ITU Telecommunication and Standardization Sector
Study Group 15
Expert Group for Video Coding and Systems
in ATM and Other Network Environments

Document AVC-719
January 1995

Source: Japan
Title: Usage of ITU-T stream_id
Purpose: proposal

1. Introduction

ITU-T SG15 Experts Group proposed the way to support ITU-T Video, Audio and C&I signals in MPEG2 SYSTEM at the previous MPEG meeting in Singapore (AVC-693/MPEG94/346). It was rejected that ITU-T Video and Audio streams use common stream_ids as those for MPEG, and 5 specific stream_ids (Type A to E), including for their usage for C&I, were allocated for ITU-T use instead. Among these 5 types, type E is allocated for a stream with unique format (i.e. not following a PES syntax).

Actual usage of these stream_ids should be specified by this experts group (in H.222.1). Concerning this usage, it is proposed in AVC-708R that the extension id should be introduced in the first byte of the PES payload. The purpose of this contribution is to provide a discussion material and straw-man proposal to fix the usage of ITU-T stream_ids.

2. Correspondence between media and stream_ids

This section discusses correspondence between media and stream_ids. There seem to be the following two solutions.

+ Each stream_id of type A-D has one-to-one correspondence with each media (method A)
+ Each stream_id of type A-D can be dynamically allocated to any media (method B)

The above two alternatives are compared according to the following three items.

(1) Demultiplexing
(i) In the case of TS

Demultiplexing is executed based on PID. Both methods will work without any problem.

(ii) In the case of PS

Demultiplexing is executed based on stream_id in MPEG2 SYSTEM (at least for MPEG defined media).

In the case of method A, plural streams belonging to the same ITU-T media cannot be identified only by a stream_id and two-stage demultiplexing becomes necessary, i.e. first demultiplexing of ITU-T media using stream_id and second demultiplexing of each stream using extension ids.

In the case of method B, correspondence between each stream and stream_id is shown by PSM (this means a dynamic id allocation like TS). For example, when an H.32X terminal uses two audio streams, i.e. audio-1 and audio-2, audio-1 and audio-2 can use Type A and Type B respectively. MPEG SYSTEM specifies PSM as optional but in this case PSM becomes mandatory for H.32X. However, since H.32X is supposed to use PSI/PSM for mode change, this is not a serious problem. Under the limitation that only 4 streams can be dealt with, one-stage demultiplexing the same as for MPEG defined media can be applied.

(2) Number of streams to be dealt with

Method B can handle only 4 streams, but it may be sufficient for a specific application such as conventional video conference. Method A can support as many streams as are desired because extra ids have yet to be specified.

(3) Media identification at the PES syntax level

If media identification only in terms of PES level syntax is necessary, adoption of extension id proposed in AVC-708R will also be required for method B. This functionality may be required when recording each demultiplexed stream. Even if method A is adopted, different coding law of the same ITU-T media cannot be identified only by stream_ids. It is recommended to include coding algorithm identifier in part of the extension id.

In summary, merit and demerit of both methods are as follows.

method A

merit: Many streams can be identified.

demerit: Needs two-stage demultiplexing for ITU-T media when PS is used.

method B

merit: The same demultiplexing method as for MPEG defined media can be applied.

demerit: Only 4 streams can be identified.

3. Usage of stream type E

Streams which have a stream_id type E can have their own syntax without PES header. Examples of this kind of stream are PSM, ECM, EMM, DSM-CC and so on. The question is if there is any data of this kind for ITU-T use. If there isn't, this stream_id will be reserved at this moment.

Candidates of such data are H.24X and video frame asynchronous user data such as JPEG or tele-writing?

4. Conclusion

Two alternatives for correspondence between media and stream_ids were compared. As the main difference of these two methods appears when PS is used as a multiplexing method and plural streams are multiplexed for the same ITU-T media, how or for what application PS will be used in the H.32X specification needs to be discussed, too.

