



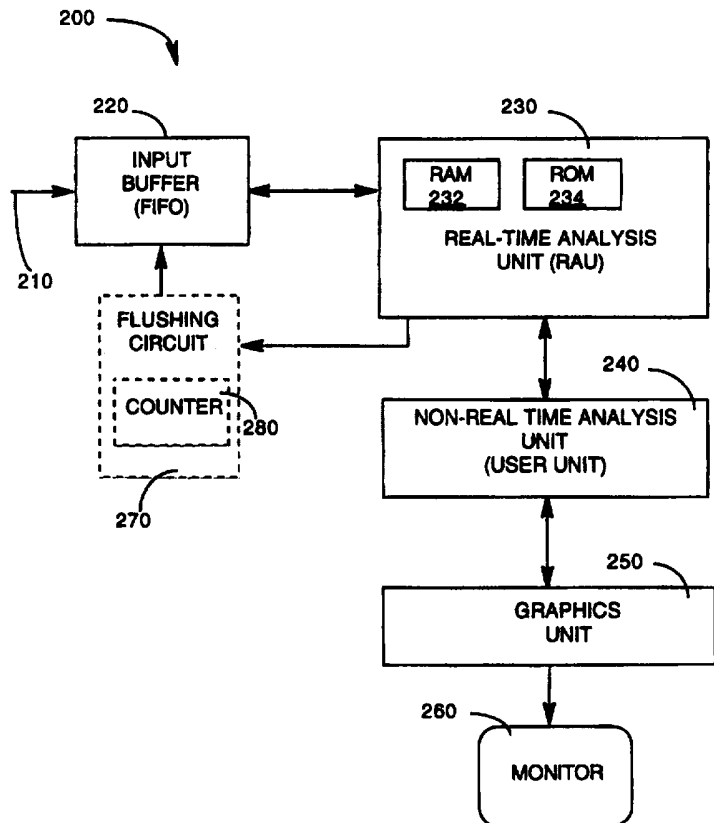
INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

<p>(51) International Patent Classification ⁶ : H04J 3/14, 3/24, H04Q 11/04</p>	<p>A1</p>	<p>(11) International Publication Number: WO 97/34386 (43) International Publication Date: 18 September 1997 (18.09.97)</p>
<p>(21) International Application Number: PCT/US97/04009 (22) International Filing Date: 13 March 1997 (13.03.97) (30) Priority Data: 60/013,361 13 March 1996 (13.03.96) US (71) Applicant: SARNOFF CORPORATION [US/US]; 201 Washington Road, CN-5300, Princeton, NJ 08543-5300 (US). (72) Inventors: DIETERICH, Charles, Benjamin; 7 Euclid Avenue, Kingston, NJ 08528 (US). GREENBERG, Arthur, Lee; 24 Mission Hill Road, Annandale, NJ 08801 (US).</p>	<p>(81) Designated States: CA, JP, KR, MX, European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE). Published <i>With international search report.</i></p>	

(54) Title: METHOD AND APPARATUS FOR ANALYZING AND MONITORING PACKET STREAMS

(57) Abstract

A method and apparatus (200) for analyzing and monitoring packet streams in "real time". The packet analyzer comprises an input buffer (220), a real-time analysis unit (230), a non-real-time analysis unit (240), a graphics unit (250), a monitor (260) and a flushing circuit (270). A packet stream is received into the input buffer where the data is either read by the real-time analysis unit or flushed by the flushing circuit. Messages are passed between the real-time analysis unit and the non-real-time analysis unit to report on detected errors or to update packet stream information. In turn, real-time packet stream information are displayed and updated on a display via the graphic unit.



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.

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

METHOD AND APPARATUS FOR ANALYZING AND MONITORING PACKET STREAMS

This application claims the benefit of U.S. Provisional Application
5 No. 60/013,361 filed March 13, 1996.

The present invention relates to an apparatus and concomitant
method for analyzing and monitoring packet streams in a communication
system. More particularly, this invention relates to a method and
10 apparatus that evaluates transport packet streams in "real time" for
detecting errors, verifying the consistency of time base information and
displaying important packet stream information.

BACKGROUND OF THE INVENTION

15 The increasing demand for digital video/audio information presents
an ever increasing problem of monitoring the transmission or storage of
data in data communication. As the transmission bandwidth increases
in response to greater demand, it becomes increasingly more difficult to
monitor the enormous amount of transmitted information in real time.

20 In order to monitor and evaluate the status of a transport stream in
real time, a method or apparatus must be able to handle the high transfer
rate of the transmission channel. However, this requirement is
computationally expensive. In fact, since time is such a scarce resource
in real time packet analysis, it may become prohibitively expensive.

25 Therefore, a need exists in the art for a method and apparatus for
performing real time packet analysis without the associated high
computational expense. Specifically, a need exists for a method and
apparatus for detecting errors, verifying the consistencies of time base
information and displaying important packet stream information.

30

SUMMARY OF THE INVENTION

The present invention is a method and apparatus for analyzing and
monitoring packet streams. The apparatus, a packet analyzer, comprises
an input buffer, a real-time analysis unit, a non-real-time analysis unit, a
35 graphics unit and a monitor. A packet stream is received into the input

buffer where the data is read by the real-time analysis unit. Messages are passed between the real-time analysis unit and the non-real-time analysis unit to report on detected errors or to update packet stream information. In turn, real time packet stream information are displayed and updated
5 on a display via the graphic unit.

BRIEF DESCRIPTION OF THE DRAWINGS

The teachings of the present invention can be readily understood by considering the following detailed description in conjunction with the
10 accompanying drawings, in which:

FIG. 1 illustrates a block diagram of a simplified conventional packet stream system;

FIG. 2 illustrates a packet analyzer of the present invention for performing real time packet analysis;

15 FIG. 3 illustrates a flowchart of a method for detecting packet framing errors within a transport stream;

FIG. 4 illustrates a flowchart of a method for detecting and flushing payloads carrying scrambled data in the video and audio packets;

20 FIG. 5 illustrates a flowchart of a method for verifying the consistency of time base information in a transport stream;

FIG. 6 illustrates a flowchart of a second method for verifying the consistency of time base information in a transport stream;

FIG. 7 illustrates a flowchart of a third method for verifying the consistency of time base information in a transport stream;

25 FIG. 8 illustrates a graphical interface display showing a program guide view;

FIG. 9 illustrates a graphical interface display showing a video rates view;

30 FIG. 10 illustrates a graphical interface display showing a video rates graph; and

FIG. 11 illustrates a graphical interface display showing a multiplexed streams scan.

To facilitate understanding, identical reference numerals have been used, where possible, to designate identical elements that are common to
35 the figures.

DETAILED DESCRIPTION

FIG. 1 depicts a block diagram of a simplified structure of a conventional packet stream system 100. More specifically, a “transport stream” as defined in accordance with the MPEG standards is used in the packet stream system illustrated in FIG. 1. Although the present invention is described below using the MPEG transport stream as an example, those skilled in the art will realize that the present invention can be applied to any packet streams, i.e., a MPEG “program stream” or any other packet streams in accordance with other formats.

System 100 includes a video encoder 120 for receiving and encoding video data on path 110 into an elementary video bitstream. Similarly, the system also includes an audio encoder 122 for receiving and encoding audio data on path 112 into an elementary audio bitstream. In turn, these bitstreams are sent to packetizers 130 and 132 where the elementary bitstreams are converted into packets. Information for using the packets independently of the transport stream may be added when the packets are formed. Thus, non-audio/video data are allowed, but they are not shown in FIG. 1.

The packets are received and multiplexed by the transport stream multiplexor 140 to produce a transport stream 145. Packets constructed from elementary streams that form a program (a group of “Packet Identifiers” (PIDs) with associated video and audio data) generally share a common time base. Thus, the transport stream may contain one or more programs with one or more independent time bases, where the time bases are used for synchronized presentation. The time bases of different programs within a transport stream may be different.

The transport stream 145 is transmitted over a transmission channel 150, which may further incorporate separate channel specific encoder and decoder (not shown). Next, the transport stream 145 is demultiplexed and decoded by a transport stream demultiplexor 160, where the elementary streams serve as inputs to video decoder 170 and audio decoder 190, whose outputs are decoded video signals on path 175 and audio signals on path 195 respectively.

Furthermore, timing information is also extracted by the transport stream demultiplexor 160 and delivered to clock control 180 for synchronizing the video and audio decoders with each other and with the channel. Synchronization of the decoders with the channel is
5 accomplished through the use of the "Program Clock Reference" (PCR) in the transport stream. The PCR is a time stamp encoding the timing of the bitstream itself and is used to derive the decoder timing.

