|  |  |
| --- | --- |
|  |  |

SUA Protocol Emulation for TTCN-3 Toolset with TITAN, Function Specification

Contents

[1 Introduction 2](#_Toc156722216)

[1.1 Revision History 2](#_Toc156722217)

[1.2 How to Read this Document 2](#_Toc156722218)

[1.3 Scope 2](#_Toc156722219)

[1.4 References 2](#_Toc156722220)

[1.5 Abbreviations 3](#_Toc156722221)

[1.6 Terminology 3](#_Toc156722222)

[2 General 4](#_Toc156722223)

[3 Function Specification 5](#_Toc156722224)

[3.1 Routing Functionality 5](#_Toc156722225)

[3.2 Service Classes 5](#_Toc156722226)

[3.3 SCCP ASPs 6](#_Toc156722227)

[3.4 SUA Messages 6](#_Toc156722228)

[3.5 SCCP Primitives of the Connectionless Service 9](#_Toc156722229)

[3.6 SCCP Primitives for Connection-oriented Services 10](#_Toc156722230)

[3.7 SUA Management Functionality 10](#_Toc156722231)

[3.8 Message Sequence Control 10](#_Toc156722232)

[3.9 Flow Control 10](#_Toc156722233)

[3.10 Segmentation and Reassembly 11](#_Toc156722234)

[3.11 State Machine 11](#_Toc156722235)

[3.12 Global Title Translation 11](#_Toc156722236)

[3.13 Capacity and Limitation 12](#_Toc156722237)

# Introduction

## Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Rev | Characteristics | Prepared |
| 2006-11-29 | PA1 | First draft version | EJMTCSO |
| 2006-12-15 | PA2 | Updated after review | EJMTCSO |
|  |  |  |  |
|  |  |  |  |

## How to Read this Document

This is the Function Specification for the SUA Protocol Emulation (SUA PE). The SUA PE is developed for the TTCN-3 Toolset with TITAN according to the requirement specification ‎[5]. This document is intended to be read together with Product Revision Information ‎[6].

## Scope

The purpose of this document is to specify the functionality of the SUA Protocol Emulation (PE). Basic knowledge in TTCN-3 is advantage but not required when reading this document (see ‎[1]).

This specification is based on specifications of Signalling Connection Control Part User Adaptation Layer (SUA) defined by IETF ‎[7].

This document only describes the differences and the restrictions related to the above references, but does not repeat their content.

## References

1. ETSI ES 201 873-1 v2.2.1 (02/2003)  
   The Testing and Test Control Notation version 3. Part 1: Core Language
2. 1/198 17-CRL 113 200 Uen  
   User Guide for TITAN TTCN-3 Test Executor
3. 2/198 17-CRL 113 200 Uen  
   Programmer´s Technical Reference for TITAN TTCN-3 Test Executor
4. 1/1531-CRL 113 200 Uen  
   Installation Guide for TITAN TTCN-3 Test Executor
5. 6/0363-FCP 101 3665/P Uen Rev A  
   TTCNv3 Requirement Specification for MSC R13
6. 109 21-CNL 113 517-1 Uen  
   SUA Protocol Emulation for TTCN-3 Toolset, Product Revision Information
7. RFC 3868 - Signalling Connection Control Part User Adaptation Layer (SUA)   
   <http://www.ietf.org/rfc/rfc3868.txt?number=3868>
8. 109 21-CNL 113 516-1 Uen  
   SUA Test Port for TTCN-3 Toolset with TITAN, Product Revision Information

## Abbreviations

ASP Abstract Service Primitive

IETF Internet Engineering Task Force

IUT Implementation Under Test

PDU Protocol Data Unit

PE Protocol Emulation

SCCP Signalling Connection Control Part

SUA SCCP User Adaptation Layer

SUA PE SUA Protocol Emulation

SS7 Signalling System No 7

TTCN-3 Testing and Test Control Notation version 3.

## Terminology

Protocol Emulation: An instance which implements messages and dynamic behaviour of a given protocol layer.

# General

The SUA protocol emulation (PE) implements the SUA protocol specified by IETF ‎[7]. The emulation is performed in TTCN-3 ‎[1] and it requires the TTCN-3 Test Tool with TITAN test executor ‎[2] as environment.

The SUA PE is developed for testing implementations of SCCP Users using TTCN and it uses the services of the underlying SUA layer (see in *Figure 1*). It is assumed that the SUA layer of the peer conforms to the same specifications as SUA PE does.

