INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

I.363

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

(03/93)

INTEGRATED SERVICES DIGITAL NETWORK (ISDN)

OVERALL NETWORK ASPECTS AND FUNCTIONS

B-ISDN ATM ADAPTATION LAYER (AAL) SPECIFICATION

ITU-T Recommendation I.363

(Previously "CCITT Recommendation")

FOREWORD

The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the International Telecommunication Union. 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, established the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

ITU-T Recommendation I.363 was prepared by the ITU-T Study Group XVIII (1988-1993) and was approved by the WTSC (Helsinki, March 1-12, 1993).

NOTES

1 As a consequence of a reform process within the International Telecommunication Union (ITU), the CCITT ceased to exist as of 28 February 1993. In its place, the ITU Telecommunication Standardization Sector (ITU-T) was created as of 1 March 1993. Similarly, in this reform process, the CCIR and the IFRB have been replaced by the Radiocommunication Sector.

In order not to delay publication of this Recommendation, no change has been made in the text to references containing the acronyms "CCITT, CCIR or IFRB" or their associated entities such as Plenary Assembly, Secretariat, etc. Future editions of this Recommendation will contain the proper terminology related to the new ITU structure.

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

CONTENTS

Page

Recommendation I.363

B-ISDN ATM ADAPTATION LAYER (AAL) SPECIFICATION

(Geneva, 1991; revised Helsinki, 1993)

1 Introduction

The ATM adaptation layer (AAL) enhances the service provided by the ATM layer to support functions required by the next higher layer. The AAL performs functions required by the user, control and management planes and supports the mapping between the ATM layer and the next higher layer. The functions performed in the AAL depend upon the higher layer requirements.

The AAL supports multiple protocols to fit the needs of the different AAL service users. The service provided by the AAL to the higher layer and the functions performed are specified in this Recommendation.

Details of the data unit naming convention used in this Recommendation can be found in Annex A.

1.1 Scope of the Recommendation

This Recommendation describes the interactions between the AAL and the next higher layer, and the AAL and the ATM layer, as well as AAL peer-to-peer operations. This Recommendation is based on the classification and the AAL functional organization described in Recommendation I.362.

Different combinations of SAR (segmentation and reassembly) sublayers and CS (convergence sublayer) provide different service access points (SAPs) to the layer above the AAL. In some applications the SAR and/or CS may be empty.

1.2 Information flow across the ATM-AAL boundary

The AAL receives from the ATM layer the information in the form of a 48 octet ATM service data unit (ATM-SDU). The AAL passes to the ATM layer information in the form of a 48 octet ATM SDU. See Recommendation I.361 for the description of primitives provided by the ATM layer.

2 AAL type 1

2.1 Service provided by AAL type 1

2.1.1 Definitions

For the purpose of this Recommendation, the following definitions apply:

The layer services provided by AAL type 1 to the AAL user are:

- transfer of service data units with a constant source bit rate and the delivery of them with the same bit rate;
- transfer of timing information between source and destination;
- transfer of structure information between source and destination;
- indication of lost or errored information which is not recovered by AAL type 1, if needed.

2.1.2 Primitives

2.1.2.1 General

At the AAL-SAP, the following primitives will be used between the AAL type 1 and the AAL user:

- From an AAL user to the AAL,

  AAL-UNITDATA-REQUEST;

- From the AAL to an AAL user,

  AAL-UNITDATA-INDICATION.

An AAL-UNITDATA-REQUEST primitive at the local AAL-SAP results in an AAL-UNITDATA-INDICATION primitive at its peer AAL- SAP.

2.1.2.2 Definition of primitives

2.1.2.2.1 AAL-UNITDATA-REQUEST

AAL-UNITDATA-REQUEST (DATA [mandatory],
STRUCTURE [optional])

The AAL-UNITDATA-REQUEST primitive requests the transfer of the AAL-SDU, i.e. contents of the DATA parameter, from the local AAL entity to its peer entity. The length of the AAL-SDU is constant and the time interval between two consecutive primitives is constant. These two constants are a function of the AAL service provided to the AAL user.

2.1.2.2.2 AAL-UNITDATA-INDICATION

AAL-UNITDATA-INDICATION (DATA (mandatory],
STRUCTURE [optional],
STATUS [optional])

An AAL user is notified by the AAL that the AAL-SDU, i.e. contents of the DATA parameter, from its peer are available. The length of the AAL-SDU should be constant and the time interval between two consecutive primitives should be constant. These two constants are a function of the AAL service provided to the AAL user.

2.1.2.3 Definition of parameters

2.1.2.3.1 STRUCTURE parameter

The STRUCTURE parameter can be used when the user data stream to be transferred to the peer AAL entity is organized into groups of bits. The length of the structured block is fixed for each instance of the AAL service. The length is an integer multiple of 8 bits. An example of the use of this parameter is to support circuit mode bearer services of the 64 kbit/s based ISDN. The two values of the STRUCTURE parameter are:

START; and

CONTINUATION.

The value START is used when the DATA is the first part of a structured block which can be composed of consecutive DATA. In other cases, the STRUCTURE parameter is set to CONTINUATION. The use of the STRUCTURE parameter depends on the type of AAL service provided. The use of this parameter is agreed prior to or at the connection establishment between the AAL user and the AAL.

2.1.2.3.2 STATUS parameter

The STATUS parameter identifies that the DATA is judged to be non-errored or errored. The STATUS parameter has two values:

VALID; and

INVALID.

The INVALID status could also imply that the DATA is a dummy value. The use of the STATUS parameter and the choice of dummy value depend on the type of AAL service provided. The use of this parameter is agreed prior to or at the connection establishment between the AAL user and the AAL.

2.2 Interaction with the management and control planes

2.2.1 Management plane

The following indications may be passed from the user plane to the management plane:

- errors in the transmission of user information;
- lost or misinserted cells (further study is required on whether it is necessary to distinguish between lost and misinserted cells for management purposes);
- cells with errored AAL protocol control information (AAL-PCI) (further study is required to determine if this indication is necessary for layer services supported by this AAL type;
- loss of timing and synchronization;
- buffer under flow and overflow.

2.2.2 Control plane

For further study.

2.3 Functions of AAL type 1

The following functions may be performed in the AAL type 1 in order to enhance the ATM layer service:

a) segmentation and reassembly of user information;

b) handling of cell delay variation;

c) handling of cell payload assembly delay;

d) handling of lost and misinserted cells;

e) source clock frequency recovery at the receiver,

f) recovery of the source data structure at the receiver;

g) monitoring of AAL-PCI for bit errors;

h) handling of AAL-PCI bit errors;

i) monitoring of user information field for bit errors and possible corrective action.

Other functions are for further study.

NOTE – For some AAL users, the end-to-end QOS may be monitored. This may be achieved by calculating a CRC for the CS-PDU payload, carried in one or more cells, and transmitting the CRC results in the CS-PDU or by the use of OAM cells. Further study is required.

2.4 Segmentation and Reassembly (SAR) sublayer

2.4.1 Functions of the SAR sublayer

The SAR sublayer functions are performed on an ATM-SDU basis.

a) *Mapping between CS-PDU and SAR PDU*

The SAR sublayer at the transmitting end accepts a 47 octet block of data from the convergence sublayer (CS), and then prepends a one octet SAR-PDU header to each block to form the SAR-PDU.

The SAR sublayer at the receiving end receives the 48 octet block of data from the ATM layer, and then separates the SAR-PDU header. The 47 octet block of SAR-PDU payload is passed to the CS.

b) *Existence of CS function*

The SAR sublayer has the capability to indicate the existence of a CS function. Associated with each 47 octet SAR-PDU payload, it receives this indication from the CS and conveys it to the peer CS entity. The use of this indication is optional.

c) *Sequence numbering*

Associated with each SAR-PDU payload, the SAR sublayer receives a sequence number value from the CS. At the receiving end, it passes the sequence number value to the CS. The CS may use these sequence number values to detect lost or misinserted SAR-PDU payloads (corresponding to lost or misinserted ATM cells).

d) *Error protection*

The SAR sublayer protects the sequence number value and the CS indication against bit errors. It informs the receiving CS when the sequence number value and the CS indication are errored and cannot be corrected.

NOTE – For certain applications such as speech, some SAR functions may not be needed. This item is for further study.

2.4.2 SAR protocol

The SAR-PDU header together with the 47 octets of the SAR-PDU payload comprises the 48 octet AIM-SDU (cell information field). The size and positions of the fields in the SAR-PDU are given in Figure 1.

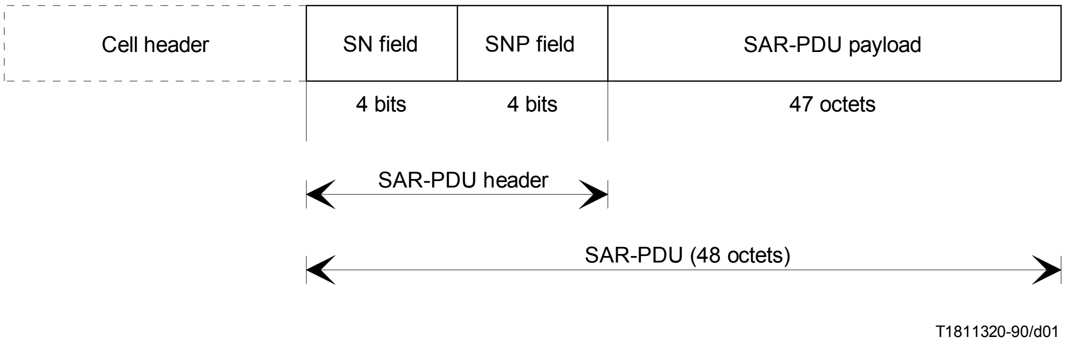


FIGURE 1/I.363

SAR-PDU format of AAL type 1

2.4.2.1 Sequence number (SN) field

The SN field is divided into two subfields as shown in Figure 2. The sequence count field carries the sequence count value provided by the convergence sublayer (CS). The CSI bit carries the CS indication provided by the CS. The default value of the CSI bit is “0”.

The least significant bit of the sequence count value is right justified in the sequence count field.

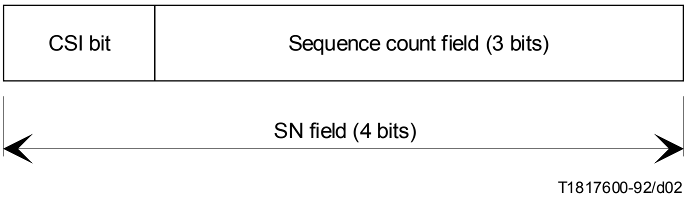


FIGURE 2/I.363

Sequence number (SN) field format

2.4.2.2 Sequence number protection (SNP) field

The SNP field provides error detection and correction capabilities over the SAR-PDU header. The format of this field is given in Figure 3. A two step approach is used for the protection:

1) The sequence number (SN) field is protected by a 3 bit CRC code.

2) The resulting 7 bit codeword is protected by an even parity check bit.

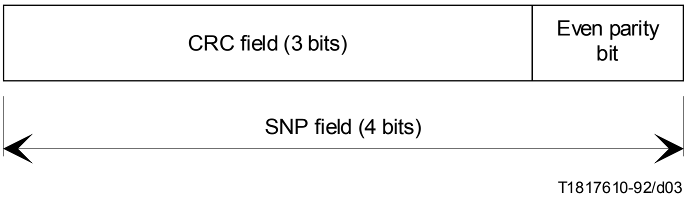


FIGURE 3/I.363

SNP field format

The receiver is capable of either single-bit error correction or multiple-bit error detection.

a) *Operations at transmitting end*

The transmitter computes the CRC value across the first 4 bits of the SAR-PDU header and inserts the result in the CRC field.

The notation used to describe the CRC is based on the property of cyclic codes. The elements of an element codeword are thus the coefficients of a polynomial of order $n - 1$. In this application, these coefficients can have the value 0 or 1 and the polynomial operations are performed using modulo 2 operations. For example a code vector such as 1011 can be represented by the polynomial $P(x) = x^3 + x + 1$. The polynomial representing the content of the SN field is generated using the first bit of the SN field as the coefficient of the highest order term.

The CRC field consists of three bits. It shall contain the remainder of the division (modulo 2) by the generator polynomial $x^3 + x + 1$ of the product x^3 multiplied by the content of the SN field.

After completing the above operations, the transmitter inserts the even parity bit.

b) *Operations at receiving end*

The receiver has two different modes of operation : correction mode and detection mode. These modes are related as shown in Figure 4. The default mode is the correction mode, which provides for single-bit error correction. At initialization, the receiver is set up in this default mode.

The receiver examines each SAR-PDU header by checking the CRC bits and even parity bit. If a header error is detected, the action taken depends on the state of the receiver. In the "correction mode", only single-bit errors can be corrected and the receiver switches to "detection mode". In "detection mode", all SAR-PDU headers with detected errors are declared to have an invalid SN; however, when a SAR-PDU header is examined and found not to be in error, the receiver switches to "correction mode".

Tables 1 and 2 give the detailed operations of the receiver in the "correction mode" and "detection mode", respectively. The operation is based on the combined validity of the CRC and parity check bit.

The receiver conveys the sequence number count and the CS indication to the CS together with SN check status (valid or invalid).

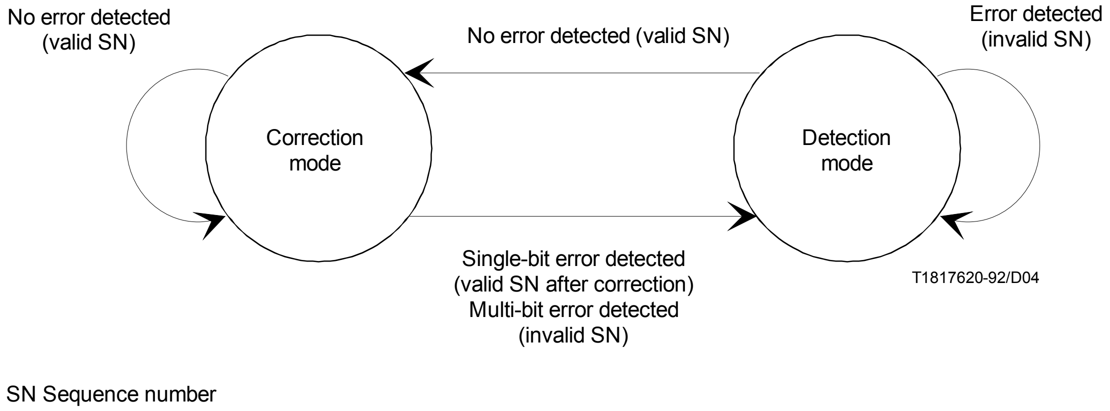


FIGURE 4/I.363

SNP: receiver modes of operation

2.5 Convergence sublayer (CS)

2.5.1 Functions of the CS

The CS may include the following functions. For performing some of these functions, the CS will need a clock. This clock may be derived from the S_B or T_B interface.

a) Handling of cell delay variation is performed at this sublayer for delivery of AAL-SDUs to an AAL user at a constant bit rate.

b) Processing of sequence count may be performed at this sublayer. The sequence count value and its error check status provided by the SAR sublayer can be used by the CS to detect cell loss and misinsertion. Further handling of lost and misinserted cells is also performed in this sublayer.

c) The CS can utilize the CS indication provided by the SAR sublayer to support CS functions for some AAL users.

d) For AAL users requiring recovery of source clock frequency at the destination end, the AAL can provide a mechanism for a timing information transfer.

e) For some AAL users, this sublayer provides the transfer of structure information between source and destination.

f) For video and high quality audio signal transport, forward error correction may be performed to protect against bit errors. This may be combined with interleaving of AAL user bits (e.g. octet interleaving) to give more secure protection against errors.

g) The CS may generate reports giving the status of end-to-end performance as deduced by the AAL. The performance measures in these reports could be based on:

  - events of lost and misinserted cells;
  - buffer underflow and overflow;
  - bit error events.

The following subclauses identify the functions of the CS for individual layer services of AAL type 1.

TABLE 1/I.363

Operations in correction mode

CRC Syndrome	Parity	Action on current SN + SNP	Reaction for next SN + SNP
Zero	No violation	No corrective action. Declare SN valid.	Continue in correction mode
Non-zero	Violation	Single bit correction based on syndrome. Declare SN valid.	Switch to detection mode
Zero	Violation	Correct parity bit. Declare SN valid.	Switch to detection mode
Non-zero	No violation	No corrective action: multi-bit errors are uncorrectable. Declare SN invalid.	Switch to detection mode
SN Sequence number SNP Sequence number protection			

TABLE 2/I.363

Operations in detection mode

CRC Syndrome	Parity	Action on current SN + SNP	Reaction for next SN + SNP
Zero	No violation	No corrective action. Declare SN valid.	Switch to correction mode
Non-zero	Violation	No corrective action. Declare SN invalid.	Continue in detection mode
Zero	Violation	No corrective action. Declare SN invalid.	Continue in detection mode
Non-zero	No violation	No corrective action. Declare SN invalid.	Continue in detection mode
SN Sequence number SNP Sequence number protection			

2.5.1.1 Functions of the CS for circuit transport

The following functions support both asynchronous and synchronous circuit transport . Asynchronous circuit transport will provide transport of signals from constant bit rate sources whose clocks are not frequency-locked to a network clock. Examples are Recommendation G.702 signals at 1544, 2048, 6312, 8448, 32 064, 44 736 and 34 368 kbit/s. Synchronous circuit transport will provide transport of signals from constant bit rate sources whose clocks are frequency-locked to a network clock. Examples are signals at 64, 384, 1536 and 1920 kbit/s as described in Recommendation I.231.

NOTE – Another possible example of synchronous circuit transport is conveyance of SDH signals described in Recommendation G.709.

a) *Handling of AAL user information*

The length of the AAL-SDU is one bit, when asynchronous circuit transport utilizes the synchronous residual time stamp (SRTS) method described in 2.5.2.2.1.

For those AAL users which require transfer of structured data, e.g. 8 kHz structured data for circuit mode bearer services of the 64 kbit/s based ISDN, the STRUCTURE parameter option of the primitives defined in 2.1.2 will be used. The CS uses the structured data transfer (SDT) method described in 2.5.2.3.

b) *Handling of cell delay variation*

A buffer is used to support this function. The size of this buffer is dependent upon specifications currently under study.

In the event of buffer underflow, it may be necessary for the CS to maintain bit count integrity by inserting the appropriate number of dummy bits. In the event of buffer overflow, it may be necessary for the CS to maintain bit count integrity by dropping the appropriate number of bits.

When Recommendation G.702 1.544 Mbit/s and 2.048 Mbit/s signals are being transported, the inserted dummy bits shall be all " 1 "s.

c) *Handling of lost and misinserted cells*

The sequence count values are further processed at this sublayer to detect lost and misinserted cells. Detected misinserted cells are discarded. The CS procedure to be used for sequence count processing is described in 2.5.2.l.

In order to maintain the bit count integrity of the AAL user information, it may be necessary to compensate for lost cells detected by buffer underflow and sequence count processing by inserting the appropriate number of dummy SAR-PDU payloads. The content of this dummy SAR-PDU payload depends on the AAL service being provided. For example, this dummy SAR-PDU payload is all "1"s for Recommendation G.702 1.544 Mbit/s and 2.048 Mbit/s signals.

d) *Handling of timing relation*

This function is required for delivery of AAL-SDUs to an AAL user at a constant bit rate.

The handling of timing relation for asynchronous circuit transport is referred to as source clock frequency recovery. Recovered source clock should have satisfactory jitter performance. For example, the jitter performance for Recommendation G.702 signals is specified in Recommendations G.823 and G.824, for which the CS procedure to be used (the SRTS method) is described in 2.5.2.2.l.

2.5.1.2 Functions of the CS for video signal transport

The following functions support transport of video signals for interactive and distributive services.

a) *Handling of AAL user information*

The length of the AAL-SDU is one octet, when utilizing the correction method described in 2.5.2.4.1.

For those AAL users which require transfer of structured data, the STRUCTURE parameter option of primitives defined in 2.1.2 will be used. The CS uses the SDT method described in 2.5.2.3.

As an option, the STATUS parameter defined in 2.1.2 will be passed to the AAL user to facilitate further picture processing, e.g. error concealment.

b) *Handling of cell delay variation*

A buffer is used to support this function. The size of this buffer is dependent upon specifications currently under study.

In the event of buffer underflow, it may be necessary for the CS to maintain bit count integrity by inserting the appropriate number of dummy bits. In the event of buffer overflow, it may be necessary for the CS to maintain bit count integrity by dropping the appropriate number of bits.

c) *Handling of lost and misinserted cells*

The sequence count values are further processed at this sublayer to detect lost and misinserted cells. Detected misinserted cells are discarded. The CS procedure to be used for sequence count processing is described in 2.5.2.1.

In order to maintain the bit count integrity of the AAL user information, it may be necessary to compensate for lost cells detected by buffer underflow and sequence count processing by inserting the appropriate number of dummy SAR-PDU payloads. The content of this dummy SAR-PDU payload depends on the AAL service being provided.

Information in lost cells may be recovered by the mechanism described in e).

d) *Handling of timing relation*

This function is required for delivery of AAL-SDUs to an AAL user at a constant bit rate.

Some AAL users may require source clock frequency recovery, e.g. recovery at the receiving end of camera clock frequency which is not locked to the network clock. The exact method is for further study.

e) *Correction of bit errors and lost cells*

This is an optional function provided for those AAL users requiring bit error and cell loss performance better than that provided by the ATM layer. Examples are unidirectional video services for contribution and distribution. This function may be performed with the CS procedure described in 2.5.2.4.1.

2.5.1.3 Functions of the CS for voice-band signal transport

The following functions support transport of voice-band signals, e.g. 64 kbit/s A-law and μ-law coded Recommendation G.711 signals, and 64 kbit/s Recommendation G.722 signals:

a) *Handling of AAL user information*

The length of the AAL-SDU is one octet.

b) *Handling of cell delay variation*

A buffer is used to support this function. The size of this buffer is dependent upon specifications provided in Recommendation I.356.

c) *Handling of lost and misinserted cells*

The detection of lost and misinserted cells, if needed, may be provided by processing the sequence count values. The monitoring of the buffer fill level can also provide an indication of lost and misinserted cells. Detected misinserted cells are discarded.

Handling of lost cells and buffer underflow is for further study.

NOTE – For transporting signals of speech and 3.1 kHz audio bearer services as specified in 64 kbit/s ISDN, the need for A/μ-law conversion is identified. This conversion function is outside the scope of this Recommendation.

2.5.1.4 Functions of the convergence sublayer for high quality audio signal transport

The capabilities of AAL type 1 are in principle applicable for transfer of high quality audio signals.

2.5.2 Convergence sublayer (CS) protocol

The following subclauses describe CS procedures to be provided for implementing CS functions. The use of each procedure depends on the required CS functions and is given in 2.5.1.1 through to 2.5.1.4.

2.5.2.1 Sequence count operations

2.5.2.1.1 Sequence count operations at the transmitting end

At the transmitting end, the CS provides the SAR with a sequence count value and a CS indication associated with each SAR-PDU payload. The count, value starts with 0, is incremented sequentially and is numbered modulo 8.

2.5.2.1.2 Sequence count operations at the receiving end

At the receiving end, the CS receives from the SAR the following information associated with each SAR-PDU payload:

- sequence count;
- CS indication;
- check status of the sequence count and CS indication.

The use of sequence count values and CS indications will be specified on a service specific basis. See 2.4.2 for details about the check status processing.

The CS processing at the receiving end may identify lost or misinserted SAR-PDU payloads. This will be useful for many CBR services.

CS processing may identify the following conditions:

- SAR-PDU payload sequence normal (i.e. in correct sequence);
- SAR-PDU payload loss;
- SAR-PDU payload misinsertion.

Processing of sequence count values may provide additional information to related entities within the CS, as required. Some examples are:

- location of lost SAR-PDU payload in the incoming SAR-PDU stream;
- number of consecutive SAR-PDU payloads lost;
- identification of misinserted SAR-PDU payload.

NOTE – Processing of sequence count values may be subject to performance specifications. The performance specifications will be applied on a service specific basis.

2.5.2.2 Source clock frequency recovery method

2.5.2.2.1 Synchronous residual time stamp (SRTS) method

a) *General*

The synchronous residual time stamp (SRTS) method uses the residual time stamp (RTS) to measure and convey information about the frequency difference between a common reference clock derived from the network and a service clock. The same derived network clock is assumed to be available at both the transmitter and the receiver. If the common network reference clock is unavailable (e.g. when working between different networks which are not synchronized), then the asynchronous clock recovery method will be in a mode of operation associated with "Plesiochronous network operation" which is described in e). The SRTS method is capable of meeting the jitter specifications of the 2048 kbit/s hierarchy in Recommendation G.823 and the 1544 kbit/s hierarchy in Recommendation G.824.

The following is a description of the SRTS method. The description uses the notation below:

fs --- service clock frequency;

fn --- network clock frequency, e.g. 155.52 MHz;

fnx -- derived network clock frequency, fnx=fn/x, where x is an integer to be defined later;

N ---- period of RTS in cycles of the service clock of frequency fs;

T ---- period of the RTS in seconds;

M(Mnom, Mmax, Mmin) ---- number of fnx cycles within a (nominal, maximum, minimum) RTS period;

Mq --- largest integer smaller than or equal to M.

