BACKPLANE ETHERNET CONSORTIUM

Clause 73 Auto-Negotiation
State Machine Test Suite

Version 1.0

Technical Document

Last Updated: March 23, 2011 12:17 PM

Backplane Ethernet Consortium

University of New Hampshire
InterOperability Laboratory

121 Technology Drive, Suite 2
Durham, NH 03824
Phone: (603) 862-0090
Fax: (603) 862-4181
http://www.iol.unh.edu/consortiums/bp

TABLE OF CONTENTS

MODIFICATION RECORD

Revision	Release Date	Author	Comments
0.1	11/30/2007	Jon Beckwith	• Initial Preliminary Draft
0.2	02/11/2009	Jon Beckwith Matthew Senerchia	• Added tests in section 2 • Re-organized tests in section 1
0.21	02/13/2009	Jon Beckwith	• Added 73.1.6-73.1.9 • Changed to include several tests in 73.1.5
0.22	02/17/2009	Jon Beckwith	• Added discussion to 73.2.1, 73.2.2 & 73.2.3 • Other minor editorial changes
0.23	02/18/2009	Jon Beckwith	• Fixed formatting issues • Added discussion to 73.2.4
0.24	02/19/2009	Jon Beckwith	• Did procedure for 2.6
0.25	02/20/2009	Jon Beckwith	• Added section 3 • Added "Range of data detect timer" and "Range of clock detect timer" tests • Probably removing Incomplete DME test and Single link ready
0.26	02/26/2009	Matthew Senerchia	• Updated procedures for several group 2 tests
0.27	03/03/2009	Jon Beckwith	• Removed Single Link ready test • Added discussion for the remainder of the group 2 tests
0.30	03/04/2009	Jon Beckwith	• Completed procedure for 2.10 • Released for Internal review
0.4	03/05/2009	Jon Beckwith	• Incorporated changes • Added 802.3ba related content • Added Ack Nonce match test
0.6	03/20/2009	Jon Beckwith	• Changed format – major overhaul
0.61	03/24/2009	Matt Senerchia	• Wrote test procedures for Group 2 and group 3
0.62	03/24/2009	Jon Beckwith	• Big overhaul • Added Group 0 for information • Added timers to group 2 • Added Next Page, Management, and Functionality groups • Added tests 73.1.7, 73.1.8, 73.2.13, 73.4.3
0.63	03/25/2009	Jon Beckwith	• Removed Next Page and management groups • Put Next Page tests in different sections
0.64	03/26/2009	Jon Beckwith	• Incorporated Geoff's comments for Group 2 • Added discussions and fixed references for 73.2.1 – 73.2.13 • Removed Group 0
0.66	03/15/2009	Geoff Mitchell	• Added numerous changes to the procedure sections.
0.69	04/20/2009	Geoff Mitchell	• Updated discussion sections.
0.70	05/08/2009	Jon Beckwith	• Made final revisions
0.80	09/11/2009	Matthew Senerchia	• Fixed tests 73.2.8 and 73.4.3b
1.0	3/23/2011	Brizer St. Cyr Jon Beckwith	• Added informative and clarifying text to 73.2.12, 73.2.13, 73.2.14, 73.2.17, 73.2.18, 73.2.19, 73.4.2, and 73.4.4 • Removed part c from 73.2.20 • Added updated setup diagram to Appendix A

ACKNOWLEDGMENTS

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

Jon Beckwith	UNH InterOperability Laboratory
Dave Estes	UNH InterOperability Laboratory
Jeff Lapak	UNH InterOperability Laboratory
Geoff Mitchell	UNH InterOperability Laboratory
Matthew Senerchia	UNH InterOperability Laboratory
Brizer St. Cyr	UNH InterOperability Laboratory

INTRODUCTION

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 particular suite of tests has been developed to help implementers evaluate the functionality of the Autonegotiation sublayer of their Backplane Ethernet products.

These tests are designed to determine if a product conforms to specifications defined in Clause 73 of the IEEE 802.3-2008 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 Backplane Ethernet environments.

The tests contained in this document are organized in such a manner as to simplify the identification of information related to a test, and to facilitate in the actual testing process. Tests are organized into groups, primarily in order to reduce setup time in the lab environment, however the different groups typically also tend to focus on specific aspects of device functionality. A three-part numbering system is used to organize the tests, where the first number indicates the clause of the IEEE 802.3 standard on which the test suite is based. The second and third numbers indicate the test's group number and test number within that group, respectively. This format allows for the addition of future tests to the appropriate groups without requiring the renumbering of the subsequent tests.

The test definitions themselves are intended to provide a high-level description of the motivation, resources, procedures, and methodologies pertinent to each test. Specifically, each test description consists of the following sections:

Purpose

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

References

This section specifies source material *external* to the test suite, including specific subclauses pertinent to the test definition, or any other references that might be helpful in understanding the test methodology and/or test results. External sources are always referenced by number when mentioned in the test description. Any other references not specified by number are stated with respect to the test suite document itself.

Resource Requirements

The requirements section specifies the test hardware and/or software needed to perform the test. This is generally expressed in terms of minimum requirements, however in some cases specific equipment manufacturer/model information may be provided.

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 initial configuration of the test environment. Small changes in the configuration should not be included here, and are generally covered in the test procedure section, below.

Test Procedure

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

Observable Results

This section lists the specific observables that can be examined by the tester in order to verify that the DUT is operating properly. When multiple values for an observable are possible, this section provides a short discussion on how to interpret them. The determination of a pass or fail outcome for a particular test is generally based on the successful (or unsuccessful) detection of a specific observable.

Possible Problems

This section contains a description of known issues with the test procedure, which may affect test results in certain situations. It may also refer the reader to test suite appendices and/or whitepapers that may provide more detail regarding these issues.

GROUP 1: DME PAGE TRANSMISSION

Overview:

The tests defined in this section verify the transmission of differential Manchester encoded (DME) pages used in the Backplane Ethernet Auto-Negotiation sublayer defined in Clause 73 of IEEE 802.3-2008.

These tests are designed to verify that the device under test transmits acceptable DME pages, which are properly formed with acceptable content making up the page transmitted by the device.

Test 73.1.1 – DME Transmit Differential Peak-to-peak Output Voltage

Purpose: To verify that the differential peak-to-peak output amplitude of the transmitted DME waveform is within the conformance limits.

References:

[1] IEEE Std 802.3-2008, subclause 73.5 – DME Transmission
[2] IEEE Std 802.3-2008, subclause 73.5.1.1, Table 73-1 – DME electrical specifications
[3] IEEE Std 802.3-2008, subclause 72.6.10.2.2 – Control channel encoding

Resource Requirements: See Appendix 73.A

Last Modification: May 8, 2009

Discussion:

Reference [1] specifies Backplane Ethernet Auto-Negotiation relative to differential Manchester encoding. Reference [2] defines the electrical characteristics of a DME page when transmitted and received. Rules of differential Manchester encoding are discussed in [3]. The peak-to-peak differential output voltage is specified in [2] to be measured at TP1. The measurement range for the peak voltage of the DME page is not specified, as such, the Manchester violation delimiter will be included in the measurement.

Test Setup: See Appendix 73.A

Test Procedure:

1. Configure the DUT to transmit DME pages according to reference [3].
2. Connect the DUT's Lane 0 transmitter to an oscilloscope as described in Appendix 73.A.
3. Measure the differential peak-to-peak output amplitude of the DME signaling, including the Manchester violation delimiter.
4. For enhanced accuracy, repeat steps 2-3 and average the results.

Observable Results:

a. The peak-to-peak differential output amplitude should be between 600 and 1200 mV when measured at TP1.

Possible Problems: None

Test 73.1.2 – DME Transition Period Spacing

Purpose: To verify that the spacing between transitions in a DME page is within conformance limits.

References:

[1] IEEE Std 802.3-2008, subclause 73.5 – DME Transmission
[2] IEEE Std 802.3-2008, subclause 73.5.3 – DME page timing
[3] IEEE Std 802.3-2008, Table 73-2 – DME page timing summary
[4] IEEE Std 802.3-2008, Figure 73-4 – DME page transition timing
[5] IEEE Std 802.3-2008, subclause 72.6.10.2.2 – Control channel encoding
[6] UNH-IOL 100BASE-TX TP-PMD Test Suite, Appendix 25.C

Resource Requirements: See Appendix 73.A

Last Modification: April 20, 2009

Discussion:

Reference [1] specifies Backplane Ethernet Auto-Negotiation relative to differential Manchester encoding. Reference [2] specifies the timing characteristics for DME Pages. Rules of differential Manchester encoding are discussed in [5]. Each cell boundary contains a transition, with an additional transition in the middle of a cell for a logical one.

Reference [3] specifies the transition spacing for DME pages, and reference [4] shows an example. To calculate the transition position spacing, a similar methodology is used as the one presented in reference [6].

Test Setup: See Appendix 73.A

Test Procedure:

1. Configure the DUT to transmit DME pages according to reference [5].
2. Connect the DUT's Lane 0 transmitter to an oscilloscope as described in Appendix 73.A.
3. Measure the spacing between all transitions in one page.
4. To calculate transition position spacing, recover the clock from the signal using a phase-locked loop method, or other means (see reference [6]).
5. Measure the crossing times for logical ones (clock-to-data and data-to-clock) and logical zeros (clock-to-clock).
6. For enhanced accuracy, take multiple captures and average the results.

Observable Results:

a. The transition position spacing (interval_timer) should be 3.2 ns ± 0.01% (0.32 ps).
b. For data 0's, the clock-to-clock transition spacing should be 6.4 ns ± 0.2 ns
c. For data 1's, the clock-to-data and data-to-clock transition spacing should be 3.2 ns ± 0.2 ns

Possible Problems: None

Test 73.1.3 – DME Page Width

Purpose: To verify that the width of a DME page is within conformance limits.

References:

[1] IEEE Std 802.3-2008, subclause 73.5 – DME Transmission
[2] IEEE Std 802.3-2008, subclause 73.5.3 – DME page timing
[3] IEEE Std 802.3-2008, Table 73-2 – DME page transition timing
[4] IEEE Std 802.3-2008, Figure 73-5 – Manchester violation
[5] IEEE Std 802.3-2008, subclause 72.6.10.2.2 – Control channel encoding

Resource Requirements: See Appendix 73.A

Last Modification: April 20, 2009

Discussion:

Reference [1] specifies Backplane Ethernet Auto-Negotiation relative to differential Manchester encoding. Reference [2] specifies the timing characteristics for DME Pages. Rules of differential Manchester encoding are discussed in [5]. Each cell boundary contains a transition, with an additional transition in the middle of a cell for a logical one.

Reference [3] specifies that the width of DME page shall be 339.2 ns ± 0.4 ns between Manchester violations (signaled as shown in reference [4]).

Test Setup: See Appendix 73.A

Test Procedure:

1. Configure the DUT to transmit DME pages according to reference [5].
2. Connect the DUT's Lane 0 transmitter to an oscilloscope as described in Appendix 73.A.
3. Measure the length of the entire DME page, starting with the beginning of a Manchester violation. delimiter and ending with the beginning of the next Manchester violation delimiter.
4. For enhanced accuracy, take multiple captures and average the results.

Observable Results:

a. The DME page width should be 339.2 ns ± 0.4 ns.

Possible Problems: None

Test 73.1.4 – DME Manchester Violation Delimiter Width

Purpose: To verify that the Manchester violation delimiter width is within conformance limits.

References:

[1] IEEE Std 802.3-2008, subclause 73.5 – DME Transmission
[2] IEEE Std 802.3-2008, subclause 73.5.3 – DME page timing
[3] IEEE Std 802.3-2008, Table 73-2 – DME page transition timing
[4] IEEE Std 802.3-2008, Figure 73-5 – Manchester violation
[5] IEEE Std 802.3-2008, subclause 72.6.10.2.2 – Control channel encoding

Resource Requirements: See Appendix 73.A

Last Modification: April 20, 2009

Discussion:

Reference [1] specifies backplane Ethernet auto negotiation relative to differential Manchester encoding. Reference [2] specifies the timing characteristics for DME Pages.

Rules of differential Manchester encoding are discussed in [5]. Each cell boundary contains a transition, with an additional transition in the middle of a cell for a logical one. Manchester violation delimiters have transitions at positions 1 and 5, and are signaled as shown in reference [4]. T6 is denoted as the half-width time, or time between the 3 adjacent transitions that make up the Manchester violation delimiter. Reference [3] specifies that the half-width of the DME Manchester violation delimiter (t6) shall be 12.8 ns ± 0.2 ns.

Test Setup: See Appendix 73.A

Test Procedure:

1. Configure the DUT to transmit DME pages according to reference [4].
2. Connect the DUT's Lane 0 transmitter to an oscilloscope as described in Appendix 73.A.
3. Measure the width of both the positive pulse and negative pulse that make up the Manchester violation delimiter.
4. For enhanced accuracy, take multiple captures and average the results.