SCCP-User

Instance 2

SCCP-User

Instance 1

N-Service Primitives

Service access points (SAPs)

SUA PE

Instance 1

SUA PE

Instance 2

SUA-Service Primitives

Service access points (SAPs)

SUA (and lower layers)

Instance 1

SUA (and lower layers)

Instance 2

*Figure 1. Service primitives in SUA PE*

The SUA PE itself is situated between the SUA and the SCCP-User (see *Figure 1*). SCCP user can communicate using SCCP ASP service primitives.

# Function Specification

## Routing Functionality

Routing functionality is not performed. SUA PE emulates a signalling endpoint.

## Service Classes

Service primitives are implemented as messages in the test ports.

There are four service classes in SCCP (see 6/Q.711 and 2/T1.112.1-2001) as follows:

0 Basic connectionless class

1 In-sequence delivery connectionless class

2 Basic connection-oriented class

3 Flow control connection-oriented class.

The SUA PE supports classes 0,1,2 but doesn’t support class 3.

There is no difference between class 0 and 1 because only one test port is being used.

## SCCP ASPs

SUA service primitives are received and sent across service access points can be found in *Table 1*.

“Not implemented” primitives are discarded by SUA PE.

|  |  |
| --- | --- |
| **SCCP ASP** | **Implementation info** |
| N\_CONNECT\_res | received from SUA user |
| N\_CONNECT\_req | received from SUA user |
| N\_DATA\_req | received from SUA user |
| N\_DISCONNECT\_req | received from SUA user |
| N\_UNITDATA\_req | received from SUA user |
| N\_RESET\_req | not implemented |
| N\_RESET\_resp | not implemented |
| N\_CONNECT\_ind | sent to SUA user |
| N\_CONNECT\_cfm | sent to SUA user |
| N\_DATA\_ind | sent to SUA user |
| N\_DISCONNECT\_ind | sent to SUA user |
| N\_UNITDATA\_ind | sent to SUA user |
| N\_NOTICE\_ind | sent to SUA user |
| N\_STATE\_ind | not implemented |
| N\_RESET\_ind | sent to SUA user |
| N\_RESET\_cfm | sent to SUA user |

*Table1 SCCP ASPs handled by SUA PE*

## SUA Messages

User data fields of SUA PDUs received by SUA PE are mapped to N-service primitives that will be sent to the SCCP User(s).

An SUA message received may be mapped to an N-service primitive or may invoke an exception handling procedure based on the state of the SUA PE. The supported message types and the related mappings are summarized in *Table 2*. Though, not supported SUA messages might be supported on lower level by SUA Test Port, refer to ‎[8] for more information on that.

|  |  |  |  |
| --- | --- | --- | --- |
| **SUA message ID** | **Name of message** | **This message is not supported by SUA PE.** | **Message handled and relayed towards SCCP-User layer if needed** |
| SUA\_ERR | Error | **X** |  |
| SUA\_NTFY | Notify | **X** |  |
| SUA\_DUNA | Destination Unavailable | **X** |  |
| SUA\_DAVA | Destination Available | **X** |  |
| SUA\_DAUD | Destination State Audit | **X** |  |
| SUA\_SCON | Signalling Congestion | **X** |  |
| SUA\_DUPU | Destination User Part Unavailable | **X** |  |
| SUA\_DRST | Destination Restricted | **X** |  |
| SUA\_ASPUP | ASP up | **X** |  |
| SUA\_ASPDN | ASP down | **X** |  |
| SUA\_BEAT | Heartbeat | **X** |  |
| SUA\_BEAT\_Ack | Heartbeat Ack | **X** |  |
| SUA\_ASPUP\_Ack | ASP up ack | **X** |  |
| SUA\_ASPDN\_Ack | ASP down ack | **X** |  |
| SUA\_ASPAC | ASP Active | **X** |  |
| SUA\_ASPIA | ASP Inactive | **X** |  |
| SUA\_ASPAC\_Ack | ASP Active Ack | **X** |  |
| SUA\_ASPIA\_Ack | ASP Inactive | **X** |  |
| SUA\_CLDT | Connectionless Data Transfer |  | **X** |
| SUA\_CLDR | Connectionless Data REsponse |  | **X** |
| SUA\_CORE | Connection Request |  | **X** |
| SUA\_COAK | Connection Acknowledge |  | **X** |
| SUA\_COREF | Connection Refused |  | **X** |
| SUA\_RELRE | Release Request |  | **X** |
| SUA\_RELCO | Release Confirm |  | **X** |
| SUA\_RESCO | Reset Confirm |  | **X** |
| SUA\_RESRE | Reset Request |  | **X** |
| SUA\_CODT | Connection Oriented Data Transfer |  | **X** |
| SUA\_CODA | Connection Oriented Data Transfer | **X** |  |
| SUA\_COERR | Connection Oriented Data Transfer |  | **X** |
| SUA\_COIT | Connection Oriented Data Transfer |  | **X** |
| SUA\_REGREQ | Registration Request | **X** |  |
| SUA\_REGRSP | Registration Response | **X** |  |
| SUA\_DEREGREQ | Deregistration Request | **X** |  |
| SUA\_DEREGRSP | Deregistration Response | **X** |  |

