CCITT SG XV
Working Party XV/1
Experts Group on ATM Video Coding

AVC- 315
July 1992

INTERNATIONAL ORGANISATION FOR STANDARDIZATION
ORGANISATION INTERNATIONALE DE NORMALISATION

ISO-IEC/JTC1/SC29/WG11

CODING OF MOVING PICTURES AND ASSOCIATED AUDIO

ISO-IEC/JTC1/SC29/WG11
MPEG 92/

Timing Recovery for Variable Bit-Rate Video on ATM Networks

Barry G. Haskell and Amy R. Reibman

7-3-92

Abstract

We consider the problem of receiver timing recovery when video and associated audio are coded for transmission on variable bit-rate channels such as ATM networks, LANs, etc. We assume the video encoder satisfies conditions that ensure the video encoder and decoder buffers do not overflow or underflow when the receiver and transmitter are perfectly synchronized. We then describe several methods for controlling receiver phase locked loops under conditions of variable cell rate as well as jitter in the transmission delay of each cell.

1 Introduction

Traditionally, video has been transmitted using channels that have constant bit-rate (CBR). Because most video compression algorithms use variable length codes to improve compression, a buffer at the encoder is necessary to translate the variable rate output by the encoder into the constant-rate channel. A similar buffer is necessary at the decoder to translate the constant channel bit-rate into a variable bit-rate. In such applications, synchronization of transmitter and receiver timing is easily accomplished by phase locking the receiver clock to the channel clock.

Recently, however, there has been much interest in sending video over Broadband Integrated Services Digital Networks (B-ISDN). These networks are able to support variable bit-rates by partitioning user data into a sequence of "cells" and inputting them to the network asynchronously. For this reason B-ISDN is referred to as an Asynchronous Transfer Mode (ATM) network.

ATM networks allow video to be transmitted with variable bit-rate (VBR). With VBR, statistical multiplexing can be employed to average the data amongst a large number of users. By this means, many more users can be accommodated by the network than with CBR.

In the ISO Moving Picture Experts Group (MPEG) audio-video coding standard [1], a Systems multiplexing capability is provided for combining bit-streams from several audio, video and/or auxiliary sources. At the receiver, the Systems demultiplexer feeds audio packets to audio decoders, video packets to video decoders and auxiliary packets to auxiliary decoders. Time stamps in the data packets enable synchronization of audio and video at the display.

The MPEG standard also provides for a Systems Time Clock (STC) of nominally 90 khz that, in principle, could be locked to a constant rate channel clock. Other clocks for audio and video could, in principle, also be locked to the STC.

However, in many variable bit-rate or packetized channels, there is no continuous channel clock for the receiver to lock to. Moreover, there may be "delay jitter" in the transmission of the data cells. In such cases, the STC must be derived by phase locking to received timing information.

Another complication arises in situations where video and audio clocks must be more stable than the STC. If this is the case, additional phase locking is needed.

2 Variable bit-rate communication channels

There are a number of situations where a video decoder may receive a variable bit-rate. For example, if compressed video is stored on a fast, random access hard disk, then data is typically accessed on demand by the decoder. Whenever there is room in the receiver buffer for a new portion of disk data, a disk read request is generated. As long as the buffer is large enough to smooth the data between disk accesses, it should never empty. In this case, the channel timing is controlled by the receiver clock, and there is no problem of channel-receiver clock conflict. However, in most other instances the channel clock is independent, and the receiver clock must accommodate.

If the transmitter and receiver both have access to the same stable channel clock, then they can both lock to it, and, as long as the transmission delay is constant, correct operation with variable bit-rate transport is possible using the methods described in [2].

However, in some networks the channel clocks may not all be synchronized. Moreover, transmission delay jitter may be significant, or clock stability may not be reliable. In some

implementations, data clocks may be present only when data is being transmitted.

In all these cases, the techniques of [2] are not completely sufficient to guarantee correct operation. Additional refinements are needed.

2.1 ATM networks

ATM networks are often proposed for transmitting video because they can accommodate the bit-rate necessary for high-quality video, and because the quality of the video can benefit from the variable bit-rate that the ATM network can theoretically provide. As a result, recent research has gone into developing video compression algorithms that have unconstrained bit-rate, but achieve constant quality [3]. By having the user select a desired quality, these algorithms can provide better compressed video than algorithms designed for a constant-rate channel, even when both algorithms produce the same average rate.

However, if the bit-rate of all streams were to vary arbitrarily, the network would be unable to provide guaranteed delivery of all cells. Two solutions to this have been proposed. The first solution is to have the user assign a priority (high or low) to each cell submitted to the network. The high priority cells are guaranteed by the network; the low priority cells can be dropped by the network. The second solution (which is still necessary even given the first) is to assume a contract between the network and the user. The network guarantees that the cell loss rate (CLR) for high priority cells will not exceed an agreed-to value, provided the user does not submit too many. A policing function monitors the user output, and either drops cells in excess of the contract, or marks these excess cells as low priority, possibly to be dropped later in the network. The advantages of priority-labeling, both for video [4,5,6,7] and for the network [8] have been well established.

For video, the existence of a policing function has a significant effect on the output bit-rate, because some information is essential to the decoder, e.g., timing data, start-of-picture codes, etc. If this information is not received, the video decoder will be unable to decode anything. Thus, virtually all high priority cells must be received for good video service, making it is essential that the video algorithm control its output bit-rate to ensure that the network-imposed policing function does not detect any excess high-priority cells.

3 MPEG Systems

3.1 Multiplexing

Fig. 1a shows a typical MPEG encoder-multiplexer for one video and one audio. System Time Clock (STC) Generator 160 consists of Oscillator 1601, whose output pulse stream on line 1602 drives Counter 1603, whose output on lines 1604, 1605 and 180 is the STC value. According to MPEG specifications, the frequency of Oscillator 1601 shall be 90 khz +- 50ppm. Moreover, its frequency drift shall not exceed 250 microhertz/second. Counter 1603 outputs a 33-bit STC value that increments once with each pulse input on line 1602.

