

University of New Hampshire

## University of New Hampshire Scholars' Repository

---

Master's Theses and Capstones

Student Scholarship

---

Fall 2006

### Conformance testing of EPON devices

Eric Lynskey

*University of New Hampshire, Durham*

Follow this and additional works at: <https://scholars.unh.edu/thesis>

---

#### Recommended Citation

Lynskey, Eric, "Conformance testing of EPON devices" (2006). *Master's Theses and Capstones*. 206.  
<https://scholars.unh.edu/thesis/206>

This Thesis is brought to you for free and open access by the Student Scholarship at University of New Hampshire Scholars' Repository. It has been accepted for inclusion in Master's Theses and Capstones by an authorized administrator of University of New Hampshire Scholars' Repository. For more information, please contact [Scholarly.Communication@unh.edu](mailto:Scholarly.Communication@unh.edu).

CONFORMANCE TESTING OF EPON DEVICES

BY

Eric Lynskey  
Bachelor of Science, University of New Hampshire, 2000

THESIS

Submitted to the University of New Hampshire  
in Partial Fulfillment of  
the Requirements for the Degree of

Master of Science  
in  
Electrical Engineering

September 2006

UMI Number: 1437635

### INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

**UMI<sup>®</sup>**

---

UMI Microform 1437635

Copyright 2006 by ProQuest Information and Learning Company.

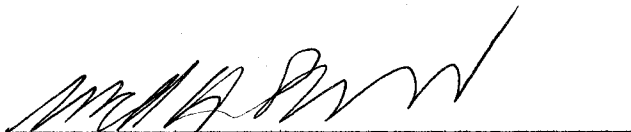
All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company  
300 North Zeeb Road  
P.O. Box 1346  
Ann Arbor, MI 48106-1346

This thesis has been examined and approved.



Thesis Director, Michael J. Carter, Ph.D.  
Associate Professor of Electrical Engineering



William H. Lenharth, Ph.D.  
Research Associate Professor of Electrical  
Engineering



W. Thomas Miller III, Ph.D.  
Professor of Electrical Engineering

August 2, 2006  
Date

## **DEDICATION**

This thesis is dedicated to my wife, Kait, for putting up with all of the nights, weekends, and years that have gone into this thesis and the rest of my graduate career. This thesis would not exist had it not been for her love and support.

## **ACKNOWLEDGEMENTS**

I would like to acknowledge the current and former employees of the UNH InterOperability Laboratory for their support and assistance over the past several years. Thanks go out to those individuals that aided in the creation of the original TIGER system, on which the OLT emulator is based, and to those involved with the PMD Matlab code. A special thanks to Bob Noseworthy for first introducing me to the IEEE 802.3 community and for giving me the chance to become an active member. Also, I would like to thank to Dr. Glen Kramer, who literally wrote the book on EPON, and has been of immense help. Finally, thanks to the members of the thesis committee, Dr. Michael Carter, Dr. William Lenharth, and Dr. Thomas Miller for their guidance and support.

## TABLE OF CONTENTS

DEDICATION .....	iii
ACKNOWLEDGEMENTS .....	iv
TABLE OF CONTENTS .....	v
LIST OF TABLES .....	xii
LIST OF FIGURES .....	xiii
ABSTRACT .....	xv

CHAPTER	PAGE
I. BACKGROUND INFORMATION .....	1
Introduction to EFM .....	1
Introduction to UNH-IOL .....	3
Consortium Model .....	3
Test Suites .....	4
Interoperability test bed .....	4
EFM Consortium .....	5
II. CONFORMANCE AND INTEROPERABILITY .....	6
Conformance vs. interoperability .....	7
Definition of conformance .....	8
Conformance of transmitter .....	10
Conformance of receiver .....	11

Stimulus and response testing .....	12
Positive and negative testing .....	12
Definition of interoperability .....	14
Interoperability and conformance .....	15
Necessity of Conformance .....	18
III. EPON INTRODUCTION .....	21
History of EPON .....	21
EPON basics .....	22
OLT .....	22
ONU .....	23
Passive optical splitter .....	24
Fiber plant .....	24
EPON Operation .....	24
EPON architecture .....	26
IV. EPON TEST TOOLS .....	28
OLT/ONU Emulation Tools .....	29
Database of results .....	29
Report generator .....	30
Intelligent analysis .....	30
User-defined test scripts .....	30
Processing engine .....	31
Transmit emulator .....	31



Complex triggering capability .....	31
Receive emulator .....	32
Detailed Transmitter tool .....	32
Modified TIGER System .....	33
Optical transceivers .....	34
Logic analyzer solution .....	35
LabVIEW code environment .....	35
Test Vector Generator .....	36
Triggering capabilities .....	37
Additional EPON VIs .....	40
Future improvements .....	41
Passive Monitoring Tools .....	41
TIGER System monitoring solution .....	42
SmartBits monitoring solution .....	42
Optical PMD Tools .....	44
Automated transmitter testing .....	44
Power measurements .....	47
Timing measurements .....	49
Automated receiver sensitivity testing .....	50
BER testing algorithm .....	55
Individual contributions to existing tools .....	59
V. CONFORMANCE TESTING OF CLAUSE 65 RS .....	60

