The ATM Forum
Technical Committee

Conformance Abstract Test Suite for the ATM Adaptation Layer (AAL) Type 5 Common Part

(Part 1)

af-test-0052.000

March 1996

Conformance Abstract Test Suite for the ATM Adaptation Layer (AAL) Type 5 Common Part (Part 1)

af-test-0052.000

March 1996

The ATM Forum
Worldwide Headquarters
2570 West El Camino Real
Suite 304
Mountain View, CA 94040
Tel: +1-415-949-6700
Fax: +1-415-949-6705

Contents

1. Introduction

This is the first version of the ATM Adaptation Layer (AAL) Type 5 Common Part Abstract Conformance Test Suite (ATS) Part I. This abstract test suite only contains test purposes that can be tested using the Remote Single Layer Test Method. Part II of the AAL Type 5 Common Part ATS for other test purposes which are not included in this ATS, but identified in the test purpose document [8], will be developed in future.

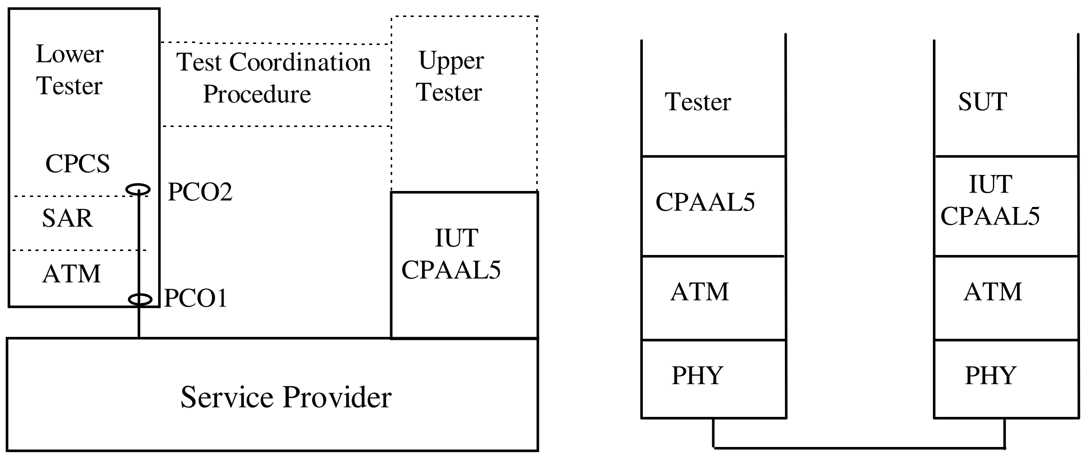
This ATS is based on the ATM Forum ATM User-Network Interface (UNI) Specification, version 3.0 and 3.1 [1][2] and the ITU-T draft Recommendation I.363 [3]. The Protocol Implementation eXtra Information for Testing (PIXIT) proforma for the test cases that are described in this document is provided in Annex A. The PIXIT questionnaire needs to be completed for a particular Implementation Under Test (IUT) prior to conformance testing.

2. Methodology

This Abstract Conformance Test Suite has been developed as described in ISO/IEC International Standard 9646-1 and 2 [4][5]. The notation used in this ATS is Tree and Tabular Combined Notation (TTCN) as described in ISO/IEC IS 9646-3 [6]. Remote Single Layer Test Method, which allows access to the lower boundary of the IUT, is used as the abstract test method for the Part I of the AAL Type 5 Common Part conformance testing.

3. Test Configuration

The Test configuration shown in Figure 1 is used for the AAL Type 5 Common Part protocol conformance testing Part I containing test purposes which require only Lower Tester (LT) function. The test system has two Points of Control and Observation (PCOs) on the LT. The test system is able to control and observe the IUT behaviours through the LT_PCO1 and LT_PCO2. LT_PCO1 is used for observation and control of ATM cells for CPAAL5 primitive parameter testing. LT_PCO2 is used for observation and control of CPCS-PDU.



9646 point of view **Protocol Stack**

Figure 1. Test Configuration for Part I of the AAL Type 5 Common Part Conformance Testing.

4. Test Suite Structure

- CPCS-PDU Structure and Encoding -------- - CPCS-PDU General Structure and Format
  - CPCS-PDU Payload
  - PAD
  - CPCS User-to-User indication (UU)
  - Common Part Indicator (CPI)
  - Length
  - Cyclic Redundancy Code (CRC-32)

- CPAAL5 Processes ---------------------------- - CPAAL5 Sender Side Output

5. Assumptions on the Capabilities of the Tester and IUT

5.1 IUT Implicit Sending

It is assumed that the IUT supports implicit sending of CPCS-PDUs for testing. So, IUT should be made to initiate CPCS-PDUs when it is requested.

5.2 Connection Setup

Prior to testing, it is assumed that the test system provides a means to establish an ATM connection, and manage the VPI/VCI values for that connection depending on the use of Switched Virtual Connections (SVC) or Permanent Virtual Connections (PVC).

If the IUT uses SVC for its ATM connection, in the case the IUT is a part of signalling control plane, the ATM layer connection will be established using signalling point-to-point VPI and VCI values. In the case the IUT is a part of user plane, the values of VPI and VCI will be determined by the signaling control plane. The equipment acting as the network side (IUT or Tester) is responsible for allocating the appropriate VPI and VCI values for testing.

If the IUT uses PVC for its ATM connection, the test laboratory is responsible for accommodating the acceptable values of VPI and VCI for the IUT.

6. Timer

One timer has been defined and used for testing.

T_Opr : A timer which is used to allow sufficient time for test operator intervention. This timer is used conjunction with an "Implicit send" for test coordination.

7. Test Preparation

The PREAMBLE and Test Suite Operation SETUP_CON have been defined to "remind" a test operator to set up an ATM connection between the tester and the IUT using the given VPI and VCI values. Setup_Count is initialized at 0 and incremented at the beginning of each test case before the PREAMBLE is called. The first time the PREAMBLE is executed, it calls the SETUP_CON routine; successive execution of the PREAMBLE will not call this routine.