Observable Results:

a. The Manchester violation delimiter half-width (t6) should be 12.8 ns ± 0.2 ns.

Possible Problems: None

Test 73.1.5 – DME Base Page Encoding

Purpose: To verify that the DUT transmits valid DME data; including an acceptable Selector Field combination, correct ability advertisements in the Technology Ability Field, and proper initial values for the Remote Fault, Acknowledge, Next Page, and FEC Capability bits.

References:

[1] IEEE Std 802.3-2008, subclause 73.5 – DME Transmission
[2] IEEE Std 802.3-2008, Table 73-2 – DME page timing summary
[3] IEEE Std 802.3-2008, subclause 73.6 – Link Codeword encoding
[4] IEEE Std 802.3-2008, subclause 72.6.10.2.2 – Control channel encoding

Resource Requirements: See Appendix 73.A

Last Modification: April 20, 2009

Discussion:

Reference [1] specifies backplane Ethernet Auto-Negotiation relative to differential Manchester encoding. Reference [2] specifies the timing characteristics for DME Pages, and the rules for encoding of each link codeword are provided in reference [3].

Rules of differential Manchester encoding are discussed in [4]. Each cell boundary contains a transition, with an additional transition in the middle of a cell for a logical one. The format of the Link Codeword is shown in Figure 73-6 below, for convenience.

D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12	D13	D14	D15
S0	S1	S2	S3	S4	E0	E1	E2	E3	E4	C0	C1	C2	RF	Ack	NP

F

D16	D17	D18	D19	D20	D21	D22	D23	D24	D25	D26	D27	D28	D45	D46	D47
T0	T1	T2	T3	T4	A0	A1	A2	A3	A4	A5	A6	A7	A24	F0	F1

Figure 73.1-1: Link codeword base page

Test Setup: See Appendix 73.A

Test Procedure:

1. Configure the DUT to transmit DME pages according to reference [4].
2. Connect the DUT's Lane 0 transmitter to an oscilloscope as described in Appendix 73.A and monitor the transmitted pages.
3. Observe the number of transitions in the DME page over several captures.
4. Decode the DME page and verify the contents.
5. For enhanced accuracy, take multiple captures.

Observable Results:

a. The number of transitions in a DME page shall be between 51 and 100.
b. The encoding of the DME page shall follow the following statements
   1. The Selector Field combination should correspond to S[4:0] = 00001.
   2. The Echoed Nonce field should contain logical zeros when Acknowledge is set to logical zero.

3. The Pause Capabilities field should advertise the proper abilities in C[1:0]. C[2] is reserved and can be either 0 or 1.
4. The Transmitted Nonce field should contain a pseudo-random number, which remains consistent until a new entry into the ABILITY DETECT state.
5. The initial value of the Remote Fault bit should be zero.
6. The initial value of the Acknowledge bit should be zero.
7. The value of the Next Page bit should be one if the DUT wishes to engage in a Next Page exchange and zero if it doesn't.
8. The Technology Ability field should advertise the proper abilities as indicated in Table 73-4.
9. The FEC Capability field should be 1 if the HCD is 10GBASE-KR and 0 if the HCD is 1000BASE-KX or 10GBASE-KX4.

Possible Problems: None

Test 73.1.6 – Transmitted Nonce Value

Purpose: To verify that the DUT generates a new transmitted nonce for each entry into the ABILITY DETECT state, and that the distribution of the values is uniform.

References:

[1] IEEE Std 802.3-2008, subclause 73.5 – DME Transmission
[2] IEEE Std 802.3-2008, subclause 72.6.10.2.2 – Control channel encoding
[3] IEEE Std 802.3-2008, subclause 73.6.3 – Transmitted Nonce field

Resource Requirements: See Appendix 73.A

Last Modification: April 20, 2009

Discussion:

Reference [1] specifies Backplane Ethernet Auto-Negotiation relative to differential Manchester encoding. Rules of differential Manchester encoding are discussed in [2]. Each cell boundary contains a transition, with an additional transition in the middle of a cell for a logical one.

Reference [3] specifies that the Transmitted Nonce field shall have a new value for each entry into the ABILITY DETECT state. While the method of generating a nonce is left to the implementer, the values used should have a uniform distribution from 0 to 2^5–1.

Test Setup: See Appendix 73.A

Test Procedure:

1. Connect the DUT to the test setup shown in Figure 73.A-1.
2. Monitor the transmitted pages and record the DUT's Transmitted Nonce value.
3. Use the Traffic Generator to send a series of DME pages with a Transmitted Nonce equal to the DUT's Transmitted Nonce value.
4. The DUT should cease transmission, enter TRANSMIT DISABLE, and re-enter ABILITY DETECT.
5. Monitor the transmitted pages and record the DUT's Transmitted Nonce value.
6. Verify that the DUT starts a re-negotiation.
7. For enhanced accuracy, take multiple captures.

Observable Results:

a. The Transmitted Nonce value should update for each entry into the ABILITY DETECT state.
b. INFORMATIVE: The distribution of the transmitted nonce values should be uniform between 0 and 2^5–1.

Possible Problems: None

Test 73.1.7 – Bit 49 value

Purpose: To verify that the value of bit 49 in the transmitted DME page is pseudo-random.

References:

[1] IEEE Std 802.3-2008, subclause 72.6.10.2.2 – Control channel encoding
[2] IEEE Std 802.3-2008, Figure 73-2 – DME page bit 49 randomizer

Resource Requirements: See Appendix 73.A

Last Modification: April 20, 2009

Discussion:

Bit 49 is generated using one of the 2 DME page bit 49 randomizer equations: $x^7 + x^3 + 1$ or $x^7 + x^6 + 1$ (shown in reference [2]). The purpose of the 49th bit is to remove the spectral peaks that would otherwise occur when sending the same AN page repeatedly.

Test Setup: See Appendix 73.A

Test Procedure:

1. Configure the DUT to transmit DME pages according to reference [1].
2. Connect the DUT's Lane 0 transmitter to an oscilloscope as described in Appendix 73.A and monitor the transmitted pages
3. Observe the value of bit 49 in the DME page over several captures.
4. For enhanced accuracy, take multiple captures.

Observable Results:

a. The value of bit 49 shall be pseudo-random, derived from the source as defined in reference [2].

Possible Problems: None

Test 73.1.8 – Next Page Generation

Purpose: To verify that the DUT transmits properly formatted Next pages.

References:

[1] IEEE Std 802.3-2008, subclause 73.5 – DME Transmission
[2] IEEE Std 802.3-2008, subclause 72.6.10.2.2 – Control channel encoding
[3] IEEE Std 802.3-2008, subclause 73.7.7 – Next Page function
[4] IEEE Std 802.3-2008, Figure 73-7 – Message next page
[5] IEEE Std 802.3-2008, Figure 73-8 – Unformatted next page
[6] IEEE Std 802.3-2008, Annex 73A – Next page message code field definitions

Resource Requirements: See Appendix 73.A

Last Modification: April 20, 2009

Discussion:

Reference [1] specifies Backplane Ethernet Auto-Negotiation relative to differential Manchester encoding. Rules of differential Manchester encoding are discussed in [2]. Each cell boundary contains a transition, with an additional transition in the middle of a cell for a logical one.

Reference [3] defines how a Next Page exchange should take place. References [4] and [5] define the formatting for both Message pages and Unformatted pages. Next Page support is mandatory for devices implementing clause 73. Currently there are no defined Next Pages for a device which supports clause 73 (other than Null Message pages), thus any Next Page transmitted by the DUT should be either a Null Message page or a vendor specific unformatted page and follow the format in reference [6].

Test Setup: See Appendix 73.A

Test Procedure:

1. Configure the DUT to transmit DME pages according to reference [4].
2. Connect the DUT's Lane 0 transmitter to an oscilloscope as described in Appendix 73.A and monitor the transmitted pages
3. Use the traffic generator to send a series of DME pages with the Next Page and Acknowledge bits set to send the DUT into the COMPLETE ACKNOWLEDGE state.
4. Continue sending DME pages with NP=0, all with proper Toggle and ACK bit values, until the DUT sets its NP=0.
5. Observe transmissions from the DUT.

Observable Results:

a. The DUT should transmit validly formatted Next Pages as defined in reference [6].

Possible Problems: None

GROUP 2: STATE DIAGRAM VARIABLES, TIMERS, AND COUNTERS

Overview:

The tests defined in this section cover operation specific to the reception of differential Manchester encoded (DME) pages used in Backplane Ethernet Auto-Negotiation defined in Clause 73 of IEEE 802.3-2008.

These tests are designed to verify that the DUT properly implements the various state diagram variables, timers, and counters as defined in subclause 73.10.1, 73.10.2, and 73.10.3 respectively.

Test 73.2.1 – Ability Match

Purpose: To verify that the DUT properly implements ability_match.

References:

[1] IEEE Std 802.3-2008, 73.10.1 – State Diagram Variables
[2] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:

Reference [1] specifies the variables used in the Arbitration State Diagram, found in reference [2]. This test is designed to verify proper ability_match behavior as defined in reference [1], which states that ability_match should be set to true upon reception of three matching consecutive link code words, ignoring the Acknowledge bit.

This test verifies proper implementation of ability_match by observing the exit conditions from the ABILITY DETECT state.

Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Use the Traffic Generator to send 1 DME page to the DUT containing a Transmitted Nonce value that does not equal the one used by the DUT.
2. Monitor the DME pages sent back by the DUT and determine if the Acknowledge bit is set.
3. If it is not set, repeat steps 2 and 3, increasing the number (n) of DME pages until the Acknowledge bit is set by the DUT.
4. Send (n) DME pages to the DUT all with the Acknowledge bit set to logic one, where (n) is the minimum number of DME pages determined in step 4 to cause the DUT to set ability_match=true.
5. Repeat step 5 using DME pages that have Acknowledge bits that alternate between 1 and 0.

Part b:

6. Use the traffic generator to send (n) DME pages, alternating between an initial DME page and a valid DME page containing different advertised abilities.
7. Monitor the DME pages sent back by the DUT and determine whether the Acknowledge bit is set.
8. Repeat steps 6-7 by sending all combinations of DME pages that are one bit different than the initial DME.

Observable Results:

a. The Acknowledge bit should be set after the reception of at least 3 complete and matching DME pages, regardless of the value of the received Acknowledge bit. Record the number of DME pages required to put the DUT into the ACKNOWLEDGE DETECT state for use in later tests.
b. The DUT should not set its Acknowledge bit.

Possible Problems: None

Test 73.2.2 – Nonce Match

Purpose: To verify that the DUT properly implements nonce_match.

References:

[1] IEEE Std 802.3-2008, 73.10.1 – State Diagram Variables
[2] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:

Reference [1] specifies the variables used in the Arbitration State Diagram, found in reference [2]. This test is designed to verify proper nonce_match behavior in reference [2], which indicates whether the transmitted nonce received from the link partner matches the DUT's transmitted nonce. The Nonce field is designed to ensure that the local device does not establish a link because of self or alien crosstalk. This variable is used as an exit condition from ABILITY DETECT.

This test verifies proper implementation of nonce_match by observing the exit conditions from ABILITY DETECT.

Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Send the DUT (n) DME pages with the Acknowledge bit not set and the Transmitted Nonce field not equal to the Transmitted Nonce field of the DUT.
2. Observe transmissions from the DUT.

Part b:

3. Send the DUT (n) DME pages with the Acknowledge bit not set and the Transmitted Nonce field equal to the Transmitted Nonce field of the DUT.
4. Observe transmissions from the DUT.

Observable Results:

a. The DUT should set its Acknowledge bit after the reception of (n) DME pages with a different Transmitted Nonce field value then its own.
b. The DUT should stop sending DME pages for approximately break_link_timer, and then restart its Base Page transmission.

Possible Problems: None

Test 73.2.3 – Acknowledge Match

Purpose: To verify that the DUT properly implements acknowledge_match.

References:

[1] IEEE Std 802.3-2008, 73.10.1 – State Diagram Variables
[2] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:

Reference [1] specifies the variables used in the Arbitration State Diagram, found in reference [2]. This test is designed to verify proper acknowledge_match behavior in reference [2], which indicates that three consecutive link code words have been received with the Acknowledge bit set.

This test verifies proper implementation of acknowledge_match by observing the exit conditions from ACKNOWLEDGE DETECT.

Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Send the DUT (n) DME pages with the Acknowledge bit not set and the Transmitted Nonce field not equal to the Transmitted Nonce field of the DUT. (n) is the value found in test #73.2.1 to cause the DUT to set ability_match=true.
2. Send the DUT (m) DME pages with the Acknowledge bit set and an Echoed Nonce field that matches the Transmitted Nonce field sent by the DUT, but otherwise identical to the initial DME pages. (m) is initially set to one.
3. Observe transmissions from the DUT.
4. Repeat steps 1-3, increasing (m) until the DUT is observed to set acknowledge_match=true.

