

UDP Socket Test Port for TTCN-3 Toolset with TITAN, User's Guide

Gábor Szalai

Version 198 17-CNL 113 346, Rev. E, 2007-01-24

Table of Contents

About This Document	1
How to Read This Document	1
Prerequisite Knowledge	1
System Requirements	1
Fundamental Concepts	1
The Test Port	1
Overview	1
Installation	2
Operation Modes	2
Basic Mode	2
Advanced Mode	2
Configuration	3
UDP Test Port Parameters in the RTE Configuration File	3
Start Procedure	5
TTCN-3 Test Executor	5
Sending UDP ASPs	5
Basic Mode	5
Advanced Mode	5
Receiving UDP ASPs	6
Basic Mode	6
Advanced Mode	6
Stop Procedure	7
TTCN-3 Test Executor	7
Examples	7
Configuration File	7
Error Messages	7
Warning Messages	8
Terminology	8
Abbreviations	8
References	9

About This Document

How to Read This Document

This is the User's Guide for the UDP test port. The UDP test port is developed for the TTCN-3 Toolset with TITAN according to the Requirement Specification [4]. This document is intended to be read together with Function Specification [3].

Prerequisite Knowledge

The knowledge of the TITAN TTCN-3 Test Executor [2] and the TTCN-3 language [1] is essential. Basic knowledge of the UDP protocol is valuable when reading this document.

System Requirements

In order to operate the UDP test port the following system requirements must be satisfied:

- Platform: Solaris 5.8
- TITAN TTCN-3 Test Executor version R7A (1.7.pl0) or higher installed. For installation guide see [2].

NOTE

This version of the test port is not compatible with TITAN releases earlier than R7A. The same C compiler gcc version installed which the TITAN was compiled with. See the Release Notes of TITAN.

For other known limitations please see the FS [3].

Fundamental Concepts

The test port establishes UDP connection between the TTCN-3 test executor and the SUT. The test port transmits and receives UDP protocol messages between the TITAN RTE and the SUT.

The Test Port

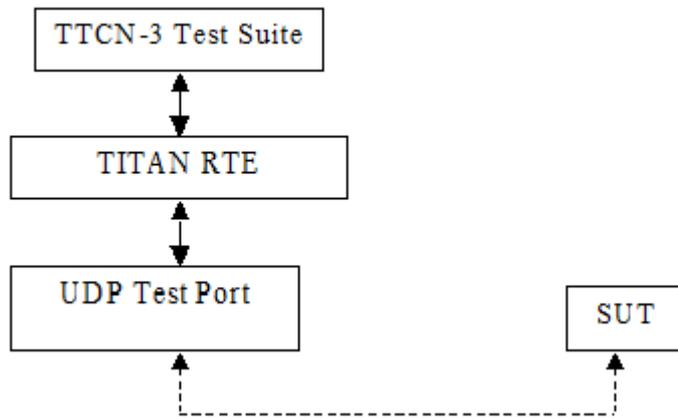
Overview

The UDP test port offers UDP primitives to the test suite in TTCN-3 format. The TTCN-3 definition of the ASPs can be found in a separate TTCN-3 module. This module must be imported into the test suite.

The test port transfers UDP primitives towards the TITAN RTE and UDP datagrams towards the SUT.

See the overview of the test system below:

Figure 1.: System Overview



Installation

Since the UDP test port is used as a part of the TTCN-3 test environment this requires TTCN-3 Test Executor to be installed before any operation of the UDP test port. For more details on the installation of TTCN-3 Test Executor see the relevant section of [\[2\]](#).

When building the executable test suite the libraries provided for the TTCN-3 Test Executor should also be linked into the executable. For more information see also [\[2\]](#).

Operation Modes

Basic Mode

Description

In basic mode the port provides a simple ASP interface and can handle only one UDP socket. The local IP address and port number can be configured only via config file. This mode is compatible with the previous release of the port. Only the basic ASP can be used.

ASP Definition

`ASP_UDP` – send and receive data

Advanced Mode

Description

In advanced mode the test port provides the following functionality:

- open socket
- close socket
- send and receive data

The port can handle several UDP sockets. The local IP address and port numbers are configured in run time via ASPs. Only the advanced ASP can be used.

