RECOMMENDATION ITU-R BT.1209-1*

Service multiplex methods for digital terrestrial television broadcasting

(Question ITU-R 31/6)

(1995-1997)

The ITU Radiocommunication Assembly,

considering

a) that digital terrestrial television broadcasting (DTTB) will be introduced in the VHF/UHF bands by some administrations;

b) that existing television broadcast systems are capable of transmitting video, sound and data services which can include multi-language services, teletext, Power Distribution Centres (PDC), etc.;

c) that digital television broadcasting systems can provide for the simultaneous transmission of video, sound, data and control signals;

d) that any service multiplex method should provide for the optional capability of multiple digital television programme services within an existing channel;

e) that the service multiplex, for example, can be implemented using structured transmission (fixed assigned method), packet transmission (variable assigned method), or a combination of the two;

f) that there are significant advantages in each of the above approaches depending upon the service requirements;

g) that it may be desirable to support the transmission of a hierarchical video service (comprising HDTV, EDTV and SDTV) within a single channel;

h) that it is desirable that the service multiplex be capable of flexible allocation of data to audio, video and data services, consistent with the data capacity required for the video;

j) that multiple digital television programme satellite broadcasting systems are under development in some countries;

k) that it is desirable to seek commonality between systems designed for different channel widths,

recommends

1 that DTTB systems should use the Transport Stream multiplexing methods specified in the International Organization for Standardization/International Electrotechnical Commission Standard 13818-1 (ISO/IEC Standard 13818-1). The outline of the specifications is shown in Annex 1.

---

* Radiocommunication Study Group 6 made editorial amendments to this Recommendation in 2003 in accordance with Resolution ITU-R 44.

Annex 1

1 Introduction

The multiplexing scheme specified by ISO/IEC Standard 13818-1 is based on a fixed packet length transport stream approach. The packet is called the Transport Stream Packet (TSP).

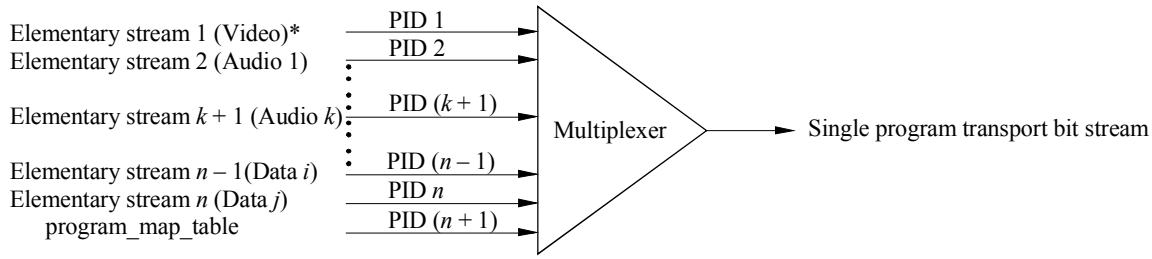
The overall system multiplexing approach can be thought of as a combination of multiplexing at two different layers. In the first layer, single program transport bit streams are formed by multiplexing transport packets from one or more elementary bit streams, and in the second layer, a number of single program transport bit streams are combined to form a system of programs. The function that contains both this program and system level multiplexing information is called the Program Specific Information (PSI).

1.1 Single program transport stream

A single program transport bit stream is formed by multiplexing individual transport packetized elementary bit streams (PESs) sharing a common time-base, and a packetized control bit stream that describes the program. Individual bit streams are identified by their unique packet identifiers (PIDs). The organization of this multiplexing function is illustrated in Fig. 1. The control bit stream contains the program_map_table (PMT) that includes information about the PIDs of the transport streams that make up the program, the identification of the applications that are being transmitted on these bit streams and the relationship among these bit streams.

FIGURE 1

Illustration of the multiplex function to form a single program transport stream



* A single program could also contain multiple video streams.