Overview of Clause 65 .....	60
Preamble for EPON devices .....	60
Preamble Calculation Example .....	63
Laser Control for ONU .....	65
EPON Forward Error Correction .....	66
Clause 65 Testing.....	67
Protocol layer test setup .....	67
Unidirectional Mode .....	68
Enable variable .....	69
Mode Variable.....	70
Logical_Link_ID Variable .....	71
CRC-8 Generation .....	73
CRC-8 Initial Value .....	74
Start of LLID delimiter (SLD) Parsing.....	75
SLD Replacement.....	76
LLID Matching.....	77
LLID Replacement .....	79
CRC-8 Checking.....	80
CRC-8 Replacement.....	80
Buffer Depth.....	81
Laser Control .....	82
FEC Coding Choice .....	83

Uncorrectable Block Replacement.....	84
FEC state diagrams .....	85
Loop Timing .....	85
Delay Variation.....	86
VI. CONFORMANCE TESTING OF CLAUSE 64 MPCP .....	87
Overview of Clause 64 .....	87
GATE messages.....	87
REGISTER_REQ messages.....	88
REGISTER messages .....	89
REGISTER_ACK messages .....	89
REPORT messages.....	89
EPON discovery process .....	90
Normal EPON operation .....	95
Clause 64 Testing.....	95
Generic MPCPDU description.....	96
GATE description .....	97
REPORT description.....	99
REGISTER_REQ description.....	100
REGISTER description .....	101
REGISTER_ACK description .....	103
Timestamp verification .....	104
OLT delay requirements .....	105

Implementation of mpcp_timer .....	105
Transmission in proper timeslot .....	106
RANDOM_WAIT .....	107
Recognition of timestampDrift .....	108
OLT transmission of discovery GATE .....	109
OLT reception of REGISTER_REQ .....	110
OLT transmission of REGISTER.....	111
OLT completion of discovery process .....	113
ONU transmission of REGISTER_REQ.....	114
ONU reception of REGISTER.....	115
ONU transmission of REGISTER_ACK .....	117
ONU deregistration .....	118
Generation of grants .....	119
Transmission of unsolicited REPORT .....	120
GATE processing by ONU .....	121
VII. CONFORMANCE TESTING OF CLAUSE 60 PMD.....	123
Overview of Clause 60 .....	123
Transmitter specifications .....	124
Receiver specifications .....	124
Measurement methodologies.....	124
Optical channel .....	125
Clause 60 Testing.....	125

Transmitter testing.....	125
Signaling speed .....	126
Average launch power .....	127
Extinction ratio and optical modulation amplitude .....	128
Transmitter eye mask .....	129
Receiver specifications.....	131
Receiver sensitivity .....	131
VIII. CONCLUSION.....	132
LIST OF REFERENCES .....	134

## LIST OF TABLES

Table 1 - Eye diagram distribution.....	48
Table 2 - Preamble Values.....	62
Table 3 - Generic MPCPDU.....	96
Table 4 - Content of GATE.....	98
Table 5 - Content of GATE flags field.....	98
Table 6 - Contents of REPORT.....	99
Table 7 - Contents of REPORT bitmap field .....	99
Table 8 - Contents of REGISTER_REQ.....	100
Table 9 - Contents of REGISTER_REQ flags field.....	101
Table 10 - Contents of REGISTER .....	102
Table 11 - Contents of REGISTER flags field .....	102
Table 12 - Contents of REGISTER_ACK.....	103
Table 13 - Contents of REGISTER_ACK flags field.....	103
Table 14 - EPON Launch powers.....	127
Table 15 - EPON OMA levels.....	128
Table 16 - Eye mask coordinates.....	130
Table 17 - Receiver sensitivity values .....	131

## LIST OF FIGURES

Figure 1 - Normal upstream operation .....	25
Figure 2 - EPON architecture example .....	27
Figure 3 - High level EPON conformance tool .....	29
Figure 4 - Detailed EPON conformance tool .....	32
Figure 5 - TIGER system and optics .....	33
Figure 6 - Test Vector Generation Tool .....	39
Figure 7 - Trigger program .....	40
Figure 8 - Transmit PMD test setup .....	44
Figure 9 - Transmitter testing program .....	45
Figure 10 - Automated Eye diagram .....	47
Figure 11 - Automated power measurements .....	49
Figure 12 - Automated timing measurements .....	50
Figure 13 - PMD Receiver Test Setup .....	51
Figure 14 - Receiver tester Automated BER screen .....	53
Figure 15 - Receiver tester manual test screen .....	55
Figure 16 - Receiver tester manual attenuation screen .....	56
Figure 17 - Automated BER plot example .....	58
Figure 18 - EPON Frame Structure .....	62
Figure 19 - Serial Implementation of Preamble CRC Generator .....	63
Figure 20 - Laser control for ONU .....	66
Figure 21 - EPON protocol layer test setup .....	68

Figure 22 - CRC-8 calculator .....	74
Figure 23 - Frame format with FEC .....	83
Figure 24 - Transmission of discovery GATE.....	90
Figure 25 - Transmission of REGISTER_REQ .....	92
Figure 26 - ONU collision .....	92
Figure 27 - Transmission of REGISTER .....	93
Figure 28 - Transmission of initial GATE.....	94
Figure 29 - Transmission of REGISTER_ACK.....	94
Figure 30 - Transmitter eye mask .....	130