The video input 110 is first fed to module 130, which inserts Presentation Time Stamps (PTS) on some or all pictures. If a PTS is put on a non-Bidirectionally predicted picture (nonB-picture), then a Decoding Time Stamp (DTS) must also be added. For these pictures, PTS exceeds DTS (in picture periods) by one plus the number of bidirectional pictures to be encoded. The resulting "stamped" data then passes to Video Encoder 120 and thence to Video Data Buffer 140 and Systems Encoder 150 to await transmission. The audio is treated similarly, except PTS are placed on some or all Audio Access Units (AAU).

The PTS/DTSs control the decoding and display of audio and video at the decoder in such a way as to maintain synchronism and to avoid overflow and underflow of the decoder buffers. Typically, the PTS/DTS values inserted by modules 130 and 135 are equal to some constant value plus the instantaneous STC values input on line 180.

Systems Encoder 150 forms packs and packets of data and sends them to the data channel. Pack headers contain System Clock Reference (SCR) values that are simply the instantaneous values of STC input on line 151. According to MPEG specifications, packs must be sent at least every 0.7 seconds. Moreover, MPEG assumes the channel data rate is constant for the duration of the pack.

Packets contain video or audio data with the PTS/DTS values, if any, moved to the packet header. In case of more than one PTS/DTS per packet, all except the first are discarded.

R_i on line 170 are the number of bits transmitted to the data channel during the coding of picture i. For the current picture, Systems Encoder 150 sends to Video Encoder 120 on line 170 the actual number of bits sent. For future pictures, Video Encoder 120 sends to Systems

Encoder 150 suggested values to be used. Systems Encoder 150 then chooses the actual values according to the principles described in [2].

3.2 Demultiplexing

Fig. 1b shows a typical MPEG receiver-demultiplexer, where the channel timing is not controlled by the receiver. The data channel provides input to the Systems Decoder 220, which demultiplexes audio and video data packets, routing them to the proper data buffers. The Systems Decoder 220 also detects received System Clock Reference (SCR) values and passes them to the Systems Time Clock Generator 210 on line 2107. Enable signal 2108 indicates when there is a valid SCR value on line 2107. Startup Value 2112 is used to set Counter 2104 to the very first SCR value that arrives.

The STC Generator 210 compares the received SCR values with the current value of the STC and corrects the STC if need be. This is done via a Phase Locked Loop (PLL) consisting of a subtractor 2101, a filter 2102, and Voltage Controlled Oscillator (VCO) 2103, which provides a pulse train for Counter 2104 that finally outputs the STC value on lines 2105 and 2106. The Filter 2102 smooths the differences SCR-STC that appear on line 2109, and uses the result to control the frequency of VCO 2103.

In the absence of transmission delay jitter, and assuming the VCO 2103 center frequency is close to that of the transmitter Oscillator 1601 , the output of Filter 2102, on line 2110, will eventually stablize to a small nearly constant value. If the frequency of Oscillator 1601 drifts slightly upward, then the SCR values will increase slightly, producing an increase in the values on lines 2109 and 2110, which in turn increases the VCO 2103 frequency. This in turn causes the STC values to increase slightly, thus tracking the values of SCR. Similarly, for a decrease in Oscillator 1601 frequency, SCR and STC values track each other by also decreasing.

The amount of smoothing by Filter 2102, ie, the time duration over which averaging occurs of the SCR-STC signal on line 2109, determines how fast the VCO 2103 can respond. A small amount of smoothing by Filter 2102 would cause a rapid alignment of STC and SCR. However, if audio and video clocks were also derived from the VCO 2103, such rapid variation could be detrimental to audio and video quality. In this case, much more smoothing by Filter 2102 may be necessary.

The gain of Filter 2102, or equivalently the input sensitivity of VCO 2103, also has an effect on the frequency stability of the pulse train on line 2111. If the gain is large, then a small difference between SCR and STC will cause a large frequency shift. If the gain is too large, then VCO 2103 might never stabilize. If the gain is too small, stable operation could occur even if SCR and STC were not close to each other.

The audio and video decoder timing is controlled via Presentation/Decode Time Stamps (PTS/DTS) that are included in the respective audio or video data. Assuming PTSs occur on every picture, the DTS (PTS, if no DTS) of the oldest picture stored in the Video Data Buffer 240 passes to Display Control 250 via line 241. Display Control 250 waits until the STC value on line 270 increases to the value of that oldest DTS. It then extracts the coded data for the corresponding picture from Buffer 240 via line 241 and passes it to Video Decoder 260 for decoding. For each picture input to Decoder 260, a picture is output via line 270. The output picture may be the same as the input (eg, for B-pictures), or it may be a stored previously decoded picture (eg, for nonB-pictures).

If a picture does not have a PTS, then Display Control 250 computes an extrapolated DTS value by simply adding the nominal picture duration to the DTS of the previous picture. Here we assume that Decoder 260 always finishes each picture before the next picture is due to be decoded.

When the oldest received (or extrapolated) audio PTS equals the STC, then the next audio access unit in Audio Data Buffer 245 is decoded and immediately passed to the audio output on line 275.

3.3 Display Control via PTS, DTS

Fig. 1c shows details of Display Control 250. Data is read from Buffer 240 via line 241, and passed upon request to Depacketizer 2501. This module strips off packet headers and detects the presence of DTS and/or PTS. (The first packet is required to contain a PTS.) If PTS/DTS is detected (set DTS=PTS if no DTS), then Switch 2504 is set to the A position by a signal on line 2502, and the DTS value is fed to the A input via line 2503 and hence to Switch 2504 output on line 2505 to appear as DTS1.

After stripping off packet headers, Depacketizer 2501 passes remaining data upon request

to Presentation Unit (PU) Detector 2519. For video, a PU is a picture; for audio it is an Audio Access Unit (AAU). At the start of the very first PU (and no others), PU Detector 2519 outputs a clock pulse on line 2517 that clocks DTS Register 2506, thus causing its input DTS1 to be read, stored and output as DTS2 on line 2507. Then PU Detector waits.