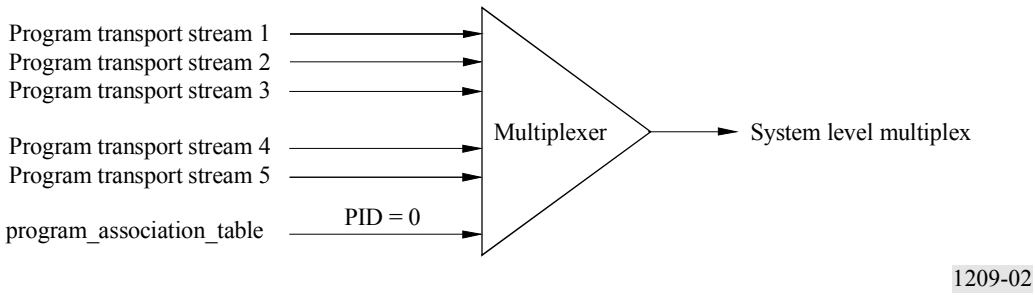
1209-01

1.2 System multiplex

The system layer of multiplexing is illustrated in Fig. 2. In addition to the single program transport bit streams (with the corresponding PIDs) that define the individual programs, a system level control bit stream with PID = 0 is defined. This bit stream carries the program_association_table (PAT) that maps program_identities to the PIDs of the bit streams containing the program_map_table for the particular program.

FIGURE 2

Illustration of the multiplex function to form the system level bit stream



The packet on this system level multiplexing is called the TSP.

The TSP and PES are described in detail in § 2, and the actual ways to multiplex data services are shown in § 3.

2 Packet format

2.1 Transport packet

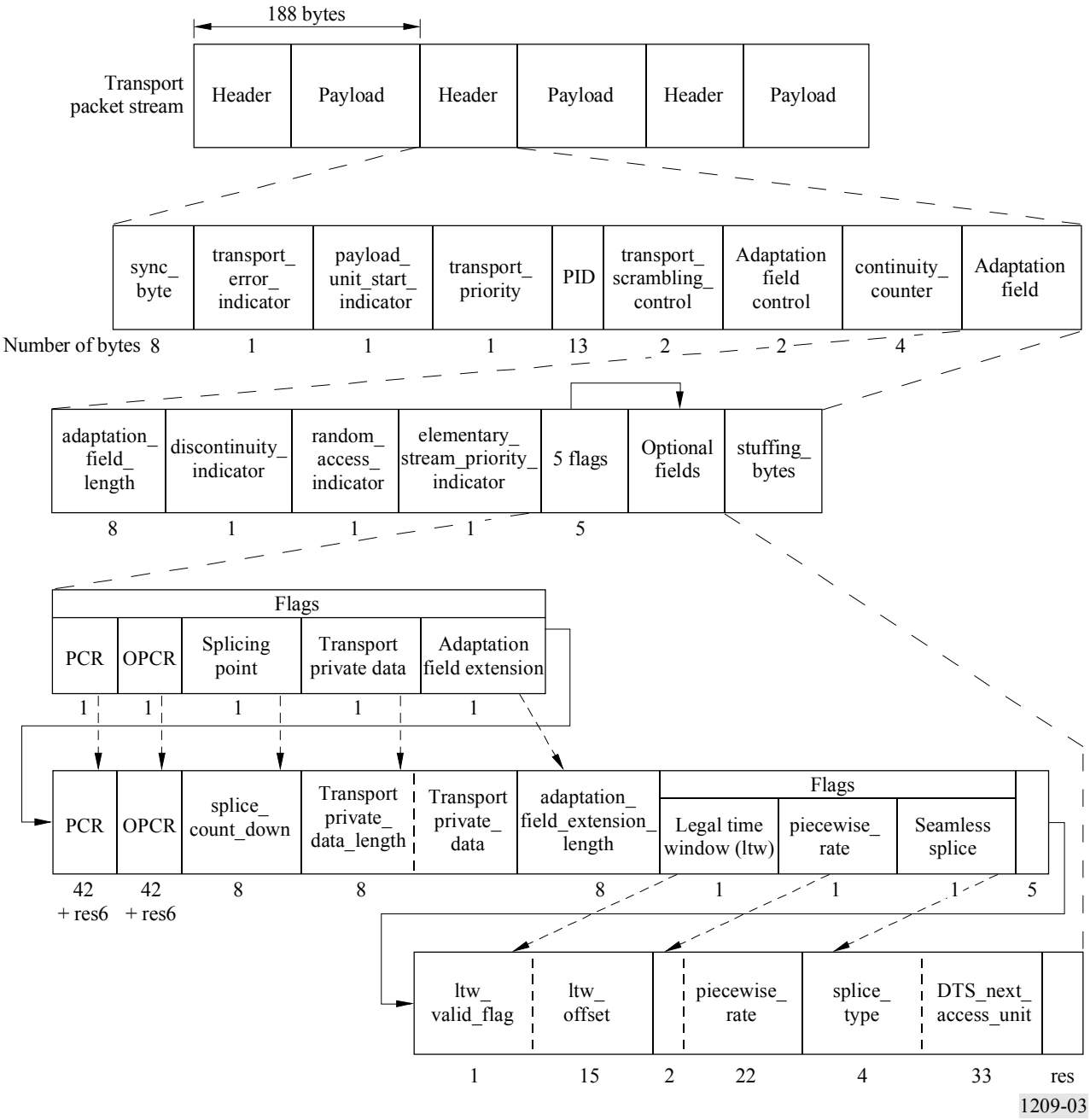
The transport packet format is shown in Fig. 3. Transport packets, 188-byte long, containing program transport streams, are generated and asynchronously multiplexed with a single bit stream channel. Each transport packet carries data belonging to one particular "data source" and these sources are identified from PIDs within the header (which are further specified using information in PSI tables). Data sources are either the applications generating the PESs, or the types of PSI information being transmitted.