Part b:

5. Send the DUT (n) DME pages with the Acknowledge bit not set and the Transmitted Nonce field not equal to the Transmitted Nonce field of the DUT. (n) is the value found in test #73.2.1 to cause the DUT to set ability_match=true.
6. Send the DUT (2*m) DME pages that alternate between DME pages identical to the first group, and DME pages that are one bit different from the first group; however, all DME pages in this group have the Acknowledge bit set and an Echoed Nonce field which matches the Transmitted Nonce field sent by the DUT.
7. Observe transmissions from the DUT.
8. Repeat steps 5-7 by sending all combinations of DME pages that are one bit different than the initial DME, except for the Acknowledge bit and the Echoed Nonce field.

.

Observable Results:

a. The DUT should enter the COMPLETE ACKNOWLEDGE state after receiving three DME pages with the Acknowledge bit set to logic one.
b. The DUT should never enter the COMPLETE ACKNOWLEDGE state, and should send out DME pages with the Acknowledge bit set. Following the DME pages should be a gap of approximately break_link_timer, at which point the DUT should resume DME page transmissions.

Possible Problems: None

Test 73.2.4 – Ack Nonce Match

Purpose: To verify that the DUT properly implements ack_nonce_match.

References:
[1] IEEE Std 802.3-2008, 73.10.1 – State Diagram Variables
[2] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:
Reference [1] specifies the variables used in the Arbitration State Diagram, found in reference [2]. This test is designed to verify proper ack_nonce_match behavior in reference [2], which indicates that the echoed nonce received from the link partner matches the DUT's transmitted nonce. The Echoed Nonce field is in place as an extra assurance that a link partner has received the local device's DME page.

This test verifies proper implementation of ack_nonce_match by observing the exit conditions from ACKNOWLEDGE DETECT.

Test Setup: See Appendix 73.A

Test Procedure:
Part a:
1. Use the traffic generator to send (n) DME pages with the Acknowledge bit not set and the Transmitted Nonce field not equal to the Transmitted Nonce field of the DUT.
2. Send (m) DME pages with the Acknowledge bit set and an Echoed Nonce field that equals the Transmitted Nonce field sent by the DUT. (m) is the value found in test #73.2.3 to cause the DUT to set acknowledge_match=true.
3. Observe transmissions from the DUT.

Part b:
4. Use the traffic generator to send (n) DME pages with the Acknowledge bit not set and the Transmitted Nonce field not equal to the Transmitted Nonce field of the DUT.
5. Send (m) DME pages with the Acknowledge bit set and an Echoed Nonce field that does not equal the Transmitted Nonce field sent by the DUT. (m) is the value found in test #73.2.3 to cause the DUT to set acknowledge_match=true.
6. Observe transmissions from the DUT.

Observable Results:
a. The DUT should enter the COMPLETE ACKNOWLEDGE state after receiving three DME pages with the Acknowledge bit set to logic one and an Echoed Nonce field that matches its Transmitted Nonce.
b. Upon reception of pages with an invalid Echoed Nonce field the DUT should stop sending DME pages for approximately break_link_timer, after which the DUT should resume DME page transmissions.

Possible Problems: None

Test 73.2.5 – Consistency Match

Purpose: To verify that the DUT performs a consistency match test on received DME pages.

References:

[1] IEEE Std 802.3-2008, 73.10.1 – State Diagram Variables
[2] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:

Reference [1] specifies the variables used in the Arbitration State Diagram, found in reference [2]. This test is designed to verify proper consistency_match behavior in reference [2], which indicates whether the link code words that caused ability_match=true is the same as the one that caused acknowledge_match=true.

This test verifies proper implementation of consistency_match by observing the exit conditions from ACKNOWLEDGE DETECT.

Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Use the traffic generator to send (n) DME pages all with the Acknowledge bit set and valid Nonce fields to cause the DUT to enter the ACKNOWLEDGE DETECT state.
2. Observe transmissions from the DUT.
3. Repeat steps 1-2, increasing the number of DME pages until the DUT is observed to enter the COMPLETE ACKNOWLEDGE state.

Part b:

4. Send the DUT (n) DME pages with the Acknowledge bit not set and the Transmitted Nonce field not equal to the Transmitted Nonce field of the DUT. (n) is the value found in test #73.2.1 to cause the DUT to set ability_match=true.
5. Send the DUT (m) DME pages that are one bit different from the first group, but all DME pages in this group have the Acknowledge bit set and a valid Echoed Nonce field. (m) is the value found in test #73.2.2.
6. Observe transmissions from the DUT.
7. Repeat steps 4-6 by sending all combinations of DME pages that are one bit different than the initial DME, except for the Acknowledge bit and the Echoed Nonce field.

Observable Results:

a. The DUT should enter COMPLETE ACKNOWLEDGE after reception of 4, 5, 6, or 7 DME pages with the ACK bit set.
b. The DUT should set consistency_match=false and enter the TRANSMIT DISABLE state immediately after receiving the inconsistent DME pages.

Possible Problems: None

Test 73.2.6 – Ack Finished

Purpose: To verify that the DUT properly sends 6 to 8 (inclusive) DME pages before setting ack_finished=true.

References:

[1] IEEE Std 802.3-2008, 73.10.1 – State Diagram Variables
[2] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram
[3] IEEE Std 802.3-2008, 73.10.3 – State Diagram Counters

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:

Reference [1] specifies the variables used in the Arbitration State Diagram, found in reference [2]. This test is designed to verify proper ack_finished behavior in reference [2], which indicates that 6 to 8 (inclusive) link code words (remaining_ack_cnt) have been transmitted with the Acknowledge bit set to logic one, to ensure the link partner receives the acknowledgement. This variable is assigned within the ABILITY DETECT and NEXT PAGE WAIT states and used as an exit condition from COMPLETE ACKNOWLEDGE.

This test verifies proper implementation of ack_finished by observing the exit conditions from COMPLETE ACKNOWLEDGE.

Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Use the Traffic Generator to send a series of (n) DME pages with the Acknowledge bit not set and valid nonce fields, followed by (m) DME pages with the Acknowledge bit set to logic one and valid nonce fields, to put the DUT into the COMPLETE ACKNOWLEDGE state.
2. Monitor the transmit line coming from the DUT and count the number of DME pages sent by the DUT after the COMPLETE ACKNOWLEDGE state has been entered.

Part b:

3. Use the Traffic Generator to send a series of (n) DME pages with the Next Page bit set, the Acknowledge bit not set, and valid nonce fields, followed by (m) DME pages with the Next Page and Acknowledge bits set and valid nonce fields, to put the DUT into the COMPLETE ACKNOWLEDGE state.
4. Send the DUT a series of Next Pages with the Acknowledge and Next Page bits not set.
5. Observe transmissions from the DUT.

Observable Results:

a. After COMPLETE ACKNOWLEDGE state has been entered, the DUT should send out 6 to 8 (inclusive) DME pages containing its Link code word before setting ack_finished=true and entering the AN GOOD CHECK state.
b. After COMPLETE ACKNOWLEDGE state has been entered, the DUT should send out 6 to 8 (inclusive) DME pages containing its Link code word before setting ack_finished=true and entering the NEXT PAGE WAIT state.

Possible Problems: None

Test 73.2.7 – Break Link Timer

Purpose: To verify that the time between when the DUT ceases and resumes transmission of DME pages is within the conformance limits.

References:

[1] IEEE Std 802.3-2008, subclause 73.7.5 – Renegotiation function
[2] IEEE Std 802.3-2008, subclause 73.10.2 – State diagram timers
[3] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:

Reference [1] specifies the Renegotiation function, and reference [2] specifies the timers for the Auto-Negotiation Arbitration State Diagram in [3]. This test is designed to verify the amount of time the DUT waits between disabling and restarting transmission.

Once a device has entered the TRANSMIT DISABLE state, it must wait a specified amount of time before it restarts the Auto-Negotiation process. This time is defined by the device's "break_link_timer," and is required to be between 60 and 75 ms. This test is designed to verify that the DUT restarts the ANEG process within the proper amount of time.

Test Setup: See Appendix 73.A

Test Procedure:

1. Use the Traffic Generator to send a series of DME pages with a Transmitted Nonce equal to the DUT's Transmitted Nonce value.
2. Measure the interval from the time when the DUT ceased DME transmission (upon entry of the TRANSMIT DISABLE state) and when the first DME page of the re-negotiation process was transmitted. This time will be the value of break_link_timer.
3. For enhanced accuracy, repeat steps 1-2 multiple times.

Observable Results:

a. Assuming a fixed value for the implemented break_link_timer, the minimum of the observed gaps is the DUT's break_link_timer, which should be between 60 to 75 ms.

Possible Problems: None

Test 73.2.8 – Link Fail Inhibit Timer

Purpose: To verify that the DUT will defer for the proper amount of time before attempting to verify the status of the link determined by the Auto-Negotiation process.

References:

[1] IEEE Std 802.3-2008, subclause 73.7.5 – Renegotiation function
[2] IEEE Std 802.3-2008, subclause 73.10.2 – State diagram timers
[3] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:

Reference [1] specifies the Renegotiation function, and reference [2] specifies the timers for the Auto-Negotiation Arbitration State Diagram in [3]. This test is designed to verify the amount of time the DUT waits before beginning a renegotiation.

Once a device has entered the AN GOOD CHECK state, it must receive a link_status=OK message (the correct technology-specific signaling) from its link partner within a specified amount of time. If this message is not received, it will enter the TRANSMIT DISABLE state and wait break_link_timer before attempting a re-negotiation. The time that the device should transmit link signaling is defined by the device's link_fail_inhibit_timer, and is required to be 40 to 50 ms for 1000BASE-KX and 10GBASE-KX4, and 500 to 510 ms for 10GBASE-KR. This test is designed to verify that the device under test sends link signaling for an appropriate amount of time before attempting to renegotiate the link.

Test Setup: See Appendix 73.A

Test Procedure:

Part a: for devices with a 1000BASE-KX PMD:

1. Use a traffic generator to put the DUT into the AN GOOD CHECK state and resolve a 1000BASE-KX link by sending enough DME pages that advertise only 1000BASE-KX abilities.
2. Measure the amount of time 1000BASE-KX link signaling is sent from the DUT.
3. Verify that the DUT restarts Auto-Negotiation.
4. For enhanced accuracy, repeat steps 1-3 multiple times.

Part b: for devices with a 10GBASE-KX4 PMD:

5. Use a traffic generator to put the DUT into the AN GOOD CHECK state and resolve a 10GBASE-KX4 link by sending enough DME pages that advertise only 10GBASE-KX4 abilities.
6. Measure the amount of time 10GBASE-KX4 link signaling is sent from the DUT.
7. Verify that the DUT restarts Auto-Negotiation.
8. For enhanced accuracy, repeat steps 5-7 multiple times.

Part c: for devices with a 10GBASE-KR PMD:

9. Use a traffic generator to put the DUT into the AN GOOD CHECK state and resolve a 10GBASE-KR link by sending enough DME pages that advertise only 10GBASE-KR abilities.
10. Measure the amount of time 10GBASE-KR link signaling is sent from the DUT.
11. Verify that the DUT restarts Auto-Negotiation.
12. For enhanced accuracy, repeat steps 9-11 multiple times.

Observable Results:

a. For devices with a 1000BASE-KX PMD, observation of the interval between DME page cessation and link signaling cessation should be between 40 and 50 ms.

b. For devices with a 10GBASE-KX4 PMD, observation of the interval between DME page cessation and link signaling cessation should be between 40 and 50 ms.
c. For devices with a 10GBASE-KR PMD, observation of the interval between DME page cessation and link signaling cessation should be between 500 and 510 ms.

Possible Problems:

a-c) If the devices max_wait_timer from Clause 72 ends before link_fail_inhibit_timer does, the DUTs link_fail_inhibit_timer would be unobservable.

Test 73.2.9 – Auto-Negotiation Wait Timer

Purpose: To verify that the implemented value of autoneg_wait_timer is within the specified range.

References:

[1] IEEE Std 802.3-2008, subclause 73.7.4.1 – Parallel Detection function
[2] IEEE Std 802.3-2008, subclause 73.10.2 – State Diagram timers
[3] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:

Reference [1] specifies the parallel detection function, and reference [2] specifies the timers for the Auto-Negotiation Arbitration State Diagram in [3]. This test is designed to verify the amount of time required to establish a valid link through parallel detection.

In order for a Clause 73 Auto-Negotiation able device to properly parallel detect a link partner, it must receive a valid single_link_ready=TRUE signal from its link partner for a period before it negotiates to that link. This time is defined by the device's autoneg_wait_timer, and is specified to be between 25 and 50 ms. This test is designed to verify that the DUT does not parallel detect a link before waiting a time within this range. Parallel detection is supported for 1000BASE-KX and 10GBASE-KX4.