The SRTS concept is illustrated in Figure 5. In a fixed duration T measured by N service clock cycles, the number of derived network clock cycles Mq is obtained at the transmitter. If Mq is transmitted to the receiver, the service clock of the source can be reconstructed by the receiver, since it has the necessary information: fnx, Mq and N. However, Mq is actually made up of a nominal part and a residual part. The nominal part Mnom corresponds to the nominal number of fnx cycles in T seconds and is fixed for the service. The residual part conveys the frequency difference information as well as the effect of the quantization and thus can vary. Since the nominal part is a constant, it can be assumed that the nominal part of Mq is available at the receiver. Only the residual part of Mq is transmitted to the receiver.

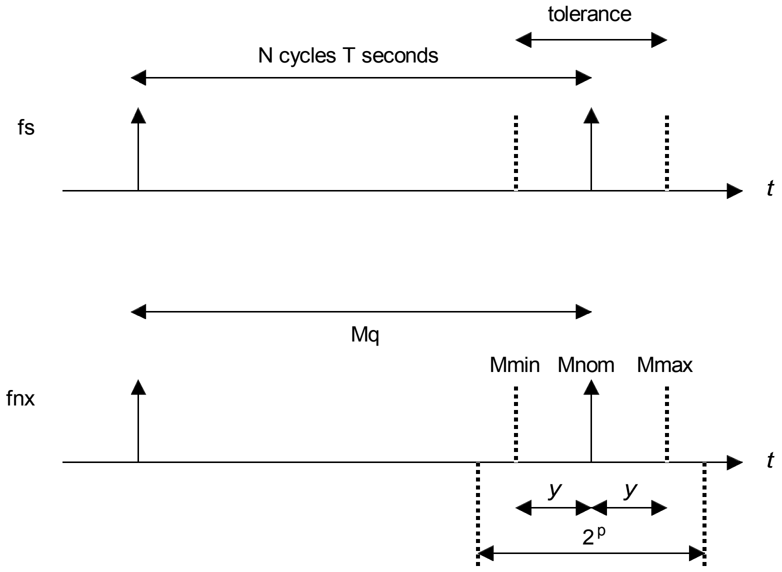


FIGURE 5./I.363

The concept of synchronous residual time stamp (SRTS)

A simple way of representing the residual part of Mq is by means of the RTS, whose generation is shown in Figure 6. Counter Ct is a P-bit counter which is continuously clocked by the derived network clock. The output of counter Ct is sampled every N service clock cycles. This P-bit sample is the residual time stamp.

With a knowledge of the RTS and the nominal part of Mq at the receiver, Mq is completely specified. Mq is used to produce a reference timing signal for a phase-locked loop to obtain the service clock.

b) *Choice of parameter*

The minimum size of the RTS required to unambiguously represent the residual part of Mq is a function of N, the ratio fnx/fs, and the service clock tolerance, $\pm \varepsilon$. Let y be the difference between Mnom and the maximum or minimum value of M (denoted as Mmax, Mmin). The difference y is given by

$$y = N * fnx / fs * \varepsilon.$$

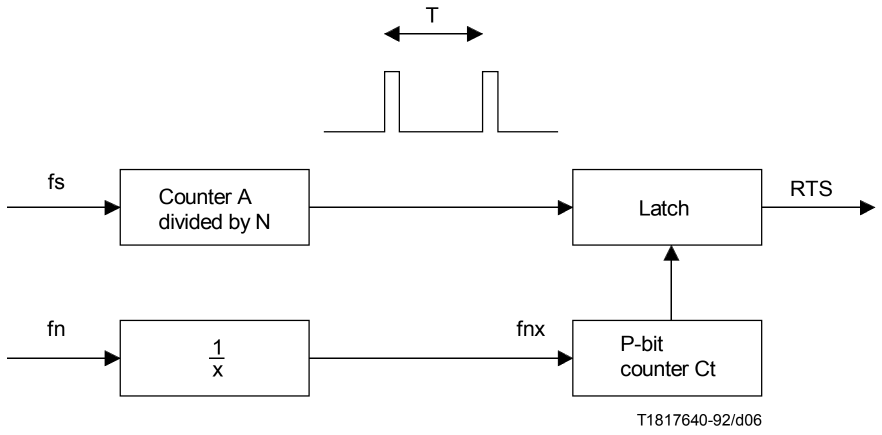


FIGURE 6/I.363

Generation of residual time stamp (RTS)

In order that Mq can be unambiguously identified, the following conditions must be satisfied (see Figure 5):

$$2 (p - 1) > [y],$$

where $[y]$ denotes the smallest integer larger than or equal to y.

The following parameter values are used for the asynchronous circuit transport of Recommendation G.702 signals:

N = 3008 (total number of bits in eight SAR-PDU payloads),

$1 < fnx/fs \leq 2$,

Tolerance = $200 * 10^{-6}$

Size of RTS = 4 bits

The introduction of any AAL convergence sublayer overhead into the SAR-PDU payload will reduce the amount of payload available for the transport of AAL user data. This will reduce the number of service clock cycles over which the RTS period is specified, since the RTS period is defined over a fixed number of SAR-PDU payloads. The RTS period parameter, N, can be adjusted to accommodate such cases. For example, if four octets of CS overhead are required from every eight SAR-PDU payloads, then N would be reduced from 3008 to 2976. However, the CS overhead has to be allocated so that the RTS period always remains a constant number of service clock cycles. Therefore, the CS overhead must reduce the user data transport capacity by a constant amount over the fixed number of SAR-PDU payloads for which the RTS period is defined. See 2.5.2.3.2 for an example.

c) *Network clocks*

For an SDH network, a 155.520 MHz network clock (fn) is available from which the following clocks can be derived:

$$155.520 \text{ MHz} * 2^{-k}, k = 0, 1, ..., 11$$

As an example, to support service rates of 64 kbit/s the fnx will be 155.520 MHz $* 2^{-11}$ (i.e. 75.9375 kHz).

This set of derived network clocks can accommodate all service rates ranging from 64 kbit/s up to the full capacity of the STM-1 payload. The derived network clock to be used for a given service rate is uniquely specified, since the frequency ratio is constrained by $1 < fnx/fs \leq 2$.

Administrations/ROAs may use existing network clocks to support national service in a non-SDH ATM network.

d) *Transport of the RTS*

The 4-bit RTS is transmitted in the serial bit stream provided by the CSI bit in successive SAR-PDU headers. The modulo 8 sequence count provides a frame structure over 8 bits in this serial bit stream. Four bits of the framed 8 bits are allocated for the RTS and the remaining 4 bits are available for other uses. If the four bits available for other uses are not utilized, they are set to 0. The SAR-PDU headers with the odd sequence count values of 1, 3, 5 and 7 are used for RTS transport. The MSB of the RTS is placed in the CSI bit of the SAR-PDU header with the sequence count of 1.

e) *Plesiochronous network operation*

The issue about the accommodation of plesiochronous operation (i.e. when a common reference clock is not available from the network) needs to be addressed. This scenario must be accommodated in such a way that the recovered clock satisfies the jitter requirements specified in Recommendations G.823 and G.824 for Recommendation G.702 signals. However, the detailed method of dealing with plesiochronous operation is not standardized.

2.5.2.2.2 Adaptive clock method

The following is a general description of the method. The receiver writes the received information into a buffer and then reads it with a local clock. The fill level of the buffer is used to control the frequency of the local clock. The control is performed by continuously measuring the fill level around its medium position, and by using this measure to drive the phased-locked loop providing the local clock. The fill level of the buffer may be maintained between two limits in order to prevent buffer overflow and underflow.

2.5.2.3 Structured data transfer (SDT) method

2.5.2.3.1 SDT without use of SRTS

The CS procedure for structured data transfer uses a pointer to delineate the structure boundaries. The procedure supports any fixed, octet-based structure. In particular, it supports 8 kHz based structures used in circuit-mode services of Recommendation I.231.

The procedure description given here is intended for data transfer which does not use the SRTS method (see 2.5.2.2.1) for recovery of the user clock. However, since the SDT method and the SRTS method use the CS indication in alternating SAR-PDU payloads, it is possible to use the two procedures simultaneously to support both structured data transfer and SRTS clock recovery. This combined use is described in the next subclause.

The STRUCTURE parameter in the AAL-UNITDATA-REQUEST and AAL-UNITDATA-INDICATION primitives is used to convey structure information between the AAL and the AAL user. See 2.1.2 for definition of primitives and parameters.

The 47 octet SAR-PDU payload used by the CS has two formats, called non-P and P format, as shown in Figure 7.

a) *Operations of the non-P format*

In the non-P format the entire CS-PDU is filled with user information.

b) *Operations of the P format*

In the P format, the first octet of the SAR-PDU payload is the pointer field. The remainder is filled with user information. This format may be used, only if the sequence count value in the SAR-PDU header is 0, 2, 4 or 6.

The format of the pointer field is shown in Figure 8.

The pointer field contains the binary value of the offset, measured in octets, between the end of the pointer field and the first start of the structured block in the 93 octet payload consisting of the remaining 46 octets of this SAR-PDU payload and the 47 octets of the next SAR-PDU payload. This offset ranges between 0 and 92 inclusive. Moreover, the offset value 93 is used to indicate that the end of the 93 octet payload coincides with the end of a structured block whose start does not lie in the 93 octet payload.

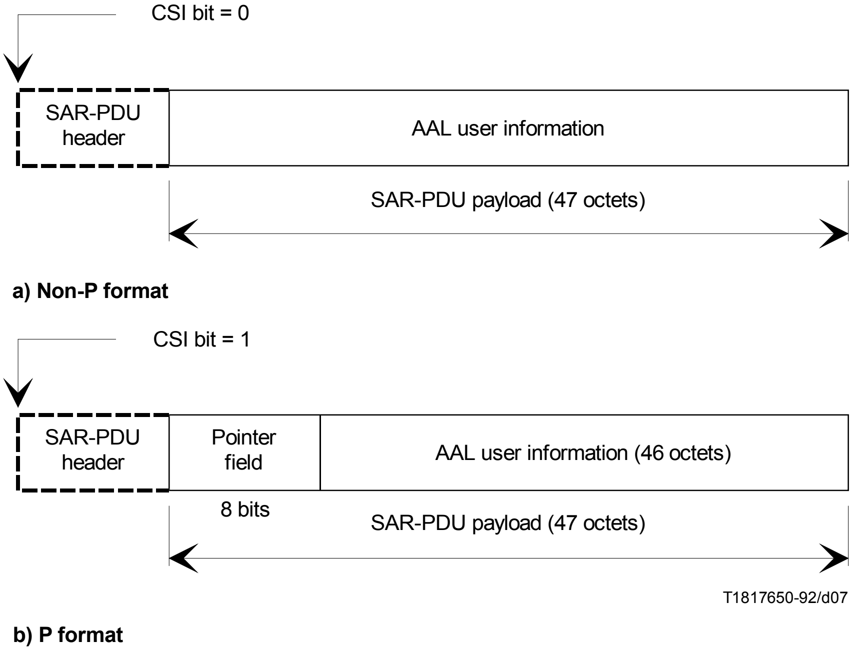


FIGURE 7/I.363

Format of SAR-PDU payload for structured data transfer method

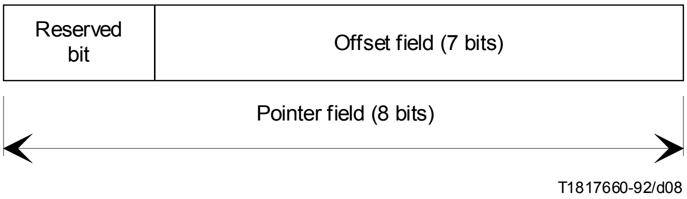


FIGURE 8/I.363

Pointer field format

The binary value of the offset is inserted right justified in the offset field, i.e., the least significant bit of the offset is transmitted last. The first bit of the pointer field is reserved for future standardization and is not used for the offset; this bit is set to 0.

The pointer should be used as often as necessary to ensure that the structure recovery is robust. The frequency of pointer utilization is an item for further study.

NOTE – The receiving CS must know the payload size of a lost SAR-PDU payload in order to maintain correct bit count and correct block delineation. When such a SAR-PDU has an even sequence count value, the number of octets to be inserted is 46 or 47 depending on the presence of the pointer field. There is a need to specify a method which assists the CS in determining whether the pointer field is present. A possible method is to require the transmitting CS to use the pointer field in a systematic manner (e.g. periodically). The exact method is for further study.

The first structured block to be transmitted after the AAL connection is established uses the P format with sequence count value in the SAR-PDU header equal to 0 and with the first octet of the structured data placed in the second octet of the SAR-PDU payload.

Partially filled cells

The SAR-PDU payload may be filled only partially with user data in order to reduce the cell payload assembly delay. In this case, the number of leading octets utilized for user information (excluding pointer field) in each SAR-PDU payload is a constant which is determined by the allowable cell payload assembly delay. The remainder of the SAR-PDU payload consists of dummy octets. The value of the dummy octet is for further study.

The offset value in the pointer field includes all octets of the SAR-PDU payload regardless of whether the octets are utilized for user data or consist of dummy data.

2.5.2.3.2 SDT with use of SRTS

The CS procedure for supporting structured data transfer together with SRTS clock recovery is basically a simple combination of the CS procedures of 2.5.2.2.1 and 2.5.2.3.1.

The 47 octet SAR-PDU payload uses the two formats shown in Figure 7.

a) *Operations of the non-P format*

The non-P format is used if the sequence count value within the SAR-PDU header is 1, 3, 5 or 7. The CS indication bits carry the RTS value as described in 2.5.2.2.1. The 47 octets of the SAR-PDU payload are filled with user information.

b) *Operations of the P format*

The P format is used if the sequence count value within the SAR-PDU header is 0, 2, 4 or 6. The first octet of the SAR-PDU payload is the pointer field and the remainder is filled with user information.

If pointer action is not needed for delineating a structured block contained in this SAR-PDU payload or in the next SAR-PDU payload, then the seven bits denoting the offset are set to the dummy value of all ones. The CS indication is set to 1 because the pointer field is present.

If pointer action is needed for delineation, the offset and pointer operation are as described in 2.5.2.3.1.

The first structured block to be transmitted after the AAL connection is established uses the P format with sequence count value in the SAR-PDU header equal to 0 and with the first octet of the structured data placed in the second octet of the SAR-PDU payload.

2.5.2.4 Correction method for bit errors and lost cells

Other methods are for further study.

2.5.2.4.1 Correction method for bit errors and cell losses for unidirectional video services

This correction method combines forward error correction (FEC) and octet interleaving, from which a CS-PDU structure is defined. FEC uses the Reed-Solomon (128,124) code which is able to correct up to 2 errored symbols (octets) or 4 erasures in the block of 128 octets. An erasure is an errored octet whose location in the block is known. The specific polynomials to be used for Reed-Solomon code are for further study. In the transmitting CS, the 4 octet Reed-Solomon code is appended to 124 octets of incoming data from the upper layer. The resulting 128 octet long blocks are then forwarded to the octet interleaver. See Figure 9 for format of the interleave matrix.

The octet interleaver is organized as a matrix of 128 columns and 47 rows. The interleaver is used as follows; at the input, incoming 128 octet long blocks are stored row by row (one block corresponding to one row); at the output, octets are read out column by column. The matrix has 128 × 47 = 6016 octets, corresponding to 128 SAR-PDU payloads. These 128 SAR-PDU payloads constitute one CS-PDU.

In this process, the loss of one SAR-PDU payload in the matrix implies one erasure, to correct in each row of the matrix. Erasures correspond to dummy cell payloads inserted in the cell flow when a cell loss has been detected. Misinserted cells which have been detected are merely discarded in the CS.

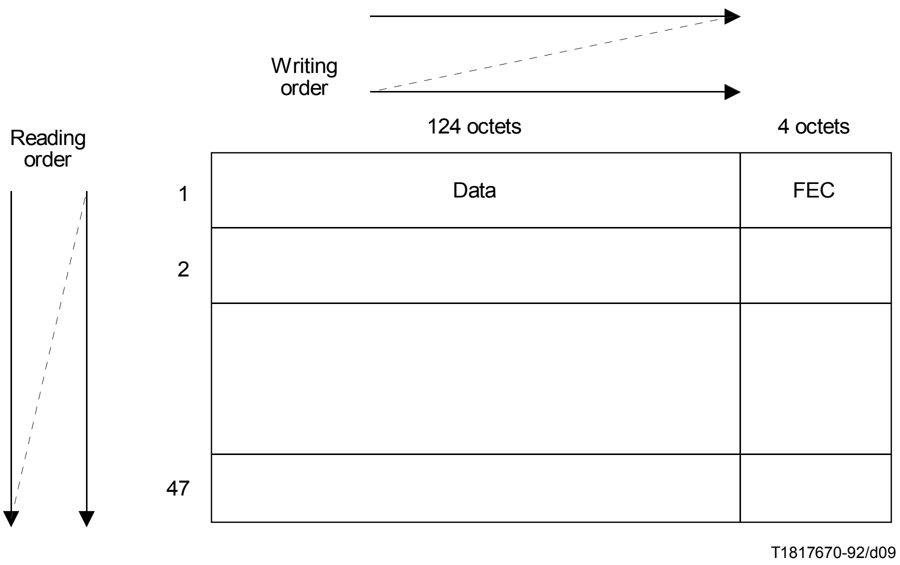


FIGURE 9/I.363

Format of the interleave matrix

For the synchronization of the CS-PDU, the CS indicator bit of the SAR-PDU header is set to 1 for the first SAR-PDU payload of the CS-PDU. This use of the CS indication bit precludes the use of the SDT method as specified in 2.5.2.3.

Within any CS-PDU matrix, this method can perform the following corrections:

- 4 cell losses ; or
- 2 cell losses and 1 errored octet in each row; or
- 2 errored octets in each row if there is no cell loss.

The overhead of this method is 3.1 %, and the delay is 128 cells.

3 AAL type 2

3.1 Service provided by AAL type 2

3.1.1 Definitions

The layer services provided by AAL type 2 to the AAL user may include:

- transfer of service data units with a variable source bit rate;
- transfer of timing information between source and destination;
- indication of lost or errored information which is not recovered by AAL type 2, if needed.

3.1.2 Primitives

For further study.

3.2 Interaction with the management and control planes

3.2.1 Management plane

The following indications may be passed from the user plane to the management plane:

- errors in the transmission of user information;
- lost or misinserted cells (further study is required on whether it is necessary to distinguish between lost and misinserted cells for management purposes);

- cells with errored AAL Protocol Control Information (AAL-PCI) (further study is required to determine if this indication is necessary for layer services supported by this AAL type);
- loss of timing and synchronization;
- buffer underflow and overflow.

3.2.2 Control plane

For further study.

3.3 Functions of AAL type 2

The following functions may be performed in the AAL type 2 in order to enhance the ATM layer service:

a) segmentation and reassembly of user information;

b) handling of cell delay variation;

c) handling of lost and misinserted cells;

d) source clock frequency recovery at the receiver,

e) recovery of the source data structure at the receiver,

f) monitoring of AAL-PCI for.bit errors;

g) handling of AAL-PCI bit errors;

h) monitoring of user information field for bit errors and possible corrective action

Other functions are for further study.

3.4 Segmentation and Reassembly (SAR) sublayer

3.4.1 Functions of the SAR sublayer

For further study.

The SAR sublayer functions are performed on an ATM-SDU basis. As the SAR accepts variable length CS-PDUs from the convergence sublayer, the SAR-PDUs may need to be partially filled.

3.4.2 SAR protocol

For further study.

3.5 Convergence Sublayer (CS)

3.5.1 Functions of the CS

For further study.

3.5.2 CS protocol

For further study.

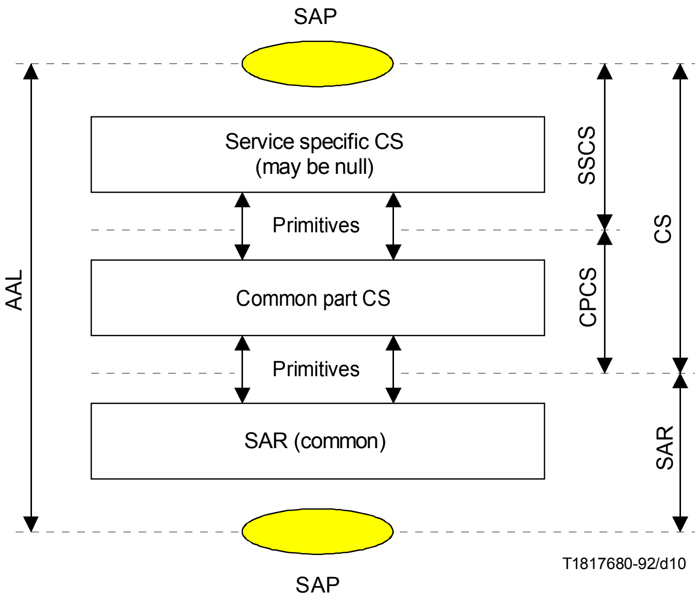
4 AAL type 3

As the enhanced specification for AAL types 3 and 4 are now equivalent, the texts have been merged and referred to as AAL type 3/4.

4.0 Framework of AAL type 3/4

The convergence sublayer (CS) has been subdivided into the common part convergence sublayer (CPCS) and the service specific convergence sublayer (SSCS) as shown in Figure 10. Further clarification can be found in Annex B.

Different SSCS protocols, to support specific AAL user services, or groups of services, may be defined. The SSCS may also be null, in the sense that it only provides for the mapping of the equivalent primitives of the AAL to CPCS and vice-versa.



CS	Convergence sublayer
CPCS	Common part convergence sublayer
SAR	Segmentation and reassembly sublayer
SAP	Service access point
SSCS	Service specific convergence sublayer

FIGURE 10/I.363

Structure of the AAL type 3/4

4.1 Service provided by the AAL type 3/4

The AAL type 3/4 provides the capabilities to transfer the AAL-SDU from one user to another AAL user through the ATM network.

Two modes of service are defined: message and streaming.

a) *Message mode service*

The AAL service data unit is passed across the AAL interface in exactly one AAL interface data unit (AAL-IDU). This service provides the transport of fixed size or variable length AAL-SDUs.

i) In case of small fixed size AAL-SDUs an internal blocking/deblocking function in the SSCS may be applied; it provides the transport of one or more fixed size AAL-SDUs in one SSCS-PDU.

ii) In case of variable length AAL-SDUs an internal AAL-SDU message segmentation/reassembling function in the SSCS may be applied. In this case, a single AAL-SDU is transferred in one or more SSCS-PDUs.

iii) Where the above options are not used, a single AAL-SDU is transferred in one SSCS-PDU. When the SSCS is null, the AAL-SDU is mapped to one CPCS-SDU.

b) *Streaming mode service*

The AAL-SDU is passed across the AAL interface in one or more AAL-IDU. The transfer of these AAL-IDUs across the AAL interface may occur separated in time. This service provides the transport of variable length AAL-SDUs. The streaming mode service includes an abort service by which the discarding of an AAL-SDU partially transferred across the AAL interface can be requested.

i) An internal AAL-SDU message segmentation/reassembling function in the SSCS may be applied. In this case, all the AAL-IDUs belonging to a single AAL-SDU are transferred in one *or more* SSCS-PDU.

ii) An internal pipelining function may be applied. It provides the means by which the sending AAL entity initiates the transfer to the receiving AAL entity before it has the complete AAL-SDU available.

iii) Where option i) is not used, all the AAL-IDUs belonging to a single AAL-SDU are transferred in one SSCS-PDU. When the SSCS is null, the AAL-IDUs belonging to a single AAL-SDU are mapped to one CPCS-SDU.

A summary of the options applicable to the modes of services described above is found in Tables 3 and 4.

TABLE 3/I.363

Combination of service mode and internal function

	AAL-SDU message segmentation/reassembly in the SSCS	AAL-SDU blocking/deblocking in the SSCS	Pipelining
Message Option 1 Option 2	 O N/A	 N/A O	 N/A N/A
Streaming	O	N/A	O
Option 1 Long variable size SDUs Option 2 Short fixed size SDUs O Optional N/A Not applicable			

TABLE 4/I.363

Combination of service mode at the sending and receiving side

Receiver	Sender		
	MM/Block	MM/Seg	SM
MM/Deblocking	A	N/A	N/A
MM/Reasssembly	N/A	A	A
SM	N/A	A	A
MM Message mode SM Streaming mode A Applicable N/A Not applicable NOTE – An end-to-end specification of the SDU length in message mode with blocking/deblocking is needed.			

Both modes of service may offer the following peer-to- peer operational procedures:

- *Assured operations*

  Every assured AAL-SDU is delivered with exactly the data content that the user sent. The assured service is provided by retransmission of missing or corrupted SSCS-PDUs. Flow control is provided as a mandatory feature. The assured operation may be restricted to point-to-point ATM adaptation layer connections.

- *Non-assured operations*

  Integral AAL-SDUs may be lost or corrupted. Lost and corrupted AAL-SDUs will not be corrected by retransmission. An optional feature may be provided to allow corrupted AAL-SDUs to be delivered to the user (i.e. optional error delivery). Flow control may be provided as an option.