2.1.1 Header

sync_byte (8 bits):	it is for packet synchronization, and sent in every packet.
transport_error_indicator (1 bit):	a flag to show if the packet contains error bits or not. This field can be used for error handling.
payload_unit_start_indicator (1 bit):	a flag to show if a PES starts in the packet or the first byte of a PSI section is included in the packet or not.
transport_priority (1 bit):	this flag shows the priority of the packet. It can be used to indicate that the packet was sent repeatedly to increase robustness against transmission error and has less priority.
PID (13 bits):	an identifier for the TSP.
transport_scrambling_control (2 bits):	"00" means "not scrambled" and the others mean "user defined". "10" and "11" can be used as "scrambled with key A" and "scrambled with key B", respectively.
continuity_counter (4 bits):	it shows the order of transmitted packets having the same PID. Error detection is enabled in the decoder through the use of this field.

FIGURE 3

Structure of the transport stream



2.1.2 Adaptation field (variable length)

adaptation_field_length (8 bits): it shows the byte length of the contents of the adaptation field.

discontinuity_indicator (1 bit): it indicates the discontinuity of PCR.

random_access_indicator (1 bit): it indicates that the next PES packet of the same PID contains an elementary stream access point. This is used for the random entry into application bit streams in such case as programme tuning or channel change.

elementary_stream_priority_indicator (1 bit): this flag can be used to indicate that the same elementary stream is sent repeatedly to increase robustness against transmission error and has less priority.

5 flags (5 bits).

2.1.3 Optional fields

program_clock_reference (PCR) (42 bits + 6 bits (for future use)): the reference data for system clock synchronization. It consists of 33 bit field which is expressed by 90 kHz and 9 bit extension field which is expressed by 27 MHz.

original_program_clock_reference (OPCR) (42 bits + 6 bits (for future use)): it assists in the reconstruction of a single TS from another TS.

splice_count_down (8 bits): it indicates the number of TSPs until or after a splicing point. It is necessary to note that when this is used in a high-speed system, the control needs to start just before the discontinuous point (several ms in the HDTV case, for example). This shows the points in the elementary bit streams at which local programme insertion such as commercials is allowed.

private_data_length (8 bits): it shows the byte length of the private data.

private_data: users can use these bit streams for themselves.

adaptation_field_extension_length (8 bits): it shows the byte length of the following optional fields.

3 flags (3 bits).

ltw_valid_flag (1 bit): it indicates that the value of the ltw_offset is valid.

ltw_offset (15 bits): it indicates the displacement in the arrival time of the TSP.

piecewise_rate (22 bits): it specifies the bit rate over all transport packets of this PID measured in units of 50 bytes/s.

splice_type (4 bits): it is used to derive splice_decoding_delay and max_splice_rate from tables in the specifications.

DTS_next_access_unit (33 bits): it indicates the value of the DTS (decoding time stamp) explained in § 2.2.3 of the next access unit of an unspliced or seamless spliced stream.