There is no strong preference for either method in Japan but further investigation has been done for method A as a case study attached as an annex to this document, where the first 4 bits are allocated for coding algorithm identifiers and the following 4 bits are allocated for stream_numbers. Therefore, we propose the adoption of method A and accompanying description in the annex provided no problem is identified in the expert group discussion.

END

Annex to AVC-719

1 Introduction

It is our objective that one integrated H.32X audiovisual terminal can be used for not only conversational services but also retrieval and broadcasting services via ATM networks. This paper describes a case study for applying H.222.0 Transport Stream to the H.32X terminal. In this paper some modifications are made on AVC-695[1] for stream_id and stream_type codes based on clause D.7 to AVC-708R [2]. Changes' parts are indicated with shade.

2 Assignment to elementary stream

2.1 Stream_id

Five stream_ids labeled "ITU-T H.222.1 Type A" to "Type E" are provided for ITU-T specific elementary streams. We defined stream_id to elementary streams as follows;

Medium	stream_id	Coding law	Note
Audio	ITU-T Rec.H.222.1 type B	G.722, etc.	
Video	ITU-T Rec.H.222.1 type C	H.261, etc.	-Synchronous with audio
	H.262 | ISO/IEC 13818-2	H.262 | ISO/IEC 13818-2	
Data	ITU-T Rec.H.222.1 type D	JPEG high resolution still pictures, etc.	-Asynchronous with audio -Data channel is assumed to be always open
Control	ITU-T Rec.H.222.1 type A	H.24X, etc.	-ITU-T auxiliary data -C&I signals which may or may not be synchronous with audio -(For in-channel negotiation, control, indication, transfer mode change)

Note1) It is clear that two stream_ids labeled "Type B" and "Type C" are provided for ITU-T specific Audio and Video elementary stream respectively. But it is not yet clear that what kind of elementary streams are allocated to three stream_ids labeled "Type A", "Type D" and "Type E.

Note2) The channel for H.24X signaling is discussed in AVC-724[4].

2.2 Stream_id extension

The stream_id extension field is located to the first byte of the PES packet payload. This field effectively extends the stream_id field. We defined stream_id extension as follows;

b7 - b4	b3 - b0
coding_algorithm	stream_number

(1)ITU-T Rec.H.222.1 type A
for ITU-T C&I stream

coding_algorithm	description
0 h	reserved
1 h	H.320
2 h	H.32X
3 h - Fh	reserved

(2)ITU-T Rec.H.222.1 type B
for ITU-T audio stream

coding_algorithm	description
0 h	reserved
1 h	A-law
2 h	mu-law
3 h	G.721
4 h	G.722(mode 1)
5 h	G.722(mode 2)
6 h	G.722(mode 3)
7 h	G.728
8 h - Fh	reserved

(3)ITU-T Rec.H.222.1 type C
for ITU-T video stream

coding_algorithm	description
0 h	reserved
1 h	H.261
2 h - Fh	reserved

(4)ITU-T Rec.H.222.1 type D
for ITU-T auxiliary data stream

coding_algorithm	description
0 h	reserved
1 h	JPEG??
2 h - Fh	reserved

3 Case Study for applying H.222.0 Transport System

A case study has been carried out for a 5 Mbit/s high quality videoconferencing service in which H.222.0 Transport Stream is used for multiplexing video, audio, data and control information. Packets are allocated to video and audio so that the multiplexing delay becomes less than 10 ms. The control signal is sent as soon as a C&I event or transfer mode changing occurs. In addition, it is sent ten times per second to facilitate recovery from transmission error, switched video multipoint conferencing, and channel hopping. Furthermore, it is assumed that the PES packet format is commonly applied to both Program and Transport

Streams, thus the PES packet rate complies with the value for the CSPS (constrained system parameter stream): 300 packets per second at maximum.

3.1 Elementary streams and their bit rate values

Medium	Bit rate (kbit/s)	Coding law	Note
Audio	64	G.722	
Video	4575	H.262 | ISO/IEC 13818-2	-Synchronous with audio
Data	64	JPEG high resolution still pictures	-Asynchronous with audio -Data channel is assumed to be always open
Control	16	H.24X H.23X	-ITU-T auxiliary data -Synchronous with audio -For in-channel negotiation, control, indication, transfer mode change

3.2 Network

- CBR at 5 Mbit/s

3.3 Packetization delay

We are concerned with packetization delay due to the packet multiplex of audio, video and other signals. Since the audio bit rate is much lower than the video rate and the PES packet rate is restricted, audio determines the packetization delay.

