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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
5 February 2009 (05.02.2009)

PCT

(10) International Publication Number
WO 2009/018043 A1

(51) International Patent Classification:
H04N 5/00 (2006.01) *H04N 7/173* (2006.01)

(21) International Application Number:
PCT/US2008/070853

(22) International Filing Date: 23 July 2008 (23.07.2008)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/829,255 27 July 2007 (27.07.2007) US

(71) Applicant *(for all designated States except US)*: **SCIENTIFIC-ATLANTA, INC.** [US/US]; 5030 Sugarloaf Parkway, Lawrenceville, Georgia 30044 (US).

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: **VERSTEEG, William, C.** [US/US]; 330 Guildhall Grove, Alpharetta, Georgia 30022 (US). **SCHLARB, John, M.** [US/US]; 4030 Boles Creek Drive, Duluth, Georgia 30096 (US). **MEADOR, Guy, Atkinson, III** [US/US]; 1852 Riverton Drive, Suwanee, Georgia 30024 (US).

(74) Agents: **BARNHARDT, Hubert, J., III** et al.; Scientific-Atlanta, Inc., Intellectual Property Dept., 5030 Sugarloaf Parkway, Lawrenceville, Georgia 30044 (US).

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

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*

(54) Title: SYSTEMS AND METHODS OF DIFFERENTIATED CHANNEL CHANGE BEHAVIOR

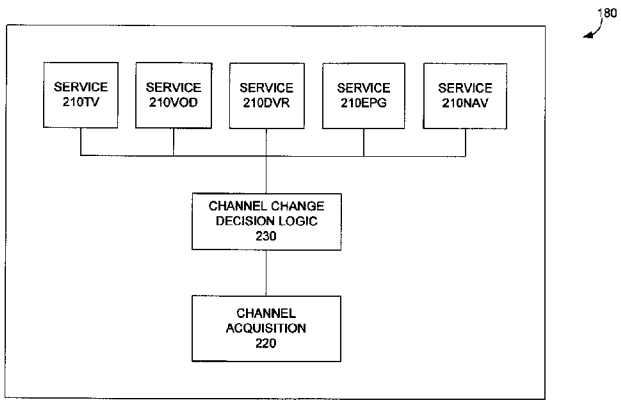


FIG. 2

(57) Abstract: Methods and systems of handling channel changes in a digital home communication terminal are disclosed. An exemplary method comprises: receiving a channel change request; determining whether the channel change request is user-initiated; setting at least one channel change parameter to a value based on the determination; and performing a channel change using the at least channel change parameter having the set value. An exemplary system comprises: a channel acquisition component, memory, and a processor. The processor is programmed by code, stored in memory, to: receive a channel change request; determine whether the channel change request is user-initiated; and instruct the channel acquisition component to perform a channel change using at least channel change parameter. The channel change parameter has a value that is based on the determination.

WO 2009/018043 A1

SYSTEMS AND METHODS OF DIFFERENTIATED CHANNEL CHANGE BEHAVIOR

CROSS REFERENCE TO RELATED APPLICATIONS

This application is related to the U.S. Patent Application entitled "Systems and Methods of Differentiated Requests for Network Access" having docket number A-11727 and filed on the same day as the present application, which is entirely incorporated by reference herein.

FIELD OF THE DISCLOSURE

The present disclosure relates to digital set-tops, and more specifically, to systems and methods of differentiated channel change behavior.

BACKGROUND

Many consumers receive entertainment programming in their homes from a cable television operator. Many of today's cable offerings are broadcast using digital signals, which make more efficient use of communication bandwidth, and thus allow more programming to be carried on the same cable. In these cable systems, video programming (e.g., television programs, movies, etc.) is encoded using a Motion Pictures Experts Group (MPEG) standard, and encapsulated into an MPEG transport stream. The MPEG transport stream may be transmitted from a cable head-end to the customer premises by a number of subscriber delivery mechanisms, such as coax cable or hybrid fiber-coax (HFC) cable, or over twisted pair cable running a broadband technology such as digital subscriber loop (DSL). At the customer premises, a digital home communication terminal (DHCT) decodes the programming and generates an analog or digital picture signal. The picture is displayed by a television connected to the DHCT.

Video programming is delivered to the DHCT over a logical channel. Logical channels are also used to deliver other kinds of data, some of which is seen by the user (e.g., an electronic program guide), and some of which is not (e.g., system data such as

entitlements and channel maps.) In some environments, a logical channel corresponds to a particular frequency. In other environments, a logical channel corresponds to a timeslot within a particular frequency. In other environments, a logical channel corresponds to a network unicast or broadcast address.

A DHCT, or group of DHCTs sharing the same subscriber connection, can receive multiple types of programs or data simultaneously, since each is carried in a separate logical channel. The set of logical channels received over the subscriber connection typically changes over time. Sometimes the change in the channels delivered to a subscriber is a result of direct channel changes by users. Other times, the channel change is indirect, resulting from requests made by the DHCT for various types of data (as described above) which are also carried over logical channels. Conventional DHCTs treat all types of channel changes in the same manner. However, it would be advantageous for different types of channel changes to be handled differently by the DHCT. Thus, a need arises for these and other problems to be addressed.

BRIEF DESCRIPTION OF THE DRAWINGS

Many aspects of the disclosure can be better understood with reference to the following drawings. The components in the drawings are not necessarily to scale, emphasis instead being placed upon clearly illustrating the principles of the present disclosure.

FIG. 1 is a block diagram of an environment in which one embodiment of a system and method of differentiated channel change behavior is located.

FIG. 2 is a block diagram showing selected software components in one embodiment of the DHCT in FIG. 1.

FIG. 3 is an object diagram of selected software components present in one embodiment of the DHCT in FIG. 1.

FIG. 4 is a flowchart showing a decision process implemented by one embodiment of the channel change logic in FIG. 1.

FIG. 5 is a flowchart showing further details of block 455 in FIG. 4.

FIG. 6 is a flowchart of a process used by another embodiment of the channel change decision logic in FIG. 2.

FIG. 7 is a block diagram showing selected components of the DHCT in FIG. 1.

DETAILED DESCRIPTION

FIG. 1 is a block diagram of an environment 100 in which one embodiment of a system and method of differentiated channel change behavior is located. A core network adaptation device 110 receives one or more digital media streams 115 for delivery to various subscribers. In this disclosure, the term "media stream" refers to a stream that includes video frames, audio frames, hypermedia, multimedia, or any combination thereof. Common encoding formats for media streams 115 include MPEG-2, MPEG-4, and VC-1. In some environments, the encoded media stream represents a single program, and thus contains a video and an audio stream multiplexed together into a single program transport stream (SPTS).

Media streams 115 may be provided from various sources. In the example environment of FIG. 1, media stream 115E is provided by an encoder 120 which encodes an analog signal from a media content source, such as a cable network or an on-air television station, and media stream 115S is provided from a digital media content server 130. Other ways of providing media streams 115 to core network adaptation device 110 should be familiar to a person of ordinary skill in the art, and are intended to be within the scope of this disclosure.

Core network adaptation device 110 prepares media streams 115 for transport over a core network 140. Though the details of this adaptation depend on the type of core network, the adaptation generally involves encapsulating media streams 115 into packets, using broadcast addressing for the packets, and combining packet program streams. The result is a stream 145 suitable for transport across core network 140 that contains multiple media streams 115, each containing a program.

A person of ordinary skill in the art should be familiar with the concept and practice of encapsulating information into packets, and with broadcast addressing techniques, so these features will not be discussed further in this disclosure. In one embodiment, MPEG Transport Stream (TS) packets are encapsulated within layer-3 Internet Protocol (IP) packets. In another embodiment, the MPEG TS packets are encapsulated within real-time transport (RTP) packets, which are in turn encapsulated within IP packets. In another embodiment, VC-1 streams are used rather than MPEG streams.

Multiple programs carried within the stream 145, destined for many different subscribers, are transported over core network 140, and delivered to switches 150 located at the network edge. Each switch 150 selects, for a particular subscriber, a subset of programs carried in stream 145, and produces a stream 155 that is addressed to that subscriber. Stream 155 is delivered, via a subscriber connection 165, to those subscribers connected to switch 150. In some embodiments the stream 155 uses multicast addresses, while other embodiments unicast addresses are used. A person of ordinary skill in the art should be familiar with the use of multicast and unicast addresses to deliver packets to groups of subscribers and single subscribers, respectively.

