EtherNet/IP™ - ODVA Conformance Test Results

Test Information	
Scheduled Test Date	
Composite Test Revision	**CT13**
ODVA File Number	
Test Type	**Select from list**

Vendor Information	
Vendor Name	**DO NOT ENTER - AUTOMATED FIELD**

Device Information

Device Information from Identity Object Instance* 1

For multiple identity object instances, additional Device Information tables are inserted into the report.

Identity Object	Attribute	Value		
Attribute 1	Vendor ID (decimal)	**0**		
Attribute 2	Device Type (hex)	**0x**		
Not an Attribute	Device Profile Name	**Enter Profile Code as a Hex Number above**		
Attribute 4	Product Revision (decimal)	Major Rev		Minor rev
Identity Object	**Attribute**	**Value for Device 1**		**Value for Device 2**
Attribute 3	Product Code (decimal)			**N/A**
Attribute 7	Product Name			**N/A**

*For multiple instances, additional Device Information tables should be inserted into the report.

TSP Information	
TSP Location	Select TSP Location
Engineer Initials or Name	
Completion Date	
Test Result	**ENTER SUGGESTED RESULT**
All advisories, warnings, and failures are summarized and described in Table 1 below.	

EtherNet/IP™ - ODVA Conformance Test Results

Table 1 Conformance Failures and Advisories

NOTE: **Advisories** indicate recommendations, W**arnings** indicate behavior that may be required to be changed before subsequent tests as indicated in Warning description, and F**ailures** must be resolved to pass

Index	Test Item	Advisories and Failures: Observed DUT Behavior	Required Behavior & Specification Reference
1			
2			
3			
4			
5			
6			
7			
8			

EtherNet/IP™ Product Family Members

The products listed in this document are regarded as the same family due to their following properties:

Identical CIP and CIP Network Adapatation Firmware	Y/N ?
All physical layer variants submitted for test	Y/N ?
All CIP functionality implemented across family members submitted for test	Y/N ?

They differ due to the following properties:

Product Code and/or Product Name	Y/N ?
Physical layer implementation	Y/N ?

No.	Product Code (Attribute 3)	Product Name (Attribute 7)	Product Revision (Attribute 4)	SOC File Name
1	0	0	0	Product1.stc
2	N/A	N/A	N/A	N/A
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				

Tested Products

EtherNet/IP™ Device Under Test

SOC Screenshot

Implemented Objects

Physical Data Screenshot

Timers Screenshot

Conformance Test Screenshot

EtherNet/IP™ Conformance Composite Test Results - CT13

DUT Name:	0

1 Protocol Conformance Test

Protocol Test Software Revision	CT13
SOC File Name	Product1.stc
Protocol Test Log Files	Product1.log
Result Pass/Fail	
Result Pass/Fail with Large_Forward_Open	

2 Physical Layer Test

If the product includes an LED identified by a label name defined in Chapter 9 of EtherNet/IP Adaptation of CIP, the product supports the LED. Supported LEDs must have the behaviors described below.

Industrial Grade Claimed in SOC	No		
2.1 Indicator check: LEDs supported		**Present in DUT**	**Result**
Module Status LED		No	N/A
Network Status LED		No	N/A
2.2 Module status LED operation			**Result**
The product contains a red/green indicator for the module status.			N/A
The indicator is labeled "MS", "Mod", "Mod Status", or "Module Status".			N/A
Indicator operation (0.25 sec GREEN, then 0.25 sec RED at a self-test).			N/A
2.3 Network status LED operation			**Result**
The product contains a red/green indicator for the network status.			N/A
The indicator is labeled "NS", "Net", "Net Status", or "Network Status".			N/A
Indicator operation (0.25 sec GREEN, then 0.25 sec RED at a self-test).			N/A
2.4 Network connector		**Present in DUT**	**Result**
The DUT has a connector per Volume 2, Chapter 8 - (No "pigtail" allowed)		Yes	Pass
The DUT has a connector per Volume 2 - Section 8-9.2.3 N/A if Industrial Grade is not claimed in SOC			N/A

3 EDS File Test

3.1 EDS File Syntax Utility	EZ-EDS Revision:	3.11.1.20151008	
EDS File Name			
EDS File Revision			
3.2 EDS File Minimum Content			**Result**
ProdType (must match Identity Object Attribute 2)	ProdType =		
ProdCode (must match Identity Object Attribute 3)	ProdCode =		
MajRev (must match Identity Object Attribute 4, byte 0)	MajRev =		
EZ-EDS Result - Minimum Content			
3.3 EDS File Connection Entries			**Result**
All connections defined: Keyword - Path and Sizes			
3.4 EDS File Port Labels (multiple Ethernet Ports only)			**Result**
All Ethernet Link Interface sections labels match Ethernet Link object labels			

4 TCP/IP Interface Object Tests Object 0xF5 (245)

(See *EtherNetIP Interop Conformance Test Specification.pdf* for test procedure details)

4.1 Interface Configuration and Subnet Test Cases			Result
Interface Configuration - BOOTP (use Attribute 3 or other applicable interface to configure)			
Interface Configuration - DHCP Client (use Attribute 3 or other interface to configure)			
Interface Configuration - SW Configurable (using stored values - use Attribute 3 to configure)			
Interface Configuration - HW Configurable (setting address switches - use switches and attr 3)			
Subnet test case 1 (Reply) (DHCP Server used for setup - Get_Attribute_Single for request)			
Subnet test case 2 (No reply) (PC interface Properties - Get_Attribute_Single for request)			
Subnet test case 3 (Reply) (DHCP Server used for setup - Network Settings/DUT power cycle)			
Subnet test case 4 (No reply) (Network Connections -> Properties)			
Subnet test case 5 (Reply) (DHCP Server used for setup - Network Settings/DUT power cycle)			
Subnet test case 6 (Widest Subnet - Reply) (Subnet mask for DUT - use 255.0.0.0)			
TTL Test (Attr. 8) - See TTL Test Below	Get_AttributeSingle Status Code/Value:		
MCast Test (Attr 9) - See TTL Test Below	Get_AttributeSingle Status Code/Value:		
4.2 TCP/IP Objects - Multiple Interfaces Tests			**Result**
4.4.1 N/A if only one instance of the TCP/IP object			

5 Ethernet Link Object Tests Object 0xF6 (246)

Connect straight into the device for speed test cases(**DO NOT USE A HUB - a crossover cable may be needed**)

5.1 Ethernet Link Object Test Cases			Result
Speed test cases (Attribute 1) - Force PC NIC to 10Mbps (Full or Half) - Value reported OK			
Speed test cases (Attribute 1) - Force PC NIC to 100Mbps, Full Duplex - Value reported OK			
Interface Flags test cases (Attribute 2) - Force PC NIC to 100Mbps Full - Value reported OK			
Interface Flags test cases (Attribute 2) - Force PC NIC to 100Mbps Half - Value reported OK			
Force DUT and PC NIC to 100Mbps Full Duplex - DUT and PC communicate			
Force DUT and PC NIC to 100Mbps Half Duplex - DUT and PC communicate			
Force DUT and PC NIC to 10Mbps Full Duplex - DUT and PC communicate			
Force DUT and PC NIC to 10Mbps Half Duplex - DUT and PC communicate			
Physical Address test cases (attribute 3) - Match IEEE OUI listings - See wireshark capture			
5.2 Ethernet Link objects - Multiple Interfaces Tests			**Result**
Class Attribute 3 (Number of Instances)	Attribute 3 Value		
Class Attribute 2 (Max Instances)	Attribute 2 Value		
Class Attribute 1 (Revision)	Attribute 1 Value		
Instance 1 Attribute 10 (Interface Label)	Attribute 10 Value		
Instance 2 Attribute 10	Attribute 10 Value		
Admin State (Attribute 9) - Port Disable			
Admin State - Port Enable			
Admin State - Last Port not disabled			
Admin State - Enable all ports			

5.3 DLR Specific AutoMDIX Tests	Result
Test Procedure, MDIX - Port 1, Forced Duplex and Speed (DUT supporting DLR only): Configure DUT **Port 1** for forced 100 Mbps, full duplex (set Ethernet Link instance 1, attribute 6 to 02 00 64 00). Connect DUT **Port 1** (only) to uplink port of network HUB with uplink button - Connect test PC to any port of the network HUB - Use the conformance test messaging tool to get any attribute of the DUT identity object (Success expected) - push the uplink network HUB button to switch RX and TX lines - Use the conformance test messaging tool to get any attribute of the DUT identity object (Success expected) a few seconds may be needed for the DUT PHY adjusts to the HUB configuration change. **Pass Result:** Get attribute single success in both HUB configurations. (Multi-port devices NOT supporting the DLR functionality are not required to meet this requirement.)	

Test Procedure MDIX - Port 1, Auto-negotiate (DUT supporting DLR only): Configure DUT **Port 1** for Auto-negotiate (set Ethernet Link instance 1, attribute 6 to 01 00 00 00). Repeat HUB uplink switch procedure above for DUT **Port 1**. **Pass Result:** Get attribute single success in both HUB configurations. (Multi-port devices NOT supporting the DLR functionality are not required to meet this requirement.)	
Test Procedure, MDIX - Port 2, Forced Duplex and Speed (DUT supporting DLR only): Configure DUT Port 2 for forced 100 Mbps, full duplex (set Ethernet Link instance 1, attribute 6 to 02 00 64 00). Repeat HUB uplink switch procedure above for DUT **Port 2**. **Pass Result:** Get attribute single success in both HUB configurations. (Multi-port devices NOT supporting the DLR functionality are not required to meet this requirement.)	
Test Procedure - Port 2, Auto-negotiate (DUT supporting DLR only): Configure DUT Port 2 for Auto-negotiate (set Ethernet Link instance 2, attribute 6 to 01 00 00 00). Repeat HUB uplink switch procedure above for DUT **Port 2.** **Pass Result:** Get attribute single success in both HUB configurations. (Multi-port devices NOT supporting the DLR functionality are not required to meet this requirement.)	