3.4 Packet assignment in PES layer

Medium	Bit rate (kbit/s)	Number of PES packets per second	Packet length(byte) *
Audio***	64	125	64 + 14
Video***	4575	130.1	4397 + 14
Data***	64	10	800 + 9
Control	16	16	125 + 14
Overhead**	36.287 (0.8 %)		
Total	4755.29	281.1	

* The packet length indicates (data bytes) + (overhead bytes).
** Overhead includes header bytes and stuffing bytes.
*** In order to simplify packetization, Audio, Video and Data packet lengths are fixed.

3.5 Packet assignment in TS layer

Medium	Bit rate (kbit/s)	Number of TS packets per second	Average rate (kbit/s)
Audio	64	125	188.000 (125*188*8)
Video	4575	3121.4	4694.655 (3121.4*188*8)
Data	64	50	75.200 (50*188*8)
Control	16	16	24.064 (16*188*8)
PAT, PMT		4	6.016 (2*2*188*8)
Null Packet		8.1	12.065 (8.1*188*8)
Total	4719 (94.38 %)	3324.5	5000

3.6 PID assignment in TS layer

Contents	PID	Note
Program Association Table	0000 h	fixed
Program Map Table	0010 h	user defined
Video ES	0011 h	user defined
Audio ES	0012 h	user defined
Control ES	0013 h	user defined
Data ES	0014 h	user defined

3.7 Descriptors in the Program Map Table

See the companion contribution AVC-692 [2].

4 Packetization algorithm

1) As soon as 64 bytes of the audio signal are stored in the buffer, an Audio TS packet is sent.
2) When a C&I event occurs, a Control TS packet is sent just after the earliest available Audio TS packet.
3) When 182 bytes of the data signal are stored in the buffer, a Data TS packet is sent just after the earliest available Audio TS packet. If a Control TS packet is also available at that time, the Data TS packet is sent just after the Control TS packet.
4) Video signal is sent by forming TS packets occupying between the end of Control or Data TS packet and the beginning of Audio TS packet.
5) If the whole payload of a TS packet is not available at the time of its transmission, stuffing bytes are inserted.
6) If the multiplex buffer approaches underflow, a Null TS packet is sent.

Reference

[1]AVC-695 Consideration of H.222.0 Transport System as multimedia multiplexing for audiovisual communications, Nov. 1994, Singapore
[2]AVC-708R Report of the seventeenth experts group meeting in Singapore - Part 3, Nov. 1994, Singapore
[3] AVC-692 Proposal for ITU-T_stream_descriptors (Japan), Nov. 1994, Singapore
[4] AVC-724 H.24X channel (Japan), Jan. 1995, Japan

Table 1 Video packet in Transport layer

Syntax	No. of Bits	Codeword bin/hex/dec	Note
sync_byte	8	47 h	
transport_error_indicator	1	0 b	
payload_unit_start_indicator	1	0 b	
transport_priority	1	0 b	
PID	13	0011 h	Video ES
transport_scrambling_control	2	00 b	
adaptation_field_control	2	01 b	
continuity_counter	4	x h	sequential increment
data_byte	8*184		

Table 2 Audio packet in Transport layer

Syntax	No. of Bits	Codeword bin/hex/dec	Note
sync_byte	8	47 h	
transport_error_indicator	1	0 b	
payload_unit_start_indicator	1	1 b	
transport_priority	1	0 b	
PID	13	0012 h	Audio ES
transport_scrambling_control	2	00 b	
adaptation_field_control	2	11 b	
continuity_counter	4	x h	
adaptation_field_length	8	69 h	105
discontinuity_indicator	1	0 b	
random_access_indicator	1	0 b	
elementary_stream_priority_indicator	1	0 b	
PCR_flag	1	1 b	
OPCR_flag	1	0 b	
splicing_point_flag	1	0 b	
transport_private_data_flag	1	0 b	
adaptation_field_extension_flag	1	0 b	
program_clock_reference_base	33	xxxxxxxxx h	
reserved	6	111111 b	
program_clock_reference_extension	9	xxx h	
stuffing_byte	98*8	FF h	
data_byte	78*8		

