

STAR-Dundee

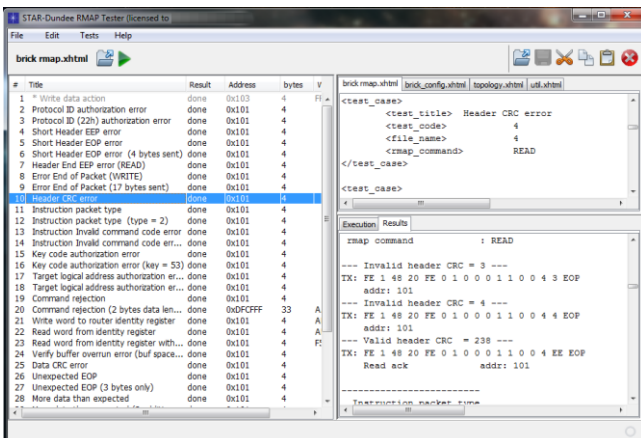
Supporting SpaceWire Applications

SpaceWire RMAP Tester

The SpaceWire RMAP Tester is a software test platform that provides the means to validate SpaceWire RMAP target implementations and to test the generic functionality of any unit that relies on the RMAP protocol for its configuration and control. The SpaceWire RMAP Tester works in conjunction with a STAR-Dundee Interface device for example SpaceWire-USB Brick.

An intuitive Graphical User Interface (GUI) allows the user to select from a list of predefined RMAP test descriptions to quickly verify that the unit under test complies with the RMAP standard. Any result that deviates from the standard is immediately highlighted and reported in detail.

The GUI is supplied with an integrated text editor and powerful test description language for the definition of customised tests. RMAP scripting is used to realise complex use cases; the exhaustive validation of a routing table can be achieved in a single run, and the boundary checking of Read-Modify-Write operations verified when defined between specified valid address ranges and with maximum data lengths.



Key Features

- Automatic RMAP conformance testing with automatic coding/decoding of RMAP fields.
- Supports the complete set of SpaceWire RMAP functions without restriction.
- Can work with non-nominal SpaceWire RMAP commands to trigger error conditions.
- The Graphical User Interface includes; a text based test editor, a log of executed tests, a progress window and a detailed test report linked to the test selected within the log.
- Single tests can operate over a range of memory addresses.
- Basic sets of tests can be defined.
- Advanced sets of tests are defined using a scripting language that supports the inclusion of files, macros, loops, variables, conditions and functions such as bitwise operations.
- Data can be defined using bytes or words and can be obtained from files.
- XML tags and test descriptions can be linked to user defined macros.

Applications

The RMAP Tester serves two main use cases:

- RMAP Conformance Tester: the supplied comprehensive test set validates an RMAP target implementation against the RMAP standard.
- RMAP Acceptance Tester: validate the functionality of a unit under test accessible using RMAP commands. For example, the functional test of the routing table of the SpW-10X router.

Example of RMAP Conformance Tester

The SpaceWire RMAP Tester can automatically check that an RMAP capable unit follows the SpaceWire RMAP standard assuring complete interoperability with other compliant RMAP devices.

The following test description example for the SpaceWire RMAP Tester executes all RMAP validation tests on a unit under test via a STAR-Dundee SpaceWire interface device. It applies the parameters specified to all the tests defined in the "RMAP-all-tests" file. A full set of tests covering the RMAP protocol are provided with the product. Suitable default values are used when parameters are not specified.

```
---- All RMAP conformance tests ----

<parameters>

<destination_path> 0 </destination_path>
<return_path> </return_path>
<dest_logical_address> 254 </dest_logical_address>
<destination_key> 0x20 </destination_key>
<start_address> 257 </start_address>
<increment_address> no </increment_address>
<acknowledge> yes </acknowledge>
<data_length> 4 </data_length>

</parameters>

<include_file> RMAP_all_tests </include_file>
```

The SpaceWire RMAP Tester also allows the user to selectively specify the value of some parameters that are only applied to the current test being defined. The following test description example executes a single test with specific parameters. Note that the RMAP Tester parser can automatically close an XML tag to improve the readability of the tests descriptions.

```
---- Single RMAP conformance test ----

<test>

<test_title> Key authorization error
<test code> 7
<invalid_destination_key> 254

</test>
```

A single test can send one or multiple RMAP commands with the RMAP Tester checking each associated RMAP response. In the example above, if the invalid destination key is not provided and the parameter "exhaustive" is enabled, the RMAP Tester will automatically try all possible destination keys and check that all commands with an invalid key produce a response with the appropriate invalid key error code.