6 Port Scans (Direct connection from PC to DUT)

6.0 Port Scans - Verify Device Reacheable during and after each Ports Scan session			**Result**
Index	Protocol		
1	TCP <nmap -n -v -r -p- -scan-delay 1ms -oX TCP.xml DUT.IP.ADDR>		
2	UDP <nmap -n -v -r -p- -scan-delay 1ms -sU -oX UDP.xml DUT.IP.ADDR>		
3	IP <nmap -n -v -r -p- -scan-delay 1ms -sO -oX IP.xml DUT.IP.ADDR>		

7 QoS Object Tests

Object 0x48 (72)

QoS Object Test Cases	**Result**
QoS Object Attributes Test (See detail below)	
QoS Behavior Test (See detail below)	

8 DLR Object Tests

Object 0x47 (71)

DLR Object Test Cases	**Result**
DLR Object Attributes Test (See detail below)	
DLR Behavior Test (See detail below)	

9 Timesync Object Tests

Object 0x43 (67)

Timesync Object Test Cases	**Result**
TimeSync Object Attributes Test (See detail below)	
Timesync Object Behavior Test (See detail below)	

10 Address Conflict Detection (ACD)Tests

10.1 ACD Test Cases	**Result**
Test Result for ACD Test Plan ran with DUT configured for HW switches or Fixed IP (N/A if no support)	
Test Result for ACD Test Plan ran with DUT configured for DHCP or BOOTP (N/A if no support)	

10.2 ACD Test Cases - Multi-Port devices	**Result**
Repeat 10.1 tests above for all device known CIP ports	

11 Redundant Owner Tests - Target

Redundant Owner Test Cases -Target	**Result**
Overall results of manual tests (see Redundant Owner TAB)	
11.1 Connection Establishment	**Result**

Two RO connections - can be established, no disconnect	
One RO connection and one listen only - can be established, no disconnect	
Two RO connections and one listen only - can be established, no disconnect	
Exclusive Owner connection already established, attempt at RO connection - Fail	
RO connection already established, attempt at exclusive owner connections - Fail	
Two redundant connections attempted, fields do not match - Fail	
11.2 O->T Data Format	**Result**
Verify use of 32 bit header in O->T traffic	
Verify use of COO flag in 32 bit header in O->T traffic	
Verify use of ROO flag in 32 bit header in O->T traffic (Unique non-zero if not connection owner and ready)	
Verify use of Run/Idle flag in 32 bit header in O->T traffic	
11.3 Data Production Data Usage	**Result**
One RO connection, no O->T timeout - T->O traffic starts and continues	
One RO connection, O->T Timeout occurs - T->O stops	
One RO connection and one listen only, O->T Timeout occurs - T->O stops	
Two RO connections, Scanner 1 is owner (COO flag set) - Scanner 1 data used	
Two RO connections, Scanner 2 is owner (COO flag set) - Scanner 2 data used	
11.4 Real-time Header Use	**Result**
Two RO connections, Both Scanners COO flag set - Last Scanner COO flag change data used	
Two RO connections, Both Scanners COO flag reset - No scanner data used? T-> O stops?	
Two RO connections, Both Scanners COO flag reset - Data of highest ROO scanner used	
Two RO connections, Both Scanners COO flag reset, Toggle Run/Idle - No effect on Target	
Two RO connections, One Scanner COO flag set, Toggle Run/Idle of owner - Change on Target	
Two RO connections, One Scanner COO flag set, Owning Scanner reset COO flag - Target Idle	

12 Redundant Owner Tests - Scanner

Redundant Owner Test Cases - Scanner	**Result**
Overall results of manual tests (see Redundant Owner TAB)	

13 Connection Configuration Object Tests

Object 0xF3 (243)

Object Test Cases	**Result**
Connection Configuration Object Protocol Test Pass in Conformance Mode.	
1- Perform Identity Reset Type-1 to DUT and Configure DUT for Network Access. 2. Assemble a network with a hub/switch, the DUT, 3 target devices, a configuration tool, and a computer hosting Wireshark.Configure DUT to originate unicast and multicast I/O messages to 3 target devices at a nominal RPI (100 ms for example).	
Connection Configuration Object Protocol Test Pass in Development Mode with all these I/O connections.	

14 Originator Connection List Object Tests

Object 0x45 (69)

Object Test Cases	**Result**
OCL Object Protocol Test Pass in Conformance Mode.	
Assemble a network with a hub/switch, the DUT, 3 target devices, a configuration tool, and a computer hosting Wireshark.	
OCL Object Protocol Test Pass in Development Mode with all these I/O connections.	

15 Dynamic Interoperabilty

Test Cases	**Result**
Originator - Normal Scan Test	
Originator - Traffic Injection Scan Test	
Target and Originator - Traffic Injection Test	

16 Position Sensor Object Tests

Object 0x23 (35)

Object Test Cases	Result
Position Sensor Object Protocol Test Pass in Conformance mode	
Position Sensor Object Protocol Test Pass in Development mode	

TCP/IP TTL and Mcast Config Test Procedure & Test Report

1- Multicast Message TTL Value - Settable TTL Value

Test Procedure	
Get TCP/IP object instance attribute 8 (TTL Value)	
If General Status = 0x14, Attribute Not Supported, then end test	
Else, note current value and Set attribute 8 value to different from current value. If General Status = 0x0E, Attribute not settable, then end test	
Get TCP/IP object instance attribute 1, and observe bit 4 (Status, Mcast pending) bit	
Start WireShark	
Run Conformance Test: Development - Connection Manager - IO Connections	
Observe the current TTL value of the multicast messages in the WireShark trace	
Requirement	**Result**
Set Attribute Single service for TCP/IP object instance Attribute 8 must return success	
TCP/IP object instance attribute 1, bit 4 (Status, Mcast pending) bit must be 1 if the DUT does not support immediate application of the new TTL value.	
The TTL value of the multicast messages must match the original value in Attribute 8 if Mcast Pending bit is 1 - or the new value if Mcast pending bit is 0	

2- Multicast Messages TTL Value - New TTL Value Applied

Test Procedure	
Power-cycle or reset DUT using reset type 0 - Power-on reset	
Observe TCP/IP object instance attribute 1, bit 4 (Status, Mcast pending) bit	
Restart WireShark	
Run Conformance Test: Development - Connection Manager - IO Connections	
Observe TTL value of DUT initiated multicast message in WireShark	
Requirement	**Result**
TCP/IP object instance attribute 1, bit 4 (Status, Mcast pending) must be reset (0)	
TTL value of DUT multicast messages in WireShark must be the value set in test case above	

3- Multicast Messages TTL Value - TTL Attribute Restored - DUT Reset

Test Procedure	
Perform a Type 1 Reset service to the Identity object if the device supports this reset type OR restore original TTL value (Set TCP/IP object instance attribute 8 (TTL Value) = 01 for example)	
Power cycle or reset the DUT via a Type 0 reset (only if the Set service was used)	
Get TCP/IP object instance attribute 1, and observe bit 4 (Status, Mcast pending) bit	
Start WireShark	
Run Conformance Test: Development - Connection Manager - IO Connections	
Observe TTL value of DUT initiated multicast message in WireShark	
Requirement	**Result**
TCP/IP object instance attribute 8 (TTL Value) must be 01	
TCP/IP object instance attribute 1, bit 4 (Status, Mcast pending) must be 0	
TTL value of DUT initiated multicast message in WireShark must be the expected original value	

4- Multicast Messages Multicast Address - Attribute Changed

Test Procedure	
Get TCP/IP object instance attribute 9 (Multicast Address)	
If General Status = 0x14, Attribute Not Supported, then end test (should be supported if TTL supported) If General Status = 0x0E, Attribute not settable, then end test (should be settable if TTL settable)	
Else, note current value and Set attribute 9 value to different from current value per table to the right	
Get TCP/IP object instance attribute 1, and observe bit 4 (Status, Mcast pending) bit	
Start WireShark	
Run Conformance Test: Development - Connection Manager - IO Connections	
Observe the current Mcast address of the multicast messages in the WireShark trace	
Requirement	**Result**
Set Attribute Single service for TCP/IP object instance Attribute 9 must return success	
TCP/IP object instance attribute 1, bit 4 (Status, Mcast pending) bit must be 1 if the DUT does not support immediate application of the new Mcast value.	
The Mcast address of the multicast messages must match the original value in Attribute 9 if Mcast Pending bit is 1 - or the new value if Mcast pending bit is 0	

5- Multicast Messages Multicast Address - New Address Applied

Test Procedure	
Power-cycle or reset DUT using reset type 0 - Power-on reset	
Observe TCP/IP object instance attribute 1, bit 4 (Status, Mcast pending) bit	
Restart WireShark	
Run Conformance Test: Development - Connection Manager - IO Connections	
Observe the current Mcast address of the multicast messages in the WireShark trace	
Requirement	**Result**
TCP/IP object instance attribute 1, bit 4 (Status, Mcast pending) must be reset (0)	
Address of DUT multicast messages in WireShark must be the value set in test case above (4)	

6- Multicast Messages Multicast Address - Multicast Attribute Restored - DUT Reset

Test Procedure	
Perform a Type 1 Reset service to the Identity object if the DUT supports this reset type OR restore original Mcast value to TCP/IP object instance attribute 09. (Set to 00 00 00 00 00 00 00 00 if DUT supports auto Mcast value.)	
Power cycle or reset the DUT via a Type 0 reset (only if the Set service was used)	
Get TCP/IP object instance attribute 1, and observe bit 4 (Status, Mcast pending) bit	
Get Mcast_Config value, attempt to write it back to Mcast_Config	
Start WireShark	
Run Conformance Test: Development - Connection Manager - IO Connections	
Observe the current Mcast address of the multicast messages in the WireShark trace	
Requirement	**Result**
TCP/IP object instance attribute 9 must be restored to its original value	
TCP/IP object instance attribute 1, bit 4 (Status, Mcast pending) must be 0	
Upon writing back the value of Mcast_Config, the device returns invalid attribute status 0x09	
Address of DUT initiated multicast message in WireShark must be the expected original value	

ACD Test Setup

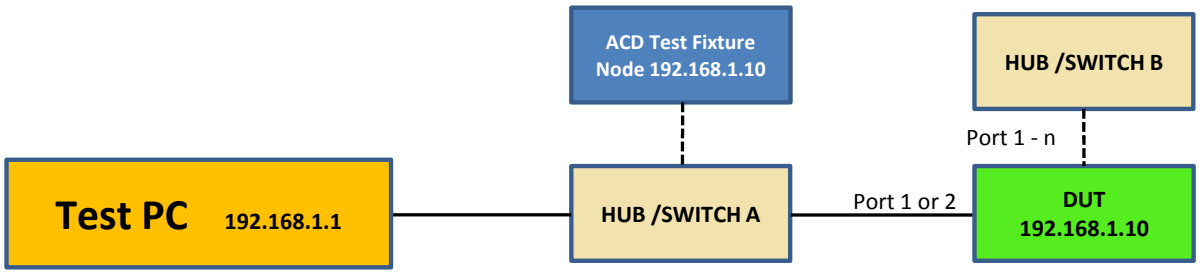

Figure 1: Test setup for all Test Cases

TEST NOTES

1. ACD Test Fixture Node: Device that will create the duplicate IP address condition
2. A dashed line in the figure above indicated that the device is only phisically connected when the test procedure instructs the user to do so
3. The Hub/Switch B shown in the figure above is used during the "Automated Test" procedure for multi-port DUTs. The "ACD Test" utility prompts the user to connect the DUT unused port to another switch or hub.

ACD Behavior Test - Manual Procedures & Test Report

1. Default Values Test Case

Test Procedure	
Connect the DUT to test system according to Figure 1	
Use the Conformance Test tool messaging tool to Get the revision value of the TCP/IP class (attr 1)	
Use the Conformance Test tool messaging tool to send a Type 1 reset request to the DUT (if supported)	
After the Reset completes, perform a Get service for TCP/IP object instance attributes 1, 10 and 11	
Requirement	**Result**
TCP/IP Object Class Attribute 1 value is 04 or higher	
TCP/IP Object Instance Attribute 1 Bit 6 AcdStatus and Bit 7 AcdFault bits are both 0	
TCP/IP Object Instance Attribute 10 value is 01	
TCP/IP Object Instance Attribute 11 value is composed of 35 bytes - each byte is 00	

2. Attribute 10 Behavior Test (Delayed Change, NV, Reset)

2.1 Test Procedure - Attribute 10 delayed change, conflict still detected	
Make sure that the DUT is connected to test system according to Figure 1	
Verify that ACD test fixture node shown in Figure 1 is properly configured: Same IP address as DUT (static address preferably), **ACD disabled**, **not connected** to Hub, Powered.	
Use the Conformance Test tool messaging tool to Set TCP/IP object instance attribute 10 to 00 (Do not power cycle the DUT)	
Start Wireshark trace on the appropriate interface	
Connect the ACD test fixture node to test system according to Figure 1	
Requirement	**Result**
DUT MS LED flashes **red** and NS LED is solid **red** [Volume 2, Ed 1.12, section F-1.2.6]	
DUT does not respond to explicit message request for Device Name - attribute 7 of Identity object (ACD test fixture node responds to request)	
2.2 Test Procedure - Attribute 10, NV check and conflict not detected	
Disconnect ACD test fixture node from test system	
Power cycle the DUT	
Use the Conformance Test tool messaging tool to Get the value of TCP/IP object instance attribute 1	
Use the Conformance Test tool messaging tool to Get the value of TCP/IP object instance attribute 10	
Use the Conformance Test tool messaging tool to Get the value of TCP/IP object instance attribute 11	
Connect the ACD test fixture node to test system according to Figure 1	
Use the Conformance Test tool messaging tool to Get the value of Identity object instance attribute 7	
Requirement	**Result**
Verify (Wireshark trace) that the DUT responds to the Get request for Identity object instance attribute 7 (if not, disconnect ACD test fixture node and clear the Test PC ARP cache: arp -d <<DUT IP Address>> from "DOS" command line window and try again)	
TCP/IP Object Instance Attribute 1 Bit 6 AcdStatus and Bit 7 AcdFault are 0	
TCP/IP Object Instance Attribute 10 value is 00	
TCP/IP Object Instance Attribute 11 value is not composed of 35 - 0x00 bytes	
IP conflict is not detected per [Volume 2, Ed 1.12, section F-1.2.6] - some form of degraded ACD may still be operational for the DUT (one or more LED may be **red** or Flashing **red**) - When Vendor Specific ACD algorithm is used, DUT CIP LED behavior should be consistent with their purpose as defined in [Vol 2, Ed 1.13, section 9-4].	