4.1.1 Description of AAL connections

The AAL type 3/4 provides the capabilities to transfer the AAL-SDU from one AAL-SAP to one or more AAL-SAPs through the ATM network (see Figures 11 and 12). The AAL-users will have the capability to select a given AAL-SAP associated with the QOS required, to transport that AAL-SDU (for example, delay and loss sensitive QOS).

AAL type 3/4 makes use of the service provided by the underlying ATM layer (see Figure 13). Multiple AAL connections may be associated with a single ATM layer connection, allowing SAR-PDU multiplexing at the AAL. The AAL user selects the QOS provided by the AAL through the choice of the AAL-SAP used for data transfer.

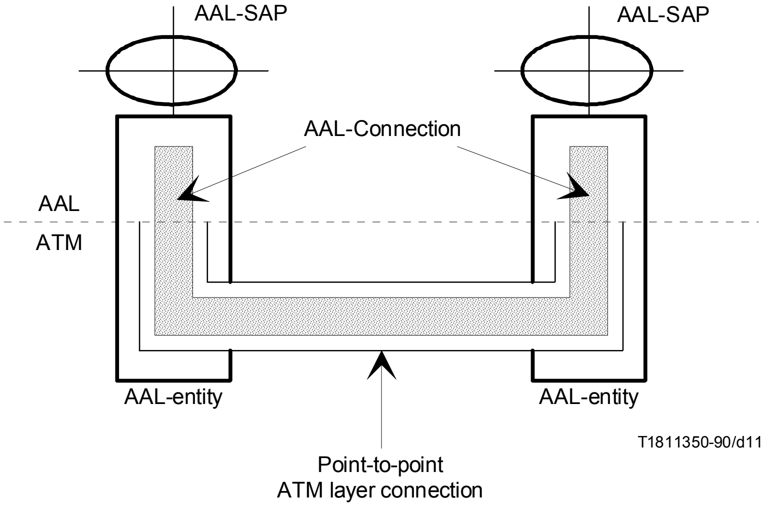


FIGURE 11/I.363

Point-to-point AAL connection

4.1.2 Primitives

The functional model for AAL type 3/4 as contained in Annex C shows the interrelation between the SAR, CPCS and SSCS sublayers, and the SAR and CPCS primitives.

4.1.2.1 Primitives for the AAL

These primitives are service specific and are for further study.

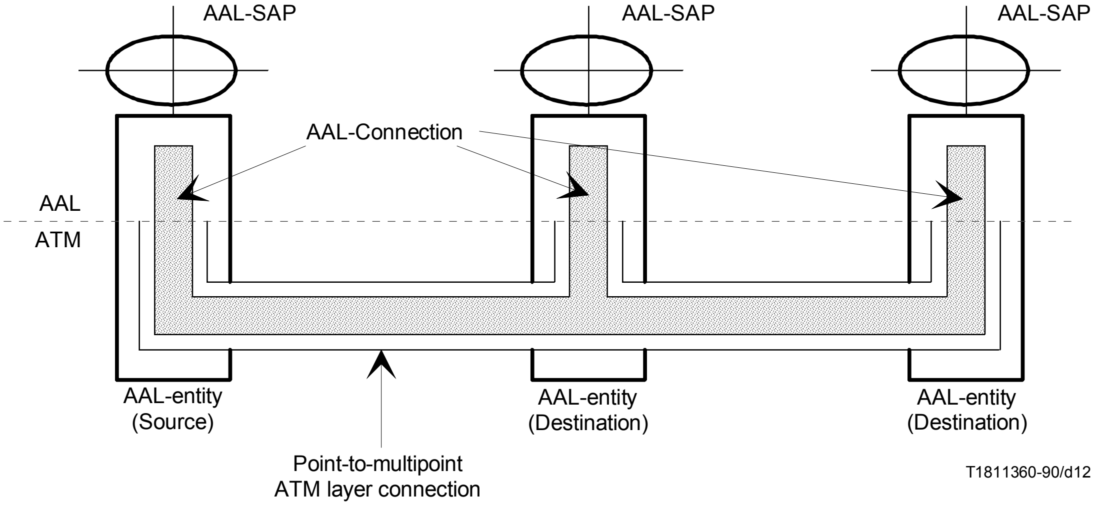
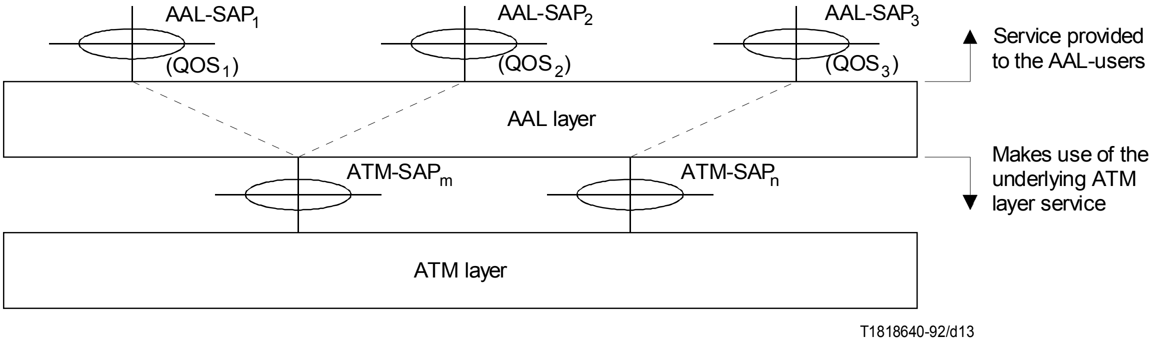


FIGURE 12/I.363

Point-to-multipoint AAL connection



QOS Quality of Service

FIGURE 13/I.363

Relation between AAL-SAP and ATM-SAP

The SSCS may be null, in the sense that it only provides for the mapping of the equivalent primitives of the AAL to CPCS and vice-versa. In this case, the primitives for the AAL are equivalent to those for the CPCS (4.1.2.2) but identified as AAL-UNITDATA-request, AAL-UNITDATA-indication, AAL-U-Abort-request, AAL-U-Abort-indication and AAL-P-Abort-indication, consistent with the primitive naming convention at a SAP.

4.1.2.2 Primitives for the CPCS of the AAL

As there exists no service access point (SAP) between the sublayers of the AAL type 3/4, the primitives are called "invoke" and "signal" instead of the conventional "request" and "indication" to highlight the absence of the SAP.

4.1.2.2.1 Primitives for the data transfer service

– *CPCS-UNITDATA-invoke and the CPCS-UNITDATA-signal*

These primitives are used for the data transfer. The following parameters are defined:

- **interface data (ID)**: This parameter specifies the interface data unit exchanged between the CPCS and the SSCS entity. The interface data is an integral multiple of one octet. If the CPCS entity is operating in the message mode service, the interface data represents a complete CPCS-SDU; when operating in the streaming mode service, the interface data does not necessarily represent a complete CPCS-SDU.
- **More (M)**: In the message mode service, this parameter is not used. In the streaming mode service, this parameter specifies whether the Interface Data communicated contains a beginning/continuation of a CPCS-SDU or the end of / complete CPCS-SDU.
- **Maximum Length (ML)**: In the message mode service, this parameter is not used. In the streaming mode service, this parameter indicates the maximum length of the CPCS-SDU. This parameter is required with the first invoke or signal primitive related to a certain CPCS-SDU; in all other cases, this parameter is not used.
- **Reception Status (RS)**: This parameter indicates that the interface data delivered may be corrupted. This parameter is only utilized if the corrupted data delivery option is used.

Depending on the service mode (message or streaming mode service, discarding or delivery of errored information), not all parameters are required. This is summarized in Table 5.

TABLE 5/I.363

Parameters of the CPCS-UNITDATA

Parameter	Type	MM	SM	Comments
Interface data (ID)	Invoke signal	M M	M M	Whole or partial CPCS-SDU
More (M)	Invoke signal	– –	M M	M = 0 End of CPCS-SDU M = 1 Not end of CPCS-SDU
Maximum length (ML)	Invoke signal	– –	M* O*	Maximum length of CPCS-SDU
Reception status (RS)	Invoke signal	– O	– O	Indication of corrupted data

MM Message mode service
SM Streaming mode service
M Mandatory
O Optional
– Not present
M* Mandatory with the first invoke or signal primitive related to a certain CPCS-SDU, otherwise absent.
O* Optional with the first invoke or signal primitive related to a certain CPCS-SDU, otherwise absent.

4.1.2.2.2 Primitives for the abort service

These primitives are used in the streaming mode service.

a) *CPCS-U-Abort-invoke and CPCS-U-Abort-signal*

This primitive is used by the CPCS user to invoke the abort service. It is also used to signal to the CPCS user that a partially delivered CPCS-SDU is to be discarded by instruction from its peer entity. No parameters are defined.

This primitive is not used in message mode.

b) *CPCS-P-Abort-signal*

This primitive is used by the CPCS entity to signal to its user that a partially delivered CPCS-SDU is to be discarded due to the occurrence of some error in the CPCS or below. No parameters are defined.

This primitive is not used in message mode.

4.1.2.3 Primitives for the SAR sublayer of the AAL

These primitives model the exchange of information between the SAR sublayer and the CPCS.

As there exists no service access point (SAP) between the sublayers of the AAL type 3/4, the primitives are called "invoke" and "signal" instead of the conventional "request" and "indication" to highlight the absence of the SAP.

4.1.2.3.1 Primitives for the data transfer service

– *SAR-UNITDATA-invoke and the SAR-UNITDATA-signal*

These primitives are used for the data transfer. The following parameters are defined:

1) **Interface Data (ID)**: This parameter specifies the interface data unit exchanged between the SAR and the CPCS entity. The interface data is an integral multiple of one octet. The interface data does not necessarily represent a complete SAR-SDU.

2) **More (M)**: This parameter specifies whether the interface data communicated contains the end of the SAR-SDU.

   If the More parameter is set to M=1, the interface data parameter must contain an integral multiple of 44 octets.

3) **Reception Status (RS)**: This parameter indicates that the interface data delivered may be corrupted. This parameter is only utilized if the corrupted data delivery option is used.

4.1.2.3.2 Primitives for the abort service

a) *SAR-U-Abort-invoke and SAR-U-Abort-signal*

This primitive is used by the SAR user to invoke the abort service. It is also used by the SAR entity to signal to the SAR user that a partially delivered SAR-SDU is to be discarded by instruction from its peer entity. This primitive has no parameters.

b) *SAR-P-Abort-signal*

This primitive is used by the SAR entity to signal to its user that a partially delivered SAR-SDU is to be discarded due to the detection of some error. This primitive is only used if the corrupted data delivery option is not used. This primitive has no parameters.

4.2 Interaction with the management and control plane

4.2.1 Management plane

For further study.

4.2.2 Control plane

For further study.

4.3 Functions, structure and coding of AAL type 3/4

4.3.1 Segmentation and Reassembly (SAR) sublayer

4.3.1.1 Functions of the SAR sublayer

The SAR sublayer functions are performed on an SAR-PDU basis. The SAR sublayer accepts variable length SAR-SDUs from the convergence sublayer (CS) and generates SAR-PDUs containing up to 44 octets of SAR-SDU data.

The SAR sublayer functions provide the means for the transfer of multiple variable length SAR-SDUs concurrently over a single ATM layer connection between AAL entities.

a) *Preservation of SAR-SDU*

This function preserves the SAR-SDU by providing for a segment type indication and a SAR-PDU payload length indication. The SAR-PDU payload length indication identifies the number of octets of SAR-SDU information contained within the SAR-PDU payload. The segment type indication identifies a SAR-PDU as a beginning of message (BOM), continuation of message (COM), end of message (EOM), or single segment message (SSM).

b) *Error Detection and Handling*

This function provides the means to detect and handle:

– bit errors in the SAR-PDU;

– lost or gained SAR-PDUs.

SAR-PDUs with bit errors are discarded. An optional feature may be provided to allow corrupted SAR-PDUs to be delivered to the CPCS (i.e. optional error delivery). However, if the optional multiplexing and demultiplexing of SAR connections is performed, such an optional errored delivery service may deliver an errored SAR-SDU to the wrong state machine. SAR-SDUs with lost or gained SAR-PDUs are discarded or are optionally delivered to the CPCS. When delivering errored information, an appropriate indication is associated with the information.

c) *SAR-SDU sequence integrity*

This function assures that the sequence of SAR-SDUs is maintained within one SAR connection.

d) *Multiplexing/demultiplexing*

This function provides for the optional multiplexing and demultiplexing of multiple SAR connections. The number of SAR connections supported over an ATM connection shall be negotiated at connection establishment. The default number of CPCS connections shall be one. Within a given SAR connection, sequence integrity will be preserved.

e) *Abort*

This function provides for the means to abort a partially transmitted SAR-SDU.

4.3.1.2 SAR-PDU structure and coding

The SAR sublayer functions require a 2 octet SAR-PDU header and a 2 octet SAR-PDU trailer. The SAR-PDU header and trailer together with the 44 octets of SAR-PDU payload comprise the 48 octet ATM-SDU (cell payload). The sizes and positions of fields for the SAR-PDU structure are given in Figure 14.

The coding of the SAR-PDU conforms to the coding conventions specified in 2.1/I.361. There are two types of SAR-PDU: Data-SAR-PDUs and Abort-SAR-PDUs.

4.3.1.2.1 Data-SAR-PDU coding

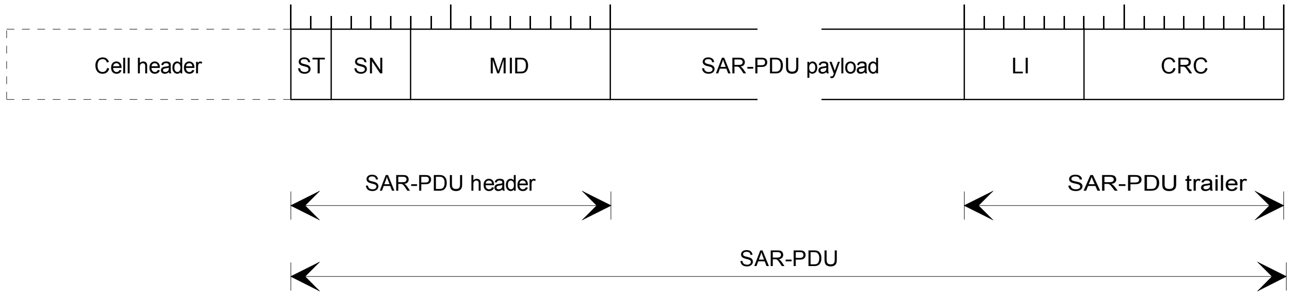
a) *Segment type (ST) field*

The segment type indication identifies a SAR-PDU as containing a beginning of message (BOM), a continuation of message (COM), an end of message (EOM), or a single segment message (SSM). The association between the encoding and the meaning of the segment type field is shown in Table 6.

b) *Sequence number (SN) field*

Four bits are allocated to the sequence Number field allowing the stream of SAR-PDUs of a CPCS-PDU to be numbered modulo 16.

Each SAR-PDU belonging to a SAR-SDU (and hence associated with a given MID value) will have its sequence number incremented by one relative to its previous sequence number. The receiver checks the sequence of the sequence number field of SAR-PDUs derived from one SAR-SDU and does not check the sequence of the sequence number field of the SAR-PDUs derived from successive SAR-SDUs. As the receiver does not check the sequence number continuity between SAR-SDUs, the sender may set the sequence number field to any value from 0 to 15 at the beginning of each SAR-SDU.



ST	Segment type	(2 bits)
SN	Sequence number	(4 bits)
MID	Multiplexing identification	(10 bits)
LI	Length indication	(6 bits)
CRC	Cyclic redundancy check code	(10 bits)

FIGURE 14/I.363

SAR-PDU format for AAL type 3/4

TABLE 6/I.363

Coding of segment type field

Segment type	Encoding	Usage
BOM	10	Beginning of message
COM	00	Continuation of message
EOM	01	End of message
SSM	11	Single segment message

c) *Multiplexing identification (MID) field*

This field is used for multiplexing. If no multiplexing is used, this field shall be set to zero.

In connection oriented applications it may be used to multiplex multiple SAR connections on a single ATM layer connection. The following restrictions may apply:

– Multiplexing/demultiplexing on a single ATM layer connection using the MID field will be on a user-to-user basis.

– A single ATM layer connection containing multiplexed AAL type 3/4 traffic will be administered as a single entity.

In connectionless and connection oriented applications, all SAR-PDUs of a SAR-SDU will have the same MID field value. The MID field is used to identify SAR-PDUs belonging to a particular SAR-SDU. The MID field assists in the interleaving of SAR- PDUs from different SAR-SDUs and reassembly of these SAR-SDUs.

An implementation of AAL type 3/4 is not obliged to support the full range of MID field values. The mechanism for restricting the range of MID field values is for further study. Examples of possible mechanisms would include those based on dynamic negotiation or on signalling.

d) *SAR-PDU payload field*

The SAR-SDU information is left justified within the SAR-PDU payload field. The remaining octets of the SAR-PDU payload field may be set to "0" and are ignored at the receiving end.

e) *Length indication (LI) field*

The length indication field is binary encoded with the number of octets of SAR-SDU information that are included in the SAR-PDU payload field. Permissible values of this field, depending on the coding of the segment type field are shown in Table 7. See also Figure B.3. Combined SAR and CPCS PDU format.

TABLE 7/I.363

Permissible values of the length indication field

Segment type	Permissible value
BOM	44
COM	44
EOM	4 ... 44, 63 (Note)
SSM	8 ... 44
NOTE – The value "63" is used in the Abort-SAR-PDU (see 4.3.1.2.2)	

f) *CRC field*

The CRC field shall be a 10-bit sequence. It shall be the remainder of the division (modulo 2) by the generator polynomial of the product of, x^{10} and the content of the SAR-PDU, including the SAR-PDU header, SAR-PDU payload, and length indication field of the SAR-PDU trailer. Each bit of the concatenated fields mentioned above are considered as coefficients (modulo 2) of a polynomial of degree 373. The CRC-10 generator polynomial is:

$$G(x) = 1 + x + x^4 + x^5 + x^9 + x^{10}$$

The result of the CRC calculation is placed with the least significant bit right justified in the CRC field. The CRC-10 is used to detect bit errors in the SAR-PDU.

4.3.1.2.2 Abort-SAR-PDU coding

The coding of the Abort-SAR-PDU conforms to the structure and coding specified above with the exception that

1) the segment type field shall be coded as EOM;

2) the payload field may be set to zero and is ignored at the receiving end;

3) the length indication field shall be set to 63.

4.3.2 Convergence Sublayer (CS)

4.3.2.1 Functions, structure and coding for the CPCS

The CPCS has the following service characteristics.

– Non-assured transfer of user data frames with any length measured in octets from 1 to 65,535 octets and with the possibility of further extension (how much it can be extended is for further study).

– One or more "CPCS connections" may be established between two CPCS peer entities (no switching of CPCS connections will be supported). The maximum number of CPCS connections that can be established is defined by the end system with the lowest capacity.

- The CPCS connections will be established by management or by the control plane.
- Error detection and indication (cell loss or gain).
- CPCS-SDU sequence integrity on each CPCS connection.

The CPCS has the basic functionality to support a connectionless network access protocol (CLNAP) layer (Class D) as well as a frame relaying telecommunication service in Class C. For the CLNAP layer (Class D) there is no need for any service specific convergence sublayer.

4.3.2.1.1 Functions of the CPCS

The CPCS functions are performed per CPCS-PDU. The CPCS provides several functions in support of the CPCS service user. Some of the functions provided depend on whether the CPCS service user is operating in message or streaming mode.

i) *Message mode service*

   The CPCS-SDU is passed across the CPCS interface in exactly one CPCS-IDU. This service provides the transport of a single CPCS-SDU in one CPCS-PDU.

ii) *Streaming mode service*

   The CPCS-SDU is passed across the CPCS interface in one or more CPCS-IDUs. The transfer of these CPCS-IDUs across the CPCS interface may occur separated in time. This service provides the transport of all the CPCS-IDUs belonging to a single CPCS-SDU into one CPCS-PDU. The streaming mode service includes an abort service by which the discarding of a CPCS-SDU partially transferred across the interface can be requested.

The functions implemented by the CPCS include:

a) *Preservation of CPCS-SDU*

   This function provides for the delineation and transparency of CPCS-SDUs.

b) *Error detection and handling*

   This function provides for the detection and handling of CPCS-PDU corruption. Corrupted CPCS-SDUs are either discarded or are optionally delivered to the SSCS. The procedures for delivery of corrupted CPCS-SDUs are for further study. When delivering errored information to the CPCS user, an error indication is associated with the delivery.

   Examples of detected errors would include: Btag/Etag mismatch, received length and CPCS-PDU length field mismatch, buffer overflow, improperly formatted CPCS-PDU, and errors indicated by the SAR sublayer.

c) *Buffer allocation size*

   This function provides for the indication to the receiving peer entity of the maximum buffering requirements to receive the CPCS-PDU.

d) *Abort*

   This function provides for the means to abort a partially transmitted CPCS-SDU.

   Other functions are for further study.

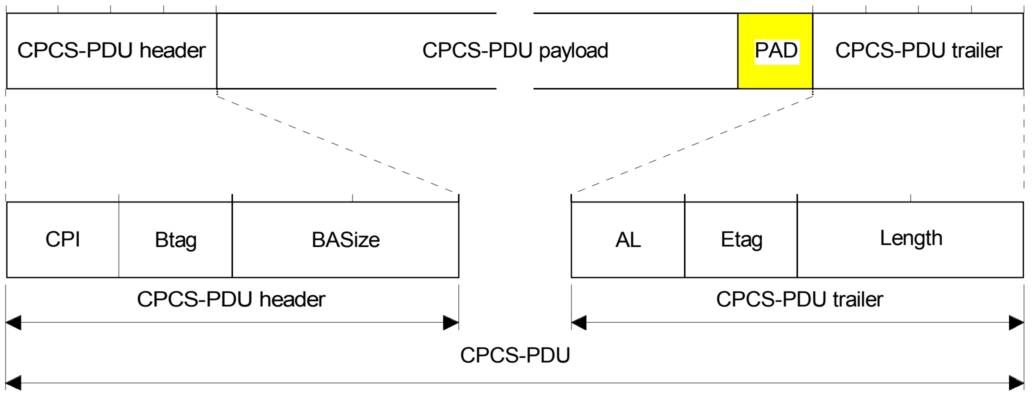
4.3.2.1.2 CPCS structure and coding

The CPCS functions require a 4 octet CPCS-PDU header and a 4 octet CPCS-PDU trailer. In addition, a padding field provides for a 32 bit alignment of the CPCS-PDU payload. The CPCS-PDU header and trailer together with the padding field and the CPCS-PDU payload comprise the CPCS-PDU. The sizes and positions of fields for the CPCS-PDU structure are given in Figure 15.

The coding of the CPCS-PDU conforms to the coding conventions specified in 2.1/I.361.

a) *Common part indicator (CPI) field*

   The CPI field is used to interpret subsequent fields for the CPCS functions in the CPCS-PDU header and trailer. The counting units for the values specified in the BAsize and length fields may be indicated; other uses are for further study. These uses shall be restricted to the CPCS and SAR sublayer functions including the means to identify related AAL layer management messages. These messages in the future could be used to perform layer management functions which may include: performance and fault monitoring, MID allocation, and transfer of OAM messages.



CPI	Common part indicator	(1 octet)
Btag	Beginning tag	(1 octet)
BASize	Buffer allocation size	(2 octets)
PAD	Padding	(0 ... 3 octets)
AL	Alignment	(1 octet)
Etag	End tag	(1 octet)
Length	Length of CPCS-PDU payload	(2 octets)

FIGURE 15/I.363

CPCS-PDU format for AAL type 3/4

Table 8 shows the agreed coding of the CPI field and indicates the related semantics of the BAsize and length fields. Additional encodings and uses of the CPI field are for further study.

TABLE 8/I.363

CPI field encoding

CPI encoding	BAsize field semantics	Length field semantics
00000000	Buffer allocation requirements in octets	Equals length of CPCS-PDU payload in octets
Other values are reserved and are for future standardization	For further study	For further study

b) *Beginning tag (Btag) field*

This field allows the association of the CPCS-PDU header and trailer. The sender inserts the same value in the Btag and the Etag in the trailer for a given CPCS-PDU and changes the value for each successive CPCS-PDU. The receiver checks the value of the Btag in the CPCS header with the value of the Etag in the CPCS trailer. It does not check the sequence of the Btag/Etags in successive CPCS-PDUs.

As an example, a suitable mechanism is as follows: The sender increments the value placed in the Btag and Etag fields for each successive CPCS-PDU sent over a given MID value. Btag values are cycled up to modulo 256.

c) *Buffer allocation size indication (BAsize) field*

The BAsize field indicates to the receiving peer entity the maximum buffering requirements to receive the CPCS-SDU. In message mode the BAsize value is encoded equal to the CPCS-PDU payload length. In streaming mode, the BAsize value is encoded equal to or greater than the CPCS-PDU payload length.

The buffer allocation size is binary encoded as number of counting units. The size of the counting units is identified by the CPI field.