Example of RMAP Acceptance Tester

The definition of more advanced tests can be used to validate the functionality of a unit that uses RMAP to configure its capabilities. A test definition is usually a nominal Read, Write or Read-Modify-Write RMAP operation with user-dependent memory address and data; the unit is configured using RMAP Write commands and the status of the device is evaluated by checking the data of an RMAP Read reply.

The following advanced test description shows the use of variables, macros, arrays, loops and bitwise operations. It validates the first three entries of the routing table of a STAR-Dundee SpaceWire-USB Brick which contains two ports.

```

---- Validation of routing table functionality ----
<var> logicalAddress = 32
<loop> 3
  <var> port = 1
  <loop> 2

    <test> <!-- Set the router table entry -->
    <rmap_command> write
    <start_address> logicalAddress
    <data> bit(DEL_HEAD_BIT ) | bit(port)
    <data_word_mask> MASK_ROUTING_ENTRY

    <test> <!-- Sends a probe packet -->
    <rmap_command> read
    <start_address> IDENTITY_REG
    <destination_path> [logicalAddress, 0]
    <return_path> PORT_USB
    <data> WORD_PATTERN

    <var> port = port + 1
  <end_loop>
  <var> logicalAddress = logicalAddress + 1
<end_loop>

```

Product Capabilities

Automatic validation of RMAP operations

For any test executed the system checks that:

- The RMAP replies are correct and do not have any invalid RMAP field values.
- The RMAP replies have the expected error codes when an RMAP command with an invalid RMAP field or with parameters not supported by the unit is received.
- The unit under test executes the RMAP commands as expected. For example, a command should not be executed in some error cases.
- The unit correctly updates the corresponding status register when an RMAP error occurs.

For Read commands, the tester can provide a test parameter that contains the data expected following a read command. The test data can be automatically written before the read command executes.

For Write commands the tester can automatically read back the written address to verify the written data and the write command has executed correctly.

For Read-Modify-Write commands, the result from the Read-Modify-Write operation can be automatically verified against the data read. This allows the generation of a test case that sends a Read-Modify-Write command with an invalid data CRC, and to check that a reply with an error code is received and the operation has not been executed.

For a non-verified Write command with an invalid data CRC, the RMAP Tester will report that the test is passed if the operation has been executed.

Graphical User interface

The GUI provides the tester with the following functionality:

- Select the main file that defines the tests sequence.
- Edit any text file using integrated basic editor capabilities
- Automatically load changes when using an external editor concurrently.
- Run the main tests definition file, show the current execution status, and provide detailed results for each test.
- Show the list of tests executed, indicating if a test has failed or a test definition was invalid.
- On selection of a test from the list of executed tests:
 - The text editor will point to the test definition.
 - The window below will show the corresponding detailed test results or the test execution process if either the test definition was invalid or an execution error occurred.

RMAP predefined test cases

In addition to the nominal Read, Write and Read-Modify-Write RMAP operations, the RMAP Tester provides the following tests that use one or multiple RMAP commands, evaluating the responses automatically:

- Invalid protocol ID
- Incomplete header
- Error end of packet
- Header CRC error
- Unused packet type
- Invalid command code
- Invalid destination key
- Invalid logical address
- Command rejection
- Buffer space exceeded
- Data CRC error
- Unexpected EOP
- More data than expected
- Data characters in a read command
- Invalid data length field
- Return path address

Specifications

The RMAP Tester operates under Windows (Windows 10, 8, 7, Vista, XP), and with a range of STAR-Dundee SpaceWire Interface devices including:

- USB Brick and Router-USB
- USB Brick Mk2, USB Brick Mk3, Router Mk2S, PCIe, PCI Mk2, cPCI Mk2, PXI

Support for Linux (3.x and 2.6 kernels) will be added in the future, check www.star-dundee.com for latest information.