## **ABSTRACT**

### **CONFORMANCE TESTING OF EPON DEVICES**

By

Eric Lynskey

University of New Hampshire, September 2006

In June of 2004, the IEEE Standards Association approved IEEE 802.3ah-2004, which defined Ethernet for subscriber access networks, or Ethernet in the First Mile (EFM). Part of this standard contained a new type of Ethernet technology, that of Ethernet Passive Optical Networks (EPON). EPON devices have introduced new and challenging requirements for both conformance and interoperability testing. This thesis lays the foundation, to be used by the University of New Hampshire InterOperability Laboratory (UNH-IOL), for comprehensive conformance testing of EPON devices. A series of recommendations are put forth that allow for the creation and implementation of EPON test suites and test tools that can be used to help verify the compliance and interoperability of EPON devices. The focus of this document is on the conformance testing of EPON devices within the MAC Control sublayer, reconciliation sublayer, and physical medium dependent sublayer. These three sublayers, defined in three individual clauses within the IEEE 802.3 standard, represent the bulk of requirements necessary to build a conformant EPON device, and therefore the need and demand for testing of these layers has never been greater.

## **CHAPTER I**

### **BACKGROUND INFORMATION**

This primary intention of this chapter is to provide background information to the reader that may help with the understanding of the material presented later in this document. There are 2 main areas in which background will be provided: EFM, and the UNH-IOL. In some instances, more detailed information will be provided in subsequent chapters. For a quick recap or overview, the reader is urged to begin here.

#### **Introduction to EFM**

In June 2004 the IEEE 802.3ah Task Force completed a set of specifications for Ethernet in the First Mile. Since that publication date, the entire set of IEEE 802.3ah specifications have been rolled into an updated version of the standard. At the publication time of this thesis, the latest version is IEEE Std. 802.3-2005. Included in this standard [1] are specifications for several new optical and copper Physical Layer Devices (PHY), and management functions. New Physical Medium Dependent (PMD) sublayers have been defined for the following technologies:

- Point to point (P2P) 100 Mb/s dual fiber and single fiber devices for operation up to 10 km at a BER of  $10^{-12}$
- P2P 1000 Mb/s dual fiber and single fiber devices for operation up to 10 km at a BER of  $10^{-12}$

- Point to multipoint (P2MP, also known as EPON) 1000 Mb/s single fiber devices for operation up to 10 km and 20 km at a BER of  $10^{-12}$
- P2P 10 Mb/s voice grade twisted pair for operation up to 750 m at a BER of  $10^{-7}$
- P2P 2 Mb/s voice grade twisted pair for operation up to 2500 m at a BER of  $10^{-7}$

A single new Physical Coding Sublayer (PCS) has been defined for both of the copper PMD's, and minor modifications have been made to the existing PCS for both 100BASE-X and 1000BASE-X to support the new optical PMD's. Extensions of the Reconciliation Sublayer (RS), including specifications for Forward Error Correction (FEC), and a Multi-Point MAC Control Protocol (MPCP) have been written for the P2MP PMD. Finally, a new section of the document specifies Operations Administration and Maintenance (OAM) functions and operations that can be supported by both new EFM devices and traditional 802.3 devices.

With these specifications now complete, and with vendors selling and marketing compliant IEEE 802.3ah components and systems, it is important that a comprehensive study be done to show that interoperability and conformance can and does exist in EFM devices. A set of test suites should be made publicly available for review and comment that thoroughly describe a group of conformance and interoperability tests that should be performed on all EFM products.

Several papers have been written to describe the recent EFM specification [2][3][4] but there has not yet been much of a focus on the testing of EFM. Organizations such as the now defunct Ethernet in the First Mile Alliance (EFMA), the PON Forum, and the IEEE EPON Forum, have all been created to provide forums for the discussion and demonstration of EFM technologies, yet the development of testing strategies has not been a top priority for these or other organizations. In order to prepare both the vendor and user communities for the successful deployment of EFM products, such an effort needs to be initiated with input from all involved parties.

### **Introduction to UNH-IOL**

The UNH-IOL was formed back in 1988 with a mission to both educate students in areas of high technology and to provide a neutral environment for multi-vendor interoperability and conformance testing. The model used by the UNH-IOL is quite simple and has proven to be very successful.

### **Consortium Model**

The UNH-IOL definition of a consortium is a group of companies that want to come together in an organized fashion to solve technical problems associated with a specific networking technology. The UNH-IOL provides a neutral environment where these competing companies can gather and work with each other to solve these problems. When enough companies are interested in a technology, and the resources are available, the consortium will be formed. The UNH-IOL will then develop testing services for the consortium members. As the consortium activities are fully funded by the members of the consortium, the

members are able to provide feedback and input regarding the types of services that are to be created. Some consortiums are primarily interested in conformance testing, whereas others are more concerned with interoperability testing.