NOTE – The length of the CPCS-PDU payload is limited to the maximum value of the BAsize field multiplied by the value of the counting unit.

d) *Padding (PAD) field*

Between the end of the CPCS-PDU payload and the 32 bit aligned CPCS-PDU trailer, there will be from 0 to 3 unused octets. These unused octets are called the padding (PAD) field; they are strictly used as filler octets and do not convey any information. It may be set to "0" and its value is ignored at the receiving end. This padding field complements the CPCS-PDU payload to an integral multiple of 4 octets.

The function of the PAD field is shown in Figure 16.

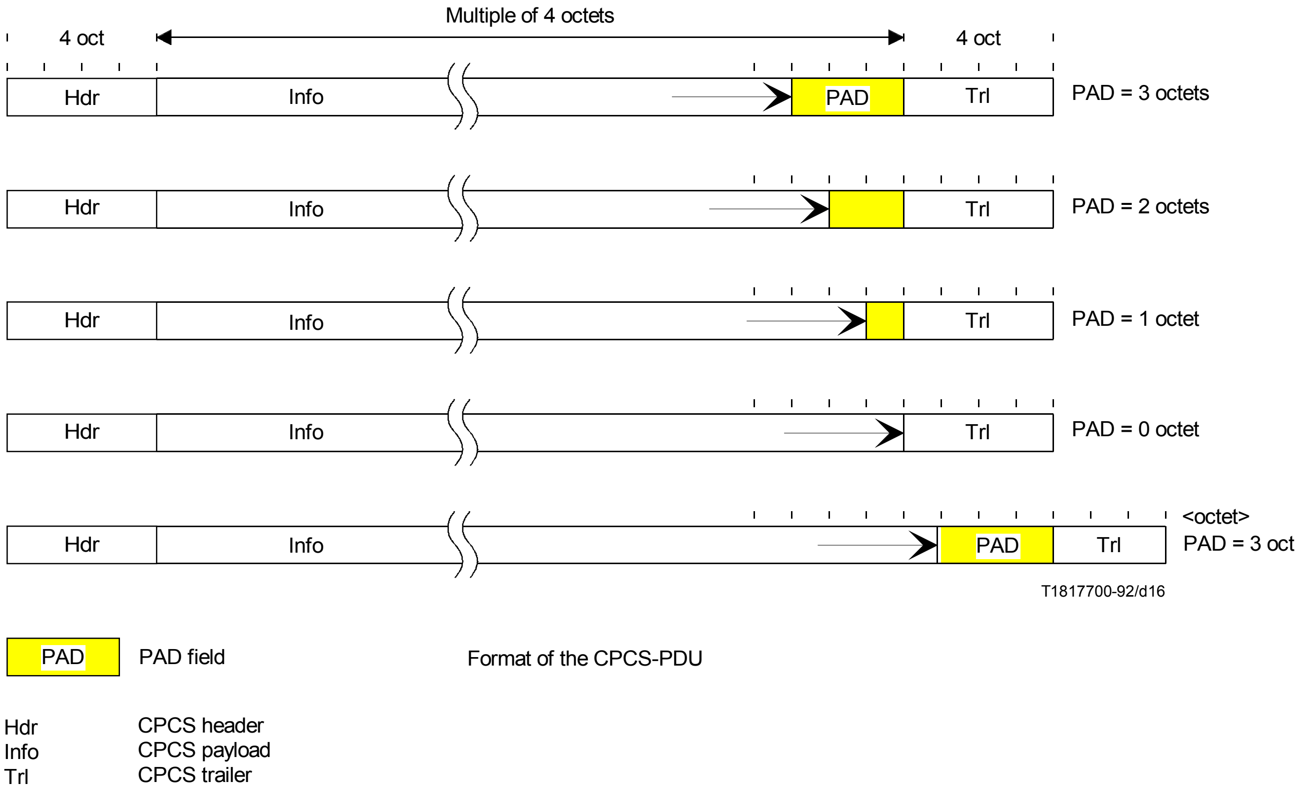


FIGURE 16/I.363

Function of the PAD field

e) *Alignment (AL) field*

The function of the alignment field is to achieve 32-bit alignment in the CPCS-PDU trailer. The alignment field complements the CPCS-PDU trailer to 32 bits. This unused octet is strictly used as a filler octet and does not convey any information.

The alignment field shall be set to zero.

f) *End tag (Etag) field*

For a given CPCS-PDU, the sender shall insert the same value in this field as was inserted in the Btag field in the CPCS-PDU header to allow the association of the CPCS-PDU trailer with its CPCS-PDU header.

g) *Length field*

The length field is used to encode the length of the CPCS-PDU payload field. This field is also used by the receiver to detect the loss or gain of information.

The length is binary encoded as number of counting units. The size of the counting units is identified by the CPI field.

NOTE – The length of the CPCS-PDU payload is limited to the maximum value of the length field multiplied by the value of the counting unit.

4.3.2.2 Functions, structure and coding for the SSCS

The CPCS has the basic functionality to support a connectionless network layer (Class D) as well as a frame relaying telecommunication service in Class C. For the connectionless network layer (Class D) there is no need for any service specific convergence sublayer. Otherwise the functions, structure and coding for the SSCS are for further study.

4.4 Procedures

There exists one segmentation and reassembly state machine per multiplexing identification (MID) field value. For each such state machine, the value of this field must be known by the protocol state machines.

4.4.1 Procedures of the SAR sublayer

The structure and coding of the SAR-PDU is defined in 4.3.1.2.

4.4.1.1 State variables of the SAR sublayer at the sender side

The SAR sender maintains the following state variable:

- *snd_SN*

  This variable is used to set the sequence number field of the SAR-PDU header. It is incremented modulo 16 after each SAR-PDU of a SAR-SDU has been forwarded to the ATM layer for transmission.

4.4.1.2 Procedures of the SAR sublayer at the sender side

The state machine of the SAR sender is shown in Figure 17.

Table 9 defines the states for the SAR sender.

TABLE 9/I.363

State definitions for the SAR sender

State	Definition
IDLE	Waiting to begin to transmit a new SAR-SDU
STREAM	Transmitting a SAR-SDU in streaming mode

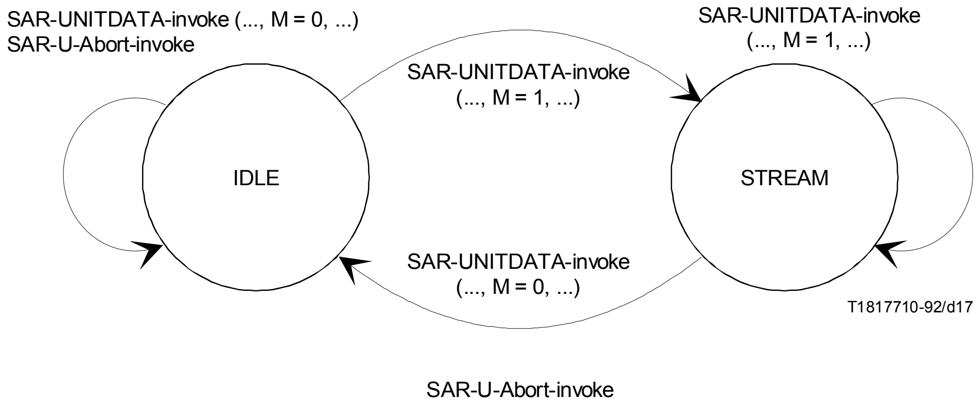


FIGURE 17/I.363

State transition diagram for the SAR sender

1) When the SAR connection is established, the SAR sender shall proceed to the IDLE state. Whenever entering the IDLE state, the SAR sender may change its state variable snd_SN to any value from 0 to 15.
2) For each SAR-PDU, the SAR sender shall set the MID field to the values governing this state machine. The sequence number field is set to the value of the state variable snd_SN and the state variable snd_SN is incremented by one (modulo 16).
3) Upon receiving a SAR-UNITDATA-invoke primitive from the CPCS, the SAR sender shall start the segmenting process. If the Interface Data has a length of more than 44 octets, the SAR sender will generate more than one SAR-PDU. In all SAR-PDUs (except possibly the last one), the SAR-PDU payload field shall be filled with 44 octets of CPCS-PDU information.
4) In each SAR-PDU, the length indication field shall be set to the number of octets of SAR-SDU data carried in the payload and the CRC field shall be computed as specified in 4.3.1.2.
5) If the SAR sender is in the IDLE state, it shall set the most significant bit of the segment type field in the first SAR-PDU to "1" ("BOM" or "SSM"); in all subsequent SAR-PDUs, this bit shall be set to "0" ("COM" or "EOM"). If the SAR sender is in the STREAM state, the most significant bit of the ST field of all SAR-PDUs shall be set to "0".
6) If the M parameter in the SAR-UNITDATA-invoke primitive has the value "0", the SAR sender shall set the least significant bit of the segment type field in the last SAR-PDU to "1" ("EOM" or "SSM"); in all other cases, this bit shall be set to "0" ("BOM" or "COM").
7) Upon completion of the segmenting process, the SAR sender shall proceed either to the IDLE state or the STREAM state. If the M parameter in the SAR-UNITDATA-invoke primitive has the value "0", the SAR sender shall proceed to the IDLE state; otherwise, it shall proceed to the STREAM state.
8) The SAR sender shall ignore a SAR-U-Abort-invoke primitive when it is in the IDLE state. When in the STREAM state, the SAR sender shall generate and transmit an Abort-SAR-PDU and proceed to the IDLE state.

NOTE – This description of the SAR sender procedures is valid for all service modes of the CPCS. If the CPCS passes only complete CPCS-PDUs to the SAR sublayer, the state machine remains always in the IDLE state.

4.4.1.3 State variables of the SAR sublayer at the receiver side

The SAR receiver maintains the following state variable:

– *rcv_SN*

This variable is used to detect loss or gain of SAR-PDUs. After the receipt of an SAR-PDU with a segment type field that indicates "COM" or "EOM", the SAR receiver compares the value in the sequence number field with this state variable. If they are equal, the SAR-PDU is assumed to be in sequence and the rcv_SN is incremented by one modulo 16.

If the segment type field of a SAR-PDU indicates “BOM” or “SSM”, the sequence number field is not compared with rcv_SN; however, the state variable rcv_SN is set to one greater (modulo 16) than the value in the sequence number field.

4.4.1.4 Procedures of the SAR sublayer at the receiver side

The state machine of the SAR receiver is shown in Figure 18.

Table 10 defines the states for the SAR receiver.

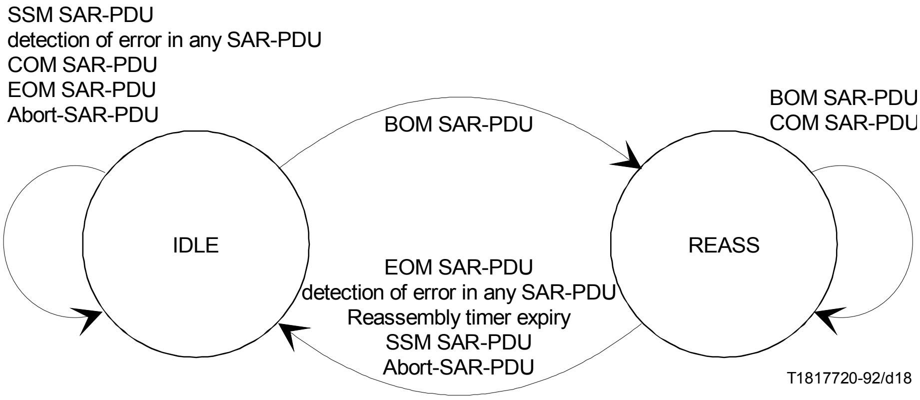


FIGURE 18/I.363

State transition diagram for the SAR receiver

TABLE 10/I.363

State definitions for the SAR receiver

State	Definition
IDLE	Waiting to begin to receive a new SAR-SDU
REASS	Receiving a SAR-SDU

The following procedures are specified for an SAR receiver that does not deliver errored data to the receiving CPCS. The procedures describing the delivery of errored information are for further study.

NOTE – The term “delivery to the CPCS” refers to the communication across the SAR - CPCS sublayer boundary via a SAR-UNITDATA-signal primitive.

1) All illegal SAR-PDUs are ignored. An illegal SAR-PDU is a SAR-PDU with either
   - a CRC verification error; or
   - an unexpected MID field value.

   NOTE – The discarding of illegal SAR-PDUs actually takes place prior to assigning the SAR-PDU to a reassembly process governed by a particular MID field value.

2) For every SAR-PDU received, the SAR receiver verifies that the value of the length indication field is permissible given the coding of the segment type field (c.f. Table 7 "Permissible values of the Length Indication Field"). If the value is outside the allowed range, the SAR-PDU is discarded. If the SAR receiver is in the REASS state, it shall issue a SAR-P-Abort-signal primitive to the receiving CPCS. In all cases, it shall proceed to the IDLE state.

3) In the absence of errors and irrespective of the state in which the SAR receiver is, the number of octets indicated in the length indication field are sent from the SAR-PDU payload to the CPCS. If the segment type field indicates "EOM" or "SSM", the M parameter is set to "0" and the SAR receiver proceeds to the IDLE state; otherwise, if the segment type field indicates "BOM" or "COM", the M parameter is set to "1" and the SAR receiver proceeds to or remains in the REASS state.

The following error recovery procedures apply:

4) If the SAR receiver is in the IDLE state and receives a SAR-PDU whose segment type field indicates "COM" or "EOM", the SAR receiver shall ignore the SAR-PDU.

5) If the SAR receiver is in the REASS state and receives a SAR-PDU whose segment type field indicates "BOM" or "SSM", the SAR receiver shall issue a SAR-P-Abort-signal to the receiving CPCS; the SAR-PDU shall be processed normally as described in 3) above.

6) If the SAR receiver is in the REASS state and it receives a SAR-PDU whose value in the sequence number field is not the same as the value of the state variable rcv_SN, it shall issue a SAR-P-Abort-signal to the receiving CPCS, in addition, if the segment type field indicates "COM" or "EOM", the SAR-PDU is discarded and the SAR receiver shall proceed to the IDLE state; otherwise, the SAR-PDU shall be processed normally as described in 3) above.

7) If the SAR receiver receives an Abort-SAR-PDU and is in the IDLE state, this SAR-PDU shall be ignored; if in the REASS state, the SAR receiver shall issue a SAR-U-Abort-signal primitive and proceed to the IDLE state.

If a reassembly timer is supported, the following procedures apply:

8) When after the processing of a SAR-PDU the SAR receiver reaches the REASS state, the reassembly timer shall be (re-)started.

9) If the timer is still running when the SAR receiver transitions from the REASS state to the IDLE state, the timer shall be stopped.

10) If the timer expires (the SAR receiver is in the REASS state) the SAR receiver shall issue a SAR-P-Abort-signal to the receiving CPCS and shall proceed to the IDLE state.

Other reassembly timer procedures are for further study.

NOTE – The timer value may be dependent on the AAL connection but is not specified in this Recommendation.

4.4.2 Procedures of the CPCS for the message mode service

The structure and coding of the CPCS-PDU are defined in 4.3.2.1.

4.4.2.1 State variables of the CPCS at the sender side

The CPCS sender maintains the following state variable:

- *snd_BEtag*

  This variable is used to set the Btag field in the CPCS-PDU header and the Etag field in the CPCS-PDU trailer.

4.4.2.2 Procedures of the CPCS at the sender side for the message mode service

The state machine of the CPCS sender is shown in Figure 19.

Table 11 defines the states for the CPCS sender.

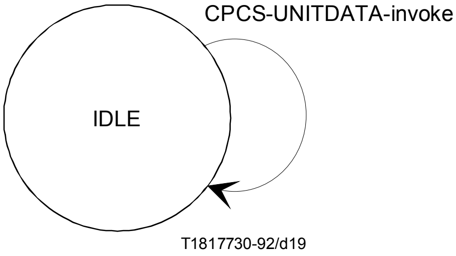


FIGURE 19/I.363

State transition diagram for the CPCS sender

TABLE 11/I.363

State definitions for the CPCS sender

State	Definition
IDLE	Waiting to transmit a new CPCS-SDU

1) When the CPCS connection is established, the CPCS sender shall set its state variable snd_BEtag to any value from 0 to 255.

2) Upon receiving an CPCS-UNITDATA-invoke from the CPCS user, the CPCS sender shall construct the CPCS-PDU header, place the received CPCS-SDU into the CPCS-PDU payload, construct the PAD field and construct the CPCS-PDU trailer. The CPCS-PDU is then forwarded in its entirety (i.e. the M parameter is set to "0") to the SAR sublayer via the SAR-UNITDATA-invoke primitive for segmentation and transmission.

3) After forwarding the CPCS-PDU to the SAR sublayer, the CPCS sender shall modify its state variable snd_BEtag. This modification must assure that the CPCS receiver can unambiguously associate the CPCS-PDU header and trailer of every CPCS-PDU even in the presence of loss of information (cell losses across CPCS-PDU boundaries). At the minimum, the snd_BEtag shall be set to any value different from the current one (modulo 256).

NOTE – A suitable mechanism is to increment the state variable snd_BEtag by one (modulo 256) after each CPCS-PDU.

4.4.2.3 State variables of the CPCS at the receiver side

The CPCS receiver maintains the following state variables:

1) *rcv_BEtag*

   This variable is used to assure that a received CPCS-PDU trailer belongs to the CPCS-PDU currently being reassembled. This is achieved by copying the Btag field value to this state variable when processing the CPCS-PDU header; when processing the associated CPCS-PDU trailer, the value in the Etag field is compared to the value in the state variable.

2) *rcv_BAsize*

   This variable is used to assure that attempts to assemble CPCS-PDUs that are longer than the requested BAsize will fail.

4.4.2.4 Procedures of the CPCS at the receiver side

The state machine of the CPCS receiver is shown in Figure 20.

Table 12 defines the states for the CPCS receiver.

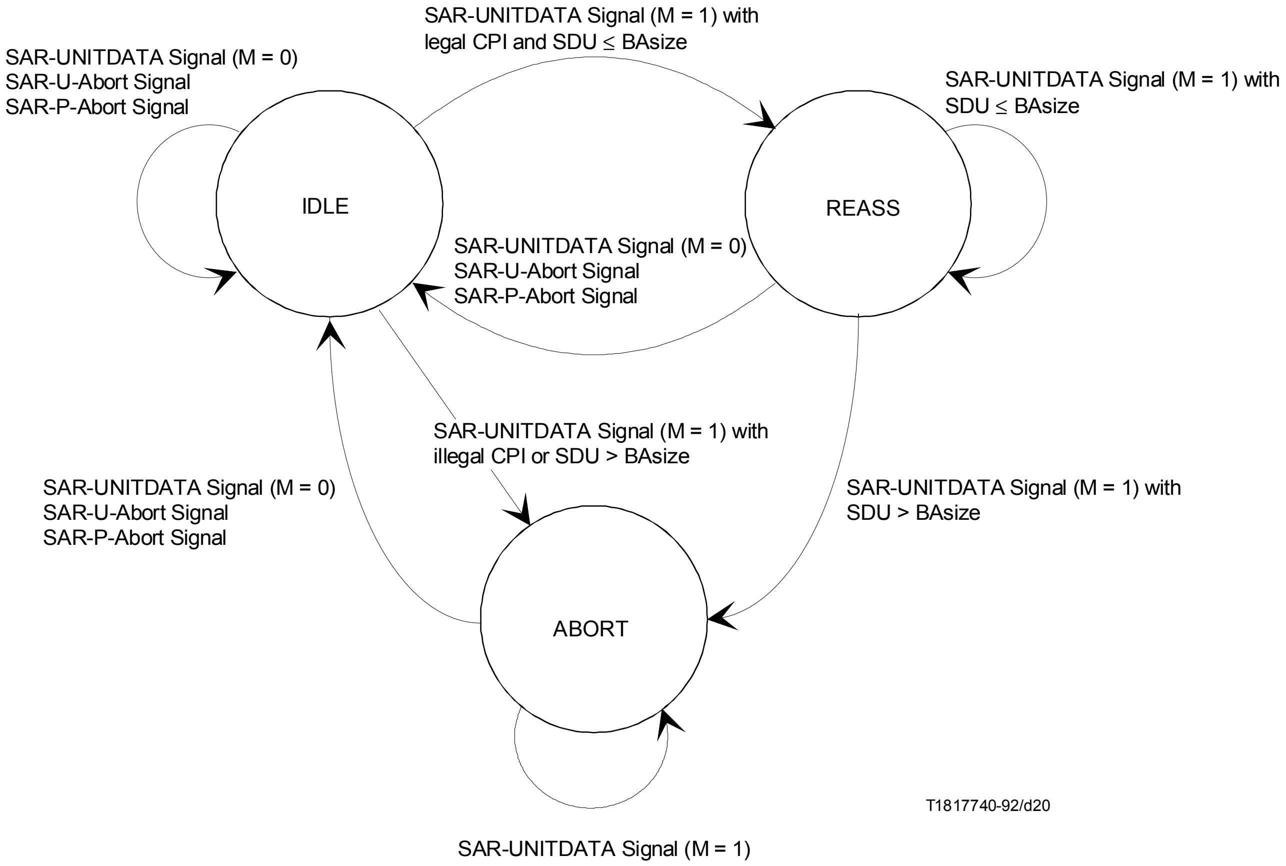


FIGURE 20/I.363

State transition diagram for the CPCS receiver

TABLE 12/I.363

State definitions for the CPCS receiver

State	Definition
IDLE	Waiting to begin to reassemble a new CPCS-PDU
REASS	Reassembling a CPCS-PDU
ABORT	Aborting an illegal CPCS-PDU

The following procedures are specified for a CPCS receiver that does not deliver errored data to the receiving CPCS user. Procedures for the optional delivery of errored information are for further study.

1) When the CPCS receiver is in the IDLE state and it receives a SAR-UNITDATA-signal primitive from the SAR sublayer, the first four octets of the information represent the CPCS-PDU header.

If the CPI field in the CPCS-PDU header is illegal, the CPCS receiver shall proceed to the ABORT state if the M parameter is set to “1” or to the IDLE state if the M parameter is set to “0”. Otherwise, the CPCS receiver shall copy the value of the Btag field into the rcv_BEtag state variable. It also shall set the state variable rcv_BAsize to the value of the BAsize field. The allocation of a reassembly buffer with at least the size indicated in the state variable rcv_BAsize is implementation dependent.

NOTES

1 This procedure description may copy up to three octets of the PAD field into the reassembly buffer before processing the CPCS-PDU trailer.

2 When the CPCS receiver is in the REASS state and it receives a SAR-UNITDATA-signal primitive from the SAR sublayer, no CPCS-PDU header information is present.

2) When the CPCS receiver is in the IDLE state or the REASS state and it receives a SAR-UNITDATA-signal primitive from the SAR sublayer with the M parameter set to “0”, the last 4 octets of the information represents the CPCS-PDU trailer. If the alignment field in the CPCS-PDU trailer is not equal to zero, the CPCS receiver shall free the reassembly buffer and proceed to or remain in the IDLE state.

   The CPCS receiver shall verify that the value of the Etag field is equal to the value in the rcv_BEtag state variable. If they are not equal, the CPCS receiver shall free the reassembly buffer and proceed to or remain in the IDLE state.

   If the value of the length field in the CPCS-PDU trailer is greater than the already reassembled information in the reassembly buffer plus the information in the interface data of the primitive currently processed (without the CPCS-PDU trailer and the CPCS-PDU header), the CPCS receiver shall free the reassembly buffer and proceed to or remain in the IDLE state.

   If the value of the length field in the CPCS-PDU trailer is less than the already reassembled information in the reassembly buffer plus the information in the interface data of the primitive currently processed (without the CPCS-PDU trailer and the CPCS-PDU header) minus 3, the CPCS receiver shall free the reassembly buffer and proceed to or remain in the IDLE state.

   If the already reassembled information in the reassembly buffer plus the information in the interface data of the primitive currently processed (without the CPCS-PDU trailer and the CPCS-PDU header) is greater than the state variable rcv_BAsize plus the maximum pad field length, the CPCS receiver shall free the reassembly buffer and proceed to or remain in the IDLE state.

   The CPCS receiver shall copy the information in the interface data of the primitive currently processed (without the CPCS-PDU trailer and possibly the CPCS-PDU header) to the reassembly buffer. The CPCS receiver shall then send the reassembled CPCS-SDU to the CPCS user in the interface data of a CPCS-UNITDATA-signal primitive; the amount of information in the interface data is equal to the value of the length field of the CPCS-PDU trailer. It shall also free the reassembly buffer, and proceed to or remain in the IDLE state.

3) When the CPCS receiver is in the IDLE state or the REASS state and it receives a SAR-UNITDATA-signal primitive from the SAR sublayer with the M parameter set to “1”, no CPCS-PDU trailer is present.

   If the already reassembled information in the reassembly buffer plus the information in the interface data of the primitive currently processed (without possibly the CPCS-PDU header) is greater than the state variable rcv_BAsize plus the maximum pad field length, the CPCS receiver shall free the reassembly buffer and proceed to the ABORT state. Otherwise, the CPCS receiver shall copy the information in the interface data of the primitive currently processed (without possibly the CPCS-PDU header) to the reassembly buffer and proceed to or remain in the REASS state.

4) If the CPCS receiver receives a SAR-U-Abort-signal or a SAR-P-Abort-signal primitive from the SAR sublayer while in the IDLE state, the primitive shall be ignored; when in the REASS state, the CPCS receiver shall free the reassembly buffer and proceed to the IDLE state.