STC values are input via line 270 and pass to Subtractor 2513 where they are compared with the stored DTS2 values on line 2507. The difference STC-DTS2 is passed via line 2514 to Zero Detector 2515. When STC increases to a value equal to DTS2, Zero Detector 2515 outputs on lines 2516, 2512 and 2518 a signal enabling the next PU to be passed to the decoder. Upon receiving this signal on line 2518, PU Detector 2519 transfers the data for the next PU to Decoder 260 via line 251.

At the same time, the PU enable signal on line 2512 causes Switch 2504 to be set to the B position. If during the output of the PU, Depacketizer 2501 detects a PTS/DTS, it sets Switch 2504 to the A position, as before. However, if no PTS/DTS is detected it remains in the B position.

At the end of PU transfer to line 251, PU Detector 2519 outputs a clock pulse on line 2517 that clocks DTS Register 2506, as before. If a new DTS value has arrived, it is read as before from the A input of Switch 2504 on line 2503, through the switch via line 2505 as DTS1, and appears as DTS2 on line 2507. However, if no new DTS value has arrived, then Switch 2504 is still in its B position. In this case, an extrapolated DTS value appears as DTS1 on line 2505, as fed to the B input of Switch 2504 via line 2510. This extrapolated DTS value is calculated by adding, via Adder 2511, a constant nominal picture duration, on line 2509 in units of STC, to the previously used DTS2 value, on line 2508, and outputting the result on line 2510. For example, if the nominal picture frequency is 29.97hz and the STC frequency is 90khz, then the constant on line 2509 is 3003.

PU Detector 2519 then waits until STC increases to this new value of DTS2, whereupon it receives a PU Enable pulse on line 2518, thus repeating the cycle.

4 MPEG Systems Demultiplexing with Jitter

If there is substantial transmission delay jitter, then the demultiplexer of Fig. 1b may not work correctly. First, Filter 2102 must average over a much longer time interval than in the jitter free situation. However, even if this is the case, a more serious problem is that the data arrival times may be considerably different than was assumed by the encoder. When that happens there is the possibility of overflow or underflow of the data buffers.

Overflow can be avoided by increasing the buffer sizes above what was assumed by the encoder. Underflow is alleviated in Fig. 2 by the addition of a "Jitter Delay" D_j, which causes an extra accumulation of data in the buffers prior to decoding. This extra stored data provides insurance against underflow. In fact, if bounds on jitter can be guaranteed, the possibility of underflow can be eliminated by choosing sufficiently large buffers and a sufficiently large D_j.

An equivalent protection against buffer problems due to jitter could, in principle, be provided by the encoder if it ensured that in the Systems Target Decoder (STD) the buffers never emptied below or filled above some prescribed values determined by the expected jitter.

In Fig. 2, D_j, on line 465, is subtracted from the STC values, on line 455, by Subtractor 460 before they reach the Display Controls 250 and 255 via line 470. This has the effect of delaying the data extraction from the buffers and causing an extra accumulation in them. The amount of the delay for MPEG is given by $D_j/90000$ seconds.

Dynamic tracking of jitter is also possible as shown in the system of Fig. 2a. Most of the operations are the same as in Fig. 2. The output of Subtractor 2101 is a measure of the instantaneous delay jitter imposed by the channel. This is fed to Jitter Delay Computation Module 500 via line 510. The resulting Jitter Delay D_j is output on line 465.

Assuming buffer overflow is less of a problem, we only have to make sure that D_j is large enough to avoid underflow. Buffer underflow may occur if data is late in arriving, i.e. $STC > SCR$. Thus, a conservative choice of D_j would be the largest measured value of $STC - SCR$. In actual operation, D_j could be very slowly increased to the desired value without much effect on audio and video quality.

The instantaneous fullness of one or more buffers could also be used to modulate the jitter compensating delay D_j. In Fig. 2a the video buffer fullness is fed to Jitter Delay Computation

Module 500 via line 520.

An example of Jitter Delay computation is shown in Fig. 2b. At the start, step 530 initializes D_j to a nominal expected jitter value. Step 535 then reads the buffer fullness from line 520. If it is too large, indicating too much jitter delay, as determined by step 540 then step 545 decrements D_j. If the buffer fullness is too small, indicating too little jitter delay, as determined by step 550 then step 555 increments D_j.

Following this, step 560 reads $SCR - STC$ from line 510. Step 565 then changes its sign to obtain $STC - SCR$. Step 570 checks to see if $D_j < STC - SCR$, and if it is step 575 increments D_j.

The amounts of the increments and decrements depends on how fast an adaptation is desired and how much stability is required in the audio and video outputs.

5 Demultiplexing with Time Base Correction

In critical, high quality applications requiring stable video timing, the stability of Figs. 1 and 2 may not be sufficient. MPEG specifications call for encoder STC stability of 50 ppm, while industry standard video clock stability is 3 ppm. Thus, a method of divorcing the video timing from STC is needed.

In Fig. 3 a time base correction circuit is shown that can be used in place of Zero Detector 2515 in Fig. 1c. The input is the timing error signal (assuming a jitter delay) $STC - D_j - DTS2$ output by Subtractor 2513 on line 2514. This signal drives a phase-locked-loop consisting of Switch 305, Filter 310, Clipper 330 and Voltage Controlled Oscillator (VCO) 360.

VCO 360 outputs a train of short pulses at a nearly constant frequency on line 370. These pulses are used to enable the decoding of Presentation Units via line 2516, as described above.

VCO 360 has a very stable center frequency conforming to video industry standards, eg, 29.9700 hz. Moreover, its frequency variation is also limited, for example to +-3ppm during normal operation, by having its input voltage on line 350 clipped to some small range by Clipper 330. During startup it may be desirable to provide for a faster synchronization by increasing the allowable voltage range on line 350. Startup is indicated by a Startup Signal input on line 340.