2.3 Test Procedure - Attribute 10 reset behavior	
Disconnect ACD test fixture node from test system	
Use the Conformance Test tool messaging tool to send a Type 1 reset request to the DUT (if supported), otherwise make sure to set attribute 10 to 01 and 11 to all zeros before proceeding with the test, mark following test results as N/A.	
After the Reset completes, perform a Get service for TCP/IP object instance attributes 1, 10 and 11	
Stop Wireshark trace	
Requirement	**Result**
TCP/IP Object Instance Attribute 1 Bit 6 AcdStatus and Bit 7 AcdFault are 0	
TCP/IP Object Instance Attribute 10 value is 01	
TCP/IP Object Instance Attribute 11 value is composed of 35 bytes - each byte is 00	

3. Conflict Detection Test Case - ACD enabled - Attribute 11 capabilities

Conduct as described below: sequence of procedures and requirements verification

3.1 Test Procedure - Attribute 11 conflict record verification - ACD Activity 02 or 03	
Connect the DUT to test system according to Figure 1	
Verify that ACD test fixture node shown in Figure 1 is properly configured: Same IP address as DUT (static address preferably), **ACD disabled**, **not connected** to Hub, Powered.	
Start Wireshark trace on the appropriate interface	
Connect the ACD test fixture node to test system according to Figure 1	
Requirement	**Result**
DUT MS LED flashes **red** and NS LED is solid **red** [Volume 2, Ed 1.12, section F-1.2.6]	
DUT does not respond to explicit message request for Device Name - attribute 7 of Identity object (ACD test fixture node responds to request)	
3.2 Test Procedure - Attribute 1 and 11 conflict record verification - ACD Activity 02 or 03 (continued)	
Disconnect ACD test fixture node from test system	
Unplug DUT network cable and plug it back, if DUT is unable to communicate, it's N/A. Otherwise read its TCP/IP Object Instance Attribute 1. The AcdStatus (bit 6) shall be 1.	
Cycle the DUT power	
Use the Conformance Test tool messaging tool to perform a Get service for TCP/IP object instance attribute 11 and 1.	
Stop Wireshark trace	
Identify specific Wireshark trace line creating the conflict detected by DUT and copy ARP PDU below	
Conflict ARP PDU:	
Identify DUT success reply to attribute 11 get request and cut CIP Command specific data to from Wireshark and paste it in storage below.	
ACD Activity Status (1st Byte): / Remote MAC Addr:	
ARP PDU Data:	
Requirement	**Result**
Verify that the ACD Activity Status byte is 02 or 03 (OngoingDetection or SemiActiveProbe)	
Verify that the Remote MAC Addr recorded is that of the the ACD test fixture node	
Verify that the ARP PDU data logged is the DUT raw ARP PDU that caused it to declare an IP address conflict	
Verify TCP/IP Object Instance Attribute 1 Bit 6 AcdStatus and Bit 7 AcdFault are 0	

3.3 Test Procedure (continued) - Attribute 11 set test	
Set Test 1: Use the Conformance Test tool messaging tool to perform a Set service for TCP/IP object instance attribute 11 using 34 "00-bytes" in the data field	
Set Test 2: Use the Conformance Test tool messaging tool to perform a Set service for TCP/IP object instance attribute 11 using 36 "00-bytes" in the data field	
Set Test 3: Use the Conformance Test tool messaging tool to perform a Set service for TCP/IP object instance attribute 11 using 34 "00-bytes" and add one "FF-byte" at the end of the data field	
Set Test 4: Use the Conformance Test tool messaging tool to perform a Set service for TCP/IP object instance attribute 11 using one "FF-byte" at the beginning and 34 "00-bytes" in the data field	
Set Test 5: Use the Conformance Test tool messaging tool to perform a Set service for TCP/IP object instance attribute 11 using 35 "00-bytes" in the data field	
Use the Conformance Test tool messaging tool to perform a Get service for TCP/IP object instance attribute 11	
Requirement	**Result**
DUT returns a Status Code of 0x13 to the Set Test 1 attempt (Set Failure)	
DUT returns a Status Code of 0x15 to the Set Test 2 attempt (Set Failure)	
DUT returns a Status Code of 0x09 to the Set Test 3 attempt (Set Failure)	
DUT returns a Status Code of 0x09 to the Set Test 4 attempt (Set Failure)	
DUT returns a Status Code of 0x00 to the Set Test 5 attempt (Set Success)	
TCP/IP Object Instance Attribute 11 value is composed of 35 bytes - each byte is 00	
3.4 Test Procedure (continued) - Attribute 11 conflict record verification - ACD Activity 01	
Power down and disconnect the DUT from test system	
Verify that ACD test fixture node shown in Figure 1 is properly configured: Same IP address as DUT (static address preferably), **ACD disabled**, **connected** to Hub, Powered.	
Start Wireshark trace on the appropriate interface	
Connect the DUT to test system according to Figure 1 and power it up	
Requirement	**Result**
DUT MS LED flashes **red** and NS LED is solid **red** [Volume 2, Ed 1.12, section F-1.2.6]	
DUT does not respond to explicit message request for Device Name - attribute 7 of Identity object (ACD test fixture node responds to request)	
3.5 Test Procedure - Attribute 11 conflict record verification - ACD Activity 01 (continued)	
Disconnect ACD test fixture node from test system	
Unplug DUT network cable and plug it back, if DUT is unable to communicate, it's N/A. Otherwise read its TCP/IP Object Instance Attribute 1. The AcdStatus (bit 6) shall be 1.	
Cycle the DUT power	
Use the Conformance Test tool messaging tool to perform a Get service for TCP/IP object instance attribute 1 and 11	
Stop Wireshark trace	
Identify specific Wireshark trace line creating the conflict detected by DUT and copy ARP PDU below	
Conflict ARP PDU:	
Identify DUT success reply to attribute 11 get request and cut CIP Command specific data to from Wireshark and paste it in storage below.	
ACD Activity Status (1st Byte):	Remote MAC Addr:
ARP PDU Data:	
Requirement	**Result**
Verify that the ACD Activity Status byte is 01 (ProbeIpv4Address)	
Verify that the Remote MAC Addr recorded is that of the the ACD test fixture node	
Verify that the ARP PDU data corresponds to the raw ARP PDU that produced the IP conflict	
Verify TCP/IP Object Instance Attribute 1 Bit 6 AcdStatus and Bit 7 AcdFault are 0	

3.6 Test Procedure - Attribute 11 reset via Type 1 Reset service to the Identity object	
Use the Conformance Test tool messaging tool to send a Type 1 Reset service to the Identity object	
Use the Conformance Test tool messaging tool to perform a Get service for TCP/IP object instance attribute 1, 10, 11	
Requirement	**Result**
TCP/IP Object Instance Attribute 1 Bit 6 AcdStatus is 0 and Bit 7 AcdFault is 0	
TCP/IP Object Instance Attribute 10 value is 01	
TCP/IP Object Instance Attribute 11 value is composed of 35 bytes - each byte is 00	
3.7 Test Procedure (continued) - Attribute 11 conflict record verification - ACD Activity 01	
Power down the DUT	
Verify that ACD test fixture node shown in Figure 1 is properly configured: Same IP address as DUT (static address preferably), **ACD enabled**, **connected** to Hub, Powered.	
Start Wireshark trace on the appropriate interface	
Power up the DUT	
Requirement	**Result**
DUT MS LED flashes **red** and NS LED is solid **red** [Volume 2, Ed 1.12, section F-1.2.6]	
DUT does not respond to explicit message request for Device Name - attribute 7 of Identity object (ACD test fixture node responds to request)	
3.8 Test Procedure - Attribute 1 and 11 conflict record verification - ACD Activity 01 (continued)	
Disconnect ACD test fixture node from test system	
Unplug DUT network cable and plug it back, if DUT is unable to communicate, it's N/A. Otherwise read its TCP/IP Object Instance Attribute 1. The AcdStatus (bit 6) shall be 1.	
Cycle the DUT power	
Use the Conformance Test tool messaging tool to perform a Get service for TCP/IP object instance attribute 1 and 11	
Stop Wireshark trace	
Identify specific Wireshark trace line creating the conflict detected by DUT and copy ARP PDU below	
Conflict ARP PDU:	
Identify DUT success reply to attribute 11 get request and cut CIP Command specific data to from Wireshark and paste it in storage below.	
ACD Activity Status (1st Byte):	Remote MAC Addr:
ARP PDU Data:	
Requirement	**Result**
Verify that the ACD Activity Status byte is 01 (ProbeIpv4Address)	
Verify that the Remote MAC Addr recorded is that of the the ACD test fixture node	
Verify that the ARP PDU data corresponds to the raw ARP PDU that produced the IP conflict	
Verify TCP/IP Object Instance Attribute 1 Bit 6 AcdStatus and Bit 7 AcdFault are 0	
3.9 Test Procedure - Attribute 11 reset via Type 1 Reset service to the Identity object	
Use the Conformance Test tool messaging tool to send a Type 1 Reset service to the Identity object	
Use the Conformance Test tool messaging tool to perform a Get service for TCP/IP object instance attribute 1, 10, 11	
Requirement	**Result**
TCP/IP Object Instance Attribute 1 Bit 6 AcdStatus is 0 and Bit 7 AcdFault is 0	
TCP/IP Object Instance Attribute 10 value is 01	
TCP/IP Object Instance Attribute 11 value is composed of 35 bytes - each byte is 00	

4. ACD Automated Test (Timing and Behavior) with v.1.13.0.1

Test Procedure	
Connect the DUT **configured for fixed IP address (via switches or TCP/IP object instance attribute 3)** to test system according to Figure 1	
Start the Automated **ACD Test Software** from PlugFest Toolset (ACDTest.exe)	
Configure Automated **ACD Test Software** for PC adapter interface, DUT IP address and MAC Id	
Configure Automated **ACD Test Software** to run all tests and check "ACD per volume 2" checkbox as well as the "Multiport device" checkbox if appropriate for the DUT.	
Start Wireshark trace on the appropriate interface	
Click on the Start Button and follow instructions on screen	
Stop Wireshark trace	
Requirement	**Result**
No unwaived failures reported by the **ACD Test Software** (confirmed by Wireshark trace)	