5) If the CPCS receiver is in the ABORT state and it receives a SAR-UNITDATA-signal primitive with the M parameter set to "1", the primitive shall be ignored and the CPCS receiver shall remain in the ABORT state.

   However, if in the ABORT state the CPCS receiver receives a SAR-U-Abort or SAR-P-Abort-signal primitive or a SAR-UNITDATA-signal primitive with the M parameter set to "0", the CPCS receiver shall proceed to the IDLE state.

4.4.3 Procedures of the CPCS for the streaming mode service

These procedures are for further study.

4.4.4 Procedures of the SSCS

These procedures are for further study.

5 AAL type 4

As the enhanced specification for AAL types 3 and 4 are now equivalent, the texts have been merged into 4 and referred to as AAL type 3/4.

See 4.

6 AAL type 5

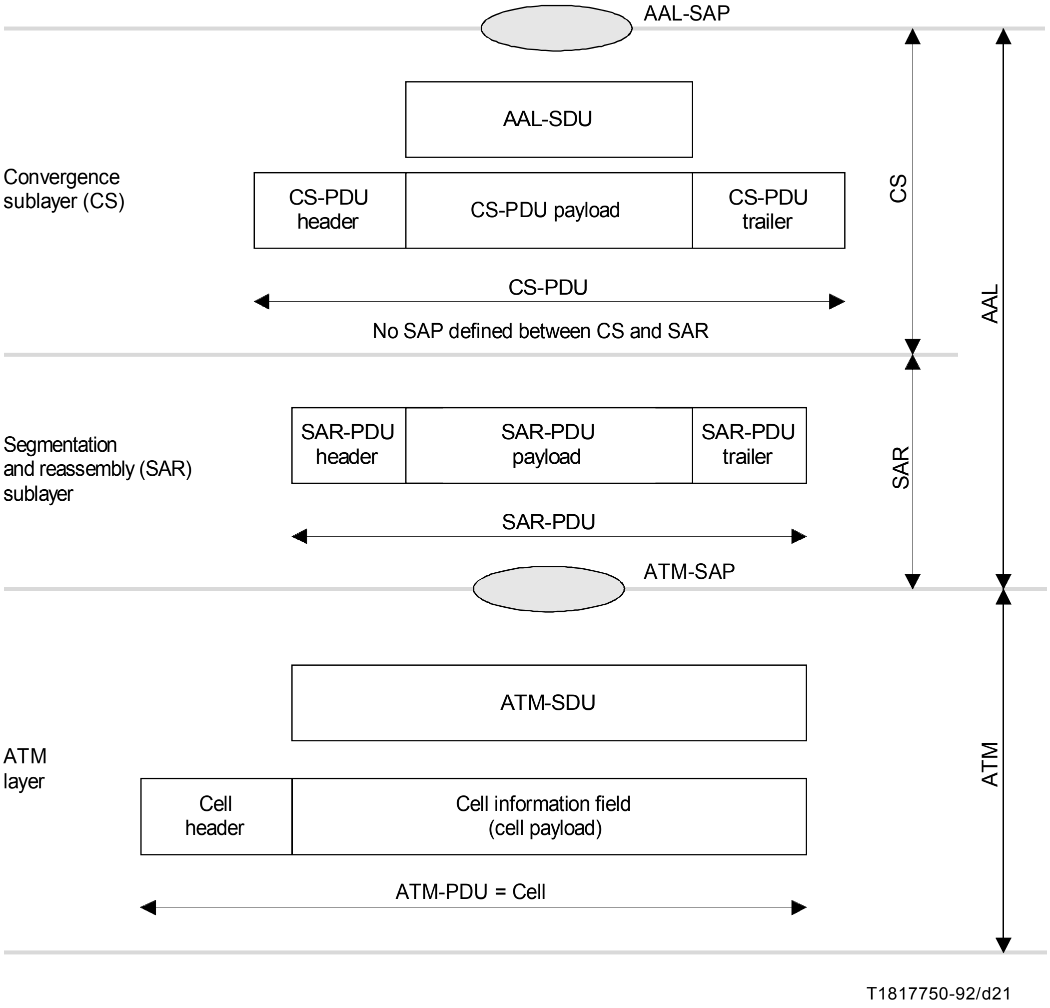
Under study.

Annex A

Details of the data unit naming convention

(This annex forms an integral part of this Recommendation)

Details of the data unit naming convention are given in Figures A.1 to A.3.



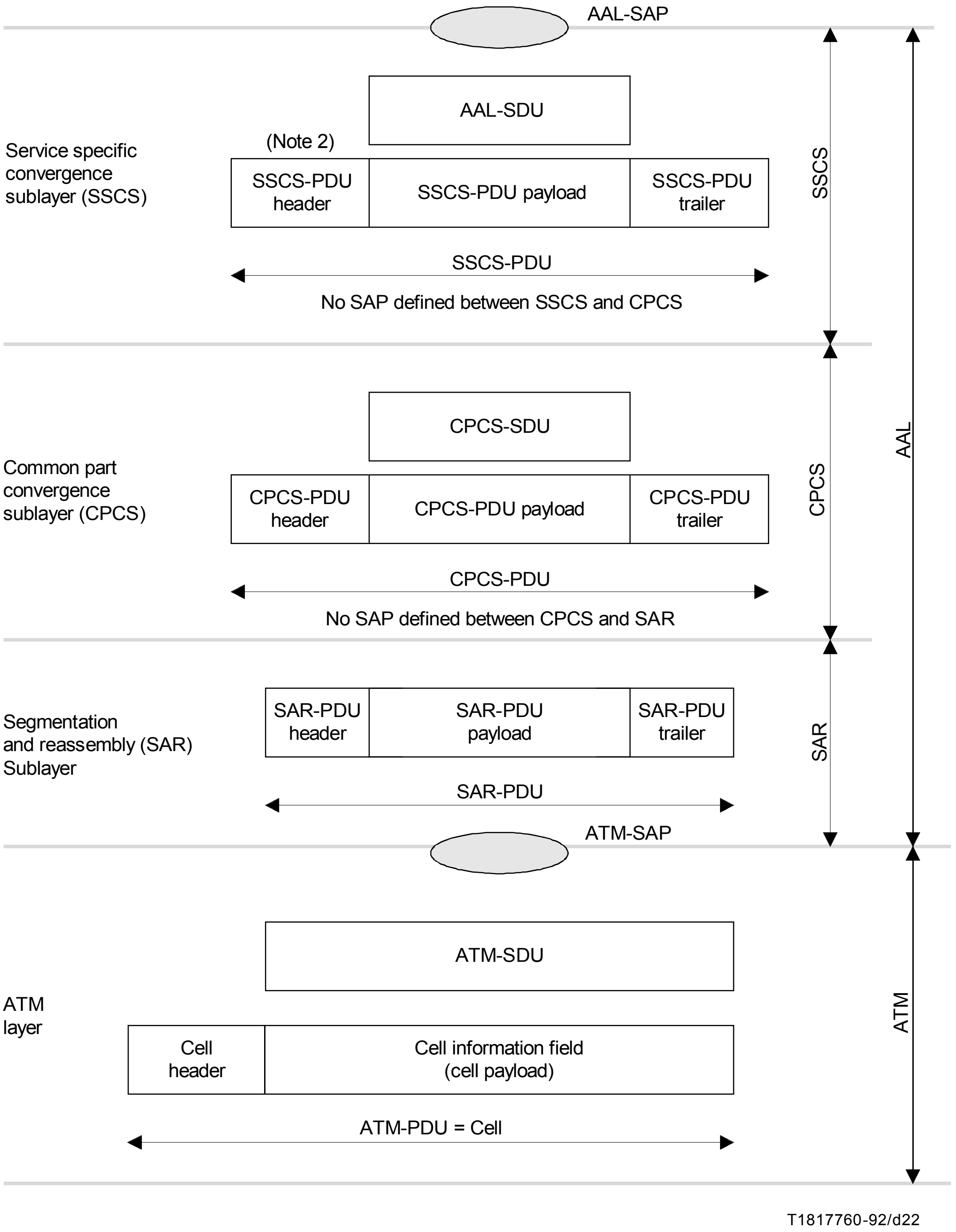
NOTES

1 ATM adaptation layer-protocol control information (AAL-PCI) consists of the SAR-PDU Header, CS-PDU header, CS-PDU trailer, and SAR-PDU trailer.

2 The figure is to indicate the naming of the AAL data units only. It is not implied that all fields are present in all cases. See Annex D for a list of abbreviations.

FIGURE A.1/I.363

General data unit naming conventions



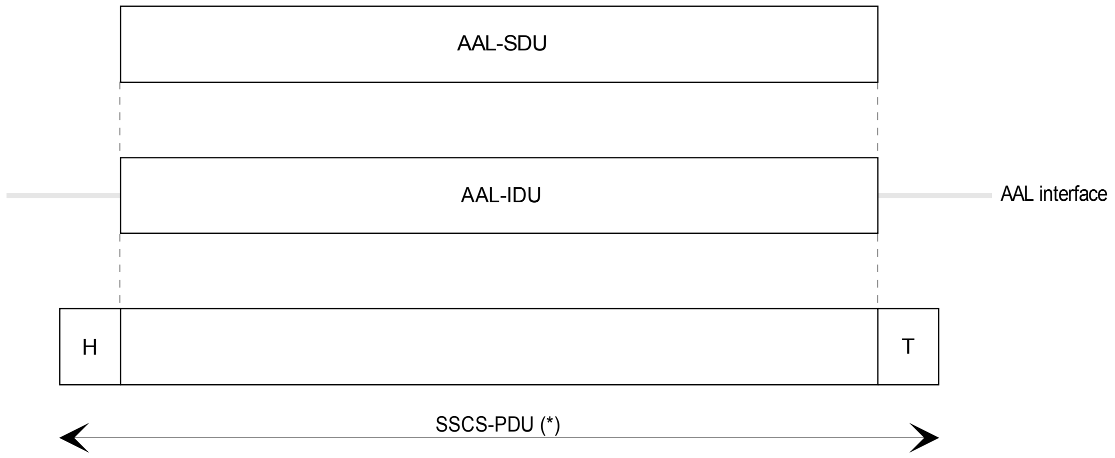
NOTES

1 The figure is to indicate the naming of the AAL data units only. It is not implied that all fields are present in all cases. See Annex D for a list of abbreviations.

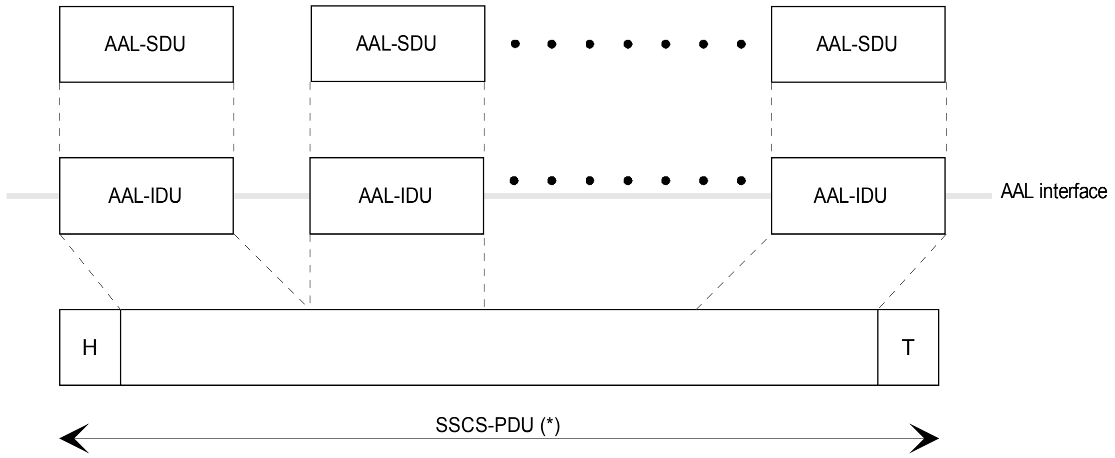
2 The exact structure of the SSCS-PDU is for further study.

FIGURE A.2/I.363

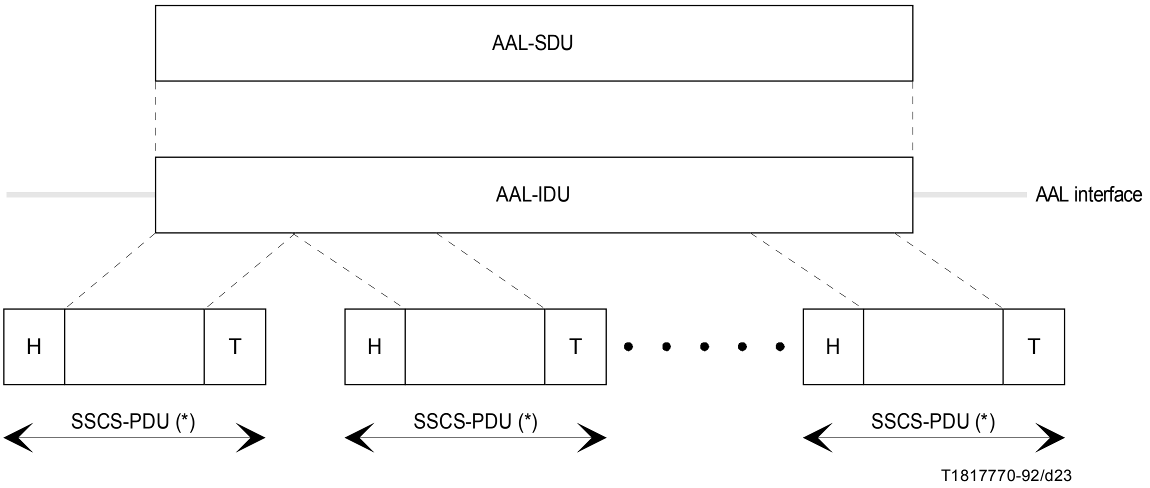
Data unit naming conventions of the ALL type 3/4



a) Message mode service



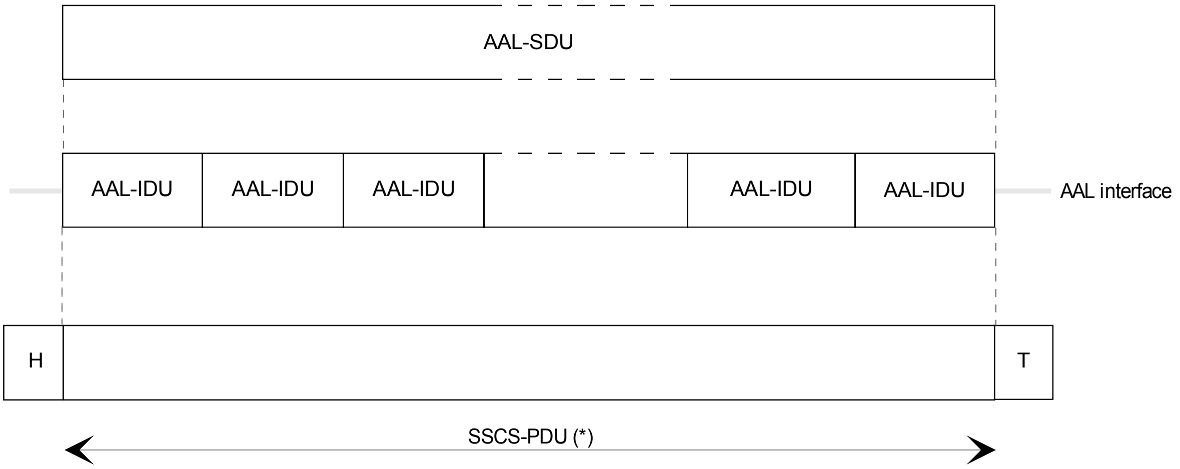
b) Message mode service plus blocking/deblocking internal function



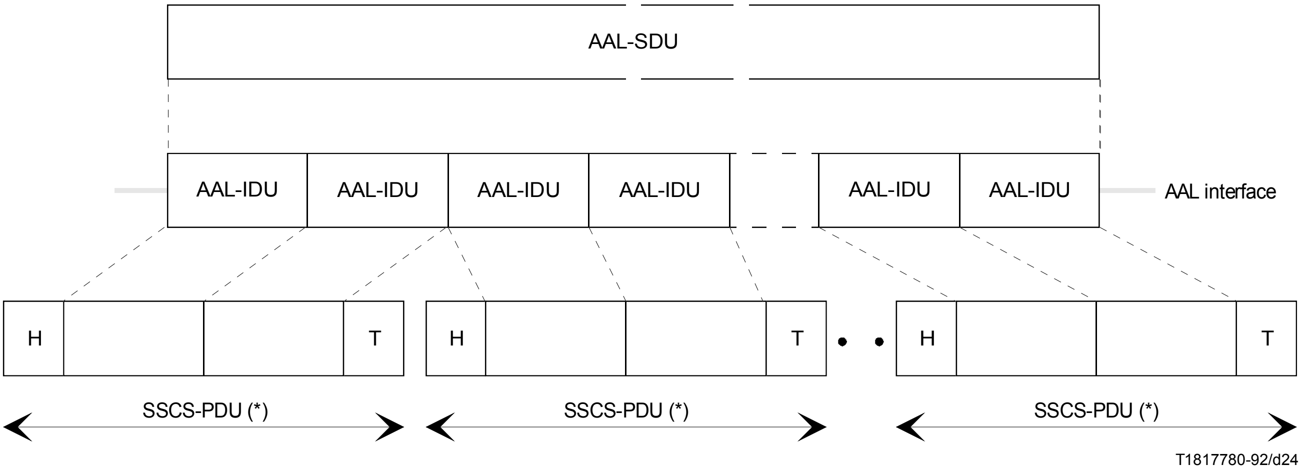
c) Message mode service plus segmentation/reassembly internal function

FIGURE A.3/I.363 (sheet 1 of 2)

Message and streaming mode of service at the AAL type 3/4 interface combined with blocking/deblocking or segmentation/reassembly internal function



d) Streaming mode service



e) Streaming mode service plus segmentation/reassembly internal function

(*) The structure of the SSCS-PDU is for further study.

FIGURE A.3/I.363 (sheet 2 of 2)

Message and streaming mode of service at the AAL type 3/4 interface combined with blocking/deblocking or segmentation/reassembly internal function

Annex B

General framework of the AAL type 3/4

(This annex forms an integral part of this Recommendation)

This annex provides a description of the general framework of the AAL type 3/4 including SAR and CPCS PDU formats.

B.1 Message segmentation and reassembly

Figure B.1 provides a generic interpretation of the segmenting of a message into beginning of message (BOM), continuation of message (COM) and end of message (EOM). Short messages are represented as a single segment message (SSM).

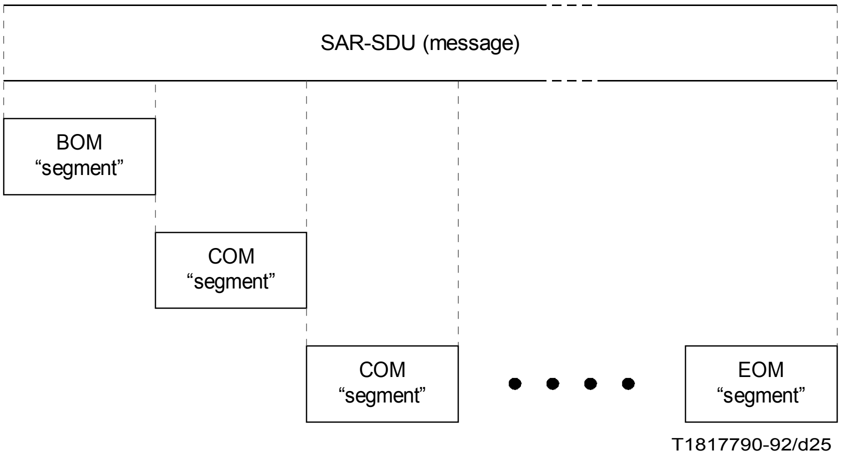


FIGURE B.1/I.363

Message segmentation and reassembly

B.2 PDU headers, trailers and terminology

Figure B.2 builds on the generic view of message segmentation of Figure B.1 to incorporate the relevant PDU headers and trailers and appropriate terminology on the basis of BOM, COM and EOM which is of particular relevance to the combined SAR and CPCS-PDU formats of Figure B.3

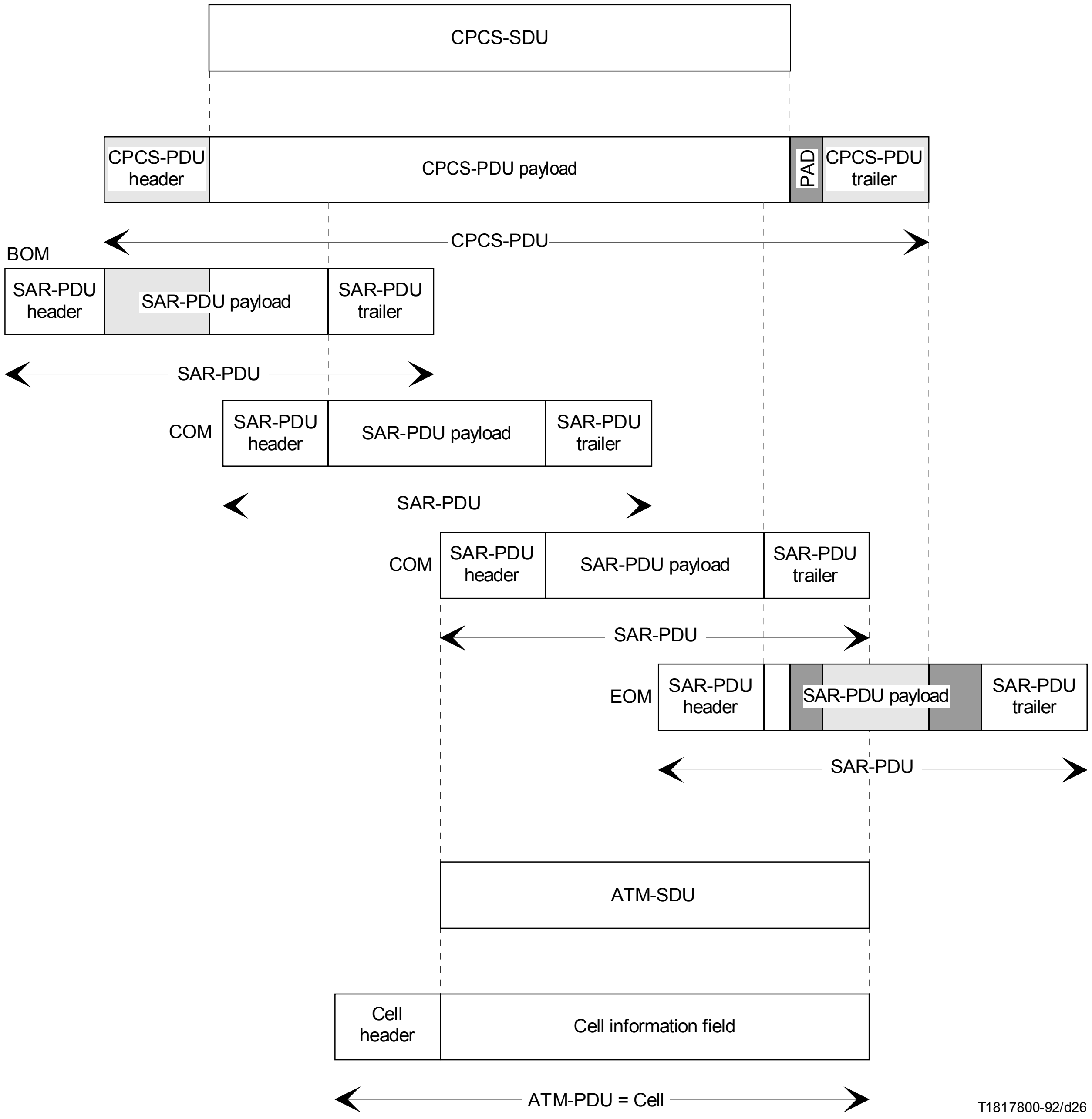


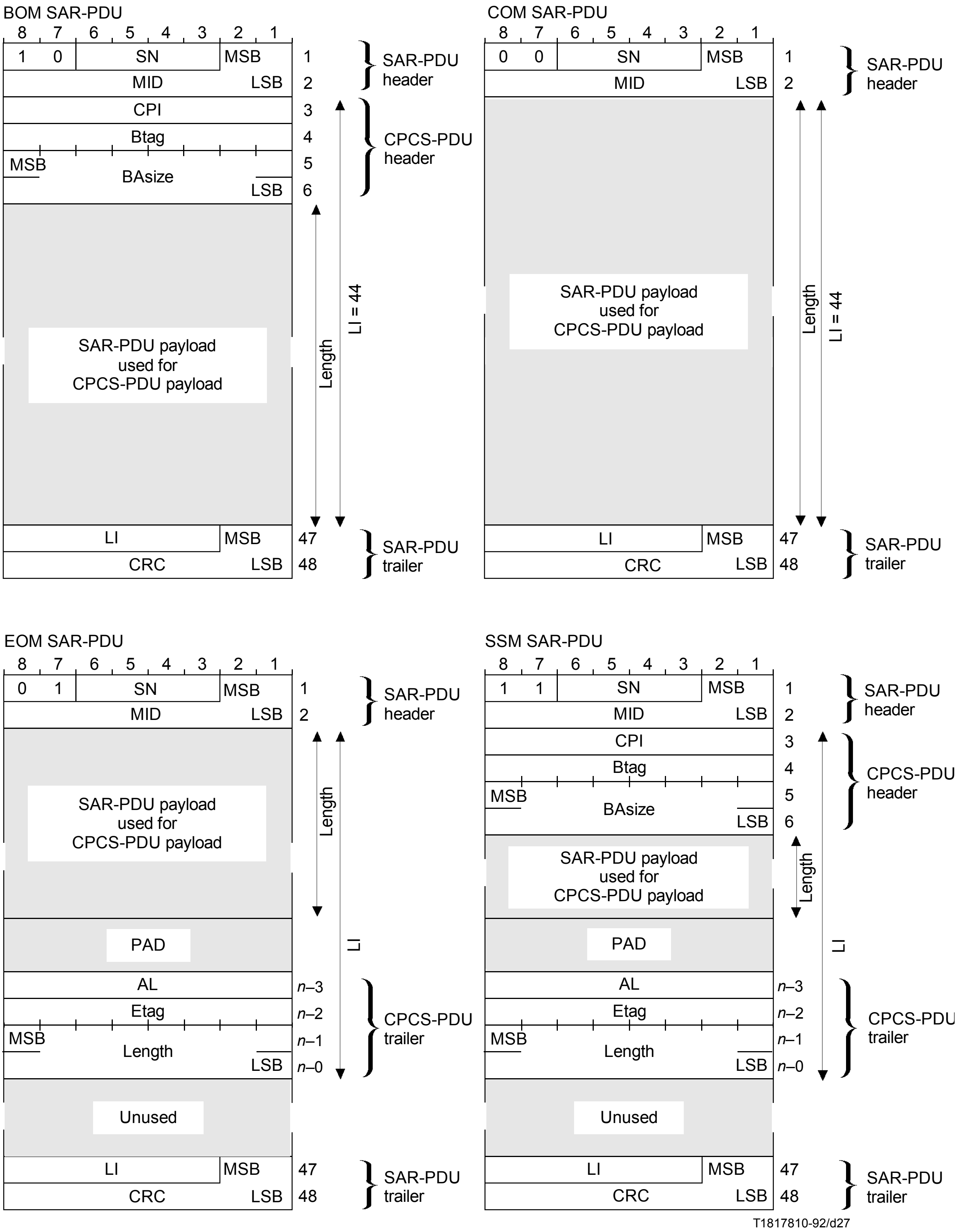
FIGURE B.2/I.363

PDU headers, trailers, and terminology

B.3 SAR and CPCS format

Figure B.3 illustrates the combined SAR and CPCS PDU format on a segment by segment basis.