### **Test Suites**

Most of the consortiums within the UNH-IOL are based on technologies that have been created from approved standards created by organizations such as the IEEE, IETF, ITU, and others. The UNH-IOL will then take the necessary standards and begin to write test suites based on these documents by examining all of the mandatory and optional features within the standard. The test suites developed by the UNH-IOL are then posted publicly on their website so that they may be reviewed and discussed. Ultimately, the UNH-IOL will then implement the test suites using a variety of commercially available and custom test equipment and will then provide reports to the consortium members listing and describing the results of the testing. These reports provide valuable information to the consortium members, and can be used to identify and fix problems within their devices. Additionally, the reports are often provided to the member's potential customers as proof of an independent verification of their equipment.

### **Interoperability test bed**

One of the greatest benefits that the UNH-IOL provides to its members is that it houses one of the largest, if not the largest, interoperability test beds in the world. In most consortiums, members are required to provide representative pieces of networking equipment to the UNH-IOL facility. Over time, this allows

the consortium to build up a rather large and broad set of equipment from many different companies. The cost of such a test bed is simply prohibitive for any one company to do it on their own, and therefore the consortium provides an excellent source of equipment for interoperability testing. When a member comes in for interoperability testing, the report received will show results obtained from a wide cross section of equipment from many different vendors.

### **EFM Consortium**

During the summer of 2004, the UNH-IOL launched the EFM Consortium. Initially, interest was primarily focused on testing the new frame-based OAM protocol. The first group of companies that became members of the consortium were developing various types of media converters and were using the OAM protocol to control the different converter boxes from the head-end station. In 2005, interest began to rise in detailed conformance testing of the EPON protocol. At the publication time of this thesis, testing in EPON has reached an even higher level of interest for the silicon providers, the system vendors, and the service providers that are deploying EPON. The industry need and demand for agreed upon testing procedures and documentation has never been greater.

Since most of the technologies tested within the EFM Consortium are based on previously defined Ethernet technologies, it is possible to reuse or modify existing test equipment and tools. A number of these tools were used in the creation of the EPON-specific tools and tests described in this thesis. However, unless otherwise mentioned, all test vectors, tools, and utilities were created specifically for this project by the author.

## **CHAPTER II**

### **CONFORMANCE AND INTEROPERABILITY**

Interoperability, or a lack thereof, is one of the most important factors to consider when developing, marketing, and ultimately deploying a new technology. Equally important, when developing products based on an accepted standard, is the concept of conformance. Together, both interoperability and conformance have the ability to help foster the acceptance and success of new technologies. These concepts take on even more importance when dealing with a network that extends through the last, or first, mile to the subscriber in a community network.

Any new technology will have its share of interoperability problems, and it is not uncommon for vendors to produce products that may not be completely conformant to the standard, especially when pre-standard products are produced, or when vendors have implemented proprietary features, which may not be recognized in the standard. Over time, as the technology matures, the number of interoperability and conformance problems will decline, and this will help to increase the success and penetration of the technology. The goal is that a device from any one company will interoperate with a similar device from any other company. Although such a reality may not be readily feasible, every attempt should be made to achieve it.

Community networks with deployments of xDSL and DOCSIS technologies have been in existence for a number of years. Recently, there has been a push to create optical community networks, commonly referred to as FTTx. Although there are several technologies available, two of the architectures are point-to-point (P2P) optical fiber and point to multipoint (P2MP) passive optical networks (PON). Specifically, beginning in November of 2000, the IEEE 802.3 Working Group began actively developing a standard containing PON and P2P last mile optical networks and last mile copper networks utilizing the SHDSL and VDSL physical layers. All of these last mile technologies were developed under the heading of Ethernet in the First Mile (EFM)<sup>1</sup>, which was finally ratified in June of 2004.

As a new technology, there are certainly some interoperability and conformance problems with EPON devices. These problems can be eliminated more quickly if the proper measures are taken. Interoperability problems will only be discovered, and then corrected, if the EPON industry comes together with a concerted and organized effort to demonstrate interoperability and conformance to each other, and then to the public. Such an effort can be helped by the development of a set of comprehensive standards based conformance tests and agreed upon scenarios under which interoperability must be achieved.

### **Conformance vs. interoperability**

An understanding of the concepts of conformance and interoperability is of paramount importance when deciding which technologies should be deployed in any network. Not only are the definitions of these concepts important, but also



an understanding of the similarities and differences between conformance and interoperability and how they relate to the various pitfalls associated with these concepts. For example, it is possible that two devices that are conformant to the standard will not necessarily be interoperable, and two devices that are interoperable are not necessarily conformant. Additionally, it is possible that one device may interoperate with another device under a certain set of conditions, but it may not be interoperable under all conditions, or with all similar devices. Only exhaustive testing and documentation can help to ensure that devices are both conformant to the standard, and interoperable with the vast majority of available products.

#### **Definition of conformance**

A device is said to be conformant, or compliant, to a standard if it has properly implemented all of the mandatory portions of that standard. All mandatory portions of IEEE 802.3 are set apart from the rest of the text by a *shall* statement. All statements that say something *shall* or *shall not* happen are mandatory, and are necessary for a device to be considered conformant. Additional statements within IEEE 802.3 include recommendations (*should*, *should not*) and options (*may*, *may not*). State diagrams are often included along with the supporting text in order to clearly and concisely describe the behavior of certain protocols or functions. Adherence to the mandatory state diagrams and portions of the text needs to be verified for every component and device before a statement of conformance may be issued.

