SCHEDULED
SCHEDULE MODE
PRIORITY
QUEUE POINTER
VBR RESET
I1
L1
I2
L2
PRESENT
DELTA 1
DELTA 2
MCR PRIORITY
MCR LIMIT
OUT OF RATE PRIORITY
OUT OF RATE POINTER
GFR PRIORITY
GFR QUEUE POINTER
90
92A
92B
92C
92D
92E
92F
92G
92H
92I
92J
92K
92L
92M
92N
92O
92P
92Q
92R
92S
92T

FIG. 4

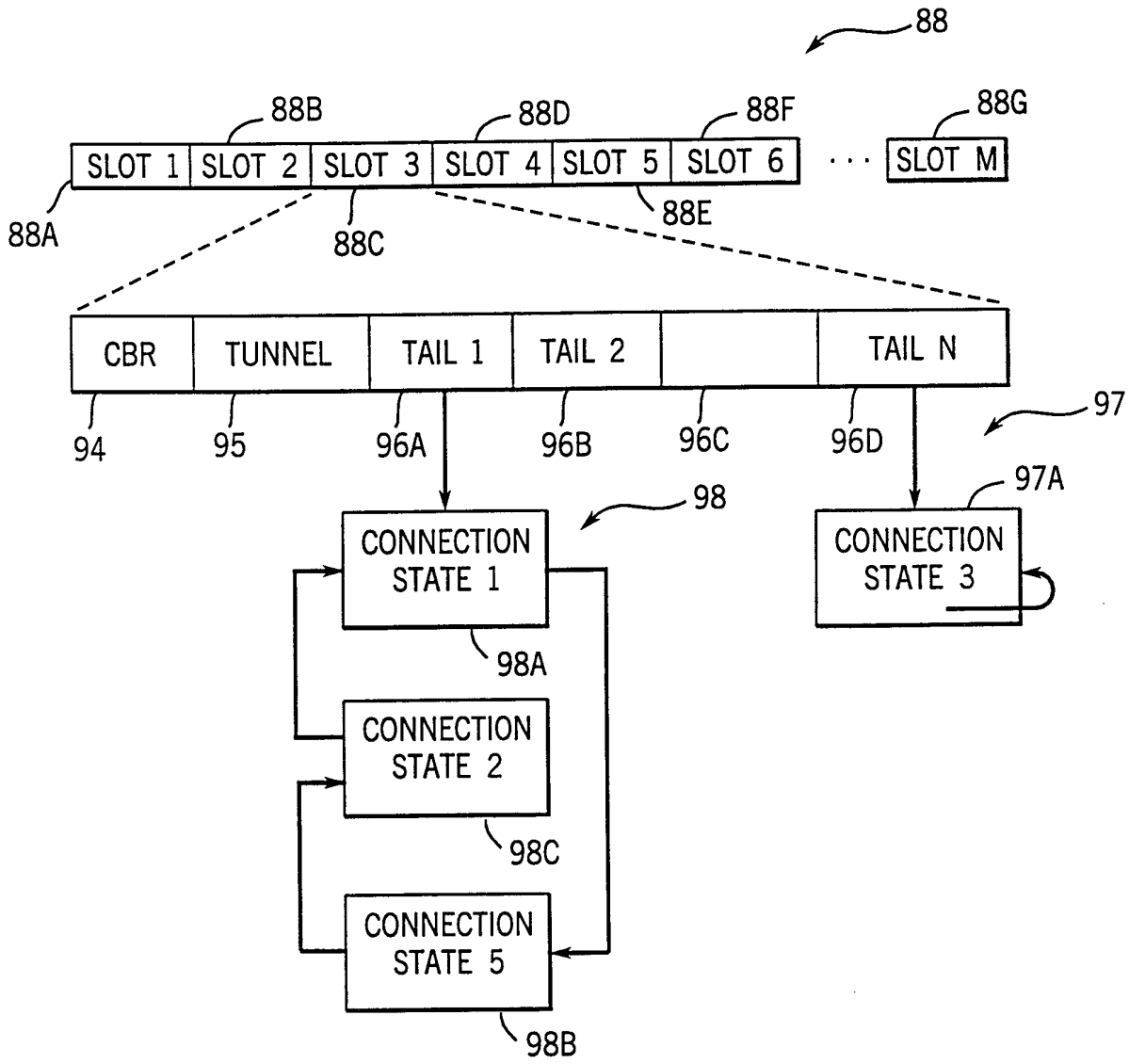

88
88B
88D
88F
88G
SLOT 1
SLOT 2
SLOT 3
SLOT 4
SLOT 5
SLOT 6
SLOT M
88A
88C
88E
CBR
TUNNEL
TAIL 1
TAIL 2
TAIL N
94
95
96A
96B
96C
96D
97
97A
98
CONNECTION STATE 1
CONNECTION STATE 3
98A
CONNECTION STATE 2
98C
CONNECTION STATE 5
98B

FIG 5

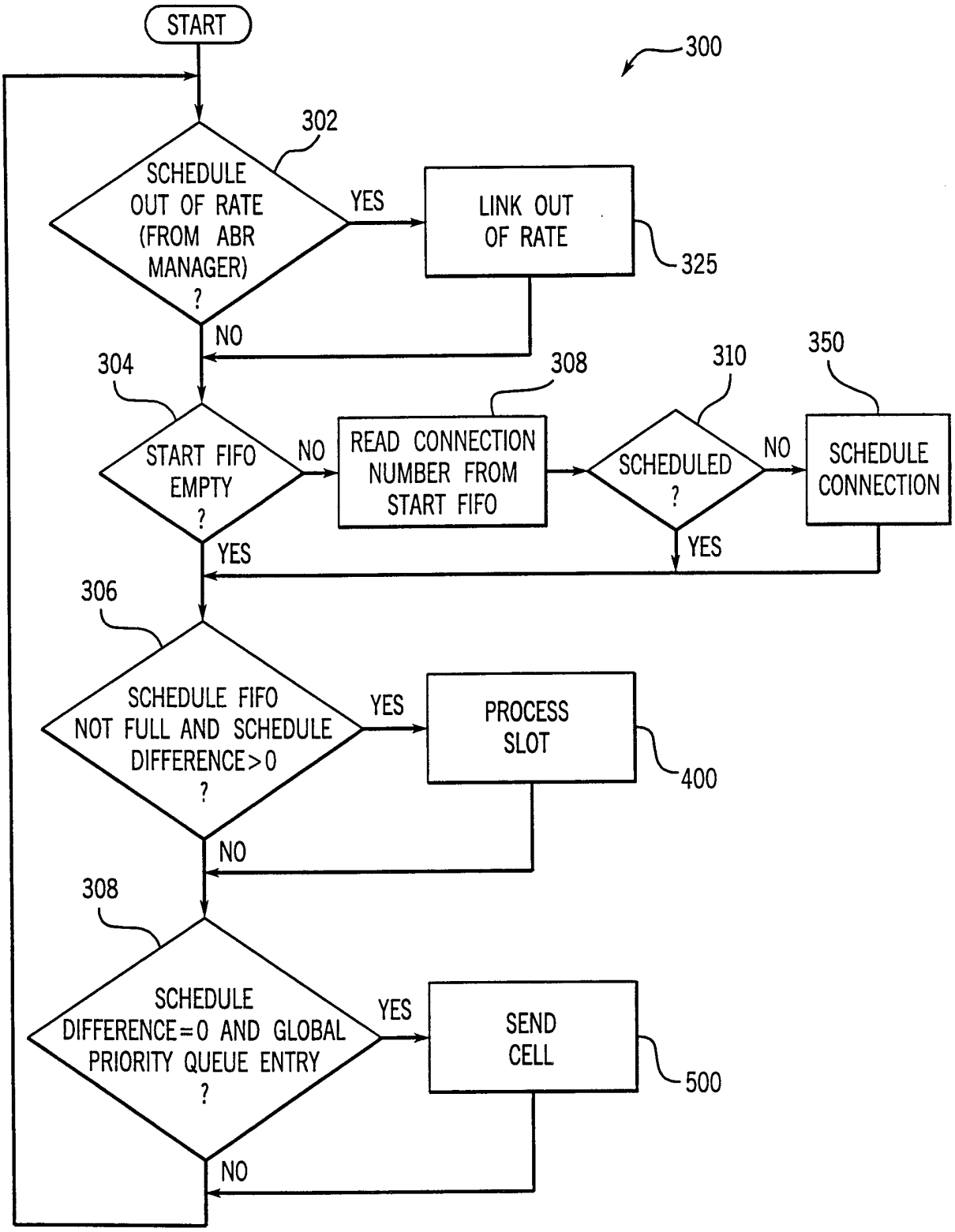

START
300
302
SCHEDULE OUT OF RATE (FROM ABR MANAGER) ?
YES
LINK OUT OF RATE
325
NO
304
START FIFO EMPTY ?
NO
308
READ CONNECTION NUMBER FROM START FIFO
310
SCHEDULED ?
NO
350
SCHEDULE CONNECTION
YES
YES
306
SCHEDULE FIFO NOT FULL AND SCHEDULE DIFFERENCE>0 ?
YES
PROCESS SLOT
400
NO
308
SCHEDULE DIFFERENCE=0 AND GLOBAL PRIORITY QUEUE ENTRY ?
YES
SEND CELL
500
NO