The PU Enable Signal also appears on line 380, where it is used to enable Switch 305. Each PU Enable on line 380 causes Switch 305 to close, thus allowing the instantaneous timing error signal on line 2514 to pass through to Filter 310 via line 307. Filter 310 averages the sampled timing error signals, typically over some time window, and passes the result to Clipper 330 via line 320. Judicious choices of filter duration and gain determine, as before with Filter 2102, the stability and tracking rate of the phase-locked-loop.

6 Conclusion

We have described methods and systems for timing recovery in video systems where the transmission delay jitter may be substantial. Such conditions can occur on packetized networks where congestion may be dealt with by delaying some cells in network buffers until such time as they can be sent. In cases where high quality, real time audio and video must be displayed, the stability specifications of MPEG may not be stringent enough. In such situations, additional time base correction is necessary.

References

[1] "ISO 11172 Committee Draft of ISO-IEC/JTC1/SC29/WG11 Moving Picture Experts Group

[2] A. R. Reibman and B. G. Haskell, "Constraints on Variable Bit-Rate Video for ATM Networks," *IEEE Transactions on Circuits and Systems for Video Technology*, to be published.

[3] J. C. Darragh and R. L. Baker, "Fixed distortion subband coding of images for packet-switched networks," *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 5, pp. 789–800, June 1989.

[4] M. Ghanbari, "Two-layer coding of video signals for VBR networks," *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 5, pp. 771–781, June 1989.

[5] A. R. Reibman, "DCT-based embedded coding for packet video," *Image Communication*, June 1991.

[6] F. Kishino, K. Manabe, Y. Hayashi, and H. Yasuda, "Variable bit-rate coding of video signals for ATM networks," *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 5, pp. 801–806, June 1989.

[7] M. Nomura, T. Fujii, and N. Ohta, "Layered packet-loss protection for variable rate video coding using DCT," in *Second International Workshop on Packet Video*, 1988.

[8] G. Ramamurthy and B. Sengupta, "Modeling and analysis of a variable bit rate video multiplexer," in *Proceedings of the 7th ITC Seminar*, 1990.