Stream 155 is provided to an access network adaptation device 170, which prepares stream 155 for travel over subscriber connection 165. The details of the

adaptation vary depending on the type of subscriber connection 165 and on the subscriber equipment. In general, access network adaptation device 170 converts between the lower layer protocols used by switch 150 (e.g., high speed Ethernet) and the protocols used on subscriber connection 165 (e.g., DSL, HFC, Fiber to the Home). Thus, access network adaptation device 170 may be implemented as a cable modem termination system (CMTS) in an HFC environment, or a DSL access multiplexer (DSLAM) in a DSL environment. Some embodiments of access network adaptation device 170 also perform encapsulation, de-encapsulation, or both (as when converting from one packet format to another). Some embodiments act as a multiplexer to combine additional streams such as a voice stream or a data stream.

After adaptation by access network adaptation device 170, stream 155 is transmitted over one of the subscriber connections 155 to a digital home communication terminal (DHCT) 180. DHCT 180 receives stream 155, and decodes the individual program streams carried within to produce a video signal. DHCT 180 supplies the video signal to a display (not shown) for viewing by the customer. In one embodiment, the display is a television. In another embodiment, the display is a computer monitor. In some embodiments, DHCT 180 also decodes an audio stream and produces audio signal which accompanies the video signal.

As explained earlier, a subset of program streams is selected by switch 150 for delivery to a particular subscriber location. Each of these program streams can be viewed as being carried on a different logical channel. DHCT 180 communicates with a channel change server 190 to request that particular program streams, or logical channels, be included in the stream 155 received by that subscriber. For example, DHCT 180, in response to a user request to watch the FOX network, requests from channel change server 190 a “channel change” to FOX. Channel change server 190 in turn requests the

switch 150 serving that subscriber to include the FOX program stream in stream 155 delivered to that DHCT 180 over the subscriber connection 165.

In some embodiments, channel change server 190 and/or switch 150 fulfill channel change requests by joining and leaving particular multicast groups, each of which carries a set of logical channels. A channel change request may also be fulfilled by switching stream 155 to a unicast stream (not shown) that is a cached version of media stream 115 but is instead is addressed to a particular DHCT 180. Use of a unicast rather than a multicast stream reduces the time it takes a DHCT 180 to receive and decode a new program source. The use of unicast streams to speed up a channel change is often referred to as "fast channel change" or "instant channel change."

FIG. 2 is a block diagram showing selected software components in one embodiment of DHCT 180. DHCT 180 includes components which provide services to the user. An exemplary list of such components includes, but is not limited to, live television 210TV, video-on-demand 210VOD, an electronic program guide 210EPG, and digital video recording 210DVR. A navigator 210NAV allows a user to navigate among these services, and may also allow the user to access various settings of the DHCT 180 (e.g., volume, parental control, etc.) In one embodiment, these services 210 are implemented as user-mode applications, rather than as kernel-mode components.

DHCT 180 also includes a channel acquisition component 220. Channel acquisition component 220 is responsible for acquiring, or changing to, a logical channel. Channel acquisition component 220 may involve hardware and/or software components, and is typically platform-specific: in some embodiments, channel acquisition involves tuning to an RF channel; in other embodiments, channel acquisition involves requesting an IPTV stream from a multicast or unicast server such as channel change server 190 (FIG. 1). In some embodiments, DHCT 180 can receive multiple channels simultaneously. In some

embodiments, channel acquisition component 220 is implemented as kernel-mode code interacting with an RF tuner or a network interface.

Requests for channel acquisition, or channel change, originate from different sources and for different reasons. Some channel acquisition requests are initiated directly by a user. One example of such a user-initiated channel change is when a user watching live TV, where the user indicates to service 210TV a channel change, for example, through a remote control. In this case, service 210TV is the source of the channel change request. Other channel change requests are system-initiated rather than user-initiated. This may occur, for example, when service 210DVR selects a channel in order to record a scheduled program. In this case, service 210DVR originates the channel change request. Other channel change requests are system-initiated even though the request is an indirect result of a user selection. One example of this type of channel change request can occur when a user interacting with the service 210EPG asks to view detailed program description information. In this scenario, service 210EPG obtains new guide information over an EPG channel. Another example of a channel change request originating from service 210EPG occurs when service 210EPG periodically downloads updates to the guide information. These channel change scenarios are merely examples. Other channel change scenarios should be familiar to a person of ordinary skill in the art, and are intended to be within the scope of this disclosure.

The channel change requests are received by channel change decision logic 230, which instructs channel acquisition component 220 to actually perform the channel change after determining appropriate parameters for the channel acquisition call, based on the source of the request and/or the reason for the request. For example, channel change decision logic 230 might utilize a “fast channel change” mode if the request was user-initiated and the configuration of subscriber connection 165 and/or core network 140 was

such that a channel change would use bandwidth. Some embodiments of channel change decision logic 230 also take into account information about characteristics of the subscriber connection 165 and/or core network 140 when determining the appropriate parameters for a channel change request. Thus, channel change decision logic 230 acts as an intermediary between services and channel acquisition component 220 and insulates these services from network specifics.

A person of ordinary skill in the art should understand that software components referred to herein include executable code that is packaged, for example, as a standalone executable file, a library, a shared library, a loadable module, a driver, or an assembly, as well as interpreted code that is packaged, for example, as a class.

FIG. 3 is an object diagram of selected software components present in one embodiment of DHCT 180. Channel change decision logic 230 includes a function MapChannelRequest 310, which is called by an application in order to request a channel change. Function MapChannelRequest 310 has inputs ChannelId, Source, and ReasonCode. As its name suggests, the ChannelId parameter identifies the target channel. As described above, the Source parameter identifies the component that made the request, and ReasonCode identifies the reason for the request (which may also be viewed as the type of request). In the example of FIG. 3, Source and ReasonCode are distinct parameters, and are each enumerated values. However, a variety of identification mechanisms can be used, as would be understood by a person of ordinary skill in the art. For example, Source and ReasonCode may be combined into a single parameter.

Channel acquisition component 220 includes a data member Settings 320, and a function AcquireChannel 330. As its name suggests, AcquireChannel 330 is invoked to change to, or acquire a particular channel, and interacts with software and/or hardware components (e.g. channel acquisition component 220) to make this happen. AcquireChannel

330 uses the values in Settings 320 to control various aspects of the channel change behavior. In the example of FIG. 3, Settings 320 include ChannelId 320I, FastMode 320F, SurfSuppress 320S, ErrorCorrection 320E, Delay 320D, Resolution 320R, and DeliveryMode 320M. As will be described in more detail in connection with FIG. 4, the function MapChannelRequest 310 in channel change decision logic 230 sets the values of one or more parameters in Settings 320, based on the source of the request and/or the reason for the channel change request, before the call to AcquireChannel 330.

ChannelId 320I controls which channel is acquired. In some scenarios, MapChannelRequest 310 sets ChannelId 320I in Settings 320 to the same value as its parameter. In other scenarios (examples of which will be described later), MapChannelRequest 310 in channel change decision logic 230 may set ChannelId 320I in Settings 320 to a different value, so that the service invoking MapChannelRequest 310 gets a different result than what was requested. A person of ordinary skill in the art should be familiar with a variety of mechanisms for passing this information back to the service.

FastMode 320F controls whether "fast channel change" techniques are used by the server. (Such techniques were described briefly in connection with FIG. 1.) ErrorCorrection 320E controls the error correction (e.g., the type of FEC code, the depth of the FEC code, the use of selective retransmission, etc.) used by channel change server 190. DeliveryMode 320M controls the rate at which the newly acquired program stream will be provided to DHCT 180 in an environment where the delivery rate is variable, such as an IPTV environment. In the example of FIG. 3, enumerated values such as BurstMode, Normal, and TrickleMode are used for DeliveryMode 320M, but in other embodiments DeliveryMode 320M may specify a bitrate.

In some embodiments, these parameters are communicated to channel change server 190. In other embodiments, these parameters are used to select an appropriate server from

multiple channel change servers 190, based on the capabilities of the various servers 190. For example, different channel change servers 190 may vary in their error correction capabilities, or in their bitrate capabilities. FastMode 320F, ErrorCorrection 320E and DeliveryMode 320M are particularly useful in an environment which uses IP as a delivery mechanism, but a person of ordinary skill in the art will recognize that these settings have wider applicability as well.