Test Setup: See Appendix 73.A

Test Procedure:

1. Use a Traffic Generator to send the station valid signaling at a speed supported by the device.
2. Verify that the DUT establishes a link.
3. Measure the time between when the DUT ceased transmission of DME pages and when the DUT began transmitting link signaling.
4. Repeat for all valid signaling speeds supported by the device.

Observable Results:

a. The DUT's autoneg_wait_timer should be in the range of 25 to 50 ms.

Possible Problems: None

Test 73.2.10 – Data Detect Timers

Purpose: To verify that the DUT properly detects data transitions with valid spacing.

References:

[1] IEEE Std 802.3-2008, subclause 73.10.2 – State Diagram timers
[2] IEEE Std 802.3-2008, Figure 73-10 – Receive State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:

Reference [1] specifies the timer summary for the Auto-Negotiation Receive State Diagram in [2]. This test is designed to verify the range of the data detect timers in reference [1]. In order to be accepted as valid data 1's, received data pulses must have a delay between them and the previous clock pulse between data_detect_min_timer and data_detect_max_timer. This test is designed to verify that the DUT properly accepts data pulses spaced within these ranges.

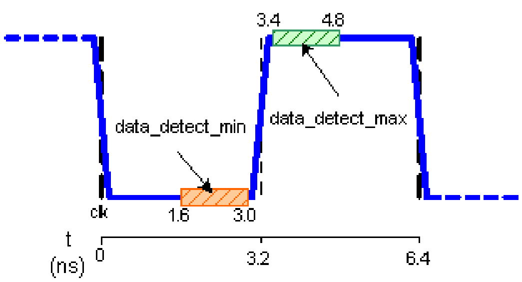


Figure 73.2-1: Data detect timers

Test Setup: See Appendix 73.A

Test Procedure:

Part a: data_detect_min_timer

1. Use the Traffic Generator to send the DUT (n) DME pages, alternating between DME pages with the first data pulse spaced 1.6 ns from the clock pulse and DME pages with normal spacing, to put it into the ACKNOWLEDGE DETECT state.
2. Observe whether the DUT entered the ACKNOWLEDGE DETECT state.
3. Repeat steps 1-2, increasing the gap from the clock pulse to the data pulse until the DUT enters the ACKNOWLEDGE DETECT state.

Part b: data_detect_max_timer

4. Repeat steps 1-3, varying the gap from the clock pulse to the data pulse, starting at 4.8 ns and decreasing, until the DUT enters the ACKNOWLEDGE DETECT state.

Observable Results:

a. The value of data_detect_min_timer should be between 1.6 and 3.0 ns.
b. The value of data_detect_max_timer should be between 3.4 and 4.8 ns.

Possible Problems: None

Test 73.2.11 – Clock Detect Timers

Purpose: To verify that the DUT properly detects clock transitions within proper valid spacing.

References:

[1] IEEE Std 802.3-2008, subclause 73.10.2 – State diagram timers
[2] IEEE Std 802.3-2008, Figure 73-10 – Receive State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:

Reference [1] specifies the timer summary for the Auto-Negotiation Receive State Diagram in [2]. This test is designed to verify the range of the clock detect timers in reference [1]. In order to be accepted as valid data 0's, received clock pulses must have a delay between them and the previous clock pulse between clock_detect_min_timer and clock_detect_max_timer. This test is designed to verify that the DUT properly accepts clock pulses spaced within these ranges.

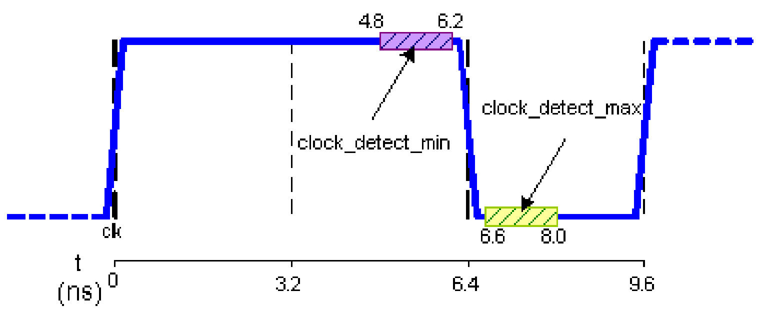


Figure 73.2-1: Clock detect timers

Test Setup: See Appendix 73.A

Test Procedure:

Part a: clock_detect_min_timer

1. Use the Traffic Generator to send the DUT (n) DME pages, alternating between DME pages with the third clock pulse spaced 4.8 ns from the second clock pulse and DME pages with normal spacing, to put it into the ACKNOWLEDGE DETECT state.
2. Observe whether the DUT entered the ACKNOWLEDGE DETECT state.
3. Repeat steps 1-2, varying the gap from clock pulse to clock pulse until the DUT enters the ACKNOWLEDGE DETECT state.

Part b: clock_detect_max_timer

4. Repeat steps 1-3, varying the gap from clock pulse to clock pulse, starting at 8.0 ns and decreasing, until the DUT enters the ACKNOWLEDGE DETECT state.

Observable Results:

a. The value of clock_detect_min_timer should be between 4.8 and 6.2 ns.
b. The value of clock_detect_max_timer should be between 6.6 and 8.0 ns.

Possible Problems: None

Test 73.2.12 – Page Test Timers

Purpose: To verify that the DUT accepts DME pages with proper spacing, and refuses those with spacing outside of the acceptable range.

References:

[1] IEEE Std 802.3-2008, subclause 73.10.2 – State diagram timers
[2] IEEE Std 802.3-2008, Figure 73-10 – Receive State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: March 23, 2011

Discussion:

Reference [1] specifies the timer summary for the Auto-Negotiation Receive State Diagram in [2]. This test is designed to verify proper values for the page test timers in reference [1].

In order to be accepted as valid, the start of DME pages must be separated by at least page_test_min_timer and at most page_test_max_timer. Manchester violation delimiters signify these boundaries between adjacent pages. This test is to verify that the DUT accepts DME pages with spacing within those ranges, and refuses DME pages with spacing outside of those ranges.

Test Setup: See Appendix 73.A

Test Procedure:

Part a: page_test_min_timer

1. Use the Traffic Generator to send the DUT (n) DME pages with proper number of transition and a width of 300 ns.
2. Observe whether the DUT entered the ACKNOWLEDGE DETECT state.
3. Repeat steps 1-2 increasing the width of the DME pages until the DUT enters the ACKNOWLEDGE DETECT state.

Part b: page_test_max_timer

4. Repeat steps 1-3 using a width of 380 ns and decreasing, until the range at which the DUT enters the ACKNOWLEDGE DETECT state.

Observable Results:

a. INFORMATIVE: The value of page_test_min_timer should be between 305 and 330 ns.
b. INFORMATIVE: The value of page_test_max_timer should be between 350 and 375 ns.

Possible Problems: Waveforms that are modified for this test violate the electrical timing requirements.

Test 73.2.13 – Rx Bit Count

Purpose: To verify that the DUT properly counts bits in the received pages.

References:

[1] IEEE Std 802.3-2008, subclause 73.10.3 – State diagram counters
[2] IEEE Std 802.3-2008, Figure 73-10 – Receive State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: March 23, 2011

Discussion:

Reference [1] specifies the counter summary for the Auto-Negotiation Receive State Diagram in [2]. This test is designed to verify that the DUT ignores additional bits in long DME pages.

A DME page normally consists of 51 to 100 pulses, with space for 49 data bits. However, if a device receives a DME with more than 49 data positions, it should still accept the page. The first 49 bits should be kept and any additional should be ignored. This test is designed to verify that the DUT properly accepts DME pages with more or less than 49 data bits.

Test Setup: See Appendix 73.A

Test Procedure:

Part a: acceptance of short DME pages

1. Use the Traffic Generator to send a series of (n) DME pages which contain less than 50 clock pulses, but whose total length is greater than page_test_min_timer.
2. Observe transmissions from the DUT
3. Increase until DUT enters ACKNOWLEDGE DETECT.

Part b: acceptance of long DME pages

4. Use the Traffic Generator to send a series of (n) DME pages which contain more than 50 clock pulses, but whose total length is less than page_test_max_timer.
5. Observe transmissions from the DUT.

Observable Results:

a. The DUT should not enter the ACKNOWLEDGE DETECT state until rx_bit_cnt equals 49.
b. The DUT should enter the ACKNOWLEDGE DETECT state, regardless of the value of rx_bit_cnt.

Possible Problems: None

Test 73.2.14 – Pulse Too Long

Purpose: To verify that the DUT properly implements pulse_too_long.

References:

[1] IEEE Std 802.3-2008, subclause 73.10.1 – State diagram variables
[2] IEEE Std 802.3-2008, Figure 73-10 – Receive State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:

Reference [1] specifies the variables used in the Auto-Negotiation Receive State Diagram, found in reference [2]. Pulse_too_long indicates that transitions separated by more than 20 ns have been detected. This variable is used as an exit condition from the DELIMITER DETECT state in the Receive State Diagram.

If the DUT receives a transition that is spaced further than 20 ns from the previous transition while in the DELMITER DETECT state, it should immediately transition into the IDLE state. However, if the DUT receives a transition that is spaced further than 20 ns from the previous transition while in any other state it should ignore the long pulse.

Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Make sure that the DUT is in the IDLE state of the Receive State Diagram.
2. Use the Traffic Generator to send a valid Manchester violation pair (mv_pair) that would cause the DUT to set detect_mv_pair=true.
3. Send a DME page containing a transition that is spaced more than 20 ns from the previous transition.
4. Send (n-1) valid DME pages.
5. Observe transmissions from the DUT.

Part b:

6. Use the Traffic Generator to send a valid DME page, in order to cause the DUT to enter the DME CAPTURE state.
7. Use the Traffic Generator to send a DME with a transition that is spaced more than 20 ns from the previous transition, and whose total length is less than page_test_max_timer, followed by two DME pages which have the same data values as the second DME transmitted. The long transition should be the last transition before the mv_pair, as the receiver should not decode any other data values after the long transition, but should remain in the DME CLOCK or DME DATA_1 state until it receives a valid mv_pair or until page_test_max_timer expires.
8. Observe transmissions from the DUT.

Observable Results:

a. The DUT should not set its Acknowledge bit after receiving a DME with a transition that is spaced more than 20 ns from the previous transition followed by (n-1) valid DME pages.
b. INFORMATIVE: The DUT should set its Acknowledge bit after receiving a valid DME followed by a DME with a long pulse, followed by two valid DME pages. This test is informative because there is not enough room between MVs to put a pulse too long and have conformant DME signaling for a minimum value of page_test_max_timer.

Possible Problems: None

Test 73.2.15 – Pulse Too Short

Purpose: To verify that the DUT properly implements pulse_too_short.

References:

[1] IEEE Std 802.3-2008, subclause 73.10.1 – State diagram variables
[2] IEEE Std 802.3-2008, Figure 73-10 – Receive State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:

Reference [1] specifies the variables used in the Auto-Negotiation Receive State Diagram, found in reference [2]. Pulse_too_short indicates that transitions separated by less than 1.6 ns have been detected. This variable is used as an exit condition from the DELIMITER DETECT state in the Receive State Diagram.

If the DUT receives a transition that is spaced less than 1.6 ns from the previous transition while in the DELMITER DETECT state, it should immediately transition into the IDLE state. However, if the DUT receives a transition that is spaced less than 1.6 ns from the previous transition while in any other state it should ignore the short pulse.

Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Make sure that the DUT is in the IDLE state of the Receive State Diagram.
2. Use the Traffic Generator to send a valid mv_pair such that the DUT sets detect_mv_pair=true.
3. Send a DME page containing a transition that is spaced less than 1.6 ns from the previous transition.
4. Send (n-1) valid DME pages.
5. Observe transmissions from the DUT.

Part b:

6. Use the Traffic Generator to send a valid DME, in order to cause the DUT to enter the DME CAPTURE state.
7. Use the Traffic Generator to send a DME with a transition that is spaced less than 1.6 ns from the previous transition, and whose total length is less than page_test_max_timer, followed by two DME pages which have the same data values as the second DME transmitted.
8. Observe transmissions from the DUT.

Observable Results:

a. The DUT should not set its Acknowledge bit following a DME with a transition that is spaced less than 1.6 ns from the previous transition followed by (n-1) valid DME pages.
b. The DUT should set its Acknowledge bit after receiving a valid DME followed by a DME with a short pulse, followed by two valid DME pages.

Possible Problems: None

Test 73.2.16 – Detect MV Pair

Purpose: To verify that the DUT properly implements detect_mv_pair.