FIG. 2 illustrates a packet analyzer 200 of the present invention for monitoring and performing real time packet analysis on a multiplexed
10 bitstream, such as a MPEG-2 transport stream. The packet analyzer comprises an input buffer 220, a real-time analysis unit (RAU) 230, a non-real-time analysis unit (User Unit) 240, a graphics unit 250, a monitor 260 and an optional flushing circuit 270.

A transport stream on path 210 is received into the input buffer 220
15 in a byte-wide manner. In the preferred embodiment, the input buffer is a First-In-First-Out buffer (FIFO) having a length of 512 bytes and a width of 9 bits. A conventional circuitry, i.e., a controller (not shown) detects the start of a packet signal and begins the process of loading bytes into the FIFO. Since data is arriving at a steady rate, the FIFO serves to buffer the
20 incoming data from a processing unit such as the RAU 230, thereby relieving the RAU from a rigid data processing schedule. The FIFO bits are organized as 8 data bits and one bit flag (9th bit). This 9th bit flag is used for packet framing and/or transmission of the "DVALID" flag information (packet status information) as discussed below. DVALID is
25 defined as a signal which indicates correct detection of the signal at the output of the channel (150). For example, DVALID would be negated if a forward error correction code decoder in the channel failed to correct a channel error. DVALID is asserted at the beginning of each packet as an indication of that packet's validity.

30 Although the FIFO is illustrated as a separate unit, those skilled in the art will realize that the FIFO can be implemented in different manners, e.g., within the RAU 230. Similarly, although the present invention incorporates a one bit flag, those skilled in the art will realize that the flag can be implemented with more than a single bit.

The packet analyzer includes a RAU 230 for processing the packets stored in the FIFO. The RAU consists of a microcomputer, such as a TMS320C31 available from Texas Instruments, Inc., with the associated random access memory (RAM) 232 and read only memory (ROM) 234.

5 The RAU may optionally incorporate other storage media, e.g., various disk drives (not shown). The RAU also contains a dual-port memory interface for communicating with the User Unit 240. Due to the high data transfer rate of the channel, the RAU serves as a dedicated processing unit in performing real time packet analysis.

10 The User Unit 240 is a general purpose computer, such as a Personal Computer with an Intel 80486 central processing unit (CPU). The User Unit 240 may optionally incorporate random access memory (RAM) and other storage media such as read only memory (ROM) and various disk drives (not shown). The User Unit executes a plurality of
15 software applications such as Microsoft Windows which allows a software application to send and receive messages. To illustrate, the messages can be sent via a dual port memory scheme over a high-speed interface such as an ISA bus. Messages are sent between the User Unit and the RAU via interrupts. The User Unit serves to process the messages from the RAU
20 and causes the results of the packet stream analysis to be displayed on the monitor 260 through a conventional graphics unit 250, such as a VGA graphics card.

Finally, the packet analyzer may incorporate an optional flushing circuit 270 for removing unwanted packets from the FIFO 220. The
25 flushing circuit incorporates a counter 280 which is controlled and set by the RAU to a particular value, X. As the counter decrements from X to zero, an X number of bytes is flushed from the FIFO, thereby reducing the computational expense of performing real time packet analysis, which is discussed below with reference to FIG. 4. Various methods of flushing a
30 FIFO are known, i.e., advancing a read pointer in the FIFO or simply reading the FIFO without sending the data to any destinations. The flushing circuit 270 is optional to the extent that it can be implemented as a hardware circuit (programmable logic arrays) or as a software algorithm. In the preferred embodiment, the flushing circuit is
35 implemented as a hardware circuit. However, those skilled in the art will

realize that the flushing circuit can be implemented as a software algorithm within the RAU.

The specific structure of the packet analyzer 200 as illustrated in FIG. 2 is used to implement a plurality of methods for analyzing and
5 monitoring packet streams.

FIG. 3 illustrates a method 300 for detecting packet framing errors within a transport stream. Generally, packet framing data consists of information about the numerical order of bytes in a packet (i.e., a first byte in the packet, a second byte in the packet,...,etc.). The packet framing data
10 is received into the FIFO at a steady rate such as every 188 clock cycles.

However, framing errors may occur where the boundary between successive packets are incorrectly defined, i.e., having too many bytes assigned to a packet (long packet error) or having too few bytes assigned to a packet (short packet error). Other framing errors may include invalid
15 data (data invalid error) in a particular byte of a packet. When a framing error occurs, it is desirable to detect the framing error in the proper sequence. Namely, if a FIFO contains 10 packets and a packet framing error is suddenly triggered, then the framing error will appear 10 packets in the future. This premature alarm is undesirable because the
20 processing unit such as the RAU will not be able to respond to the error until many cycles later, thereby adding additional computational overhead in tracking the framing error. Thus, it is preferable to have the framing error reported when the packet arrives.

FIG. 3 illustrates a method for detecting and presenting packet
25 framing errors to the RAU. Referring to FIG. 3, the method 300 begins at step 310 and proceeds to step 320 where packet data is received and routed into the 8 data bits of the FIFO 220.

In step 330, the first byte of all packets receives a "start-of-packet" indicator as it is loaded into the FIFO 220. In the preferred embodiment,
30 the start-of-packet indicator is implemented by setting the 9th bit of the first byte of a packet to a value of "1". This will indicate to the RAU that the current byte is the start of a new packet.

In step 340, the second byte of all packets receives a "data valid" indicator as it is loaded into the FIFO 220. In the preferred embodiment,
35 the data valid indicator is implemented by multiplexing the existing

“DVALID” signal into the 9th bit of the second byte of a packet. In MPEG, the DVALID is a flag used for indicating errors in the transmitted data. Thus, it is efficient to simply multiplex this existing indicator into the 9th bit of the second byte of a packet. A DVALID value of “1” indicates that a data invalid error exists within the packet, while a DVALID value of “0” indicates that the data in the packet is valid.

However, those skilled in the art will realize that other systems, i.e., non-MPEG systems, may incorporate other methods of indicating errors in the transmitted data. Such methods may include a dedicated wire to signal to the system that erroneous data is being transmitted and that the system should ignore these data bits. As such, these other indicators can be multiplexed for setting the 9th bit of a second byte of the packet to indicate data invalid errors in the transmitted packet.

In step 350, all other bytes of each packet receive a “short-packet” indicator as they are loaded into the FIFO 220. The short-packet indicator is implemented by setting the 9th bit of all subsequent bytes of a packet to a value of “0”. This will indicate to the RAU that the current byte is not the start of a new packet.

In step 360, method 300 determines whether the 9th bit of the first byte of a packet is correctly set at the value of “1”. In the preferred embodiment, the RAU reads a byte from the FIFO in a memory-mapped fashion, while relying on the controller of the FIFO (or a state machine) to read the 9th bit of the associated byte. The controller verifies that the first byte of every packet is read from one address, “FIRST ADDRESS” and that all other bytes are read from “NEXT ADDRESS”. If the decision is affirmatively answered in step 360, then method 300 proceeds to step 370 where the current byte is confirmed as the start of a new packet. If the decision is negatively answered, method 300 proceeds to step 365 where a “framing error-long packet” flag is set, indicating a framing error within the packet. Since a read is being performed at the FIRST ADDRESS, the RAU expects the current byte to be the start of a new packet. If that is not the case, then the current packet is considered to be part of the previous packet, thereby erroneously increasing the length of the previous packet.

In step 370, method 300 determines whether the 9th bit of the second byte of a packet is set at the value of “0”. If the decision is affirmatively

answered, then method 300 proceeds to step 380 where the current packet is deemed to be carrying valid data. If the decision is negatively answered, method 300 proceeds to step 375 where a "data invalid error" flag is set, indicating a data error within the packet. This may cause the RAU to
5 flush or ignore the bytes of the current packet.

In step 380, method 300 determines whether the 9th bit of the subsequent bytes of a packet is correctly set at the value of "0". If the decision is affirmatively answered for all subsequent bytes, then method 300 proceeds to step 390 where the current packet is confirmed to be
10 correctly framed. If the decision is negatively answered, method 300 proceeds to step 385 where a "framing error-short packet" flag is set, indicating a framing error within the packet. Since a read is being performed at the NEXT ADDRESS, the RAU does not expect the current byte to be the start of a new packet. If that is not the case, then the current
15 packet is considered to have ended too soon, thereby erroneously decreasing the length of the current packet.

In step 390, method 300 determines whether there are additional packets in the FIFO. If the decision is affirmatively answered, method 300 returns to step 360 and the steps of 360 through 380 are repeated for the
20 next packet. If the decision is negatively answered, method 300 ends in step 395.

FIG. 4 illustrates a method for minimizing the number of bytes that are processed by the RAU by "flushing" payloads carrying scrambled data in the video and audio packets. In a MPEG-2 transport stream, the
25 packets contain a header, which may consist of four (4) or more bytes, and a payload having any length up to 184 bytes. Since all packets are not of the same type, some packets carry video or audio data while other carry control streams. Since the payloads of video and audio packets are often scrambled, these packets provide little information to the RAU except for
30 the fact that they have been transmitted and received. As such, the data bytes in such packets are flushed from the input buffer, thereby reducing significantly the number of bytes of data that the RAU must process.