2.1.4 Stuffing

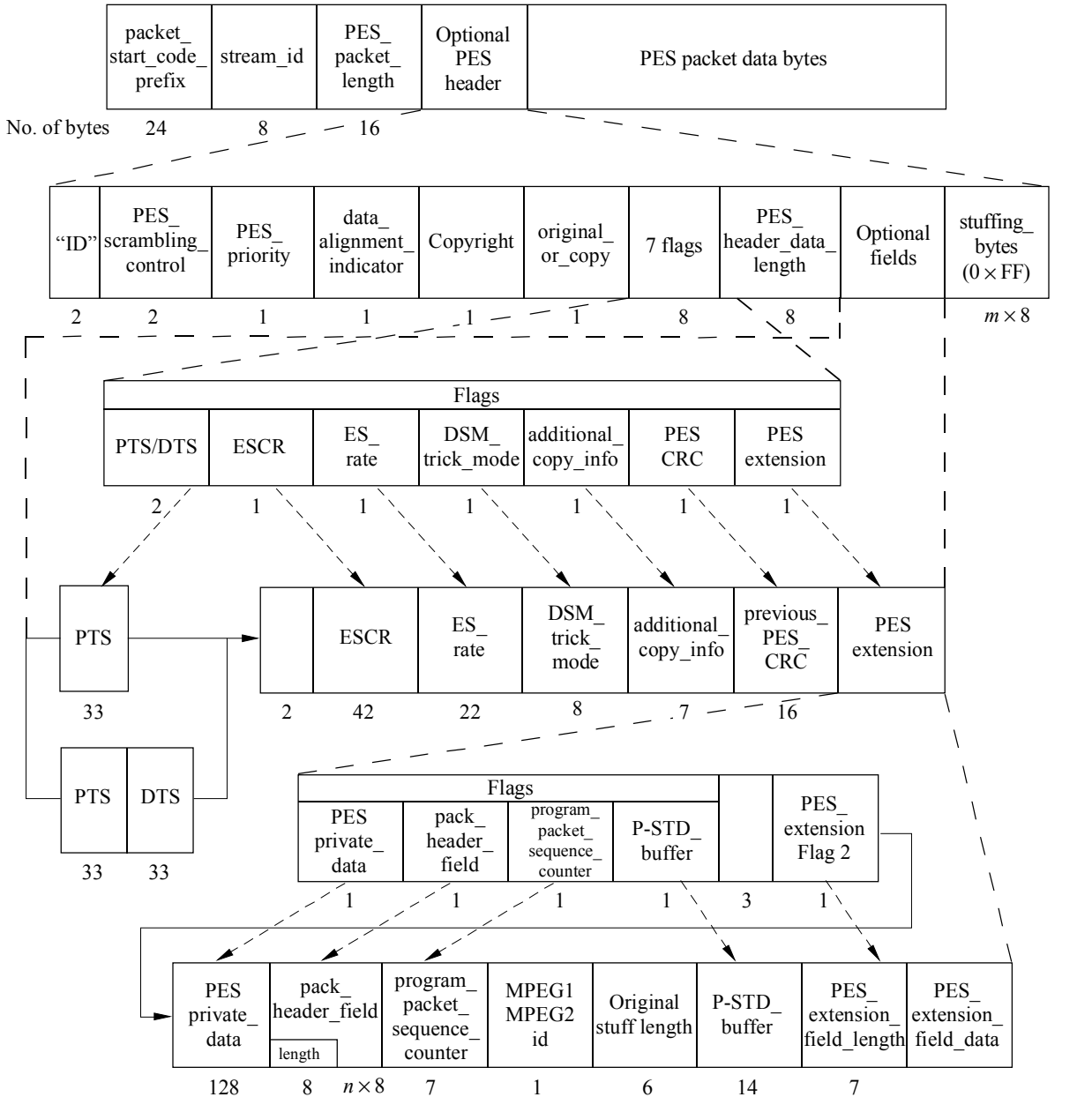
stuffing-bytes (8X bytes): for stuffing.

2.2 PES packet

The structure of the PES packet is shown in Fig. 4. The PES packet consists of the header, and the followed elementary stream of video or audio signals standardized in the Moving Pictures Expert Group (MPEG) or the elementary stream for data services, and its length is variable. Namely, both the header and the content of the packet are variable.

FIGURE 4

Structure of the PES packet



(Marker bits are not shown.)

