Fast Ethernet Consortium

100BASE-X MII PCS Test Suite
Version 1.0

Technical Document

Last Updated: May 16, 2014

Fast Ethernet Consortium

University of New Hampshire
InterOperability Laboratory

121 Technology Drive, Suite 2
Durham, NH 03824
Phone: (603) 862-1529
Fax: (603) 862-0898

http://www.iol.unh.edu/consortiums/fe

MODIFICATIONS

- February 28, 2014, Version 1.0 Initial Release
  - Based largely on UNH-IOL 100BASE-X PCS Test Suite Version 3.4

ACKNOWLEDGMENTS

The University of New Hampshire would like to acknowledge the efforts of the following individuals in the development of this test suite.

Adam Healey	University of New Hampshire
Stephen Kelsey	University of New Hampshire
Matthew Kilpeck	University of New Hampshire
Robert Noseworthy	University of New Hampshire
Peter Scruton	University of New Hampshire

INTRODUCTION

Overview

The University of New Hampshire's InterOperability Laboratory (IOL) is an institution designed to improve the interoperability of standards based products by providing an environment where a product can be tested against other implementations of a standard. This suite of tests is designed to determine if a product conforms to some of the specifications defined in Clause 24 of the IEEE 802.3-2012 Standard. Successful completion of all tests contained in this suite does not guarantee that the tested device will operate with other devices. However, combined with satisfactory operation in the IOL's interoperability test bed, these tests provide a reasonable level of confidence that the Device Under Test (DUT) will function properly in many Fast Ethernet environments.

Organization of Tests

The tests contained in this document are organized to simplify the identification of information related to a test and to facilitate in the actual testing process. Each test contains an identification section that describes the test and provides cross-reference information. The discussion section covers background information and specifies why the test is to be performed. Tests are grouped by similar functions and further organized by technology. Each test contains the following information:

Test Number

The Test Number associated with each test follows a simple grouping structure. Listed first is the Test Group Number followed by the test's number within the group. This allows for the addition of future tests to the appropriate groups of the test suite without requiring the renumbering of the subsequent tests.

Purpose

The purpose is a brief statement outlining what the test attempts to achieve. The test is written at the functional level.

References

The references section lists cross-references to the IEEE 802.3 standards and other documentation that might be helpful in understanding and evaluating the test and results.

Resource Requirements

The requirements section specifies the hardware, and test equipment that will be needed to perform the test. The items contained in this section are special test devices or other facilities, which may not be available on all devices.

Last Modification

This specifies the date of the last modification to this test.

Discussion

The discussion covers the assumptions made in the design or implementation of the test as well as known limitations. Other items specific to the test are covered here.

Test Setup
The setup section describes the configuration of the test environment. Small changes in the configuration should be included in the test procedure.

Procedure
The procedure section of the test description contains the step-by-step instructions for carrying out the test. It provides a cookbook approach to testing, and may be interspersed with observable results.

Observable Results
The observable results section lists specific items that can be examined by the tester to verify that the DUT is operating properly. When multiple values are possible for an observable result, this section provides a short discussion on how to interpret them. The determination of a pass or fail for a certain test is often based on the successful (or unsuccessful) detection of a certain observable result.

Possible Problems
This section contains a description of known issues with the test procedure, which may affect test results in certain situations.

TABLE OF CONTENTS

Test #24.1.1 - End of Stream Delimiter Test

Purpose: To verify that RX_DV and RX_ER are appropriately asserted and de-asserted during and immediately after different abnormal stream termination events.

References:

- IEEE 802.3 Standard, 2012 - sections 22.2.1.5, 24.2.4.4.4
  - Figure 24-11: Receive state diagram.

Resource Requirements:

- A testing station capable of transmitting arbitrary five-bit code groups as specified in clause 24 and sending these code groups using the signaling method described in clause 25 or clause 26.
- The capability to monitor the MII signaling of the DUT.