8. Abbreviations

AAL	ATM Adaptation Layer
ASP	Abstract Service Primitive
ATM	Asynchronous Transfer Mode
ATS	Abstract Test Suite
CI	Congestion Indication
CPAAL5	Common Part AAL Type 5
CPCS	Common Part Convergence Sublayer
CPI	Common Part Indicator
CRC	Cyclic Redundancy Code
IUT	Implementation Under Test
LP	Loss Priority
LT	Lower Tester
PAD	Padding
PCO	Point of Control and Observation
PDU	Protocol Data Unit
PHY	Physical layer
PICS	Protocol Implementation Conformance Statement
PIXIT	Protocol Implementation eXtra Information for Testing
PVC	Permanent Virtual Connection
SAP	Service Access Point
SAR	Segmentation And Reassembly
SDU	Service Data Unit
SUT	System Under Test
SVC	Switched Virtual Connection
TTCN	Tree and Tabular Combined Notation
UNI	User-Network Interface
UT	Upper Tester
UU	User-to-User indication
VCI	Virtual Channel Identifier
VPI	Virtual Path Identifier

9. References

[1] "ATM User-Network Interface Specification, Version 3.0", ATM Forum, 1993.

[2] "ATM User-Network Interface Specification, Version 3.1", ATM Forum, 1994.

[3] ITU-T Draft Recommendation I.363, Section 6 : 1993, B-ISDN ATM Adaptation Layer (AAL) Type 5.

[4] ISO/IEC IS 9646 Part 1, "Information Technology - Open Systems Interconnection - Conformance Testing Methodology and Framework - Part 1 : General concepts", 1993.

[5] ISO/IEC IS 9646 Part 2, "Information Technology - Open Systems Interconnection - Conformance Testing Methodology and Framework - Part 2 : Abstract test suite specification General concepts", 1993.

[6] ISO/IEC IS 9646 Part 3, "Information Technology - Open Systems Interconnection - Conformance Testing Methodology and Framework - Part 3 : Tree and Tabular Combined Notation (TTCN)", 1993.

[7] "PICS Proforma for the AAL Type 5", ATM Forum, af-test-0042.000).

[8] "Test Purposes for the AAL Type 5 Common Part Conformance Test Suite", ATM_Forum/95-0033R2, June, 1995.

Annex A

Protocol Implementation eXtra Information for Testing (PIXIT) Proforma

IUT

Name:

Version:

Machine Configuration:

Operating System Identification:

IUT Identification:

PICS Reference for IUT:

Limitations of the IUT :

General

1. Can the IUT be forced to send CPCS-PDU(s) when requested by test operator?

____

If yes, please answer the following questions.

CPCS-PDU Fields Content

2. Can the content of the field(s) of a CPCS-PDU be known by the tester when it is requested prior to conformance testing?

____

If yes,

2.1 Can the CPCS-SDU length size of the CPCS-PDU be known by the tester when it is requested prior to conformance testing? (LENGTH)

____

2.1.1 Enter the length of the CPCS-SDU for testing. (LENGTHval):(INTEGER)

____

2.2 Can the content of the CPCS-SDU (payload) be known by the tester when it is requested prior to conformance testing? (PAYLOAD)

____

2.2.1 Enter the content of the CPCS-SDU for testing. (PAYLOADval):(OCTETSTRING)

____

____

2.3 Can the content of the one octet CPCS_UU (User-to-User indication) be known by the tester when it is requested prior to conformance testing? (USER)

____

2.3.1 Enter the content of the CPCS_UU field for testing. (USERval):(OCTETSTRING)

____

2.4 Can the pattern of the CPCS-PDU PAD field be known by the tester when it is requested prior to conformance testing? (PADDING)

____

2.4.1 Enter the length of the pattern of the CPCS_PDU PAD field.
(PATTERNlen):(INTEGER)

____

2.4.2 Enter the pattern of the CPCS-PDU PAD field for testing.
(PADDINGval):(OCTETSTRING)

____

2.5 Can the IUT generate a CPCS-SDU with the following specific lengths?
(GENPDU) ____________________

1) 88 octets
2) 58 octets
3) 41 octets

Parameters of the CPCS Primitive

3. Can the value of the CPCS_Congestion Indication parameter be known by the tester when it is requested prior to conformance testing? (CONG_IN)

____

3.1 Enter the value of the CPCS_Congestion Indication parameter for testing.
(CONG_INval):(BITSTRING)

____

4. Can the value of the CPCS_Loss Priority parameter be known by the tester when it is requested prior to testing? (LOSS_PR)

____

4.1 Enter the value of the CPCS_Loss Priority parameter for testing.
(LOSS_PRval):(BITSTRING)

____

CPCS-PDU Sending

5. Does the IUT support generating the CPCS-PDU having the length greater than 48 octets when it is requested for testing? (MULTI)

____

5.1 Enter the number of integral multiples of 48 octets for the CPCS-PDU sent by the IUT for testing. (MULTIval):(INTEGER)

____

6. Enter the maximum size of the CPAAL 5 user data that may be sent to the tester by the IUT. (Max_SDUval):(INTEGER)

____

Annex B

Abstract Test Suite

I

Test Suite Overview

Test Suite Structure

Suite Name : AAL_CP_PART_I

Standards Ref : ATM Forum User Network Interface Specification, Versions 3.0 and 3.1.
ITU−T Draft Recommendation I.363, Section 6, 1993.

PICS Ref : ATM Forum af−test−0042, PICS Proforma for the AAL5.

PIXIT Ref : ATM Forum af−test−0052, Conformance Abstract Test Suite for the ATM Adaptation Layer (AAL) Type 5 Common Part (Part 1), Annex A.

Test Method(s) : Remote Single Layer Test Method.

Comments : Part I of the Abstract Test Suite (ATS) for the AAL Type 5 Common Part protocol conformance testing includes only test purposes that can be tested using the Remote Test Method.

Test Group Reference	Selection Ref	Test Group Objective	Page Nr
AAL_CP_PART_I/		AAL Common Part conformance testing	37
AAL_CP_PART_I/FMT/		CPCS−PDU encoding and format	37
AAL_CP_PART_I/FMT/GEN/		Verify the ability of the IUT to support the CPCS−PDU format.	37
AAL_CP_PART_I/FMT/PAYLOAD/		Verify the format of the CPCS−PDU Payload field.	37
AAL_CP_PART_I/FMT/PAD/		Verify the format of the CPCS−PDU PAD field.	38
AAL_CP_PART_I/FMT/UU/		Verify the format of the CPCS−PDU UU field.	42
AAL_CP_PART_I/FMT/CPI/		Verify the format of the CPCS−PDU CPI field.	42
AAL_CP_PART_I/FMT/LENGTH/		Verify the format of the CPCS−PDU Length field.	43
AAL_CP_PART_I/FMT/CRC/		Verify the format of the CPCS−PDU CRC field.	44
AAL_CP_PART_I/PROC/		CPAAL5 Processing	45
AAL_CP_PART_I/PROC/SEND/		Verify the behaviours of the IUT sender side.	45