FIG. 6

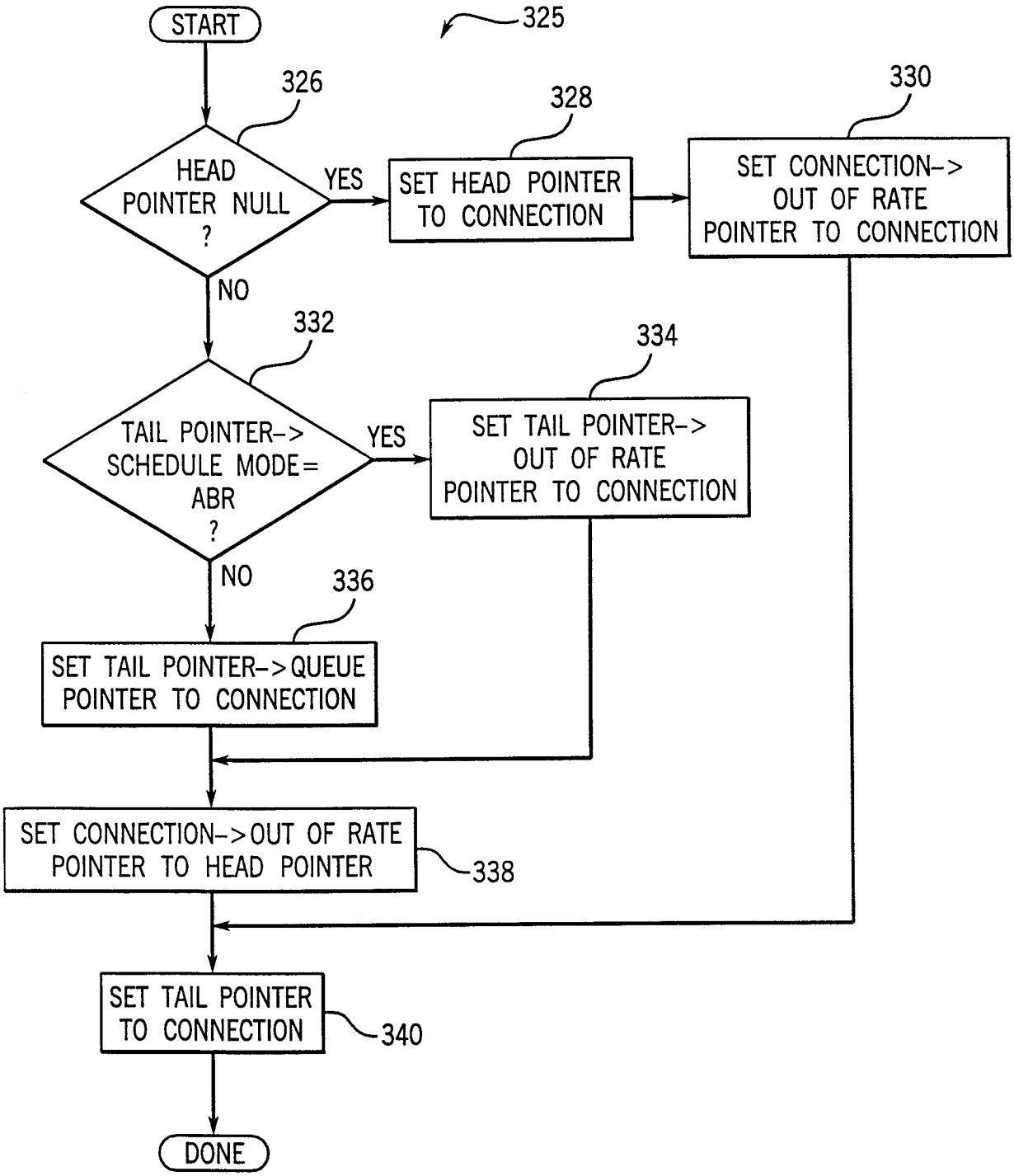

START
325
326
HEAD POINTER NULL ?
YES
328
SET HEAD POINTER TO CONNECTION
330
SET CONNECTION-> OUT OF RATE POINTER TO CONNECTION
NO
332
TAIL POINTER-> SCHEDULE MODE= ABR ?
YES
334
SET TAIL POINTER-> OUT OF RATE POINTER TO CONNECTION
NO
336
SET TAIL POINTER->QUEUE POINTER TO CONNECTION
SET CONNECTION->OUT OF RATE POINTER TO HEAD POINTER
338
SET TAIL POINTER TO CONNECTION
340
DONE

FIG. 7

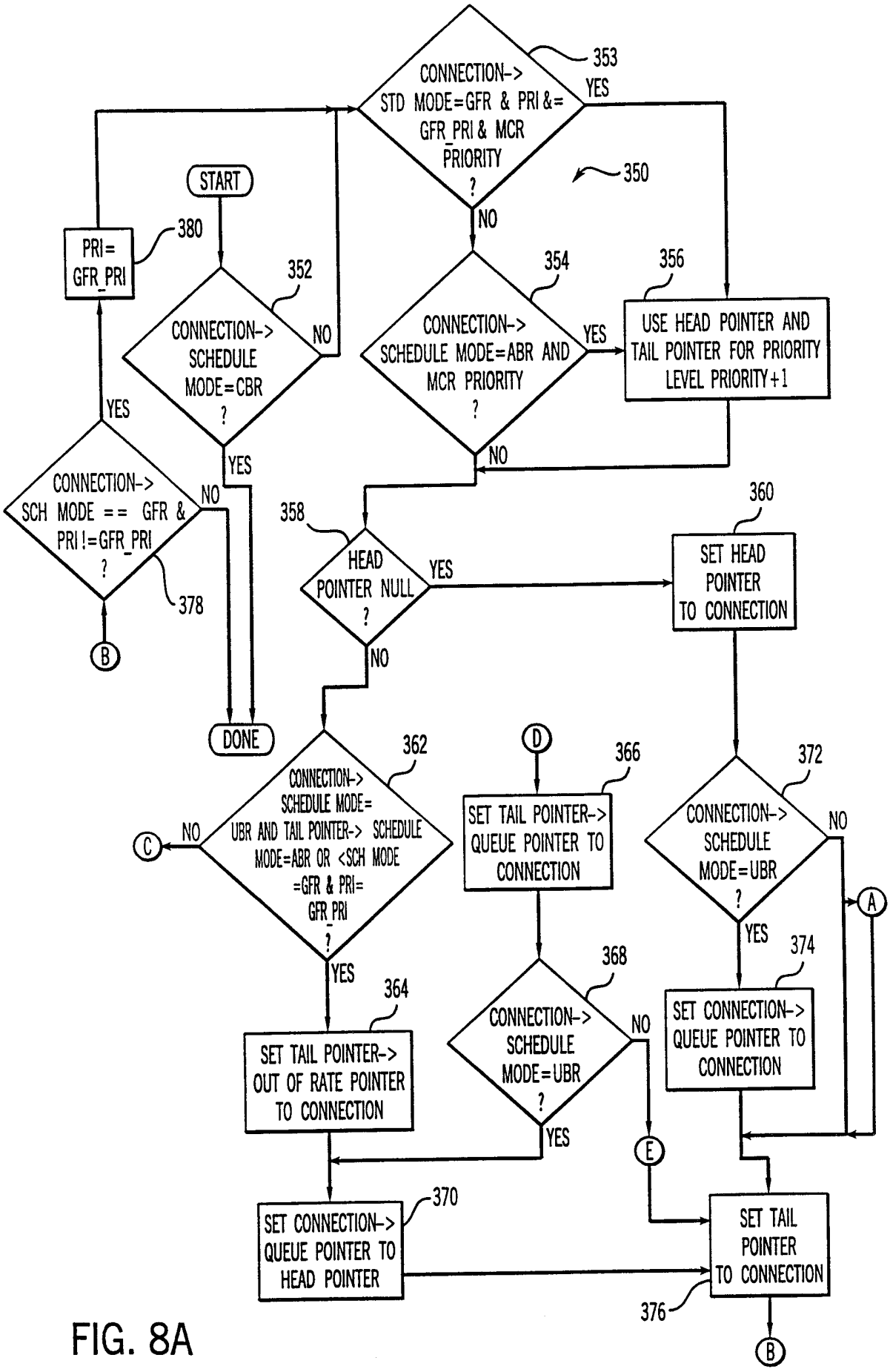

START
CONNECTION-> STD MODE=GFR & PRI &= GFR_PRI & MCR PRIORITY ?
353
YES
NO
350
PRI= GFR_PRI
380
CONNECTION-> SCHEDULE MODE=CBR ?
352
NO
YES
CONNECTION-> SCHEDULE MODE=ABR AND MCR PRIORITY ?
354
YES
USE HEAD POINTER AND TAIL POINTER FOR PRIORITY LEVEL PRIORITY+1
356
NO
CONNECTION-> SCH MODE == GFR & PRI!=GFR_PRI ?
NO
YES
378
B
DONE
HEAD POINTER NULL ?
358
YES
SET HEAD POINTER TO CONNECTION
360
NO
CONNECTION-> SCHEDULE MODE= UBR AND TAIL POINTER-> SCHEDULE MODE=ABR OR <SCH MODE =GFR & PRI= GFR_PRI ?
362
NO
C
YES
SET TAIL POINTER-> OUT OF RATE POINTER TO CONNECTION
364
D
SET TAIL POINTER-> QUEUE POINTER TO CONNECTION
366
CONNECTION-> SCHEDULE MODE=UBR ?
368
NO
YES
E
CONNECTION-> SCHEDULE MODE=UBR ?
372
NO
YES
A
SET CONNECTION-> QUEUE POINTER TO CONNECTION
374
SET CONNECTION-> QUEUE POINTER TO HEAD POINTER
370
SET TAIL POINTER TO CONNECTION
376
B

