INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

H.223

Annex C

(02/98)

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS

Infrastructure of audiovisual services – Transmission multiplexing and synchronization

Multiplexing protocol for low bit rate multimedia communication

Annex C: Multiplexing protocol for low bit rate multimedia mobile communication over highly error-phone channels

ITU-T Recommendation H.223 – Annex C

(Previously CCITT Recommendation)

ITU-T H-SERIES RECOMMENDATIONS

AUDIOVISUAL AND MULTIMEDIA SYSTEMS

Characteristics of transmission channels used for other than telephone purposes	H.10–H.19
Use of telephone-type circuits for voice-frequency telegraphy	H.20–H.29
Telephone circuits or cables used for various types of telegraph transmission or simultaneous transmission	H.30–H.39
Telephone-type circuits used for facsimile telegraphy	H.40–H.49
Characteristics of data signals	H.50–H.99
CHARACTERISTICS OF VISUAL TELEPHONE SYSTEMS	H.100–H.199
INFRASTRUCTURE OF AUDIOVISUAL SERVICES	
General	H.200–H.219
Transmission multiplexing and synchronization	**H.220–H.229**
Systems aspects	H.230–H.239
Communication procedures	H.240–H.259
Coding of moving video	H.260–H.279
Related systems aspects	H.280–H.299
Systems and terminal equipment for audiovisual services	H.300–H.399

For further details, please refer to ITU-T List of Recommendations.

ITU-T RECOMMENDATION H.223

MULTIPLEXING PROTOCOL FOR LOW BIT RATE MULTIMEDIA COMMUNICATION

ANNEX C

Multiplexing protocol for low bit rate multimedia mobile communication over highly error-phone channels

Summary

This Annex describes an extension of Recommendation H.223. It defines specific adaptation layers to allow the use of H.324 terminals in highly error-prone transmission environments. These adaptation layers include specific options for H.324 terminals, e.g:

- error detection and correction;
- sequence numbering;
- automatic repeat request;
- retransmission capabilities (hybrid ARQ-type I & II);
- segmentation procedure to transmit frames which shall be transmitted unframed;
- support of framed in unframed transmission mode.

Source

Annex C to ITU-T Recommendation H.223 was prepared by ITU-T Study Group 16 (1997-2000) and was approved under the WTSC Resolution No. 1 procedure on the 6th February 1998.

FOREWORD

ITU (International Telecommunication Union) is the United Nations Specialized Agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the ITU. The ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

INTELLECTUAL PROPERTY RIGHTS

The ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. The ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, the ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

CONTENTS

Recommendation H.223

MULTIPLEXING PROTOCOL FOR LOW BIT RATE MULTIMEDIA COMMUNICATION

ANNEX C

Multiplexing protocol for low bit rate multimedia mobile communication over highly error-prone channels

(Geneva, 1998)

C.1 Scope

This Annex specifies the level 3 protocol of the mobile H.223 extensions as described in Annex C/H.324. Level 3 defines the most error robust scheme of the mobile extensions of Recommendation H.324. This Annex changes both the multiplex layer and the adaptation layer of Recommendation H.223.

C.2 Acronyms and definitions

The following abbreviations are used in this Annex:

ARQ	Automatic Repeat Request
CEC	Control Error Code
CF	Control Header Field
EGolay	Extended Golay (code)
FEC	Forward Error Correction
N(R)	Receive Sequence Number
N(S)	Send Sequence Number
RCPC	Rate Compatible Punctured Convolutional (code)
RN	Retransmission Number
SEBCH	Systematic Extended Bose-Chaudhuri-Hocquenghem (SEBCH) (Code)
SN	Sequence Number
SRC	Systematic Recursive Convolutional (code)
TB	Tail Bit

C.3 Multiplex (MUX) layer specification

This Annex uses nearly the same multiplex layer specifications as defined in Annex B/H.223, except for the stuffing mode of B 3.2.3/H.223.

C.3.1 Stuffing mode

Initially, prior to a dynamic level change, Level 3 stuffing mode shall have a structure identical to the stuffing mode used in Level 2 with MPL field set to "0000 0000". However, the MC field shall be set to "1111". The header may also include the optional header field of Annex B/H.223 (see B.3.2.1).

The stuffing mode may be inserted consecutively an arbitrary number of times. After a level change the terminal may also use the exact stuffing mode of B.3.2.3/H.223.

C.4 Adaptation Layer

C.4.1 AL1M

C.4.1.1 Framework of AL1M

AL1M is a highly flexible adaptation layer designed primarily for transfer of data and control information in highly error-prone environments such as could be encountered in a wireless environment. AL1M supports the use of error detection, Forward Error Correction (FEC), and retransmission (ARQ). AL1M also supports framed and unframed transfer mode.

AL1M provides two transfer modes:

a) framed transfer mode; and

b) unframed transfer mode.

In the framed transfer mode, AL1M may be used to transfer frames generated by a higher-layer protocol such as the data link layer protocol LAPM/V.42 or LAPF/Q.922. In this case, frames are first mapped to AL-SDUs and these are then passed by AL1M in MUX-SDUs to the MUX layer.

AL1M may also be used to carry an unframed octet sequence. In this mode, any internal framing present in the octet sequence is not visible to AL1M which passes the octets received from the higher layer to the MUX layer without paying any attention to framing.

The transfer mode of AL1M is selected by the transmitter in the H.245 OpenLogicalChannel message.

AL1M includes an optional EGolay or SEBCH coded header. AL1M also supports optional sequence numbering, which may be used to detect missing and misdelivered AL-PDUs. AL1M transfers variable-length AL-SDUs. In addition to Recommendation H.223, AL1M also supports the capability that a long framed AL-SDU may be split into several packets and delivered as one AL-SDU to the AL1-user.

C.4.1.2 Primitives exchanged between AL1M and AL1 user

The primitives exchanged are identical to those specified in 7.2.2/H.223, whereby the word AL1 shall be changed to AL1M.

C.4.1.2.1 Description of primitives

The description of primitives are identical to those specified in 7.2.2.1/H.223, whereby the word AL1 shall be changed to AL1M.

C.4.1.2.2 Description of parameters

- AL-SDU: This parameter specifies the unit of information exchanged between AL1M and the AL1 user. Each AL-SDU shall contain an integral number of octets. The length of AL-SDUs may be variable. The octets in an AL-SDU are numbered from 1 to n, and in each octet, bits are numbered from 1 to 8. Bit 1 of the octet 1 is transmitted first. An AL1M receiving entity may deliver an empty AL-SDU to the AL1 user to indicate that an AL-SDU is missing.

- Error Indication (EI): This parameter may be used in the AL1M receiver to pass error indications to the AL1 user. This may also be used if the AL1M receiver entity delivers an empty AL-SDU to the AL1 user. The precise procedures for using this parameter, and its numerical coding, are outside the scope of Recommendation H.223.

C.4.1.3 Functions of AL1M

AL1M provides the following functions:

- optional error detection and indication;
- optional sequence numbering;
- optional forward error correction;
- optional support of retransmission, either via ARQI or ARQII;
- optional AL-SDU splitting for framed frames.

C.4.1.4 Format and structure of AL1M

The format of the AL1M can be seen in Figure C.1.

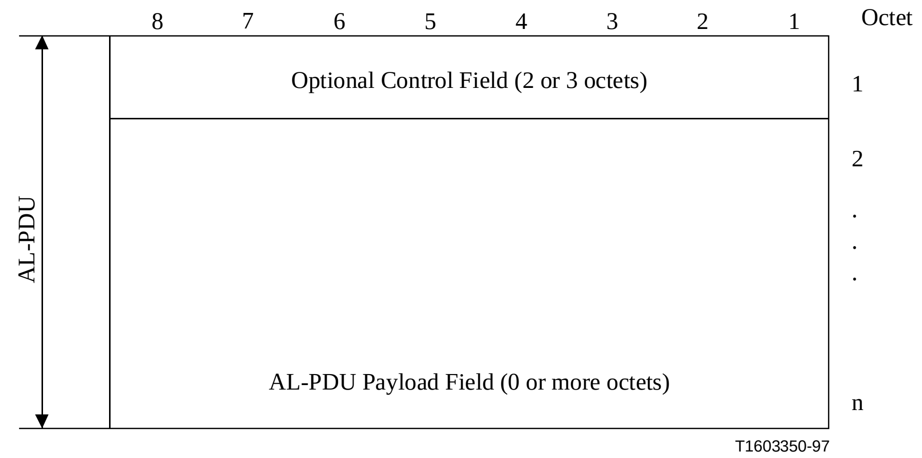

Figure C.1/H.223 – Format of the AL-PDU of the AL1M

The AL-PDU payload shall consist of either an I-PDU or an S-PDU. If an S-PDU is transmitted the length of the AL-PDU payload is 0, otherwise it is an I-PDU. In the following descriptions the AL-PDU payload is related as an I-PDU, if no other explicit explanation is given. The maximum length of AL-PDUs that an AL1M receiver can accept shall be signalled via the H.245 capability exchange.

In contrast to AL1 of Recommendation H.223, the AL-SDU is not always directly mapped to the AL-PDU payload, (see Figure C.2). The application layer (AL1 user) transfers its data through AL-SDUs to the adaptation layer. The adaptation layer forms its own AL-SDUs* from the AL-SDUs. The length of the AL-PDU can be derived from the procedure given in C.4.1.7.1. The AL-PDU is formed by the AL-PDU payload and the optional Control Field (CF). Optional bit interleaving may be applied to the entire AL-PDU.