IEEE 802.3 includes at the end of every chapter, or clause, a Protocol Implementation Conformance Statement (PICS). The PICS section allows the supplier to fill out a form indicating which options and mandatory portions of the standard have actually been implemented for a particular device or component. The supplier of any component or system that is said to conform to a particular clause or set of clauses must fill out the PICS associated with each clause that is relevant for that device. The PICS sections include a unique item for all mandatory features of the specification. Each PICS item should have an associated *shall* statement, and each *shall* statement should have an associated PICS item.

It should be stressed that even though these PICS forms do exist and are completed by the supplier of a device, it does not guarantee that the device is conformant to every item that has been checked off. In many instances, it is desirable for an independent third party to verify the legitimacy of such a claim by performing a set of conformance tests, which are usually based on the PICS for the device in question. Additionally, in any given full system, there are likely to be components from a number of different suppliers, each of which needs to be compliant to the respective part of the standard. Whether it is an optics module, SERDES chip, MAC chip, or another component, testing must be done to verify that the individual components conform to the standard. When all of the components are put together into a full system, it is imperative that all PICS items are re-evaluated to ensure that conformance has not been compromised due to board layout, power, thermal, or other problems that may arise when the

components are incorporated into the system. Testing and verification is necessary at the component and system level to provide proof that a device is truly conformant.

### **Conformance of transmitter**

Testing the transmitter of the Device Under Test (DUT) for conformance is usually the easiest type of testing to perform. Through the use of monitoring equipment (traffic monitor, oscilloscope, logic analyzer) it is possible to capture the various waveforms transmitted by the DUT and analyze the waveforms at many different levels.

The analog waveforms can be captured directly on a sampling or real-time oscilloscope and analyzed to verify that the DUT properly meets all electrical or optical transmitter requirements. It is common to perform transmitter tests such as mask testing, jitter, and power/voltage requirements.

Moving up a level, equipment such as logic analyzers or certain types of specialized test equipment can capture the DUT's transmissions and analyze them at the digital level by looking at each bit that is sent out on to the medium. This type of testing allows the tester to verify low-level protocol and coding conformance.

The testing can be further abstracted by capturing and analyzing the frame level transmissions from the DUT. Depending on the type of traffic and PHY, freely available software such as Ethereal may be used, or more specialized hardware such as network sniffers may be used to perform the

analysis. This type of testing is necessary to verify that frame level protocols are properly functioning.

### **Conformance of receiver**

Receiver conformance testing is usually much more difficult to perform than transmitter conformance testing. The main reason for this is that specific waveforms, bit patterns, or frame sequences often need to be created in order to test the receiver of the DUT. Depending on the technology or the level of access that is needed, this can sometimes be very difficult to accomplish.

For example, in order to fully test the analog receiver of the DUT, many standards define receiver sensitivity tests that usually include the creation of some type of worst-case analog waveform to then apply to the receiver of the DUT. It is often difficult to both create and calibrate this waveform, and very expensive and specialized equipment is usually needed.

For bit level protocol testing, commercially available test equipment is very hard to come by, and most of the testing performed is done using simulations instead of actual devices and test equipment. Highly specialized test equipment is needed to perform these types of tests, and this is one area in which the UNH-IOL has made a name for itself over the years, as it has helped develop bit level test equipment and services that are not available anywhere else in the world.

At the frame level, receiver testing is somewhat easier, as there are usually commercially available traffic generators that can be used to create certain frame sequences. Companies such as Spirent, IXIA, and Agilent, create dedicated traffic generation hardware for this type of testing. Depending on the

hardware and software available, there may be certain limitations on the types of traffic that can be created.

### **Stimulus and response testing**

In the previous two sections, descriptions were provided on how to test the transmitter and receiver of the DUT. Although passively monitoring the transmitter or providing a test sequence to the receiver is useful and necessary to perform some testing, it is often necessary to combine both transmitter and receiver testing together with stimulus and response testing.

This type of testing provides a specific pattern to the receiver of the DUT (stimulus) and looks for the correct action or reaction from the transmitter of the DUT (response). Such a test covers both receiver and transmitter functionality and conformance. Especially when attempting to comprehensively test finite state machines within the DUT, this type of testing is needed to walk the DUT through the different states and through the different paths that are available.

In addition, it is possible to provide other types of stimuli to the DUT, beyond providing a test pattern to the receiver. The tester can initiate some type of stimuli using the provided management interfaces, including issuance of a reset command and enabling or disabling a function. Other stimuli could include varying the power provided to the DUT, or changing the ambient temperature and humidity conditions.

### **Positive and negative testing**

While it is very important to make sure that the DUT acts or reacts properly when it receives a validly formed waveform, bit stream and frame

sequence, it is equally important to verify that the DUT acts or reacts properly when it receives an invalid waveform or bit stream or frame sequence. These two types of testing are referred to as positive and negative testing, respectively.

Whereas there are usually only a small number of options when creating a valid sequence, there are many more possibilities to create an invalid sequence, and it is often not feasible to test all of them. Therefore, the tester must determine a prioritized list of test vectors that are to be used to cover a necessary percentage of both the positive and negative test cases.