Delay 320D controls the amount of delay between the request to channel acquisition component 220 and the actual channel change. For example, random amounts of delay may be used to insure that large numbers of DHCTs do not download guide data at midnight, or change channels on the hour or half-hour, either of which could overload equipment at the head-end. In the example of FIG. 3, enumerated values such as RandomLongPeriod, RandomShortPeriod, and Immediate are used, but in other embodiments Delay 320D may specify an actual amount of time. In some embodiments, the delay is implemented in DHCT 180, while in other embodiments Delay 320D is communicated to the channel change server 190 or to the tuner.

SurfSuppress 320S controls whether network signaling is suppressed when the user is changing from one channel to another in a relatively rapid fashion (i.e., "channel surfing"). Under such conditions, sending each and every channel change within the series of channel changes is wasteful of network bandwidth. That is, if the user moves quickly from channel 2 to channels 5, 8, and 11, it is unnecessary to send the requests for channels 5 and 8 to channel change server 190, since the user has moved on to another channel before the previous channel is delivered. A person of ordinary skill in the art should be aware of various techniques to detect this user "channel surfing" behavior. For example, a series of multiple keypresses indicating channel change can be treated as a single request. In one embodiment, SurfSuppress 320S is a boolean, where a value of True indicates that

signaling to the network (e.g., channel change requests) is suppressed during episodes of user channel surfing behavior. A value of False for this parameter indicates that network signaling is transmitted to the network as usual.

Resolution 320R controls the resolution of the program stream delivered on the requested channel. This feature is useful when multiple channels are available that carry the same programming but differ in their resolution (e.g., one program delivers in high definition, and another in standard definition). In such cases, channel change decision logic 230 may transform a request by an application (e.g., service 210DVR or service 210TV) for a logical channel with high resolution to a change to an analogous channel with lower resolution, or vice versa. In some embodiments, Resolution 320R also supports thumbnails, which are frames that are smaller than normal size.

As mentioned above, channel change decision logic 230 uses Source and/or ReasonCode to determine appropriate values for Settings 320. Channel change decision logic 230 may also use data provided by other components in making this decision. In the example of FIG. 3, service 210DVR tracks the amount of free space available for program recording, and makes this available as data member FreeSpaceAvailable 340. A Bandwidth Manager component 350 tracks the subscriber connection bandwidth that is in use by DHCT 180 and (in some embodiments) by other DHCTs also sharing the same subscriber connection 165. Bandwidth Manager 350 provides this information through data member SubscriberBandwidthAvailable 360.

A person of ordinary skill in the art should recognize the objects illustrated in FIG. 3 as abstractions chosen to illustrate how one embodiment of a system and method of differentiated channel change behavior partitions functionality among components. Such a person should also recognize that other divisions of functionality are also possible, and these other possibilities are intended to be within the scope of this disclosure.

Furthermore, although the objects in FIG. 3 may be described in terms of specific data structures such as lists, flags, pointers, and collections, a person of ordinary skill in the art should appreciate that other data structures providing similar functionality can be used instead. As just one example, a particular implementation might use an array instead of a linked list.

In general, the objects used by the systems and methods of differentiated channel change behavior are described herein in terms of code and data, rather than with reference to a particular hardware device executing that code. Furthermore, although the system and methods are described in object-oriented terms, there is no requirement that the systems and methods be implemented in an object-oriented language. Rather, one of ordinary skill in the art will understand that the systems and methods can be implemented in any programming language, and executed on any hardware platform.

As described above, channel change decision logic 230 encapsulates network-specific knowledge about the appropriate parameters to use in a channel change, while channel acquisition component 220 encapsulates platform-specific knowledge about how to effect a channel change. The decision process used by channel change decision logic 230 to choose appropriate values for these parameters will now be described in connection with the flowchart of FIG. 4.

The process 400 begins at block 410, where a channel change request is received. Next, block 415 examines the request to determine whether it was user-initiated. If the request was user-initiated, processing continues at block 420, which initializes Settings 320 as follows: FastMode 320F to True; ErrorCorrection 320E to Robust; and Delay 320D to Random. Processing then continues at block 425, which acquires the channel identified in the request using Settings 320 (e.g., through AcquireChannel 330). Process 400 is then complete.

If the request was not user-initiated, processing continues instead at block 430, which sets FastMode 320F to False. Next, at block 435 the source of the request is examined. If the request originated from service 210DVR, processing continues at block 440, where further examination occurs in order to appropriately set the values in Settings 320. However, if the request originated from a source *other* than service 210DVR, processing continues at block 425. As described earlier, at block 425 the channel identified in the request is acquired, using Settings 320, and process 400 is complete.

Returning now to block 440, which is executed if service 210DVR was the source of the channel change request, Delay 320D is set to Random and ErrorCorrection 320E is set to Robust. In one embodiment, Robust indicates a level of error correction which is more stringent compared to the level used when a user is the source of the channel change request. A person of ordinary skill in the art should appreciate that a variety of combinations of error-correcting behavior meet this condition. In one embodiment, "Robust" error correction includes the use of forward error-correcting codes (FEC) and/or selective retransmission.

Blocks 445 and 450 are optional. Block 445 gets the bandwidth available on subscriber connection 165. Block 450 determines the DVR storage space available for programming. Processing then continues at block 455 (FIG. 5), which determines a desired resolution based on available bandwidth and/or DVR storage space. The details of 455 will be discussed below in connection with the flowchart of FIG. 5.

Next, block 460 compares the resolution of the programming carried by the channel identified in the request to the desired resolution determined at block 455. If the two values are equal, then no channel adjustment is necessary and processing continues at block 480, described below. However, if the resolution of the requested channel is different than the desired resolution, block 465 attempts to find an analogous channel

carrying the same program but equal to, or at least closer to, the desired resolution. A person of ordinary skill in the art should be familiar with mechanisms to determine what program is carried on a particular channel at a particular time, as well as the resolution of the program and the running time (length) of the program.

Block 470 determines if an analogous channel is found. If Yes, block 475 updates ChannelId 320I in Settings 320 with the analogous channel, and updates Resolution 320R to the desired resolution determined at block 455. Otherwise, ChannelId 320I and Resolution 320R are unchanged.

Next, block 480 sets Delivery Mode 320M to BurstDelivery, NormalDelivery or TrickleDelivery, depending on available subscriber bandwidth and the resolution of the identified channel. If the available bandwidth is more than enough to support the chosen resolution, channel change decision logic 230 uses BurstDelivery. If the available bandwidth is not enough to support the chosen resolution, channel change decision logic 230 uses TrickleDelivery. Otherwise, DeliveryMode 320M is set to NormalDelivery. Having set DeliveryMode 320M, processing continues at block 425 where the selected channel is acquired using Settings 320, and process 400 is complete.

FIG. 5 is a flowchart showing further details of block 455 (ChooseResolution) from FIG. 4. The process of choosing an appropriate resolution for the requested channel begins with block 510, which determines a first maximum resolution based on the available DVR storage space and the length of the program. Next, at block 520, channel change decision logic 230 determines a second maximum resolution based on the available bandwidth of subscriber connection 165. This available bandwidth value is provided to channel change decision logic 230 by a bandwidth manager, where the functionality of a bandwidth manager should be familiar to a person of ordinary skill in the art. In some embodiments, the available bandwidth is subject to arbitration and

adjustment by the manager, and in such cases process 400 is executed again to use the adjusted value.

After block 520, block 530 chooses the resolution by setting a desired resolution to the minimum of the two maximum values calculated in blocks 510 and 520. The process completes, returning the desired resolution to the caller (block 455 in FIG. 4).

Although channels can carry video programming (e.g., viewing via live TV service 210TV or recording via DVR service 210DVR), other types of data are also provided via channels. Examples of such data include EPG data, channel icon bitmaps for the EPG, entitlements, channel maps, and updates to code within DHCT 180. Other types of data carried over a channel will also be recognized by a person of ordinary skill in the art, and are intended to be within the scope of this disclosure. A request to acquire one of these types of channels can be viewed as a request for access to, or bandwidth of, the downstream network, since such a request will result in data being sent from core network 140 down to DHCT 180 over subscriber connection 165.