Detailed Comments :

Test Case Index				
Test Group Reference	**Test Case Id**	**Selection Ref**	**Description**	**Page Nr**
AAL_CP_PART_I/FMT/GEN/	FMT_GEN_1	CPCS_FIELDS	Verify that the IUT supports the CPCS−PDU format as specified.	37
AAL_CP_PART_I/FMT/PAYLOAD/	FMT_PAYLOAD_2	CPCS_FIELDS	Verify that the IUT puts the CPCS−SDU into the CPCS−PDU payload field when the value of the Length field is greater than zero.	37
AAL_CP_PART_I/FMT/PAD/	FMT_PAD_3		Verify that the IUT limits the PAD field to 47 octets.	38
AAL_CP_PART_I/FMT/PAD/	FMT_PAD_4	CPCS_PADDING	Verify that the IUT uses the PAD field to complement the CPCS−PDU to an integral multiple of 48 octets.	38
AAL_CP_PART_I/FMT/PAD/	FMT_PAD_5	PAD_CHK	Verify that the IUT uses the PAD field (length of PAD =0) to complement the CPCs−PDU (length of CPCS−SDU = 88) to an integral multiple of 48 octets.	39
AAL_CP_PART_I/FMT/PAD/	FMT_PAD_6	PAD_CHK_MORE	Verify that the IUT uses the PAD field (length of PAD = 30) to complement the CPCS−PDU (length of CPCS−SDU = 58) to an integral multiple of 48 octets.	40
AAL_CP_PART_I/FMT/PAD/	FMT_PAD_7	PAD_CHK_MORE	Verify that the IUT uses the PAD field (length of PAD = 47) to complement the CPCS−PDU (length of CPCS−SDU =41) to an integral multiple of 48 octets.	41
AAL_CP_PART_I/FMT/UU/	FMT_UU_8	CPCS_USER	Verify that the IUT uses CPCS_UU field to transfer transparently CPCS User_to_User information.	42
AAL_CP_PART_I/FMT/CPI/	FMT_CPI_9		Verify that the IUT encodes the CPI field to all zeros when only the 64−bit alignment function is used.	42
AAL_CP_PART_I/FMT/LENGTH/	FMT_LENGTH_10	CPCS_LENGTH	Verify that the IUT encodes the Length field with the number of octet of CPCS−PDU payload.	43

Continued on next page

Continued from previous page

Test Case Index				
Test Group Reference	**Test Case Id**	**Selection Ref**	**Description**	**Page Nr**
AAL_CP_PART_I/FMT/LENGTH/	FMT_LENGTH_11	ABORT	Verify that the IUT encodes the Length field to all zeros when the Abort function is invoked.	43
AAL_CP_PART_I/FMT/CRC/	FMT_CRC_12		Verify that the IUT uses the CRC−32 to detect bit errors in the CPCS−PDU.	44
AAL_CP_PART_I/PROC/SEND/	PROC_SEND_13	MULTI_PDU	Verify that the IUT generates more than one SAR−PDU, if the CPCS−PDU has the length greater than 48 octets.	45
AAL_CP_PART_I/PROC/SEND/	PROC_SEND_14	MULTI_PDU	Verify that the IUT transfers all segments of a CPCS−PDU except the last, to the ATM layer with AUU parameter in the ATM−DATA.request primitive equal to zero.	46
AAL_CP_PART_I/PROC/SEND/	PROC_SEND_15	MULTI_PDU	Verify that the IUT transfers the last segment of a CPCS−PDU, to the ATM layer with AUU parameter in the ATM−DATA.request primitive equal to one.	47
AAL_CP_PART_I/PROC/SEND/	PROC_SEND_16	CPCS_CI	Verify that the IUT transfers all segments of a CPCS−PDU to the ATM layer with an indication of congestion set to the value of the corresponding CPCS−CI parameter received from the CPAAL5 user.	48
AAL_CP_PART_I/PROC/SEND/	PROC_SEND_17	CPCS_LP	Verify that the IUT transfers all segments of a CPCS−PDU to the ATM layer with an indication of loss priority set to the value of the corresponding CPCS_LP parameter received from the CPAAL5 user.	49
Detailed Comments :				

Test Step Index			
Test Step Group Reference	**Test Step Id**	**Description**	**Page Nr**
AAL_CP_PART_I/	PREAMBLE	Setup the connection.	50
Detailed Comments :			

II

Declarations Part

Simple Type Definitions		
Type Name	**Type Definition**	**Comments**
OCTSTRING0to47	OCTETSTRING[0..47]	
OCTSTRING2	OCTETSTRING[2]	
OCTSTRING1	OCTETSTRING[1]	
Detailed Comments :		

Structured Type Definition		
Type Name : SDUDATA		
Comments :		
Element Name	**Type Definition**	**Comments**
S1	OCTETSTRING[1..Max_SDUval]	
Detailed Comments :		

Structured Type Definition		
Type Name : PATT		
Comments :		
Element Name	**Type Definition**	**Comments**
P1	OCTETSTRING[1..PATTERNlen]	
Detailed Comments :		

Test Suite Operation Definition
Operation Name : PAD_CHECK(PADVALUE:OCTSTRING0to47;PATTERN:PATT)
Result Type : BOOLEAN
Comments :
Description
This operation checks whether the octetstring "PADVALUE" is coded as the coding pattern "PATTERN". If the "PADVALUE" is coded as the coding pattern rule, then this operation returns TRUE. Else, FALSE is returned.
Detailed Comments :

Test Suite Operation Definition
Operation Name : GET_PAD(PADPATTERN:PATT;LENGTH_PAY:OCTSTRING2)
Result Type : OCTETSTRING
Comments :
Description
This operation creates a PAD field with the given "PADPATTERN" type for 48 octet alignment of the CPCS−PDU trailer , when the length of CPCS−PDU payload is "LENGTH_PAY", as specified in section 6.3.2.1.1 of I.363 recommendation. The returned PAD field is from 0 to 47 octets.
Detailed Comments :