The flushing method must distinguish payload carrying scrambled data in video and audio packets from those packets carrying important
35 control data such as adaptation information relating to time base.

Specifically, the control stream, "Program Association Table" (PAT), contains the important information on the location of other control streams in the form of "Packet Identifier" (PID) values. A PID is a unique 13 bit field, indicating the type of data that is stored in the packet payload.

5 The PID values of 0 and 1 are reserved for the PAT and the "Conditional Access Table" (CAT) respectively.

Another important control stream is the "Program Map Tables" (PMT), which contains a list of PIDs and defines whether a PID carries video, audio, or "other data" in its payload. Thus, in order to determine the content of a packet payload, the PAT and PMT must be decoded before a packet stream's PID can be determined. Since PID values do not change very often, the PAT is sent at a low data rate and successive transmissions of the PAT typically contain identical data. Thus, once a PID is determined, it can be used repeatedly to determine the payloads of its packets, thereby permitting the RAU to flush various payloads from the FIFO to conserve precious CPU cycles. Since both the RAU and the User Unit incorporate some storage media, e.g., RAM or disk drives, both unit will maintain various tables to track packet information, e.g., PID values, number of received packets and/or time base information as they are

15
20 received.

Referring to FIG. 4, the method 400 begins at step 410 and proceeds to step 420 where the first four bytes of a packet is read to determine its PID number or value. The first three bytes of a packet contain the "packet_start_code_prefix" field, which is then followed by a fourth byte containing the "stream_id" field from which the PID is obtained.

25

In step 430, the method attempts to determine the packet type from the PID value extracted in step 420 by comparing the PID value with a look-up table. Initially, although the RAU is able to extract the PID number from the packet, it may not be able to determine which PIDs are associated with "flushable" payloads because the PAT and PMT may not have been received and decoded by the RAU or the User Unit. As such, a "Table of PID types" in the RAU is initially reset to "Other" for all PID values with the exceptions of PID values 0 and 1 which are reserved and defined by the MPEG standards. If the type of the packet cannot be

30

determined, the RAU will process the header of the packet and its payload will be flushed by the FIFO.

When the User Unit is provided with the PAT and PMT, then the lists of PIDs and its associated stream types are decoded and forwarded to the RAU to update its Table of PID types. Thus, in step 430, the PID value is compared with the Table of PID types to determine its type with respect to four broad classifications.

If the method 400 determines from the PID value that a packet is a control stream packet in step 430, then the packet is sent to step 440 where the entire payload is read into the memory of the RAU and then sent to the User Unit via the dual-port RAM interface in step 480. The control information will be collected and displayed in various manners by the User Unit for evaluating the transport stream.

If the method 400 determines from the PID value that a packet is a video packet with adaptation data in step 430, then the packet is sent to step 450 where the adaptation header is read into the memory of the RAU. Video packets with adaptation fields are processed by reading the adaptation field where the first byte will define how many bytes must be read to obtain the adaptation data. Adaptation data is important because it often contains time base information such as Program Clock Reference (PCR). Again, the adaptation data is forwarded to the User Unit for evaluating the transport stream, but the remaining portion of the packet is then flushed from the FIFO and a counter for tracking the reception of this particular PID is also incremented to record its reception in step 490.

If the method 400 determines from the PID value that a packet is a video packet with no adaptation data in step 430, then the packet is sent to step 460 where the entire payload is flushed from the FIFO. Since such video packet often contains scrambled video data, the payload does not contain information that is relevant in evaluating the transport stream. As such, a counter for tracking the reception of this particular PID is also incremented to record its reception in step 460.

If the method 400 determines from the PID value that a packet belongs to "other data type" in step 430, then the packet is sent to step 470 where the entire payload is again flushed from the FIFO. Since these other data type packets often contain scrambled data or non-control-type

data, the information contained in these payloads cannot be used to evaluate the transport stream. As such, a counter for tracking the reception of this particular PID is also incremented to record its reception in step 470.

5 In step 495, the method 400 determines whether there are additional packets in the FIFO. If the decision is affirmatively answered, method 400 returns to step 420 and the steps of 430 through 490 are repeated for the next packet. If the decision is negatively answered, method 400 ends in step 497.

10 In this manner, method 400 is able to minimize the number of FIFO read cycles that the RAU must perform. During the periods of time that the FIFO is being flushed, the RAU is able to utilize the conserved CPU cycles to pass or receive messages to and from the User Unit. Using the flush method 400, slightly more than 4 out of every 188 byte packets are
15 read on average, thereby reducing the read cycle to approximately one read for every 188 instruction cycles (i.e., for a transport stream that is arriving at a rate of 5 Mega-bytes per second and a processing unit that is operating at 20 Mega-instructions per second).

 FIG. 5 illustrates a method 500 for verifying the consistency of time
20 base information in a transport stream. A transport stream may have a number of programs with each having a separate time base "clock". This creates the problem of measuring and verifying the consistency of the clocks on each of the programs. Each PID may contain adaptation sections for carrying PCR values, which are different from those of other
25 PIDs. The MPEG standards require PCRs to advance with limited amount of jitter, since time base information is used by the decoder to maintain synchronization with the encoder. The time base information must be very accurate down to 500 nano-seconds.

 The packet analyzer 200 is able to determine the correctness of PCRs
30 by using entries in the PID data records to keep track of PCR related data. The PID data records are a 8192 entry table, in which each entry contains information about that PID. As discussed in FIG. 4, the Table of PID types may contain other information aside from the PID type of a packet. Other important entries include information about the arrival time of the PCRs
35 which can be stored in a queue, the count of how many packets of a

particular PID type have been received, including a total count of the number of packets received regardless of the PID number, and the received PCR values. Alternatively, some entries could be assigned from a pool of memory while the table could simply contain a pointer into the
5 memory pool.

Referring to FIG. 5, the method 500 begins at step 510 and proceeds to step 520, where the method computes the number of packets received since the last PCR value. This net number of packets is then multiplied in step 530 with the number of bits per packet to produce a net number of bits.

10 In step 540, the method computes the difference between the current PCR value with the last PCR value. This net PCR time is then multiplied in step 550 with the bitrate in units of bits per PCR tick to produce a net number of bits.

In step 560, the method 500 computes the transport rate jitter by
15 obtaining a discrepancy between net number of bits calculated in steps 530 and 550. The method then ends in step 570.

In effect, method 500 compares two values: the number of packets (and therefore the number of bits) since the last PCR, and the product of the bitrate times the difference between the last PCR and the current PCR.
20 If the time base information is correct, then the product should match the number of bits, where the discrepancy is the jitter or error. Once the transport rate jitter is calculated, the jitter can be compared with a limit, added into an average, included into a histogram data file or sent to the User Unit for storage or display.

25 FIG. 6 illustrates a second method 600 for verifying the consistency of time base information in a transport stream. In this second method, a detector circuit is employed to record when a certain part of an incoming packet is present in the input to the FIFO 220. To illustrate, the detector should be able to detect the arrival of the 11th byte of a packet, which
30 corresponds to the least significant bit of the `program_clock_reference_base` field. The detector should then simultaneously cause a counter value to be stored in a queue of Recorded PCR Values. The counter is clocked at a constant rate such that its value indicates time. In this fashion, it is not necessary to read the PCR values, but instead, the
35 method places a "timestamp" associated with the packet in a queue.

Referring to FIG. 6, the method 600 begins at step 610 and proceeds to step 620, where the method computes the number of packets received since reception of the last PCR value. This net number of packets is then multiplied in step 630 with the number of bits per packet to produce a net
5 number of bits.

In step 640, the method computes the difference between the current recorded PCR value with the last recorded PCR value. This net recorded PCR time is then multiplied in step 650 with the bitrate in units of bits per PCR tick to produce a net number of bits.

10 In step 660, the method 600 computes the transport rate jitter by obtaining a discrepancy between net number of bits calculated in steps 630 and 650. The method then ends in step 670.

In essence, method 500 is very similar to method 600, where the only difference is that method 500 uses the PCR values while method 600 uses
15 the timestamps when the PCRs were actually received.

FIG. 7 illustrates a third method 700 for verifying the consistency of time base information in a transport stream. Referring to FIG. 7, the method 700 begins at step 710 and proceeds to step 720, where the method computes the difference between the current recorded PCR value with the
20 last recorded PCR value. In step 730, the method computes the difference between the current PCR value with the last PCR value.

In step 740, the method 700 computes the transport rate jitter by obtaining a discrepancy between the time calculated in steps 720 and 730. The method then ends in step 750.

25 The present invention also provides a plurality of methods in viewing the details of a transport packet stream. The displayed information are grouped in a manner to easily alert an operator as to potential problems in a packet stream or a network of packet streams.