One quick example of this type of testing is as follows. Suppose there is a one-byte field within a frame that has 6 defined values, and the remaining 250 values are invalid. Upon reception of a frame with a valid field value, the DUT should perform some action X, and should perform some action Y upon reception of a frame with an invalid field value. At a minimum, the tester should verify that action X occurs when each of the 6 valid values is received by the DUT. In an ideal situation, the tester would also be able to test all of the invalid cases. However, depending on the difficulty of the test or the amount of time it takes to perform, it may not be possible to test all 250 invalid values. At this point, the tester must make a decision on which invalid values will be tested. One possible solution would be to test invalid values that were one bit different from the valid values. This would cover a decent number of cases, and would account for all instances in which a single bit was in error. Other options also exist, and detailed knowledge of the test and the invalid values that are most likely to exist is necessary to fully understand the problem and create the best set of test vectors.

### **Definition of interoperability**

Two or more devices are said to be interoperable if, under a given set of conditions, the devices are able to successfully establish, sustain, and, if necessary, tear down a link while maintaining a certain level of performance. This definition is somewhat more problematic and complicated than the definition of conformance. In order to claim the interoperability of a set of devices, it is necessary to first establish an accepted set of criteria that will be used to judge these claims. The set of criteria may include: definitions of the communications channel over which interoperability testing will take place, specifications of the type or amount of data that will be transmitted and received over the channel, events that are triggered when certain defined states or conditions have been reached or completed, and the level of performance at which these criteria must be maintained. A common set of guidelines must be developed and accepted by the industry as a whole so that claims of interoperability from one vendor will have been made under the same circumstances as another competing vendor, and thus allow the end-user to fairly evaluate one product over another.

While a standard may not always explicitly define these criteria, the conditions under which interoperability must exist can and should be derived from the standard. In many instances, the standard will define the worst-case conditions over which a device must be able to properly operate. This is usually written in such a manner as to define a particular Bit Error Ratio (BER) that must be supported over these conditions. However, it should be noted that the worst-case conditions do not always exist on a given link between two devices, nor are

they always defined as realistic conditions. Additionally, the statement that two devices work under worst-case conditions does not necessarily imply that the two devices will work under all conditions, including those conditions that may be less stressful than the worst-case conditions.

Using IEEE 802.3 as an example, it is typically an external organization that defines an initial set of interoperability criteria. Recently, during the development of IEEE 802.3ae, 10 Gigabit Ethernet, a joint effort between the UNH-IOL and 10 Gigabit Ethernet Alliance (10GEA) created documentation<sup>2</sup> specifying the means and metrics by which 10 Gigabit Attachment Unit Interface (XAUI) devices should be tested. The document specifies the channel over which testing is to be performed, the data that will be sent over the channel, the duration of the test, and the pass/fail criteria of the test. This document was used as the basis for all XAUI interoperability testing, having been defined and agreed upon by a large group of participating companies and individuals, and thus supplying the industry with a common set of criteria from which to judge interoperability.

### **Interoperability and conformance**

As previously stated, a device having either conformance or interoperability status does not necessarily imply that the other status also exists. A device that claims to be conformant should, by definition, have implemented all of the mandatory portions of the standard. Although the standard may define the interfaces of a layer and its requirements, it does not define nor make an attempt to define how such interfaces and requirements are implemented by a designer.



Various implementations are allowed to exist to, among other reasons, promote competition, and in many cases it is necessary for designs and implementations to be available from multiple sources before a technology can be successful. With the ability and desire to have multiple implementations comes the potential to have implementations that, although individually conformant, are not interoperable.

The ability to implement multiple options is another inhibitor of interoperability. As the number of optional features increases, so, too, does the risk of having interoperability problems. There are some options that, whether implemented or not, will have no impact on interoperability. For example, IEEE 802.3 Annex 31B within IEEE Std. 802.3–2005<sup>3</sup> defines a frame-based flow control protocol. All frames transmitted in this protocol must not exceed a size of 64-bytes. When a device is receiving one of these frames however, it may optionally accept protocol frames that are larger than 64-bytes in length. The implementation of such an option will not impede the interoperability between two conformant devices that only transmit 64-byte frames. If one device does accept the larger frames and the other device does not, there will be no problems observed due to this difference.

Other options may exist that do have a large impact on the interoperability of two devices. It is possible for two conformant devices that have implemented options differently to have interoperability problems. IEEE 802.3ah-2004 specifies an optional mechanism for Forward Error Correction (FEC) that can be implemented on an EPON. The draft implies that the FEC may be used by the

both the Optical Line Terminal (OLT) and Optical Network Unit (ONU), one of the two devices, or neither of them. These four different options would provide for four very different examples of interoperability. In the first example, when both the OLT and ONU have implemented FEC, the results will be the best, and the two devices will interoperate over the greatest length of fiber, or similarly, with a higher split ratio. In the other cases, when one or neither of the two devices has implemented FEC, the ability to interoperate over the same conditions as previously stated will have been altered. It is likely that a shorter length of fiber or fewer number of splits would be necessary in order to operate at the same BER as in the first case. Obviously, this option has the potential to significantly impact the link between the OLT and ONU, and therefore great care must be taken when deploying a network of this type such that the number of available options is either reduced or clearly presented to the end-users.