1209-04

2.2.1 PES header

packet_start_code_prefix (24 bits): a fixed 24-bit pattern indicates the start of the PES packet.

stream_id (8 bits): it is used for identifying each PES. The 4th bit is available for video signals and the 5th bit for audio signals, and the 11 kinds of data service can be identified. Although it is impossible to predict the number of identifications for the future services, the PES_extension_field allows it to increase, and one example is explained in the third section.

PES_packet_length (16 bits): it shows the byte length of the contents of the packet.

2.2.2 Optional PES header

PES_scrambling_control (2 bits): "00" means "not scrambled" and the others mean "user defined".

PES_priority (1 bit): it indicates the priority of the payload of the PES packet. It is not used in the broadcast stream.

data_alignment_indicator (1 bit): it indicates whether the access unit data is aligned after the PES packet header or not.

copyright (1 bit): it indicates whether the material of the associated PES packet payload is copyrighted or not.

original_or_copy (1 bit): it indicates whether the contents of the PES packet payload is an original or copy.

7 flags (8 bits).

PES_header_data_length (8 bits): it specifies the total number of bytes of the optional fields and stuffing bytes contained in this PES packet header.

2.2.3 Optional fields

PTS (presentation_time_stamp) (33 bits): it indicates the intended time of presentation of the presentation unit of the first access unit that commences in the packet. The value of PTS is measured by 90 kHz.

DTS (decoding_time_stamp) (33 bits): it indicates the intended time of decoding of the first access unit that commences in the packet. The value of DTS is measured by 90 kHz.

ESCR (elementary_stream_clock_reference) (42 bits): it is used in a PES stream, and it is not used for broadcasting.

ES_rate (22 bits): it is used in a PES stream, and it is not used for broadcasting.

DSM_trick_mode (8 bits): it is used for digital storage media, and it is not used for broadcasting.

additional_copy_info (7 bits): it contains private data relating to copyright information. It needs to be reserved because there will be various kinds of services in broadcasting.

previous_PES_CRC (16 bits): the CRC for the previous PES packet is sent in the header of the following PES packet which has the same PID value. It is intended for use in network maintenance, and it is not intended for use by elementary stream decoders.

2.2.4 PES_extension

5 flags (5 bits and 3 bits reserved).

2.2.5 Optional field (variable length)

private_data (128 bits): it can be used for future services.

pack-header-field (8 bits): it is used when PS is transmitted by TS and the PS is reconstructed at the receiver, and it is not used for broadcasting.

Program_packet_sequence_counter (8 bits): it can be used as the continuity counter for the PES packets although it is intended to be used in PS.

P-STD_buffer (16 bits): it is used in PS, and it is not used for broadcasting.

PES_extension_field_length (7 bits): it means the byte length of the content of the extension field.

PES_extension_field (variable length): it is used for future extension.

3 Scheme for the multiplexing of various data

Examples of a multiplexing scheme of general purpose data are shown in Appendix 1 to Annex 1. An example for the insertion of teletext into the MPEG-2 transport multiplex is shown in Appendix 2 to Annex 1.

Appendix 1
to Annex 1

Multiplexing of general purpose data for digital terrestrial television broadcasting

1 Scheme for the multiplexing of various data

In digital TV broadcasting, the specifications should allow the system to transmit various kinds of data service not only related but also non-related to the TV programmes.

According to ISO/IEC Standard 13818-1, there are two ways to multiplex data. One is to use the elementary streams and the other is to use the private sections defined in the PSI.

1.1 Data multiplexing by PES-type data group

In this case, the start of each data group is synchronized with the start of the payload of the TSP. The stream_id code is assigned for data services.

This scheme is suitable for the data of presentation media which uses the Presentation Time Stamp (PTS).

When so-called multimedia broadcasting is carried out, various kinds of data such as caption, detailed information, text data for printing, computer software, and their combination, etc., may be sent in a program, and the relationship among them is specified by the PMT. In this case, each data is identified by the stream_id, and it needs to be unique in one program. However, the number of stream_ids which are reserved for data transmission in the ISO/IEC Standard 13818-1 specifications is only 5. Therefore, in the case where more than 5 stream IDs are required, the subdivisions of the stream ID can be used by defining the PES_extension_field.