Test Suite Operation Definition
Operation Name : CRC_CHECK(A:SDUDATA;B:OCTSTRING0to47;C,D:OCTSTRING1;E:OCTSTRING2)
Result Type : OCTETSTRING
Comments : The octetstring "A" could have up to the Max_SDUval octets.The octetstring "B" could have up to 47 octets. The octetstring "C" and "D" have one octet. The octetstring "E" has two octets.
Description
This operation calculates the proper value of the CPCS−PDU CRC field for the combined octetstrings (A,B,C,D, and E). This operation returns a four octet valid CRC value.
Detailed Comments :

Test Suite Operation Definition
Operation Name : INT_TO_OCT(A,B:INTEGER) **Result Type** : OCTETSTRING **Comments** :
Description
This operation converts a single INTEGER value to a single OCTETSTRING value. The resulting string is "B" length octets long.
Detailed Comments :

Test Suite Operation Definition
Operation Name : SETUP_CON(VPI,VCI:BITSTRING) **Result Type** : BOOLEAN **Comments** : Setup an ATM connection for testing.
Description
Setup an ATM connection between the tester and the IUT using the given VPI and VCI values. If the connection is established successfully, then this operation returns TRUE. Else, FALSE is returned.
Detailed Comments :

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
PAYLOAD	BOOLEAN	PIXIT 2.2	Can the content of the CPCS−SDU be known by the tester when it is requested prior to testing?
PAYLOADval	SDUDATA	PIXIT 2.2.1	Enter the content of the CPCS−SDU for testing.
USER	BOOLEAN	PIXIT 2.3	Can the content of the CPCS_UU be known by the tester when it is requested prior to testing?
USERval	OCTETSTRING	PIXIT 2.3.1	Enter the content of the one octet CPCS_UU field for testing.
LENGTH	BOOLEAN	PIXIT 2.1	Can the CPCS−SDU length size of the CPCS−PDU be known by the tester when it is requested prior to testing?
LENGTHval	INTEGER	PIXIT 2.1.1	Enter the length of the CPCS−SDU for testing.
ABORTpar	BOOLEAN	PICS 3.4.2.7	Does the IUT support the abort function?
TOPRtime	INTEGER		Enter the sufficient amount of time to allow test operator intervention.
PADDING	BOOLEAN	PIXIT 2.4	Can the content of the CPCS−PDU PAD field be known by the tester when it is requested prior to testing?
PATTERNlen	INTEGER	PIXIT 2.4.1	Enter the length of the pattern of the CPCS−PDU PAD field.
PADDINGval	PATT	PIXIT 2.4.2	Enter the content of the CPCS−PDU PAD field for testing.
GENPDU	BOOLEAN	PIXIT 2.5	Can the IUT generate a CPCS−SDU with following specific value? 1) 88 octets 2) 58 octets 3) 41 octets

Continued on next page

Continued from previous page

Test Suite Parameter Declarations			
Parameter Name	**Type**	**PICS/PIXIT Ref**	**Comments**
MULTI	BOOLEAN	PIXIT 5	Does the IUT support generating the CPCS−PDU having the length greater than 48 octets when it is requested for testing?
MULTIval	INTEGER	PIXIT 5.1	Enter the number of integral multiples of 48 octets for the CPCS−PDU sent by the IUT for testing.
CONG_IN	BOOLEAN	PIXIT 3	Can the value of the CPCS_Congestion Indication parameter be known by the tester when it is requested prior to testing?
CONG_INval	BITSTRING	PIXIT 3.1	Enter the value of the CPCS_Congestion Indication parameter for testing.
LOSS_PR	BOOLEAN	PIXIT 4	Can the value of the CPCS_Loss Priority parameter be known by the tester when it is requested prior to testing?
LOSS_PRval	BITSTRING	PIXIT 4.1	Enter the value of the CPCS_Loss Priority parameter for testing.
Max_SDUval	INTEGER	PIXIT 6	Enter the maximum size of the CPAAL5 user data that may be sent to the tester by the IUT.
VPIval	BITSTRING		The VPI value for testing.
VCIval	BITSTRING		The VCI value for testing.
Detailed Comments :			

Test Case Selection Expression Definitions		
Expression Name	**Selection Expression**	**Comments**
CPCS_FIELDS	PAYLOAD AND LENGTH AND USER AND PADDING	Tester can know the CPCS–SDU, , length of CPCS–SDU, CPCS User–to_User data, and PAD contents prior to testing.
CPCS_PAYLOAD	PAYLOAD	Tester can know the CPCS–SDU content prior to testing.
CPCS_USER	USER	Tester can know the CPCS–UU content prior to testing.
CPCS_LENGTH	LENGTH	Tester can know the length size of the CPCS–SDU prior to testing.
ABORT	ABORTpar	The IUT supports the abort function.
CPCS_PADDING	PADDING AND LENGTH	Tester can know the pattern of the CPCS–PDU PAD field prior to testing.
PAD_CHK	GENPDU	The IUT can generate a CPCS–SDU with specific (88, 58, 41 octets) lengths.
PAD_CHK_MORE	PADDING AND GENPDU	Tester can know the pattern of CPCS–PDU PAD field prior to testing and the IUT can generate a CPCS–SDU with specific (88, 58, 41 octets) lengths.
MULTI_PDU	MULTI	Tester can know the number of ingegral multiples of 48 octets for the CPCS–PDU sent by IUT prior to testing.
CPCS_CI	CONG_IN	Tester can know the value of the CPCS_CI parameter prior to testing.
CPCS_LP	LOSS_PR	Tester can know the value of the CPCS_LP parameter prior to testing.
Detailed Comments :		

Test Suite Constant Declarations			
Constant Name	**Type**	**Value**	**Comments**
LENGTH_O	OCTETSTRING	INT_TO_OCT(LENGTHval,2)	Expected Length field value from the IUT.
Detailed Comments :			

Test Suite Variable Declarations			
Variable Name	**Type**	**Value**	**Comments**
PAD_VALUE	OCTETSTRING		PAD contents
CRC_RST	OCTETSTRING		CRC checking result
CRC_VALUE	OCTETSTRING		Saving the CRC value
COUNTER	INTEGER		Counter
AUU_R	BITSTRING		AUU value
CI_R	BITSTRING		CI value
CLP_R	BITSTRING		CLP value
Setup_Count	INTEGER	0	
Detailed Comments :			

PCO Declarations			
PCO Name	**PCO Type**	**Role**	**Comments**
LT_PCO1	PHYSAP	LT	PHY layer service access point (SAP) on the lower tester 1.
LT_PCO2	L_CPCS	LT	Lower boundary of CPCS.
Detailed Comments :			