It is also possible to have two devices that are able to interoperate but are not conformant to the standard. There are multiple scenarios in which this statement can be made true. First, it is possible for two different devices to have implemented a very important mandatory feature incorrectly, but in a similar way. For example, if two devices both reversed the bit ordering of their frames, then they would be obviously non-conformant and highly unlikely to interoperate with other conformant devices. However, when connected to each other, the two devices would interoperate as if they both were conformant to the same standard, and it is possible that the users may not even recognize the non-conformant behavior. An additional scenario could be that the two devices are

able to interoperate but are non-conformant with features that are unrelated to interoperability. For example, in Clause 57<sup>1</sup> the Operations Administration and Maintenance (OAM) protocol requires that at most ten OAM frames be sent each second in order to keep the OAM link alive and provide the periodic feedback of information from one device to another. If one device, or both of them, were to violate the maximum count of ten OAM frames per second and increased that value to eleven or twelve, then although strictly non-conformant, the two devices would still interoperate perfectly fine. There is no part of the OAM protocol itself that would break or cease to function if additional frames were received each second. This introduces the concept that not all mandatory portions of a standard need to be treated equally.

### **Necessity of Conformance**

Throughout the development of a complete system, there are hundreds if not thousands of conformance checks that need to be validated. As shown above, it is clear that the proper implementation of some of these features is more important than others. That being said, it is often difficult to determine what features need to be implemented as specified in the standard so that interoperability problems will not arise. The answers to these questions can only be found through exhaustive conformance and interoperability testing of a wide range of products and implementations. When two devices fail to interoperate with each other, one of the first steps in debugging the problem is to evaluate the results of the conformance testing. Over time, the database of information gained from collecting conformance results of a variety of products will allow

testers and users to not only determine which conformance issues will affect interoperability, but to also predict certain interoperability problems based on conformance results, and to predict conformance problems based on interoperability results.

For example, in the early days of Gigabit Ethernet, observations were made between certain devices that roughly half of all frames transmitted from one device to another across the same optical link were dropped. Both devices were observed to interoperate perfectly with other devices, but not with each other. The cause of the problem was discovered through the conformance testing. The Clause 36 PCS<sup>3</sup> allowed for frames to be transmitted with either six or seven bytes of preamble, which are the beginning bytes of the frame that have traditionally been used for synchronization of the receiving clock. Certain devices implementing this PCS, however, were not capable of receiving frames with six bytes of preamble and would discard those frames. On a link that sent randomly sized traffic, approximately half of the transmitted frames could be sent with six bytes of preamble, thus accounting for the large frame loss. The conformance testing provided the cause of the lost frames, which was an observable interoperability problem. After this discovery, it was observed that in most instances that the frame loss occurred, the conformance issue also existed, and in cases that the conformance issue existed, the interoperability problem was observed.

For those features that are clearly defined in the standard, conformance testing is fairly straightforward and interoperability issues that arise from those

features can usually be explained in a timely fashion. An even greater problem lies within those features that are not dealt with by the standard. For various political and technical reasons, a single standards body may not define some features of a technology. In many cases, it may be considered out of the scope of what the standards body is allowed to do. When this occurs, it is important that all interested parties come together to define implementation agreements that can be tested against and followed to maximize the likelihood of interoperability. One such example, the development of pluggable optics modules, can be shown through a number of Multi-Source Agreements (MSA) such as XENPAK, XPAK, and X2.

It has been shown that conformance and interoperability, while not interchangeable concepts, are nonetheless related to each other in an often strange and difficult to define manner. It is clear, however, that one of the keys to identifying and solving interoperability problems is defining and implementing a comprehensive set of both conformance and interoperability tests. The development and acceptance of these test procedures and the observations of the test results can be powerful tools to aid in the documentation, analysis, and solution of various interoperability problems.

## **CHAPTER III**

### **EPON INTRODUCTION**

The chapter provides the reader with a basic introduction to EPON, and seeks to provide a number of references and resources that contain additional information about EPON. It is recommended that the reader have at least a cursory understanding of EPON before attempting to read the rest of this thesis. In future chapters, the level of detail necessary to describe some of the testing techniques may be too much for those unfamiliar with EPON.

#### **History of EPON**

Although officially standardized during the summer of 2004, the concepts and much of the work relating to PONs in general and EPON specifically had gone on for many years prior to the actual standardization. The concept of passive optical networks has been around almost two decades, and there has been active research, development, and deployment of PON for just as long [5].

More recently, a number of papers and books have been published, signifying that research into EPON is still valuable to the academic and technical community [6][7][8]. These works show that EPON is an adaptive and flexible system that can and will meet the varied needs of service providers and users for many years to come. Just months prior to publication of this thesis, the IEEE 802.3 Working Group created a Study Group to investigate higher speed EPON technology following a successful Call For Interest (CFI) sponsored by the author

of this thesis [9]. Specifically, the Study Group was chartered to look at 10 gigabits per second in the upstream and/or downstream directions. EPON will be a very successful access technology that can provide the bandwidth and services that are demanded.