References:

[1] IEEE Std 802.3-2008, subclause 73.10.1 – State diagram variables
[2] IEEE Std 802.3-2008, Figure 73-10 – Receive State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 21, 2009

Discussion:

Reference [1] specifies the variables used in the Auto-Negotiation Receive State Diagram, found in reference [2]. Detect_mv_pair indicates that the receiver has detected a sequence of three consecutive transitions with 12.8ns +/- 200 ps between each pair of transitions. This variable is used as an exit condition from the IDLE, DELIMITER DETECT, and DME DATA_1 states in the Receive State Diagram.

Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Use a traffic generator to send a series of (n) DME pages to send the DUT into the ACKNOWLEDGE DETECT state.
2. Observe transmissions from the DUT.

Part b:

3. Send the DUT (n) DME pages, such that the first mv_pair in the first DME page is outside the valid range.
4. Observe transmissions from the DUT.

Observable Results:

a. The DUT should set its Acknowledge bit to a logic one, indicating that detect_mv_pair=true.
b. The DUT should not set its Acknowledge bit to logic one, indicating that the DUT did not set detect_mv_pair to true upon reception of an invalid mv_pair.

Possible Problems: None

Test 73.2.17 – np_rx

Purpose: To verify that the DUT properly implements np_rx.

References:

[1] IEEE Std 802.3-2008, subclause 73.10.1 – State diagram variables
[2] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: March 23, 2011

Discussion:

Reference [1] specifies the variables used in the Auto-Negotiation Arbitration State Diagram, found in reference [2]. np_rx is a flag which holds the value of rx_link_code_word[NP] upon entry to the COMPLETE ACKNOWLEDGE state.

Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Use a traffic generator to send a series of DME base pages with the Next Page and Acknowledge bits set to put the DUT into the COMPLETE ACKNOWLEDGE state.
2. Send DME next pages with NP=0, all with proper Toggle and ACK bit values, until the DUT sets its NP=0.
3. Observe transmissions from the DUT.

Part b:

4. Use a traffic generator to send a series of DME base pages with the Next Page and Acknowledge bits set to put the DUT into the COMPLETE ACKNOWLEDGE state.
5. Send DME next pages with NP=1, all with proper Toggle and ACK bit values, until the DUT sets its NP=0.
6. Send one additional next page with NP=0, with proper Toggle and ACK bit values.
7. Observe transmissions from the DUT.

Observable Results:

a. The DUT should continue to send DME pages until both its and the link partners Next Page bits equal 0.
b. The DUT should continue to send DME pages until both its and the link partners Next Page bits equal 0.

Possible Problems: None

Test 73.2.18 – Toggle Tx

Purpose: To verify that the DUT properly implements toggle_tx.

References:

[1] IEEE Std 802.3-2008, subclause 73.10.1 – State diagram variables
[2] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: March 23, 2011

Discussion:

Reference [1] specifies the variables used in the Auto-Negotiation Arbitration State Diagram, found in reference [2]. Toggle_tx is a flag that holds the value of the DUT's toggle bit.

Test Setup: See Appendix 73.A

Test Procedure:

Parts a & b:

1. Use a traffic generator to send a series of DME base pages with the Next Page and Acknowledge bits set to send the DUT into the COMPLETE ACKNOWLEDGE state.
2. Send DME next pages with NP=1, all with proper Toggle and ACK bit values, until the DUT sets its NP=0.
3. Send one additional set of DME next page with NP=0, with proper Toggle and ACK bit values.
4. Observe transmissions from the DUT.

Observable Results:

a. The value of the toggle bit in the first Next Page should have the opposite value of bit D11 in the DUT's Base Page.
b. The value of the Toggle bit of the Next Page transmitted by the DUT's should always take the opposite value of the Toggle bit of the previous Next Page (if the previous value was a 0, it should be a 1, and vice versa).

Possible Problems: None

Test 73.2.19 – Toggle Rx

Purpose: To verify that the DUT properly implements toggle_rx.

References:

[1] IEEE Std 802.3-2008, subclause 73.10.1 – State diagram variables
[2] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: March 23, 2011

Discussion:

Reference [1] specifies the variables used in the Auto-Negotiation Arbitration State Diagram, found in reference [2]. Toggle_rx is a flag that holds the value of the link partner's toggle bit.

Test Setup: See Appendix 73.A

Test Procedure:

1. Use a traffic generator to send a series of DME base pages with the Next Page and Acknowledge bits set to send the DUT into the COMPLETE ACKNOWLEDGE state.
2. Send a series of DME pages with NP=0 with a Toggle bit equal to bit D11 in the previously transmitted Base Page.
3. Send next pages with NP=0, all with proper Toggle and ACK bit values, until the DUT sets its NP=0.
4. Observe transmissions from the DUT.

Observable Results:

a. The DUT should not set its ACK bit upon reception of pages with an improper Toggle bit value, but should remain in the NEXT PAGE WAIT state. Upon reception of pages with proper Toggle bit values the DUT should complete the Next Page exchange and begin sourcing appropriate link signaling.

Possible Problems: None

Test 73.2.20 – Single Link Ready

Purpose: To verify that the DUT properly implements single_link_ready.

References:

[1] IEEE Std 802.3-2008, subclause 73.10.1 – State diagram variables
[2] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: March 23, 2011

Discussion:

Reference [1] specifies the variables used in the Auto-Negotiation Arbitration State Diagram, found in reference [2]. Single_link_ready is a status indicating an_receive_idle=true and that either link_status_[1GKX]=OK or link_status_[10GKX4]=OK.

Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Use a traffic generator to send the DUT valid 1000BASE-KX signaling for 100 ms.
2. Observe transmissions from the DUT.
3. Repeat using valid 10GBASE-KX4 signaling

Part b:

4. Use a traffic generator to send the DUT valid 1000BASE-KX signaling for 20 ms.
5. Observe transmissions from the DUT.
6. Repeat using valid 10GBASE-KX4 signaling

Observable Results:

a. The DUT should set single_link_ready=true and enter the AN GOOD CHECK state.
b. The DUT should set single_link_ready=false, enter the PARALLEL DETECTION FAULT and ABILITY DETECT states, and restart DME transmissions.

Possible Problems: None

GROUP 3: ARBITRATION STATE DIAGRAM

Overview:

The tests defined in this section cover operation specific to the reception of differential Manchester encoded (DME) pages used in Backplane Ethernet Auto-Negotiation defined in Clause 73 of IEEE 802.3-2008.

These tests are designed to verify that the DUT properly transitions through the states in the Arbitration state diagram, which is defined in Figure 73-11.

Test 73.3.1 – ABILITY DETECT

Purpose: To verify that the DUT properly exits the ABILITY DETECT state under the appropriate conditions.

References:

[1] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

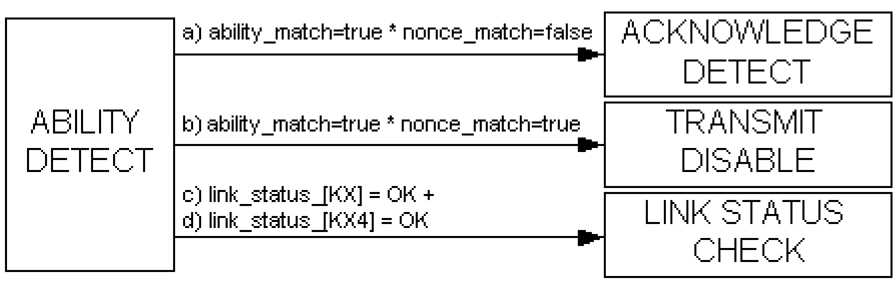
Resource Requirements: See Appendix 73.A

Last Modification: April 22, 2009

Discussion:

An Auto-Negotiating device will enter the ACKNOWLEDGE DETECT state only after ability_match=TRUE and nonce_match=FALSE. Ability_match requires that it receives at least 3 complete, consecutive and consistent Link code words from its link partner, ignoring the Acknowledge bit. Nonce_match indicates whether the received Transmitted Nonce value matches the transmitted one. Conversely, when ability_match=TRUE and nonce_match=TRUE the DUT should enter the TRANSMIT DISABLE state.

Upon reception of 1000BASE-KX or 10GBASE-KX4 link signaling the DUT should enter the LINK STATUS CHECK state, which is part of the parallel detection function.



Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Send (n) DME pages to the DUT all with the Acknowledge bit not set and the Transmitted Nonce Field not equal to the DUTs Transmitted Nonce field.
2. Observe transmissions from the DUT.

Part b:

3. Use the traffic generator to send (n) DME pages, all with a Transmitted Nonce value that is equal to the Transmitted Nonce value used by the DUT.
4. Observe transmissions from the DUT.

Part c:

5. Send the DUT valid 1000BASE-KX signaling.
6. Observe transmissions from the DUT.

Part d:

7. Send the DUT valid 10GBASE-KX4 signaling.
8. Observe transmissions from the DUT.

Observable Results:

a. The DUT should enter the ACKNOWLEDE DETECT state and begin sending DME pages with the Acknowledge bit set.
b. The DUT should enter the TRANSMIT DISABLE state and cease all transmissions.
c. The DUT should enter the LINK STATUS CHECK state and cease DME page transmissions.
d. The DUT should enter the LINK STATUS CHECK state and cease DME page transmissions.

Possible Problems: None

Test 73.3.2 – TRANSMIT DISABLE

Purpose: To verify that the DUT properly exits the TRANSMIT DISABLE state under the appropriate condition.

References:

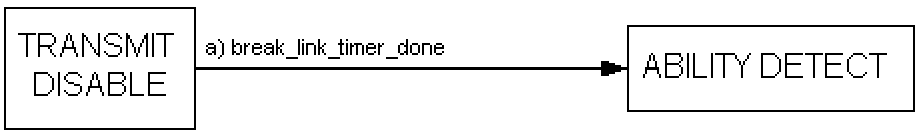
[1] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 22, 2009

Discussion:

Once a device has entered the TRANSMIT DISABLE state, it must cease all signaling for a period of break_link_timer before it restarts the Auto-Negotiation process.



Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Send the DUT (n) DME pages with the Acknowledge bit not set, and with a Transmitted Nonce field equal to the Transmitted Nonce field of the DUT to cause the DUT to enter the TRANSMIT DISABLE state.
2. Observe transmissions from the DUT.

Observable Results:

a. The DUT should enter the ABILITY DETECT state after break_link_timer_done equals true.

Possible Problems: None

Test 73.3.3 – ACKNOWLEDGE DETECT

Purpose: To verify that the DUT properly exits the ACKNOWLEDGE DETECT state under the appropriate conditions.

References:

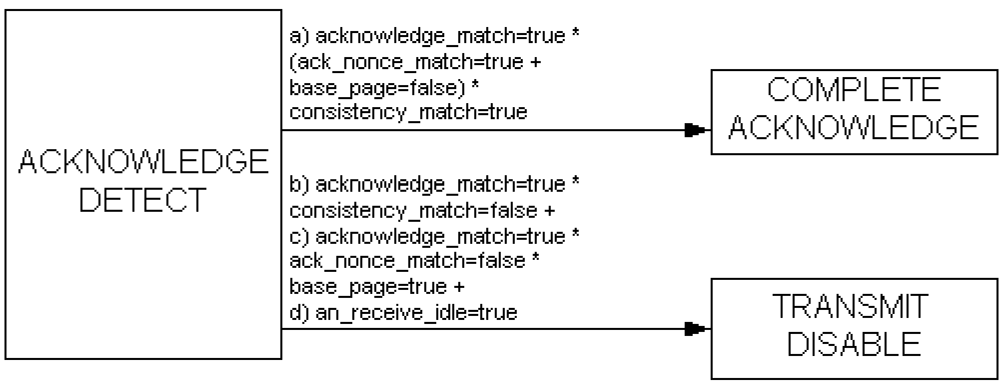
[1] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 22, 2009

Discussion:

Once an Auto-Negotiating device has entered the ACKNOWLEDGE DETECT state, it must receive an additional three DME pages with the Acknowledge bit set and having an Echoed Nonce field equal to the Nonce field transmitted by the local device but otherwise identical to the first set of pages received, before entering the COMPLETE ACKNOWLEDGE state. If any of these conditions are not met, or if the DUT ceases to receive DME pages then the DUT should enter the TRANSMIT DISABLE state and attempt a renegotiation.



Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Send the DUT (n) DME pages with the Acknowledge bit not set, followed by (m) DME pages with the Acknowledge bit set, the Echoed Nonce field set to the value of the DUTs Transmitted Nonce field, but otherwise identical to the initial DME pages.
2. Observe transmissions from the DUT.

Part b:

3. Send the DUT (n) DME pages with the Acknowledge bit not set, followed by (m) DME pages with the Acknowledge bit set, but are one bit different from the first group. All pages in the second group should be the identical.
4. Observe transmissions from the DUT.
5. Repeat steps 3 and 4 using all one bit different DME pages.