Timer Declarations			
Timer Name	**Duration**	**Unit**	**Comments**
T_Opr	TOPRtime	s	The sufficient amount of time to allow test operator intervention.
Detailed Comments :			

TTCN PDU Type Definition

PDU Name : ATM_CELL
PCO Type : PHYSAP
Comments :

Field Name	Field Type	Comments
GFC	BITSTRING[4]	
VPI	BITSTRING[8]	
VCI	BITSTRING[16]	
PTI_U	BITSTRING[1]	User Data Cell
PTI_CI	BITSTRING[1]	Congestion Indication
PTI_AUU	BITSTRING[1]	AUU parameter
CLP	BITSTRING[1]	
HEC	OCTSTRING1	
Payload	OCTETSTRING[48]	

Detailed Comments :

TTCN PDU Type Definition

PDU Name : CPCS_PDU
PCO Type : L_CPCS
Comments :

Field Name	Field Type	Comments
Payload	SDUDATA	
PAD	OCTSTRING0to47	
UU	OCTSTRING1	
CPI	OCTSTRING1	
Length	OCTSTRING2	
CRC	OCTETSTRING[4]	

Detailed Comments :

III

Constraints Part

TTCN PDU Constraint Declaration

Constraint Name : CPCS_PDU_I_1
PDU Type : CPCS_PDU
Derivation Path :
Comments :

Field Name	Field Value	Comments
Payload	PAYLOADval	
PAD	GET_PAD(PADDINGval,LENGTH_O)	
UU	USERval	
CPI	’00’O	
Length	LENGTH_O	
CRC	CRC_CHECK(PAYLOADval,GET_PAD (PADDINGval,LENGTH_O),USERval,’0 0’O,LENGTH_O)	

Detailed Comments : Valid CPCS−PDU (received from the IUT).

TTCN PDU Constraint Declaration

Constraint Name : CPCS_PDU_I_2
PDU Type : CPCS_PDU
Derivation Path :
Comments :

Field Name	Field Value	Comments
Payload	PAYLOADval	
PAD	?	
UU	?	
CPI	?	
Length	?	
CRC	?	

Detailed Comments : Valid CPCS−PDU (received from the IUT).

TTCN PDU Constraint Declaration

Constraint Name : CPCS_PDU_I_3
PDU Type : CPCS_PDU
Derivation Path :
Comments :

Field Name	Field Value	Comments
Payload	?	
PAD	?	
UU	?	
CPI	?	
Length	LENGTH_O	
CRC	?	

Detailed Comments : Valid CPCS−PDU (received from the IUT).

TTCN PDU Constraint Declaration

Constraint Name : CPCS_PDU_I_4
PDU Type : CPCS_PDU
Derivation Path :
Comments :

Field Name	Field Value	Comments
Payload	?	
PAD	?	
UU	USERval	
CPI	?	
Length	?	
CRC	?	

Detailed Comments : Valid CPCS−PDU (received from the IUT).

TTCN PDU Constraint Declaration

Constraint Name : CPCS_PDU_I_5
PDU Type : CPCS_PDU
Derivation Path :
Comments :

Field Name	Field Value	Comments
Payload	?	
PAD	?	
UU	?	
CPI	’00’O	
Length	?	
CRC	?	

Detailed Comments : Valid CPCS−PDU (received from the IUT).

TTCN PDU Constraint Declaration

Constraint Name : CPCS_PDU_I_6
PDU Type : CPCS_PDU
Derivation Path :
Comments :

Field Name	Field Value	Comments
Payload	?	
PAD	?	
UU	?	
CPI	?	
Length	LENGTH_O	
CRC	?	

Detailed Comments : Valid CPCS−PDU (received from the IUT).

TTCN PDU Constraint Declaration

Constraint Name : CPCS_PDU_P1
PDU Type : CPCS_PDU
Derivation Path :
Comments : CPCS−PDU with 88 octets CPCS−SDU size.

Field Name	Field Value	Comments
Payload	?	
PAD	–	
UU	?	
CPI	?	
Length	'58'O	
CRC	?	

Detailed Comments : Valid CPCS−PDU (received from the IUT).

TTCN PDU Constraint Declaration

Constraint Name : CPCS_PDU_P2
PDU Type : CPCS_PDU
Derivation Path :
Comments : CPCS−PDU with 58 octets CPCS−SDU size.

Field Name	Field Value	Comments
Payload	?	
PAD	?	
UU	?	
CPI	?	
Length	'3A'O	
CRC	?	

Detailed Comments : Valid CPCS−PDU (received from the IUT).

TTCN PDU Constraint Declaration

Constraint Name : CPCS_PDU_P3
PDU Type : CPCS_PDU
Derivation Path :
Comments : CPCS−PDU with 41 octets CPCS−SDU size.

Field Name	Field Value	Comments
Payload	?	
PAD	?	
UU	?	
CPI	?	
Length	’29’O	
CRC	?	

Detailed Comments : Valid CPCS−PDU (received from the IUT).

TTCN PDU Constraint Declaration

Constraint Name : CPCS_PDU_A
PDU Type : CPCS_PDU
Derivation Path :
Comments :

Field Name	Field Value	Comments
Payload	?	
PAD	?	
UU	?	
CPI	?	
Length	’0000’O	
CRC	?	

Detailed Comments : Valid CPCS−PDU when Abort function is invoked (received from the IUT).

TTCN PDU Constraint Declaration

Constraint Name : ATM_CELL_0
PDU Type : ATM_CELL
Derivation Path :
Comments : ATM Cell (AUU = 0)

Field Name	Field Value	Comments
GFC	?	
VPI	VPIval	
VCI	VCIval	
PTI_U	'0'B	User Data Cell
PTI_CI	?	Congestion Indication
PTI_AUU	'0'B	AUU parameter
CLP	?	
HEC	?	
Payload	?	

Detailed Comments :

TTCN PDU Constraint Declaration

Constraint Name : ATM_CELL_1
PDU Type : ATM_CELL
Derivation Path :
Comments : ATM Cell (AUU = 1)

Field Name	Field Value	Comments
GFC	?	
VPI	VPIval	
VCI	VCIval	
PTI_U	'0'B	User Data Cell
PTI_CI	?	Congestion Indication
PTI_AUU	'1'B	AUU parameter
CLP	?	
HEC	?	
Payload	?	