FIG. 8 illustrates a graphical interface display 800 showing a
30 program guide view. The interface display includes a display area 810 where "Guide PIDs" are compared with "Active PIDs". The Guide PIDs indicate the types of PIDs to be expected, while Active PIDs indicate the types of PIDs that are actually received. This particular arrangement of PID information provides a quick reference to match the two sets of PID

data. If the transport stream is functioning properly, the two rows of data should contain identical numbers.

Specifically, display area 820 illustrates a missing PID value in the Guide PIDs row, while a PID value of 0006 is illustrated in the Active PIDs row. This indicates to an operator that a particular PID was not expected but yet it was received. In contrast, display area 830 illustrates missing PIDs in the Active PIDs row, while the PID values of 0060 and 0061 are illustrated in the Guide PIDs row. This indicates to an operator that a particular set of PIDs were expected but have yet to be received. Thus, graphical interface display 800 provides a quick and efficient method of monitoring the program guide.

FIG. 9 illustrates a graphical interface display 900 showing a video rates view. The interface display includes a display area 910 showing the percentage that is not nulled. Namely, this particular arrangement provides a quick reference as to bandwidth utilization. In this particular display, the operator can quickly gauge the bandwidth utilization to be 99.9%.

FIG. 10 illustrates a graphical interface display 1000 showing a video rates graph. The interface display includes a display area 1010 showing a plurality of "inch worm" graphs 1020. These inch worm graphs illustrate the minimum and maximum data rate for a plurality of different PIDs. Furthermore, an average line 1030 is provided for each inch worm graph to illustrate the average data rate for a specific PID. This particular arrangement provides effective method of showing real-time information regarding data rate. One benefit of this graphical interface display is the ability to evaluate whether PIDs are being sent evenly. In essence, an operator can easily gain some insight into the PID scheduling scheme, i.e., a method of evaluating the multiplexing algorithm of the system.

FIG. 11 illustrates a graphical interface display 1100 showing a multiplexed streams scan. The interface display includes a display area 1120 for showing the states of a plurality of multiplexed streams. Each stream being shown as a block. A display area 1110 shows a plurality of status patterns. These patterns may incorporate different colors or

symbols within the block. A display area 1130 shows a record of detected problems for a selected multiplexed stream 1140.

There has thus been shown and described a novel method and apparatus for evaluating transport packet streams in "real time" for
5 detecting errors, verifying the consistency of time base information and displaying important packet stream information. Many changes, modifications, variations and other uses and applications of the subject invention will, however, become apparent to those skilled in the art after
10 considering this specification and the accompanying drawings which disclose the embodiments thereof. All such changes, modifications, variations and other uses and applications which do not depart from the spirit and scope of the invention are deemed to be covered by the invention, which is to be limited only by the claims which follow.

What is claimed is:

- 1) Apparatus (200) for evaluating in real-time a packet stream having a plurality of packets, said apparatus comprising:
 - 5 a buffer (220) for receiving the plurality of packets; and
 - a first analysis unit (230), coupled to said buffer, for analyzing said plurality of packets stored within said buffer in real-time.

- 2) The apparatus of claim 1, wherein said buffer (220) includes a flag
10 bit for storing packet status information.

- 3) The apparatus of claim 1, further comprises:
 - a flushing circuit (270), coupled to said buffer (220), for removing
unwanted packets from said buffer.

- 15 4) The apparatus of claim 1, further comprises:
 - a storage medium (232) (234), coupled to said first analysis unit
(230), for storing a plurality of time base information for said plurality of
packets for detecting transport rate jitter.

- 20 5) The apparatus of claim 1, further comprises:
 - a second analysis unit (240), coupled to said first analysis unit (230);
and
 - a monitor (260), coupled to said second analysis unit, where said
25 second analysis unit interfaces with said first analysis unit to display a
packet stream analysis on said monitor.

- 6) Method for evaluating in real-time a packet stream having a
plurality of packets, where each packet having a plurality of bytes, said
30 method comprising the step of:
 - a) receiving the plurality of bytes into a buffer;
 - b) assigning a flag to each byte to indicate "start of a new packet",
"data valid" indicator or "short packet"; and
 - c) reading said bytes from said buffer to detect packet framing
35 errors.

- 7) Method for evaluating in real-time a packet stream having a plurality of packets with each packet having a packet identifier and a plurality of bytes, said method comprising the steps of:
- 5 a) receiving the plurality of bytes into a buffer;
b) determining which bytes in said buffer carry scrambled data or non-control-type data; and
c) flushing said bytes that carry scrambled data or non-control-type data from said buffer.
- 10
- 8) Method for evaluating in real-time a packet stream having a plurality of packets, said method comprising the steps of:
- a) receiving the plurality of packets into a buffer;
b) extracting a plurality of time base information from said plurality
15 of packets; and
c) comparing said time base information to detect transport rate jitter.
- 9) The method of claim 8, wherein said extracting step (b) comprises
20 the steps of:
- b1) computing a difference between a current recorded program clock reference (PCR) value and a last recorded PCR value; and
b2) computing a difference between a current PCR value and a last PCR value.
- 25
- 10) In a system for evaluating in real-time a packet stream having a plurality of packets, a display (800) (900) (1000) (1100) comprising a plurality of regions for displaying a plurality of results from a packet stream analysis.

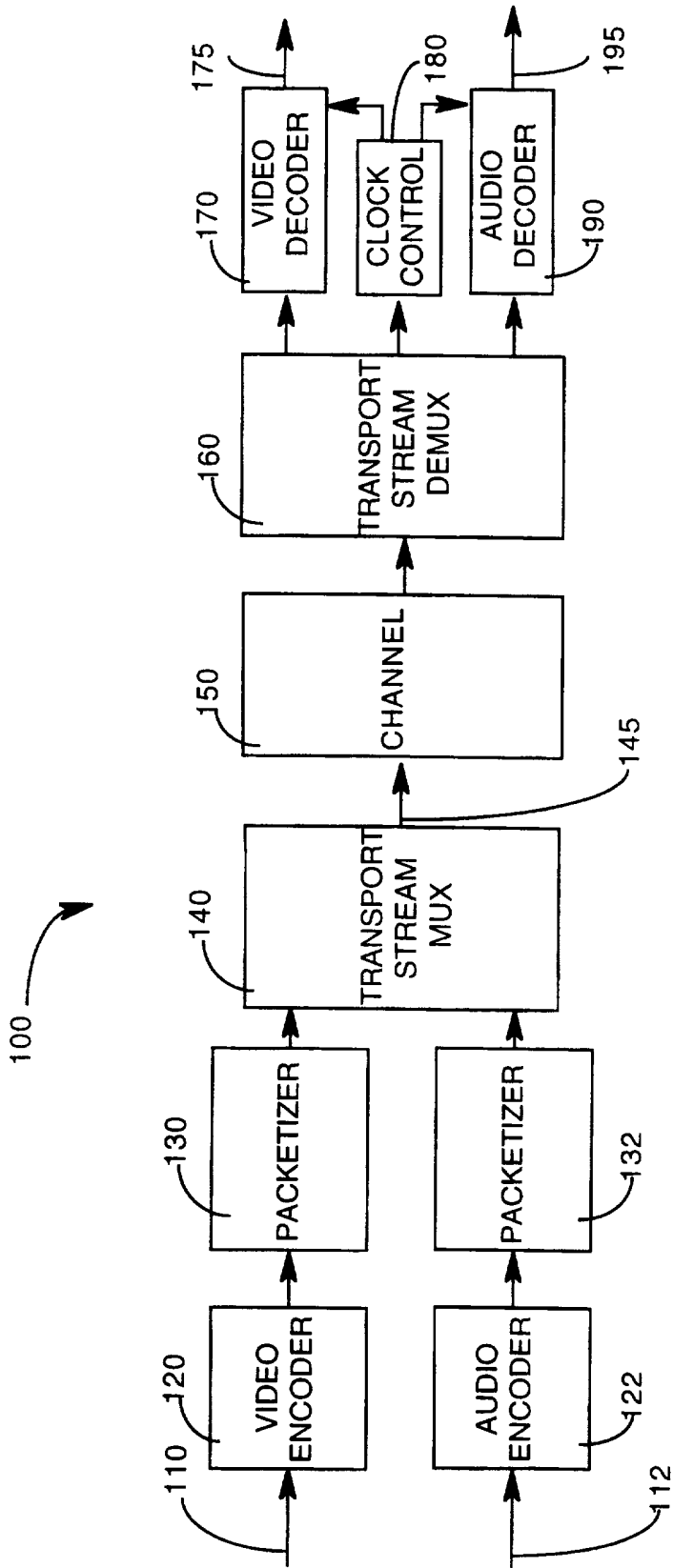


FIG. 1
(PRIOR ART)