FIG. 8A

FIG. 8B

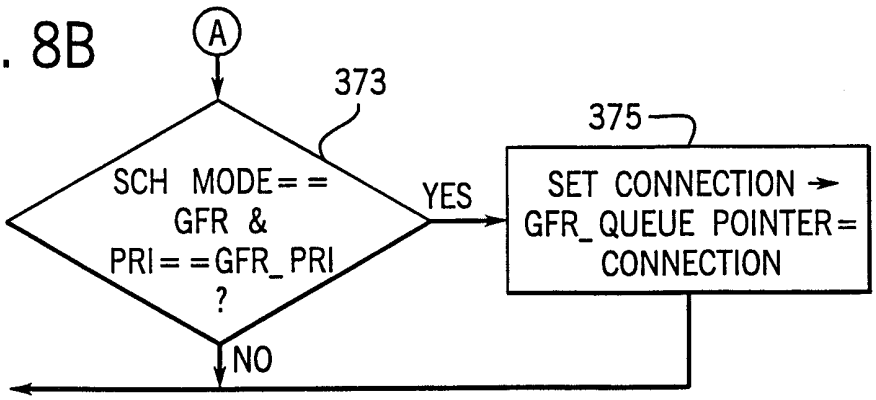


FIG. 8C

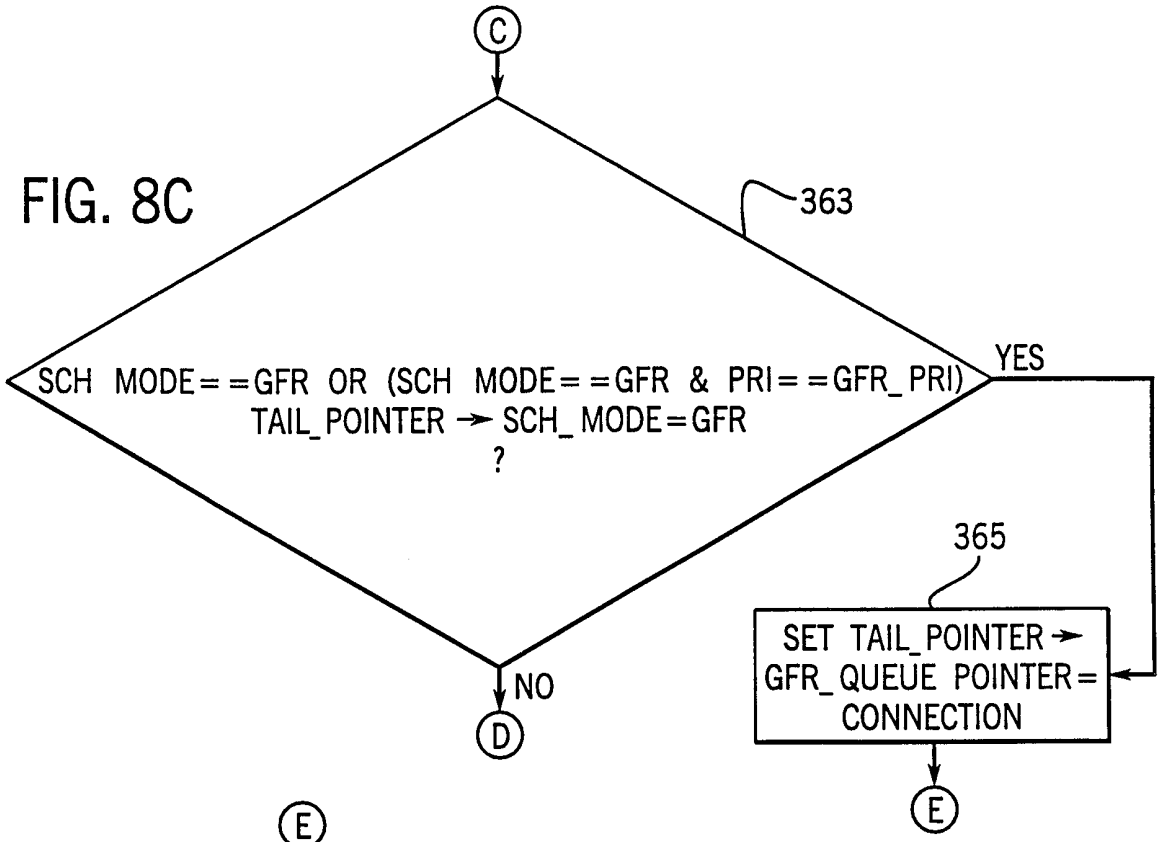


FIG. 8D

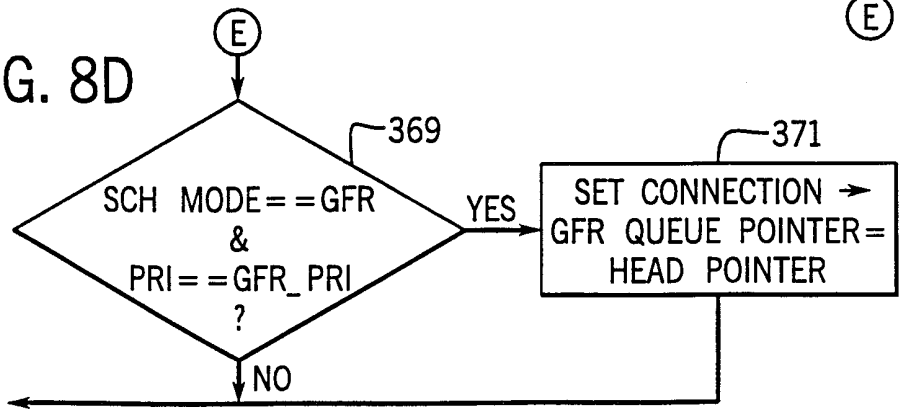


FIG. 9

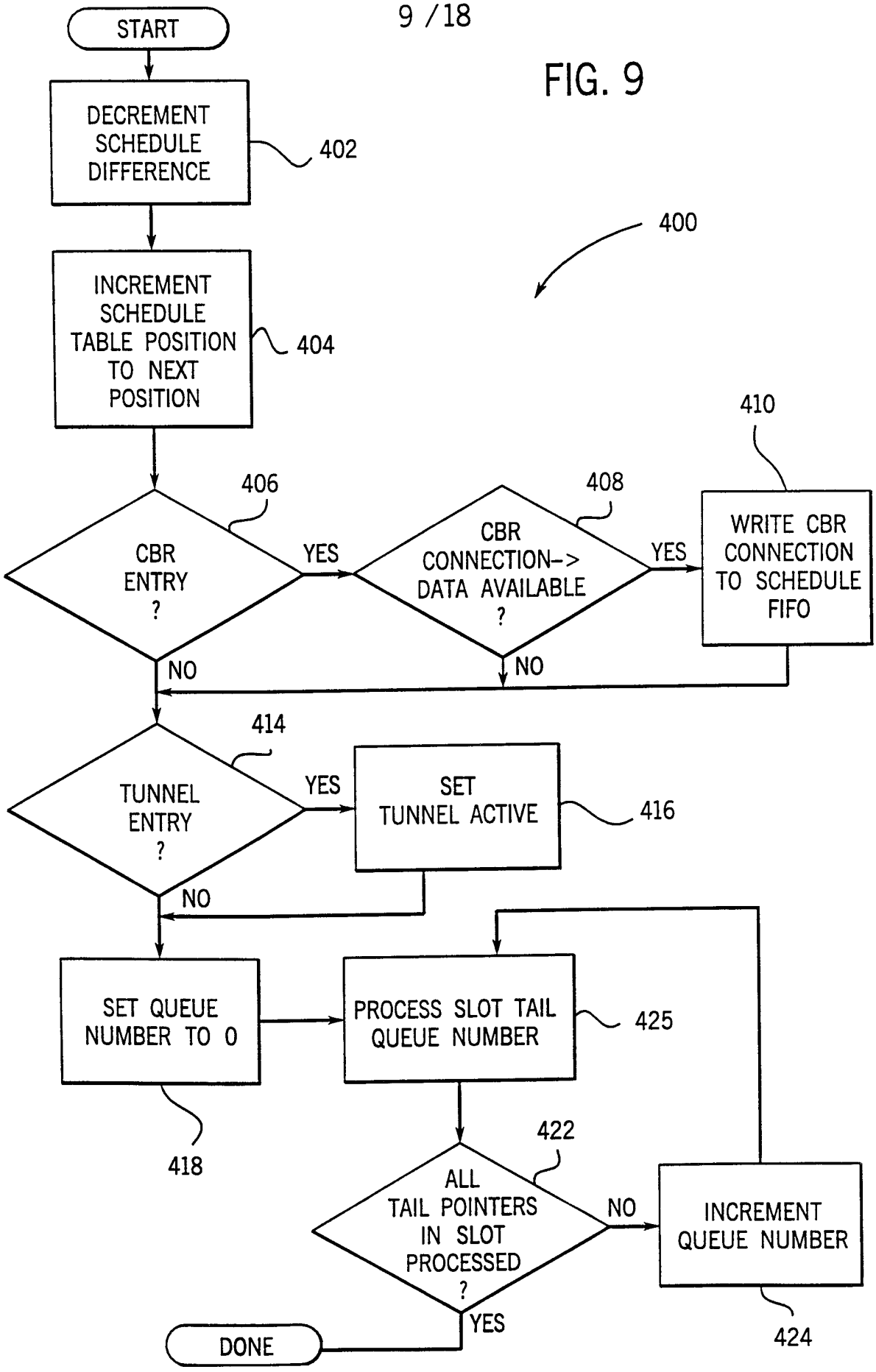

START
DECREMENT SCHEDULE DIFFERENCE
402
400
INCREMENT SCHEDULE TABLE POSITION TO NEXT POSITION
404
410
406
408
CBR ENTRY ?
YES