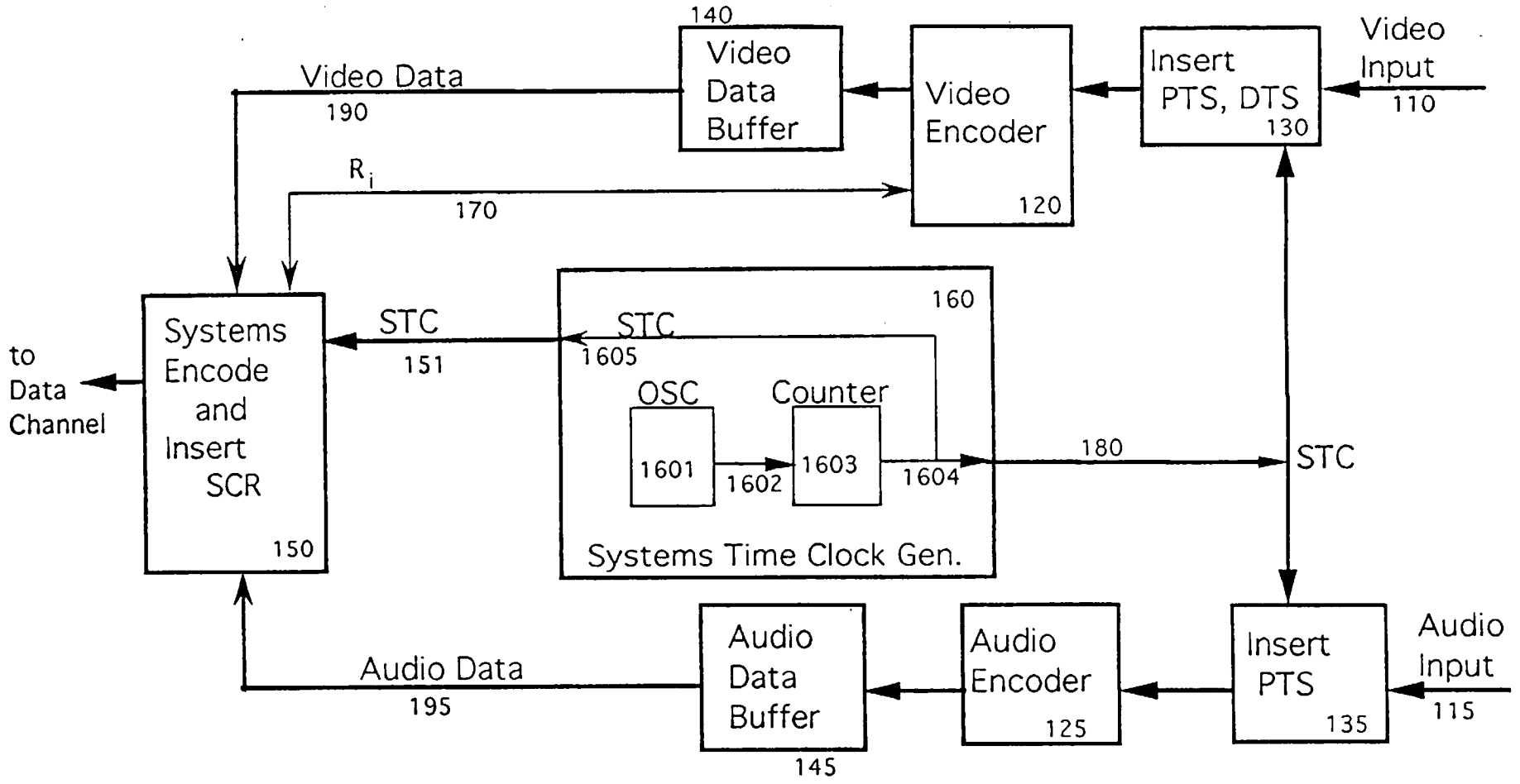

Fig. 1a MPEG Systems Multiplexer

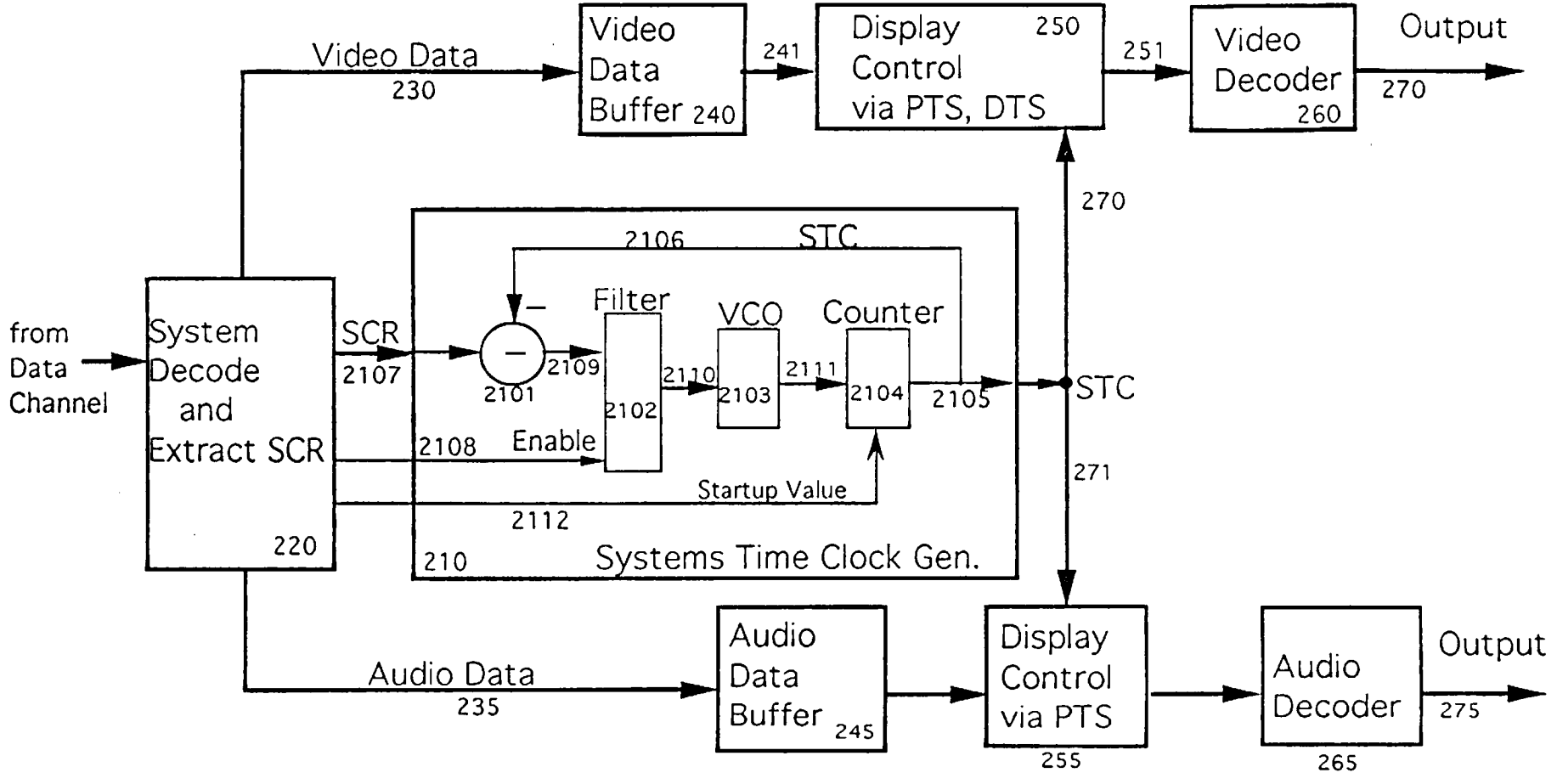

Fig. 1b MPEG Systems Demultiplexer