FIG. 6 is a flowchart of a process used by another embodiment of channel change decision logic 230, one which chooses appropriate values for parameters used in a channel change that is a request for access to the downstream network. Channel change decision logic 230 chooses values based on the reason for, and/or source of, the request for access. In some scenarios, the source of the request is the user. In other scenarios, the reason for request involves the user, but indirectly: for example, a user viewing the electronic program guide (EPG) may scroll ahead to the future, so that DHCT 180 requests additional program guide data not already available in DHCT 180. In other scenarios, DHCT 180 initiates the request without any input from the user (e.g., a regularly scheduled update of EPG data).

The process 600 begins at block 610, where the request for network access is received. If channel change decision logic 230 determines (at block 620) that the request is for an activity that involves background data collection, then block 630 sets a priority variable to the lowest value. Background activities are those that do not require immediate display of requested data to the user. Examples of background activities are scheduled updates to guide data, or download of channel icons. If, on the other hand, block 620 determines that the request does not involve background data collection, then block 640 sets the priority variable to an intermediate value between lowest and highest. Processing then continues at block 650, which will be discussed below.

Returning to block 630 (executed if collection in background), next to execute is block 660, which determines whether the network access request was user-initiated. If Yes, then block 670 initializes Delay 320D (within Settings 320 of channel acquisition component 220) to RandomShort, and continues to block 675, where the highest priority request is chosen. However, if block 660 determines that the network access request was *not* user-initiated, then block 680 initializes Delay 320D to RandomLong. Processing then continues at block 675 to choose the highest priority request. (Block 675 will be discussed in further detail below.) A person of ordinary skill in the art should appreciate that "RandomShort" and "RandomLong" are randomly chosen values, with the range for RandomShort values being less than the range for RandomLong values.

Returning to block 650 (executed after blocks 620 and 640), it has been determined that high priority is appropriate for these requests because the request are not background data collection and are not user initiated. Such requests are for "critical activities", in that the DHCT has an immediate use for the data. Under such conditions, block 685 executes next to determine the source of the network access request. If the network access request was *not* received from navigator 210NAV, then block 690 sets the

priority variable to the highest value, and block 670 initializes Delay 320D to RandomShort. Examples of a source other than the navigator include: a channel map component requesting an update to the DHCT channel map; an entitlement component requesting an update to DHCT entitlements or conditional access keys; and an update component requesting an update to code within the DHCT. Such requests can be considered "system critical activities." Processing continues at block 675, which chooses the network access request with the highest priority, if more than one request has been received but not yet processed. Finally, at block 695 the channel identified in the request is acquired using Settings 320 (e.g., through AcquireChannel 330). Process 600 is then complete.

However, if block 685 determines the source of the network access request was navigator 210NAV (this is a "navigator critical activity"), then processing continues at block, which initializes Delay 320D to RandomLong. Processing then completes with the execution of block 675, which chooses the network access request with the highest priority, and block 695, which acquires the channel identified in the request.

A person of ordinary skill in the art should realize that process 600 may use a variety of mechanisms when testing various characteristics of the request for network access (e.g., user-initiated, source of request, whether request involves background data collection.) As merely one example, the source and/or reason code associated with the request may be implemented as a bitmap, where each bit encodes one of the aforementioned characteristics. Various enumerated values can then be formed from the bitmap so that process 600 operates in a desired manner. For example, to achieve a random delay with a long period and the lowest priority, the bits would be set/cleared as follows: Background=Set; UserInitiated=Set; Source=Don't Care. As another example,

the following bitmap will result in a random delay with a short period and the highest priority: Background=Clear; UserInitiated=Clear; Source=NOT(Navigator).

FIG. 7 is a block diagram showing selected components of an exemplary DHCT 180 which implements systems and methods of differentiated channel change behavior disclosed herein. DHCT 180 comprises: a network interface 710; a peripheral I/O interface 720; a display system 730; a decoder module 740; a storage device 750. a processor 760; and memory 770. These components are coupled by a bus 780. Omitted from FIG. 7 are a number of conventional components, known to those skilled in the art, that are unnecessary to explain the operation of the systems and methods of differentiated channel change behavior disclosed herein.

Network interface 710 receives a stream of packets over subscriber connection 165, delivered through core network 140 (FIG. 1). In some embodiments, subscriber connection 165 is implemented by HFC. In these embodiments, DHCT 180 includes a tuner/demodulator (not shown) which processes the digital signals received from the head-end over the HFC connection. Decoder module 740 decodes the incoming stream into a stream of decoded video frames. In some embodiments, decoder module 740 also removes frames from packets, and demultiplexes of multiple streams (e.g., audio and video). In some embodiments, decoder module 740 also decrypts the MPEG stream. Display system 730 converts the decoded video frames into a video signal for display by a computer monitor or a television (not shown).

Residing in memory 770 are services 210 and channel change decision logic 230 (FIG. 2). Memory 770 contains instructions that are executed by processor 760 to control operations of DHCT 180. Peripheral I/O interface 720 provides input and output signals, for example, user inputs from a remote control or front panel buttons or a keyboard, and outputs such as LEDs or LCD on the front panel.

Any process descriptions or blocks in flowcharts should be understood as representing modules, segments, or portions of code which include one or more executable instructions for implementing specific logical functions or steps in the process. As would be understood by those of ordinary skill in the art of the software development, alternate implementations are also included within the scope of the disclosure. In these alternate implementations, functions may be executed out of order from that shown or discussed, including substantially concurrently or in reverse order, depending on the functionality involved.

The systems and methods disclosed herein can be embodied in any computer-readable medium for use by or in connection with an instruction execution system, apparatus, or device. Such instruction execution systems include any computer-based system, processor-containing system, or other system that can fetch and execute the instructions from the instruction execution system. In the context of this disclosure, a "computer-readable medium" can be any means that can contain, store, communicate, propagate, or transport the program for use by, or in connection with, the instruction execution system. The computer readable medium can be, for example but not limited to, a system or propagation medium that is based on electronic, magnetic, optical, electromagnetic, infrared, or semiconductor technology.

Specific examples of a computer-readable medium using electronic technology would include (but are not limited to) the following: an electrical connection (electronic) having one or more wires; a random access memory (RAM); a read-only memory (ROM); an erasable programmable read-only memory (EPROM or Flash memory). A specific example using magnetic technology includes (but is not limited to) a portable computer diskette. Specific examples using optical technology include (but are not limited to) an optical fiber and a portable compact disk read-only memory (CD-ROM).

The foregoing description has been presented for purposes of illustration and description. It is not intended to be exhaustive or to limit the disclosure to the precise forms disclosed. Obvious modifications or variations are possible in light of the above teachings. The implementations discussed, however, were chosen and described to illustrate the principles of the disclosure and its practical application to thereby enable a person of ordinary skill in the art to utilize the disclosure in various implementations and with various modifications as are suited to the particular use contemplated. All such modifications and variation are within the scope of the disclosure as determined by the appended claims when interpreted in accordance with the breadth to which they are fairly and legally entitled.

CLAIMS

What is claimed is:

1. A method of handling channel changes in a digital home communication terminal (DHCT) comprising:

receiving a channel change request;

determining whether the channel change request is user-initiated;

setting at least one channel change parameter to a value based on the determination; and

performing a channel change using the at least channel change parameter having the set value.

2. The method of claim 1, wherein the at least one channel change parameter comprises a fast channel change parameter and the setting further comprises:

setting the fast channel change parameter to true if the channel change is user-initiated and to false otherwise.

3. The method of claim 1, wherein the at least one channel change parameter comprises a plurality of channel change parameters and the setting further comprises:

setting a fast channel change parameter, of the plurality of channel change parameters, to true if the channel change is user-initiated; and

setting a delay parameter, of the plurality of channel change parameters, to specify no delay if the channel change is user-initiated.

4. The method of claim 3, the setting further comprising:

setting a channel surf suppression parameter, of the plurality of channel change parameters, to specify suppression if the channel change is user-initiated.

5. The method of claim 1, wherein the at least one channel change parameter comprises a delay parameter and the setting further comprises:

setting the delay parameter to specify a random amount of delay, if the channel change is not user-initiated.

6. The method of claim 1, further comprising:

determining whether the channel change is initiated by a DVR service;

setting the at least one channel change parameter to another value based on the determination; and

performing a channel change using the at least channel change parameter having the set value.

