CCITT SGXV
Working Party XV/1
Experts Group for ATM Video Coding

Document AVC-273
July 1992

SOURCE: Japan
TITLE: Clock Recovery For Video
PURPOSE: Discussion

1 Introduction

It has been discussed which part should support video clock recovery function. The comparison of two cases, where the clock recovery function is supported by ATM adaptation layer (AAL) and supported by codec, is described in this contribution. Furthermore, three clock recovery methods are examined about their performances.

2 Comparison

2.1 Supported by AAL

It is needed to examine whether the clock recovery function in AAL can meet the various required jitter performances of the services such as the contribution service and the distribution service.

For the multimedia transmission service in which source clocks are asynchronous to each other, the media multiplexing functions should be carried out in AAL or virtual channel (VC) should be provided for each media. Furthermore the source clock should be recovered for each media. So, the AAL protocol becomes complex.

2.2 Supported by codec

The video source clock data must be sent with the coded video data. However, the clock recovery function can be designed to satisfy the jitter performance requirements of the service and is applicable to both STM and ATM. For the multimedia transmission service in which source clocks are asynchronous to each other, each source clock can be recovered independently of the multimedia multiplexing scheme.

¿From these discussions, the video source clock recovery function can be performed in codec more flexibly for multimedia multiplexing.

3 Clock Recovery Method

The following three clock recovery methods in codec are examined about their performances.

3.1 Buffer Control Method

This is the method that the recovered source clock is controlled by fill level of the transmission buffer. It is used in the adaptive clock supported by AAL type 1. The clock recovery circuit using the buffer control method is simple but the performance is degraded by large cell delay variation. Moreover, it is unable to recover source clock for VBR services.

3.2 Stuffing Method

The stuffing method is used in Part 1/H.120 and H.130. See Annex.

3.3 Frequency Counting Method

The frequency counting method is used in Part 3/H.120, H.130 and SRTS (Synchronous Residual Time Stamp) supported by AAL type 1. See Annex.

To compare three methods, although the frequency counting method and the stuffing method are better than the buffer control method in terms of the jitter performance, the frequency counting method might be the best method as it is better than other methods in terms of the adaptability for VBR services.

4 Conclusion

The source clock recovery function can be performed more flexibly in codec because it can be designed to satisfy the jitter performance requirements of the service. The frequency counting method might be the best of three methods in codec. If this method is adapted for H.26X, some information field should be provided in the syntax, e.g. 8 bits in the frame/field header.

Annex

COMPARISON OF TWO METHODS FOR VIDEO SOURCE CLOCK RECOVERY

1 Introduction

Two video source clock frequency recovery methods are described in Part 1 and Part 3 of H. 120 and H. 130 for the systems using 2.048 Mbit/sec and 1.544 Mbit/sec transmission lines, respectively.

This annex compares the two methods when they are applied to H. 26X.

2 Review of the methods in H. 120 and H. 130

2.1 Stuffing Method

Part 1 describes a stuffing method, namely:

- The video sampling frequency is 5.0 MHz $\pm$ 200 ppm and is locked to the line-scanning frequency of the incoming video signal.
- The justification is controlled by a comparison frequency of 22 500/11 (= 2045.4545) kHz, which is locked to the video clocks.
- The clock for the digital channel has a frequency of 2048 kHz $\pm$ 50 ppm.
- The phase of the channel clock is compared with that of comparison frequency every 250 μs interval, i.e. two frame periods, and when the channel-clock phase exceeds that of the comparison frequency by 2π radians, a 1 is transmitted, and otherwise a 0 is transmitted.

2.2 Frequency Counting Method

Part 3 describes a frequency counting method, namely:

- The video sampling frequency is 4 f_{sc} (= 14.3 MHz) and is not locked to the channel frequency of 1.544 MHz.
- The video sampling frequency clock is binary counted for the interval of 32 ms, i.e. two supermultiframe periods, and its lower 8 bits are transmitted.

3 Comparison

Item	Stuffing	Frequency Counting
Jitter Performance	O.K.	O.K.
Frame frequency[1]	Must be larger than allowable tolerance for video source clock frequency (2 kHz or so)	Not critical
Cell loss	Sync loss occurs, but the same as other VLCed words	Prior count values can be used with little or no degradation in jitter performance
Overhead	Frame frequency times 1 bit, 4 kbit/sec, or 0.2 % at 2048 kbit/sec	Frame frequency times 8 bits, 250 bit/sec
CBR Operation	Available bitrate may be confined to the multiple of the frame frequency	Arbitrary bitrate can be supported since frame frequency is not critical
VBR Operation	Difficult due to framing structure?	Perhaps no problem

[1] Inverse of the interval to send a stuff control bit or frequency count values.

4 Discussions

4.1 Using AAL functions or not

A stuffing method and a frequency counting method described above both belong to the last category of the three possible methods mentioned in AVC-214, "Introduction to AAL type 1", namely:

- Use of adaptive clock supported by AAL type 1,
- Use of SRTS supported by AAL type 1,
- Use of synchronization pattern with AAL user information flow (AAL type 1 is not involved for source clock frequency recovery).

Further discussion will be necessary to determine of which category one should try to seek our solution for video source clock frequency recovery.

4.2 Availability of common clock

The methods discussed in this document can be applicable only when a common network clock is available at both an encoder and a decoder.

Considerations for the case where LAN or PBX are involved must be discussed separately.

5 Conclusion

Two video source clock frequency recovery methods described in H. 120 and H. 130 are discussed for a possible solution for the problem.