CBR CONNECTION-> DATA AVAILABLE ?
YES
WRITE CBR CONNECTION TO SCHEDULE FIFO
NO
NO
414
TUNNEL ENTRY ?
YES
SET TUNNEL ACTIVE
416
NO
SET QUEUE NUMBER TO 0
PROCESS SLOT TAIL QUEUE NUMBER
425
418
422
ALL TAIL POINTERS IN SLOT PROCESSED ?
NO
INCREMENT QUEUE NUMBER
YES
DONE
424

FIG. 10

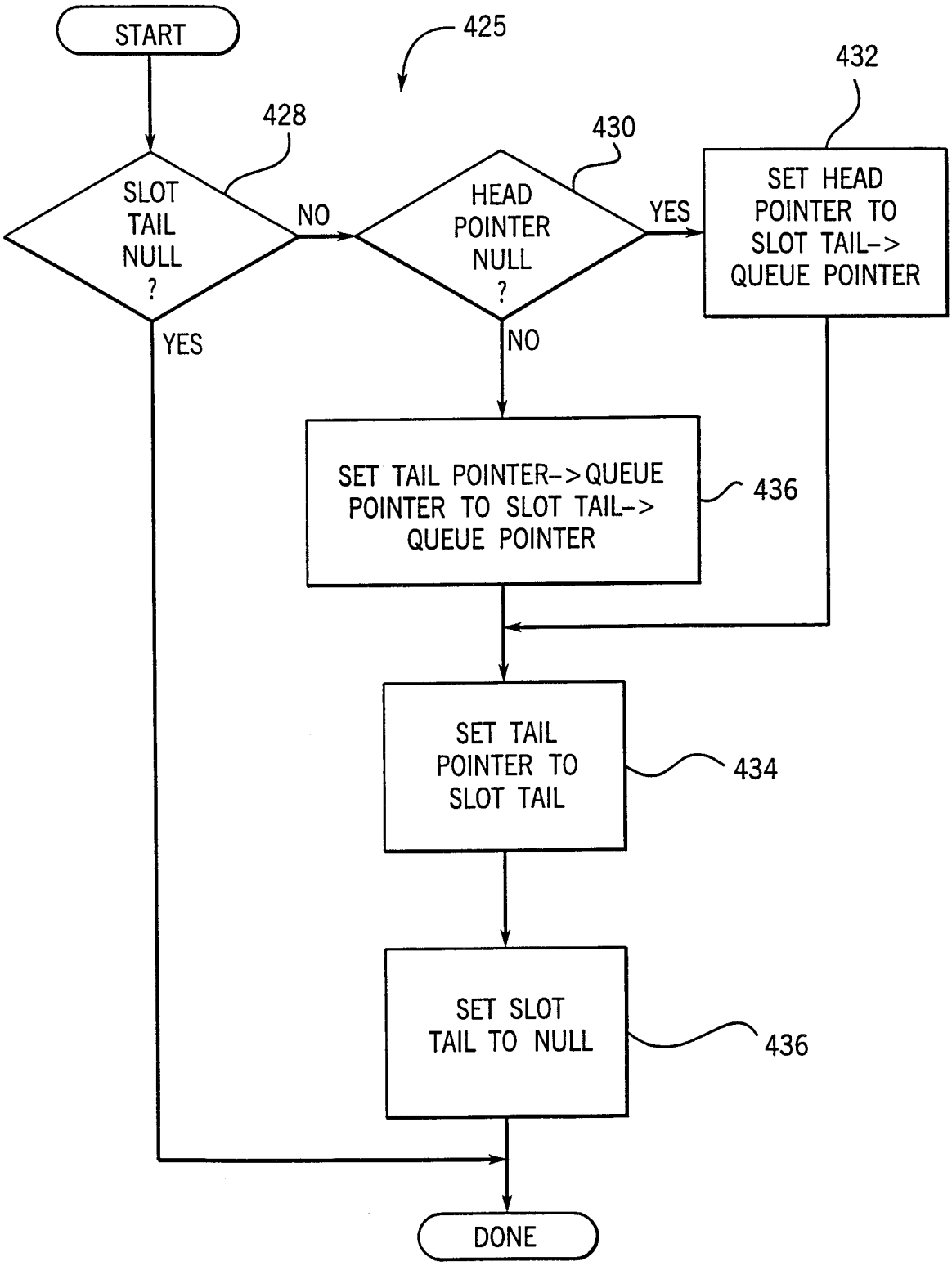
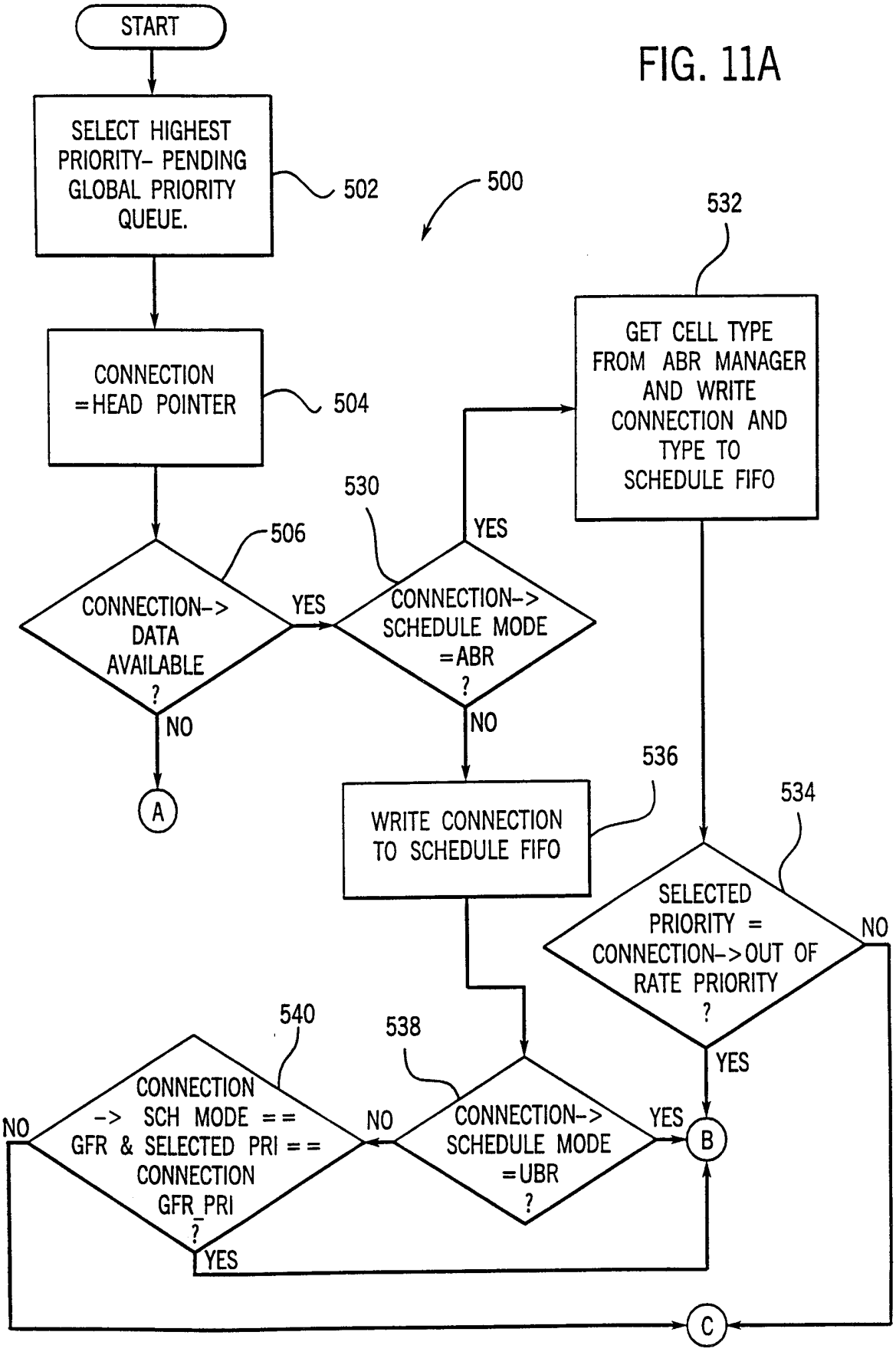