Last Modification: May 16, 2014

Discussion: Following detection of the SSD, the signal RX_DV is asserted. The RX_ER signal is asserted upon decoding any symbol following the SSD, which is not either, a valid data symbol or a defined stream termination sequence. Simultaneous assertion of RX_DV and RX_ER should cause the Reconciliation sublayer to force the MAC to detect a FrameCheckError. Refer to subclause 22.2.1.5 and Figure 24-11: Receive state diagram. The DUT is sent a valid packet with the ESD (/T/R/) removed. The DUT is also sent packets with an invalid ESD placed at the end of the packet. These two circumstances should cause RX_ER to occur while RX_DV is asserted. The other case involves testing where a valid ESD terminates the packet and is followed by one additional non-idle code group before idle resumes. This case is tested for each of the 31 non-idle code groups. Following the valid ESD, RX_DV should be de-asserted. Twenty-two of these final 5-bit code groups transmitted immediately after the ESD and before idle should cause RX_ER to be asserted, while nine other 5-bit code groups should not cause this to occur. The nine 5-bit code groups that should not cause RX_ER to be asserted are: 00111, 01111, 10011, 10111, 11001, 11011, 11100, 11101, and 11110.

Test Setup: Connect the MDI of the device under test (DUT) to the testing station (transmit to receive, receive to transmit) with the appropriate medium (i.e. balanced copper, multi-mode fiber, etc.). Appropriately attach a device capable of monitoring the MII signaling to the DUT.

Procedure:

Test Packet Definitions:

- **Part A Test Packet Group:** The test packet is comprised of an SSD, remaining preamble, SFD, a valid test frame with a proper 32-bit CRC value in the FCS field, but no ESD (/T/R/). (Simply transitioning to idle (/I/) following the FCS field.)

- **Part B Test Packet Group:** The test packets are comprised of an SSD, a valid test packet with proper checksums and 32-bit CRC values, a valid ESD (/T/R/) and an additional code group immediately following the packet. This is repeated to include each non-idle code group for a total of 31 packets.

- **Part C Test Packet Group:** The test packets are comprised of an SSD, remaining preamble, SFD, a valid test frame with a proper 32-bit CRC value in the FCS field, and an invalid ESD. The 62 different invalid ESDs to send are a /T/ followed by all code groups, except /R/, (creating 31 different pairings) and all code groups, except /T/, followed by an /R/ (creating the remaining 31 pairings). Each packet with these invalid ESDs should cause RX_ER to be asserted while RX_DV remains asserted.

1. The testing station is instructed to transmit a valid packet to the DUT.
2. After approximately a minimum interPacketGap the testing station is instructed to transmit the first frame in **Part A Test Packet Group** to the DUT.
3. After approximately a minimum interPacketGap the testing station then transmits another valid packet.
4. Steps 2 and 3 are repeated for each defined test packet in the Test Packet Group.[*1]
5. The MII signaling of the DUT is observed.
6. The observed results for steps 1 – 5 using **Part A Test Packet Group** should match observable result a. Steps 1 - 5 are repeated for all packets in **Part B Test Packet Group** (observable result b). Steps 1 – 5 are again repeated for all packets in **Part C Test Packet Group** (observable result c).

NOTE 1: This step is not applicable to **Part A Test Packet Group**, as there is only one test packet in this group.

.

Observable Results:

a. RX_ER should be asserted while RX_DV should remain asserted for the first idle code group upon reception of rx_bits[9:0] = IDLES. The following idle code groups should not flag RX_ER and RX_DV.
b. Following the ESD, RX_DV should be not asserted. RX_ER should be asserted while RX_DV should remain not asserted upon the reception of any of the test packets containing one of the 22 code groups, as specified in the discussion. RXD[3:0] should also be set to 1110 while RX_ER is asserted. RX_DV and RX_ER should remain not asserted for the reception of the other 9 test packets containing each of the other 9 code groups (00111, 01111, 10011, 10111, 11001, 11011, 11100, 11101, and 11110) following the ESD.
c. RX_ER should be asserted while RX_DV remains asserted for each of the invalid ESD test packets. RX_DV and RX_ER should be simultaneously asserted for two code groups worth of time if the first code group of invalid ESD is a data code group and three otherwise.

Possible Problems: None.

Test #24.1.2 - Invalid Data Symbol Test

Purpose: To verify that an error (RX_ER) is detected when an invalid data symbol is sent following the transmission of the SSD (/J/K/)

References:

- IEEE 802.3 Standard, 2012 - sections 4.2.4.1.2, 4.2.4.1.3, 24.2.4.4.3, 24.2.2.1.7 and 22.2.1.5

Resource Requirements:

- A testing station capable of transmitting arbitrary five-bit code groups as specified in clause 24 and sending these code groups using the signaling method described in clause 25 or clause 26.
- The capability to monitor the MII signaling of the DUT.