ASP Definition

`ASP_UDP_message` – send and receive data

`ASP_UDP_open` – request a new socket

`ASP_UDP_open_result` – the result of the open request

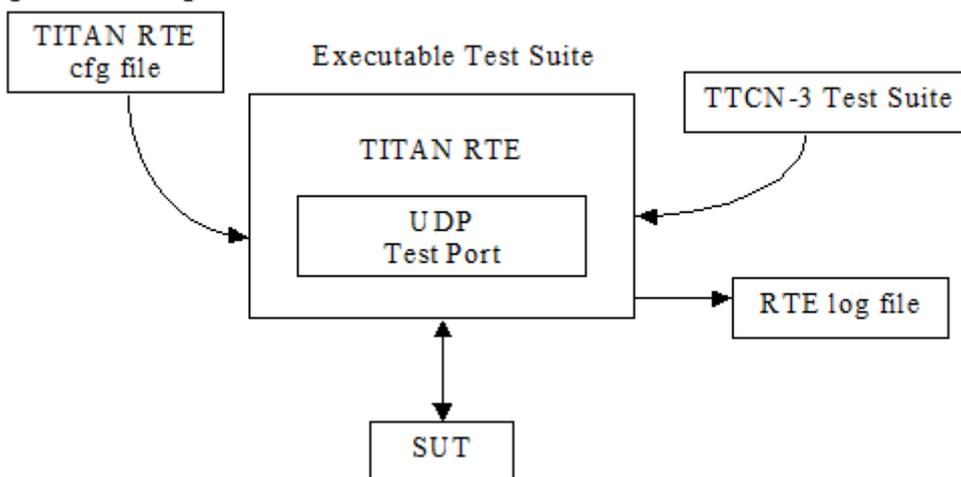
`ASP_UDP_close` – close the socket

Configuration

The executable test program behavior is determined via the RTE configuration file. This is a simple text file, which contains various sections (e.g. `[TESTPORT_PARAMETERS]`) after each other. The usual suffix of the RTE configuration file is `.cfg`. For further information about the configuration file see [\[2\]](#).

See the overview of the configuration process below:

Figure 2: Configuration Overview



UDP Test Port Parameters in the RTE Configuration File

In the `[TESTPORT_PARAMETERS]` section you can specify parameters that are passed to the test ports. Each parameter definition consists of a component name, a port name, a parameter name and a parameter value. The component name can be either an identifier or a component reference (integer) value. The port and parameter names are identifiers while the parameter value always must be a charstring (with quotation marks). Instead of component name or port name (or both of them) the asterisk ("`*`") sign can be used, which means "all components" or "all ports of the component". More information about the RTE configuration file can be found in [\[2\]](#).

In the `[TESTPORT_PARAMETERS]` section the following parameters can be set for the UDP test port:

- **mode**

This sets the port operation mode.

Possible values:

- "basic" - Basic operation mode
- "advanced" - Advanced operation mode

The default mode is the "basic" operation mode.

- **localIPAddr**

This identifies the IP address of the local host. Only IPv4 is supported.

Possible values of this parameter are:

IP address in IPv4 format or the host name like: "127.0.0.1" or "localhost".

The default is "all network".

Used only in basic mode.

- **localPort**

This gives the local port number where the test port receives datagrams.

The default is "50000".

Used only in basic mode.

- **debugging**

Activates/deactivates logging in the test port.

Possible values are:

- "no" - Logging is not active
- "yes" - Logging is active

The default value is "no".

In the RTE configuration file it is possible to define the name and path of the RTE log file. Debug output from the test port is written into the RTE log file. If no path is given in the configuration file, the RTE log file is stored in the working directory.

- **broadcast**

Enables/disables the broadcast sending capabilities of the test port.

- "enabled" Broadcast sending is enabled
- "disable" Broadcast sending is disabled

The default value is "disabled".

Start Procedure

TTCN-3 Test Executor

Before the executable test suite can be run the TTCN-3 modules and C++ codes should be compiled and linked into an executable program. This process can be automated using the `make` utility. The way how the *Makefile* should be generated is described in [2].

NOTE

The C++ implementation files *UDPasp_PT.hh* and *UDPasp_PT.cc* of the test port must be included in the *Makefile*.

If the executable test suite is ready, run it giving the RTE configuration file as argument in your terminal:

```
Home> ExecutabletestSuite RTEConfigurationFile.cfg
```

For more information, see [2].

Sending UDP ASPs

Basic Mode

At `send` operation the `addressf` and `portf` must contain the remote host IP address and port number. The IP address can be given either as the remote hostname (ex: `localhost`) or in IPv4 format (ex: `127.0.0.1`).

The `data` field should contain the datagram to be sent.