QoS Object Behavior Manual Test Setup

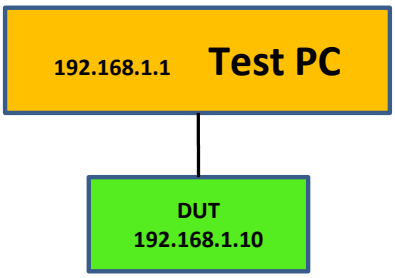

Figure 1: Test setup for QoS Object Base Test

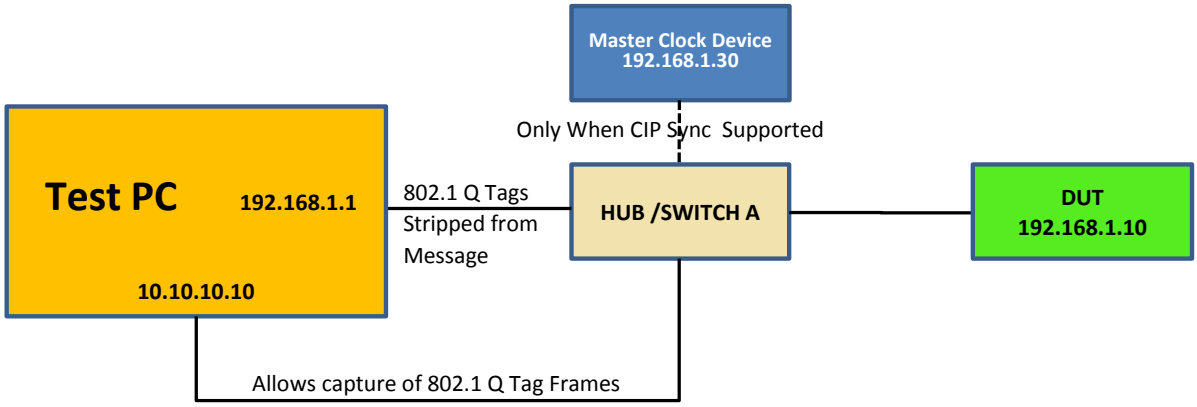

Figure 2: Test setup for QoS Object PTP DSCP and 802.1 Q Tagging Test

QoS Object Behavior Test Coverage Matrix

Test Case Name	Base Test	If CIP Sync Supported*	If 802.1 Q Tagging Supported**	If 802.1 Q and CIP Sync Supported
1.1 Default DSCP Values Scan	X	X	X	X
1.2 Observe Use of Default DSCP Values	X	X	X	X
1.3 Observe Use of Default PTP DSCP Values		X		X
1.4 Change Default DSCP Values	X	X	X	X
1.5 Observe Use of New DSCP Values	X	X	X	X
1.6 Observe Use of New PTP DSCP Values		X		X
1.7 Restore Default DSCP Values	X	X	X	X
1.8 Observe Use of Default DSCP Values	X	X	X	X
1.9 Observe Use of Default PTP DSCP Values		X		X
1.10 Enable 802.1 Q Tagging			X	X
1.11 Observe Use of 802.1 Q Tagging			X	X
1.12 Disable 802.1 Q Tagging			X	X
1.13 Verify Disuse of 802.1 Q Tagging			X	X

TESTING NOTE: Every applicable test must be performed at least once

* CIP Sync is supported if the TimeSync Object is listed

** 802.1 Q Tagging is supported if QoS Instance Attribute 1 can be set to 1

QoS Object Behavior Test - Manual Procedures & Test Report

1.1 Default DSCP Values Scan

Test Procedure	
Start the conformance test software - use STC file from DUT Vendor - and issue a type 1 reset service request to the Identity object	
Verify proper configuration according to Figure 1 or Figure 2 of QoS Test Setup (depends on DUT supported options)	
Use the **Object Scan** menu option of the **Tools** menu to scan instances 0 and 1, attributes 1 to 10 of class ID 72 (0x48) - QoS Object ID - Name the output file appropriately	
Open the output file (.CSV or .Log) located in the conformance tool installation folder and verify default attributes values per [Vol 2, Table 5-6.4]	
Requirement	**Result**
All values for supported instance attributes are conforming to [Vol 2 Ed 1.11 Table 5-6.4]	
Instance attribute 1 value returned by device is 0, or service returns error code 0x14	

1.2 Observe Use of Default DSCP Values

Test Procedure	
Start Wireshark capture on PC interface (192.168.1.1)	
Use the conformance test Messaging Tool to Get_Attribute_Single of any valid DUT object attribute	
Observe DSCP value in IP layer data packet - Differentiated Services Field associated to DUT service response to above request	
Use the conformance test Run Tests Tool in development mode to run the Connection Manager test, I/O Connections only checked	
Observe DSCP value in IP layer data packet - Differentiated Services Field associated to DUT multicast data messages	
Stop Wireshark capture	
Requirement	**Result**
Verify that the DSCP value included in the DUT Get_Attribute_Single response is 0x1b	
Verify that the DSCP value included in the DUT multicast data messages is 0x1f, 0x2b, 0x2f, or 0x37 depending on the I/O connections priority supported by the DUT	

1.4 Change Default DSCP Values

Test Procedure	
Use the conformance test Messaging Tool and attempt to set all the supported QoS object DSCP value attributes to **invalid** values (i.e. set attribute 2 to value 64, attribute 3 to value 65…)	
Use the conformance test Messaging Tool and attempt to set all the supported QoS object DSCP value attributes to **valid** values (i.e. set attribute 2 to value 02, attribute 3 to value 03…)	
Run test cases 1.2 and 1.3 above without power cycling the DUT or sending a type 0 reset request to the identity object	
Requirement	**Result**
Set_Attribute_Single with invalid values for all supported attributes returns an error code	
Set_Attribute_Single with valid values for all supported attributes completes successfully	
DSCP values observed when running test cases 1.2 and 1.3 are the same as in the original test - not the new values set in the two first steps of this test case	

1.5 Observe Use of New DSCP Values

Test Procedure	
Send a type 0 reset request to the DUT identity object or cycle power to the DUT	
Run test case 1.2 above	
Requirement	**Result**
Verify that the DSCP value included in all the observable DUT originated explicit and I/O messages are the new values set in test case 1.4 above	

1.6 Observe Use of New PTP DSCP Values

Test Procedure	
Run test case 1.3 above	
Requirement	**Result**
Verify that the DSCP value included in all the observable DUT originated PTPv2 messages are the new values set in test case 1.4 above	

1.7 Restore Default DSCP Values

Test Procedure	
Send a type 1 reset request to the DUT identity object (if supported) - If type 1 reset service to the identity object is not supported, restore the default values manually and stop the test at this point	
Verify proper configuration according to Figure 1 or Figure 2 of QoS Test Setup (depends on DUT supported options)	
Use the **Object Scan** menu option of the **Tools** menu to scan instances 0 and 1, attributes 1 to 10 of class ID 72 (0x48) - QoS Object ID - Name the output file appropriately	
Open the output file (.CSV or .Log) located in the conformance tool installation folder and verify default attributes values per [Vol 2 Ed 1.11 Table 5-6.4]	
Requirement	**Result**
All values for supported instance attributes are conforming to [Vol 2 Ed 1.11 Table 5-6.4]	
Instance attribute 1 value returned by device is 0, or service returns error code 0x14	

1.8 Observe Use of Default DSCP Values

Test Procedure	
Run test case 1.2 above	
Requirement	**Result**
Verify that the DSCP value included in the DUT Get_Attribute_Single response is 0x1b	
Verify that the DSCP value included in the DUT multicast data messages is 0x1f, 0x2b, 0x2f, or 0x37 depending on the I/O connections priority supported by the DUT	

1.9 Observe Use of Default PTP DSCP Values

Test Procedure	
Run test case 1.3 above	
Requirement	**Result**
Verify that the DSCP value included in the DUT PTPv2 management response is 0x2f	
Verify that the DSCP value included in the DUT Delay_Req messages is 0x3b	

1.10 Enable 802.1 Q Tagging

Test Procedure (Run only is device supports QoS object attribute 1)	
Verify proper configuration according to Figure 2 of QoS Test Setup	
Start Wireshark on both 192.168.1.1 and 10.10.10.10 interfaces	
Use the conformance test Messaging Tool to set QoS object attribute 1 value to 1 (enabled)	
Use the conformance test Messaging Tool to get attribute 1 of the identity object a few times	
Stop Wireshark capture on both interfaces	
Requirement	**Result**
Verify that QoS attribute 1 is successfully set to 1	
Verify in the 192.168.1.1 Wireshark trace that no 802.1 Q tag frame is observed for the entire test case	N/A
Verify in the 10.10.10.10 Wireshark trace that the DUT does not send packets including 802.1 Q tag frame in all messages sent by the DUT after the successful completion of the setting of QoS object attribute 1 value to 1	

1.11 Observe Use of 802.1 Q Tagging

Test Procedure	
Start Wireshark on both 192.168.1.1 and 10.10.10.10 interfaces	
Use the conformance test Messaging Tool to issue a type 0 reset request to the DUT identity object	
Use the conformance test Messaging Tool to get attribute 1 of the identity object a few times	
Stop Wireshark capture on both interfaces	
Requirement	**Result**
Verify in the 10.10.10.10 Wireshark trace that the DUT sends packets including 802.1 Q tag frame starting after the successful completion of the type 0 reset issued to the DUT identity object	

1.12 Disable 802.1 Q Tagging

Test Procedure	
Start Wireshark on both 192.168.1.1 and 10.10.10.10 interfaces	
Use the conformance test Messaging Tool to set QoS object attribute 1 value to 0 (disabled)	
Use the conformance test Messaging Tool to get attribute 1 of the identity object a few times	
Stop Wireshark capture on both interfaces	
Requirement	**Result**
Verify in the 10.10.10.10 Wireshark trace that the DUT sends packets including 802.1 Q tag frame for all the Get_Attribute_Single messages issued to the DUT identity object	

1.13 Verify Disuse of 802.1 Q Tagging

Test Procedure	
Start Wireshark on both 192.168.1.1 and 10.10.10.10 interfaces	
Use the conformance test Messaging Tool to issue a type 0 reset request to the DUT identity object	
Use the conformance test Messaging Tool to get attribute 1 of the identity object a few times	
Stop Wireshark capture on both interfaces	
Requirement	**Result**
Verify in the 10.10.10.10 Wireshark trace that the DUT does not send packets including 802.1 Q tag frame after the successful completion of the type 0 reset issued to the DUT identity object	

DLR Test Setup - Announce-based and Beacon-based Nodes

DLR Ring Nodes - Announce-based and Beacon based Setup

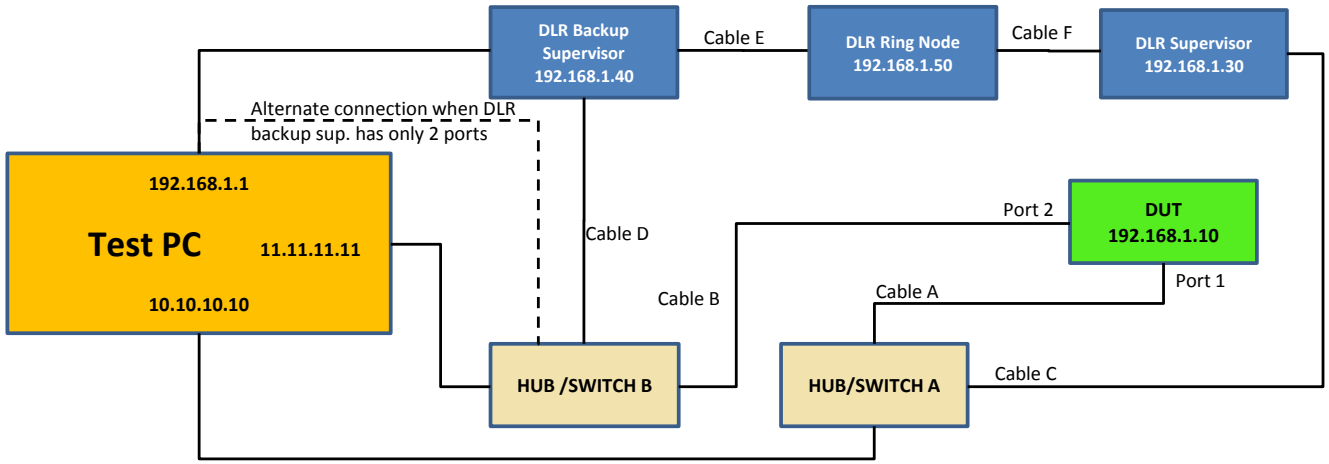

Figure 1: DLR Ring Node Conformance Test Setup

NOTE 1: HUB/SWITCH devices can be any device that enables network traffic to be captured on a third port.