The error protocol allows the AL1M to operate the following two modes:

– FEC_ONLY: In this mode a AL-SDU* with an CRC is RCPC encoded with a code rate $r \leq 1.0$. The resulting AL-PDU only consists of an AL-PDU payload field. Splitting mode is not supported.

– ARQ: If the mode is set to ARQ (either ARQI or ARQII), it is possible to request retransmissions. The mandatory error detection code (CRC) and the mandatory tail bits (TB)[1] are added to the AL-SDU*. This new field is encoded with the convolutional code of mother rate r = ¼. The encoded data may be filled to a linear buffer[2] according to the puncturing rule. For filling the AL-PDU payload, the octets of the buffer may be read in linear order from this buffer. The first octet of this buffer shall be the first octet of the AL-PDU payload.

  When only ARQI is used, each (re)-transmission shall contain the same encoded data. Therefore, the AL-PDU of each retransmission of the same SN shall contain the identical number of octets.

  Using ARQII, each (re)-transmission can contain different buffer encoded data, which may lead to different lengths for (re)-transmitted AL-PDU payloads. The first transmitted AL-PDU payload shall contain the octets of the buffer by linear reading from the beginning of the buffer. Each retransmission shall transmit data by reading the buffer after the last read octet. If the reading procedure comes to the end of the buffer, the procedure proceeds reading from the beginning of the buffer.

---

[1] The tail bits are required due to the use of the error-correction scheme with convolutional codes. In this case, the TB field has the length of 4 bits.

[2] The buffer scheme is used only for achieving an easy description of the encoding/decoding system. Therefore, this is no description on how to implement the system.

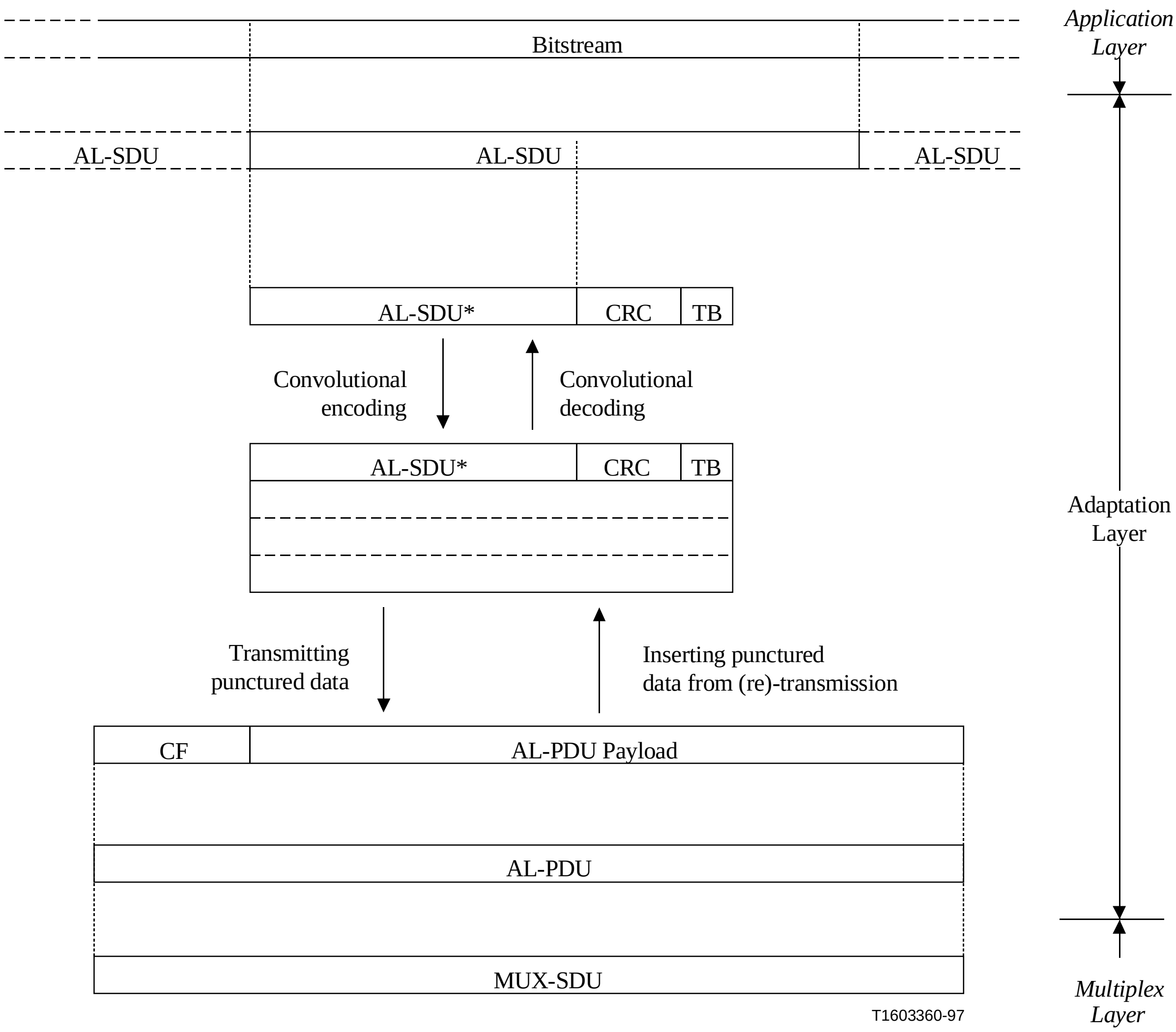

Figure C.2/H.223 – AL1M structure

C.4.1.5 Control Field (CF)

The optional control field consists of the Sequence Number (SN) field, the Retransmission Number (RN) field, the 1-bit field (X), and of the Control Error Code (CEC) field. The CEC uses either SEBCH or EGolay code, as illustrated in Figure C.3. These codes provide error detection and correction capability to the SN, RN and X fields.

8	7	6	5	4	3	2	1	Octet
P1	X	RN	SN5	SN4	SN3	SN2	SN1	1
P9	P8	P7	P6	P5	P4	P3	P2	2

Figure C.3/H.223 – Control field format of the AL-PDU for AL1M with SN = 5 and SEBCH code

8	7	6	5	4	3	2	1	Octet
SN8	SN7	SN6	SN5	SN4	SN3	SN2	SN1	1
P4	P3	P2	P1	X	RN	SN10	SN9	2
P12	P11	P10	P9	P8	P7	P6	P5	3

Figure C.4/H.223 – Control field format of the AL-PDU for AL1M with SN = 12 and EGolay code

NOTE – The bit order of the fields in Figures C.3 and C.4 is not in accordance to the general convention of Recommendation H.223.

Depending on the code used in the CEC field, the length of the SN field may vary, as indicated in Table C.1. When the control field is absent, the retransmission procedure is not used.

Table C.1/H.223 – Length of SN field according to different CECs

CEC	Length of SN field	Reference to
SEBCH(16,7,6)	5	Table C.6
EGolay(24,12,8)	10	B.3.2.1.3/H.223

C.4.1.5.1 Sequence Number (SN) field

The sequence number field shall be 5 or 10 bits, depending on the chosen CEC code. The SN field shall contain a send sequence number, N(S), except for in the case of SREJ messages. In this case it shall contain a receive sequence number, N(R).

When the SN field is in use, the AL1M receiver may detect that an AL-PDU is missing or has been misdelivered by the MUX layer. The AL1M receiver should discard any detected misdelivered AL-PDUs.

C.4.1.5.2 RN field

For an S-PDU in the reverse channel (SREJ message), the RN field shall contain the modulo 2 equivalent of the receive Retransmission Number (RN). Otherwise, this field is set to "0".

For an I-PDU frame, this field shall be used to signal the last transmitted packet resulting from the splitting of an AL-SDU into several AL-SDUs*. This shall only be done in a framed transfer mode. The splitting mode is described in C.4.1.6.

C.4.1.5.3 X field

For an S-PDU, the X field shall indicate either an SREJ or a DRTX message (see Table C.2).

Table C.2/H.223 – Definition of supervisory messages

Message	Bit value in X field
Selective reject (SREJ)	1
Declined retransmission (DRTX)	0

For an I-PDU, the X field shall be used as indication of the length of the AL-SDU* field. The X field shall be the modulo 2 equivalent of the number of octets within an AL-SDU*. If the AL-SDU* contains an odd number of octets, X = "1"; otherwise, X = "0".

C.4.1.5.4 Control Error Code (CEC) field

The CEC field defined by the parity bits P in Figures C.3 and C.4 provides error detection and/or error correction capability.

NOTE – The bit order of the fields in Figures C.3 and C.4 is not in accordance to the general convention of Recommendation H.223.

Depending on the code used in the CEC field, the length of the SN field may vary, as indicated in Table C.1. When the control field is absent, the retransmission procedure is not used.