1.2 Data multiplexing by section-type data group

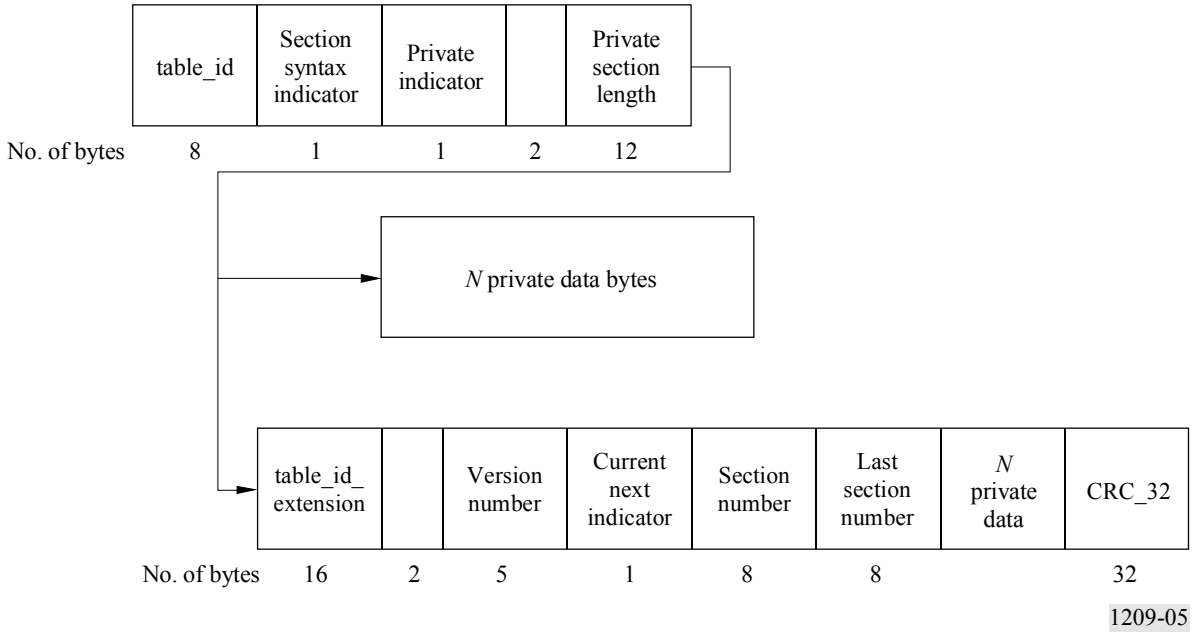
The structure of the private section is as shown in Fig. 5. The identification of each data group in one program is carried out using table_id_extension. (In the PMT, stream_type is assigned as the user private (0x80-0xFF), and the sections which are sent using the packets which have the designated elementary_PID are received. From the received sections, the section which has the designated table_id is extracted.)

In the case of the section-type data group, its start point does not need to be synchronized to the start of the payload of TSP, and the plural number of sections can be transmitted in a TSP. In this case, the start point of the first section is indicated by the pointer which is placed at the beginning of the payload of the TSP, and then the next start point is designated using the section_length.

This scheme is suitable for the transmission of the signals which have a little data, because the plural number of sections can be transmitted in a TSP.

FIGURE 5

Structure of the private section



Appendix 2
to Annex 1

Multiplexing method for teletext

If teletext is contained in the bitstream the following applies:

1 Transport Stream

The standard TSP syntax and semantics are followed noting the following constraints:

adaptation field control: only the values “01” and “10” are permitted.

2 PES header

The standard PES packet syntax and semantics are followed noting the following constraints:

stream_id: set to “1011 1101” (0xBD) (private_stream_1).

PES_packet_length: set to the value (*N*:*184)-6, where *N* is an integer (so that the PES packet finishes at the end of a TSP).

data_alignment_indicator: set to “1” (indicating that the teletext access units are aligned with the PES packets).

PES_header_data_length: set to “0x24”.

stuffing_byte: the PES header is followed by as many stuffing bytes as are required to make up the header data length, so that the entire PES header is 45 bytes long.

PES_packet_data_byte: these bytes are coded in accordance with Annex 1 of Recommendation ITU-R BT.1301.

PTS and other optional fields may be present in the PES header, but the header length is always fixed for streams identified in the PSI by the digital video broadcasting teletext descriptor.

FIGURE 6

Teletext PES header and payload