NOTE 2: Using **unmanaged** switches do not allow the observation of the Link Status/Neighbor Status DUT message (point to point message) Configured **managed** switches or HUBs may be used to show all DLR traffic

DLR Supervisor Setup

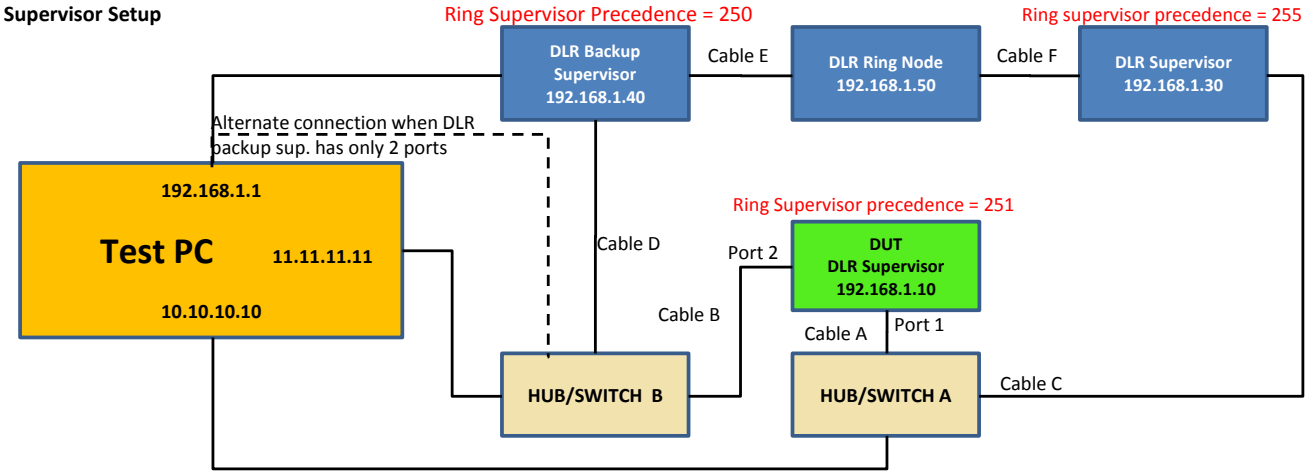

Figure 2: DLR Supervisor Conformance Test Setup for Supervisor Mode

NOTE 1: HUB/SWITCH devices can be any device that enables network traffic to be captured on a third port.

NOTE 2: Using **unmanaged** switches do not allow the observation of the Link Status/Neighbor Status DUT message (point to point message) Configured **managed** switches or HUBs may be used to show all DLR traffic

Alternate DLR Supervisor Setup

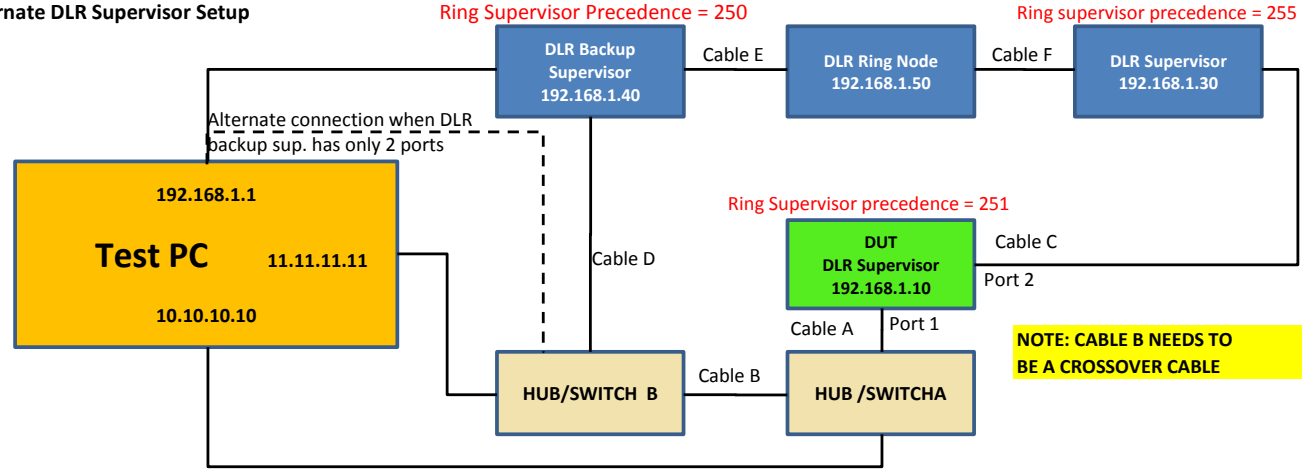

Figure 3: DLR Supervisor Alternate Conformance Test Setup for Supervisor Mode

Beacon Timeout setup

NOTE 1: HUB/SWITCH devices can be any device that enables network traffic to be captured on a third port.

NOTE 2: Using **unmanaged** switches do not allow the observation of the Link Status/Neighbor Status DUT message (point to point message) Configured **managed** switches or HUBs may be used to show all DLR traffic

1. DLR Baseline

1. Physical aspect and Configurability Considerations - No Test Procedure (performed above)	
Requirement	**Result**
DUT has at least 2 Ethernet ports	
DUT supports 100 Mbs, full duplex, forced and auto negotiation, and **auto-MDIX in both auto negotiate and forced mode** (see Ethernet Link object test procedure - item 5 above)	

2. Linear Operations

2. Test Procedure
1. Refer to Figure 1 or Figure 2 and disconnect Cable A to produce a linear configuration
2. Make sure to disable all DLR supervisors, set VLAN ID to 10, Ring Supervisor Precedence to 255, Becaon Interval to 100ms, Beacon Timeout to 400ms, by writing [00 FF A0 86 01 00 80 1A 06 00 00 00] to attribute 4 of the DLR object, 0x47 (71) of each of potential supervisor)
3. Start capturing network traffic using Wireshark on Hub/Switch A and Hub/Switch B
4. Use the Conformance test messaging tool to get all attributes of the DUT DLR object - Service 1 (Get_Attribute_ALL) on class number 71, instance 1
5. Stop capturing network traffic with Wireshark

Requirement	Result
DUT DLR attributes match **TABLE 1** below when ring is broken and includes no supervisor	
No ring protocol messages were transmitted by DUT (verify with Wireshark capture)	
Verify Device Capabilities recorded in DLR object (attribute 12 - capability flags in table 1) against claimed	

TABLE 1 - Non supervisor capable ring nodes

Network Topology:	0	Linear Topology
Network Status:	0	Normal Operation

Additional attributes for supervisor capable ring nodes

ring supervisor status:	
ring supervisor enable:	
ring supervisor precedence:	
beacon interval:	
beacon timeout:	
DLR VLAN ID:	

Per data returned by the Get_Attribute_All service - Linear operations, step 4. Data is expected to be retruned by supervisor nodes only

Device Capabilities

Capability Flags:		*Enter value reported and verify against specification*

3. Physical Ring Operations - No Enabled Ring Supervisor

3. Test Procedure
1. Refer to Figure 1 or Figure 2 and re-connect Cable A to produce a physical ring configuration
2. Make sure that all DLR supervisors are still disabled
3. Start capturing network traffic using Wireshark on Hub/Switch A and Hub/Switch B
4. Use the Conformance test messaging tool to get all attributes of the DUT DLR object
5. Stop capturing network traffic with Wireshark

Requirement	Result
DUT DLR attributes match **TABLE 2** below when network is a ring and includes no supervisor	N/A
No ring protocol messages were transmitted by DUT (verify with Wireshark capture)	

TABLE 2 - All DLR nodes

Network Topology:	0	Linear Topology
Network Status:	2	Unexpected loop detected - (2) not required but may note that this criteria is met

4. DLR Operations - Beacon or Announce-Based Nodes

4.1 - Broken Ring/Restore Detection	
4.1.a Test Procedure	
1. Refer to Figure 1 or Figure 2 and disconnect Cable A	
2. Enable DLR supervisor - i.e. ring node @192.168.1.30 (write [01 FA A0 86 01 00 80 1A 06 00 0A 00] to attribute 4 of the supervisor device DLR object) - **Never enable DUT (if supervisor capable)**	
3. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
DUT DLR attributes match **TABLE 3** below when network is a broken ring and includes a supervisor	

TABLE 3 - All DLR nodes

Network Topology:	1	Ring Topology
Network Status:	1	Ring Fault
Active Supervisor Address:	Valid	Should be <MACID>192.168.1.30 per figure 1 or 2

4.1.b Test Procedure	
1. Start capturing network traffic using Wireshark on Hub/Switch A and Hub/Switch B	
2. Refer to Figure 1 or Figure 2 and reconnect Cable A	
3. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
4. Stop capturing network traffic with Wireshark	
Requirement	**Result**
DUT DLR attributes match **TABLE 4** below when network is a ring and includes a supervisor	
verify that beacon and announce frames are properly transmitted on both DUT ports	
Verify that a Sign_on message is received on one DUT port and transmitted on the other port	

TABLE 4 - All DLR nodes

Network Topology:	1	Ring Topology
Network Status:	0	Normal Operation
Active Supervisor Address:	Valid	Should be <MACID>192.168.1.30 per figure 1 or 2

4.1.c Test Procedure - Broken Ring/Restore Detection (continued)	
Repeat test procedure disconnect cable, get attributes and verify against table 3 above, reconnect cable, get DLR object attributes, verify against table 4 above for cables B, C, D and E in figure 1 or 2, and verify that a Sign_on frame is received on one DUT port and transmitted on the other port and that beacon and announce frames are properly transmitted on both DUT ports when the system recovers. Report results in table below	
Requirement - Cable B	**Result**
DUT DLR attributes match **TABLE 3** above when network is a broken ring and includes a supervisor	
DUT DLR attributes match **TABLE 4** above when network is a ring and includes a supervisor	
Requirement - Cable C	**Result**
DUT DLR attributes match **TABLE 3** above when network is a broken ring and includes a supervisor	
DUT DLR attributes match **TABLE 4** above when network is a ring and includes a supervisor	
Requirement - Cable D	**Result**
DUT DLR attributes match **TABLE 3** above when network is a broken ring and includes a supervisor	
DUT DLR attributes match **TABLE 4** above when network is a ring and includes a supervisor	
Requirement - Cable E	**Result**
DUT DLR attributes match **TABLE 3** above when network is a broken ring and includes a supervisor	
DUT DLR attributes match **TABLE 4** above when network is a ring and includes a supervisor	