7. The method of claim 6, wherein the at least one channel change parameter comprises an error correction parameter and the setting further comprises:

setting the error correction parameter to specify at least one of forward error correction (FEC) or selective retransmission, if the channel change is initiated by the DVR component.

8. The method of claim 6, wherein the at least one channel change parameter comprises a program resolution parameter and the setting further comprises:

setting the program resolution parameter to specify a predefined resolution if the channel change is not initiated by the DVR component or by the user.

9. A digital home communication terminal (DHCT) comprising:

a channel acquisition component;

memory having stored thereon program code; and

a processor that is programmed by at least the program code to enable the DHCT to:

receive a channel change request;

determine whether the channel change request is user-initiated; and

instruct the channel acquisition component to perform a channel change using at least channel change parameter, the channel change parameter having a value that is based on the determination.

10. The DHCT of claim 9, wherein the at least one channel change parameter comprises a fast channel change parameter having a value of true if the channel change is user-initiated and of false otherwise.

11. The DHCT of claim 9, wherein the at least one channel change parameter comprises a plurality of channel change parameters, wherein a fast channel change parameter, of the plurality of channel change parameters, is true if the channel change is

user-initiated, and wherein a delay parameter, of the plurality of channel change parameters, specifies no delay if the channel change is user-initiated.

12. The DHCT of claim 11, wherein a channel surf suppression parameter, of the plurality of channel change parameters, specifies suppression if the channel change is user-initiated.

13. The DHCT of claim 9, wherein the at least one channel change parameter comprises a delay parameter and the setting further comprises:

setting the delay parameter to specify a random amount of delay, if the channel change is not user-initiated.

14. The DHCT of claim 9, further comprising a DVR service executing on the processor, wherein the processor is further programmed to enable the DHCT to:

determine whether the channel change is initiated by the DVR service; and

instruct the channel acquisition component to perform a channel change using at least channel change parameter, the channel change parameter having another value that is based on the determination.

15. The DHCT of claim 14, wherein the at least one channel change parameter comprises an error correction parameter specifying at least one of forward error correction (FEC) or selective retransmission, if the channel change is initiated by the DVR service.

16. The DHCT of claim 14, wherein the at least one channel change parameter comprises a program resolution parameter specifying a predefined resolution if the channel change is not initiated by the DVR service or by the user.

17. A computer-readable medium having a computer program for processing data comprising:

logic configured to receive a channel change request;

logic configured to determine whether the channel change request is user-initiated;

logic configured to set at least one channel change parameter to a value based on the determination; and

logic configured to perform a channel change using the at least channel change parameter having the set value.

18. The computer-readable medium of claim 17, wherein the at least one channel change parameter comprises a fast channel change parameter and the setting further comprises:

setting the fast channel change parameter to true if the channel change is user-initiated and to false otherwise.

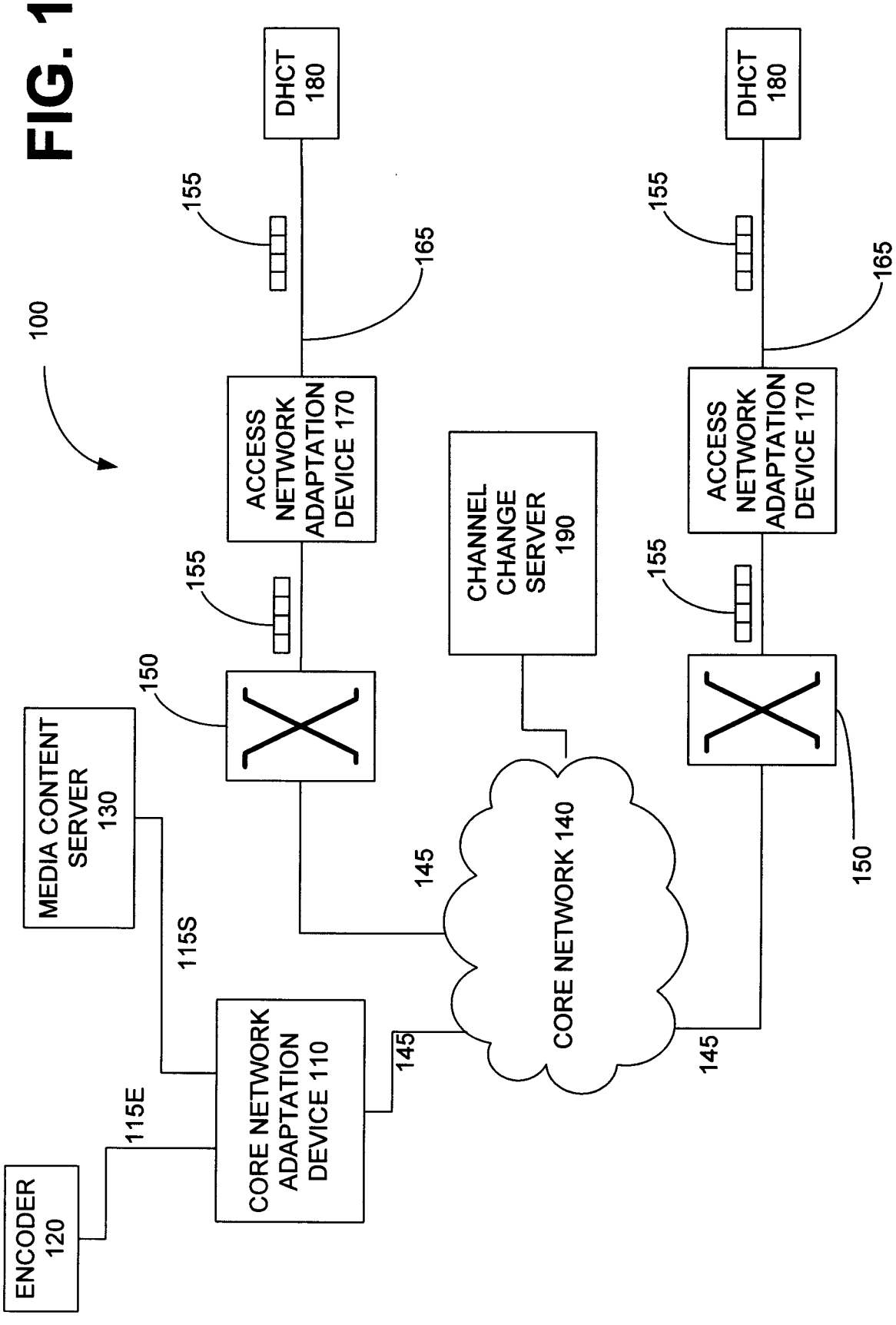
19. The computer-readable medium of claim 17, wherein the at least one channel change parameter comprises a channel surf suppression parameter and the setting further comprises:

setting the channel surf suppression parameter to specify suppression if the channel change is user-initiated.

20. The computer-readable medium of claim 17, wherein the at least one channel change parameter comprises a delay parameter and the setting further comprises:

setting the delay parameter to specify a random amount of delay, if the channel change is not user-initiated.

FIG. 1


100
ENCODER 120
115E
CORE NETWORK ADAPTATION DEVICE 110
115S
MEDIA CONTENT SERVER 130
145
CORE NETWORK 140
CHANNEL CHANGE SERVER 190
150
155
ACCESS NETWORK ADAPTATION DEVICE 170
165
DHCT 180