Part c:

6. Send the DUT (n) DME pages with the Acknowledge bit not set, followed by (m) DME pages with the Acknowledge bit set, the Echoed Nonce field set to any value other than the value of the DUTs Transmitted Nonce field, but otherwise identical to the initial DME pages.
7. Observe transmissions from the DUT.

Part d:

8. Send the DUT (n) DME pages with the Acknowledge bit not set to put the DUT into the ACKNOWLEDGE DETECT state.
9. Observe transmissions from the DUT.

Observable Results:

a. The DUT should enter the COMPLETE ACKNOWLEDGE state and send 6 to 8 more DME pages with the ACK bit set before transmitting valid link signaling.
b. The DUT should enter the TRANSMIT DISABLE state and cease all transmissions.
c. The DUT should enter the TRANSMIT DISABLE state and cease all transmissions.
d. The DUT should enter the TRANSMIT DISABLE state and cease all transmissions after page_test_max_timer expires after the last DME page was sent to the device.

Possible Problems: None

Test 73.3.4 – COMPLETE ACKNOWLEDGE

Purpose: To verify that the DUT properly exits the COMPLETE ACKNOWLEDGE state under the appropriate conditions.

References:

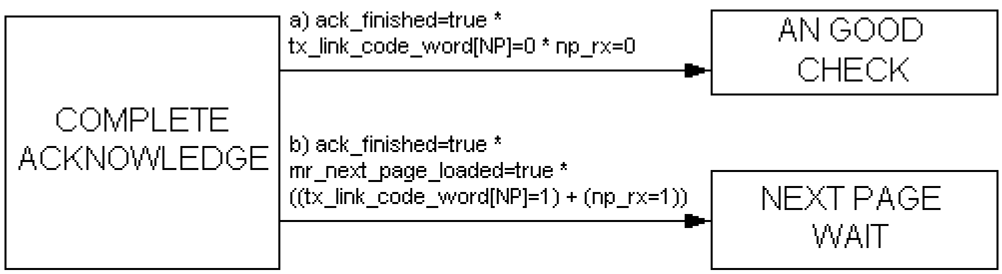
[1] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 22, 2009

Discussion:

Once an Auto-Negotiating device has entered the COMPLETE ACKNOWLEDGE state and neither it nor the link partner desires to send additional pages via a Next Page exchange it should send out 6 to 8 (inclusive) more DME pages containing its Link Codeword and then enter the AN GOOD CHECK state where it will begin to source link signaling. If either the local device or link partner desires a Next Page exchange then the local device should still send out 6 to 8 more DME pages before entering into the NEXT PAGE WAIT state and transmitting its first Next Page.



Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Send the DUT a series of (n) DME pages with the Acknowledge bit not set, followed by (m) DME pages with the Acknowledge bit set to put the DUT into the COMPLETE ACKNOWLEDGE state.
2. If the DUTs Next Page bit is set, send the DUT Null Message pages until the DUT has finished transmitting all of its Next Pages.
3. Observe transmissions from the DUT.

Part b:

4. Send the DUT enough valid DME pages with the Next Page and ACK bits set to put the DUT into the COMPLETE ACKNOWLEDGE state.
5. Observe transmissions from the DUT.

Observable Results:

a. The DUT should enter the AN GOOD CHECK state and begin transmitting valid link signaling.
b. The DUT should enter the NEXT PAGE WAIT state and begin sending its first Next Page.

Possible Problems: None

Test 73.3.5 – AN GOOD CHECK

Purpose: To verify that the DUT properly exits the AN GOOD CHECK state under the appropriate conditions.

References:

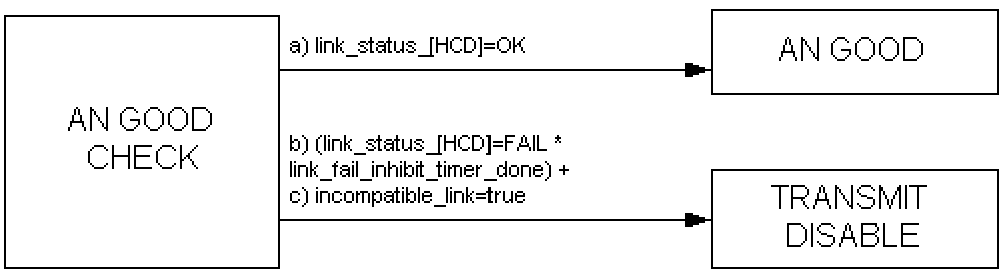
[1] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram
[2] IEEE Std 802.3-2008, Table 73-5 – Priority Resolution

Resource Requirements: See Appendix 73.A

Last Modification: April 22, 2009

Discussion:

Once an Auto-Negotiating device has entered the AN GOOD CHECK state it should disable DME page transmissions and attempt to establish a link at the highest common denominator (HCD) technology, as determined by the Priority Resolution function [2]. If a link is not established by the time link_fail_inhibit_timer is done or if there is no HCD technology the DUT should enter the TRANSMIT DISABLE state and attempt a renegotiation.



Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Send the DUT a series of (n) DME pages with the Acknowledge bit not set, followed by (m) DME pages with the Acknowledge bit set to put the DUT into the COMPLETE ACKNOWLEDGE state. If necessary, send the DUT as many Null Message pages as needed to put the DUT into the AN GOOD CHECK state.
2. Send the DUT valid link signaling according to the DUT and Traffic Generator's HCD.
3. Observe transmissions from the DUT.

Part b:

4. Repeat step 1.
5. Observe transmissions from the DUT.

Part c:

6. Repeat step 1, transmitting DME pages with the Technology Ability field bits cleared.
7. Observe transmissions from the DUT.

Observable Results:

a. The DUT should enter the AN GOOD state and establish a valid link.
b. The DUT should enter the TRANSMIT DISABLE state and cease all transmissions.
c. The DUT should enter the TRANSMIT DISABLE state and cease all transmissions.

Possible Problems: None

Test 73.3.6 – NEXT PAGE WAIT

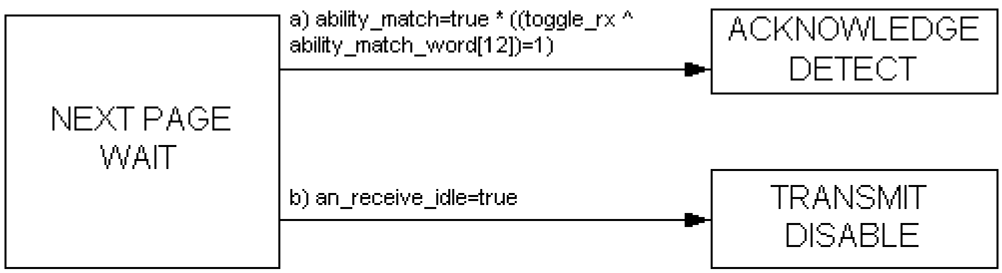
Purpose: To verify that the DUT properly exits the NEXT PAGE WAIT state under the appropriate conditions.

References:
[1] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 22, 2009

Discussion:
Once an Auto-Negotiating device has entered the NEXT PAGE WAIT state it enter the ACKNOWLEDGE DETECT state after it receives enough DME pages with a valid Toggle bit value to set ability_match=TRUE. Otherwise, if it ceases to receive DME pages it will enter the TRANSMIT DISABLE state.



Test Setup: See Appendix 73.A

Test Procedure:
Part a:
1. Send the DUT enough valid DME pages with the Next Page bit set, to put it into the NEXT PAGE WAIT state.
2. Send the DUT (n) Next Pages with a valid Toggle bit value.
3. Observe transmissions from the DUT.

Part b:
4. Repeat step 1.
5. Observe transmissions from the DUT.

Observable Results:
a. The DUT should enter the ACKNOWLEDGE DETECT state and begin sending DME pages with the Acknowledge bit set.
b. The DUT should enter the TRANSMIT DISABLE state and cease all transmissions.

Possible Problems: None

Test 73.3.7 – LINK STATUS CHECK

Purpose: To verify that the DUT properly exits the LINK STATUS CHECK state under the appropriate conditions.

References:

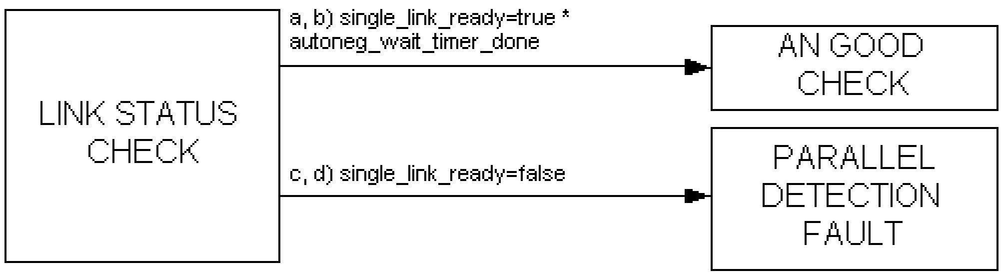
[1] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 22, 2009

Discussion:

The LINK STATUS CHECK state is an integral part of the Parallel Detection function. Once in the LINK STATUS CHECK state the local device should disable DME page transmissions and determine whether single_link_ready is true. If single_link_ready remains true for the duration of autoneg_wait_timer the DUT should transition into the AN GOOD CHECK state and begin sending appropriate link signaling. However, if single_link_ready becomes false the DUT should enter the PARALLEL DETCTION FAULT state and then the ABILITY DETECT state and restart sending DME pages.



Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Send the DUT valid 1000BASE-KX signaling for at least 50 ms.
2. Observe transmissions from the DUT.

Part b:

3. Repeat steps 1-2, using 10GBASE-KX4 signaling.

Part c:

4. Send the DUT valid 1000BASE-KX signaling for less than 20 ms.
5. Observe transmissions from the DUT.

Part d:

6. Repeat steps 4-5, using 10GBASE-KX4 signaling.

Observable Results:

a. The DUT should enter the AN GOOD CHECK state and begin sending 1000BASE-KX link signaling.
b. The DUT should enter the AN GOOD CHECK state and begin sending 10GBASE-KX4 link signaling.
c. The DUT should enter the ABILITY DETECT state via the PARALLEL DETECTION FAULT state and restart sending base pages.
d. The DUT should enter the ABILITY DETECT state via the PARALLEL DETECTION FAULT state and restart sending base pages.

Possible Problems: None

Test 73.3.8 – AN GOOD

Purpose: To verify that the DUT properly exits the AN GOOD state under the appropriate condition.

References:

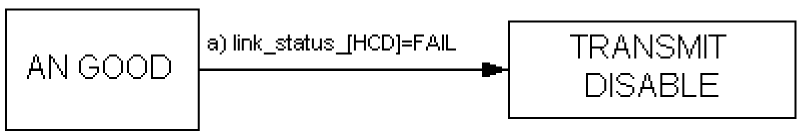
[1] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 22, 2009

Discussion:

Once the Auto-Negotiation process has been completed, the local device should remain in the AN GOOD state for the duration of the link. If the link is dropped for any reason the local device should enter the TRANSMIT DISABLE state before attempting a renegotiation.



Test Setup: See Appendix 73.A

Test Procedure:

1. Send the DUT enough DME pages and valid link signaling to put the DUT into the AN GOOD state.
2. Stop sending the DUT valid signaling.
3. Observe transmissions from the DUT.

Observable Results:

a. The DUT should enter the TRANSMIT DISABLE state and cease all transmissions.

Possible Problems: None

GROUP 4: RECEIVE STATE DIAGRAM

Overview:

The tests defined in this section cover Auto-Negotiation operation specific to reception of DME pages defined in Clause 73 of IEEE 802.3-2008.

These tests are designed to verify that the DUT properly transitions through the states in the Receive state diagram, which is defined in Figure 73-10.

Test 73.4.1 – IDLE

Purpose: To verify that the DUT properly exits the IDLE state under the appropriate condition only.

References:

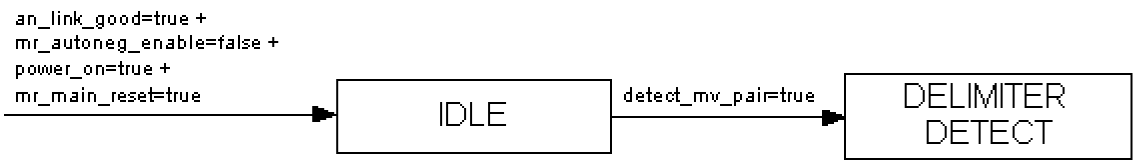
[1] IEEE Std 802.3-2008, Figure 73-10 – Receive state diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 22, 2009

Discussion:

The local device's receiver should remain in the IDLE state unless a valid mv_pair is received. Upon reception of a valid mv_pair, the receiver should enter the DELIMITER DETECT state.



Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Send the DUT (n) valid DME pages.
2. Observe transmissions from the DUT.