4.2 - Beacon Passing	
4.2.1 - Forward	
This test is performed at the ODVA TSP with certified ring nodes and a supervisor that has been verified to send the same beacon sequence ID out both ports at the same time for every beacon sent. Sending identical sequence ID in the beacon frames simultaneously is a DLR supervisor requirement ***for this test*** but ***not*** a requirement for DLR Supervisors per the CIP specification. See ODVA for information on obtaining a DLR supervisor that behaves in this manner for this test.	
4.2.2 - Verify that DLR ring can perform this test (required once for DLR test stand readiness).	
1. Refer to figure 1, power off or remove the DUT.	
2. Start DLR Ring, start Wireshark capture. Allow all nodes to initialize. Use Wireshark capture to:	
3. Verify RING_FAULT ring state on Hub/Switch A and Hub/Switch B	
4. Verify consecutive beacon IDs in frames from each supervisor port on Hub/Switch A and Hub/Switch B. (eg. id 1, id 2, etc.)	
5. Connect Cable A into DLR Backup Supervisor in place of Cable D. (Hub/Switch B is not used for this section of the test.)	
6. Start DLR Ring, start Wireshark capture. Allow all nodes to initialize. Use Wireshark capture to:	
7. Verify that the DLR ring is in RING_NORMAL_STATE on Hub/Switch A.	
8. Verify consecutive order of two identical (***duplicate***) Beacon Sequence IDs at Hub/Switch A (eg. id 1, id 1, id 2, id 2, etc.)	
9. Verify Beacon (Sequence ID) Interval is value set in Supervisor (TSPs use 100000us for Beacon Interval)	
10. Verify that the duplicate beacon frames, sent from Supervisor appear well within 100 us of each other, (only a few us).	
4.2.3 - Verify DUT beacon passing.	
1. Configure DLR test ring according to Figure 1 (or Figure 2) with DUT and both Hub/Switches (A and B) in the ring.	
2. Start DLR Ring then start Wireshark capture. Allow all nodes to initialize.	
3. Verify that DLR ring is in RING_NORMAL_STATE for 5 secs, stop capture.	
Requirement	**Result**
Verify ***duplicate*** beacons well within 100 us of each other on Hub/Switch A and B (both sides of DUT)	

4.3 - DUT Power On (Sign_on Frame)	
4.3.1 - DUT Sign_On	
1. Verify that DLR ring is RING_NORMAL with DUT in ring, then power off DUT	
2. Start Wireshark capture on (Hub/Switch A) and (Hub/Switch B)	
3. Power On DUT - Observe on-going capture in Wireshark - stop capturing the traffic 5 seconds after the ring status contained within the beacon data packet transitions from RING_FAULT_STATE to RING_NORMAL_STATE (See packet capture to the right for an example). If Sign_on only showed up on one port, repeat this step and watch how long it takes for the DUT to forward Sign_on on the second port.	
Requirement	**Result**
Verify that the Sign_on frame is received on one DUT port and transmitted on the other port	
Verify that the DUT adds its own information (MAC ID and IP address) to the Sign_on frame list of devices before passing it on (IP address may not yet be available)	
Send service 77 request to the Ring Supervisor DLR object to restart the Sign_on process. Verify that the DUT adds its IP address in the list.	
Verify that the reserved field pad is set to all zeros. Note: there may be no zero pad bytes required to complete the minimum frame size (60).	
Optional - requires enough ring nodes to fill the Sign On Packet - Verify that the DUT performs as specified under boundary conditions (network message list full and needs to be restarted - case of ring node and supervisor node to be handled)	N/A

4.4 - DUT Fault Response Part 1, Cables A and B

4.4.1.1 - Cable A Disconnect - Verify DUT Link_Status & Neighbor_Check_Response Frames	
1. Verify that DLR ring is functioning in RING_NORMAL state	
2. Start capturing network traffic using Wireshark on (Hub/Switch A) and (Hub/Switch B)	
3. Disconnect Cable A	
4. Stop capturing traffic after 5 seconds (i.e. after RING_FAULT state transition)	
5. Use the Conformance test messaging tool to Get_Attributes_All of the DUT DLR object	
Requirement	**Result**
Verify DUT transmits Link_Status frame on port 2 with **Status** = 0x02 (i.e. port 2 is up, port 1 is down).	
Verify DUT transmits Neighbor_Check_Response on DUT port 2 with **Source Port** = 2, then also…	
…verify **Request Source Port** matches **Source Port** in Neighbor_Check_Request frame from Neighbor (1)	
Verify Link_Status frame header VLAN_ID value matches active ring supervisor (10). (See capture right)	
DUT DLR attributes match **TABLE 3** above when network is a ring and includes a supervisor	
Verify that the reserved field is set to all zeros	
4.4.1.2 - Cable A Reconnect - Verify Ring Heal	
1. Refer to Figure 1 or Figure 2 and reconnect Cable A	
2. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
DUT DLR attributes match **TABLE 4** below when network is a ring and includes a supervisor	

4.4.1.3 - Cable B Disconnect - Verify DUT Link Status and Neighbor_Check_Response Frames	
1. Verify that DLR ring is functioning in RING_NORMAL state	
2. Start capturing network traffic using Wireshark on (Hub/Switch A) and (Hub/Switch B)	
3. Referring to Figure 1 or Figure 2 of DLR Test Setup, disconnect Cable B	
4. Stop capturing traffic after 5 seconds (i.e. after RING_FAULT state transition)	
5. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
Verify DUT transmits Link_Status frame on port 1 with **Status** = 0x01 (i.e. port 2 is down, port 1 is up).	
Verify DUT transmits Neighbor_Check_Response on DUT port 1 with **Source Port** = 1, then also…	
…verify **Request Source Port** matches **Source Port** in Neighbor_Check_Request frame from Neighbor (2)	
Verify that the Link_Status frame header VLAN_ID value (10) matches that of the active ring supervisor.	
DUT DLR attributes match **TABLE 3** above when network is a ring and includes a supervisor	
Verify that the reserved field is set to all zeros	
4.4.1.4 - Cable B Reconnect - Verify Ring Heal	
1. Refer to Figure 1 or Figure 2 and reconnect Cable B	
2. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
DUT DLR attributes match **Table 4** above when network is a ring and includes a supervisor	

4.4.2 - DUT Fault Response Part 2, Cables C and D

4.4.2.1 - Cable C - Disconnect - Verify DUT Neighbor_Status and Neighbor_Check_Request Frames	
1. Verify that DLR ring is functioning in RING_NORMAL state	
2. Start capturing network traffic using Wireshark on (Hub/Switch A) and (Hub/Switch B)	
3. Refer to Figure 1 or Figure 2 and disconnect Cable C	
4. Stop capturing traffic after 5 seconds (i.e. after RING_FAULT state transition)	
5. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
Verify DUT transmits Neighbor_Check_Request 3 times following disconnect, with **Source Port** = 1.	
Verify DUT transmits a Neighbor_Status frame on port 2 with **Status** = 0x82 (i.e. port 2 up, port 1 down).	
Verify that the Neighbor_Status frame header VLAN_ID value (10) matches that of the active ring supervisor. (See	
DUT DLR attributes match **TABLE 3** above when network is a ring and includes a supervisor	
Verify that the reserved field is set to all zeros	
4.4.2.2 - Cable C - Reconnect - Verify Ring Heal	
1. Refer to Figure 1 or Figure 2 and reconnect Cable C	
2. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
DUT DLR attributes match **TABLE 4** below when network is a ring and includes a supervisor	

4.4.2.3 Test Procedure - Cable D - Disconnect - Verify DUT Neighbor_Status and Neighbor_Check_Request Frames	
1. Verify that DLR ring is functioning in RING_NORMAL state	
2. Start capturing network traffic using Wireshark on (Hub/Switch A) and (Hub/Switch B)	
3. Refer to Figure 1 or Figure 2 and disconnect Cable D	
4. Stop capturing traffic after 5 seconds (i.e. after RING_FAULT state transition and related traffic)	
5. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
Verify DUT transmits Neighbor_Check_Request 3 times following disconnect, with **Source Port** = 2.	
Verify DUT transmits a Neighbor_Status frame on port 2 with **Status** = 0x81 (i.e. port 1 up, port 2 down).	
Verify that the Neighbor_Status frame header VLAN_ID value (10) matches that of the active ring supervisor. (See	
DUT DLR attributes match **TABLE 3** above when network is a ring and includes a supervisor	
Verify that the reserved field is set to all zeros	
4.4.2.4 Test Procedure - Cable D - Reconnect - Verify Ring Heal	
1. Refer to Figure 1 or Figure 2 and reconnect Cable D	
2. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
DUT DLR attributes match **TABLE 4** below when network is a ring and includes a supervisor	

4.5 - DUT Fault Response Part 3 - Cable E

4.5.1 Cable E - Disconnect - Observe Neighbor_Check_Request Frame	
1. Verify that DLR ring is functioning in RING_NORMAL state	
2. Start capturing network traffic using Wireshark on (Hub/Switch A) and (Hub/Switch B)	
3. Refer to Figure 1 or Figure 2 and disconnect Cable E	
4. Stop capturing the traffic 5 seconds after transition to RING_FAULT_STATE	
5. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
Verify that the DUT transmits a Neighbor_Check_Request frame on both ports. (see packet capture screens to the right for examples)	
DUT DLR attributes match **TABLE 3** above when network is a ring and includes a supervisor	
Verify that the reserved field is set to all zeros	
Test Procedure - Cable E - Reconnect	
1. Refer to Figure 1 or Figure 2 and reconnect Cable E	
2. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
DUT DLR attributes match **Table 4** above when network is a ring and includes a supervisor	

4.6 - I/O Connection Data Behavior

Test Procedure - Cable E - Disconnect/Reconnect during I/O Connection	
1. Verify that DLR ring is functional	
2. Start capturing network traffic using Wireshark on (Hub/Switch A) and (Hub/Switch B)	
3. Use the Conformance test tool to run the Connection Manager test - run only for I/O connections	
4. Refer to Figure 1 or Figure 2 and disconnect Cable E and wait for 5 seconds	
5. Reconnect cable E and wait for 2 seconds	
6. Stop capturing network traffic with Wireshark	
Requirement	**Result**
Examine the I/O SequenceNumber in the Wireshark trace to verify that the I/O data keep flowing between the Test PC and the DUT before, while, and after all the phases of the test procedure. There are no more than 3 coninuous I/O packets loss during Normal Ring Mode.	
Verify I/O data keep flowing between the Test PC and the DUT before, while, and after all the phases of the test procedure.	
No more than 3 continuous unicast I/O packet loss during Normal Ring node.	
No more than 3 continuous multicast I/O packet loss during Normal Ring node.	