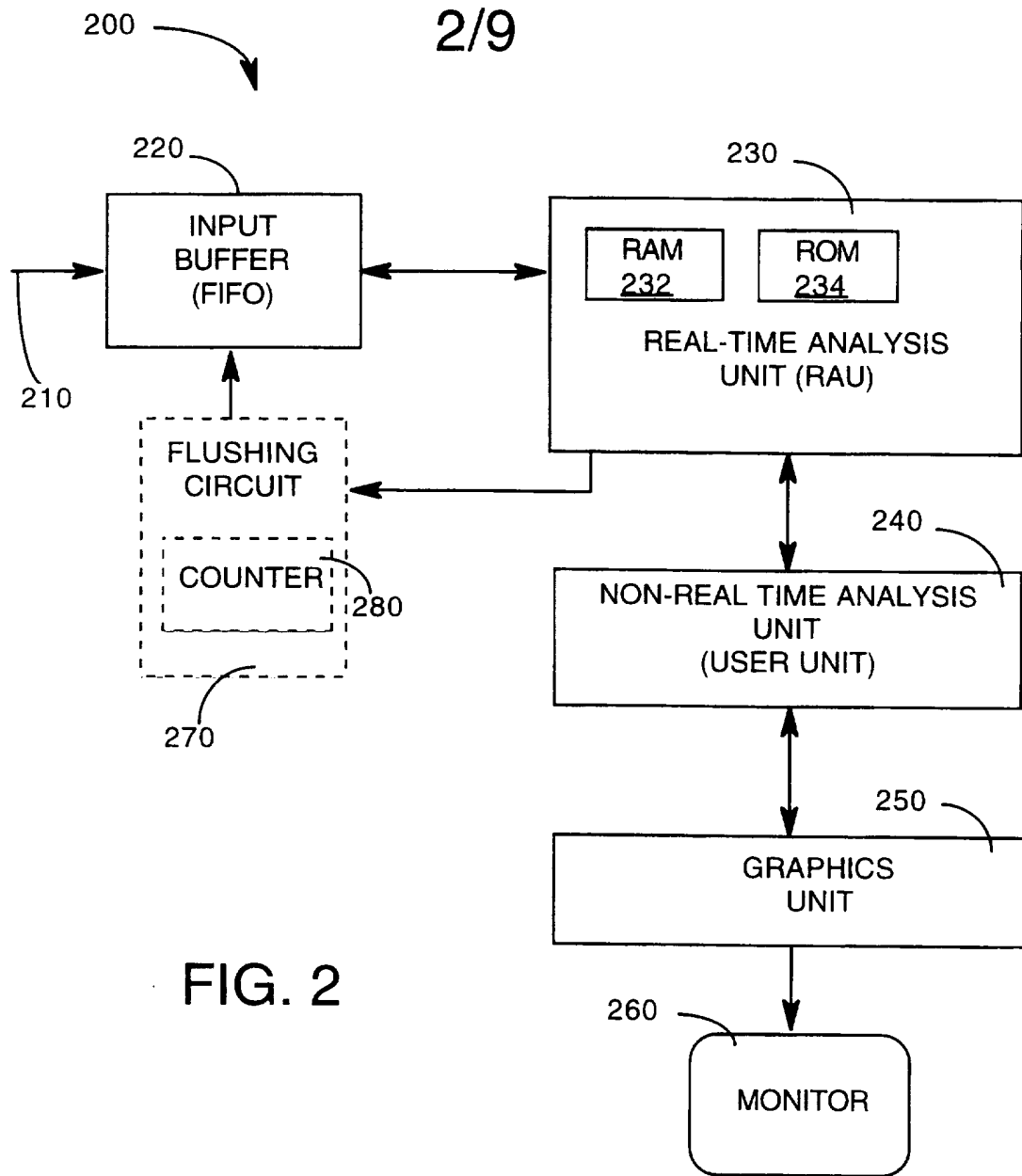


FIG. 2

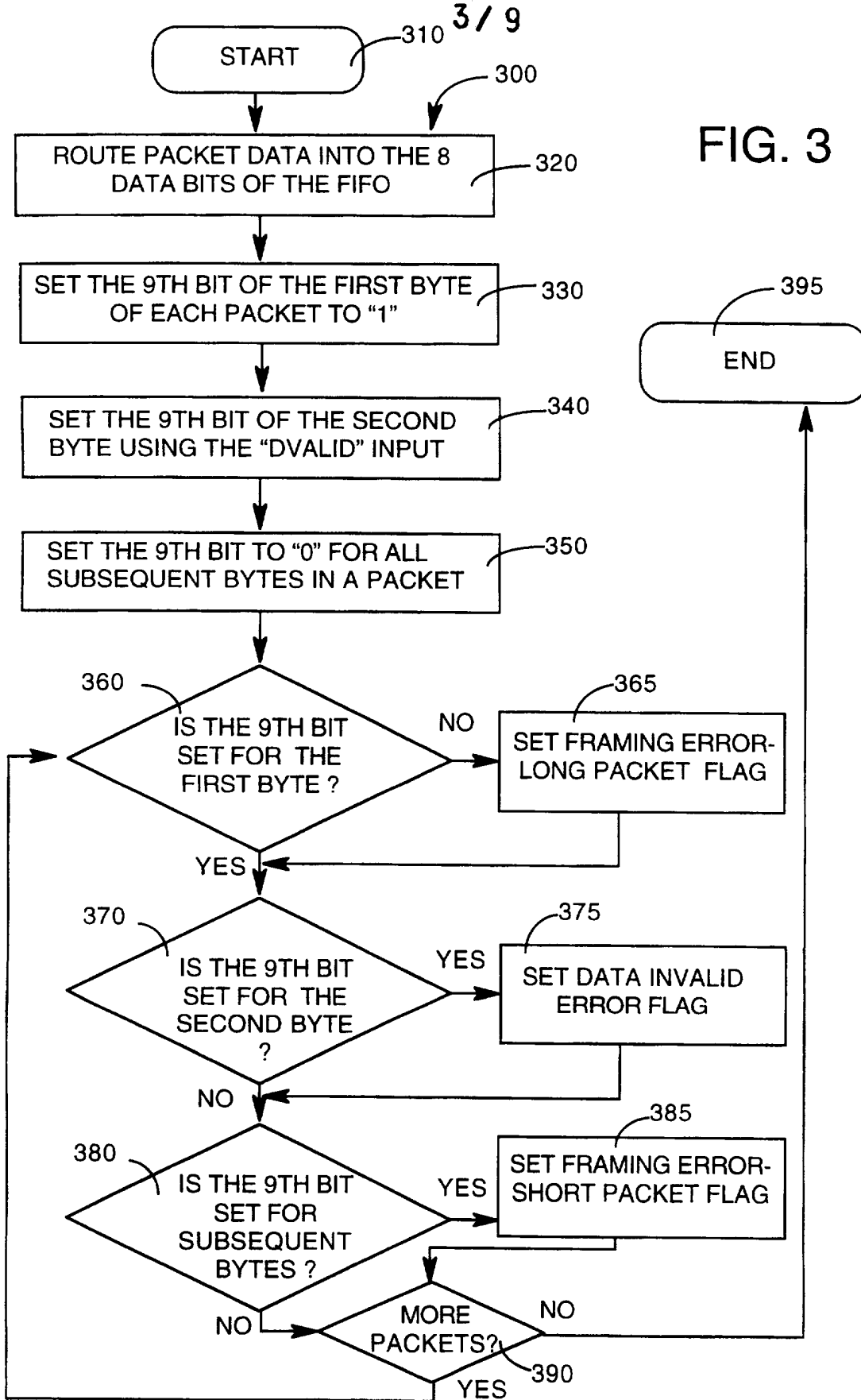


FIG. 3

4/9

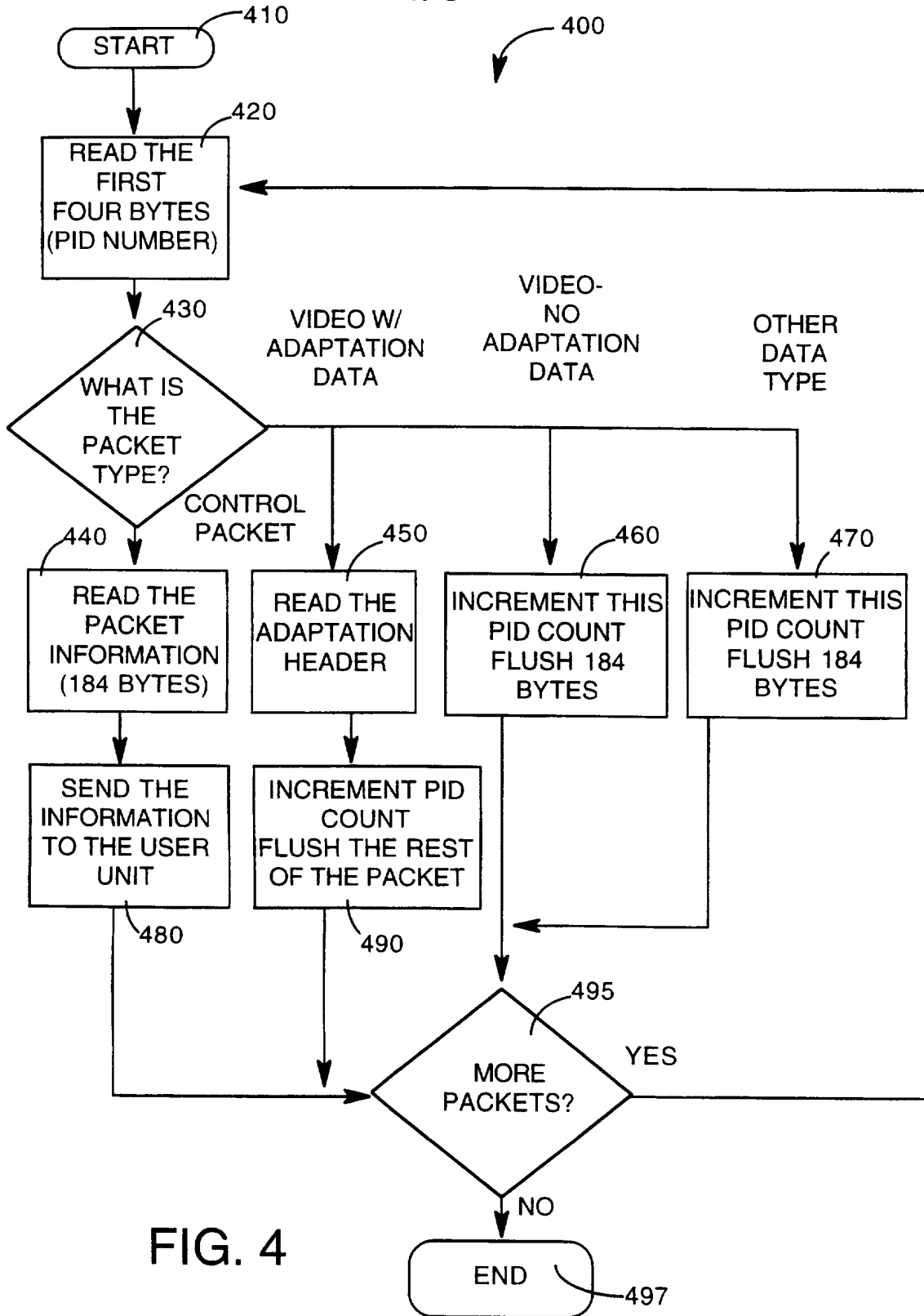


FIG. 4

5/9

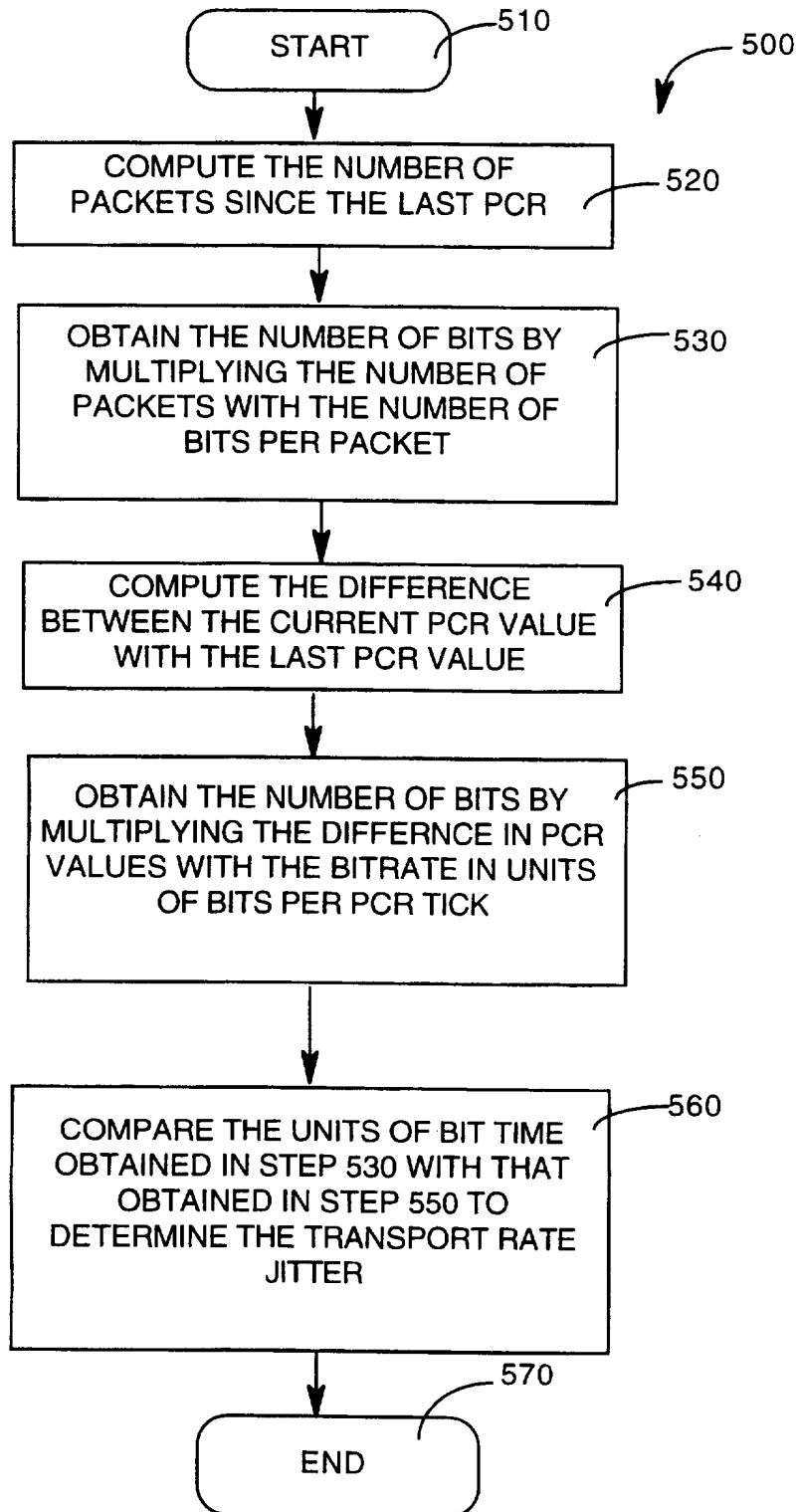


FIG. 5

6/9

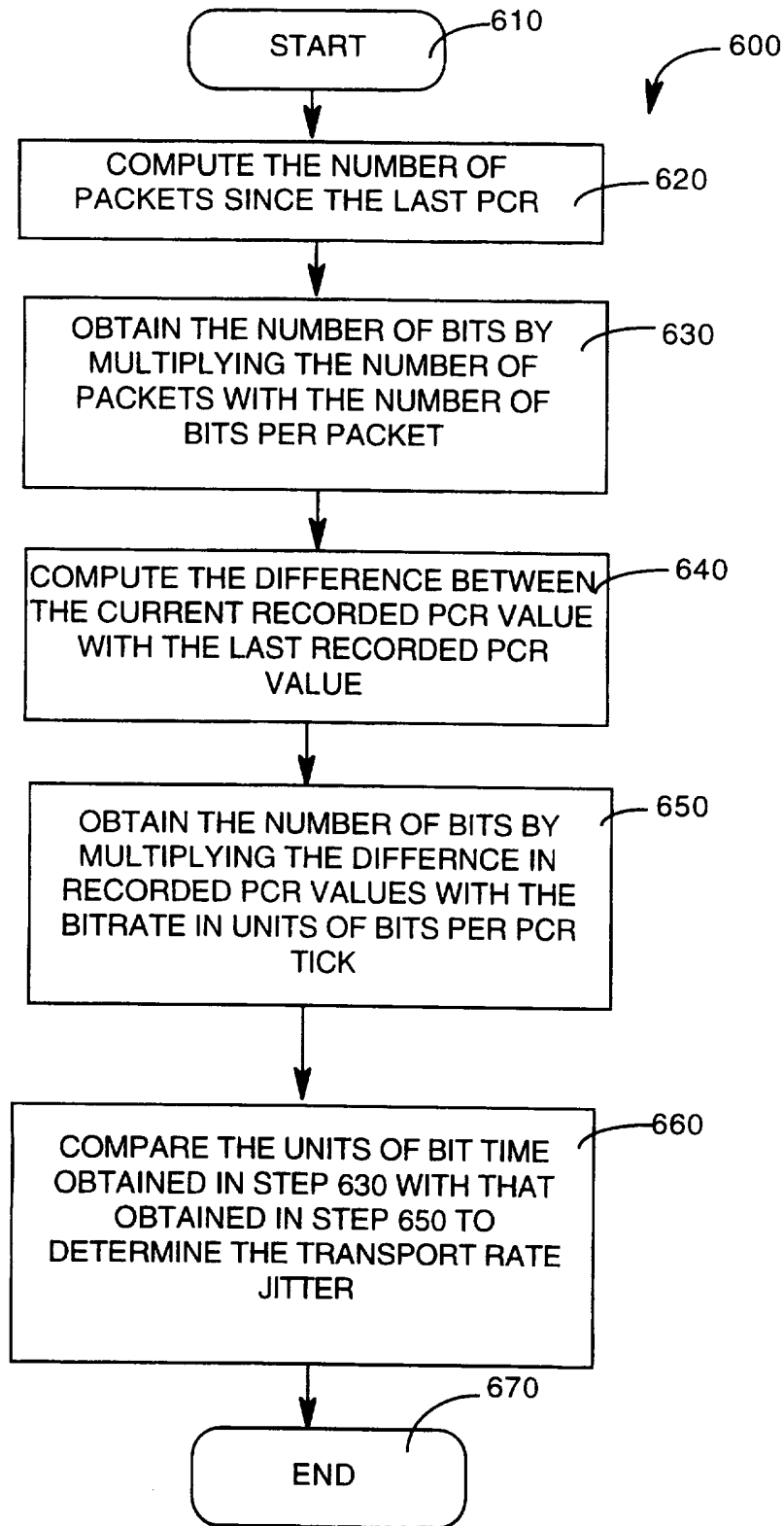


FIG. 6

7/9

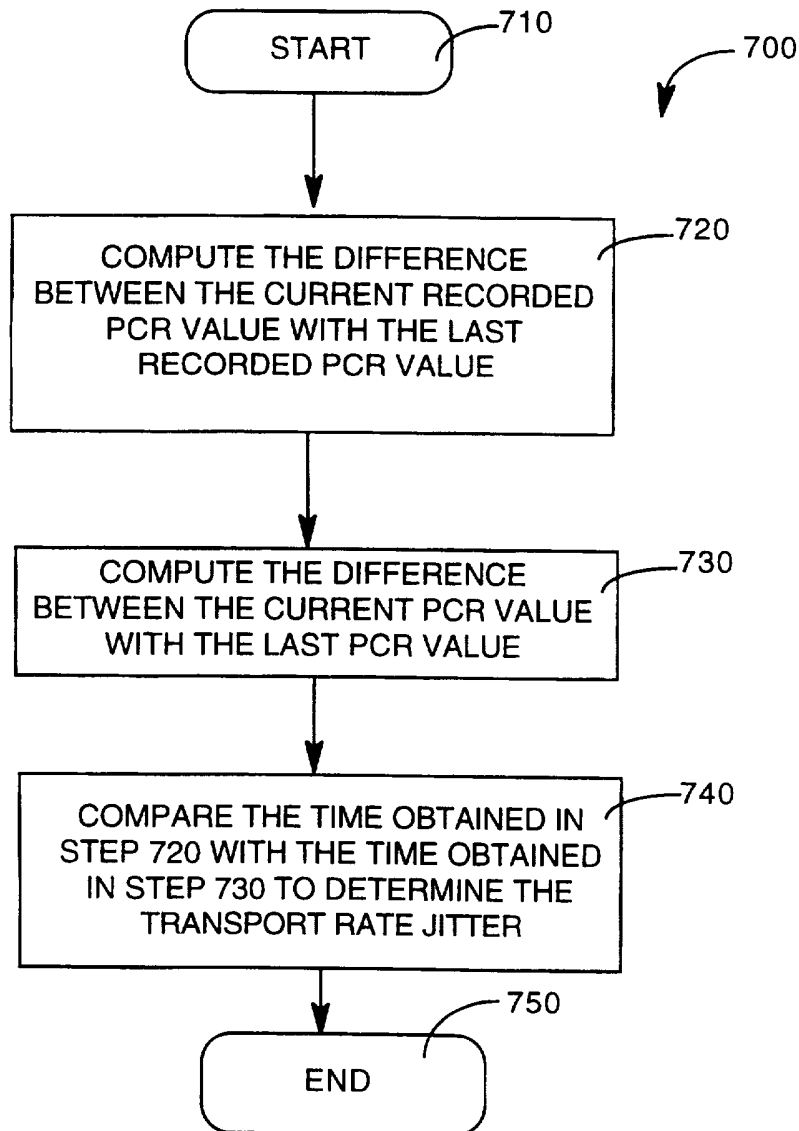


FIG. 7

8/9

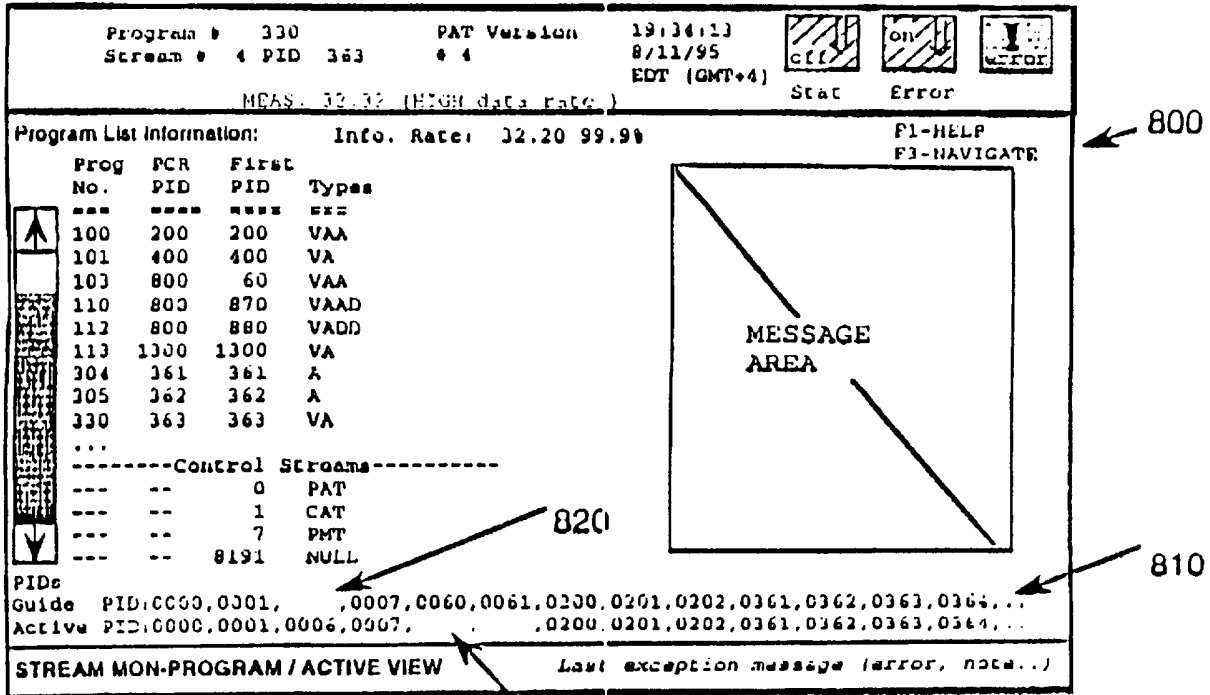


FIG. 8

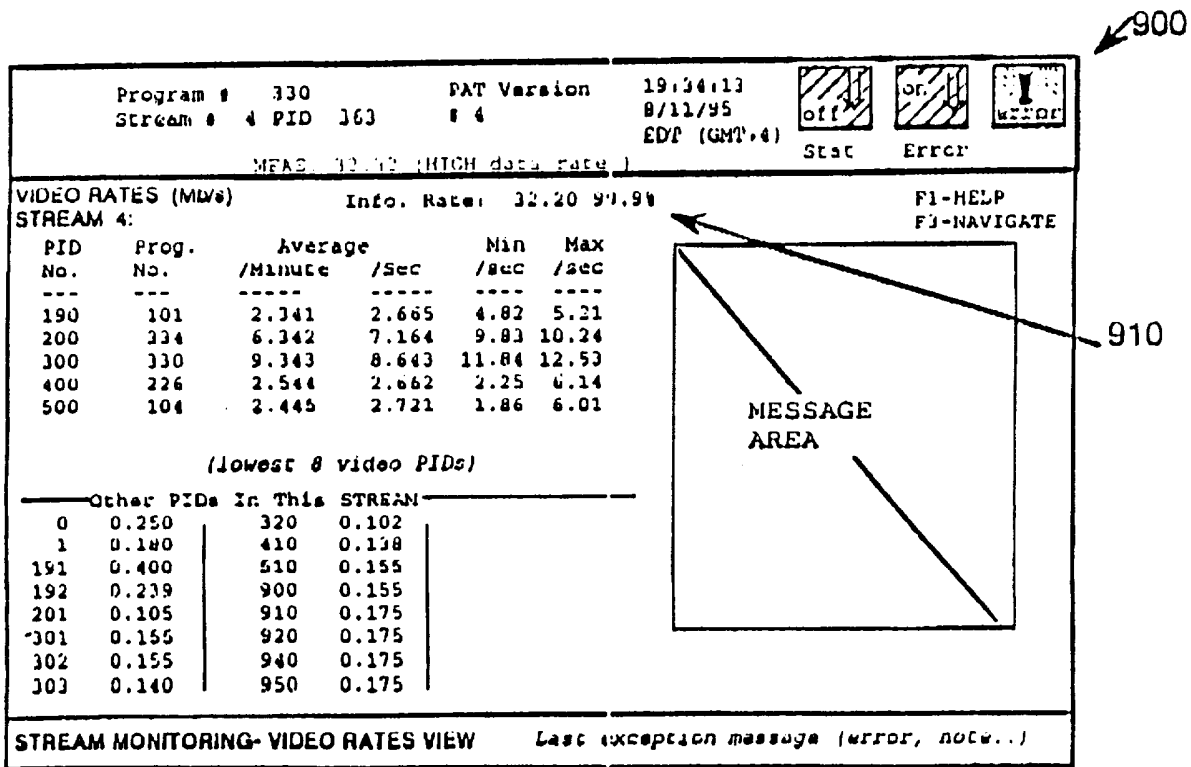


FIG. 9

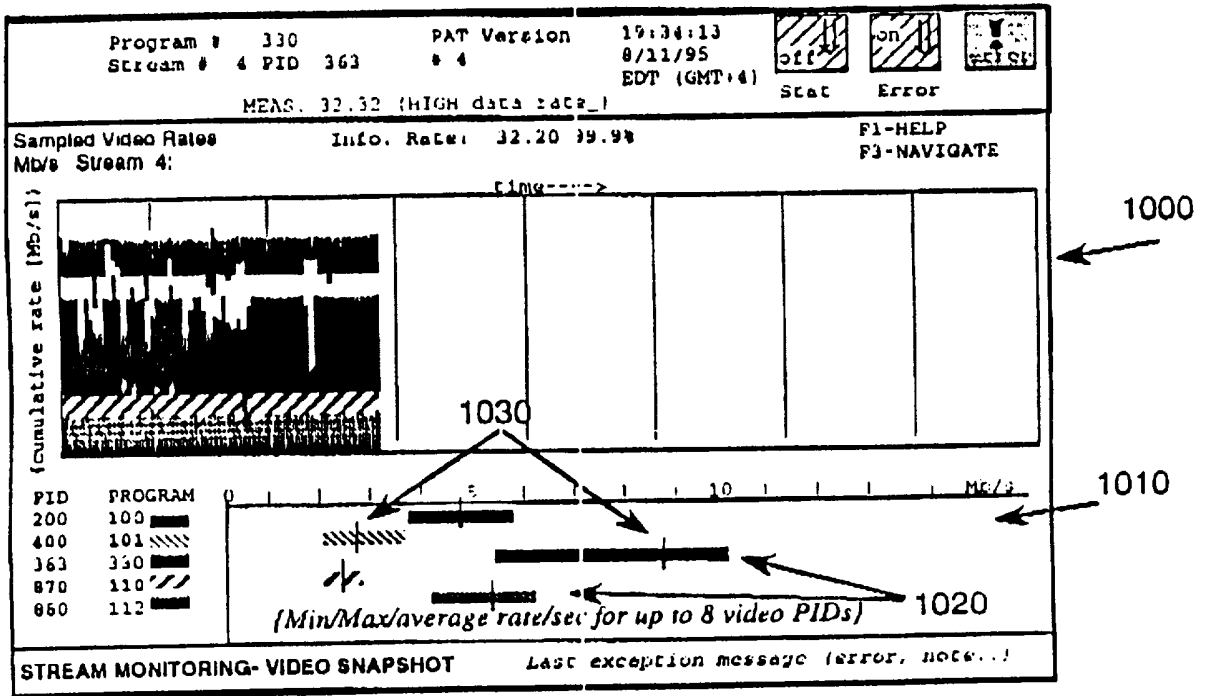


FIG. 10

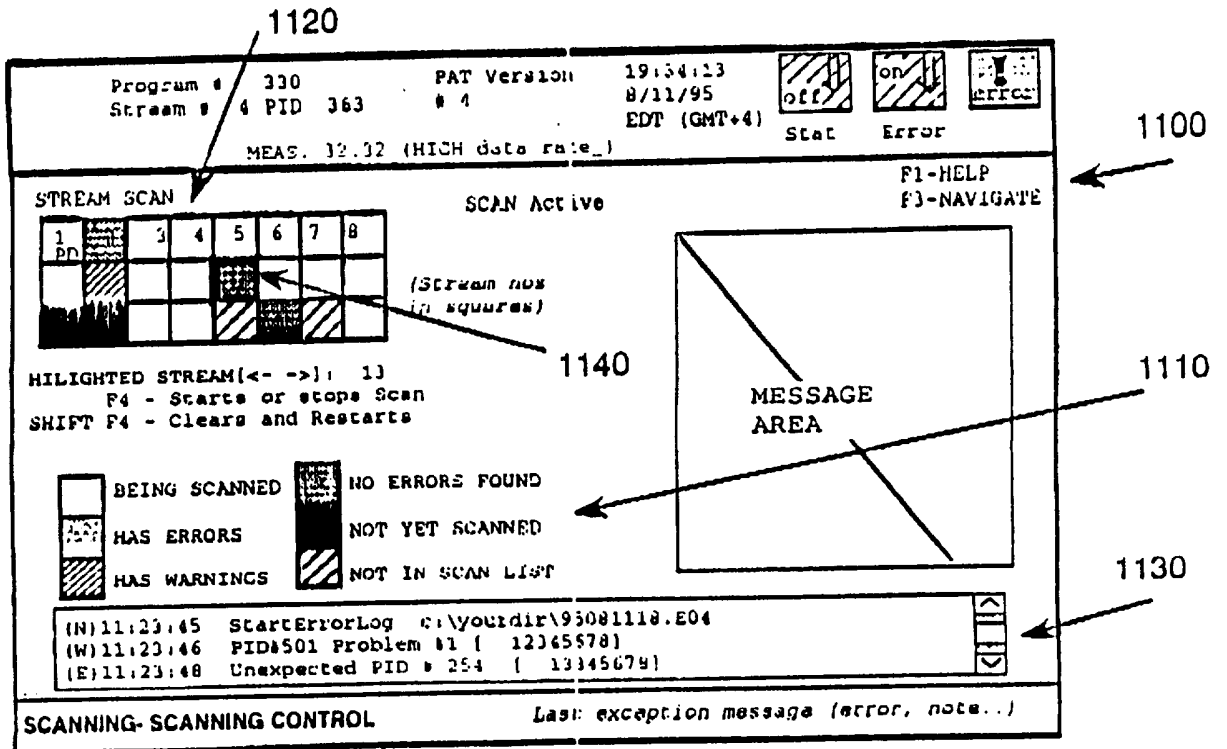


FIG. 11

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US97/04009