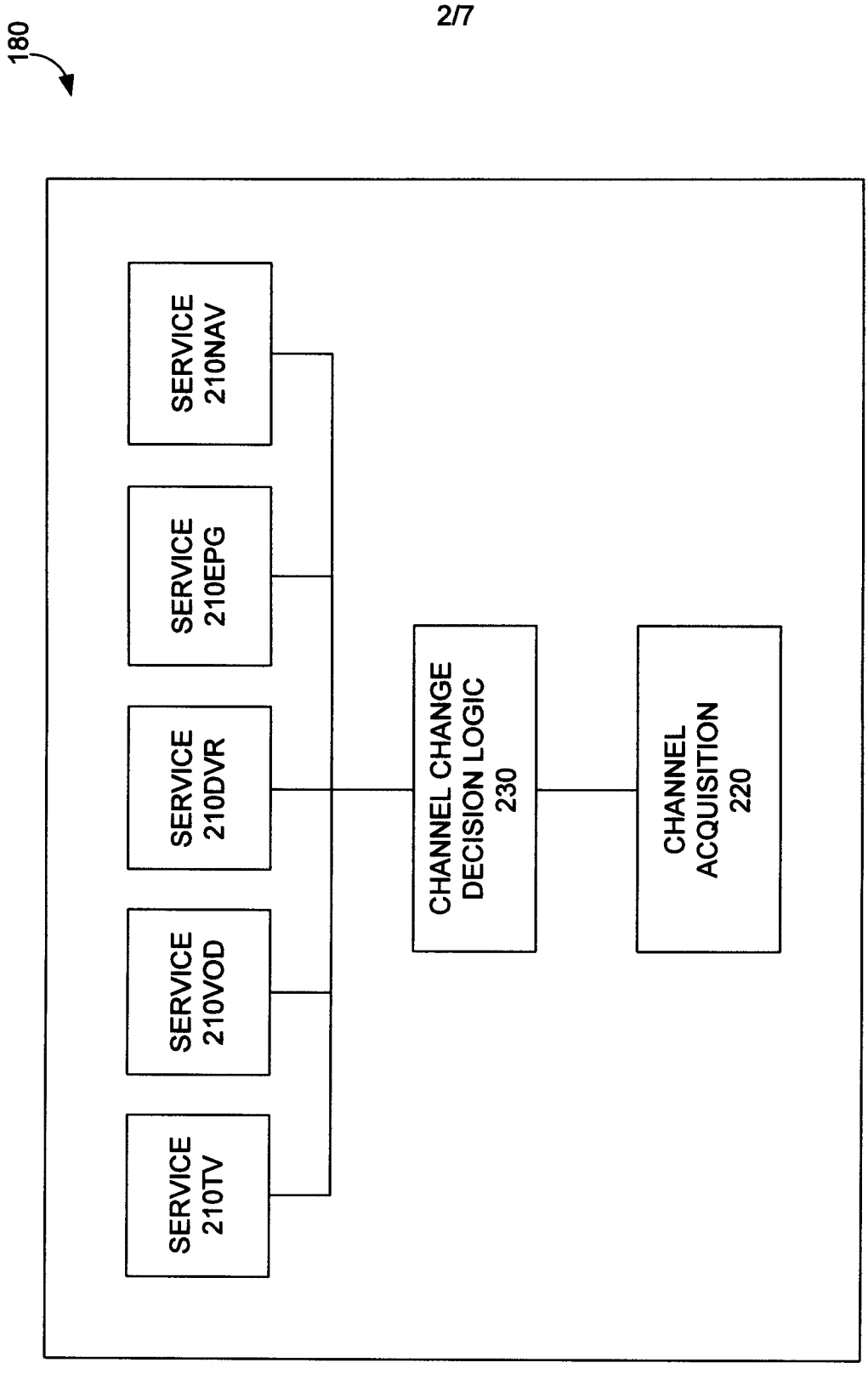

180
SERVICE 210NAV
SERVICE 210EPG
SERVICE 210DVR
SERVICE 210VOD
SERVICE 210TV
CHANNEL CHANGE DECISION LOGIC 230
CHANNEL ACQUISITION 220

FIG. 2

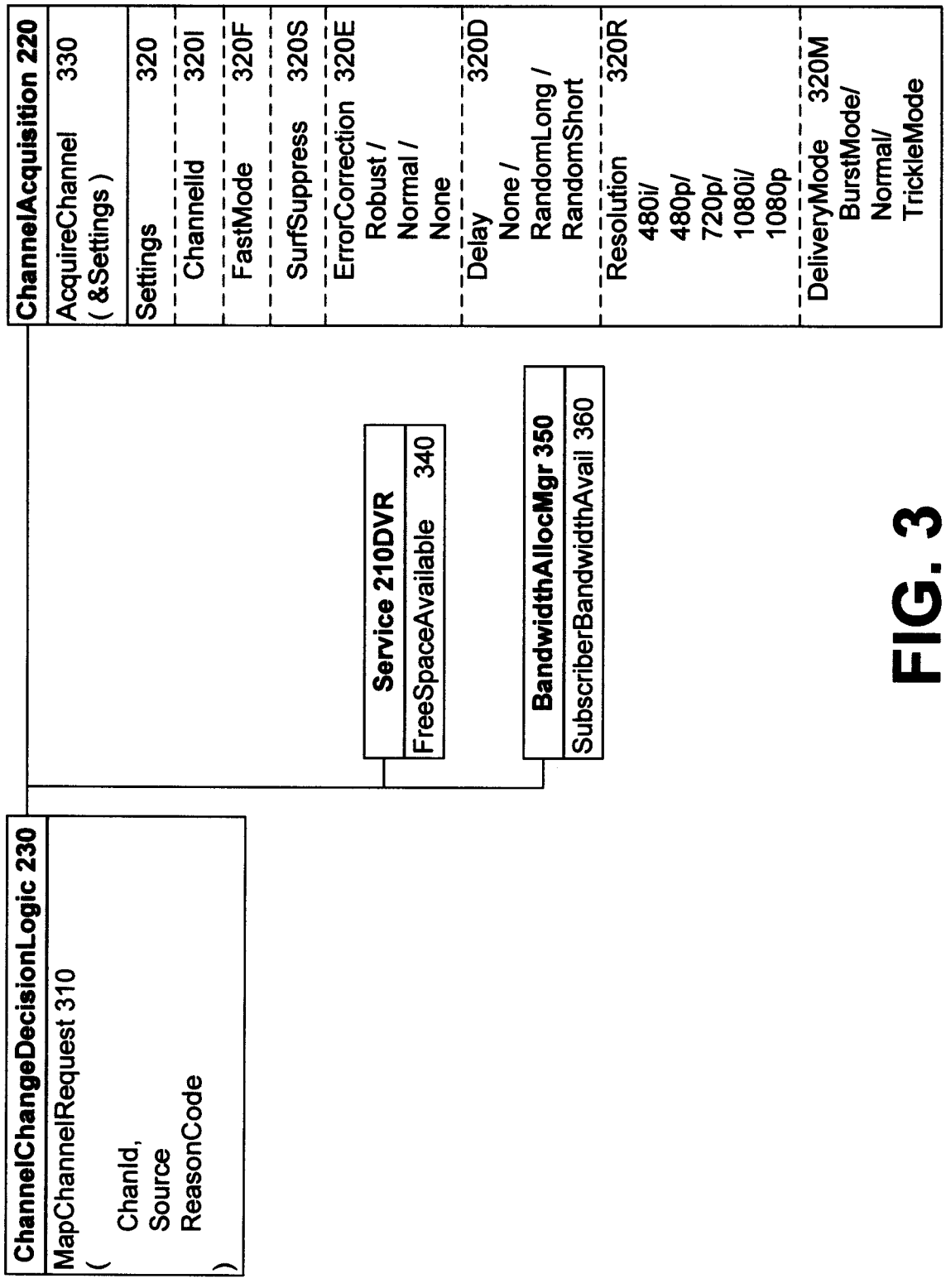

ChannelChangeDecisionLogic 230
MapChannelRequest 310
(
ChanId,
Source
ReasonCode
)
Service 210DVR
FreeSpaceAvailable 340
BandwidthAllocMgr 350
SubscriberBandwidthAvail 360
ChannelAcquisition 220
AcquireChannel 330
(&Settings)
Settings 320
ChannelId 320I
FastMode 320F
SurfSuppress 320S
ErrorCorrection 320E
Robust /
Normal /
None
Delay 320D
None /
RandomLong /
RandomShort
Resolution 320R
480i/
480p/
720p/
1080i/
1080p
DeliveryMode 320M
BurstMode/
Normal/
TrickleMode