*Table 2. Message types implemented by SUA PE*

This table describes the types of messages SUA PE accepts from the lower layer. Details of SUA message translation into SCCP N-ASPs can be found in ‎[7].

## SCCP Primitives of the Connectionless Service

SCCP can provide 2 classes of connectionless services (0 and 1) but there is no difference between them in this implementation (see 6/Q.711 and 2/T1.112.1-2001).

The primitives to the upper layers and the corresponding parameters for connectionless service are implemented as follows:

|  |  |  |
| --- | --- | --- |
| **Generic name** | **ASP name** | **Message mapping** |
| N-UNITDATA | N\_UNITDATA\_req  N\_UNITDATA\_ind | send SUA CLDT  SUA CLDT received |
| N-NOTICE | N\_NOTICE\_ind | SUA CLDR received |

*Table 3. Primitives and their Mappings for Connectionless Service*

## SCCP Primitives for Connection-oriented Services

|  |  |  |
| --- | --- | --- |
| **Generic Name** | **ASP name** | **Message mapping** |
| N-CONNECT | N\_CONNECT\_req  N\_CONNECT\_ind  N-CONNECT\_res  N-CONNECT\_cfm | send SUA CORE  SUA CORE received  send SUA COAK  SUA COAK received |
| N-DATA | N\_DATA\_req  N\_DATA\_ind | send SUA CODT  SUA CODT received |
| N-DISCONNECT | N\_DISCONNECT\_req  N\_DISCONNECT\_ind | send SUA COREF  SUA COREF received |
| N-RESET | N\_RESET\_ind  N\_RESET\_cfm | SUA RESRE received  SUA RESRE received |

*Table 4. Network service primitives and mapping for connection-oriented services*

## SUA Management Functionality

SUA management functionality is implemented in the SUA Test Port ‎[8]. However, support for connection-oriented transfer is enabled by SUA PE by handling SUA CORE, COAK, COREF, RELRE, RELCO, RESCO, RESRE, COERR and COIT messages.

## Message Sequence Control

SUA PE maintains the order of messages between the upper and lower layer interfaces.

## Flow Control

Not supported by SUA PE.

## Segmentation and Reassembly

Segmentation is supported in as much as data received from the SCCP User layer can be transferred in SUA CODT messages using segmentation. Maximum allowed transfer message size can be altered using the module parameter tsp\_SUA\_data\_maxlen, which is taken into account when transferring N-DATA\_req messages. Also, incoming CODT messages are checked for segmentation and a receiving buffer is used if needed.

## State Machine

SUA PE maintains a state machine behaviour for each connection-oriented services according to Figure 8/Q.711.

## Global Title Translation

Not supported.

## Capacity and Limitation

Few parameters are hardcoded in SUA PE implementation, here is the list of them:

| **Feature** | **Restriction** | **Remark** |
| --- | --- | --- |
| Fields <translation type>, <numbering plan> and <nature of address> | = 0 (unknown), if not specified otherwise | See 3.10.2.3. in ‎[7] |
| Management | Partially implemented | See 3.7. in this document |
| Field <encoding scheme> in a N-SCCP ASP Global Title | = 1 or 2 | 1 if (<number of digits> mod 2 == 1), 2 otherwise |
| Service class 3 | NOT IMPLEMENTED |  |
| Routing | NOT IMPLEMENTED | SUA PE is a signalling endpoint |
| Message sequence control | NOT IMPLEMENTED | Indifferent |
| Flow control | NOT IMPLEMENTED | Because class 3 not supported |
| Global title translation | NOT IMPLEMENTED |  |

*Table 5. Features with restriction in SUA PE*

There shall be exactly one SCCP User test component instance for each SUA PE instance. An SUA PE instance is able to handle up to 16 SUA connections and 16 segmented N-UNITDATA messages at the same time.