The definition of the EGolay code shall be the same as described in B.3.2.1.3/H.223, whereby the CEC shall be derived by the following equation:

$$\begin{bmatrix} P1 \\ P2 \\ P3 \\ P4 \\ P5 \\ P6 \\ P7 \\ P8 \\ P9 \\ P10 \\ P11 \\ P12 \end{bmatrix} = \begin{bmatrix} 1\,0\,1\,0\,1\,1\,1\,0\,0\,0\,1\,1 \\ 1\,1\,1\,1\,1\,0\,0\,1\,0\,0\,1\,0 \\ 1\,1\,0\,1\,0\,0\,1\,0\,1\,0\,1\,1 \\ 1\,1\,0\,0\,0\,1\,1\,1\,0\,1\,1\,0 \\ 1\,1\,0\,0\,1\,1\,0\,1\,1\,0\,0\,1 \\ 0\,1\,1\,0\,0\,1\,1\,0\,1\,1\,0\,1 \\ 0\,0\,1\,1\,0\,0\,1\,1\,0\,1\,1\,1 \\ 1\,0\,1\,1\,0\,1\,1\,1\,1\,0\,0\,0 \\ 0\,1\,0\,1\,1\,0\,1\,1\,1\,1\,0\,0 \\ 0\,0\,1\,0\,1\,1\,0\,1\,1\,1\,1\,0 \\ 1\,0\,1\,1\,1\,0\,0\,0\,1\,1\,0\,1 \\ 0\,1\,0\,1\,1\,1\,0\,0\,0\,1\,1\,1 \end{bmatrix}^{T} \bullet \begin{bmatrix} SN1 \\ SN2 \\ SN3 \\ SN4 \\ SN5 \\ SN6 \\ SN7 \\ SN8 \\ SN9 \\ SN10 \\ RN \\ X \end{bmatrix}$$

NOTE – The symbol T denotes matrix transposition.

The CEC bits of SEBCH code shall be derived by the following equation:

$$\begin{bmatrix} P1 \\ P2 \\ P3 \\ P4 \\ P5 \\ P6 \\ P7 \\ P8 \\ P9 \end{bmatrix} = \begin{bmatrix} 1\,0\,0\,0\,1\,0\,1\,1\,1 \\ 1\,1\,0\,0\,1\,1\,1\,0\,0 \\ 0\,1\,1\,0\,0\,1\,1\,1\,0 \\ 1\,0\,1\,1\,1\,0\,0\,0\,1 \\ 0\,1\,0\,1\,1\,1\,0\,0\,1 \\ 0\,0\,1\,0\,1\,1\,1\,0\,1 \\ 0\,0\,0\,1\,0\,1\,1\,1\,1 \end{bmatrix}^{T} \bullet \begin{bmatrix} SN1 \\ SN2 \\ SN3 \\ SN4 \\ SN5 \\ RN \\ X \end{bmatrix}$$

ALlM receivers conforming to Recommendation H.223 shall be capable of receiving and correctly interpreting AL-PDUs with these two different CECs. The actual CEC in the control field is determined by the transmitter and shall be signalled to the far end in the OpenLogicalChannel message of Recommendation H.245.

C.4.1.6 Procedures for splitting an AL-SDU (splitting mode)

Only in framed transfer the adaptation layer may split the AL-SDU into one or several AL-SDUs*, if the use of this splitting procedure is signalled by the OpenLogicalChannel message. This procedure is mandatory for the receiver.

Each AL-SDU* is transmitted as described in C.4.1.7. To identify the end of an AL-SDU, the last AL-SDU* of the AL-SDU shall be marked by setting the RN field to logical "1"; otherwise, the RN field shall be set to "0".

C.4.1.7 Procedures for encoding and decoding the AL-PDU payload

The payload field contains either a complete I-PDU or an S-PDU. The I-PDU is an octet-aligned field, which shall consist of one or more octets of encoded data. The S-PDU is only used if the retransmission capability was exchanged. The S-PDU is a 0-octet AL-PDU payload field. According to the direction of the S-PDU, either forward or reverse channel (see C.4.1.13.2) this 0-octet payload field represents different messages.

The transmitter shall construct an AL-PDU such that its size does not exceed the maximum AL-PDU size that the AL1M receiver can accept. This AL-PDU size is signalled in the H.245 capability exchange.

C.4.1.7.1 Evaluation of the AL-PDU (I-PDU) length

The following parameters are given:

- l_v Length of AL-PDU in bits;
- t Length of AL-SDU* in bits;
- r_{target} Code rate of the rate compatible punctured convolutional (RCPC) code;
- l_h Length of the control header (CF) field in bits;
- l_{CRC} Length of the cyclic redundancy check (CRC) field in bits;
- l_{TB} Number of tail bits of the RCPC code.

The length l_v of the AL-PDU can be evaluated by the following equation:

$$l_v = \min_{\lambda \in \Im, \lambda \bmod 8 = 0} \left\{ \lambda \geq l_h + \left\lceil \frac{\left(t + \left(l_{CRC} + l_{TB}\right)\right)}{r_{target}} \right\rceil \right\}, \qquad \text{with } \Im \text{ all integers} \tag{C-1}$$

The parameters l_v, t and (l_{CRC}+l_{TB}) shall be integer number of octets. However, equation C-1 only guarantees that the resulting coding rate r_{result} is equal or smaller than the original rate r_{target}. Equation C-1 shall be used by the AL1M transmitter. At the AL1M receiver, the length of the AL-SDU* t shall be evaluated by the following equation:

$$t = \max_{\tau \in \Im, \bmod 8 = 0} \left\{ \tau \leq \left\lfloor \left(l_v - l_h\right) \right\rfloor \cdot r_{target} \right\} - 1_{CRC} - 1_{TB}, \qquad \text{with } \Im \text{ all integers} \tag{C-2}$$

Both equations shall be calculated in octets, as illustrated by the following example:

Example:

The AL1M wants to transmit a AL-SDU* of t = 376 bits (47 octets), r_{target} = 8/10, l_h = 24 bits (3 octets), l_{CRC} = 20 bits, l_{TB} = 4 bits. Using the equation C-1, the length of the AL-PDU is l_v = 66 octets. The parameter r_{result} can be evaluated by:

$$r_{result} = \frac{t + (l_{CRC} + l_{TB})}{l_v - l_h} \leq r_{target}. \tag{C-3}$$

In this example:

$$r_{result} = \frac{50}{63} \approx 0.794 \leq r_{target} = 0.800.$$

C.4.1.7.2 Cyclic Redundancy Check (CRC)

The CRC provides error detection capability across the entire AL-SDU*. The CRC is appended to the AL-PDU before the error correction coding procedure is done. The CRC is used by the AL1M receiver to verify whether the decoding procedure of the error correction algorithm is error-free. CRC lengths of 4, 12, 20 and 28 bits are supported. The length of the CRC field shall be specified during the H.245 OpenLogicalChannel procedure. The evaluation of the CRC shall be performed by the same procedure as described as in 7.3.3.2.3/H.223.

Description of the CRC polynomials:

a) 4-bit CRC: $x^4 + x^3 + x^2 + 1$;

b) 12-bit CRC: $x^{12} + x^{11} + x^3 + x^2 + x + 1$;

c) 20-bit CRC: $x^{20} + x^{19} + x^6 + x^5 + x^3 + 1$;

d) 28-bit CRC: $x^{28} + x^{27} + x^6 + x^5 + x^3 + 1$.

C.4.1.7.3 Systematic convolutional encoder

The channel encoder is based on a Systematic Recursive Convolutional (SRC) encoder with rate R = ¼. With the puncturing procedure described in Table 4 we obtain a Rate Compatible Punctured Convolutional (RCPC) code. At the AL1M sending unit, the AL-PDU payload is generated by convolutional encoding of the concatenated field of the AL-SDU* and CRC field. The convolutional encoding of the CRC field starts with the highest term of the polynomial representing the CRC field. At the AL1M receiving entity, the concatenation of AL-SDU* and CRC field may be reconstructed by convolutional decoding, for example Viterbi decoding. As this code is systematic the receiver may also directly extract the CRC protected AL-SDU* from the received bitstream without convolutional decoding.

The SRC code is generated from a rational generator matrix by using a feedback loop. A shift register realization of the encoder is shown in Figure C.5.

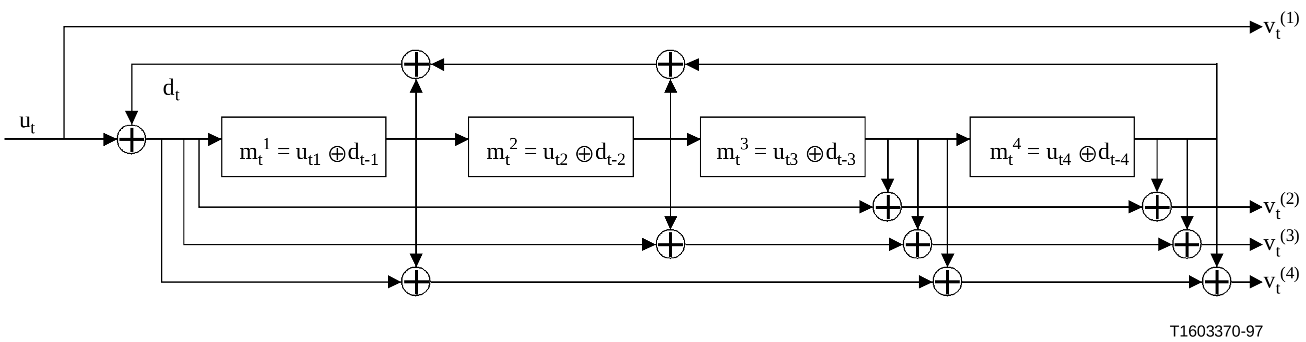

Figure C.5/H.223 – Shift register realization for systematic recursive convolutional encoder

To obtain the output vectors $\mathbf{v}_t$ at each time instant t, one has to know the content of the shift registers m_t^1, m_t^2, m_t^3, m_t^4 (corresponds to the state) and the input bit u_t at time t.