Advanced Mode

Open a New Socket

The opening of a new socket is requested by the sending of the `ASP_UDP_open` message to the test port. The `local_addr` contains the local IP address or hostname. If it is omitted the default is any address. The `local_port` contains the local port number. If omitted a random port number will be used.

The `remote_addr` and `remote_port` contain the address and port number of the remote host. If specified this address will be the default remote address for the socket.

The test port answers the `open` request with the `ASP_UDP_open_result`.

Sending Data

The data sending is requested with the `ASP_UDP_message` message.

The `remote_addr` and `remote_port` can contain the address and port number of the remote host. If specified this address will be used and overwrite the default remote address for the socket. If omitted the default remote address will be used. They are mandatory if no socket id specified.

The `id` specifies the socket used during data sending. If omitted the port will open a socket for the data sending. This socket will be closed after the send is completed.

The `data` field should contain the datagram to be sent.

Close the Socket

The `close` operation is requested by the `ASP_UDP_close` message.

The `id` identifies the socket to be closed.

Receiving UDP ASPs

Basic Mode

At `receive` operation the `addressf` and `portf` contain the remote host IP address and port number. The IP address is given in IPv4 format (ex: 127.0.0.1).

The `data` field contains the received datagram.

Advanced Mode

Open Result

The `open` request is answered with `ASP_UDP_open_result` message. It carries the details of the opened socket.

The `local_addr` contains the local IP address. The `local_port` contains the port number.

The `id` is the unique identifier of the socket.

Receiving Data

The data receiving is indicated via `ASP_UDP_message` message.

At receive operation the `remote_addr` and `remote_port` contain the remote host IP address and port number. The IP address is given in IPv4 format (ex: 127.0.0.1).

The `id` contains the unique id of the socket that received the data.

The `data` field contains the received datagram.

Stop Procedure

TTCN-3 Test Executor

The test port should stop automatically after it finished the execution of all test cases. It closes down the open UDP sockets towards the SUT and terminates.

The execution of the test suite can be stopped at any time by pressing `<Ctrl>-c`. It will shut down the socket and terminate.

Examples

Configuration File

An example RTE configuration file is shown below:

```
[LOGGING]
LogFile := "UDPtest.log"
FileMask := LOG_ALL | TTCN_DEBUG | TTCN_MATCHING
ConsoleMask := TTCN_ERROR | TTCN_WARNING | TTCN_TESTCASE | TTCN_STATISTICS |
TTCN_PORTEVENT
LogSourceInfo := Yes

[EXECUTE]
UDPtest.tc1

[TESTPORT_PARAMETERS]
*.UDP1.debugging := "yes"
*.UDP1.localPAddr := "localhost"
*.UDP1.localPort := "5679"
*.UDP2.debugging := "yes"
*.UDP2.localPAddr := "localhost"
*.UDP2.localPort := "5678"
```

Error Messages

The error messages have the following general form:

```
Dynamic test case error: <error text>
```

Error messages are written into the log file. In the log file a time stamp is also given before the message text.

The list of the possible error messages is shown below. Note that this list contains the error messages produced by the test port. The error messages coming from the TITAN are not shown:

<Parameter Name> is not defined in the configuration file

Mandatory parameter is missing from the configuration file.

Error when reading the received UDP PDU.

An error occurred when a UDP PDU was received.

Cannot open socket

The socket system call fails.

Cannot bind port

The bind system call fails.

Sendto system call failed: <n> bytes was sent instead of <m>

The `sendTo` system call fails.

The host name <name> is not valid.

The specified host name is not a valid host name.

No remote host name specified.

The remote host name is not specified and there is no default remote host to send the data.

No remote port specified.

The remote port is not specified and there is no default port to send the data.

Warning Messages

The following list shows the possible warning messages produced by the test port:

UDPasp__PT::set_parameter() : Unsupported Test Port parameter <parameter>.

The parameter set in the configuration file is unknown for the UDP test port.

Terminology

None.

Abbreviations

ASP

Abstract Service Primitive

RTE

Run-Time Environment

SUT

System Under Test

TTCN-3

Testing and Test Control Notation version 3

UDP

User Datagram Protocol

References

[1] ETSI ES 201 873-1 (2002)

The Testing and Test Control Notation version 3. Part 1: Core Language

[2] User Documentation for the TITAN TTCN-3 Test Executor

[3] UDP Socket Test Port for TTCN-3 Toolset with TITAN, Function Specification

[4] Functional Test System Requirement Specification for GSN

[5] [RFC 768](#) (1980)

User Datagram Protocol