FIG. 3

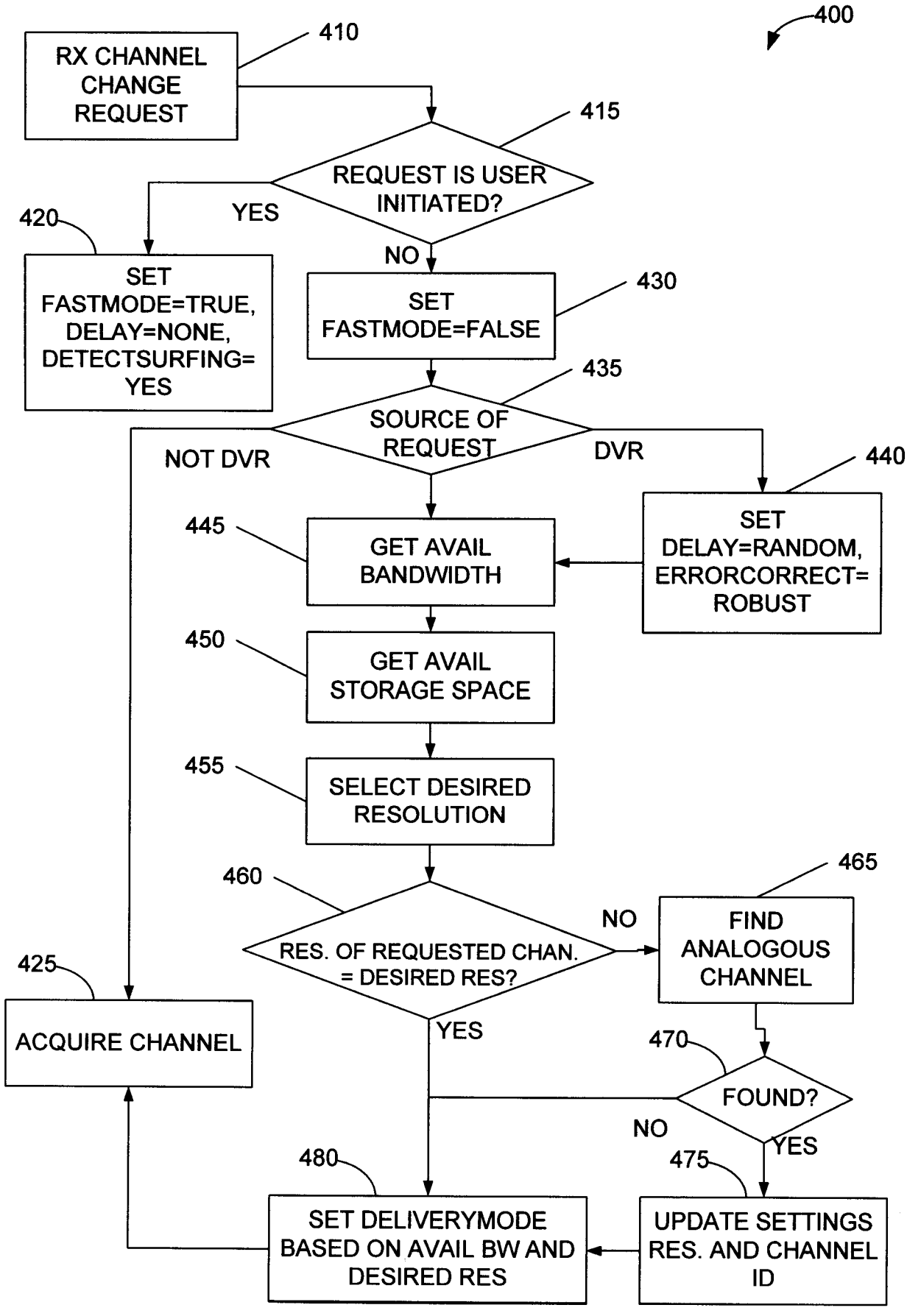

400
RX CHANNEL CHANGE REQUEST
410
REQUEST IS USER INITIATED?
415
YES
NO
420
SET FASTMODE=TRUE, DELAY=NONE, DETECTSURFING= YES
SET FASTMODE=FALSE
430
SOURCE OF REQUEST
435
NOT DVR
DVR
440
SET DELAY=RANDOM, ERRORCORRECT= ROBUST
445
GET AVAIL BANDWIDTH
450
GET AVAIL STORAGE SPACE
455
SELECT DESIRED RESOLUTION
460
RES. OF REQUESTED CHAN. = DESIRED RES?
NO
465
FIND ANALOGOUS CHANNEL
425
ACQUIRE CHANNEL
YES
470
FOUND?
NO
YES
480
475
SET DELIVERYMODE BASED ON AVAIL BW AND DESIRED RES
UPDATE SETTINGS RES. AND CHANNEL ID

FIG. 4

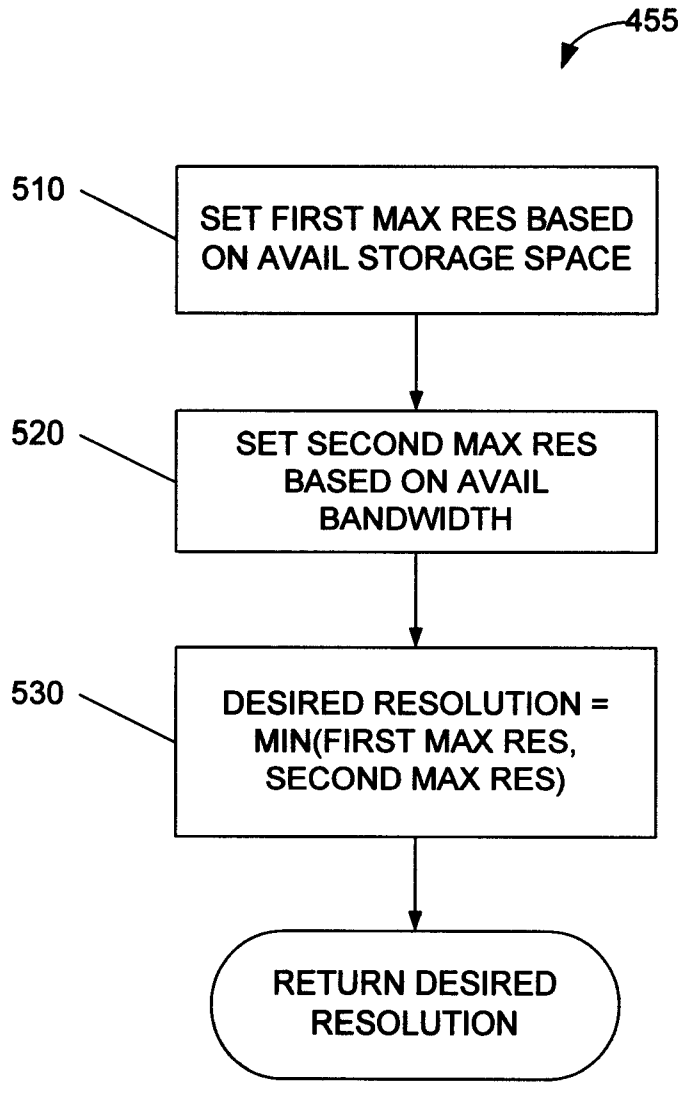
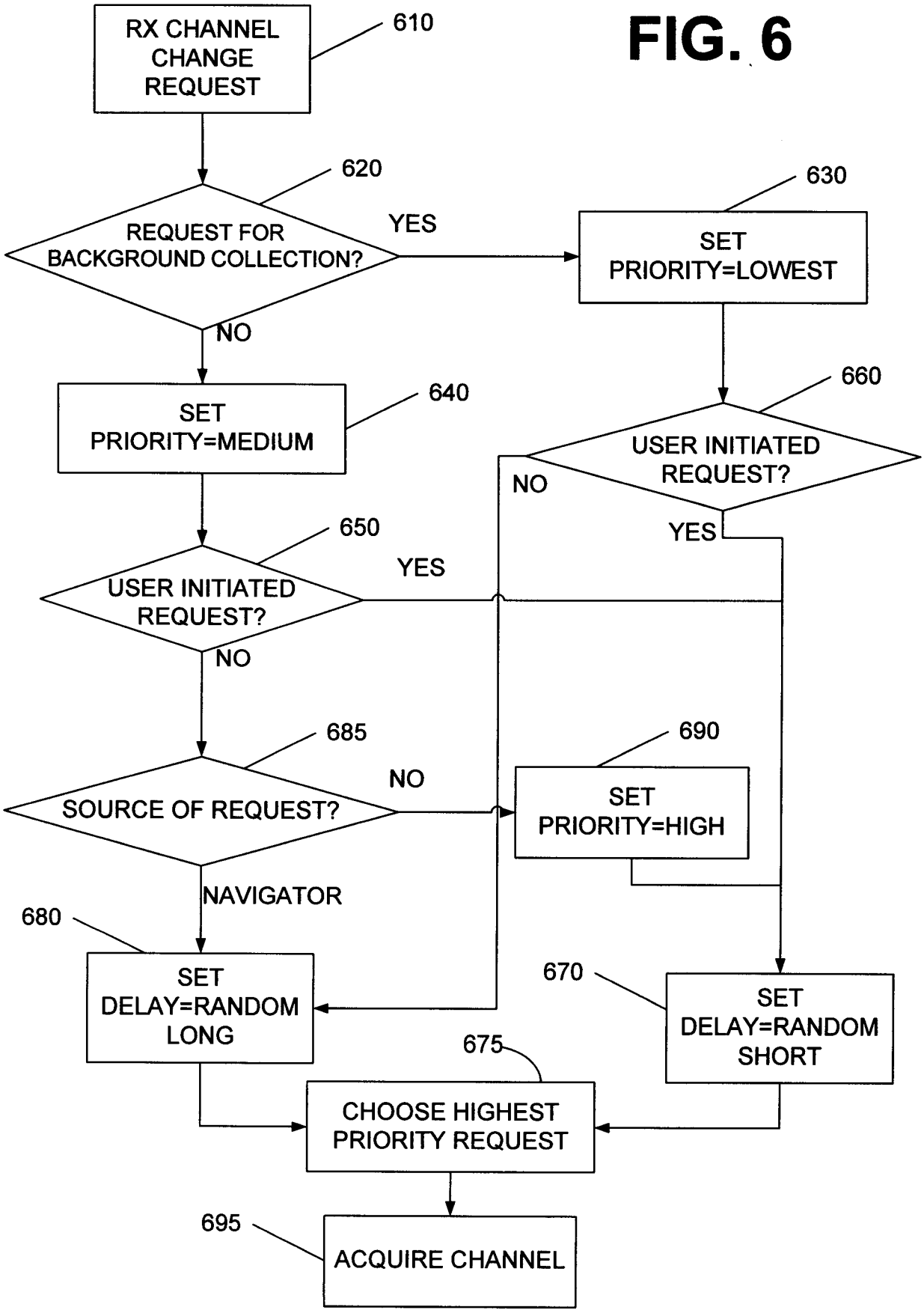