We obtain the outputs $v_t^{(2)}$, $v_t^{(3)}$ and $v_t^{(4)}$:

$$v_t^{(2)} = m_t^4 \oplus m_t^3 \oplus (u_t \oplus d_t)$$
$$v_t^{(3)} = m_t^4 \oplus m_t^3 \oplus m_t^2 \oplus (u_t \oplus d_t)$$
$$v_t^{(4)} = m_t^4 \oplus m_t^3 \oplus m_t^1 \oplus (u_t \oplus d_t)$$

with

$$d_t = m_t^4 \oplus m_t^2 \oplus m_t^1,\ m_t^4 = u_{t-4} \oplus d_{t-4},\ m_t^3 = u_{t-3} \oplus d_{t-3},\ m_t^2 = u_{t-2} \oplus d_{t-2},\ m_t^1 = u_{t-1} \oplus d_{t-1}$$

Finally we obtain for the output vector $\underline{\boldsymbol{v}}_t = (v_t^{(1)}, v_t^{(2)}, v_t^{(3)}, v_t^{(4)})$ at time t depending on the input bit u_t and the current state $\underline{\boldsymbol{m}}_t = (m_t^1, m_t^2, m_t^3, m_t^4)$:

$$v_t^{(1)} = u_t$$
$$v_t^{(2)} = m_t^4 \oplus m_t^3 \oplus (u_t \oplus d_t) = m_t^3 \oplus m_t^2 \oplus m_t^1 \oplus u_t$$
$$v_t^{(3)} = m_t^4 \oplus m_t^3 \oplus m_t^2 \oplus (u_t \oplus d_t) = m_t^3 \oplus m_t^1 \oplus u_t$$
$$v_t^{(4)} = m_t^4 \oplus m_t^3 \oplus m_t^1 \oplus (u_t \oplus d_t) = m_t^3 \oplus m_t^2 \oplus u_t$$

with $\underline{\boldsymbol{m}}_1 = (m_1^1, m_1^2, m_1^3, m_1^4) = (0, 0, 0, 0) = \underline{\boldsymbol{0}}$

The initial state shall always be $\underline{\boldsymbol{0}}$, i.e. each memory cell contains a *0* before the input of the first information bit u_t. The tail bits following the information sequence $\boldsymbol{u}$ for returning to state $\underline{\boldsymbol{m}}_n = \underline{\boldsymbol{0}}$ (termination) depends on the last state $\underline{\boldsymbol{m}}_{n-3}$ (state after the input of the last information bit u_{n-4}.). The termination sequence for each state described by $\underline{\boldsymbol{m}}_{n-3}$ is given in Table C.3. The receiver may use these Tail Bits (TB) for additional error detection.

The appendix $(u_{n-3}, u_{n-2}, u_{n-1}\ u_n)$ to the information sequence can be calculated with the following condition:

For all t with $n\text{-}3 \le t \le n$: $u_t \oplus d_t = 0$.

Hence we obtain for the tail bit vector $\underline{\boldsymbol{u}}' = (u_{n-3}, u_{n-2}, u_{n-1}, u_n)$ depending on the state $\underline{\boldsymbol{m}}_{n-3} = (m_{n-3}^1, m_{n-3}^2, m_{n-3}^3, m_{n-3}^4)$.

$$u_{n-3} = d_{n-3} = m_{n-3}^4 \oplus m_{n-3}^2 \oplus m_{n-3}^1$$
$$u_{n-2} = d_{n-2} = m_{n-2}^4 \oplus m_{n-2}^2 \oplus m_{n-2}^1 = m_{n-3}^3 \oplus m_{n-3}^1 \oplus 0 = m_{n-3}^3 \oplus m_{n-3}^1$$
$$u_{n-1} = d_{n-1} = m_{n-1}^4 \oplus m_{n-1}^3 \oplus m_{n-1}^2 = m_{n-3}^2 \oplus 0 \oplus 0 = m_{n-3}^2$$
$$u_n = d_n = m_{n-3}^1 \oplus 0 \oplus 0 = m_{n-3}^1$$

Table C.3/H.223 – Tail bits for systematic recursive convolutional code

State $\underline{m}_{n-3}$	m_{n-3}^4	m_{n-3}^3	m_{n-3}^2	m_{n-3}^1	u_{n-3}	u_{n-2}	u_{n-1}	u_n
0	0	0	0	0	0	0	0	0
1	0	0	0	1	1	1	0	1
2	0	0	1	0	1	0	1	0
3	0	0	1	1	0	1	1	1
4	0	1	0	0	0	1	0	0
5	0	1	0	1	1	0	0	1

Table C.3/H.223 – Tail bits for systematic recursive convolutional code *(concluded)*

State $\underline{m}_{n-3}$	m_{n-3}^4	m_{n-3}^3	m_{n-3}^2	m_{n-3}^1	u_{n-3}	u_{n-2}	u_{n-1}	u_n
6	0	1	1	0	1	1	1	0
7	0	1	1	1	0	0	1	1
8	1	0	0	0	1	0	0	0
9	1	0	0	1	0	1	0	1
10	1	0	1	0	0	0	1	0
11	1	0	1	1	1	1	1	1
12	1	1	0	0	1	1	0	0
13	1	1	0	1	0	0	0	1
14	1	1	1	0	0	1	1	0
15	1	1	1	1	1	0	1	1

C.4.1.7.4 Puncturing tables

Puncturing of the output of the SRC encoder allows different rates for transmission. The puncturing tables are listed in Table C.4. As all rates include all bits of all lower rates, this code is rate compatible.

Table C.4/H.223 – Puncturing tables (all values in hexadecimal representation)

Rate r	**8/8**	**8/9**	**8/10**	**8/11**	**8/12**	**8/13**	**8/14**	**8/15**	**8/16**	**8/17**	**8/18**	**8/19**	**8/20**
$P_r(0)$	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
$P_r(1)$	00	80	88	A8	AA	EA	EE	FE	FF	FF	FF	FF	FF
$P_r(2)$	00	00	00	00	00	00	00	00	00	80	88	A8	AA
$P_r(3)$	00	00	00	00	00	00	00	00	00	00	00	00	00

Rate r	**8/21**	**8/22**	**8/23**	**8/24**	**8/25**	**8/26**	**8/27**	**8/28**	**8/29**	**8/30**	**8/31**	**8/32**
$P_r(0)$	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
$P_r(1)$	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
$P_r(2)$	EA	EE	FE	FF	FF	FF	FF	FF	FF	FF	FF	FF
$P_r(3)$	00	00	00	00	80	88	A8	AA	EA	EE	FE	FF

C.4.1.8 Interleaving

For some channels, block interleaving may be used.

If interleaving is used, it shall be a applied to the entire AL-PDU including the control field. As the length of the AL-PDU varies, the dimension of the block interleaver matrix has to be recalculated for each length. Given a AL-PDU of length l_v, the dimensions of the interleaver can be calculated:

$$a = \max_{\alpha \in \Im, l_v \bmod \alpha = 0} \left\{ \alpha \le \sqrt{l_v} \right\}, \text{ with } \Im \text{ all integers}$$

$$b = l_v / a$$

b describes the distance between two before interleaving consecutive bits after interleaving. As the AL-PDU is in octets, the minimum b is 8.

The receiver shall calculate the dimensions of the interleaver with the upper equation and the length of the received AL-PDU l_v. De-interleaving shall also be applied to the entire AL-PDU.

C.4.1.9 Encoding procedure: AL-SDU* (I-PDU) to AL-PDU

The following steps are necessary to obtain an AL-PDU from an AL-SDU*.

1) Calculate the length of the AL-PDU payload l_p according to C.4.1.7.1 and the first rate required in the H.245 OpenLogicalChannel message.
2) The CRC of the length required in the H.245 OpenLogicalChannel message field shall be added.
3) Due to the of convolutional codes of mother rate r = ¼ with memory 4, four Tail Bits (TB) shall be appended from Table C.3.
4) Generate the encoded data by passing them through the convolutional encoder.
5) According to the puncturing rules of Table C.4, fill the bits of the convolutional encoder output into a linear buffer. Put AL-SDU* with appended CRC and TB at the beginning of this buffer.
6) For the first transmission read l_p (AL-PDU payload length) bits from the buffer, starting from the beginning of the buffer, fill these bits into the AL-PDU payload field. The first octet of the buffer is the first octet of the AL-PDU payload field.
7) If required in the H.245 OpenLogicalChannel message, the Control Field (CF) shall be added at the beginning of the AL-PDU.
8) Interleaving according to C.4.1.8 shall be applied for the entire AL-PDU, if this has been required in the H.245 OpenLogicalChannel message.

These steps are valid for the modes FEC_ONLY, ARQI and ARQII. If FEC_ONLY is used, no retransmission is possible.

Using the ARQ mode, the content of the AL-PDU varies for the retransmissions:

– ARQI: In this mode, the content of each (re)-transmitted AL-PDU is the same and has the same length.
– ARQII: The transmitting entity shall first transmit the first code rate according to the H.245 OpenLogicalChannel message and may choose any AL-PDU payload length for following incremental retransmissions.

  However if the mother code rate is reached, the transmitter begins transmitting at the beginning of the linear buffer and is still free to choose the code rate, if the maximum number of retransmissions is not reached.

Figure C.6 illustrates the encoding procedures of the ALM at the transmit side.