The definition of the encoding and functions associated with the fields is described in 4.3.1.2 and 4.3.2.1.2.



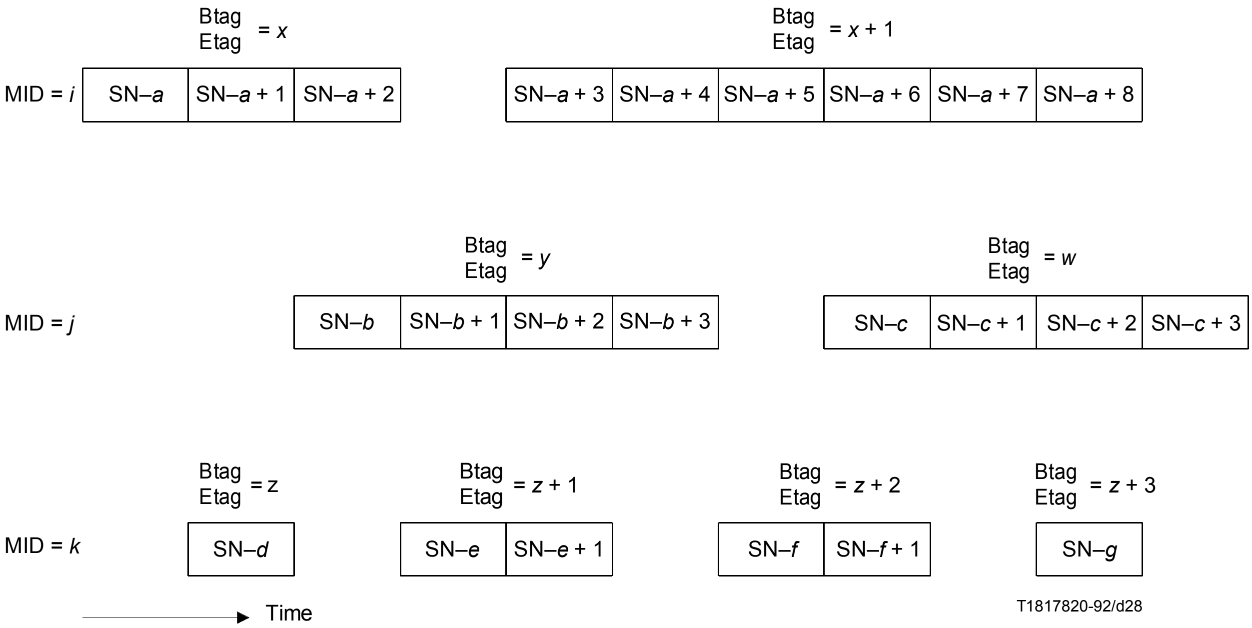
ST	Segment type	Btag	Begin tag
SN	Sequence number	Etag	End tag
MID	Multiplexing identification	BAsize	Buffer allocation size
LI	Length indicator	AL	Alignment
CRC	Cyclic redundancy check	Length	CPCS-PDU payload length
CPI	Common part indicator		

FIGURE B.3/I.363

Combined SAR and CPCS-PDU format

B.4 Relation of the MID field to the SN field and Btag/Etag fields

As an example, the following Figure B.4 illustrates the possible relation of the MID field values to the SN field and Btag/Etag field values for the AAL type 3/4.



NOTE – Modulo 16 and modulo 256 apply to determine the SN field and the Btag/Etag fields.

FIGURE B.4/I.363

The relation of MID field values to the SN field and Btag/Etag field values for AAL type 3/4

B.5 Examples of the segmentation and reassembly process

The Figure B.5 shows schematically a successful segmentation and reassembly of a CPCS user PDU in message mode. In Figure B.6, a SAR-PDU is assumed lost due to a transmission error, hence, the reassembly cannot be completed.

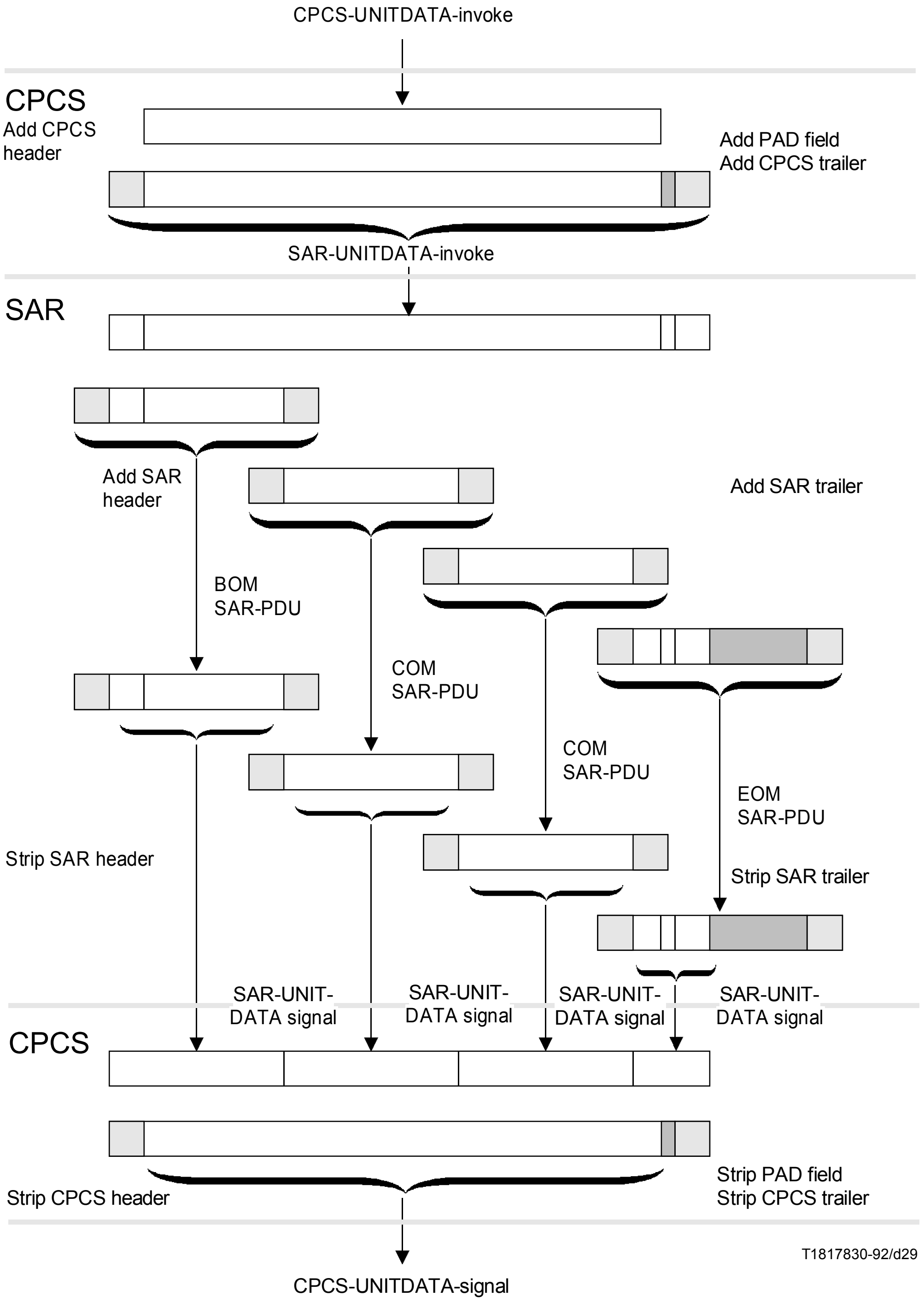
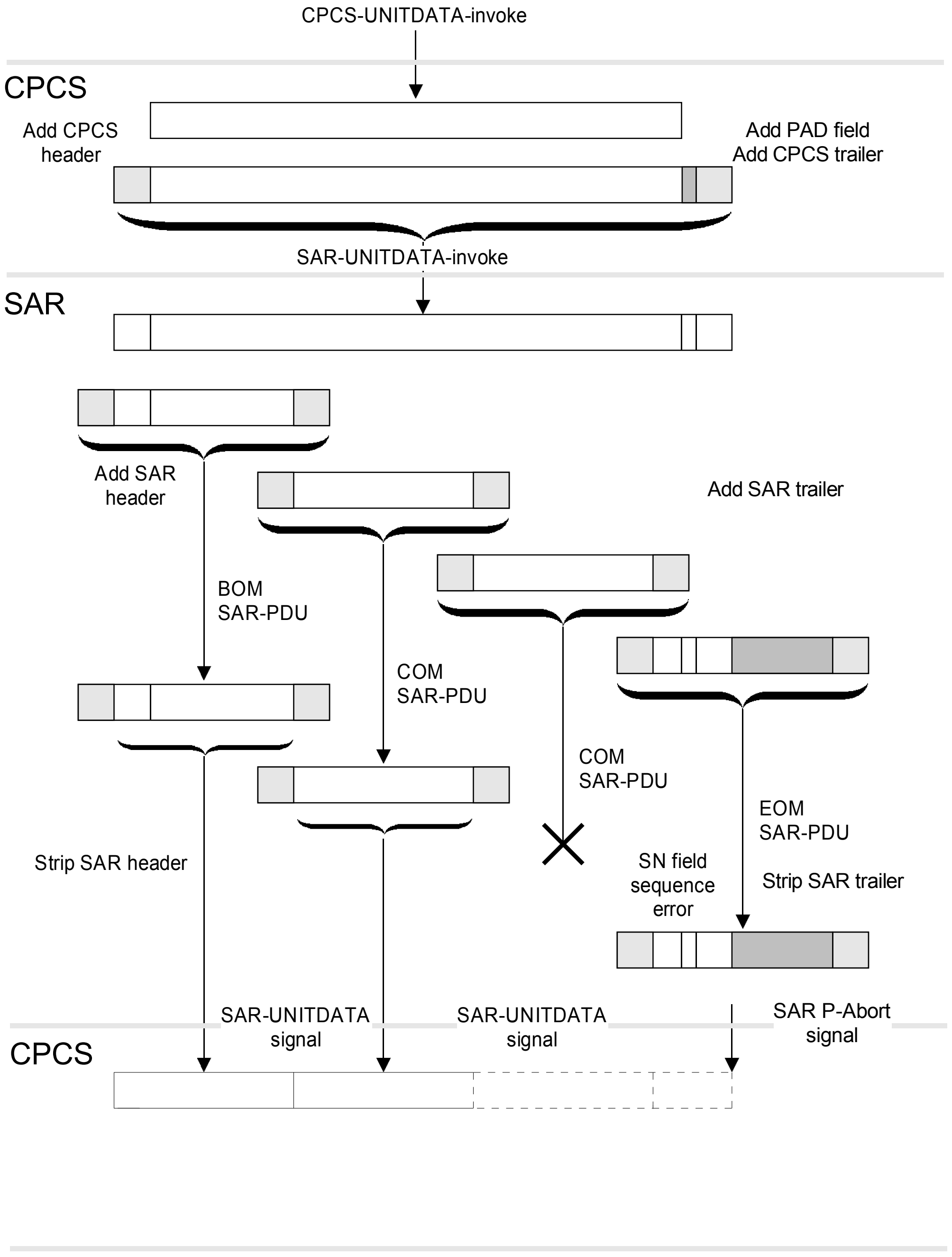


FIGURE B.5/I.363

Successful segmentation and reassembly of a CPCS user PDU



T1817840-92/d30

FIGURE B.6/I.363

Segmentation and unsuccessful reassembly of a CPCS user PDU

Annex C

Functional model for the AAL type 3/4

(This annex forms an integral part of this Recommendation)

For the AAL type 3/4, the functionality of the SSCS may provide only for the mapping of the equivalent primitives of the AAL to the CPCS and vice versa. The SSCS may also implement functions such as assured data transfer, etc. Such functions, however, are not shown in the following figures.

The functional model of the AAL type 3/4 at the sender side is shown in Figure C.1. The model consists of several blocks that cooperate to provide the AAL type 3/4 services. Each SAR and CPCS block that are paired represent one segmentation state machine.

The interleaver allocates the available bit rate of the ATM connection to the SAR-PDUs generated by the segmentation state machines according to some internal policy.

The functional model of the AAL type 3/4 at the receiver side is shown in Figure C.2. The model consists of several blocks that cooperate to provide the AAL type 3/4 services. Each SAR and CPCS block that are paired represesnet one reassembly state machine. The dispatcher (R_DSP) routes the primitives from the ATM layer to the appropriate reassembly state machine based on the value of the MID field within the SAR-PDU.

NOTE – Layer management interactions require further study.

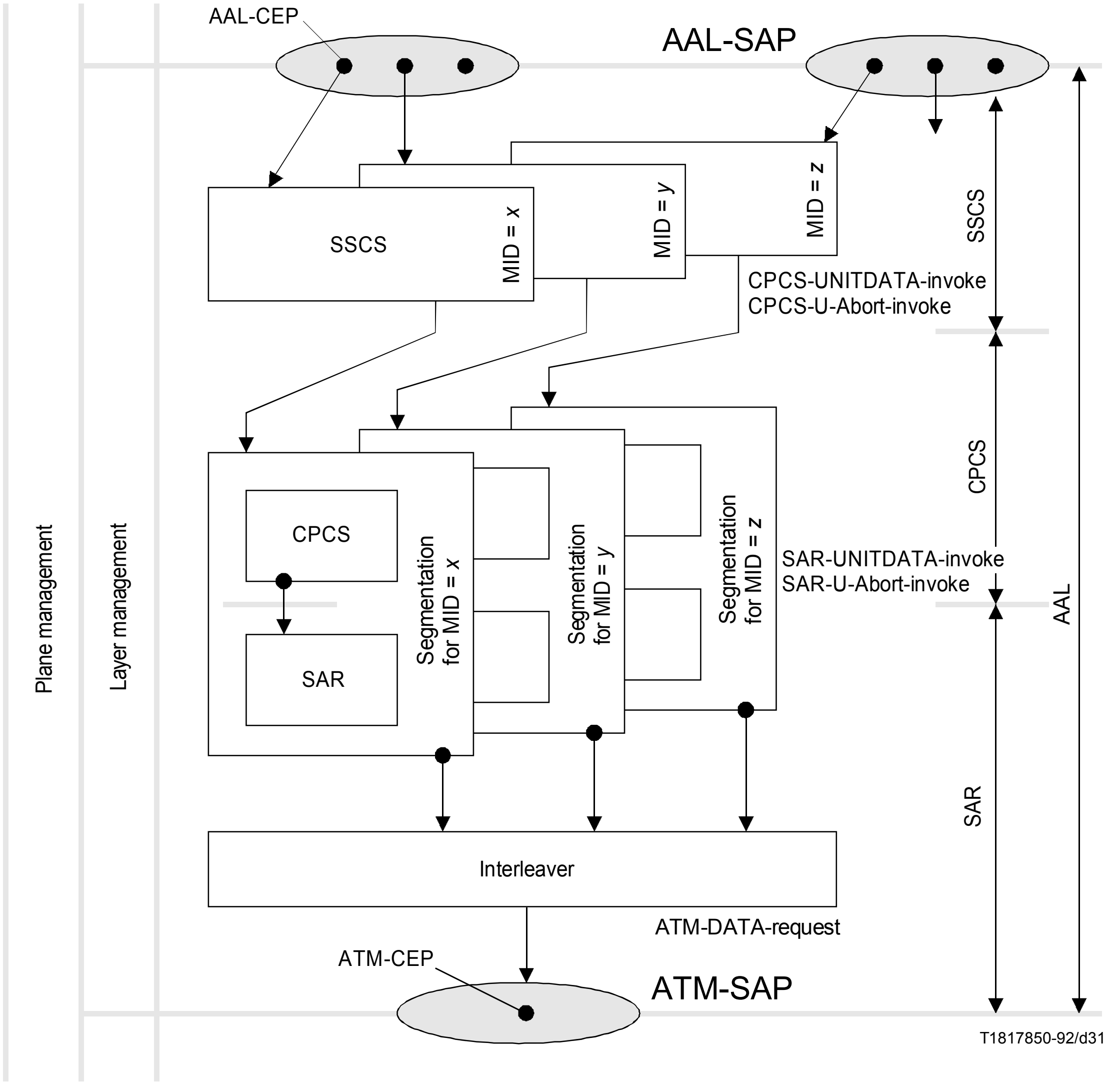


FIGURE C.1/I.363

Functional model for the AAL Type 3/4 (Sender side)

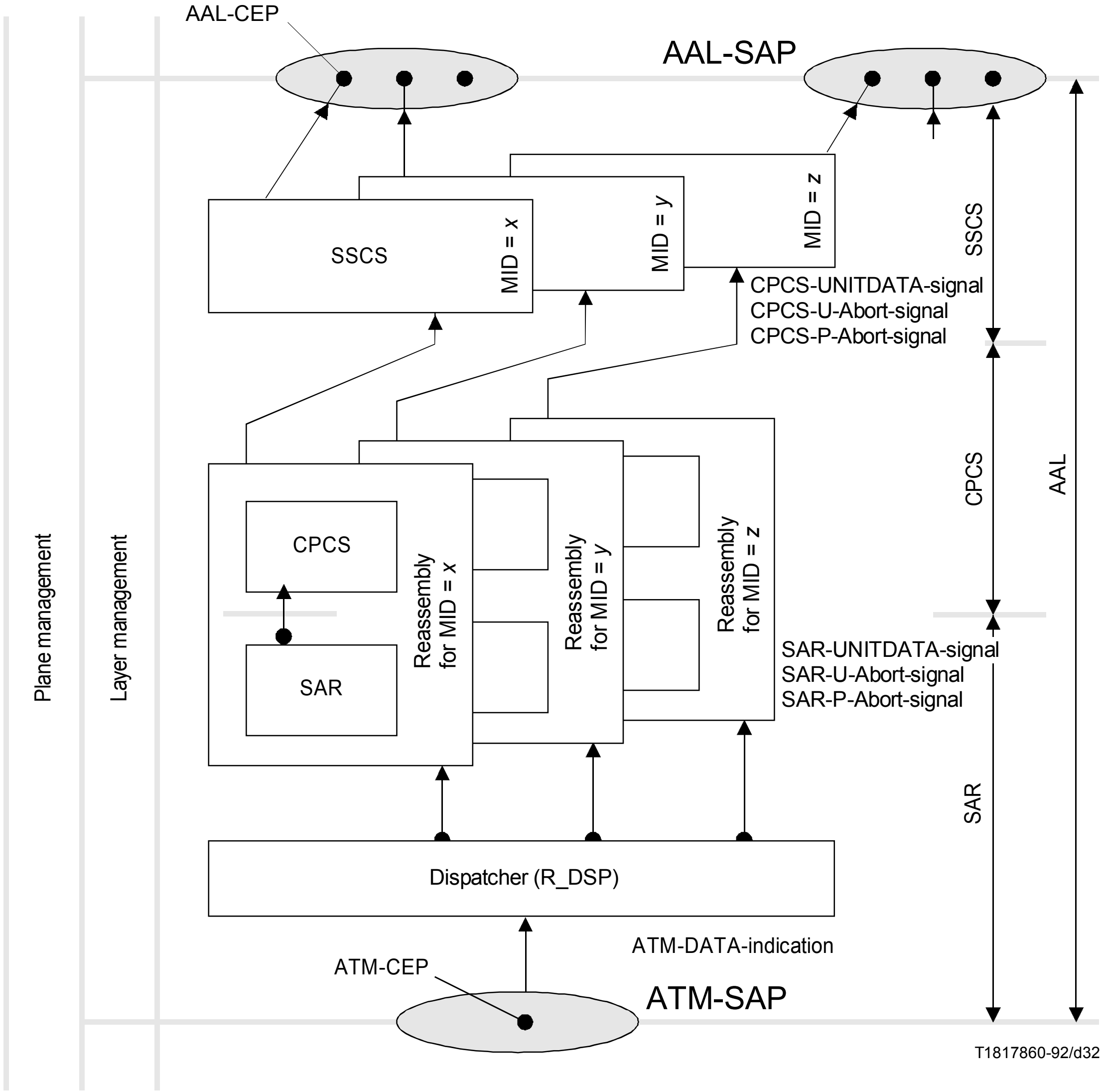


FIGURE C.2/I.363

Functional model for the AAL Type 3/4 (Receiver side)

Annex D

Alphabetical list of abbreviations used in this Recommendation

(This annex forms an integral part of this Recommendation)

AAL	ATM adaptation layer
AAL-IDU	AAL interface data unit
AAL-SDU	AAL service data unit
ATM-SDU	ATM service data unit
BOM	Beginning of message
COM	Continuation of message
CPCS	Common part convergence sublayer
CPCS-PDU	CPCS protocol data unit
CPCS-SDU	CPCS service data unit
CRC	Cyclic redundancy check
CS	Convergence sublayer
CS-PDU	CS protocol data unit
CSI	Convergence sublayer indication
EOM	End of message
FEC	Forward error correction
LSB	Least significant bit
M	More
MID	Multiplexing identification
MSB	Most significant bit
OAM	Operation Administration maintenance
RTS	Residual time stamp
SAP	Service access point
SAR	Segmentation and reassembly sublayer
SAR-PDU	SAR protocol data unit
SAR-SDU	SAR service data unit
SDT	Structure data transfer
SNP	Sequence number protection
SRTS	Synchronous residual time stamp
SSCS	Service specific convergence sublayer
SCS-PDU	SSCS protocol data unit
SSM	Single segment message

Appendix I

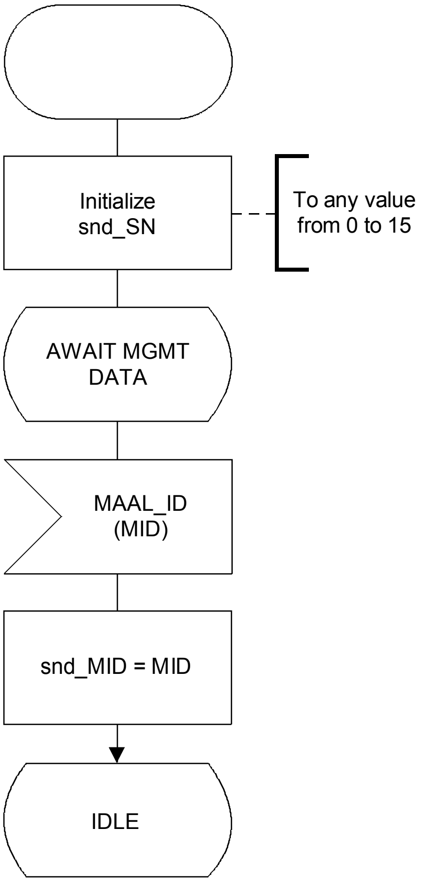
SDL diagrams for the SAR and the CPCS of the AAL type 3/4

(This appendix does not form an integral part of this Recommendation)

I.1 SDL for the SAR sublayer

The purpose of this appendix is to provide one example of an SDL representation of the SAR procedures and with it to assist in the understanding of this Recommendation. This representation does not describe all of the possible actions of the SAR sublayer entity as a non-partitioned representation (i.e. the state machine is shown for one MID field value) was chosen in order to minimize its complexity. Therefore, the SDL representation does not constrain implementations from exploiting the full potential inherent in this highly parallel and fast environment. The text description of the procedures in the main part of this Recommendation is definitive.