Detailed Comments :

TTCN PDU Constraint Declaration

Constraint Name : ATM_CELL_ANY
PDU Type : ATM_CELL
Derivation Path :
Comments : ATM Cell

Field Name	Field Value	Comments
GFC	?	
VPI	VPIval	
VCI	VCIval	
PTI_U	'0'B	User Data Cell
PTI_CI	?	Congestion Indication
PTI_AUU	?	AUU parameter
CLP	?	
HEC	?	
Payload	?	

Detailed Comments :

TTCN PDU Constraint Declaration

Constraint Name : CELL_UNASSIGNED
PDU Type : ATM_CELL
Derivation Path :
Comments : Unassigned ATM Cell

Field Name	Field Value	Comments
GFC	?	
VPI	'00000000'B	
VCI	'0000000000000000'B	
PTI_U	?	
PTI_CI	?	Congestion Indication
PTI_AUU	?	AUU parameter
CLP	'0'B	
HEC	?	
Payload	?	

Detailed Comments :

IV

Dynamic Part

Test Case Dynamic Behaviour

Test Case Name : FMT_GEN_1
Group : AAL_CP_PART_I/FMT/GEN/
Purpose : Verify that the IUT supports the CPCS–PDU format as specified.
Configuration :
Default :
Comments : Ref. 6.3.2 / PICS 3.4.4.1.1

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_I_1		IUT sends a CPCS–PDU.
4		START T_Opr			
5		LT_PCO2?CPCS_PDU	CPCS_PDU_I_1	P	Correct CPCS–PDU
6		LT_PCO2?OTHERWISE		F	Incorrect CPCS–PDU
7		?TIMEOUT T_Opr		I	

Detailed Comments : Selection Ref. : CPCS_FIELDS

Test Case Dynamic Behaviour

Test Case Name : FMT_PAYLOAD_2
Group : AAL_CP_PART_I/FMT/PAYLOAD/
Purpose : Verify that the IUT puts the CPCS–SDU into the CPCS–PDU payload field when the value of the Length field is greater than zero.
Configuration :
Default :
Comments : Ref. 6.3.2.1 / PICS 3.4.4.2.2

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_I_2		IUT sends a CPCS–PDU.
4		START T_Opr			
5		LT_PCO2?CPCS_PDU	CPCS_PDU_I_2	P	Correct CPCS–PDU payload
6		LT_PCO2?OTHERWISE		F	Incorrect CPCS–PDU payload
7		?TIMEOUT T_Opr		I	

Detailed Comments : Selection Ref. : CPCS_PAYLOAD

Test Case Dynamic Behaviour

Test Case Name : FMT_PAD_3
Group : AAL_CP_PART_I/FMT/PAD/
Purpose : Verify that the IUT limits the PAD field to 47 octets.
Configuration :
Default :
Comments : Ref. 6.3.2.1.2 / PICS 3.4.4.3.1

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_I_3		IUT sends a CPCS−PDU.
4		START T_Opr			
5		LT_PCO2?CPCS_PDU(PAD_VALUE: =CPCS_PDU.PAD)	CPCS_PDU_I_3		
6		[LENGTH_OF(PAD_VALUE)<=47]		P	
7		[LENGTH_OF(PAD_VALUE)>47]		F	Incorrect CPCS−PDU PAD length.
8		?TIMEOUT T_Opr		I	

Detailed Comments :

Test Case Dynamic Behaviour

Test Case Name : FMT_PAD_4
Group : AAL_CP_PART_I/FMT/PAD/
Purpose : Verify that the IUT uses the PAD field to complement the CPCS−PDU to an integral multiple of 48 octets.
Configuration :
Default :
Comments : Ref. 6.3.2.1.2 / PICS 3.4.4.3.2

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_I_3		IUT sends a CPCS−PDU.
4		START T_Opr			
5		LT_PCO2?CPCS_PDU(PAD_VALUE: =CPCS_PDU.PAD)	CPCS_PDU_I_3		
6		[(PAD_CHECK(PAD_VALUE,PADD INGval))=TRUE]		P	Correct use of the PAD field.
7		[(PAD_CHECK(PAD_VALUE,PADD INGval))=FALSE]		F	Incorrect use of the PAD field.
8		?TIMEOUT T_Opr		I	

Detailed Comments : Selection Ref. : CPCS_PADDING

Test Case Dynamic Behaviour

Test Case Name : FMT_PAD_5
Group : AAL_CP_PART_I/FMT/PAD/
Purpose : Verify that the IUT uses the PAD field (length of PAD = 0) to complement the CPCS−PDU (length of CPCS−SDU = 88) to an integral multiple of 48 octets.
Configuration :
Default :
Comments : Ref. 6.3.2.1.2 / PICS 3.4.4.3.2

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_P1		IUT sends a CPCS−PDU with 88 octets CPCS−SDU.
4		START T_Opr			
5		LT_PCO2?CPCS_PDU	CPCS_PDU_P1	P	Correct use of the PAD field.
6		LT_PCO2?OTHERWISE		F	Incorrect use of the PAD field.
7		?TIMEOUT T_Opr		I	

Detailed Comments : Selection Ref. : PAD_CHK

Test Case Dynamic Behaviour

Test Case Name : FMT_PAD_6
Group : AAL_CP_PART_I/FMT/PAD/
Purpose : Verify that the IUT uses the PAD field (length of PAD = 30) to complement the CPCS−PDU (length of CPCS−SDU = 58) to an integral multiple of 48 octets.
Configuration :
Default :
Comments : Ref. 6.3.2.1.2 / PICS 3.4.4.3.2

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_P2		IUT sends a CPCS−PDU.
4		START T_Opr			
5		LT_PCO2?CPCS_PDU(PAD_VALUE:=CPCS_PDU.PAD)	CPCS_PDU_P2		
6		[(PAD_CHECK(PAD_VALUE,PADDINGval))=TRUE]		P	Correct use of the PAD field.
7		[(PAD_CHECK(PAD_VALUE,PADDINGval))=FALSE]		F	Incorrect use of the PAD field.
8		LT_PCO2?OTHERWISE		F	Incorrect PAD field length.
9		?TIMEOUT T_Opr		I	

Detailed Comments : Selection Ref. : PAD_CHK_MORE

Test Case Dynamic Behaviour

Test Case Name : FMT_PAD_7
Group : AAL_CP_PART_I/FMT/PAD/
Purpose : Verify that the IUT uses the PAD field (length of PAD = 47) to complement the CPCS−PDU (length of CPCS−SDU = 41) to an integral multiple of 48 octets.
Configuration :
Default :
Comments : Ref. 6.3.2.1.2 / PICS 3.4.4.3.2