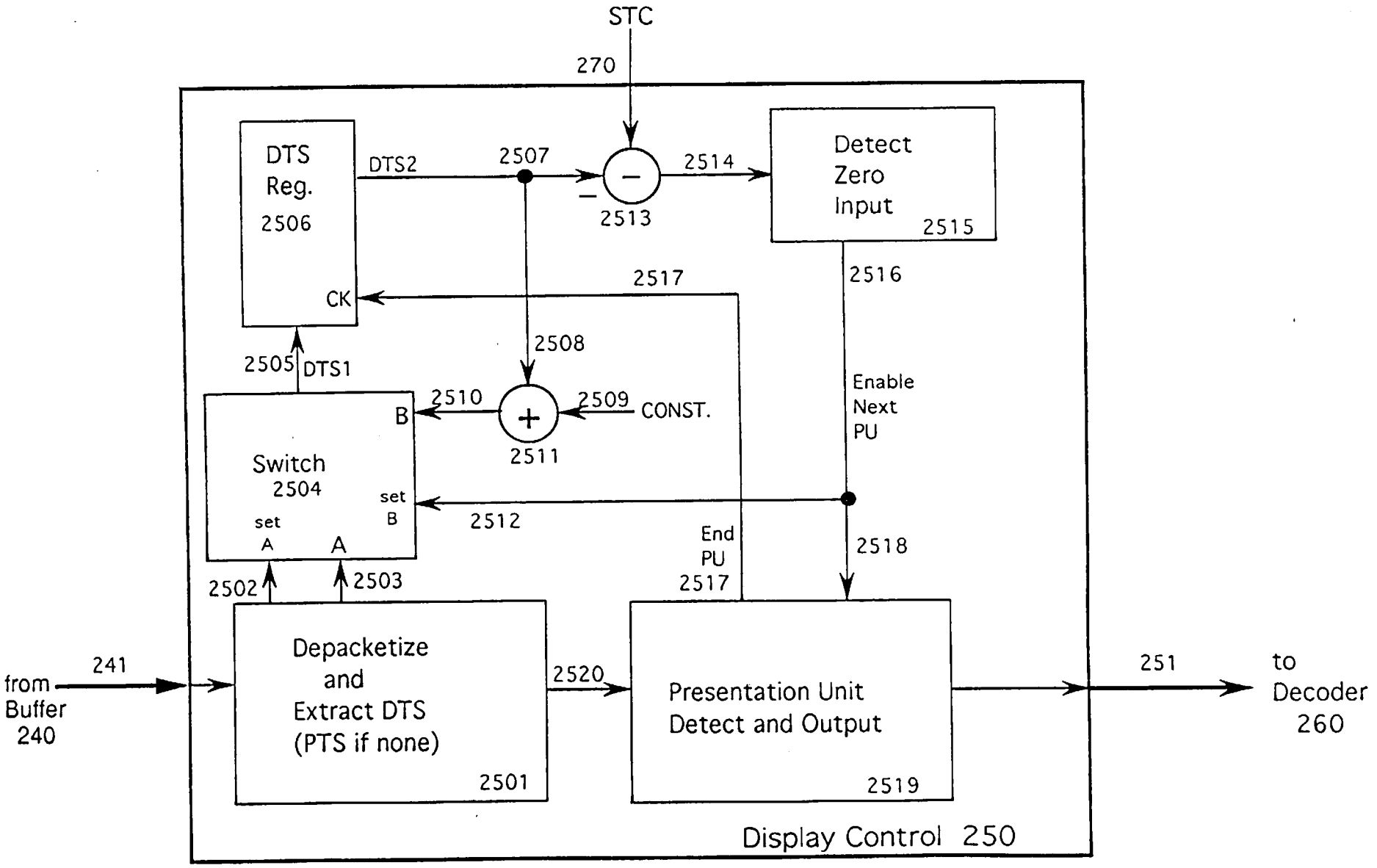

Fig. 1c Display Control via PTS,DTS

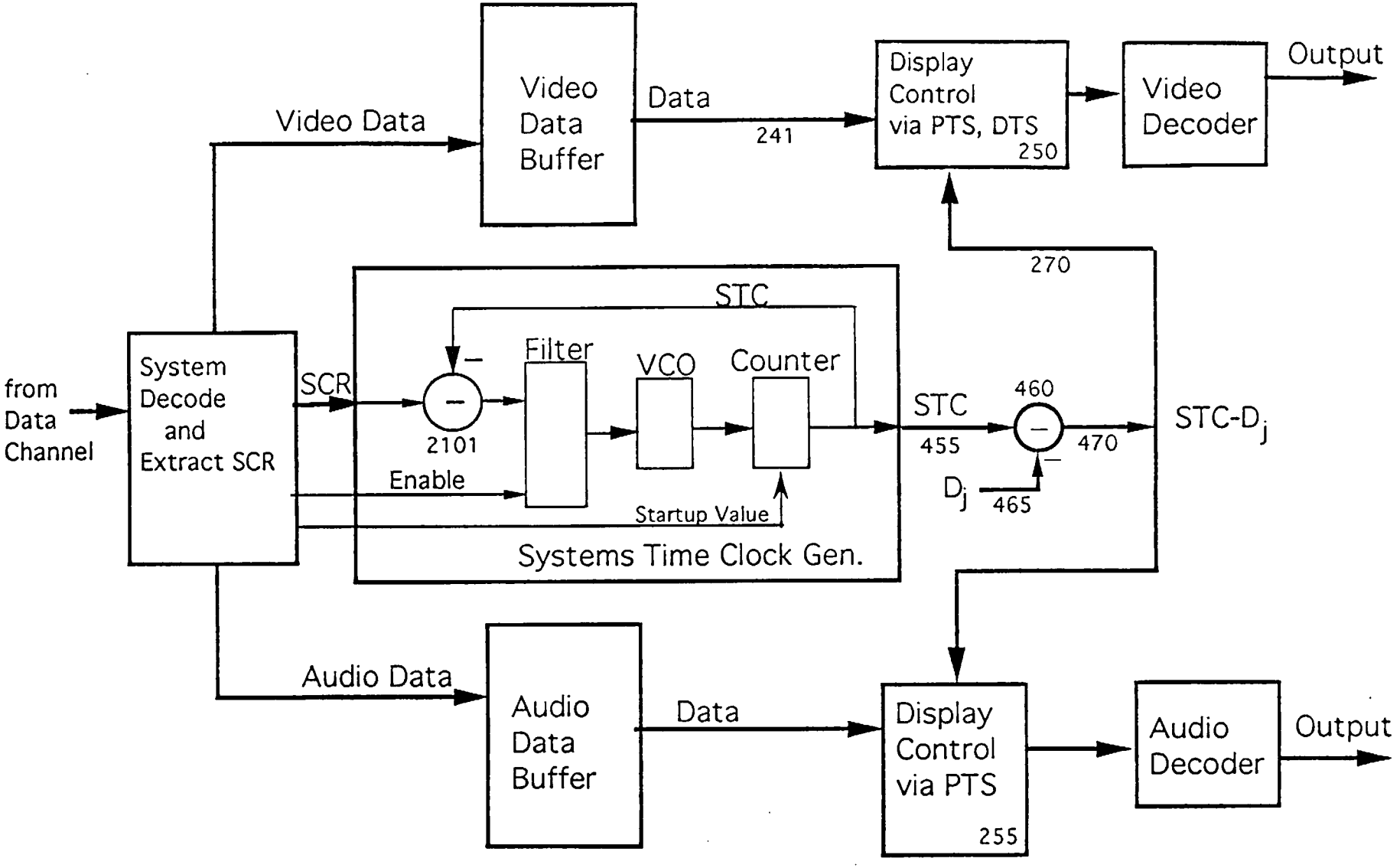