Table 3 Data packet in Tranport layer

Syntax	No. of Bits	Codeword bin/hex/dec	Note
sync_byte	8	47 h	
transport_error_indicator	1	0 b	
payload_unit_start_indicator	1	1 b	
transport_priority	1	0 b	
PID	13	0014 h	Data ES
transport_scrambling_control	2	00 b	
adaptation_field_control	2	11 b	
continuity_counter	4	x h	
adaptation_field_length	8	1,66 h	1 or 102
discontinuity_indicator	1	0 b	
random_access_indicator	1	0 b	
elementary_stream_priority_indicator	1	0 b	
PCR_flag	1	1 b	
OPCR_flag	1	0 b	
splicing_point_flag	1	0 b	
transport_private_data_flag	1	0 b	
adaptation_field_extension_flag	1	0 b	
stuffing_byte	0, 101*8	FF h	101 ; last TS packet in PES packet
data_byte	182,81*8		81 ; last TS packet in PES packet

Table 4 ITU-T control packet in Tranport layer

Syntax	No. of Bits	Codeword bin/hex/dec	Note
sync_byte	8	47 h	
transport_error_indicator	1	0 b	
payload_unit_start_indicator	1	0 b	
transport_priority	1	0 b	
PID	13	0013 h	Control ES
transport_scrambling_control	2	00 b	
adaptation_field_control	2	11 b	
continuity_counter	4	x h	
adaptation_field_length	8	xx h	181-M
discontinuity_indicator	1	0 b	
random_access_indicator	1	0 b	
elementary_stream_priority_indicator	1	0 b	
PCR_flag	1	0 b	
OPCR_flag	1	0 b	
splicing_point_flag	1	0 b	
transport_private_data_flag	1	0 b	
adaptation_field_extension_flag	1	0 b	
stuffing_byte	(182-M) *8	FF h	
data_byte	M*8		variable length information

Table 5 Program Association Table in Tranport layer

Syntax	No. of Bits	Codeword bin/hex/dec	Note
sync_byte	8	47 h	
transport_error_indicator	1	0 b	
payload_unit_start_indicator	1	1 b	
transport_priority	1	0 b	
PID	13	0000 h	PAT
transport_scrambling_control	2	00 b	
adaptation_field_control	2	11 b	
continuity_counter	4	x h	
adaptation_field_length	8	A6 h	
discontinuity_indicator	1	0 b	
random_access_indicator	1	0 b	
elementary_stream_priority_indicator	1	0 b	
PCR_flag	1	0 b	
OPCR_flag	1	0 b	
splicing_point_flag	1	0 b	
transport_private_data_flag	1	0 b	
adaptation_field_extension_flag	1	0 b	
stuffing_byte	165*8	FF h	
pointer	8	00 h	
table_id	8	00 h	Program_association _section
section_syntax_indicator	1	1 b	
'0'	1	0 b	
reserved	2	11 b	
section_length	12	00D h	13
transport_stream_id	16	0000 h	User defined
reserved	2	11 b	
version_number	5	00 h	for the renewal of this table
current_next_indicator	1	x b	for the renewal of this table
section_number	8	00 h	
last_section_number	8	00 h	
program_number	16	0001 h	User defined
reserved	3	111 b	
program_map_PID	13	0010 h	PMT
CRC_32	32	xxxxxxxx h	