455
510
SET FIRST MAX RES BASED ON AVAIL STORAGE SPACE
520
SET SECOND MAX RES BASED ON AVAIL BANDWIDTH
530
DESIRED RESOLUTION = MIN(FIRST MAX RES, SECOND MAX RES)
RETURN DESIRED RESOLUTION

FIG. 5

FIG. 6


RX CHANNEL CHANGE REQUEST
610
620
REQUEST FOR BACKGROUND COLLECTION?
YES
630
SET PRIORITY=LOWEST
NO
640
SET PRIORITY=MEDIUM
660
USER INITIATED REQUEST?
NO
YES
650
USER INITIATED REQUEST?
YES
NO
685
SOURCE OF REQUEST?
NO
690
SET PRIORITY=HIGH
NAVIGATOR
680
SET DELAY=RANDOM LONG
670
SET DELAY=RANDOM SHORT
675
CHOOSE HIGHEST PRIORITY REQUEST
695
ACQUIRE CHANNEL

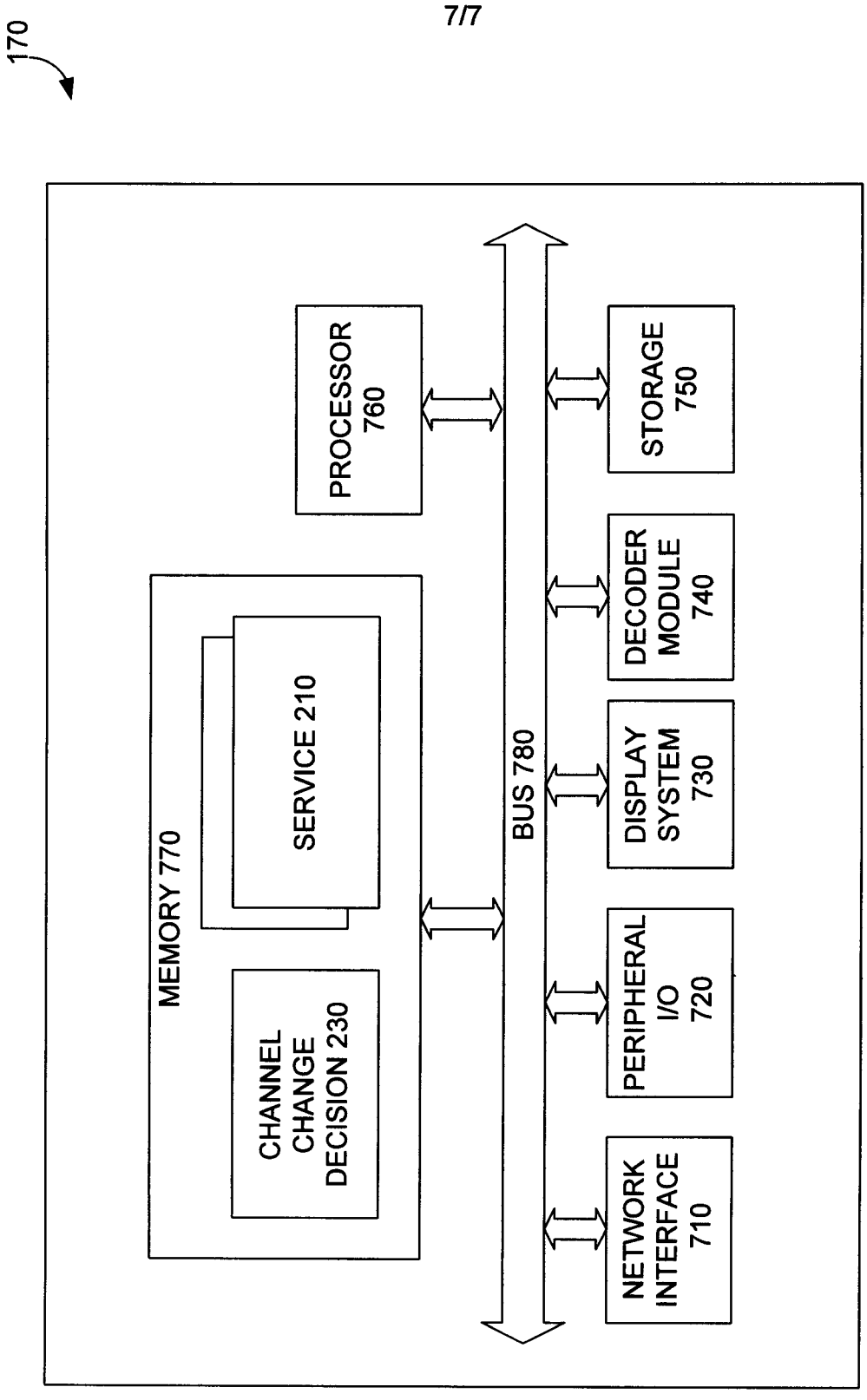

170
PROCESSOR 760
STORAGE 750
DECODER MODULE 740
DISPLAY SYSTEM 730
PERIPHERAL I/O 720
NETWORK INTERFACE 710
BUS 780
MEMORY 770
SERVICE 210
CHANNEL CHANGE DECISION 230

FIG. 7

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2008/070853

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04N5/00
ADD. H04N7/173

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)
H04N

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)
EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 6 628 301 B1 (ACTON COLIN L [US] ET AL) 30 September 2003 (2003-09-30) column 1, line 1 - line 14 column 1, line 51 - line 64 column 2, line 10 - line 15 column 3, line 32 - line 37 column 5, line 33 - line 51 column 6, line 25 - line 54 column 7, line 10 - line 16 column 7, line 36 - line 45 column 8, line 15 - line 22 column 9, line 38 - line 57	1,3-6,9, 11-14, 17,19,20
A	US 2006/080707 A1 (LAKSONO INDRA [CA]) 13 April 2006 (2006-04-13) paragraph [0107]	1,9,17

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

☒ See patent family 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
30 September 2008	07/10/2008

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	Dockhorn, Hans

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2008/070853

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 6628301 B1	30-09-2003	NONE	
US 2006080707 A1	13-04-2006	NONE	