Fig. 2 Demux with Jitter Compensating Delay

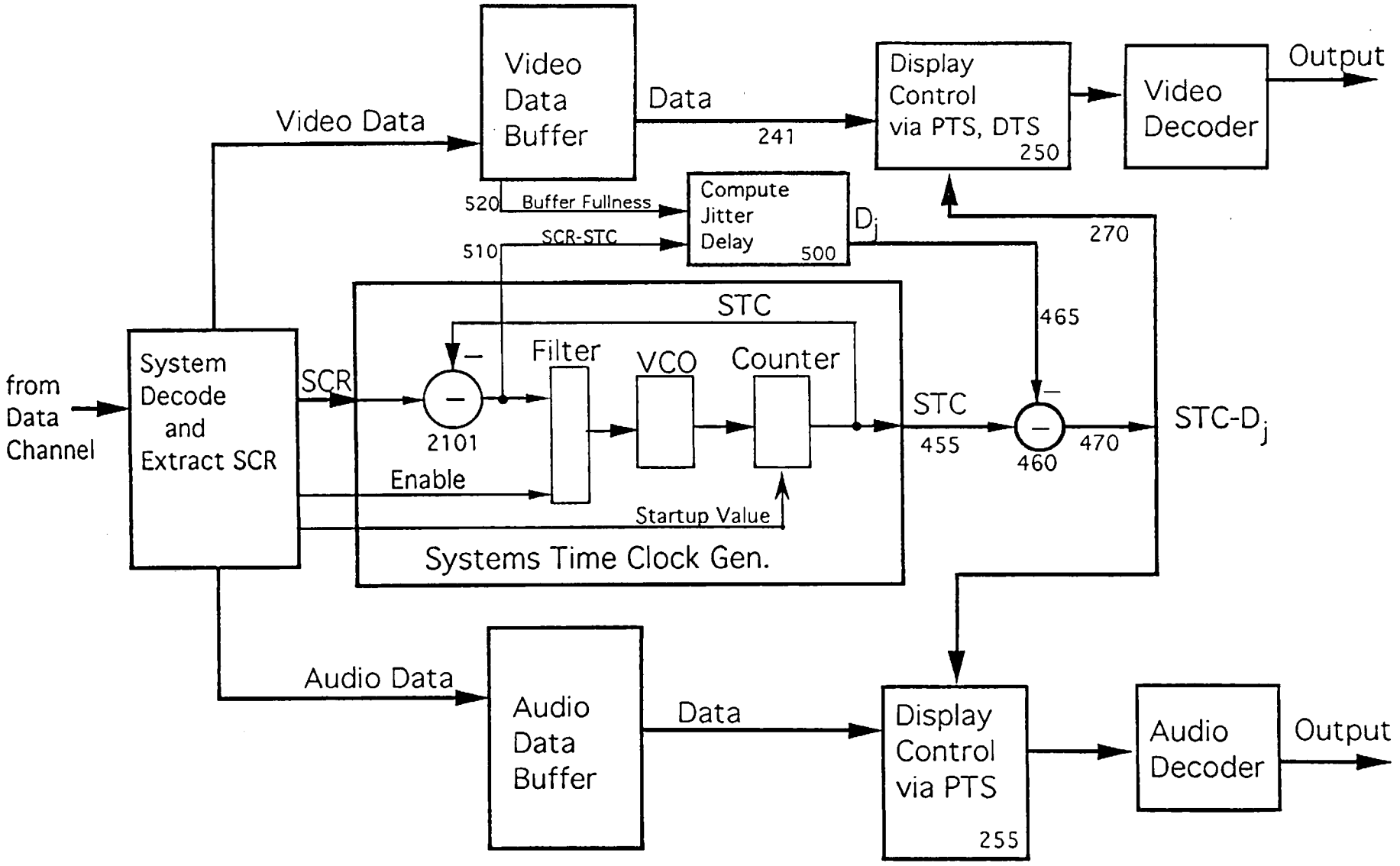

Fig. 2a Demux with Adaptable Jitter Compensating Delay