Table 6 Program Map Table in Tranport layer

Syntax	No. of Bits	Codeword bin/hex/dec	Note
sync_byte	8	47 h	
transport_error_indicator	1	0 b	
payload_unit_start_indicator	1	1 b	
transport_priority	1	0 b	
PID	13	0010 h	PMT
transport_scrambling_control	2	00 b	
adaptation_field_control	2	11 b	
continuity_counter	4	x h	
adaptation_field_length	8	81 h	129
discontinuity_indicator	1	0 b	
random_access_indicator	1	0 b	
elementary_stream_priority_indicator	1	0 b	
PCR_flag	1	0 b	
OPCR_flag	1	0 b	
splicing_point_flag	1	0 b	
transport_private_data_flag	1	0 b	
adaptation_field_extension_flag	1	0 b	
stuffing_byte	128*8	FF h	128
pointer	8	00 h	
table_id	8	02 h	program_map _section
section_syntax_indicator	1	1 b	
'0'	1	0 b	
reserved	2	11 b	
section_length	12	00D h	13
program_number	16	0001 h	
reserved	2	11 b	
version_number	5	xx h	for the renewal of this table
current_next_indicator	1	x b	for the renewal of this table
section_number	8	00 h	
last_section_number	8	00 h	
reserved	3	111 b	
PCR_PID	13	0012 h	Audio Packet
reserved	4	1111 b	
program_info_length	12	000 h	Nothing for Now
stream_type	8	02 h	H.262
reserved	3	111 b	
elementary_PID	13	0011 h	Video Packet
reserved	4	1111 b	
ES_info_length	12	005 h	

descriptor_tag	8	02 h	video_stream _descriptor
descriptor_length	8	03 h	
multiple_frame_rate_flag	1	0 b	single frame rate
frame_rate_code	4	4 h	29.97 Hz
MPEG_1_only_flag	1	1 b	MPEG2
constrained_parameter_flag	1	0 b	
still_picture_flag	1	0 b	
profile_and_level_indication	8	58 h	SP@ML
chroma_format	2	01 b	4:2:0
frame_rate_extension_flag	1	0 b	
reserved	5	11111 b	
stream_type	8	09 h	ITU-T Auxiliary
reserved	3	111 b	
elementary_PID	13	0012 h	Audio Packet
reserved	4	1111 b	
ES_info_length	12	004 h	
descriptor_tag	8	65 h	ITU-T Audio tag (New)
descriptor_length	8	02 h	
coding_algolithm	8	04 h	G.722 Mode 1 (New)
reserved	8	FF h	(New)
stream_type	8	09 h	ITU-T auxiliary
reserved	3	111 b	
elementary_PID	13	0013 h	Control Packet
reserved	4	1111 b	
ES_info_length	12	004 h	
descriptor_tag	8	67 h	ITU-T Control (New)
descriptor_length	8	02 h	
protocol	8	01 h	H.24X, H.23X (New)
reserved	8	FF h	(New)
stream_type	8	09 h	ITU-T auxiliary
reserved	3	111 b	
elementary_PID	13	0014 h	Data Packet
reserved	4	1111 b	
ES_info_length	12	004 h	
descriptor_tag	8	66 h	ITU-T Data (New)
descriptor_length	8	02 h	
reserved	16	FFFF h	(New)
CRC_32	32	xxxxxxxx h	

Table 7 Video packet in PES layer

Syntax	No. of Bits	Codeword bin/hex/dec	Note
packet_start_code_prefix	24	000001 h	
stream_id	8	E0 h	H.262
PES_packet_length	16	1135 h	4405
'10'	2	10 b	
PES_scrambling_control	2	00 b	
PES_priority	1	0 b	
data_alignment_indicator	1	0 or 1 b	
copyright	1	0 b	
original_or_copy	1	1 b	
PTS_DTS_flags	2	00 / 10 b	10 b; with PTS
ESCR_flag	1	0 b	
ES_rate_flag	1	0 b	
DSM_trick_mode_flag	1	0 b	
additonal_copy_info_flag	1	0 b	
PES_CRC_flag	1	0 b	
PES_extension_flag	1	0 b	
PES_header_data_length	8	0A h	
'0010' *	4	0010 b	
PTS [32..30] *	3	x h	
marker_bit *	1	1 b	
PTS [29..15] *	15	xxxx h	
marker_bit *	1	1 b	
PTS [14..0] *	15	xxxx h	
marker_bit *	1	1 b	
stuffing_byte	0,5*8	FF h	0; with PTS
PES_packet_data_byte	4397*8		

* When PTS_DTS_flags is '00 b', these fields don't exist.