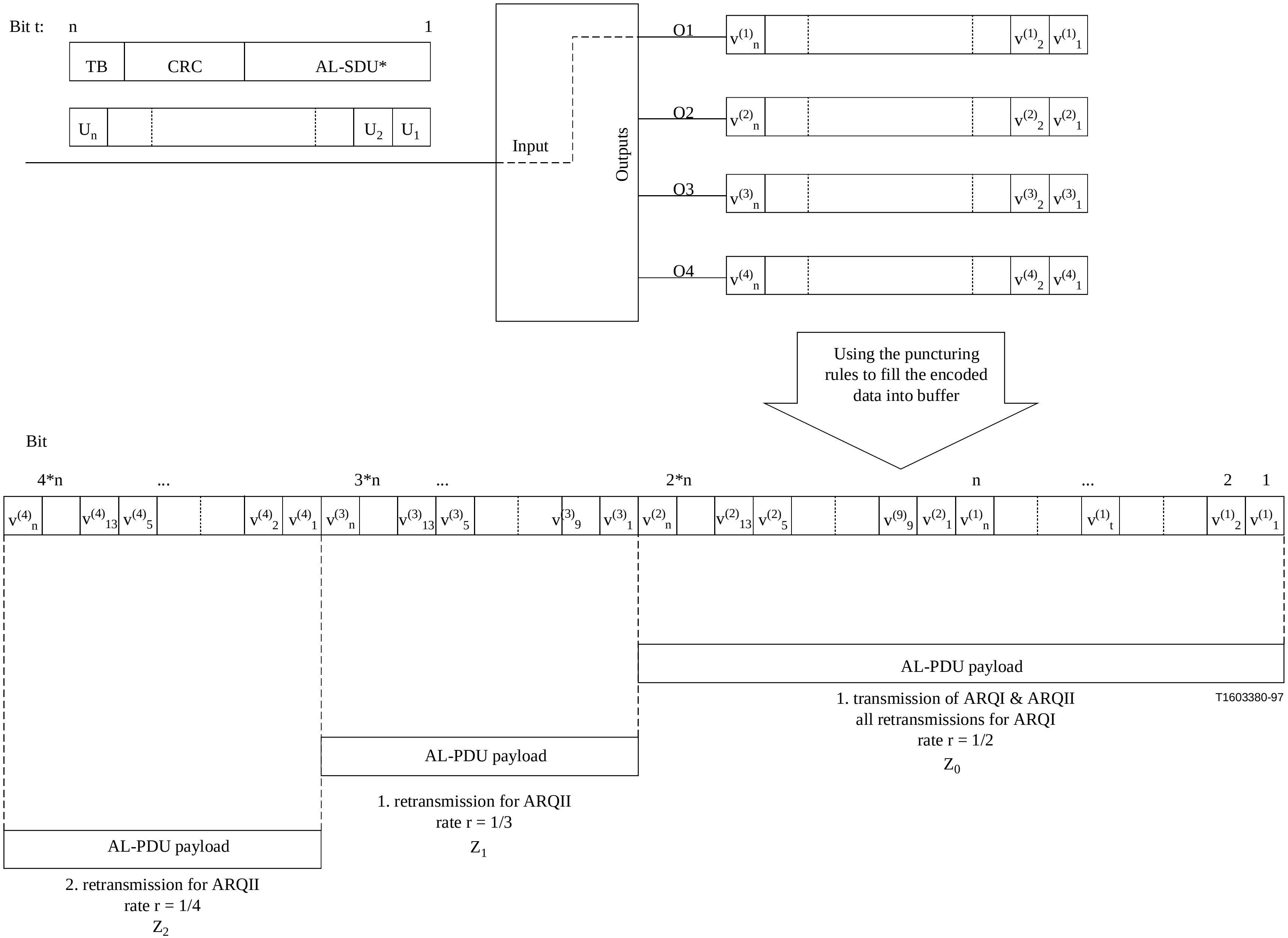

Figure C.6/H.223 – Encoding procedure of the AL3M at the transmitter side

Example:

The following example demonstrates how to interpret the puncturing tables and how to fill the encoded data into the linear buffer.

The following parameters are given:

- l_{CRC} = 4 bits;
- l_{TB} = 4 bits;
- t = 8 bits;
- l_{buffer} = 4*16 bits = 64 bits.

The convolutional encoder gives four output streams. Each of these output streams consists of 16 bits. The numbering is equivalent to that of Figure C.6. The output of line 1, i.e. the systematic bits are directly transferred to the linear buffer. All bits of output 2, 3 and 4 are filled into a temporary matrix, which is used to give a simple example description. The bits are read line by line from the convolutional encoder output and are written from column 1 to 8 and then row by row into the temporary matrix, as shown Figure C.7. Then this matrix is mapped using the puncturing rule and appended to the linear buffer. The puncturing rule describes in which order the columns are read. To achieve a code rate of r = 1/3, the first 48 bits are transmitted to the AL-PDU payload by reading the bits from the start position of the buffer (right bottom in Figure C.6). Another code rate r = 8/13 requires that 26 bits be read from the matrix. However, 32 bits shall be transmitted to obtain a octet-aligned AL-PDU payload.

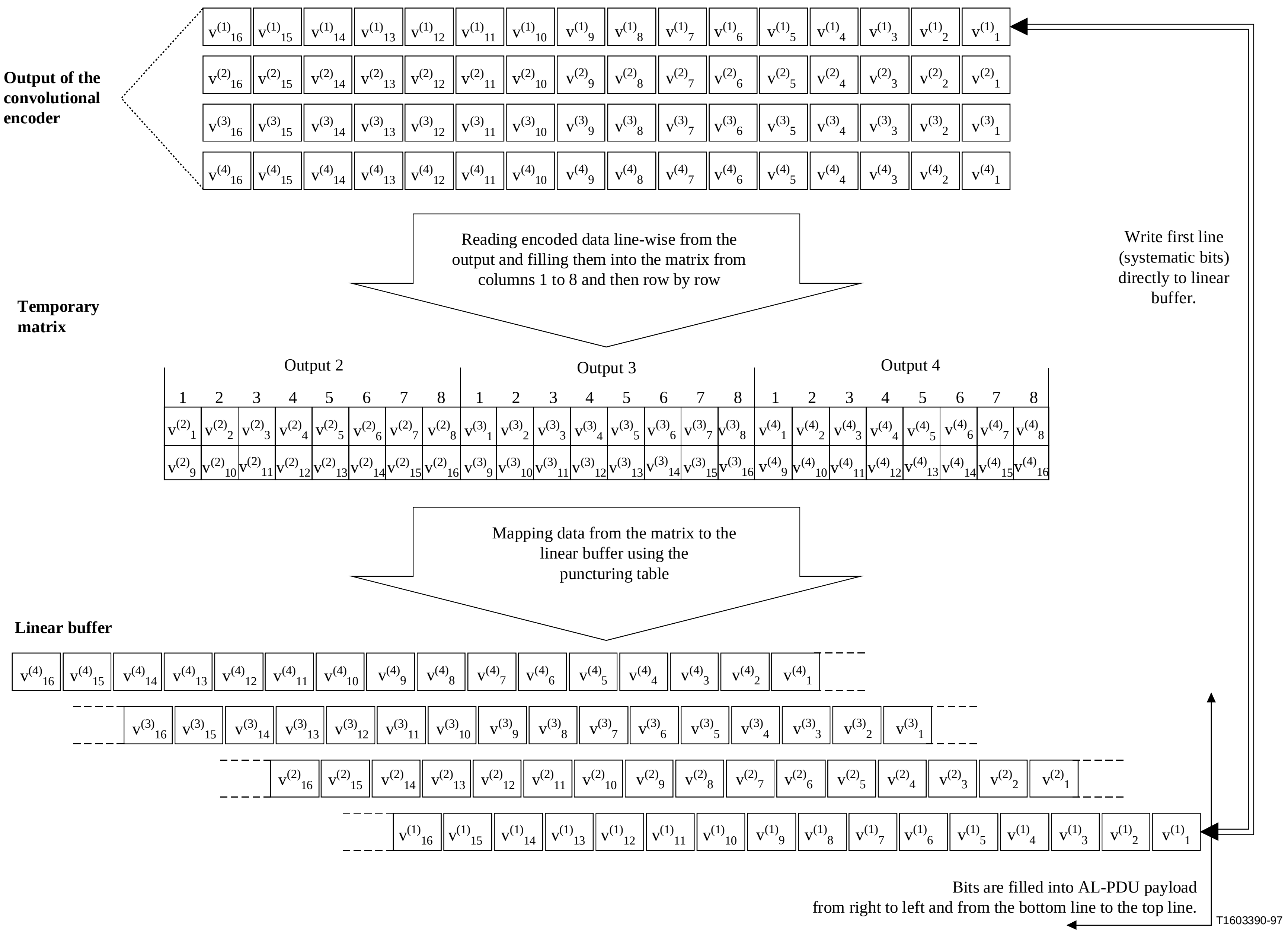

Figure C.7/H.223 – Ordering of the bit streams of the encoded data

C.4.1.10 Decoding of the AL-PDU payload (I-PDU)

The receiver may check the received systematic bits before decoding the convolutional code. It may also use the tail bits for error detecting. If CRC or TB check fails, any kind of convolutional decoding may be used.