A. CLASSIFICATION OF SUBJECT MATTER

IPC(6) :H04J 3/14, 3/24; H04Q 11/04
US CL :370/241, 250, 252, 253, 400, 409, 463

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)

U.S. : 370/241, 250, 252, 253, 400, 409, 463

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

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

APS
search terms: packet, analyzer, status, framing error, display, buffer

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 4,775,973 A (TOMBERLIN et al) 04 October 1988, see entire document.	1-10
Y	US 5,212,724 A (NAZARENKO et al) 18 May 1993, col. 12, lines 39-45 and col. 28, lines 1-16.	1-10
Y	US 4,887,260 A (CARDEN et al) 12 December 1989, col. 1, line 60 to col. 2, line 11, col. 3, lines 20-41, col. 4, lines 63-68, and Fig. 2.	1-10

Further documents are listed in the continuation of Box C. See patent family annex.

* Special categories of cited documents:	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A" document defining the general state of the art which is not considered to be part of particular relevance	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E" earlier document published on or after the international filing date	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"Z" document member of the same patent family
"O" document referring to an oral disclosure, use, exhibition or other means	
"P" document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search

02 MAY 1997

Date of mailing of the international search report

09 JUN 1997

Name and mailing address of the ISA/US
Commissioner of Patents and Trademarks
Box PCT
Washington, D.C. 20231

Facsimile No. (703) 305-3230

Authorized officer

Douglas Olms
DOUGLAS OLMS

Telephone No. (703) 305-4703