Table 8 Audio packet in PES layer

Syntax	No. of Bits	Codeword bin/hex/dec	Note
packet_start_code_prefix	24	000001 h	
stream_id	8	F5 h	ITU-T Rec.H.222.1 type B (audio)
PES_packet_length	16	0048 h	73
'10'	2	10 b	
PES_scrambling_control	2	00 b	
PES_priority	1	0 b	
data_alignment_indicator	1	0 or 1 b	
copyright	1	0 b	
original_or_copy	1	1 b	
PTS_DTS_flags	2	00 / 10 b	10 b; with PTS
ESCR_flag	1	0 b	
ES_rate_flag	1	0 b	

DSM_trick_mode_flag	1	0 b	
additonal_copy_info_flag	1	0 b	
PES_CRC_flag	1	0 b	
PES_extension_flag	1	0 b	
PES_header_data_length	8	0A h	
'0010' *	4	0010 b	
PTS [32..30] *	3	x h	
marker_bit *	1	1 b	
PTS [29..15] *	15	xxxx h	
marker_bit *	1	1 b	
PTS [14..0] *	15	xxxx h	
marker_bit *	1	1 b	
stuffing_byte	0,5*8	FF h	0; with PTS
stream_id extension	8	20 h	G.711 mu-law stream number 0
PES_packet_data_byte	64*8		Audio data

* When PTS_DTS_flags is '00 b', these fields don't exist.

Table 9 auxilialy Data packet in PES layer

Syntax	No. of Bits	Codeword bin/hex/dec	Note
packet_start_code_prefix	24	000001 h	
stream_id	8	F7 h	ITU-T Rec.H.222.1 type D (data)
PES_packet_length	16	0323 h	803
'10'	2	10 b	
PES_scrambling_control	2	00 b	
PES_priority	1	0 b	
data_alignment_indicator	1	0 b	
copyright	1	0 b	
original_or_copy	1	1 b	
PTS_DTS_flags	2	00 b	no PTS/DTS
ESCR_flag	1	0 b	
ES_rate_flag	1	0 b	
DSM_trick_mode_flag	1	0 b	
additonal_copy_info_flag	1	0 b	
PES_CRC_flag	1	0 b	
PES_extension_flag	1	0 b	
PES_header_data_length	8	0 h	
stream_id extension	8	10 h	data stream number 0
PES_packet_data_byte	800*8		JPEG

Table 10 ITU-T C&I data in PES layer

Syntax	No. of Bits	Codeword bin/hex/dec	Note
packet_start_code_prefix	24	000001 h	
stream_id	8	F4 h	ITU-T Rec.H.222.1 type A (ITU-T C&I)
PES_packet_length	16	0080 h	128
'10'	2	10 b	
PES_scrambling_control	2	00 b	
PES_priority	1	0 b	
data_alignment_indicator	1	0 b	
copyright	1	0 b	
original_or_copy	1	1 b	
PTS_DTS_flags	2	00 / 10 b	10 b; with PTS
ESCR_flag	1	0 b	
ES_rate_flag	1	0 b	
DSM_trick_mode_flag	1	0 b	
additonal_copy_info_flag	1	0 b	
PES_CRC_flag	1	0 b	
PES_extension_flag	1	0 b	
PES_header_data_length	8	0A h	
'0010' *	4	0010 b	
PTS [32..30] *	3	x h	
marker_bit *	1	1 b	
PTS [29..15] *	15	xxxx h	
marker_bit *	1	1 b	
PTS [14..0] *	15	xxxx h	
marker_bit *	1	1 b	
stuffing_byte	0,5*8	FF h	0; with PTS
stream_id extension	8	20 h	ITU-T C&I stream (H.32X) number 0
PES_packet_data_byte	64*8		

* When PTS_DTS_flags is '00 b', these fields don't exist.