5. DLR Operations - Supervisor Nodes

5.1 - Passive Supervisor	
Test Procedure	
1. Refer to **figure 2** and ensure proper network setup - Ring supervisors enabled and precedences set, DUT supervisor not enabled	
2. Start capturing network traffic using Wireshark on (Hub/Switch A) and (Hub/Switch B)	
3. Enable the DUT as a ring supervisor, making sure it does not have the highest precedence	
4. Stop traffic capture after 10 seconds of recording	
5. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
Verify no extra packets introduced by DUT having lower precedence in steady state	
Verify that current ring supervisor is reported as higher precedence supervisor	

5.2 - Supervisor Take Over	
Test Procedure - Take over	
1. Ensure setup for previous test is still working properly - all supervisors enabled	
2. Start capturing network traffic using Wireshark on 10.10.10.10 and 11.11.11.11	
3. Disable the ring supervisor which is configured with the highest precedence (255)	
4. Stop traffic capture 10 seconds after step 3.	
5. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
6. Save trace if needed and start capturing network traffic using Wireshark on 10.10.10.10 and 11.11.11.11	
7. Use the Conformance test messaging tool to send a service 77 (0x4D - Sign_on restart) request to the DUT DLR object	
8. Stop traffic capture 5 seconds after step 7.	
Requirement	**Result**
Verify that the DUT supervisor takes over when higher precedence supervisor is disabled	
Verify that the DUT is now sending the beacons on the network (wireshark log file)	
Verify beacons at configured beacon interval - 2 beacon with identical sequence number (wireshark trace)	
Verify that the DUT is now sending Announce frames once per second	
Verify Sign_on frame sent/received back by the DUT when DUT takes over	
Verify Sign_on frame sent/received back by the DUT after service 77 request	
Test Procedure - Give up to higher precedence supervisor	
1. Ensure setup for previous test is still working properly - only 2 supervisors enabled (DUT and Backup)	
2. Start capturing network traffic using Wireshark on 10.10.10.10 and 11.11.11.11	
3. Enable the ring supervisor which is configured with the highest precedence (255)	
4. Stop traffic capture 10 seconds after step 3.	
5. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
Verify no extra packets introduced by DUT having lower precedence in steady state	
Verify that higher precedence supervisor introduction causes the DUT to give up supervisorship.	

5.3 - Supervisor Fault Detection Mechanisms	
Test Procedure - Locate fault	
1. Configure the network according to **figure 3** of DLR test setup (push uplink button on hub A **or** B)	
2. Disable the higher precedence ring supervisor and verify that DUT is now the ring supervisor	
3. Start capturing network traffic using Wireshark on 10.10.10.10 and 11.11.11.11	
4. Disconnect cable B (between both hubs)	
5. Stop traffic capture 10 seconds after step 4.	
6. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
Verify "Locate Fault" frame produced on both ports upon beacon loss (no Link Status/Neighbor Status frame	
Verify "Neighbor Check Request" frame produced on one of the DUT ports (3 times)	
Verify supervisor reported status in DLR object according to **TABLE 6** below	
Test Procedure - Ring Recovery	
1. Reconnect cable B (between both hubs)	
2. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
Verify Sign_on frame sent/received back by the DUT (wireshark trace)	
Verify that Announce frame with Normal state produced on Change of State is sent before the Sign_on message	
Verify supervisor reported status in DLR object according to **TABLE 7** below	

TABLE 6	
Network Topology:	1
Network Status:	1
Ring Supervisor Status:	1
Ring Supervisor Enable:	1
Ring Faults Count:	> 0
Last Active Node on Port 1:	DUT or .40
Last Active Node on Port 2:	DUT or .40
Active Supervisor Address:	DUT
Active Super. Precedence:	DUT

TABLE 7	
Network Topology:	1
Network Status:	0
Ring Supervisor Status:	1
Ring Supervisor Enable:	1
Ring Faults Count:	> 0
Last Active Node on Port 1:	DUT or .40
Last Active Node on Port 2:	DUT or .40
Active Supervisor Address:	DUT
Active Super. Precedence:	DUT

Test Procedure - Fault detection	
1. Configure the network according to **figure 2** of DLR test setup (Restore original setup)	
2. Ensure setup for previous test is still working properly - DUT is supervisor and enabled	
3. Start capturing network traffic using Wireshark on 10.10.10.10 and 11.11.11.11	
4. Disconnect cable F	
5. Stop traffic capture 10 seconds after step 4.	
6. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	
Requirement	**Result**
Verify first beacon reporting faulted ring **after** receiving the Link Status - **On one of the ports**	
Verify beacon reporting faulted ring **before** receiving the Link_Status - **On the other port**	
Verify supervisor reported status in DLR object according to **TABLE 6** above **except that last active node on one**	
Test Procedure - Ring Recovery	
1. Start capturing network traffic using Wireshark on 10.10.10.10 and 11.11.11.11	
2. Reconnect cable F	
3. Stop traffic capture 10 seconds after step 2.	
4. Use the Conformance test messaging tool to get all attributes of the DUT DLR object	

Requirement		Result
Verify Sign_on frame sent/received back by the DUT (wireshark trace)		
Verify that beacon state returns to Normal after both beacons received		
Verify that Announce frame with Normal state produced on Change of State is sent before the Sign_on message		
Verify supervisor reported status in DLR object according to **TABLE 7** above **except that last active node on one**		
Test Procedure - Rapid Ring Fault		
1. Ensure setup for previous test is still working properly - DUT is supervisor and enabled		
2. Disconnect cable F - **5 times within 30 seconds**		
3. Use the Conformance test messaging tool to get all attributes of the DUT DLR object		
Requirement		**Result**
Verify supervisor reported status in DLR object - should be 4		
Test Procedure - Rapid Ring Fault Recovery		
1. Send service 76 (Clear_Rapid_Fault) to DUT supervisor		
2. Use the Conformance test messaging tool to get all attributes of the DUT DLR object		
Requirement		**Result**
Verify supervisor reported status in DLR object - should be 0		
Test Procedure - Partial Fault		
Partial Fault Test (Requires special Supervisor firmware to setup - Could not find it)		
Note:	Partial Fault could not be tested - No hardware setup defined	N/A

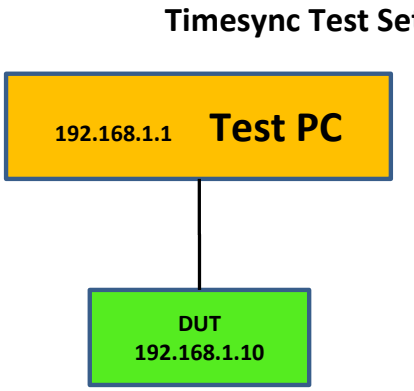

Figure 1: Test setup for Default Mode, PC Grand Master enabled Test Cases, and PTP automated tests

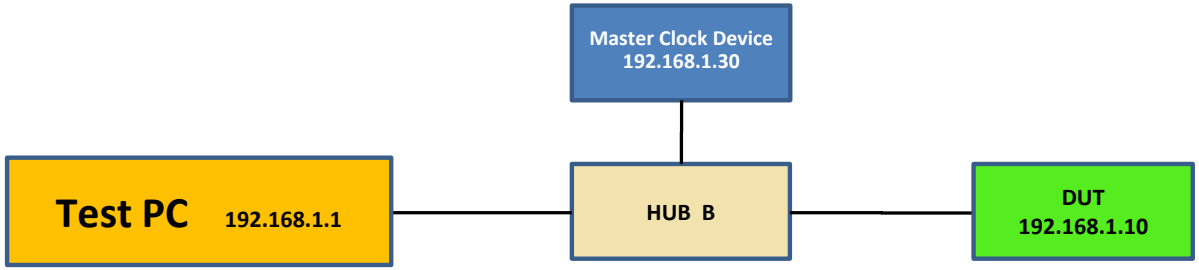

Figure 2: Test setup for Reference Master enabled Test Case

Time Sync test Manual Procedures & Test Report

1 Object Scan - Default Mode

Test Procedure	
Identity Reset Type-1 & Configure DUT for Network Access	
Conformance Test - Object Scan (Class ID 67, Instances 0..1, Attributes 1..28)	
Copy scan output (all lines of last 3 columns of .csv file) into TimeSync Test-1 data worksheet	
Start Wireshark on DUT-PC interface	
Start PTPLog utility and enable the GM clock (check GM box)	
Requirement	**Result**
Attribute 1 (PTPEnable) Verify per [Vol 1 Ed 3.14 5-47.4.1] N/A E2E	
Attribute 2 (IsSynchronized) Verify value 0 [Vol 1 Ed 3.14 5-47.4.2]	
Attribute 10 (LocalClockInfo) Verify ClockClass (Default 0xF8) TimeSource (0xA0)	
Attribute 11 (NumberOfPorts) Verify reported ports against reported clock type in attr 19	
Attribute 13 (PortEnableCfg) per [Vol 1 Ed 3.14 5-47.4.13] Verify all enabled	
Attribute 19 (ClockType) per [Vol 1 Ed 3.14 5-47.4.19] Verify valid clock type	
Verify no PTP activity in Wireshark before Grand Master enabled (Slave and Master)	
Verify no PTP activity in Wireshark after Grand Master enabled (Master only - N/A if slave)	
Verify the DUT sends PTP synchronization messages in Wireshark after Grand Master enabled (Slave only - N/A if **Master** or **End to End Transparent Clock**)	

2 Object Scan - PC Grand Master Enabled

Test Procedure	
Configure network according to Figure 1	
Run PTPLogger (v1.8.01) - Enable Grandmaster (click GM checkbox)	
Enable DUT Time Synchronization	
Conformance Test - Object Scan (Class ID 67, Instances 0..1, Attributes 1..28)	
Copy scan output (all lines of last 3 columns of .csv file) into TimeSync Test-2 data worksheet	
Confirm NV Attributes Behavior (valid, invalid, power-cycle persistent, type-1 reset to default)	
Requirement	**Result**
In Wireshark, Verify Delay_Req message from DUT N/A E2E	
Attribute 1 (PTPEnable) Verify per [Vol 1 Ed 3.14 5-47.4.1]	
Attribute 2 (IsSynchronized) Verify value 1 [Vol 1 Ed 3.14 5-47.4.2] (N/A for E2E T)	
Attribute 5 (OffsetFromMaster) Verify small values (N/A for E2E T)	
Attribute 8 (GrandmasterInfo) Verify PC GM information is recorded by DUT - local for E2E T	
Attribute 19 (ClockType) per [Vol 1 Ed 3.14 5-47.4.19] Verify valid clock type	
Attribute 27 (StepsRemoved) Shall be 0x0001 (0x0000 for E2E T)	
NV Attributes Accept Valid Value Range per [Vol 1 Ed 3.14 5-47.15.2]	
NV Attributes Reject Invalid Values per [Vol 1 Ed 3.14 5-47.15.2]	
NV Attributes Persist after Power-cycle/Type-0 Reset	
NV Attribute Revert to Default Values after Type-1 Reset per [Vol 1 Ed 3.9 5-47.15.2]	

3 Object Scan - Reference Master Enabled

Test Procedure	
Configure network according to Figure 2	
Enable Reference Master (EtherNet/IP Conformance Test Bed - Interop Rack)	
Enable DUT Time Synchronization	
Conformance Test - Object Scan (Class ID 67, Instances 0..1, Attributes 1..28)	
Copy scan output (all lines of last 3 columns of .csv file) into TimeSync Test-3 data worksheet	
Requirement	**Result**
Attribute 1 (PTPEnable) Verify per [Vol 1 Ed 3.14 5-47.4.1] **N/A for E2E**	
Attribute 2 (IsSynchronized) Verify value 1 [Vol 1 Ed 3.14 5-47.4.2] **N/A for E2E**	
Attribute 5 (OffsetFromMaster) Verify negative values (expect much smaller than PC values)	
Attribute 8 (GrandmasterInfo) Verify Reference GM information is recorded by DUT	
Attribute 27 (StepsRemoved) Should be 0x0001	