NOTE – The SDL diagrams in Figures I.1 and I.2 represent the SAR for one MID field value.



T1817870-92/d33

FIGURE I.1/I.363 (sheet 1 of 5)

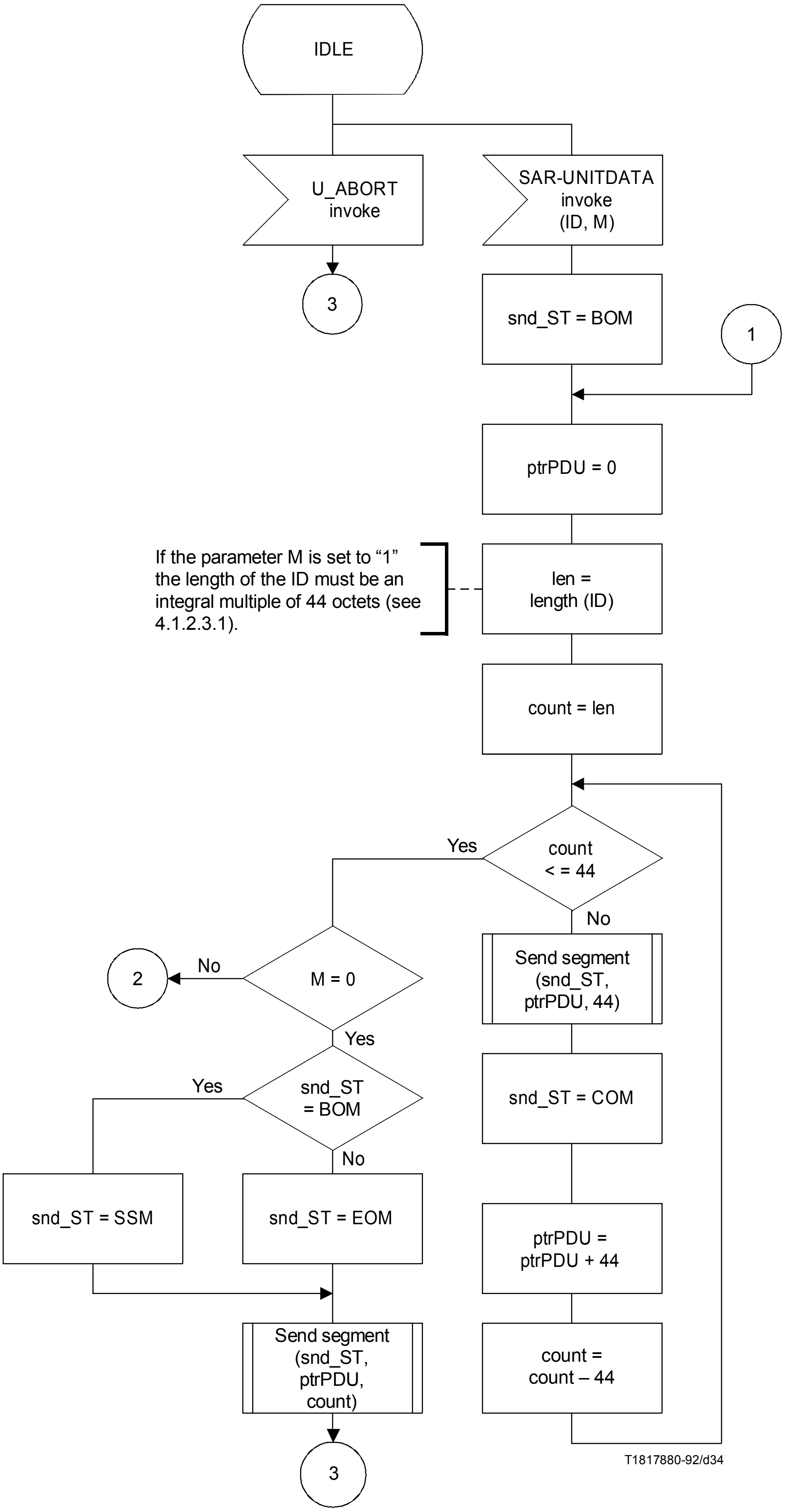


FIGURE I.1/I.363 (sheet 2 of 5)

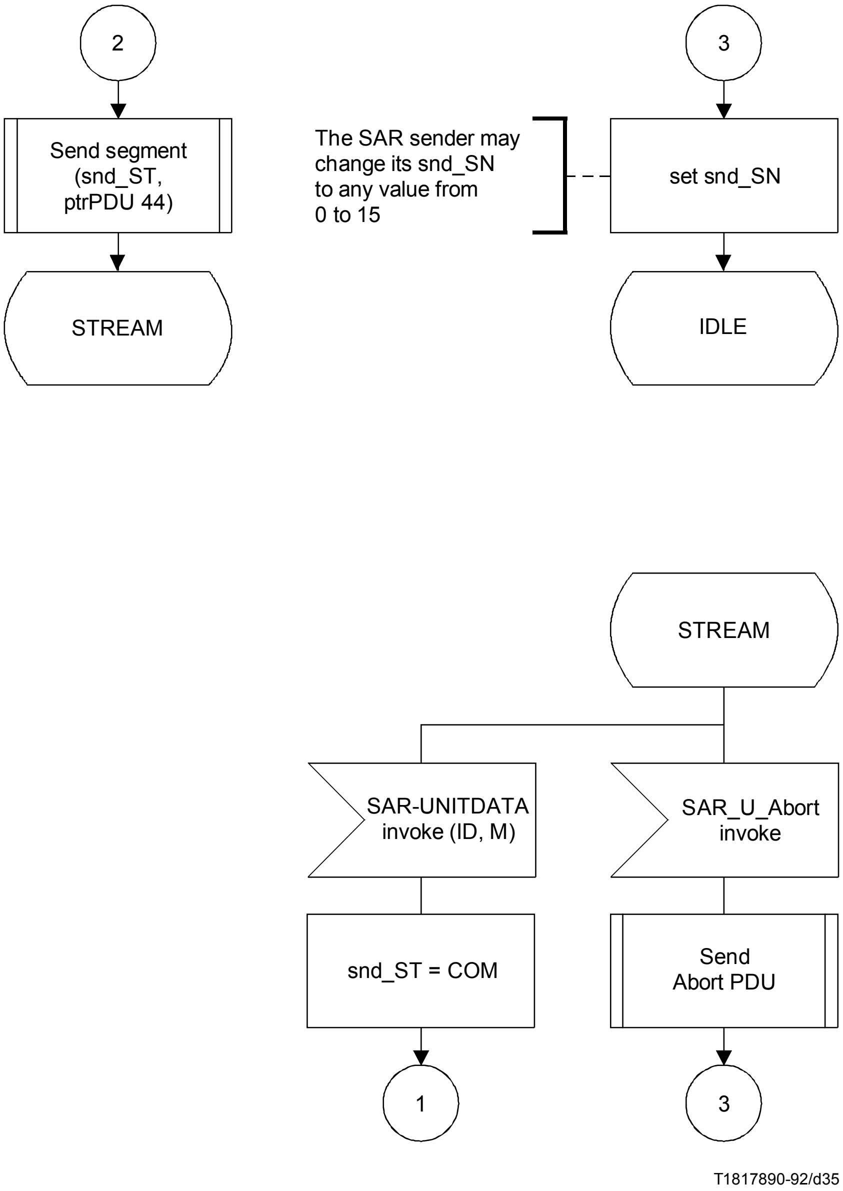


FIGURE I.1/I.363 (sheet 3 of 5)

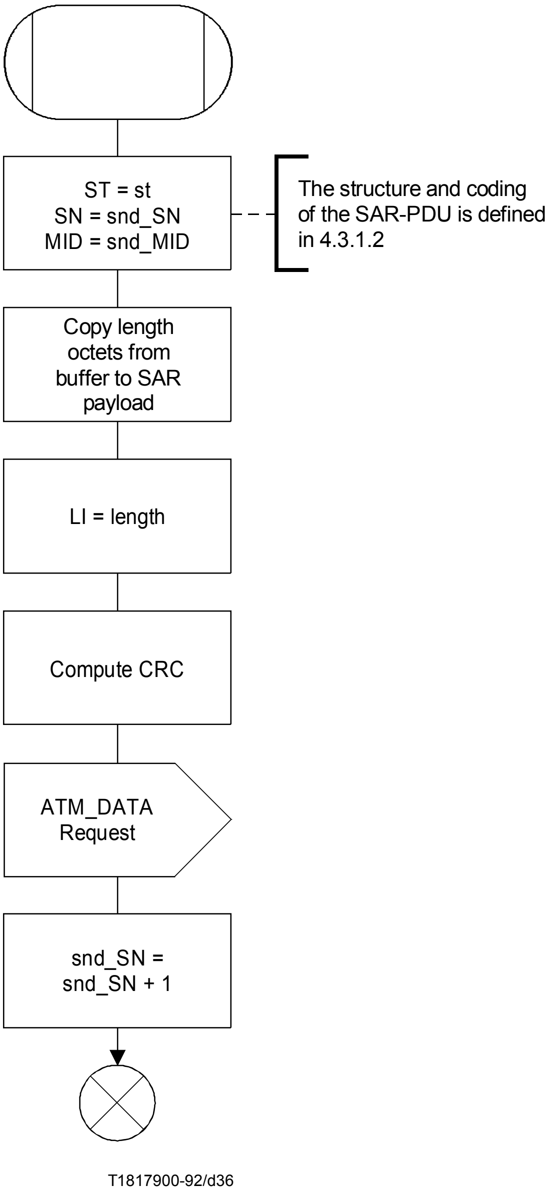
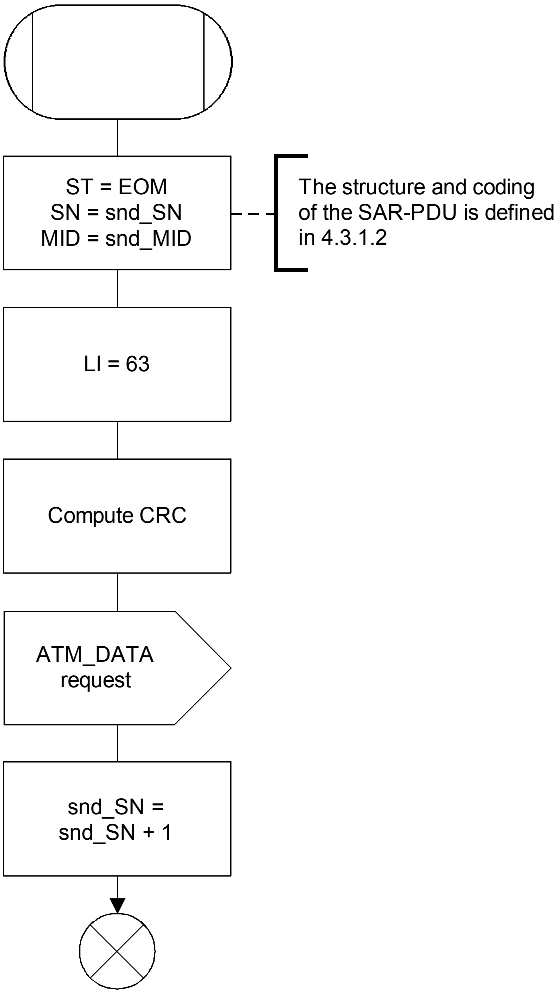


FIGURE I.1/I.363 (sheet 4 of 5)



T1817910-92/d37

FIGURE I.1/I.363 (sheet 5 of 5)

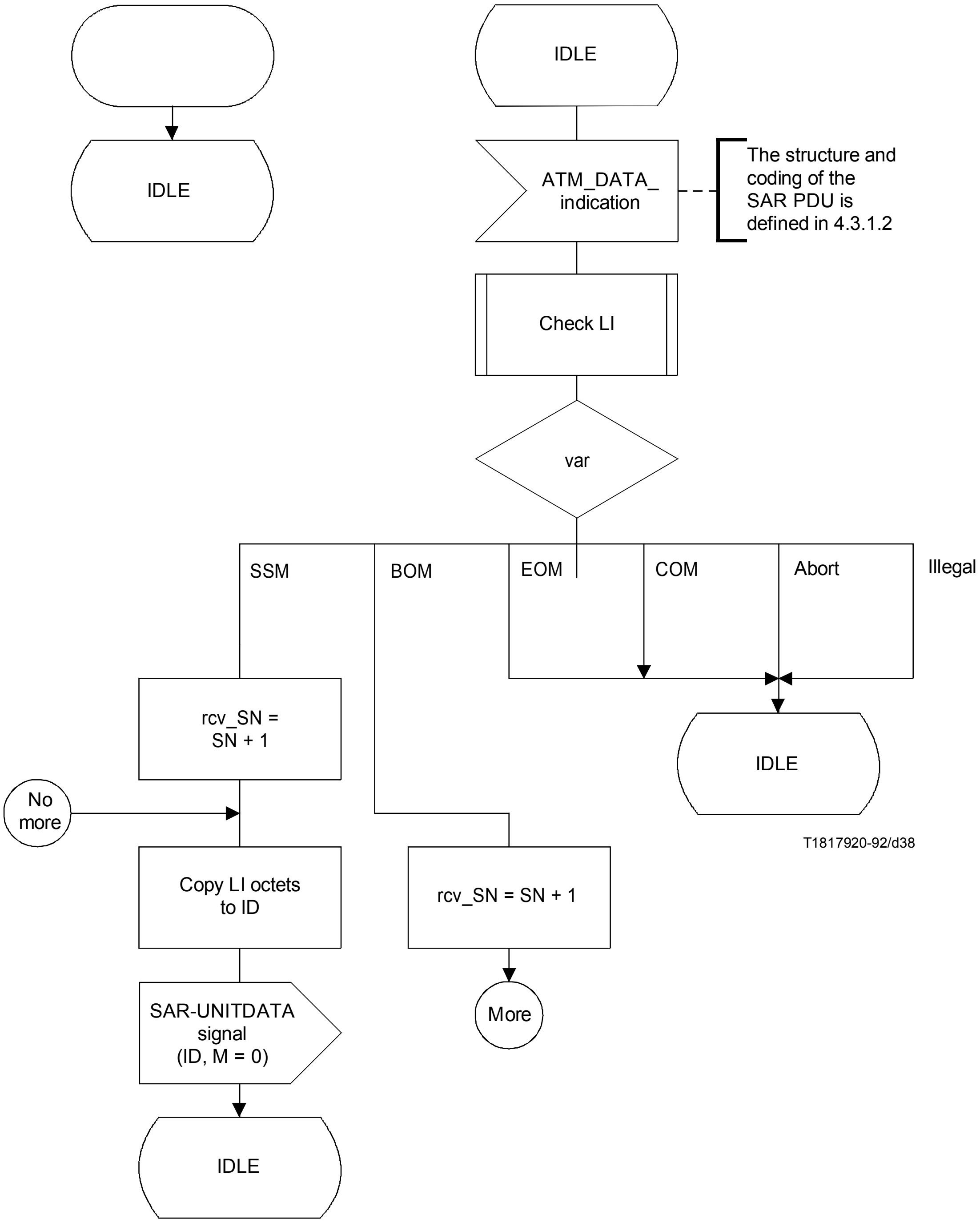


FIGURE I.2/I.363 (sheet 1 of 4)

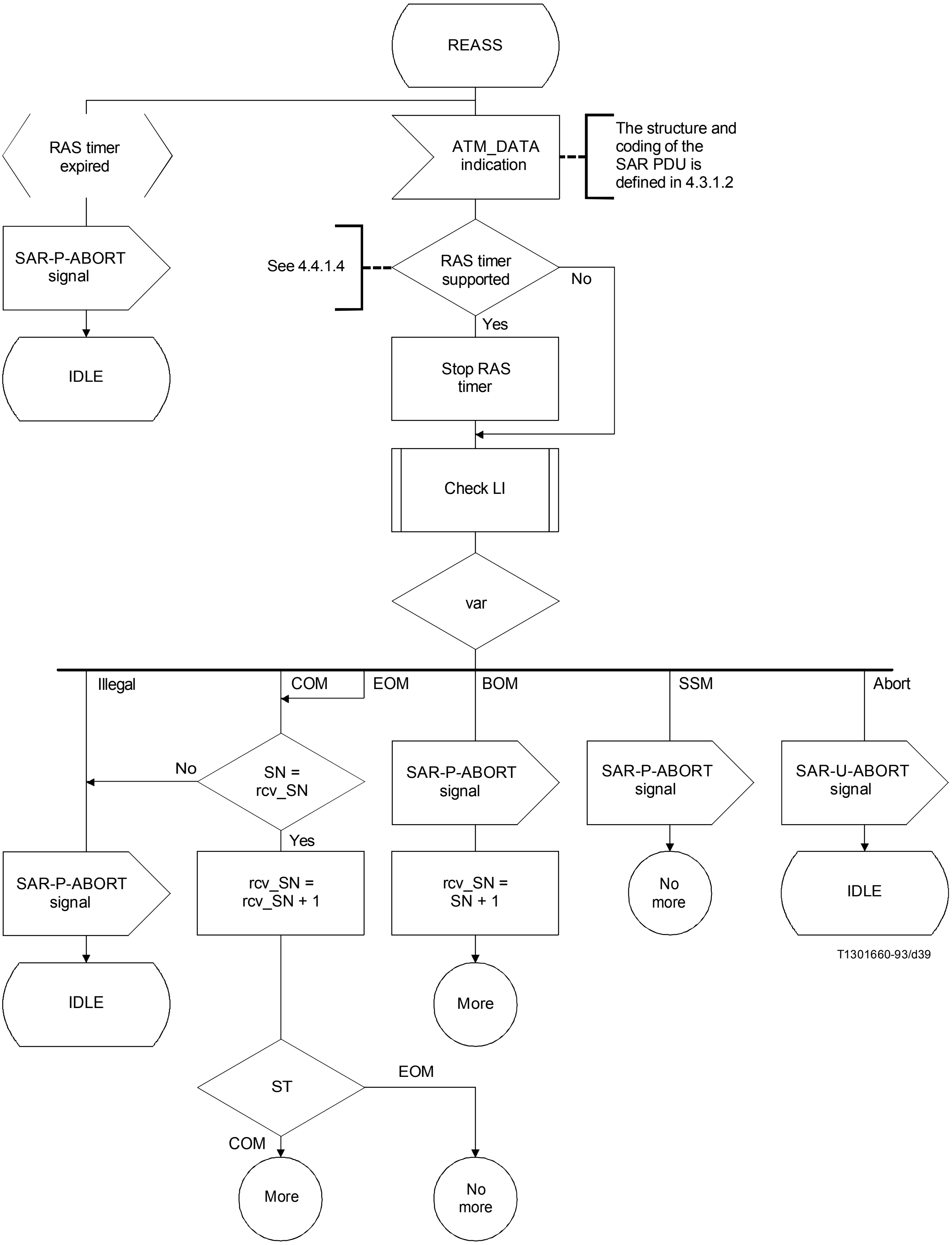


FIGURE I.2/I.363 (sheet 2 of 4)

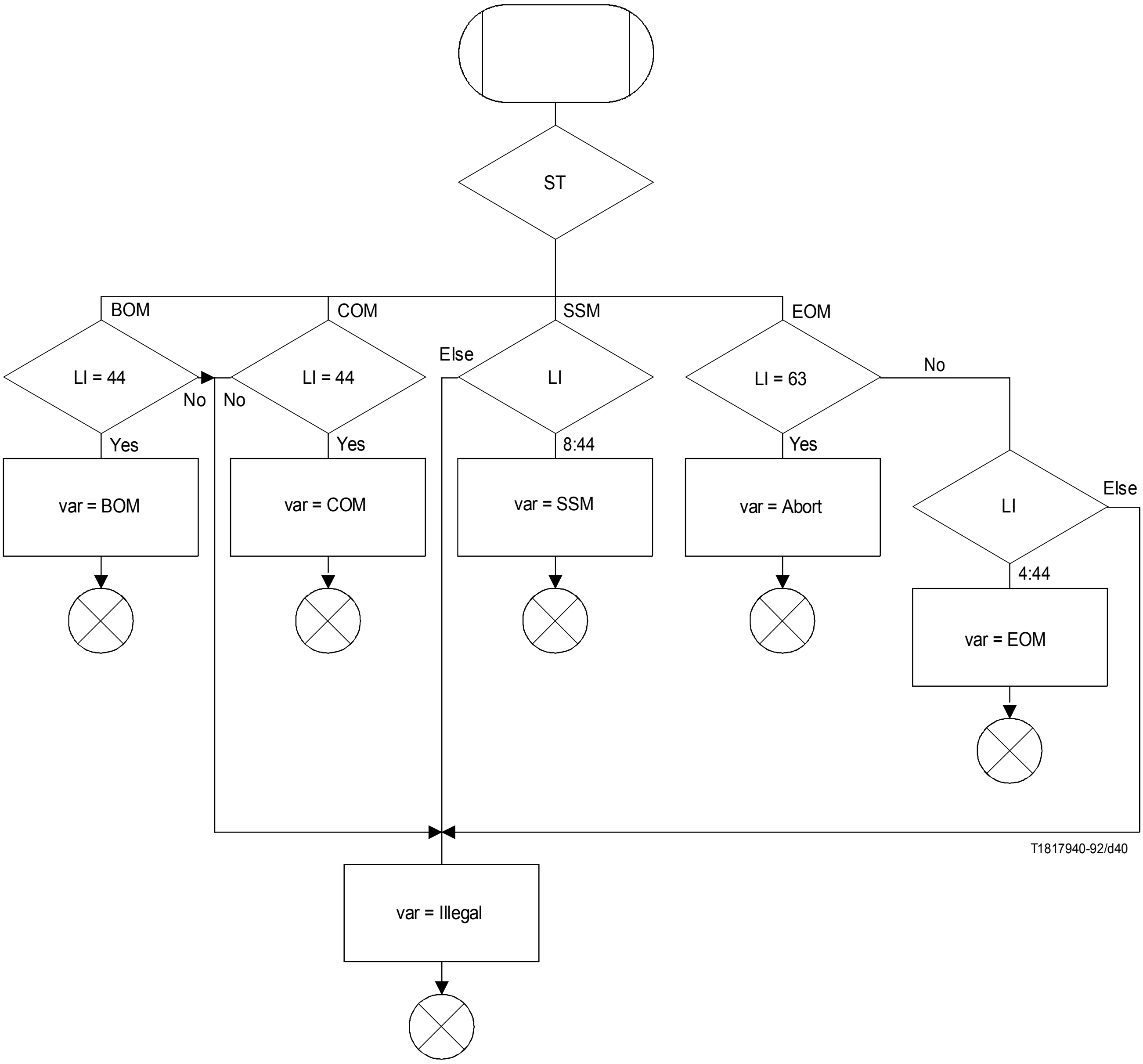


FIGURE I.2/I.363 (sheet 3 of 4)

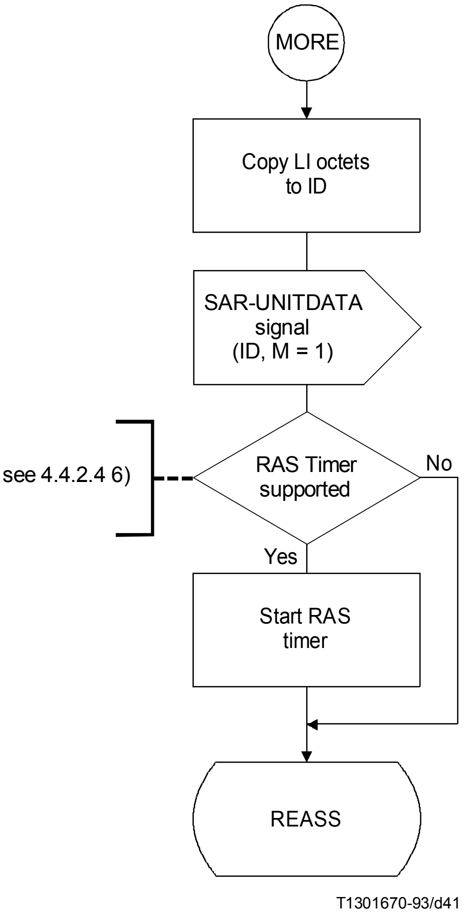


FIGURE I.2/I.363 (sheet 4 of 4)

I.1.1 The SAR sender

The SAR sender makes use of the state variable snd_SN (as defined in 4.4.1.1). In addition, it utilizes four further variables:

a) *ptrPDU*

This is a temporary variable that points into the (partial) CPCS-PDU received via the SAR-UNITDATA-invoke primitive. As successive parts of the CPCS-PDU are filled into SAR-PDU payloads, this pointer keeps pointing at the first octet within the CPCS-PDU that has not yet been sent within a SAR-PDU.

b) *len*

This temporary variable is set to the length of the (partial) CPCS-PDU received via the SAR-UNITDATA-invoke primitive.

c) *count*

This temporary variable keeps track of the number of octets still awaiting segmentation and transmission within a SAR-PDU.

d) *snd_ST*

This temporary variable is used to set the ST field of the SAR-PDU header. It can take the values: "BOM", "COM", "EOM" or "SSM".

e) *snd_MID*

This variable contains the value of the MID field that is put into every SAR-PDU.