START
425
428
430
432
SLOT TAIL NULL ?
NO
HEAD POINTER NULL ?
YES
SET HEAD POINTER TO SLOT TAIL-> QUEUE POINTER
YES
NO
SET TAIL POINTER-> QUEUE POINTER TO SLOT TAIL-> QUEUE POINTER
436
SET TAIL POINTER TO SLOT TAIL
434
SET SLOT TAIL TO NULL
436
DONE

FIG. 11A


START
SELECT HIGHEST PRIORITY- PENDING GLOBAL PRIORITY QUEUE.
502
500
532
GET CELL TYPE FROM ABR MANAGER AND WRITE CONNECTION AND TYPE TO SCHEDULE FIFO
CONNECTION =HEAD POINTER
504
530
YES
506
CONNECTION-> DATA AVAILABLE ?
YES
CONNECTION-> SCHEDULE MODE =ABR ?
NO
NO
A
536
WRITE CONNECTION TO SCHEDULE FIFO
534
SELECTED PRIORITY = CONNECTION->OUT OF RATE PRIORITY ?
NO
YES
540
538
CONNECTION -> SCH MODE == GFR & SELECTED PRI == CONNECTION GFR_PRI ?
NO
CONNECTION-> SCHEDULE MODE =UBR ?
YES
B
NO
YES
C

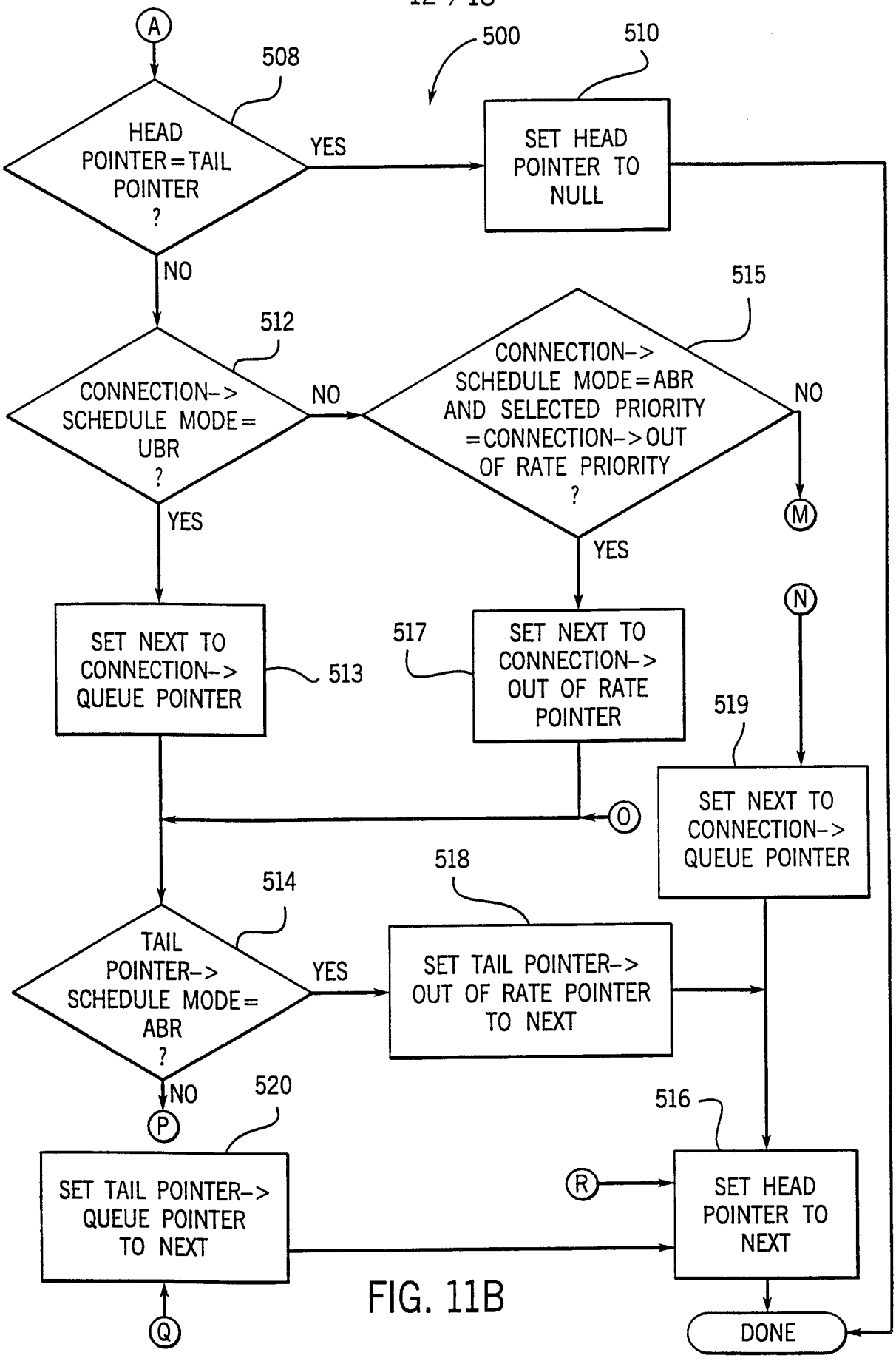

A
500
508
510
HEAD POINTER=TAIL POINTER ?
YES
SET HEAD POINTER TO NULL
NO
515
512
CONNECTION-> SCHEDULE MODE= UBR ?
NO
CONNECTION-> SCHEDULE MODE=ABR AND SELECTED PRIORITY =CONNECTION-> OUT OF RATE PRIORITY ?
NO
M
YES
YES
N
SET NEXT TO CONNECTION-> QUEUE POINTER
513
517
SET NEXT TO CONNECTION-> OUT OF RATE POINTER
519
SET NEXT TO CONNECTION-> QUEUE POINTER
O
518
514
TAIL POINTER-> SCHEDULE MODE= ABR ?
YES
SET TAIL POINTER-> OUT OF RATE POINTER TO NEXT
NO
P
520
516
R
SET TAIL POINTER-> QUEUE POINTER TO NEXT
SET HEAD POINTER TO NEXT
Q
DONE