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_P3		IUT sends a CPCS−PDU with 41 octets CPCS−SDU.
4		START T_Opr			
5		LT_PCO2?CPCS_PDU(PAD_VALUE:=CPCS_PDU.PAD)	CPCS_PDU_P3		
6		[(PAD_CHECK(PAD_VALUE,PADDINGval))=TRUE]		P	Correct use of the PAD field.
7		[(PAD_CHECK(PAD_VALUE,PADDINGval))=FALSE]		F	Incorrect use of the PAD field.
8		LT_PCO2?OTHERWISE		F	Incorrect PAD field length.
9		?TIMEOUT T_Opr		I	

Detailed Comments : Selection Ref. : PAD_CHK_MORE

Test Case Dynamic Behaviour

Test Case Name : FMT_UU_8
Group : AAL_CP_PART_I/FMT/UU/
Purpose : Verify that the IUT uses CPCS_UU field to transfer transparently CPCS User_to_User information.
Configuration :
Default :
Comments : Ref. 6.3.2.1.2 / PICS 3.4.3.2, 3.4.4.4.2, 3.5.2.8

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_I_4		IUT sends a CPCS−PDU.
4		START T_Opr			
5		LT_PCO2?CPCS_PDU	CPCS_PDU_I_4	P	Correct CPCS User_to_User information.
6		LT_PCO2?OTHERWISE		F	Incorrect CPCS User_to_User information.
7		?TIMEOUT T_Opr		I	

Detailed Comments : Selection Ref. : CPCS_USER

Test Case Dynamic Behaviour

Test Case Name : FMT_CPI_9
Group : AAL_CP_PART_I/FMT/CPI/
Purpose : Verify that the IUT encodes the CPI field to all zeros when only the 64−bit alignment function is used.
Configuration :
Default :
Comments : Ref. 6.3.2.1.2 / PICS 3.4.4.5.2

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_I_5		IUT sends a CPCS−PDU.
4		START T_Opr			
5		LT_PCO2?CPCS_PDU	CPCS_PDU_I_5	P	CPI field is encoded to all zeros.
6		LT_PCO2?OTHERWISE		F	CPI field is not encoded to all zeros.
7		?TIMEOUT T_Opr		I	

Detailed Comments :

Test Case Dynamic Behaviour

Test Case Name : FMT_LENGTH_10
Group : AAL_CP_PART_I/FMT/LENGTH/
Purpose : Verify that the IUT encodes the Length filed with the number of octet of CPCS−PDU payload.
Configuration :
Default :
Comments : Ref. 6.3.2.1.2 / PICS 3.4.4.6.2

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_I_6		IUT sends a CPCs−PDU.
4		START T_Opr			
5		LT_PCO2?CPCS_PDU	CPCS_PDU_I_6	P	Correct Length field.
6		LT_PCO2?OTHERWISE		F	Incorrect length field.
7		?TIMEOUT T_Opr		I	

Detailed Comments : Selection Ref. : CPCS_LENGTH

Test Case Dynamic Behaviour

Test Case Name : FMT_LENGTH_11
Group : AAL_CP_PART_I/FMT/LENGTH/
Purpose : Verify that the IUT encodes the Length field to all zeros when the Abort function is invoked.
Configuration :
Default :
Comments : Ref. 6.3.2.1.2 / PICS 3.4.4.6.3

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_A		Invoke Abort function
4		START T_Opr			
5		LT_PCO2?CPCS_PDU	CPCS_PDU_A	P	The Length field is encoded to all zeros.
6		LT_PCO2?OTHERWISE		F	Teh Length field is not encoded to all zeros.
7		?TIMEOUT T_Opr		I	

Detailed Comments : Selection Ref. : ABORT
IUT should invoke the Abort function prior to CPCS−PDU sending.

Test Case Dynamic Behaviour

Test Case Name : FMT_CRC_12
Group : AAL_CP_PART_I/FMT/CRC/
Purpose : Verify that the IUT uses the CRC−32 to detect bit errors in the CPCS−PDU.
Configuration :
Default :
Comments : Ref. 6.3.2.1.2 / PICS 3.4.4.7.2

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_I_3		IUT sends a CPCS−PDU.
4		START T_Opr			
5		LT_PCO2?CPCS_PDU(CRC_RST:=CRC_CHECK(CPCS_PDU.Payload,CPCS_PDU.PAD,CPCS_PDU.UU,CPCS_PDU.CPI,CPCS_PDU.Length), CRC_VALUE:=CPCS_PDU.CRC)	CPCS_PDU_I_3		
6		[CRC_RST=CRC_VALUE]		P	Correct coding of CRC field.
7		[CRC_RST<>CRC_VALUE]		F	Incorrect coding of CRC field.
8		?TIMEOUT T_Opr		I	

Detailed Comments :

Test Case Dynamic Behaviour

Test Case Name : PROC_SEND_13
Group : AAL_CP_PART_I/PROC/SEND/
Purpose : Verify that the IUT generates more than one SAR−PDU, if the CPCS−PDU has the length greater than 48 octets.
Configuration :
Default :
Comments : Ref. 6.4.1.2 / PICS 3.5.1.1

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1,COUNTER:=0)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_I_3		IUT sends the CPCS−PDU which has the length greater than 48 octets.
4		START T_Opr			
5	LB1	LT_PCO1?ATM_CELL	ATM_CELL_1		ATM Cell payload had the last (or complete) SAR−SDU.
6		(COUNTER:=COUNTER+1)			
7		[COUNTER=MULTIval]		P	
8		[COUNTER<>MULTIval]		F	
9		LT_PCO1?ATM_CELL	ATM_CELL_0		ATM cell payload has not the last (or complete) SAR−SDU.
10		(COUNTER:=COUNTER+1)			
11		GOTO LB1			
12		LT_PCO1?ATM_CELL	CELL_UNASSIGNED		Lower tester receives an unassigned cell.
13		GOTO LB1			
14		LT_PCO1?OTHERWISE		F	
15		?TIMEOUT T_Opr		I	

Detailed Comments : Selection Ref. : MULTI_PDU

Test Case Dynamic Behaviour

Test Case Name : PROC_SEND_14
Group : AAL_CP_PART_I/PROC/SEND/
Purpose : Verify that the IUT transfers all segments of a CPCS−PDU except the last, to the ATM layer with AUU parameter in the ATM−DATA.request primitive equal to zero.
Configuration :
Default :
Comments : Ref. 6.4.1.2 / PICS 3.4.2.1