Last Modification: May 16, 2014

Discussion: Following detection of the SSD, the signal RX_DV is asserted. The RX_ER signal is asserted upon decoding any symbol following the SSD which is not either a valid data symbol or a defined stream termination sequence. Simultaneous assertion of RX_DV and RX_ER will cause the Reconciliation sublayer to force the MAC to detect a FrameCheckError. Refer to subclause 22.2.1.5 and Figure 24-11: Receive state diagram. In this test, all valid data symbols will be replaced with all combinations of the invalid symbols. This is done to ensure that when an invalid symbol is detected, RX_ER is asserted rather than arbitrarily replacing the invalid symbols with valid data symbols.

Test Setup: Connect the MDI of the device under test (DUT) to the testing station (transmit to receive, receive to transmit) with the appropriate medium (i.e. balanced copper, multi-mode fiber, etc.). Appropriately attach a device capable of monitoring the MII signaling to the DUT.

Procedure:

Test Packet Description:

In this test, the testing station transmits a valid 64-byte test packet with a data field containing all valid data symbols (0 thru F). Each data symbol is individually replaced with each of the 5-bit non-data code groups: 00000 (/P/), 00001, 00010, 00011, 00100 (/H/), 00101, 00110, 00111 (/R/), 01000, 01100, 01101 (/T/), 10000, 10001 (/K/), 11000 (/J/), 11001, and 11111 (/I/). Thus, 256 different invalid packets are tested.

1. The testing station is instructed to transmit a valid packet to the DUT.
2. After approximately a minimum interPacketGap the testing station is instructed to transmit a test packet to the DUT.
3. After approximately a minimum interPacketGap the testing station then transmits another valid packet.
4. Steps 2 and 3 are repeated for each of the remaining defined test packets in the Test Packet Description.
5. The MII signaling of the DUT is observed.

Observable Results:

a. The DUT should assert RX_ER corresponding to when the non-data code group is in position rx_bits[9:5] in the test packet, while RX_DV remains asserted.
b. The reception of either following valid packets should not cause the DUT to assert RX_ER. RX_DV should be appropriately asserted during the reception and RXD[3:0] should also be set appropriately.

Possible Problems: None.

Test #24.1.3 - False Carrier Detect

Purpose: To verify that the device under test can detect false carrier events.

References:

- IEEE 802.3 Standard, 2012 – sections 22.2.2.7, 22.2.2.8, 22.2.2.10, Table 22-2, sections 24.2.2.1.4, 24.2.4.4.2, 24.3.4.3, and figure 24-14.

Resource Requirements:

- A testing station capable of transmitting arbitrary five-bit code groups as specified in clause 24 and sending these code groups using the signaling method described in clause 25 or clause 26.
- The capability to monitor the MII signaling of the DUT.

Last Modification: May 16, 2014

Discussion: After channel activity is detected, the Receive process first aligns the incoming code-bits on code-group boundaries for subsequent data decoding. This is achieved by scanning the rx_bits vector for a SSD (/J/K/). Detection of the SSD causes the Receive process to enter the START OF STREAM J state.

Well-formed streams contain SSD (/J/K/) in place of the first 8 preamble bits. In the event that something else is sensed immediately following the detection of carrier, a False Carrier Indication is signaled to the MII by asserting the RX_ER and setting RXD to 1110 while RX_DV remains deasserted.

Test Setup: Connect the MDI of the device under test (DUT) to the testing station (transmit to receive, receive to transmit) with the appropriate medium (i.e. balanced copper, multi-mode fiber, etc.). Appropriately attach a device capable of monitoring the MII signaling to the DUT.

Procedure:

1. Let bad_ssd be a vector of 10 code-bits and let bad_ssd[0] be fixed at ZERO. Initialize bad_ssd[9:1] to the code-bit pattern “111111101”. Command the testing station to transmit a valid packet. Approximately one minimum interPacketGap later the testing station should send bad_ssd (most significant bit first) followed by the remainder of a valid test packet (excluding the SSD). Another minimum interPacketGap later another valid packet should be transmitted.
2. Shift bad_ssd[9:1] left one code-bit, discarding the carry bit and setting bad_ssd[1] to ONE. Approximately one minimum interPacketGap after the previous transmission the testing station should send bad_ssd followed by the remainder of a valid test packet (excluding the SSD). A minimum interPacketGap later transmit another valid packet.
3. Repeat step 2 until bad_ssd[9:2] contains the pattern “01111 111”, which is the last one sent.
4. Set bad_ssd[9:5] to the /J/ code group and set bad_ssd[4:0] to the code-bit pattern “00000”. Command the testing station to send bad_ssd followed by the remainder of

a valid packet (excluding the SSD). A minimum interPacketGap later transmit another valid packet.