Part b:

3. Send the DUT (n) DME pages, omitting the first mv_pair.
4. Observe transmissions from the DUT.

Observable Results:

a. The DUT should enter the ACKNOWLEDGE DETECT state and begin sourcing DME pages with the ACK bit set.
b. The DUT should remain in the ABILITY DETECT state and continue to send its base page without the ACK bit set.

Possible Problems: None

Test 73.4.2 – DELIMITER DETECT

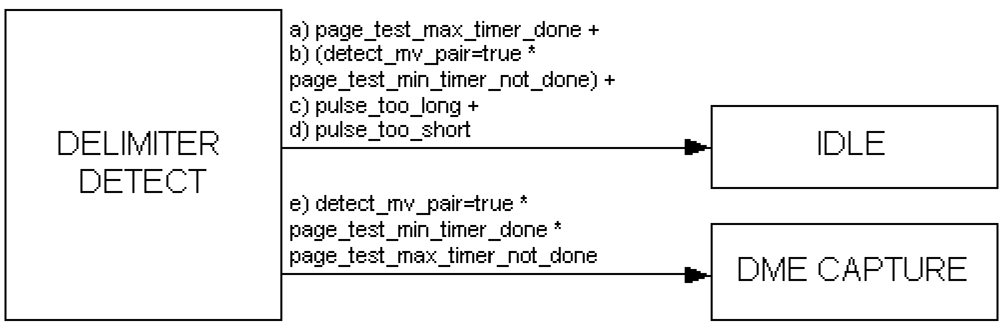
Purpose: To verify that the DUT properly exits the DELMITER DETECT state under the appropriate conditions only.

References:
[1] IEEE Std 802.3-2008, Figure 73-10 – Receive state diagram

Resource Requirements: See Appendix 73.A

Last Modification: March 23, 2011

Discussion:
Once the local device's receiver has entered the DELIMITER DETECT state it should not record any of the information in the received DME page, instead only looking to see that another mv_pair is received after a valid period of time. If, in this state, it receives a transition, which is longer or shorter than the conformant ranges, it should return to the IDLE state. Similarly, if the DUT detects an mv_pair before page_test_min_timer is done, or if page_test_max_timer is done, the receiver should return to the IDLE state.



Test Setup: See Appendix 73.A

Test Procedure:
Part a:
1. Make sure the DUT is in the IDLE state of the Receive state diagram.
2. Send the DUT two mv_pairs separated by more than page_test_max_timer. The mv_pairs should have valid Manchester encoding between them so as not to skew the receiver.
3. Send the DUT (n-1) valid DME pages.
4. Observe transmission from the DUT.

Part b:
5. Send the DUT two mv_pairs separated by less than page_test_min_timer. The mv_pairs should have valid Manchester encoding between them so as not to skew the receiver.
6. Send the DUT (n-1) valid DME pages.
7. Observe transmissions from the DUT.

Part c:
8. Send a Manchester violation followed by a DME page containing a transition that is spaced more than 20 ns from the previous transition.
9. Send the DUT (n-1) valid DME pages.
10. Observe transmissions from the DUT.

Part d:

11. Send a Manchester violation followed by a DME page containing a transition that is spaced less than 1.6 ns from the previous transition.
12. Send the DUT (n-1) valid DME pages.
13. Observe transmissions from the DUT.

Part e:

14. Send the DUT (n) valid DME pages.
15. Observe transmissions from the DUT.

Observable Results:

a. The DUT should remain in the ABILITY DETECT state and continue to send its base page without the ACK bit set.
b. INFORMATIVE: The DUT should remain in the ABILITY DETECT state and continue to send its base page without the ACK bit set. This waveform violates the electrical timing requirements.
c. The DUT should remain in the ABILITY DETECT state and continue to send its base page without the ACK bit set.
d. The DUT should remain in the ABILITY DETECT state and continue to send its base page without the ACK bit set.
e. The DUT should enter the ACKNOWLEDGE DETECT state and begin sourcing DME pages with the ACK bit set.

Possible Problems: None

Test 73.4.3 – DME CLOCK

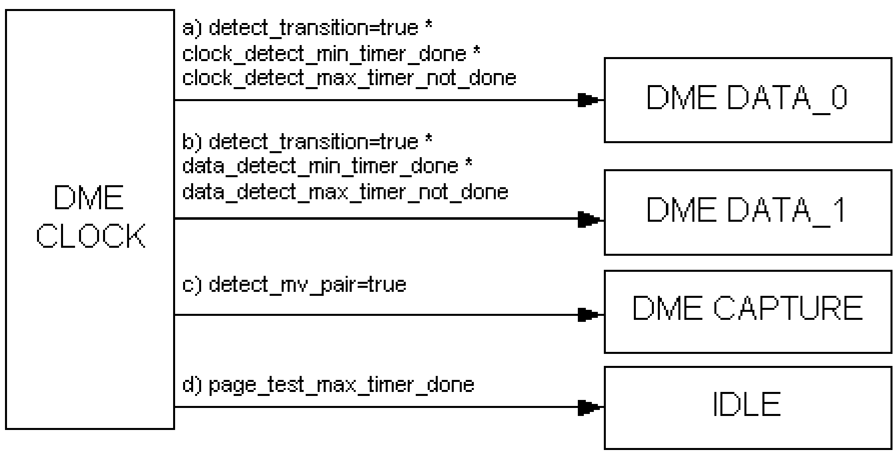
Purpose: To verify that the DUT properly exits the DME CLOCK state under the appropriate conditions only.

References:
[1] IEEE Std 802.3-2008, Figure 73-10 – Receive state diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 22, 2009

Discussion:
The local device's receive state diagram will enter the DME CLOCK state any time it receives a valid clock transition or mv_pair (after passing through the DELIMITER DETECT state). If the DUT receives another valid clock transition, it should enter the DME DATA_0 state and then return to the DME CLOCK state. Similarly, if it receives a valid data transition, it should enter the DME DATA_1 state. If it receives a valid mv_pair, indicating the end of a page, it should enter the DME CAPTURE state and then return to the DME CLOCK state. Finally, if page_test_max_timer is done the DUT should return to the IDLE state, regardless of whether or not it is still receiving valid data.



Test Setup: See Appendix 73.A

Test Procedure:
Part a:
1. Send the DUT a valid DME page followed by a Manchester violation delimiter to cause it to enter the DME CLOCK state.
2. Send the DUT a valid DME page, with the exception that the first data 0 shall have a clock to clock transition spaced greater than clock_detect_max_timer apart.
3. Send the DUT (n-2) valid DME pages.
4. Observe transmissions from the DUT.

Part b:
5. Send the DUT a valid DME page followed by a Manchester violation delimiter to cause it to enter the DME CLOCK state.
6. Send the DUT a valid DME page, such that the first data pulse is between data_detect_min_timer and data_detect_max_timer.
7. Send the DUT (n-2) valid DME pages.
8. Observe transmissions from the DUT.

Part c:

9. Send the DUT a valid DME page followed by a Manchester violation delimiterto cause it to enter the DME CLOCK state.
10. Send the DUT a DME page which contains only 10 data bits, but which is otherwise properly formatted.
11. Send the DUT (n-1) valid DME pages. The data bits of these pages should match the first 10 data bits sent by the previous page; the other data bits should all be zero.
12. Observe transmissions from the DUT.

Part d:

13. Send the DUT a two valid DME pages.
14. Cease transmissions for at least page_test_max_timer.
15. Send the DUT two more valid DME pages, which match the first two DME pages transmitted.
16. Observed transmissions from the DUT.

Observable Results:

a. The DUT should remain in the ABILITY DETECT state and continue to send its Base page without the Acknowledge bit set.
b. The DUT should enter the ACKNOWLEDGE DETECT state and begin sending its Base page with the Acknowledge bit set.
c. INFORMATIVE: The DUT may enter the ACKNOWLEDGE DETECT state and begin sending its Base page with the Acknowledge bit set.
d. The DUT should remain in the ABILITY DETECT state and continue to send its Base page without the Acknowledge bit set.

Possible Problems: None

Test 73.4.4 – DME DATA_1

Purpose: To verify that the DUT properly exits the DME DATA_1 state under the appropriate conditions only.

References:

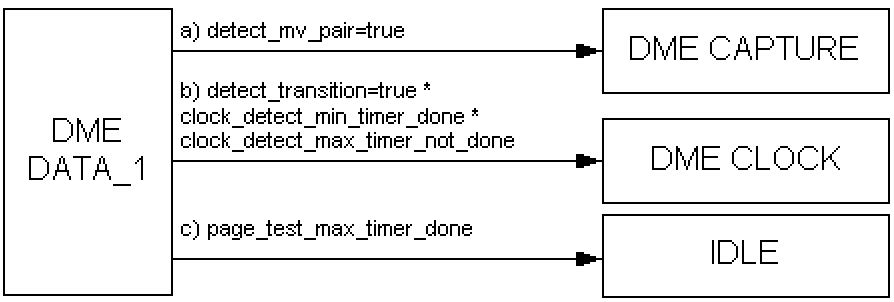
[1] IEEE Std 802.3-2008, Figure 73-10 – Receive state diagram

Resource Requirements: See Appendix 73.A

Last Modification: March 23, 2011

Discussion:

The local device's receive state diagram will enter the DME DATA_1 state any time it receives a valid data transition. If it receives a valid clock transition, it should enter the DME CLOCK state. If it receives a valid mv_pair it should enter the DME CAPTURE state and then return to the DME CLOCK state, however, this transition should not occur under normal circumstances. Finally, if page_test_max_timer is done the DUT should return to the IDLE state, regardless of whether or not it is still receiving valid data.



Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Send the DUT a valid DME page.
2. Send the DUT a DME page, with the last data bit being a 1 to cause the DUT to enter the DME_DATA_1 state omitting the last clock transition before the Manchester Violation.
3. Send the DUT (n-2) valid DME pages which match the second DME page transmitted.
4. Observe transmissions from the DUT.

Part b:

5. Send the DUT a valid DME page.
6. Send the DUT a valid DME page, with the exception that the clock transition after the first data 1 shall be spaced more than clock_detect_max_timer apart than the previous clock transition.
7. Send (n-2) valid DME pages.
8. Observe transmissions from the DUT.

Part c:

9. Send the DUT a valid DME page.
10. Send the DUT a DME page, with the last bit being a 1 to cause the DUT to enter the DME_DATA_1 state, followed by a gap, such that the total length of the DME page + the gap is at least page_test_max_timer.
11. Send (n-2) valid DME pages.
12. Observe transmissions from the DUT.

Observable Results:

a. The DUT should enter the ACKNOWLEDGE DETECT state and begin sending its Base page with the Acknowledge bit set.
b. The DUT should remain in the ABILITY DETECT state and continue to send its Base page without the Acknowledge bit set.
c. The DUT should remain in the ABILITY DETECT state and continue to send its Base page without the Acknowledge bit set.

Possible Problems: None

GROUP 5: FUNCTIONALITY

Overview:

The tests defined in this section cover operation specific to the reception of differential Manchester encoded (DME) Base Pages used in Backplane Ethernet Auto-Negotiation defined in Clause 73 of IEEE 802.3-2008.

These tests are designed to test various aspects of the DUTs Auto-Negotiation functionality.

Test 73.5.1 – Parallel Detection Function

Purpose: To verify that the DUT properly implements the Parallel Detection function.

References:

[1] IEEE Std 802.3-2008, subclause 73.7.4.1 – Parallel Detection function

Resource Requirements: See Appendix 73.A

Last Modification: April 23, 2009

Discussion:

The Parallel Detection function is designed to allow detection of link partners that are sending 1000BASE-KX or 10GBASE-KX4 signaling. Upon detection of 1000BASE-KX or 10GBASE-KX4 signaling the local device should enter the LINK STATUS CHECK state and start the autoneg_wait_timer. If single_link_ready is true when the autoneg_wait_timer expires then the local device should enter the AN GOOD CHECK state and enable link signaling from the PMA that caused single_link_ready to be true.

Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Configure a link partner to send the DUT valid 1000BASE-KX signaling and establish a link.
2. Send and receive valid 1000BASE-KX frames.

Part b:

3. Configure a link partner to send the DUT valid 10GBASE-KX4 signaling and establish a link.
4. Send and receive valid 10GBASE-KX4 frames.

Observable Results:

a. The DUT should send and receive valid 1000BASE-KX frames after establishing a link through parallel detection.
b. The DUT should send and receive valid 10GBASE-KX4 frames after establishing a link through parallel detection.

Possible Problems: None

Test 73.5.2 – Renegotiation Function

Purpose: To verify that the DUT properly implements the Renegotiation function.

References:

[1] IEEE Std 802.3-2008, subclause 73.7.5 – Renegotiation function

Resource Requirements: See Appendix 73.A

Last Modification: April 23, 2009

Discussion:

The Renegotiation function allows the local device to restart Auto-Negotiation and attempt to establish a new link after a valid link has been interrupted. Renegotiation can occur either when link_status=FAIL or if the local device receives a request for renegotiation from any entity, such as a management entity.