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1,COUNTER:=0)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_I_3		IUT sends the CPCS−PDU which has the length greater than 48 octets.
4		START T_Opr			
5	LB1	LT_PCO1?ATM_CELL(AUU_R:=ATM_CELL.PTI_AUU, COUNTER:COUNTER+1)	ATM_CELL_ANY		Receving any ATM Cell.
6		[COUNTER<MULTIval−1]			
7		[AUU_R='0'B]			
8		GOTO LB1			
9		[AUU_R<>'0'B]		F	
10		[COUNTER>=MULTIval]			
11		GOTO LB1			
12		LT_PCO1?ATM_CELL	CELL_UNASSIGNED		Lower tester receives an unassigned cell.
13		GOTO LB1			
14		LT_PCO1?OTHERWISE		F	
15		?TIMEOUT T_Opr			
16		[COUNTER=0]		I	
17		[COUNTER=MULTIval]		P	
18		[COUNTER>0,COUNTER<>MULTIval]		F	

Detailed Comments : Selection Ref. : MULTI_PDU

Test Case Dynamic Behaviour

Test Case Name : PROC_SEND_15
Group : AAL_CP_PART_I/PROC/SEND/
Purpose : Verify that the IUT transfers the last segment of a CPCS–PDU, to the ATM layer with AUU parameter in the ATM–DATA.request primitive equal to one.
Configuration :
Default :
Comments : Ref. 6.4.1.2 / PICS 3.4.2.1

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1,COUNTER:=0)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_I_3		IUT sends the CPCS–PDU which has the length greater than 48 octets.
4		START T_Opr			
5	LB1	LT_PCO1?ATM_CELL(AUU_R:=ATM_CELL.PTI_AUU)	ATM_CELL_ANY		
6		[COUNTER<MULTIval]			
7		(COUNTER:=COUNTER+1)			
8		GOTO LB1			
9		[COUNTER=MULTIval]			Receiving the last segment.
10		[AUU_R='1'B]		P	AUU parameter of the last segment of a CPCS–PDU is equal to one.
11		[AUU_R<>'1'B]		F	AUU parameter of the last segment of a CPCS–PDU is not equal to one.
12		LT_PCO1?ATM_CELL	CELL_UNASSIGNED		Lower tester receives an unassigned cell.
13		GOTO LB1			
14		LT_PCO1?OTHERWISE		F	
15		?TIMEOUT T_Opr		I	

Detailed Comments : Selection Ref. : MULTI_PDU

Test Case Dynamic Behaviour

Test Case Name : PROC_SEND_16
Group : AAL_CP_PART_I/PROC/SEND/
Purpose : Verify that the IUT transfers all segments of a CPCS−PDU to the ATM layer with an indication of congestion set to the value of the corresponding CPCS_CI parameter received from the CPAAL5 user.
Configuration :
Default :
Comments : Ref. 6.4.1.2, 6.4.2.2 / PICS 3.4.1.1

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_I_3		IUT send a CPCS−PDU with the Congestion Indication parameter received from the CPAAL5 user.
4		START T_Opr			
5	LB1	LT_PCO1?ATM_CELL(CI_R:=ATM_CELL.PTI_CI)	ATM_CELL_1		ATM cell payload has the last (or complete) ATM−SDU
6		[CI_R=CONG_INval]		P	Congestion Indication (CI) is set to the value received from the CPAAL5 user.
7		[CI_R<>CONG_INval]		F	CI is not set to the value received from the CPAAL5 user.
8		LT_PCO1?ATM_CELL(CI_R:=ATM_CELL.PTI_CI)	ATM_CELL_0		ATM cell payload has not the last (or complete) ATM−SDU
9		[CI_R=CONG_INval]			
10		GOTO LB1			
11		[CI_R<>CONG_INval]		F	
12		LT_PCO1?ATM_CELL	CELL_UNASSIGNED		Lower tester receives an unassigned cell.
13		GOTO LB1			
14		LT_PCO1?OTHERWISE		F	
15		?TIMEOUT T_Opr		I	

Detailed Comments : Selection Ref. : CPCS_CI

Test Case Dynamic Behaviour

Test Case Name : PROC_SEND_17
Group : AAL_CP_PART_I/PROC/SEND/
Purpose : Verify that the IUT transfers all segments of a CPCS–PDU to the ATM layer with an indication of loss priority set to the value of the corresponding CPCS_LP parameter received from the CPAAL5 user.
Configuration :
Default :
Comments : Ref. 6.4.1.2, 6.4.2.2 / PICS 3.4.1.2

Nr	Label	Behaviour Description	Constraints Ref	Verdict	Comments
1		(Setup_Count:=Setup_Count+1)			
2		+PREAMBLE			
3		<IUT!CPCS_PDU>	CPCS_PDU_I_3		IUT send a CPCS–PDU with the Loss Priority parameter received from the CPAAL5 user.
4		START T_Opr			
5	LB1	LT_PCO1?ATM_CELL(CLP_R:=ATM_CELL.CLP)	ATM_CELL_1		ATM cell payload has the last (or complete) ATM–SDU
6		[CLP_R=LOSS_PRval]		P	Cell Loss Priority (CLP) is set to the value received from the CPAAL5 user.
7		[CLP_R<>LOSS_PRval]		F	CLP is not set to the value received from the CPAAL5 user.
8		LT_PCO1?ATM_CELL(CLP_R:=ATM_CELL.CLP)	ATM_CELL_0		ATM cell payload has not the last (or complete) ATM–SDU
9		[CLP_R=LOSS_PRval]			
10		GOTO LB1			
11		[CLP_R<>LOSS_PRval]		F	
12		LT_PCO1?ATM_CELL	CELL_UNASSIGNED		Lower tester receives an unassigned cell.
13		GOTO LB1			
14		LT_PCO1?OTHERWISE		F	
15		?TIMEOUT T_Opr		I	

Detailed Comments : Selection Ref. : CPCS_LP

Test Step Dynamic Behaviour					
Test Step Name : PREAMBLE					
Group : AAL_CP_PART_I/					
Objective : Setup of the connection.					
Default :					
Comments :					
Nr	**Label**	**Behaviour Description**	**Constraints Ref**	**Verdict**	**Comments**
1		[Setup_Count =1]			
2		[SETUP_CON(VPIval,VCIval) = TRUE]			
3		[SETUP_CON(VPIval,VCIval) = FALSE]		I	
4		[Setup_Count > 1]			
Detailed Comments :					