FIG. 11B

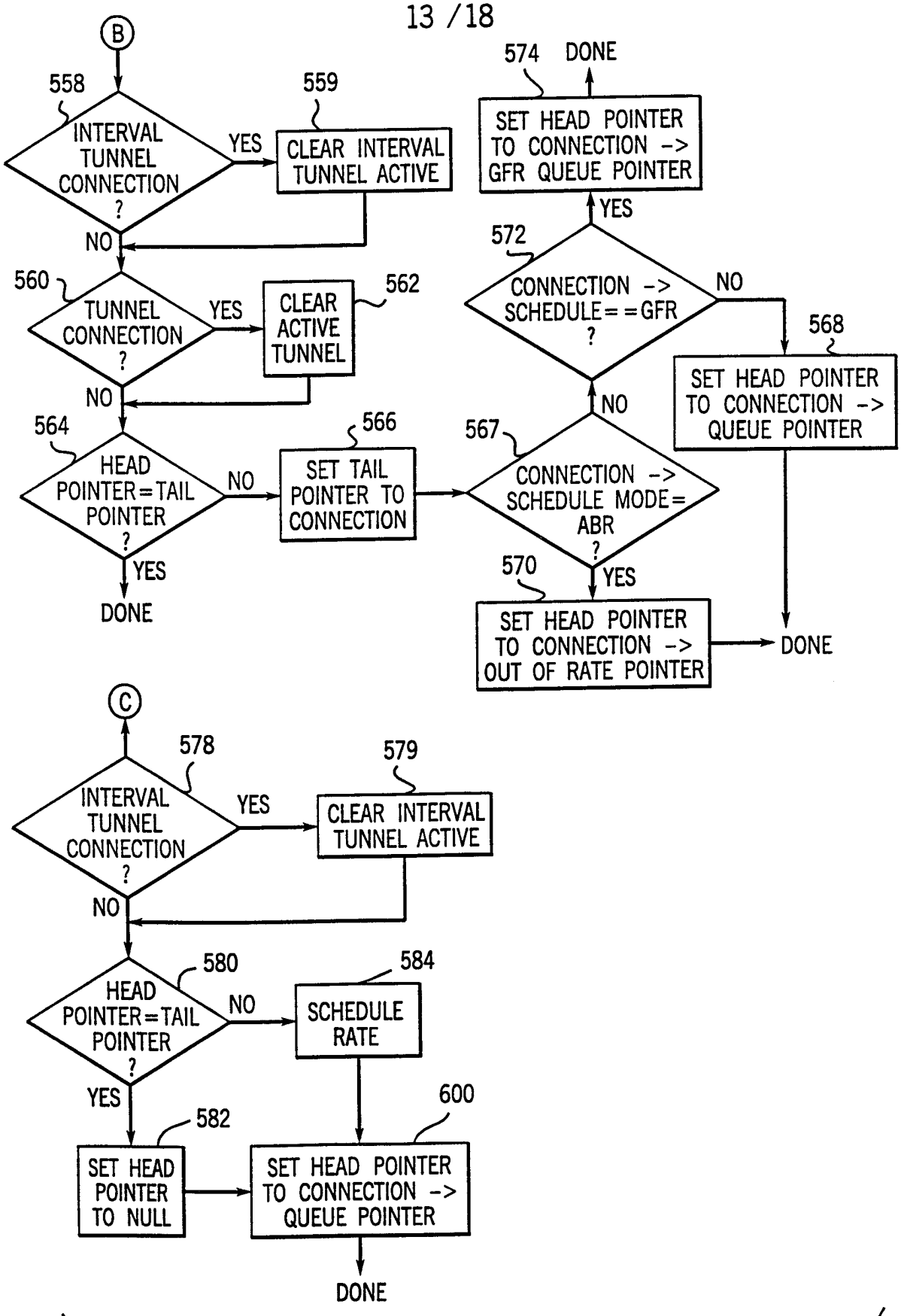

B
558
INTERVAL TUNNEL CONNECTION ?
YES
559
CLEAR INTERVAL TUNNEL ACTIVE
NO
560
TUNNEL CONNECTION ?
YES
562
CLEAR ACTIVE TUNNEL
NO
564
HEAD POINTER=TAIL POINTER ?
NO
566
SET TAIL POINTER TO CONNECTION
YES
DONE
567
CONNECTION -> SCHEDULE MODE= ABR ?
YES
570
SET HEAD POINTER TO CONNECTION -> OUT OF RATE POINTER
DONE
NO
572
CONNECTION -> SCHEDULE==GFR ?
NO
568
SET HEAD POINTER TO CONNECTION -> QUEUE POINTER
YES
574
DONE
SET HEAD POINTER TO CONNECTION -> GFR QUEUE POINTER
C
578
INTERVAL TUNNEL CONNECTION ?
YES
579
CLEAR INTERVAL TUNNEL ACTIVE
NO
580
HEAD POINTER=TAIL POINTER ?
NO
584
SCHEDULE RATE
YES
582
SET HEAD POINTER TO NULL
600
SET HEAD POINTER TO CONNECTION -> QUEUE POINTER
DONE