After convolutional decoding, the CRC should be used to check the correctness of the decoding attempt again. If the CRC fails, another retransmission may be requested, or the wrong data may be given to the AL1 user with an appropriate error indication (EI) message. If only wrong data is available, the receiver may use the decoded information bits or the systematic bits before decoding as received AL-SDU*.

If ARQI retransmission procedure is used, each retransmission gives the same data as the previous one. If the ARQII procedure is applied, each retransmission is delivering new data which may be combined with the previous received data to achieve a more powerful error correction code. After each decoding attempt, the decoding result should be checked by the CRC.

C.4.1.11 Procedures for Abort

This primitive is discarded and no action is undertaken.

C.4.1.12 Procedures for error control

C.4.1.12.1 Invalid AL-PDUs

An invalid AL-PDU occurs in either of the following cases:

a) The associated AL-PDU has fewer than the minimum number of octets specified in C.4.1.4.

b) The AL-PDU does not contain an integral number of octets.

c) The AL-PDU is longer than the maximum AL-PDU size.

d) The AL-SDU* does not contain an integral number of octets.

An AL-PDU which is not invalid is referred to as a valid AL-PDU.

C.4.1.12.2 Errored AL-PDUs

An errored AL-PDU occurs at the AL1M receiver in the following case:

– The AL-PDU is valid and the associated error decoded AL-PDU Payload contains a CRC error.

An AL-PDU which is valid and not errored is referred to as an error-free AL-PDU.

C.4.1.12.3 Error control: CF absent

In case of a CRC failure at the AL1M receiver when the Control Field (CF) is absent and CRC error detection is used for the AL-PDU Payload, the associated AL-PDU Payload shall be delivered to the AL1 user, together with an appropriate EI parameter, via the AL-DATA.indication primitive.

C.4.1.12.4 Error control: Forward control field present

When the CF is present, the AL1M receiver has the option of invoking the retransmission procedures ARQI or ARQII. Which of these are used shall be indicated by the AL1M sending entity in the H.245 OpenLogicalChannel message. The AL1M sending entity shall respond to a retransmission request according to the procedures defined in C.4.1.13. The error control procedures for retransmission are described in C.4.1.13.8.

When the CF is in use and the AL1M receiver does not invoke the retransmission procedure, the procedure described in 7.4.5.3.1/H.223 should be followed by using AL1M instead of AL3.

C.4.1.13 Retransmission procedures (ARQI, ARQII)

This subclause addresses the two retransmission procedures ARQI and ARQII. The transmitter procedures defined in this subclause shall be used when the control field is present. The receiver procedures defined in this subclause shall be used when retransmission is used.

C.4.1.13.1 Definitions

a) *Modulo*

Each AL-PDU Payload is sequentially numbered modulo 2^5 or 2^{10} and may have the value 0 through 2^5 or 2^{10}. The length of the sequence number field (SN) is set with the OpenLogicalChannel message of Recommendation H.245.

NOTE – All arithmetic operations on state variables and sequence numbers contained in this subclause are modulo 2^5 or 2^{10}.

b) *Send sequence variable V(S)*

V(S) is an internal variable of the transmitting AL1M entity. It denotes the sequence number of the next AL-PDU Payload to be transmitted to the far end. V(S) can take on the values 0 through 2^5 or 2^{10}. The value of V(S) shall be incremented by 1 after each in-sequence AL-PDU is passed to the MUX layer in a MUX-SDU.

c) *Send retransmission variables $V^j(S)$*

$V^j(S)$ are internal variables of the transmitting AL1M entity. A separate counter $V^j(S)$ exists for each possible value j of V(S). $V^j(S)$ can take on the values 0 through R_{max}. The value of $V^j(S)$ shall be incremented by 1 after each (re-)transmission of an AL-PDU for an AL-SDU with sequence number j. The value of $V^j(S)$ shall be set to 0 in the following cases:

- at initialization;
- when the send buffer B_S no longer contains information for the corresponding AL-PDU payload.

d) *Send sequence number N(S)*

AL-PDUs contain N(S), the send sequence number of the corresponding AL-PDU Payloads. At the time that an in-sequence AL-PDU is designated for transmission, the value of N(S) is set equal to V(S).

e) *Maximum number of retransmissions R_{max}*

R_{max} is a parameter that indicates the maximum number of retransmissions allowed. Its value shall be indicated by the AL1M transmitting unit in the H.245 OpenLogicalChannel message.

f) *Send buffer B_S*

Each AL1M entity shall maintain a send buffer, B_S, used for storing the most recently transmitted AL-PDU Payload information. The minimum size of B_S that all AL1M transmitters shall support is specified in the System Recommendation (e.g. H.324) that uses this Annex. The actual size of B_S shall be indicated to the far end in the H.245 OpenLogicalChannel message.

g) *Receive sequence variable V(R)*

V(R) is an internal variable of the AL1M receiving entity. It denotes the sequence number of the next in-sequence AL-PDU expected to be received. V(R) can take on the values 0 through 2^5 or 2^{10} . The value of V(R) shall be incremented by 1 with the receipt of a valid, in-sequence AL-PDU whose N(S) equals V(R).

h) *Receive retransmission variables $V^j(R)$*

$V^j(R)$ are internal variables of the AL1M receiving entity. $V^j(R)$ can take values 0 through R_{max}. The value of a variable $V^j(R)$ shall be used to monitor the number of retransmissions requested. When the ARQII error protection scheme is used, the value of a variable $V^j(R)$ shall also be used to determine the number i of the next AL-PDU payload z_i to be received from the AL1M transmitting entity.

The value of a variable $V^j(R)$ shall be incremented by 1 with the receipt of an errored AL-PDU with N(S) = j.

The value of a variable $V^j(R)$ shall be set to 0 when the received AL-PDU with N(S) = j results in an error-free decoding of the corresponding AL-PDU Payload.

i) *Receive retransmission number RN*

Only the header field of the reverse channel contains RN, the receive retransmission number. When requesting a re-transmission, this 1-bit number shall be set to the parity of the receive retransmission variable of the requested AL-PDU payload.

j) *Receive sequence number N(R)*

Only the header field of the reverse channel contains N(R), the receive sequence number of an AL-PDU that is referred to by the reverse header field.

C.4.1.13.2 Supervisory messages

According to the direction, either forward or reverse channel, an S-PDU is sent with different messages is transmitted:

- an S-PDU from the transmitter to receiver (forward channel) signals a *DRTX* message;
- an S-PDU from the receiver to the transmitter (reverse channel) carries an *SREJ* message.

Selective reject (SREJ) message

SREJ is used by an AL1M receiver to request a retransmission for the single AL-PDU Payload numbered N(R). An SREJ message shall not be transmitted more times than the negotiated maximum number of retransmissions R_{max} for the same AL-PDU Payload.

Declined retransmission (DRTX) message

Since the error recovery procedures defined here only support negative acknowledgment, in certain conditions, the AL-PDU Payload information transmitted previously may have been discarded before the request for retransmission is received. The DRTX message is used by an AL1M transmitter to decline the requested retransmission for an AL-PDU Payload, when the information for that AL-PDU Payload is not available in the send buffer at the time the SREJ message is received.

C.4.1.13.3 Initialization procedures

The retransmission procedures require that a reverse logical channel exist for sending supervisory messages.

Once the reverse logical channel has been established according to the procedure defined in Recommendation H.245, the AL1M entity shall:

- set V(S), V(R), $V^j(S)$, $V^j(R)$ to 0;
- clear any existing exception conditions.

C.4.1.13.4 Transmitting in-sequence I-PDUs

Information received from the AL1 user in an AL-SDU by means of an AL-DATA.request primitive shall be passed to the MUX layer in an I-PDU using the frame structure defined in C.4.1.4. The SN field of the I-PDU shall be assigned the value V(S). V(S) shall be incremented by 1 after the I-PDU has been passed to the MUX layer.

C.4.1.13.5 Receiving in-sequence I-PDUs

When an AL1M entity receives a valid I-PDU whose N(S) is equal to the current V(R), the AL1M entity shall increment its V(R) by 1.

C.4.1.13.6 Receiving SREJ-PDUs

On receipt of a valid SREJ-PDU, the AL1M entity shall act as follows:

a) If the I-PDU, whose N(S) is equal to the N(R) of the SREJ message is still in the send buffer, the AL1M entity shall pass a corresponding AL-PDU to the MUX layer as soon as possible.

   When ARQI error protection is used, the same AL-PDU payload shall be used for re-transmission.

   When ARQII is used, the parity of the send retransmission variable $V^j(S)$ is checked against the 1-bit receive retransmission number N(R). If the parity differs, $V^j(S)$ will be decremented by 1. Then the next I-PDU payload, according to the procedure described in C.4.1.9, shall be retransmitted to the receiver.

   No other previously transmitted I-PDUs shall be retransmitted as a result of receiving the SREJ-PDU.

b) If the AL-PDU whose N(S) is equal to the N(R) of the SREJ message has been previously discarded, the AL1M entity shall enter a declined-retransmission exception condition. The procedures for this exception condition are defined in C.4.1.13.8 e).

C.4.1.13.7 Transmitting SREJ messages

When a valid but errored I-PDU is received and $V^j(R) < R_{max}$, a SREJ message shall be produced with the receive sequence number N(R) set to the N(S) from the errored I-PDU and the modulo 2 of $V^j(R)$ set to RN field. The corresponding receive retransmission variable $V^j(R)$ shall be incremented.

C.4.1.13.8 Exception condition reporting and recovery

Exception conditions may occur as a result of errors on the physical connection or procedural errors by an AL1M entity.