Test Setup: See Appendix 73.A

Test Procedure:

1. Configure a link partner to send the DUT valid 1000BASE-KX signaling and establish a link.
2. Send and receive valid 1000BASE-KX frames.
3. Break the link.
4. Observe transmissions from the DUT.

Observable Results:

a. Upon reception of the link_status=FAIL message, the DUT should disable all transmission for approximately break_link_timer and restart Auto-Negotiation.

Possible Problems: None

Test 73.5.3 – Priority Resolution Function

Purpose: To verify that the device under test properly configures the highest common denominator (HCD) technology for the transmitted technologies in a link code word.

References:

[1] IEEE Std 802.3-2008, subclause 73.7.6 – Priority Resolution function

Resource Requirements: See Appendix 73.A

Last Modification: April 23, 2009

Discussion:

Once a station has received its link partner's link code word and completed the exchange of DME pages, the technology at which communication is to be established must be resolved. Through the Priority Resolution function, the highest common denominator (HCD) technology should be found. Upon reception of a non-Ethernet selector field, it is recommended that a device disable all Ethernet PMAs, as it could cause problems with a non-Ethernet link partner. This test is designed to verify that the device under test resolves the proper HCD for all possible technology and selector field combinations

Test Setup: See Appendix 73.A

Test Procedure:

Part a: Ethernet Selector Field Resolution

1. Use the Traffic Generator to send a series of DME pages that advertise a set of abilities with an Ethernet selector field.
2. Verify that the DUT establishes a link when it can, and refuses a link otherwise.
3. Verify that the DUT resolved the highest common technology by sending frames to the DUT in that format and determining whether they were received.
4. Repeat this procedure for all possible combinations of the first three bits of the technology ability field.

Part b: Non-Ethernet Selector Field Resolution

5. Repeat steps 1-4 with DME pages advertising selector fields other than 802.3.

Observable Results:

a. In every case, the DUT should resolve a link at the highest priority possible based on the priority resolution function for the technologies advertised.
b. INFORMATIVE: The DUT should enter the AN GOOD CHECK state and may disable all non-HCD Ethernet PMDs.

Possible Problems: When a 10GBASE-KR link has been established, a training procedure begins to optimize transmit and receive filter coefficients. If this process is unable to complete, then a frame will be unable to be transmitted, and link status will be indicated by the presence or absence of valid 10GBASE-KR signaling.

Test 73.5.4 – Remote Fault

Purpose: To verify that the DUT properly implements the Remote Fault function.

References:

[1] IEEE Std 802.3-2008, subclause 73.6.7 – Remote Fault
[2] IEEE Std 802.3-2008, subclause 73.7.4.1 – Parallel Detection function
[3] IEEE Std 802.3-2008, subclause 28.2.3.5 – Remote fault sensing function

Resource Requirements: See Appendix 73.A

Last Modification: April 23, 2009

Discussion:

The Remote Fault bit is used to indicate that a fault has occurred on the far side. From reference [2], a device may signal a Remote Fault if more than one technology-dependent PHY indicates link_status=OK when the autoneg_wait_timer expires during Parallel Detection. There is currently no defined method in Clause 73 that indicates when the transmitted Remote Fault bit should be reset, thus after setting the Remote Fault bit a device may set it continuously. Alternatively, a device may use the definition from [3], which states that the Remote Fault bit shall remain set until after a successful negotiation has occurred, at which time the Remote Fault bit shall be reset.

Test Setup: See Appendix 73.A

Test Procedure:

Part a: Multiple technologies indicating link_status=OK

1. Send the DUT valid 1000BASE-KX and 10GBASE-KX4 signaling for at least 100 ms.
2. Observe transmissions from the DUT.

Part b: Transmitted Remote Fault bit

3. Observe transmissions from the DUT.
4. If the DUT is sending DME pages with the Remote Fault bit set, send enough validly formed DME pages which will cause the DUT to enter the AN GOOD CHECK state.
5. Observe transmissions from the DUT.

Observable Results:

a. INFORMATIVE: The DUT may set its Remote Fault bit upon reception of 1000BASE-KX and 10GBASE-KX4 signaling at the same time.
b. INFORMATIVE: The DUT may reset its Remote Fault bit upon renegotiation.

Possible Problems: If there are no means to cause the DUT to transmit DME pages with the Remote Fault bit set part b cannot be performed.

Test 73.5.5 – Selector Field Combinations

Purpose: To verify that the DUT accepts DME pages with the Selector Field set to a reserved or undefined combination.

References:

[1] IEEE Std 802.3-2008, subclause 73.6.1 Selector Field
[2] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 23, 2009

Discussion:

A local device should only transmit an IEEE 802.3 Selector Field, however, there are no defined actions to take when receiving a non-802.3 Selector Field. Therefore, as long as complete and consistent Link code words are received, a device should accept them as valid regardless of the Selector Field combination. This test is designed to verify that the DUT will accept Link code words with the Selector Field set to reserved and undefined combinations.

Other valid Selector Field combinations that have been specified are: Isochronous Ethernet (01000), IEEE-802.5 Token Ring (11000), and IEEE 1394 (00100).

Test Setup: See Appendix 73.A

Test Procedure:

Part a: ACKNOWLEDGE DETECT with different Selector Field combinations

1. Use the Traffic Generator to send (n) DME pages with the Acknowledge bit not set, and with Selector Field combination of 00000b.
2. Observe Transmissions from the DUT.
3. Repeat Steps 1-2 using different Selector Field combinations such as: 11000b, 11111b, 01000b, and 00100b.

Part b: COMPLETE ACKNOWLEDGE with different Selector Field combinations

4. Use the Traffic Generator to send (n) DME pages with the Acknowledge bit not set, followed by (m) DME pages with the Acknowledge bit set, all with a Selector Field of 00000b.
5. Observe Transmissions from the DUT.
6. Repeat Steps 1-2 using different Selector Field combinations such as: 11000b, 11111b, 01000b, and 00100b.

Observable Results:

a. The DUT should enter the ACKNOWLEDGE DETECT state.
b. The DUT should enter the COMPLETE ACKNOWLEDGE state and may or may not attempt to establish a link.

Possible Problems: None

Test 73.5.6 – Technology Ability Field Bits

Purpose: To verify that the DUT accepts DMEs with different combinations of the Technology Ability Field bits set.

References:

[1] IEEE Std 802.3-2008, subclause 73.6.4 – Technology Ability Field
[2] IEEE Std 802.3-2008, Figure 73-11 – Arbitration State Diagram

Resource Requirements: See Appendix 73.A

Last Modification: April 23, 2009

Discussion:

The Technology Ability Field contains 24 bits that indicate supported technologies specific to the Selector Field value. Bits A0:A2 are encoded as 1000BASE-KX, 10GBASE-KX4, and 10GBASE-KR, respectively. The last 21 bits of the Technology Ability Field are currently reserved for future use. A conformant device should be able to receive any of the Technology Ability Field bits set without affecting normal operation. This test is designed to verify that the DUT will accept DME pages with various combinations of the Technology Ability Field bits set.

Test Setup: See Appendix 73.A

Test Procedure:

Part a:

1. Send the DUT (n) DME pages with the Acknowledge bit not set and bit A0 set.
2. Observe transmissions from the DUT.
3. Repeat steps 1-2 but send DME pages encoded with all possible one-bit differences in the Technology Ability Field.

Part b:

4. Send the DUT (n) DME pages with the Acknowledge bit not set, followed by (m) DME pages with the Acknowledge bit set, all with a bit A0 set.
5. Observe transmissions from the DUT.
6. Repeat steps 4-5 but send DME pages encoded with all possible one-bit differences in the Technology Ability Field.

Observable Results:

a. The DUT should enter the ACKNOWLEDGE DETECT state regardless of the received DME pages encoded with varying Technology Ability Field bits.
b. The DUT should enter the COMPLETE ACKNOWLEDGE state regardless of the received DME pages encoded with varying Technology Ability Field bits.

Possible Problems: None

APPENDICES

Overview:

Test suite appendices are intended to provide additional low-level technical detail pertinent to specific tests contained in this test suite. These appendices often cover topics that are outside of the scope of the standard, and are specific to the methodologies used for performing the measurements in this test suite. Appendix topics may also include discussion regarding a specific interpretation of the standard (for the purposes of this test suite), for cases where a particular specification may appear unclear or otherwise open to multiple interpretations.

Scope:

Test suite appendices are considered informative supplements, and pertain solely to the test definitions and procedures contained in this test suite.

Appendix 73.A – Test Fixtures and Setups

Purpose: To specify the measurement hardware, test fixtures, and setups used in this test suite

References:

[1] IEEE Std 802.3-2008, subclause 72.7.1.1 – Test Fixture

Last Modification: March 10, 2009

Discussion:

The reader will need the following in order to perform tests in Group 1 and 2.

1. Digital Storage Oscilloscope, 1 GHz bandwidth (minimum)
2. Transmitter Test Fixture (dependent upon the DUT's interface)
3. Clause 73 capable configurable transmitting station.
4. Post Processing capabilities (UNH-IOL custom Matlab based platform, or equivalent)

To perform tests in group 1, the post processing must be capable of determining the exact time and amplitude of the crossing times in a DME page. The user may choose to implement this in any way they choose. Timers are verified by observing waveforms on the oscilloscope.

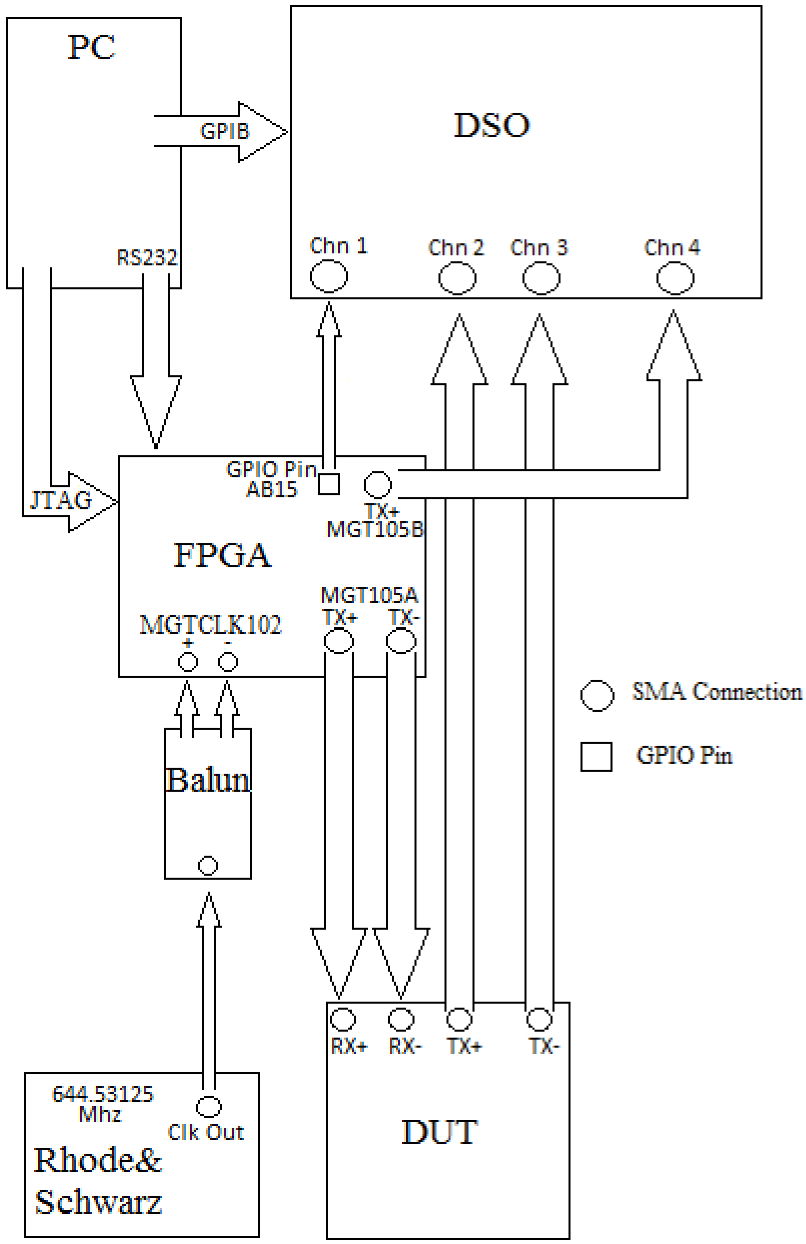


Figure 73.A-1: Test Setup used to perform Clause 73 Base Page Auto-Negotiation

To perform the tests in group 2-5, the post-processing must be capable of decoding DME pages sent from the DUT and the traffic generator.