FIG. 11C

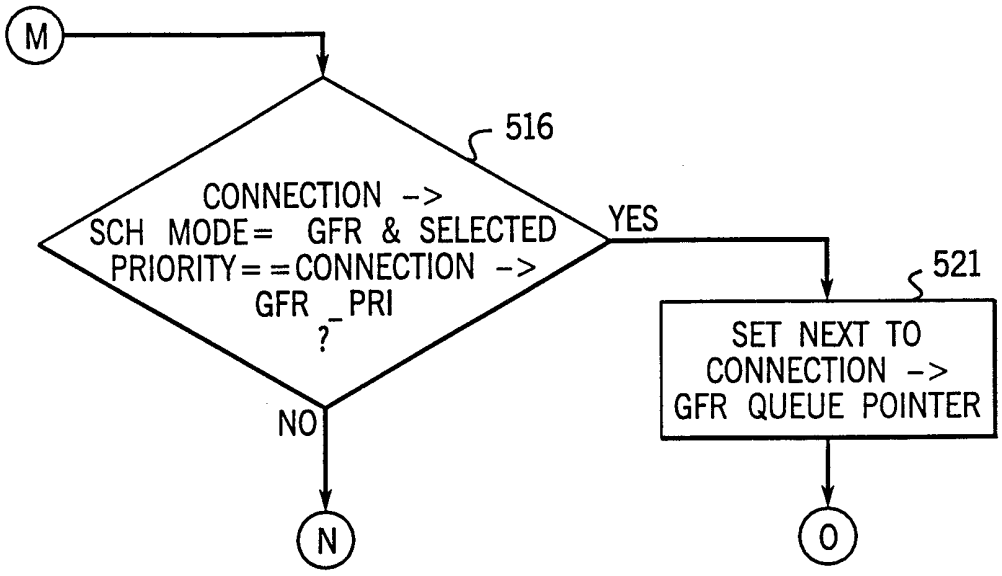


FIG. 11D

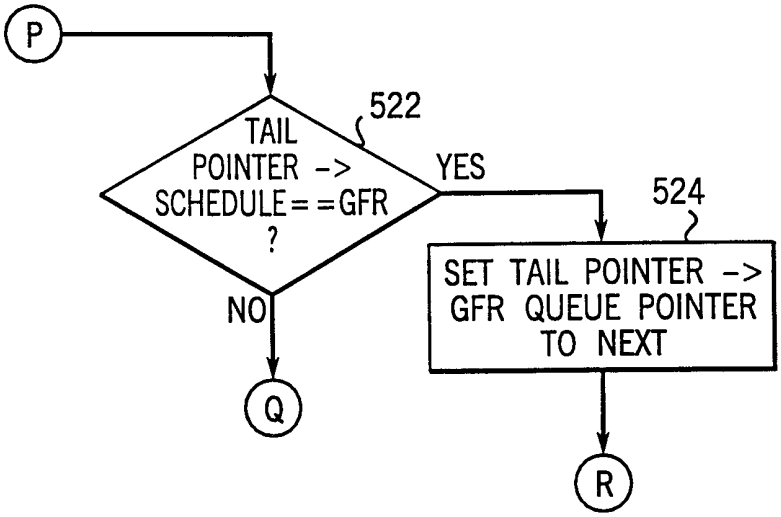


FIG. 11E

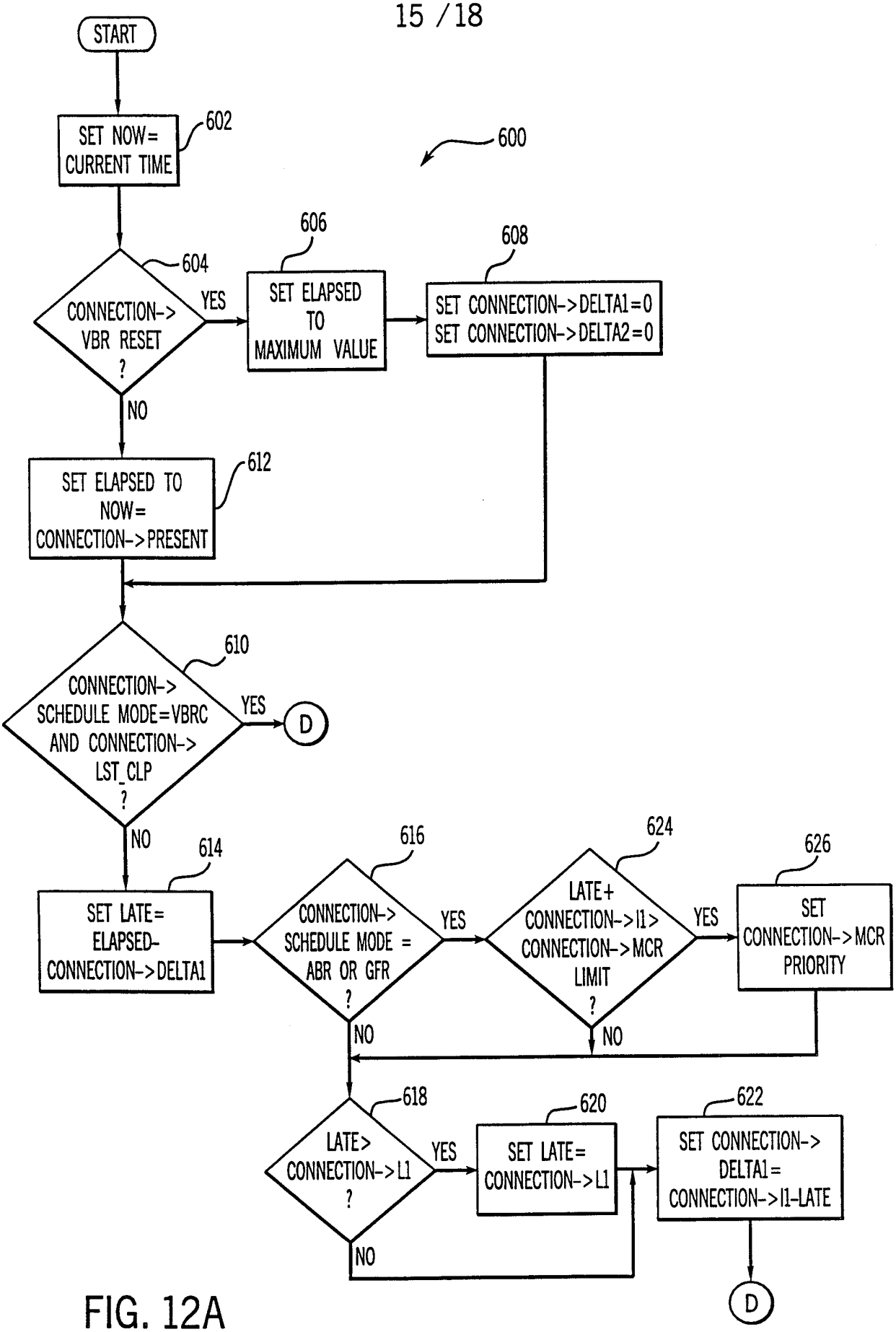
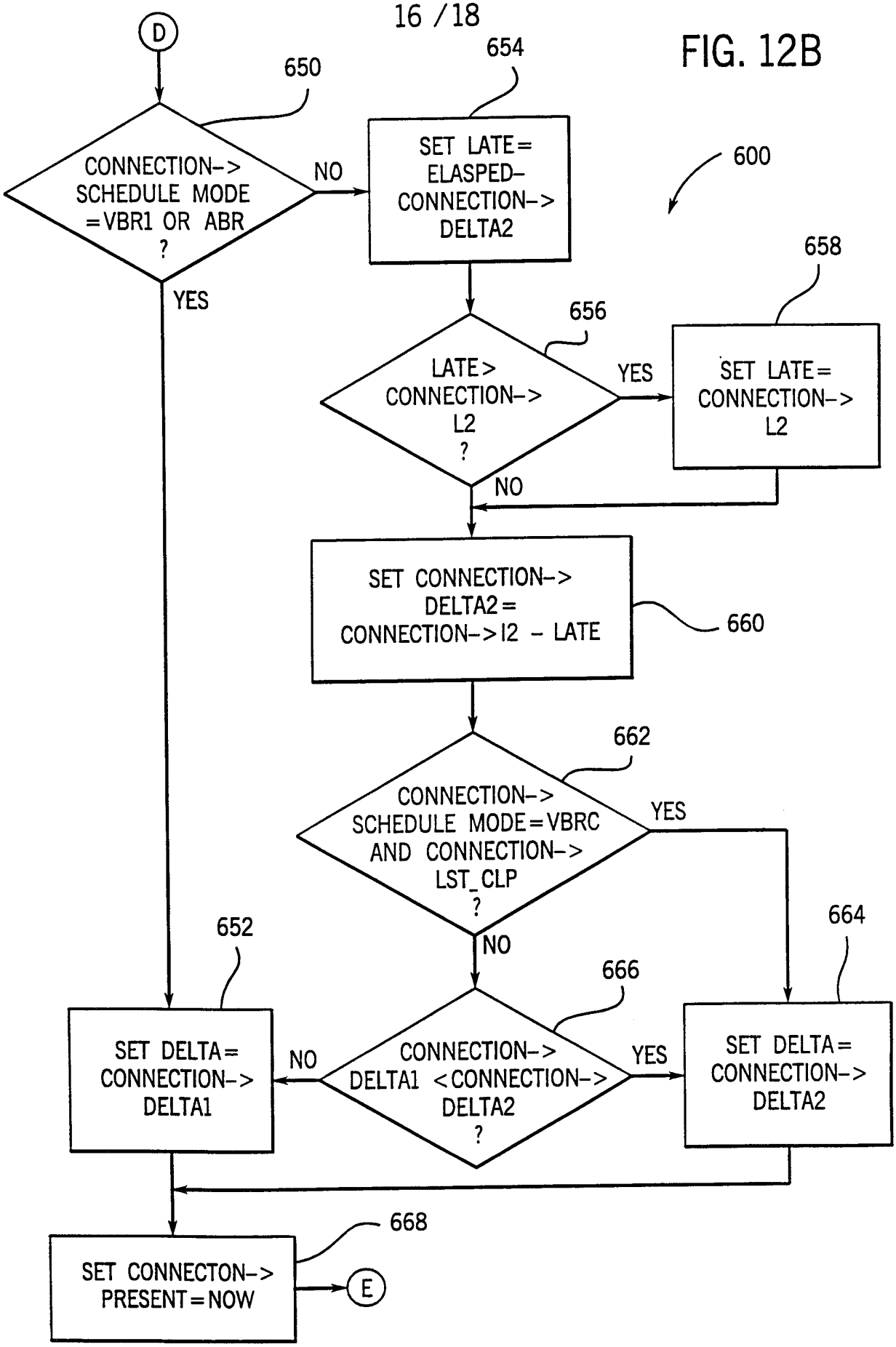

START
SET NOW= CURRENT TIME
602
600
CONNECTION-> VBR RESET ?
604
YES
606
SET ELAPSED TO MAXIMUM VALUE
608
SET CONNECTION->DELTA1=0
SET CONNECTION->DELTA2=0
NO
SET ELAPSED TO NOW= CONNECTION->PRESENT
612
CONNECTION-> SCHEDULE MODE=VBRC AND CONNECTION-> LST_CLP ?
610
YES
D
NO
SET LATE= ELAPSED- CONNECTION->DELTA1
614
CONNECTION-> SCHEDULE MODE = ABR OR GFR ?
616
YES
LATE+ CONNECTION->I1> CONNECTION->MCR LIMIT ?
624
YES
SET CONNECTION->MCR PRIORITY
626
NO
NO
LATE> CONNECTION->L1 ?
618
YES
SET LATE= CONNECTION->L1
620
SET CONNECTION-> DELTA1= CONNECTION->I1-LATE
622
NO
D

FIG. 12A

FIG. 12B


D
650
CONNECTION-> SCHEDULE MODE =VBR1 OR ABR ?
NO
YES
654
SET LATE= ELASPED- CONNECTION-> DELTA2
600
658
656
LATE> CONNECTION-> L2 ?
YES
SET LATE= CONNECTION-> L2
NO
SET CONNECTION-> DELTA2= CONNECTION-> I2 - LATE
660
662
CONNECTION-> SCHEDULE MODE=VBRC AND CONNECTION-> LST_CLP ?
YES
664
NO
652
666
SET DELTA= CONNECTION-> DELTA1
NO
CONNECTION-> DELTA1 <CONNECTION-> DELTA2 ?
YES
SET DELTA= CONNECTION-> DELTA2
668
SET CONNECTON-> PRESENT=NOW
E