5. Increment bad_ssd[4:0]. Approximately one minimum interPacketGap after the previous transmission the testing station should send bad_ssd followed by the remainder of a valid packet (excluding the SSD). A minimum interPacketGap later transmit another valid packet.
6. Repeat step 5 until bad_ssd[4:0] exceeds “11111”. Skip the iteration in which bad_ssd[4:0] equals “10001” as this is the /K/ code-group (this makes bad_ssd[9:0] /J/K/, the valid start of stream delimiter).

Observable Results:

a. The DUT should assert RX_ER upon the reception of an invalid SSD. RXD[3:0] should be set to 1110. RX_DV should remain de-asserted.
b. The reception of following valid packets should not cause the DUT to assert RX_ER. RX_DV should be appropriately asserted during the reception and RXD[3:0] should also be set appropriately

Possible Problems: None.

ANNEX A (informative) Table of Definitions

(informative)

Table of Acronym Definitions

8802-3	ISO/IEC 8802-3 (IEEE Std 802.3)
ANSI	American National Standards Institute
ASIC	application-specific integrated circuit
ASN.1	abstract syntax notation one as defined in ISO/IEC 8824: 1990
MDI, AUI	attachment unit interface
BER	bit error ratio
BPSK	binary phase shift keying
BR	bit rate
BT	bit time
CAT3	Category 3 balanced cable
CAT4	Category 4 balanced cable
CAT5	Category 5 balanced cable
CD0	clocked data zero
CD1	clocked data one
CMIP	common management information protocol as defined in ISO/IEC 9596-1: 1991
CMIS	common management information service as defined in ISO/IEC 9595: 1991
CMOS	complimentary metal oxide semiconductor
CRC	cyclic redundancy check
CRV	code rule violation
CS0	control signal zero
CS1	control signal one
CVH	clocked violation high
CVL	clocked violation low
CW	continuous wave
DA	Destination Address
DTE	data terminal equipment
DUT	Device Under Test
EIA	Electronic Industries Association.
ELFEXT	equal-level far-end crosstalk
EMB	effective modal bandwidth
EMI	Electromagnetic Interference
EPD	End_of_Packet Delimiter
ESD	end of stream delimiter
FCS	Frame Check Sequence
FC-PH	Fibre Channel - Physical and Signaling Interface
FOTP	fiber optic test procedure
GMII	Gigabit Media Independent Interface
IEC	International Electrotechnical Commission
IPG	interPacketGap

IH	intermediate hub
IRL	inter-repeater link
ISI penalty	intersymbol interference penalty
ISO	International Organization for Standardization
LAN	local area network
LLC	logical link control
LSDV	link segment delay value
MAC	medium access control
MAU	medium attachment unit
MC	message code
MDELFEXT	multiple-disturber equal-level far-end crosstalk
MDFEXT	multiple-disturber far-end crosstalk
MDI	medium dependent interface
MDNEXT	multiple-disturber near-end crosstalk
MIB	management information base
MII	media independent interface
MMF	multimode fiber
MP	message page
NEXT	near-end crosstalk
NLP	normal link pulse
NPA	next page algorithm
NRZI	non return to zero and invert on ones
OFL	overfilled launch
OFSTP	optical fiber system test procedure
PCS	physical coding sublayer
PDV	path delay value
PHY	Physical Layer entity sublayer
PICS	protocol implementation conformance statement
PLS	physical signaling sublayer
PMA	physical medium attachment
PMD	physical medium dependent
PMI	physical medium independent
PPD	peak-to-peak differential
PVV	path variability value
RD	running disparity
RFI	Radio Frequency Interference
RIN	relative intensity noise
ROFL	radial overfilled launch
RS	reconciliation sublayer
RX_DV	An MII Signal (see IEEE 802.3 section 22.2.2.6)
RX_ER	An MII Signal (see IEEE 802.3 section 22.2.2.8)
SA	Source Address
SDV	segment delay value
SFD	start-of-frame delimiter
SMF	singlemode fiber
SPD	Start_of_Packet Delimiter
SR	symbol rate

SSD	start-of-stream delimiter
ST	symbol time
STE	station management entity
STP	shielded twisted pair (copper)
SVV	segment variability value
TDR	Time Domain Reflectometer
TIA	Telecommunications Industry Association
UCT	unconditional transition
UP	unformatted page
UTP	unshielded twisted pair
WCMB	worst case modal bandwidth