The error-recovery procedures that are available following the detection of an exception condition by an AL1M entity are defined in this subclause.

a) *Receiving invalid AL-PDUs*

   When a received AL-PDU is invalid, it is either discarded or saved for possible delivery later to the AL1 user.

b) *N(S) sequence error*

   When there are no other outstanding exception conditions, an N(S) sequence error exception condition occurs in the receiving AL1M entity when a valid I-PDU is received containing an N(S) value that is not equal to the V(R) at the receiver. In this case, V(R) shall not be incremented, and one or more SREJ-PDUs, each containing a different N(R), may be transmitted by the AL1M receiving entity to initiate an exception condition recovery for each SREJ-PDU. After passing each SREJ-PDU to the MUX layer, the AL1M entity shall start a local timer. Several factors that affect the length of the timer are given in Appendix IV/V.42. A different timer is maintained for each outstanding SREJ-PDU. Successive SREJ-PDUs are transmitted in the order indicated by their N(R) field.

For each SREJ-PDU that it transmits, the AL1M receiver may pass an empty AL-SDU or an invalid received AL-SDU (previously saved), with an appropriate EI parameter, to the AL1 user via the AL-DATA.indication primitive.

When the retransmitted I-PDU with N(S) = V(R) is received, the exception condition for that I-PDU shall be cleared. The AL1M receiver should pass the associated AL-SDU, together with an appropriate EI parameter, to the AL1 user via the AL-DATA.indication primitive. When the exception condition is cleared, the associated timer shall be stopped and V(R) shall be incremented as many times as necessary so that V(R) represents the send sequence number of the next expected in-sequence I-PDU.

When a retransmitted I-PDU with N(S) ≠ V(R) is received, the AL1M receiving unit shall stop the timers associated with all exception conditions related to previously sent SREJ-PDUs for which retransmission is received, by stopping the associated timers. For each exception condition cleared, the AL1M receiver shall increment V(R) by 1, and may deliver an empty AL-SDU, together with an appropriate EI parameter, to the AL1 user via the AL-DATA.indication primitive, prior to delivering the AL-SDU associated with the received I-PDU.

The information in all other received valid I-PDUs should be delivered to the AL1 user in AL-SDUs, together with an appropriate EI parameter.

c) *N(R) sequence error*

An N(R) sequence error exception condition occurs when a valid S-PDU with an invalid N(R) value is received. An invalid N(R) value will be produced when a first SREJ-PDU received with sequence number N(R) = N1 is followed by another SREJ-PDU with N(R) = N2, and (V(S) – N2) is greater than, or equal to, (V(S) – N1).

An invalid N(R) may also occur when the N(R) value in a DRTX-PDU is not equal to the N(R) value in an outstanding SREJ-PDU.

The AL1M entity should ignore the message in such S-PDUs.

d) *Procedure on expiration of timer*

If the timer expires, the associated exception condition shall be cleared by stopping the timer and incrementing V(R). The AL1M receiver may pass an empty AL-SDU or an invalid received AL-SDU (previously saved), with an appropriate error indication, to the AL1 user via the AL-DATA.indication primitive.

e) *Declined-retransmission condition*

Error recovery procedures at the AL1M sending unit

Upon receiving an SREJ retransmission request, when the *AL1*M transmitter does not have the information for the requested AL-PDU Payload stored in the send buffer, it shall:

- send a declined-retransmission (DRTX) message whose N(R) value is equal to the N(R) value of the received SREJ message as soon as possible;
- send an AL-DRTX.indication to the *AL1 user;*
- resume transmission of AL-PDUs not yet transmitted.

Error recovery procedures at the AL1M receiving unit

When a DRTX message is received, the associated exception condition should be cleared by stopping the timer and incrementing V(R). The AL1M receiver may pass an empty AL-SDU or an invalid received AL-SDU (previously saved), with an appropriate error indication, to the AL1 user via the AL-DATA.indication primitive.

C.4.1.13.9 Unsolicited supervisory PDUs

An unsolicited DRTX-PDU received by the AL1M entity should be ignored.

C.4.2 AL2M

C.4.2.1 Framework of AL2M

AL2M is primarily designed for the transfer of digital audio in highly error-prone channels.

AL2M only provides optional sequence numbering and optional AL-PDU interleaving. Therefore, any additional error control may be provided by the higher layer protocol. For example, the Annex C/G.723.1 defines such an error control procedure.

The AL-SDU and the AL-PDU shall be octet aligned.

Audio frames are first mapped to AL-SDU and these are then passed by AL2M in MUX-SDUs with an optional AL2M-header and optional interleaving to the MUX layer.

C.4.2.2 Primitives exchanged between AL2M and AL2 user

The information exchanged between AL2M and an AL2 user includes the following primitives:

– AL-DATA.request (AL-SDU);

– AL-DATA.indication (AL-SDU, EI);

– AL-Abort.request.

C.4.2.2.1 Description of primitives

– AL-DATA.request: This primitive is issued by an AL2 user to AL2M to request the transfer of an AL-SDU to a corresponding AL2 user.

– AL-DATA.indication: This primitive is issued to an AL2 user by AL2M to indicate the arrival of an AL-SDU.

– AL-Abort.request: This primitive is issued to AL2M by an AL2 user to signal that a partially delivered AL-SDU is to be aborted.

C.4.2.2.2 Description of Parameters

• AL-SDU: This parameter specifies the unit of information exchanged between AL2M and the AL2 user. Each AL-SDU shall contain an integral number of octets. The length of AL-SDUs may be variable. The maximum length of AL-SDUs that an AL2M receiver can accept shall be signalled via the H.245 control channel. The octets in an AL-SDU are numbered from 1 to n, and in each octet, bits are numbered from 1 to 8. Bit 1 of the octet 1 is transmitted first. An AL2M receiving entity may deliver an empty AL-SDU to the AL2 user to indicate that an AL-SDU is missing.

• Error Indication (EI): This parameter should be used in the AL2M receiver to pass error indications to the AL2 user. This may also be used if the AL2M receiver entity delivers an empty AL-SDU to the AL2 user. The precise procedures for using this parameter, and its numerical coding, are outside the scope of Recommendation H.223.

C.4.2.3 Format and Coding of AL2M

The format of the AL-PDU is illustrated in Figure C.8. The entire AL-PDU interleaving as described in C.4.2.3.2 shall be used, if this has been required in the H.245 OpenLogicalChannel message.

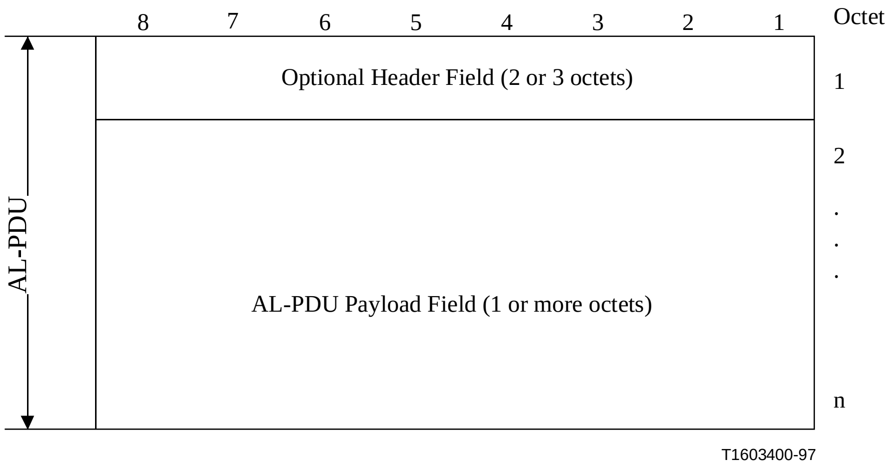

Figure C.8/H.223 – AL-PDU format for AL2M

C.4.2.3.1 Header Field

The optional header consists of the 5-bit or 12-bit Sequence Number (SN) and a corresponding Header Error Correction (HEC) field. This HEC uses either an SEBCH(16,5) or an EGolay Code(24,12), (see Figures C.9 and C.10).

8	7	6	5	4	3	2	1	Octet
P3	P2	P1	SN5	SN4	SN3	SN2	SN1	1
P11	P10	P9	P8	P7	P6	P5	P4	2

Figure C.9/H.223 – Control field format of the AL-PDU for AL2M with SN = 5 and SEBCH code

8	7	6	5	4	3	2	1	Octet
SN8	SN7	SN6	SN5	SN4	SN3	SN2	SN1	1
P4	P3	P2	P1	SN12	SN11	SN10	SN9	2
P12	P11	P10	P9	P8	P7	P6	P5	3

Figure C.10/H.233 – Control field format of the AL-PDU for AL2M with SN = 12 EGolay code

C.4.2.3.1.1 Sequence Number (SN) field

The optional 5-bit/12-bit SN provides a capability for sequencing AL-PDUs. The sequence number may be used by the AL2M receiving entity to detect missing and misdelivered AL-PDUs.

AL2M receivers conforming to Recommendation H.223 shall be capable of receiving and correctly interpreting AL-PDUs that include the SN field. The use of the SN field shall be determined by the transmitter and shall be signalled to the far end in the OpenLogicalChannel message of H.245.

When the SN field is in use, the AL2M receiver may detect that an AL-PDU is missing or has been misdelivered by the MUX layer. The AL2M receiver should discard any misdelivered AL-PDUs that it detects.