The primitive MAAL-ID is used in the SAR sender. Its only parameter communicates a MID field value from layer management to the SAR sender. The details of this primitive and all other interactions with layer management are for further study.

I.1.2 The SAR receiver

The SAR receiver makes use of the state variable rcv_SN (as defined in 4.4.1.3). It utilizes no further variables.

All illegal SAR-PDUs are ignored. An illegal SAR-PDU is a SAR-PDU with either:

- a CRC verification error, or
- an unexpected MID field value.

NOTES

1 The discarding of illegal SAR-PDUs actually takes place prior to assigning the SAR-PDU to a reassembly process governed by a particular MID field value, hence, this is not shown in the SDL diagrams.

2 No interactions with layer management are shown; these interactions require further study.

I.2 SDL for the common part CS (CPCS) procedures

The purpose of this appendix is to provide one example of an SDL representation of the CPCS procedures and with it to assist in the understanding of this Recommendation. This representation does not describe all of the possible actions of the CPCS entity as a non-partitioned representation (i.e. the state machine is shown for one MID field value) was chosen in order to minimize its complexity. In particular, neither delivery of errored data nor streaming mode procedures are included. Therefore, the SDL representation does not constrain implementations from exploiting the full potential inherent in this highly parallel and fast environment. The text description of the procedures in the main part of this Recommendation is definitive.

NOTE – The SDL diagrams of Figures I.4 and I.5 represent the CPCS for one MID field value.

I.2.1 The CPCS sender

The CPCS sender makes use of the state variable snd_BEtag (as defined in 4.4.2.1). In addition, it utilizes one further variable:

- *len*

  This temporary variable is set to the length of the interface data parameter received via the CPCS-UNITDATA-invoke primitive. It is used to set the BAsize field, the Length field, and to calculate the length of the PAD field.

NOTE – No interactions with layer management are shown; these interactions require further study.

I.2.2 The CPCS receiver

The CPCS receiver makes use of the state variable rcv_BEtag and rcv_BAsize (as defined in 4.4.2.3). In addition, it utilizes three further variables:

a) *len*

   This temporary variable is set to the length of the CPCS-PDU information received from the SAR sublayer for reassembly.

b) *reassembly buffer*

   The reassembly buffer is allocated while processing the CPCS-PDU header and freed once the reassembly of a CPCS-PDU is complete (or abandoned due to errors).

c) *ptrRAB*

   This variable points into the reassembly buffer to the octet where the next information received from the SAR sublayer is to be stored.

NOTE – No interactions with layer management are shown; these interactions require further study.

Figure I.3 illustrates the use of the reassembly buffer during the reassembly of a CPCS-SDU.

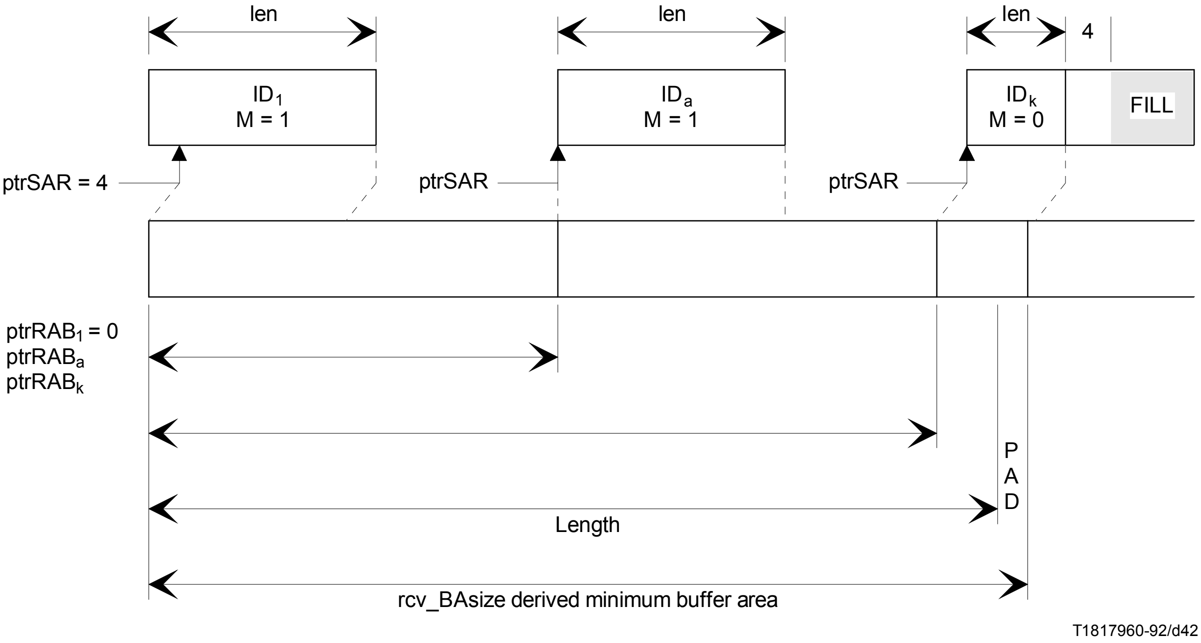


FIGURE I.3/I.363

The mechanism of the reassembly buffer

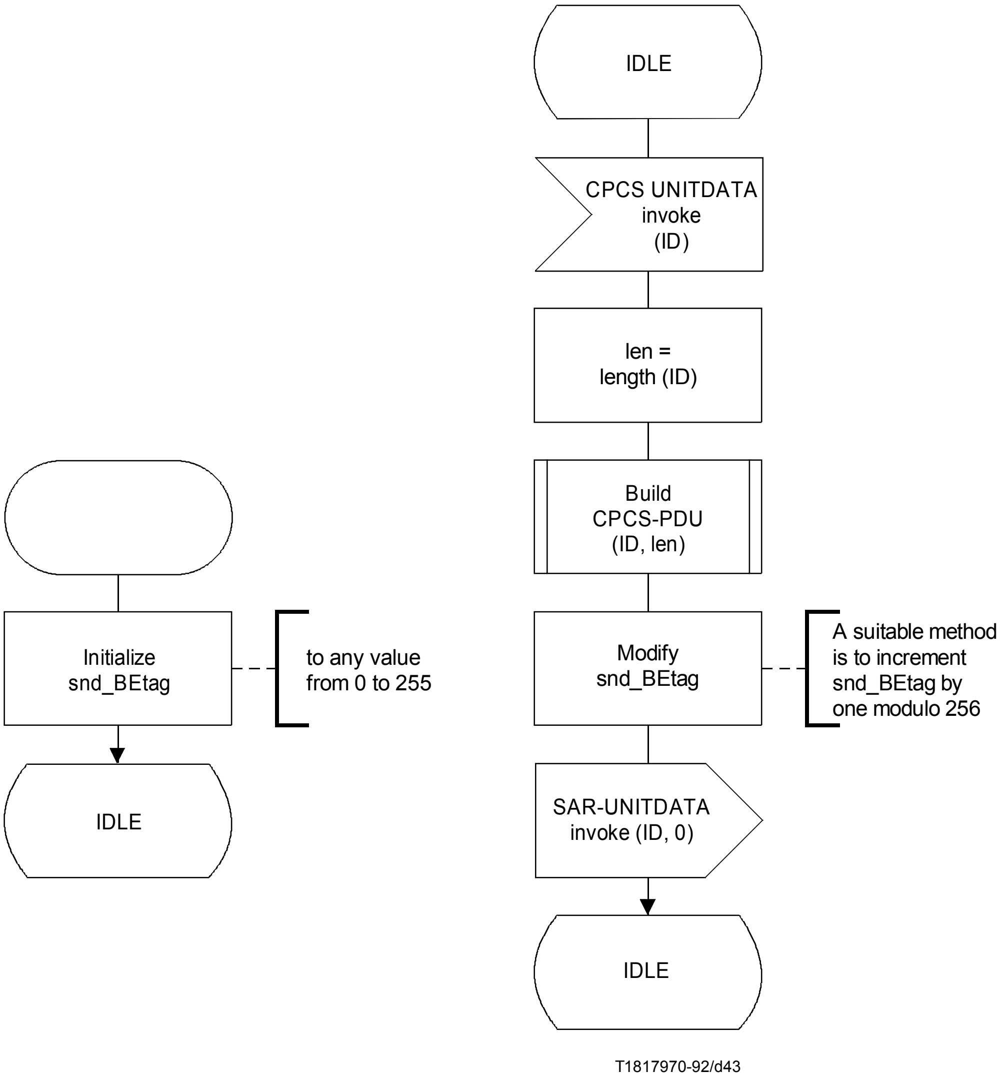


FIGURE I.4/I.363 (sheet 1 of 2)

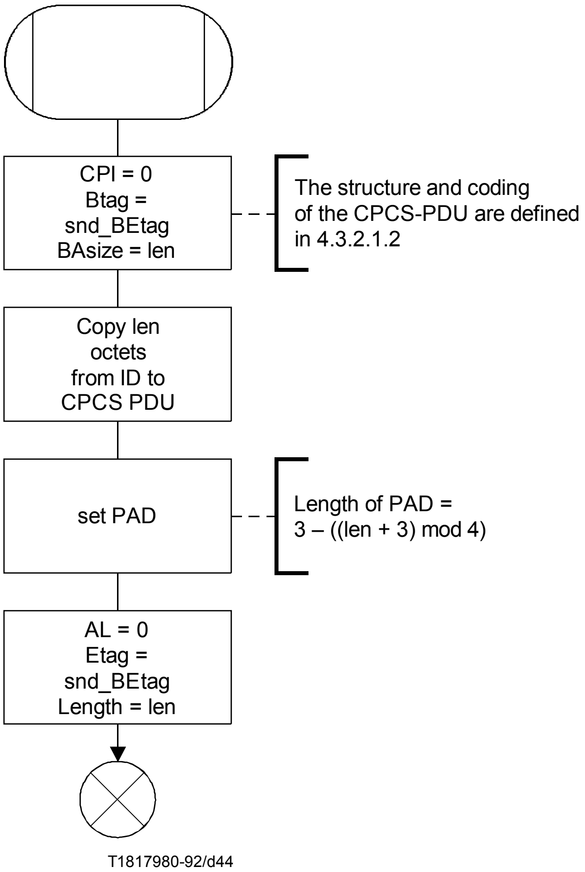


FIGURE I.4/I.363 (sheet 2 of 2)

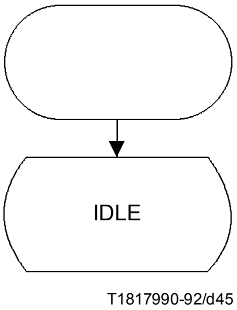


FIGURE I.5/I.363 (sheet 1 of 5)

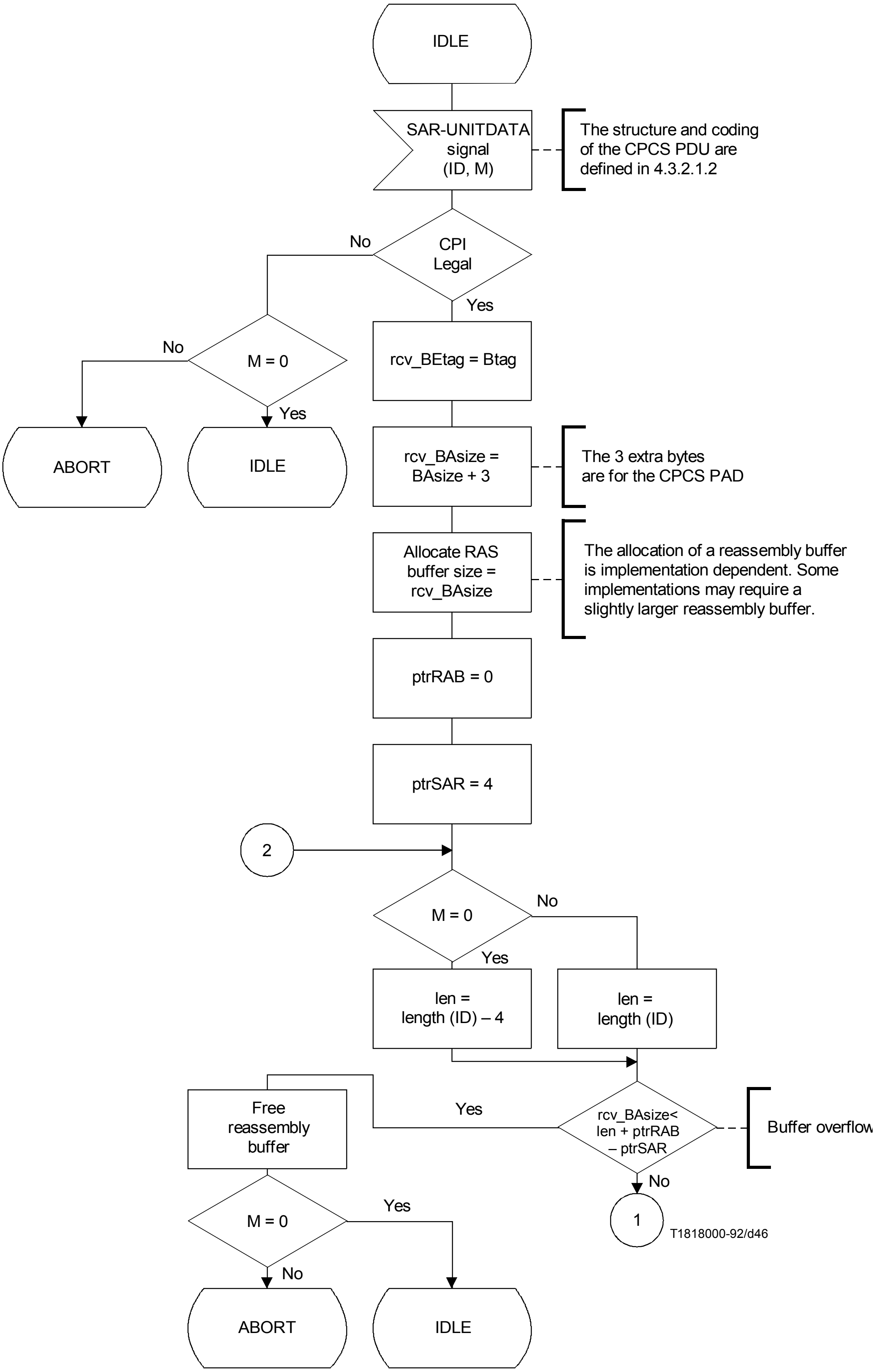


FIGURE I.5/I.363 (sheet 2 of 5)

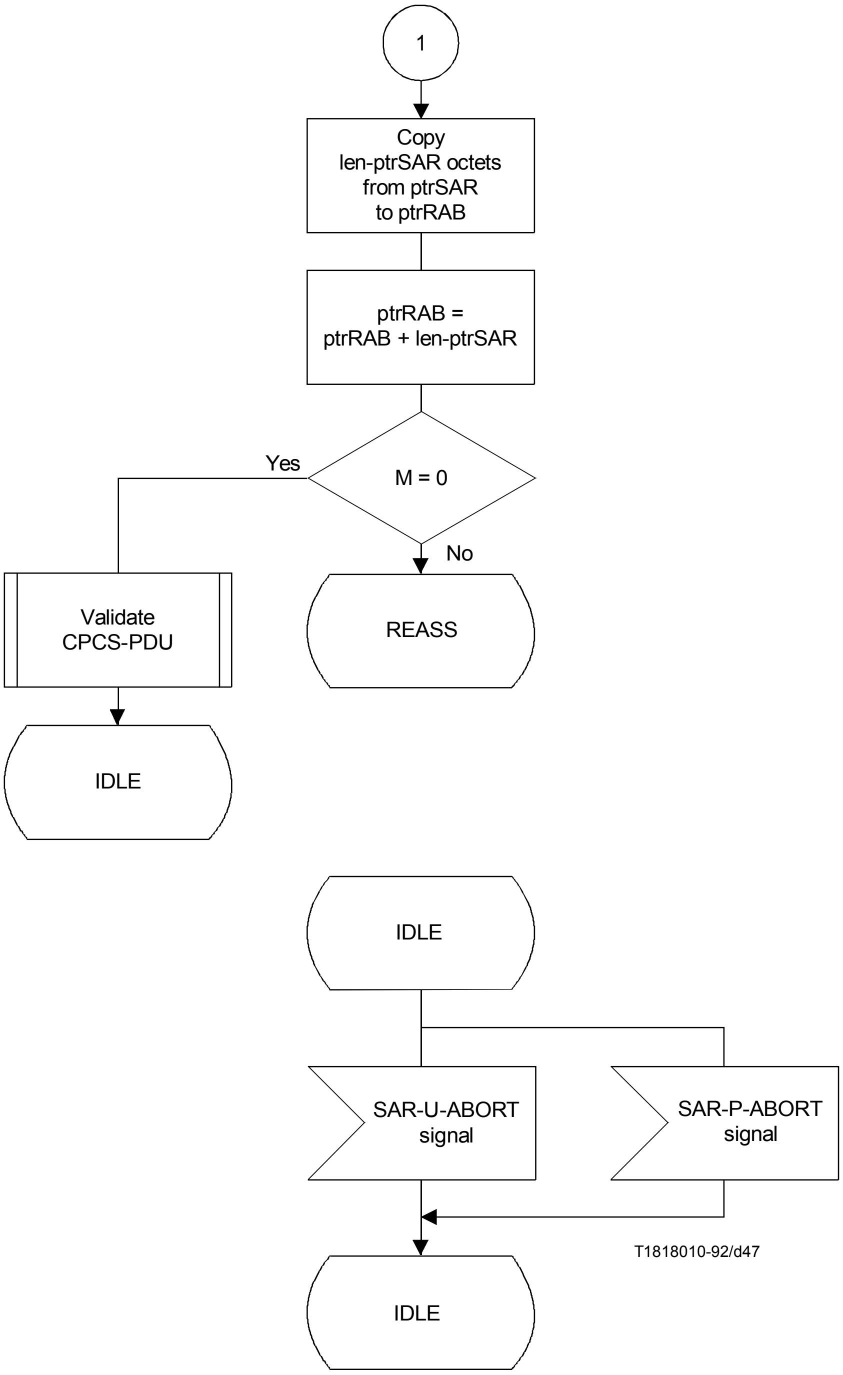
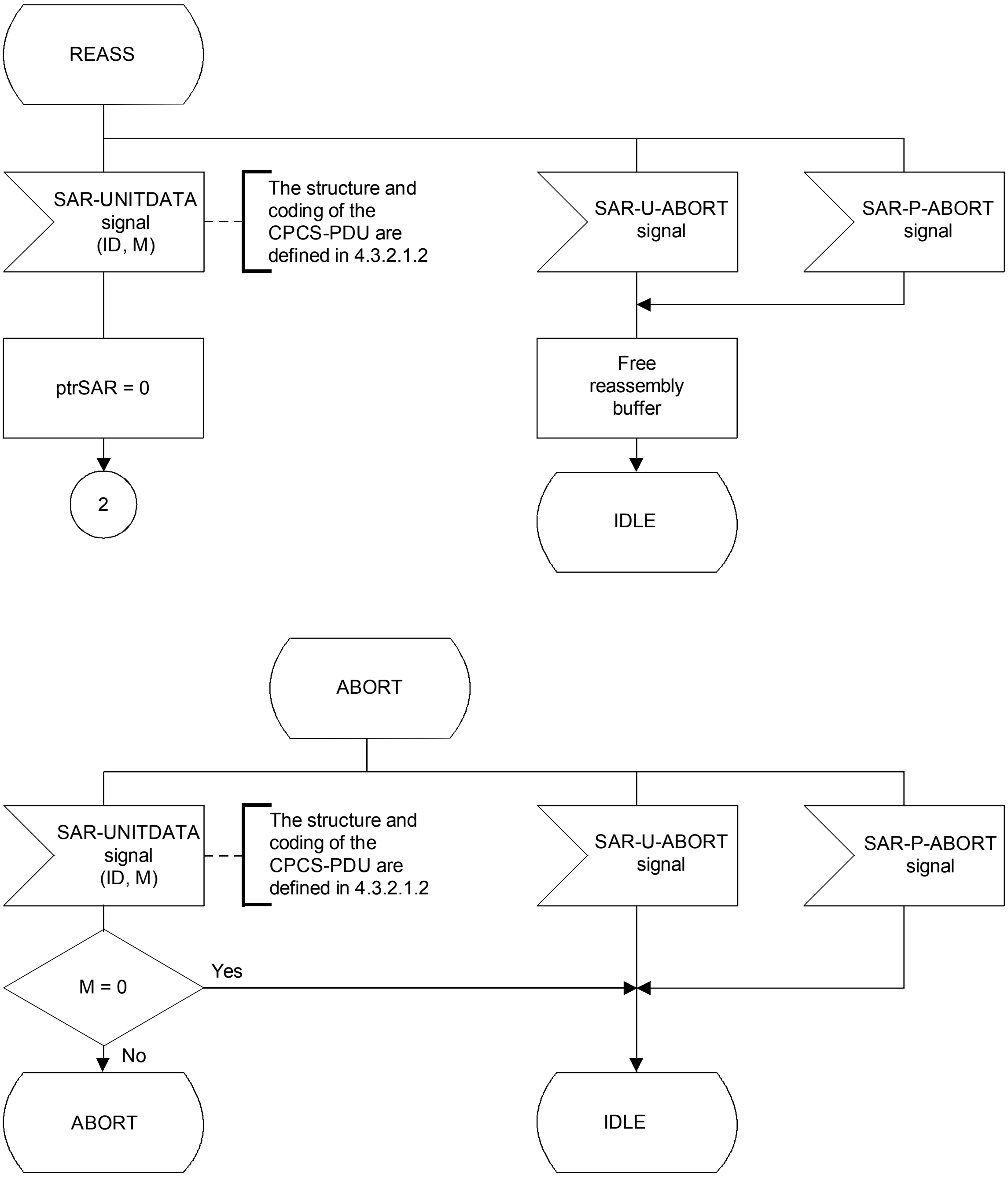


FIGURE I.5/I.363 (sheet 3 of 5)



T1818020-92/d48

FIGURE I.5/I.363 (sheet 4 of 5)

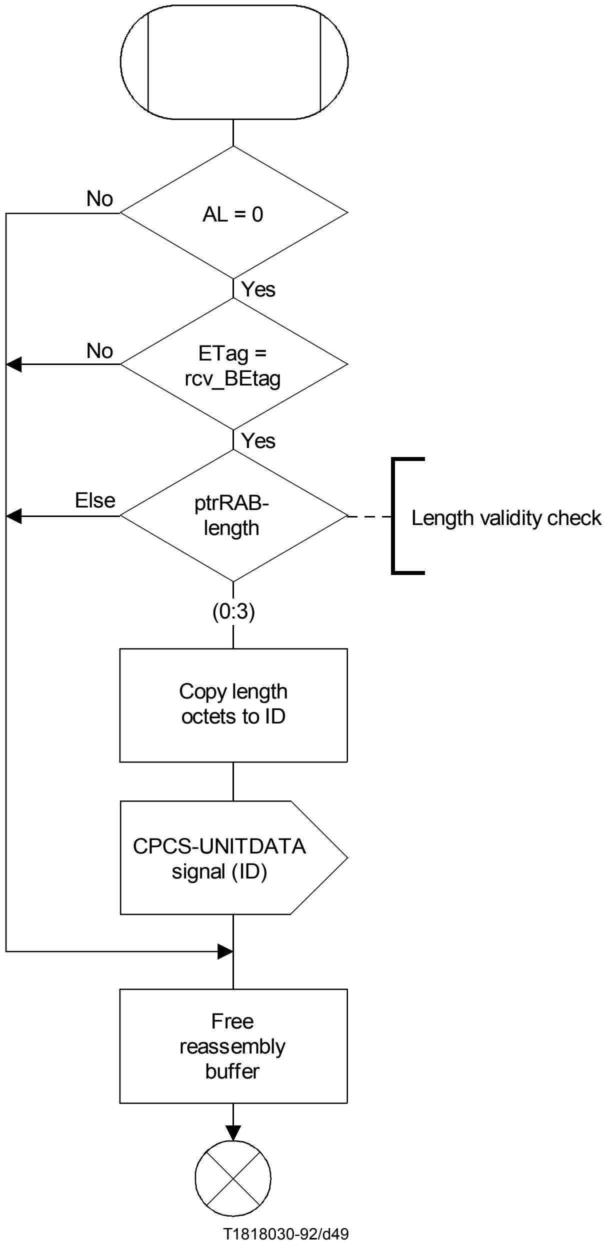


FIGURE I.5/I.363 (sheet 5 of 5)

Printed in Switzerland

Geneva, 1993