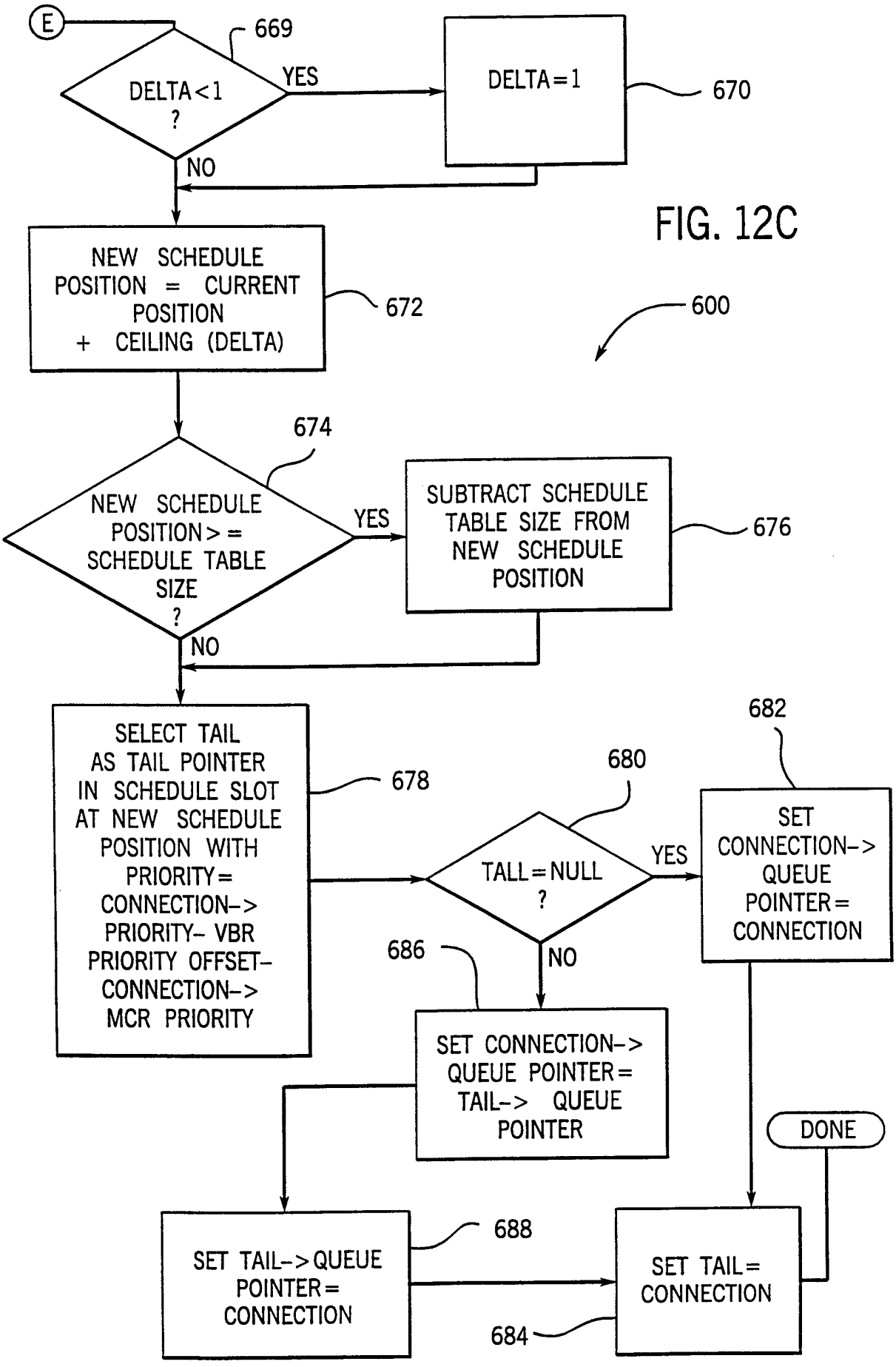
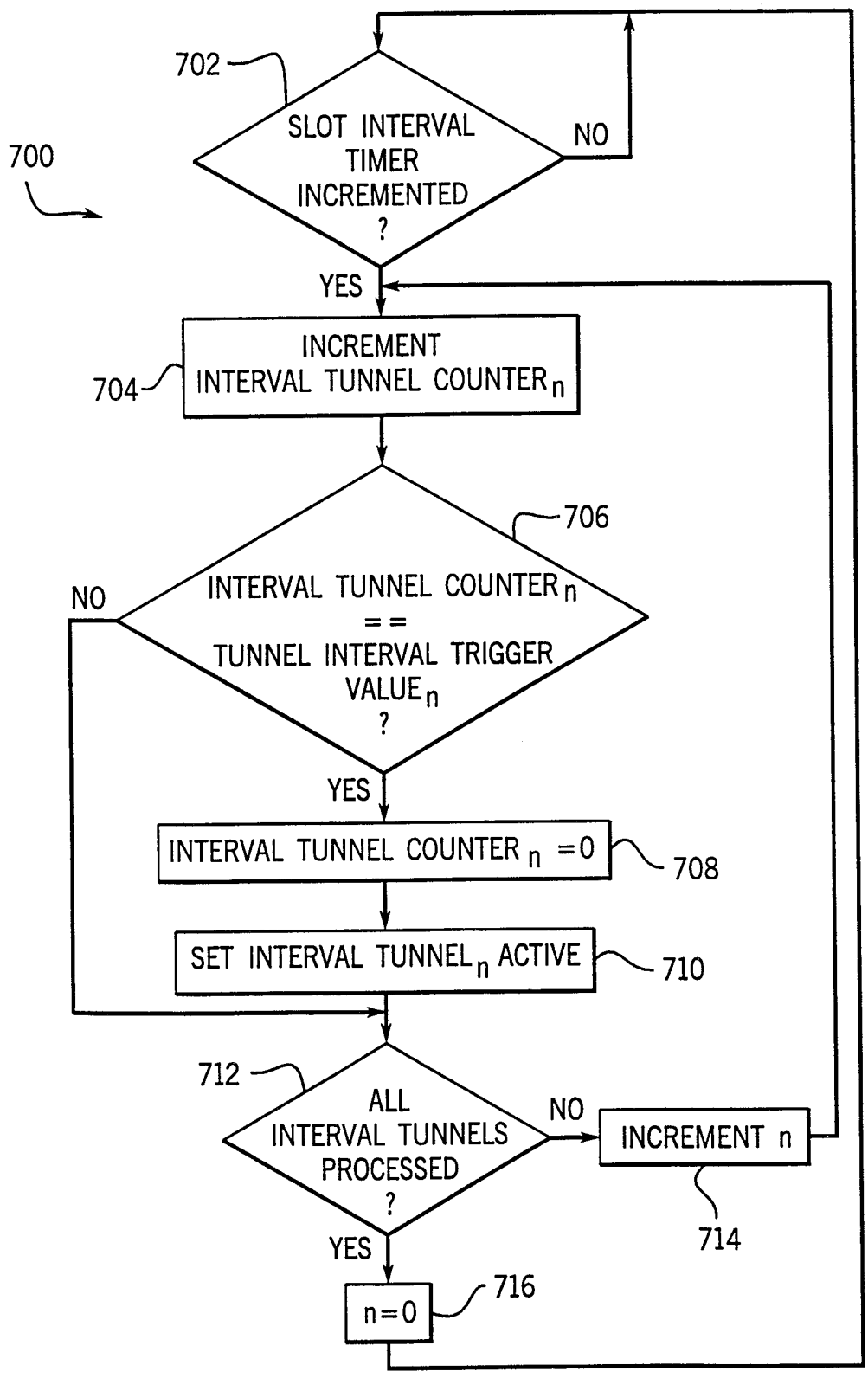

E
669
DELTA<1 ?
YES
DELTA=1
670
NO
NEW SCHEDULE POSITION = CURRENT POSITION + CEILING (DELTA)
672
600
674
NEW SCHEDULE POSITION>= SCHEDULE TABLE SIZE ?
YES
SUBTRACT SCHEDULE TABLE SIZE FROM NEW SCHEDULE POSITION
676
NO
SELECT TAIL AS TAIL POINTER IN SCHEDULE SLOT AT NEW SCHEDULE POSITION WITH PRIORITY= CONNECTION-> PRIORITY- VBR PRIORITY OFFSET- CONNECTION-> MCR PRIORITY
678
680
682
TALL=NULL ?
YES
SET CONNECTION-> QUEUE POINTER= CONNECTION
686
NO
SET CONNECTION-> QUEUE POINTER= TAIL-> QUEUE POINTER
DONE
SET TAIL-> QUEUE POINTER= CONNECTION
688
SET TAIL= CONNECTION
684

FIG. 12C

FIG. 13


700
702
SLOT INTERVAL TIMER INCREMENTED ?
NO
YES
704
INCREMENT INTERVAL TUNNEL COUNTER n
706
INTERVAL TUNNEL COUNTER n == TUNNEL INTERVAL TRIGGER VALUE n ?
NO
YES
INTERVAL TUNNEL COUNTER n =0
708
SET INTERVAL TUNNEL n ACTIVE
710
712
ALL INTERVAL TUNNELS PROCESSED ?
NO
INCREMENT n
714
YES
716
n=0

INTERNATIONAL SEARCH REPORT

Inter nal Application No
PCT/US 99/05513

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 H04L12/56 H04Q11/04

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 6 H04L H04Q

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 practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5 392 280 A (ZHENG OIN) 21 February 1995 see column 3, line 13 - column 4, line 4 --- -/--	1,4-6, 11, 17-19, 21,25,26

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
4 June 1999	15/06/1999

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Veen, G

1

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Inter nal Application No
PCT/US 99/05513

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
A	MARK B L ET AL: "LARGE CAPACITY MULTICLASS ATM CORE SWITCH ARCHITECTURE" ISS '97. WORLD TELECOMMUNICATIONS CONGRESS. (INTERNATIONAL SWITCHIN SYMPOSIUM), GLOBAL NETWORK EVOLUTION: CONVERGENCE OR COLLISION? TORONTO, SEPT. 21 - 26, 1997, vol. 1, 21 September 1997, pages 417-423, XP000720547 ABE S ET AL see abstract see page 419, left-hand column, line 49 - right-hand column, line 5 ---	1-13, 20-24,30
A	WO 97 42735 A (MAKER COMMUNICATIONS INC) 13 November 1997 see page 7, line 1 - page 8, line 11 -----	6,21,22, 25,26

1

Form PCT/ISA/210 (continuation of second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

...information on patent family members

Inter...nal Application No
PCT/US 99/05513

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5392280 A	21-02-1995	JP 8046590 A	16-02-1996
WO 9742735 A	13-11-1997	AU 2940197 A	26-11-1997