4 PTP Automatic Test Tool (PTPLog v1.8)

Test Procedure	
Configure network according to Figure 1	
Enable DUT Time Synchronization	
Verify that the device clock is detected by the PTPLog utility (for appropriate clock types)	
Run PTPLogger (v1.8.01) - select Test (Find Clocks) - ***Refer to PTP Test Matrix Tab/Page***	
Requirement	**Result**
Clock Detected by PTPLog utility (N/A for E2E T)	
No failure reported by the PTP Log utility (N/A for E2E T)	

Time Sync Object (USE PASTE SPECIAL => VALUES TO PASTE DATA FROM .CSV FILE)						
	Attr	Name	Data Type	Ret Code	Ret Size	Data
Class Attributes	1	Revision	UINT			
	2	Max Instance	UINT			
	3	Number of Instances	UINT			
	4	Optional attribute list	STRUCT of			
	5	Optional service list	STRUCT of			
	6	Max ID # Class Attrs	UINT			
	7	Max ID # Instance Attrs	UINT			
Instance Attributes	1	PTPEnable	BOOL			
	2	IsSynchronized	BOOL			
	3	SystemTimeMicro	ULINT			
	4	SystemTimeNano	ULINT			
	5	OffsetFromMaster	LINT			
	6	MaxOffsetFromMaster	ULINT			
	7	MeanPathDelayToMaster	LINT			
	8	GrandMasterClockInfo	STRUCT of			
	9	ParentClockInfo	STRUCT of			
	10	LocalClockInfo	STRUCT of			
	11	NumberofPorts	UINT			
	12	PortStateInfo	STRUCT of			
	13	PortEnableCfg	STRUCT of			
	14	PortLogAnnounceIntervalCfg	STRUCT of			
	15	PortLogSyncIntervalCfg	STRUCT of			
	16	Priority1	USINT			
	17	Priority2	USINT			
	18	DomainNumber	USINT			
	19	ClockType	WORD			
	20	ManufactureIdentity	USINT[4]			
	21	ProductDescription	STRUCT of			
	22	RevisionData	STRUCT of			
	23	UserDescription	STRUCT of			
	24	PortProfileIdentityInfo	STRUCT of			
	25	PortPhysicalAddressInfo	STRUCT of			
	26	PortProtocolAddressInfo				
	27	StepsRemoved	UINT			
	28	SystemTimeAndOffset	STRUCT of			

Attributes of interest for test

Paste data area

Time Sync Object (USE PASTE SPECIAL => VALUES TO PASTE DATA FROM .CSV FILE)						
	Attr	Name	Data Type	Ret Code	Ret Size	Data
Class Attributes	1	Revision	UINT			
	2	Max Instance	UINT			
	3	Number of Instances	UINT			
	4	Optional attribute list	STRUCT of			
	5	Optional service list	STRUCT of			
	6	Max ID # Class Attrs	UINT			
	7	Max ID # Instance Attrs	UINT			
Instance Attributes	1	PTPEnable	BOOL			
	2	IsSynchronized	BOOL			
	3	SystemTimeMicro	ULINT			
	4	SystemTimeNano	ULINT			
	5	OffsetFromMaster	LINT			
	6	MaxOffsetFromMaster	ULINT			
	7	MeanPathDelayToMaster	LINT			
	8	GrandMasterClockInfo	STRUCT of			
	9	ParentClockInfo	STRUCT of			
	10	LocalClockInfo	STRUCT of			
	11	NumberofPorts	UINT			
	12	PortStateInfo	STRUCT of			
	13	PortEnableCfg	STRUCT of			
	14	PortLogAnnounceIntervalCfg	STRUCT of			
	15	PortLogSyncIntervalCfg	STRUCT of			
	16	Priority1	USINT			
	17	Priority2	USINT			
	18	DomainNumber	USINT			
	19	ClockType	WORD			
	20	ManufactureIdentity	USINT[4]			
	21	ProductDescription	STRUCT of			
	22	RevisionData	STRUCT of			
	23	UserDescription	STRUCT of			
	24	PortProfileIdentityInfo	STRUCT of			
	25	PortPhysicalAddressInfo	STRUCT of			
	26	PortProtocolAddressInfo				
	27	StepsRemoved	UINT			
	28	SystemTimeAndOffset	STRUCT of			

NV Attributes to be verified

Attributes of interest for test

Paste data area

Time Sync Object (USE PASTE SPECIAL => VALUES TO PASTE DATA FROM .CSV FILE)						
	Attr	Name	Data Type	Ret Code	Ret Size	Data
Class Attributes	1	Revision	UINT			
	2	Max Instance	UINT			
	3	Number of Instances	UINT			
	4	Optional attribute list	STRUCT of			
	5	Optional service list	STRUCT of			
	6	Max ID # Class Attrs	UINT			
	7	Max ID # Instance Attrs	UINT			
Instance Attributes	1	PTPEnable	BOOL			
	2	IsSynchronized	BOOL			
	3	SystemTimeMicro	ULINT			
	4	SystemTimeNano	ULINT			
	5	OffsetFromMaster	LINT			
	6	MaxOffsetFromMaster	ULINT			
	7	MeanPathDelayToMaster	LINT			
	8	GrandMasterClockInfo	STRUCT of			
	9	ParentClockInfo	STRUCT of			
	10	LocalClockInfo	STRUCT of			
	11	NumberofPorts	UINT			
	12	PortStateInfo	STRUCT of			
	13	PortEnableCfg	STRUCT of			
	14	PortLogAnnounceIntervalCfg	STRUCT of			
	15	PortLogSyncIntervalCfg	STRUCT of			
	16	Priority1	USINT			
	17	Priority2	USINT			
	18	DomainNumber	USINT			
	19	ClockType	WORD			
	20	ManufactureIdentity	USINT[4]			
	21	ProductDescription	STRUCT of			
	22	RevisionData	STRUCT of			
	23	UserDescription	STRUCT of			
	24	PortProfileIdentityInfo	STRUCT of			
	25	PortPhysicalAddressInfo	STRUCT of			
	26	PortProtocolAddressInfo				
	27	StepsRemoved	UINT			
	28	SystemTimeAndOffset	STRUCT of			

Attributes of interest for test

Paste data area

Redundant Owner - Conformance Test Matrix				
Tests List		**Scanner**	**Target**	**Comment/Description**
Connection Establishment	Connection establishment - Positive Test 1	X	X	Two redundant owner connections - can be established, no disconnect
	Connection establishment - Positive Test 2	X	X	One redundant owner connection and one listen only - can be established, no disconnect
	Connection establishment - Positive Test 3	X		Two redundant owner connections and one listen only - can be established, no disconnect
	Connection establishment - Negative Test 1		X	Exclusive owner connection already established, attempt at redundant owner connection - Fail
	Connection establishment - Negative Test 2		X	Redundant connection already established, attempt at exclusive owner connections - Fail
	Connection establishment - Negative Test 3		X	Two redundant connections attempted, fields do not match - Fail
O->T Data Format		Two Scanners connected to target		
	O->T Data Format Test 1	X		Verify use of 32 bit header in O->T traffic
	O->T Data Format Test 2	X		Verify use of COO flag in 32 bit header in O->T traffic
	O->T Data Format Test 3	X		Verify use of ROO flag in 32 bit header in O->T traffic (Unique non-zero if not connection owner and ready)
	O->T Data Format Test 4	X		Verify use of Run/Idle flag in 32 bit header in O->T traffic
Data Production Data Use	T->O Requirements Test 1		X	One redundant owner connection, no O->T timeout - T->O traffic starts and continues
	T->O Requirements Test 2		X	One redundant owner connection, O->T Timeout occurs - T->O stops
	T->O Requirements Test 3		X	One redundant owner connection and one listen only, O->T Timeout occurs - T->O stops
	O->T Data Use Test 1		X (*)	Two redundant owner connections, Scanner 1 is owner (COO flag set) - Scanner 1 data used
	O->T Data Use Test 2		X (*)	Two redundant owner connections, Scanner 2 is owner (COO flag set) - Scanner 2 data used
Real Time header use	COO Flag Test 1		X (*)	Two redundant owner connections, Both Scanners COO flag set - Last Scanner COO flag change data used
	COO Flag Test 2 (All ROO priorities are 0)		X (*)	Two redundant owner connections, Both Scanners COO flag reset - No scanner data used? T-> O stops?
	ROO Priority Value Test 1 (unique non-zero)	X	X (*)	Two redundant owner connections, Both Scanners COO flag reset - Data of highest ROO scanner used
	Run/Idle flag Test 1	X	X (*)	Two redundant owner connections, Both Scanners COO flag reset, Toggle Run/Idle - No effect on Target
	Run/Idle flag Test 2	X	X (*)	Two redundant owner connections, One Scanner COO flag set, Toggle Run/Idle of owner - Change on Target
	Run/Idle flag Test 3	X	X (*)	Two redundant owner connections, One Scanner COO flag set, Owning Scanner reset COO flag - Target Idle

(*) if verifiable

X	*passing test condition observed*
X	*indeterminate - unable to configure listen only connection due to scanner establishment of T->O connectionsas point-to-point instead of multicast*
X (*)	*passing test condition observed using Get_Connection_Owner service on Target; test system did not allow for analysis of actual data usage*
X (*)	*indeterminate - unable to achieve test condition using system test fixture (Scanner implementation ensured mutually exclusive use of COO flag)*
	test does not apply to DUT
X	*failure*

NOTE: ODVA will rely on Vendor's demonstration of the functionality per original test plan provided

Dynamic Interoperability Test

1 Normal Scan

Test Procedure	
Assemble a network with a hub, the DUT,3 target devices, a configuration tool, and a computer hosting Wireshark.	
Configure DUT to originate I/O messages to 1 or more target device(s) at a (nominal) 100 ms RPI	
Verify the target multicast I/O messages and the DUT responses from the Wireshark trace	
Verify that the network continues to function for a minimum of 90 minutes without any scanner or target going off-line.	
Requirement	**Result**
DUT configured to produce and consume I/O data from target(s)	
Targets produce multicast I/O data at 100 ms RPI	
Network continues to function for a minimum of 90 minutes without any scanner or target going off-line	

2 Traffic Injection

Test Procedure	
Inject Steady-State Managed Background Traffic into the network for approximately 60 seconds.	
Inject Steady-State Unmanaged Background Traffic into the network for approximately 60 seconds.	
Inject Burst Managed Background Traffic into the network for approximately 60 seconds.	
Inject Burst Unmanaged Background Traffic into the network for approximately 60 seconds.	
Requirement	**Result**
The DUT does not loose connections during the traffic injection	
The network is operating normally at the end of the Steady-State Managed Background Traffic injection	
The DUT does not loose connections during the traffic injection	
The network is operating normally at the end of the Steady-State Unmanaged Background Traffic	
The DUT does not loose connections during the traffic injection	
The network is operating normally at the end of the Burst Managed Background Traffic	
The DUT does not loose connections during the traffic injection	
The network is operating normally at the end of the Burst Unmanaged Background Traffic	

3 Traffic Injection

Test Procedure	
Connect two Test PC Ethernet ports to a hub/switch and connect the DUT to the same hub/switch	
Start Injecting Burst Unmanaged Background Traffic into the network on one of the test PC ports connected to the hub/switch	
Run the full protocol conformance test on the second test PC port	
Observe the error and warning count when the protocol conformance test completes and stop the traffic injection then	
Requirement	**Result**
The number of conformance test software declared errors and warnings is the same as when the test is run without traffic injection	
The DUT is operating normally at the end of the Steady-State Managed Background Traffic injection test	