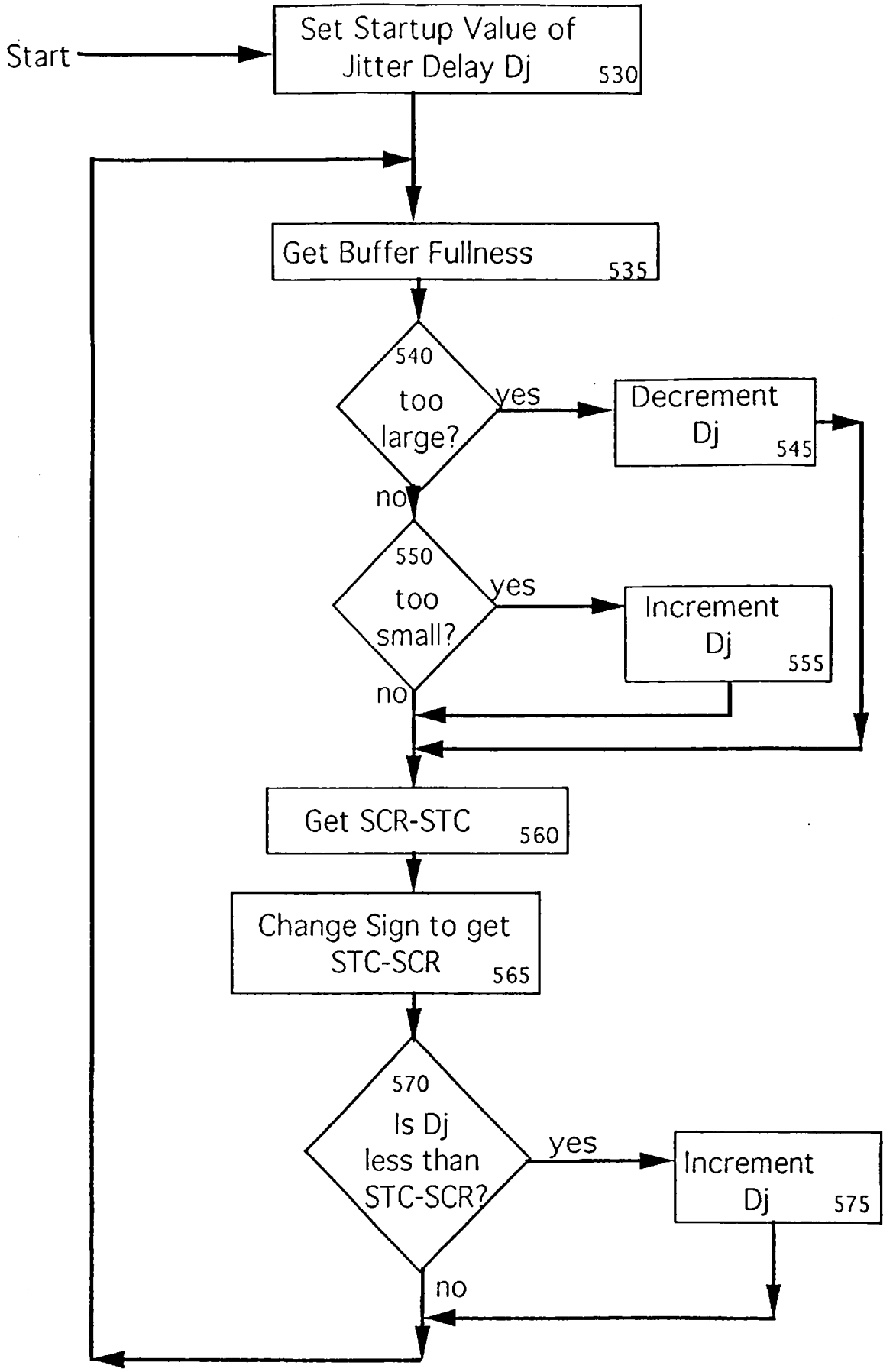

Fig. 2b Flow Chart of Jitter Delay Calculation

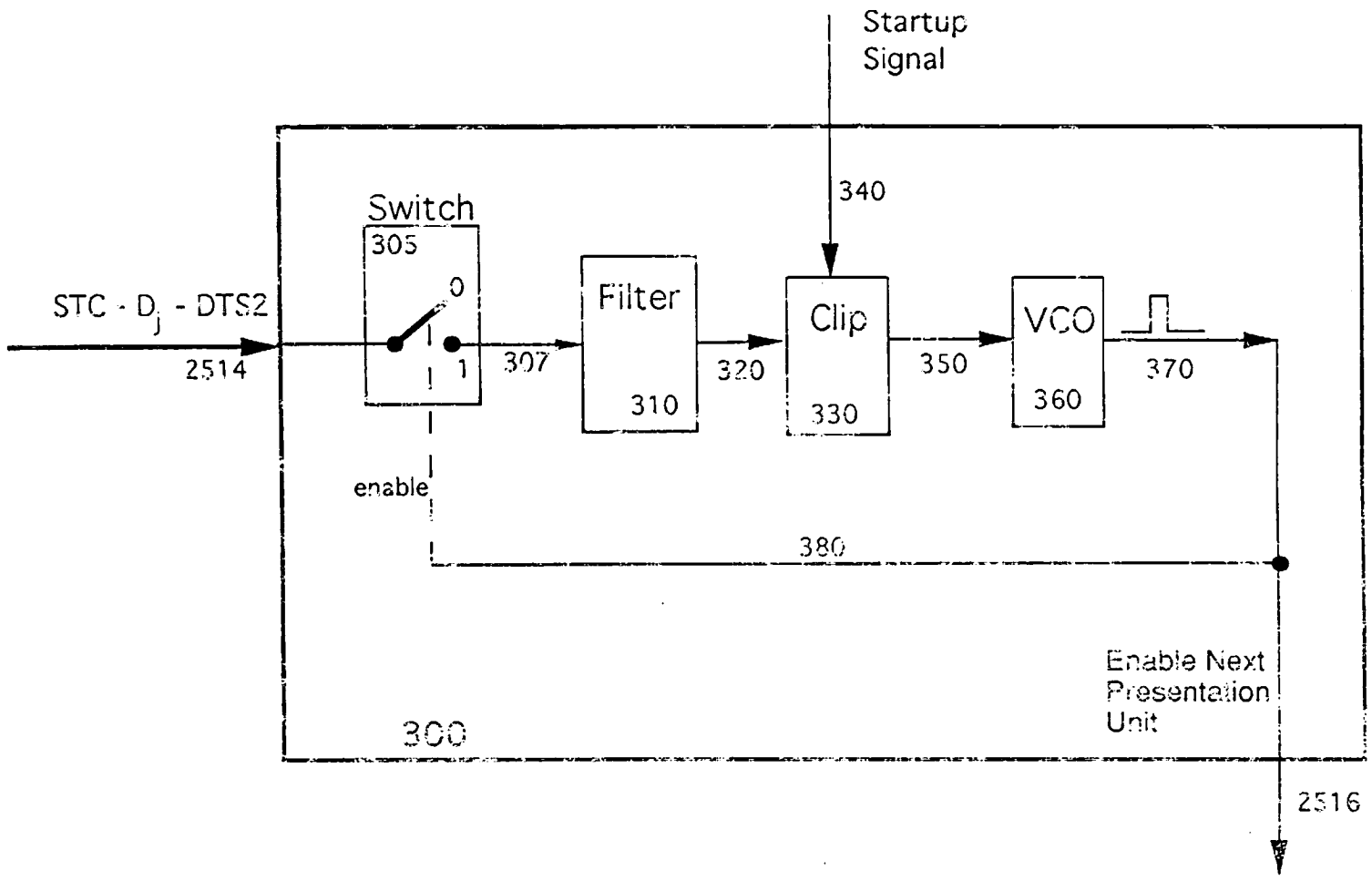

Fig. 3 Time Base Corrector