C.4.2.3.1.2 Header Error Correction (HEC) field of the AL2M header

The optional AL2M header uses an SEBCH(16,5) or an EGolay (24,12) code. The definition of the EGolay code shall be the same as described in C.4.1.5.4, whereby the field RN is replaced by SN11 and X by SN12. The SEBCH code shall be used by the definition of Table I.1. The CEC bits of SEBCH code shall be derived by the following equation:

$$\begin{bmatrix} P1 \\ P2 \\ P3 \\ P4 \\ P5 \\ P6 \\ P7 \\ P8 \\ P9 \\ P10 \\ P11 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}^{T} \bullet \begin{bmatrix} SN1 \\ SN2 \\ SN3 \\ SN4 \\ SN5 \end{bmatrix}$$

C.4.2.3.1.3 AL-PDU Payload Field

The AL-PDU field contains a complete AL-SDU, where the first octet corresponds to the first octet of the AL-SDU. Both the AL-SDU and the AL-PDU are octet aligned.

C.4.2.3.2 Interleaving

If interleaving is required in the H.245 OpenLogicalChannel message, it shall be applied to the entire AL-PDU including the header field. The same interleaver as described in C.4.1.8 shall be used for AL2M.

De-interleaving also has to be applied at receiver side in this case.

C.4.2.4 Procedures for Abort

This primitive is discarded and no action is undertaken.

C.4.2.5 Procedure for Sequence Numbering

The following procedures apply when the SN field is in use.

Once a logical channel using AL2M is successfully opened according to the procedure defined in H.245, the first AL-PDU transmitted by the AL2M sending entity shall have the SN field set to 0. For each subsequent transmitted AL-PDU which belongs to that logical channel, the value of the SN field shall be incremented by 1 modulo 32 for a 5-bit SN-field (or modulo 4096 for the 12-bit SN-field).

C.4.2.6 Procedures for Error Control

When the SEBCH/EGolay decoding fails at the AL2M receiver, the associated AL-SDU may be delivered to the AL2 user, together with an appropriate error indication, via the AL-DATA.indication primitive.

When the SN field is in use, the AL2M receiver may detect that an AL-PDU is missing or has been misdelivered by the MUX layer. The AL2M receiver should discard any misdelivered AL-PDUs that it detects. For each missing AL-PDU that it detects, the AL2M receiver may deliver to the AL2 user an empty AL-SDU, together with an appropriate error indication, via the AL-DATA.indication primitive.

C.4.3 AL3M

AL3M is designed primarily for the transfer of video. The format, structure, definitions and procedures are identical to the adaptation layer AL1M, see C.4.1, except that:

- AL3M shall support one framed transfer mode; and
- AL3M shall always operate in splitting mode, while using the ARQ-mode and shall not use splitting mode, while operating in FEC_ONLY mode.

In AL3M it is possible that additional error control can be provided by the higher layer protocol, e.g. by the procedures in Annex N/H.263.

APPENDIX I

Generator matrixes of the Systematic Extended BCH

This appendix describes Systematic Extended Bose-Chaudhuri-Hocquenghem (SEBCH) codes and includes the generator matrixes, which are used this Annex.

I.1 BCH codes

BCH codes are linear cyclic block codes, hence they can be described using a generator polynomial. However, the easiest way to describe short block codes is using a generator matrix which describes all characteristics of the code. With a generator matrix $\underline{\boldsymbol{G}}$ and a information sequence $\underline{\boldsymbol{i}}$ of length $\boldsymbol{k}$ the code vector $\underline{\boldsymbol{c}}$ of length $\boldsymbol{n}$ can be obtained by:

$$\underline{\mathbf{c}} = \underline{\mathbf{i}} \cdot \underline{\mathbf{G}} = [\ \underline{\mathbf{i}}^{\mathbf{T}} \mid \underline{\mathbf{c}}_{\mathbf{o}}^{\mathbf{T}}\ ]^{\mathbf{T}}$$

with $\underline{\boldsymbol{G}} = [\ \underline{\boldsymbol{1}} \mid \underline{\boldsymbol{A}}\ ]$ a $(\boldsymbol{k} \times \boldsymbol{n})$ matrix containing a $(\boldsymbol{k} \times \boldsymbol{k})$ identity matrix in the first $\boldsymbol{k}$ columns/rows to obtain a systematic code. For a primitive BCH code the length of the code $\boldsymbol{n}$ is always $\boldsymbol{n = 2^{h} - 1}$. For $\boldsymbol{k}$ there are some constraints, not all values are possible.

The third parameter describing a block code besides code length $\boldsymbol{n}$ and information length $\boldsymbol{k}$ is the minimum distance between two code words $\boldsymbol{d}$. If a code has minimal distance $\boldsymbol{d}$, it can correct at most $\lfloor \boldsymbol{(d-1)/2} \rfloor$ errors or detect $\boldsymbol{(d-1)}$ errors.

I.2 Systematic Extended BCH codes

As all linear cyclic block codes can be made systematic, there always exits a systematic BCH code.

As we evaluated earlier, primitive BCH codes always have the length $\boldsymbol{n = 2^{h} - 1}$. To make these codes octet aligned, extension has to be applied. The extension of a BCH($\boldsymbol{n, k, d}$) has the length $\boldsymbol{n + 1}$. One digit is appended, so that each code word has even weight. The extended BCH code then always has minimal distance $\boldsymbol{d + 1}$. Hence we derived from BCH($\boldsymbol{n, k, d}$) a code EXBCH($\boldsymbol{n + 1, k,}$

d + 1). Extended codes are still linear, but no more cyclic. Hence the description using generator polynomials is impossible.

The generator matrix of the extended code from $\underline{G}$ of the mother code can be derived by adding one column which contains the parity check bit of each row. The generator matrices of the codes used in this proposal are given in Tables I.1 and I.2.

I.3 Decoder overview

For decoding BCH codes, usually Berleykamp-Massey algorithm is used. This is an efficient method to determine error locations in the received vector. There are also some approaches to use reliability information for decoding block codes. However, these algorithms yield in high complexity.

One main feature of BCH codes is the possibility to use these codes for error correction and detection at the same time. For example a code with $d = 5$ could correct up to one error and detect up to three errors in parallel. With the usage of BCH codes only, the decoder has the flexibility to decide how many errors to correct and use the rest of redundancy for error detection. Berleykamp-Massey algorithm can also be used for this.

Example

In this example we use the SEBCH(16,5,8). The information vector $\underline{c}$ is given as:

$$\underline{c} = [1\ 0\ 0\ 1\ 1]$$

By using the generator matrix $\underline{G}$, the code word $\underline{c}$ can be evaluated by:

$$\underline{c} = \underline{i} \cdot \underline{G} = [1\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0]$$

For transmission these bits are filled into octet-aligned fields. The LSB-bit of the vector $\underline{c}$ is at its left side, the MSB at its right. The LSB of $\underline{c}$ is filled to the lowest numbered bit of the last octet (octet 2) and the MSB of $\underline{c}$ to the highest-numbered bit of the first octet (octet 1) (see Figure I.1).

8	7	6	5	4	3	2	1	Octet
0	0	0	1	1	1	1	1	1
0	1	0	1	1	0	0	1	2

Figure I.1/H.223 – Field mapping convention of SEBCH codes

I.4 Generator matrices for Systematic Extended BCH codes

In this subclause we provide some tables to calculate a code sequence $\underline{c}$ of length n from a given input sequence $\underline{i}$ of length k using the generator matrix $\underline{G}$ with the equation: $\underline{c} = \underline{i} \cdot \underline{G}$. SEBCH(16,5,8) is derived from BCH(15,5,7) with generator polynomial $g(x) = x^{10} + x^8 + x^5 + x^4 + x^2 + x^1 + 1$, SEBCH(16,7,6) is derived from BCH(15,7,5) with generator polynomial $g(x) = x^8 + x^7 + x^6 + x^4 + 1$.

Table I.1/H.223 – Generator matrix for Systematic Extended BCH(16,5,8) code

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	0	0	0	0	1	1	1	0	1	1	0	0	1	0	1
1	0	1	0	0	0	0	1	1	1	0	1	1	0	0	1	1
2	0	0	1	0	0	1	1	0	1	0	1	1	1	1	0	0
3	0	0	0	1	0	0	1	1	0	1	0	1	1	1	1	0
4	0	0	0	0	1	1	1	0	1	1	0	0	1	0	1	1

Table I.2/H.223 – Generator matrix for Systematic Extended BCH(16,7,6) code

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	1	0	0	0	0	0	0	1	0	0	0	1	0	1	1	1
1	0	1	0	0	0	0	0	1	1	0	0	1	1	1	0	0
2	0	0	1	0	0	0	0	0	1	1	0	0	1	1	1	0
3	0	0	0	1	0	0	0	1	0	1	1	1	0	0	0	1
4	0	0	0	0	1	0	0	0	1	0	1	1	1	0	0	1
5	0	0	0	0	0	1	0	0	0	1	0	1	1	1	0	1
6	0	0	0	0	0	0	1	0	0	0	1	0	1	1	1	1

ITU-T RECOMMENDATIONS SERIES

Series A	Organization of the work of the ITU-T
Series B	Means of expression: definitions, symbols, classification
Series C	General telecommunication statistics
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	**Audiovisual and multimedia systems**
Series I	Integrated services digital network
Series J	Transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks and open system communications
Series Y	Global information infrastructure
Series Z	Programming languages