The EPON standard defines a series of specifications that create a passive optical network infrastructure that carries traditional Ethernet MAC frames while utilizing a MAC Control sublayer that controls the newly defined Multi-Point Control Protocol.

### **EPON basics**

An Ethernet passive optical network is comprised of four main components: OLT, ONU, passive optical splitter, fiber plant. Each of these components is a necessary part of the EPON architecture and will be described in the following sections.

#### **OLT**

The Optical Line Terminal is the device responsible for controlling the entire network, and is analogous to the CMTS in a DOCSIS network. Responsibilities of the OLT include, but are not limited to: bandwidth allocation to the various ONUs, registering new ONUs, keeping track of all ONUs currently connected to the PON, and transmission of all downstream traffic. In a typical EPON configuration, all traffic generated by an ONU is first transmitted to the OLT. The OLT then determines where to forward this traffic (either upstream or back downstream). In addition, it is the OLT that is connected to the Internet and to any other services that may be available, such as voice or video. Of particular

interest in today's market are networks that can provide the triple-play services of voice, video, and data.

The OLT keeps its laser on continuously, using a transmission wavelength of 1480 nm to 1500 nm, and all downstream traffic is received by each ONU; the ONUs then apply different filtering methods to determine whether or not to pass the frames up to higher layers. The receiver of the OLT must be able to synchronize to the different optical signals from each of the ONUs that may be on the network through the implementation of a burst mode receiver.

The OLT typically resides in the central office (CO) of the service provider, and is the most expensive piece of active equipment needed in the PON due to the amount of intelligence needed to deal with all of the complexities found in the network.

### **ONU**

The Optical Network Unit typically resides at the customer's premises, and may support one or more logical links to provide access to one or more subscribers. The ONU is analogous to the cable modem in a DOCSIS network. The ONU contains functionality to queue user traffic and request bandwidth from the OLT. The ONU also has the ability to automatically attempt to register and join the network once it is powered on. Each ONU turns its laser off when it is not transmitting data during a granted transmission window, and only enables the laser when the transmission window has opened. This is accomplished by using a burst mode transmitter.



### **Passive optical splitter**

Central to the concept of the passive optical network is the passive optical splitter. The optical splitters are what create the point to multipoint aspect of EPON. There is typically a trunk fiber coming from the OLT in the CO that connects to one or more splitters at various distances from the CO. To support 16 ONUs, one implementation may use a single 1:16 splitter, and another implementation may use 8 1:2 splitters. Although the standard defines split ratios of 1:16, other split ratios may be used provided that the optical power budget is not violated, or if components and devices with sufficient margin are used. For example, some service providers use connectorized splitters, while others use fusion-splicing techniques to reduce loss at the splitter.

### **Fiber plant**

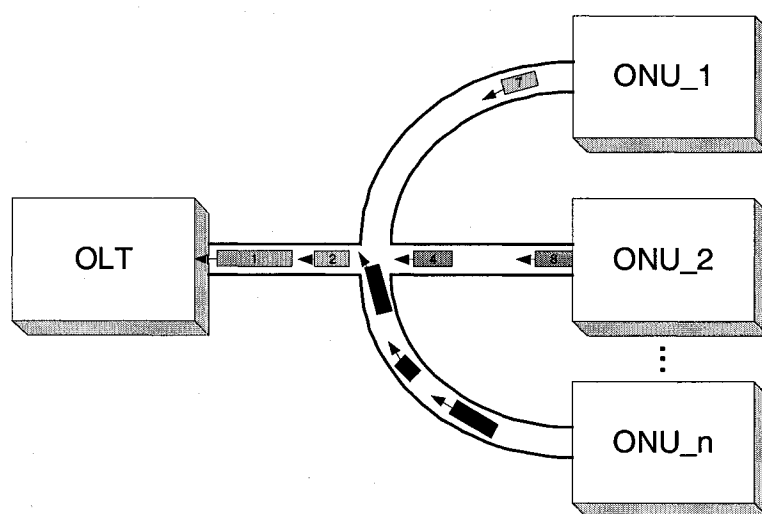
There are two fiber plants defined by the standard, one supporting operation up to 10 km, and one supporting operation up to 20 km. The fiber plant consists of a single strand of fiber, and the ONU and OLT each transmit over the same fiber, at the same time, but on different wavelengths. Standard single mode fiber (B1.1) and low water peak fiber (B1.3) types are included in the EPON specification. The channel insertion loss, including splitters, can be as high as 20 dB for the 10 km solution and up to 24 dB on the 20 km solution.

### **EPON Operation**

The protocol created for EPON, MPCP, defines a number of message types that are used to setup, maintain, and tear down the connection between an OLT and an ONU. The messages defined are all MAC Control messages, and

are therefore generated and parsed by the MAC Control sublayer within the device. These messages are utilized by the EPON discovery process along with the bandwidth allocation algorithm to allow data to pass from the OLT to the ONUs and from the ONUs to the OLT.

At its heart, the EPON protocol uses a fairly simple request and granting mechanism. Each ONU requests bandwidth when it has data to transmit, and the OLT is responsible for granting the ONU an individual time window during which it will be the only device transmitting in the upstream direction. In the downstream direction the OLT continuously transmits frames that are received by all ONUs and then filtered so that only the correct ONU passes the frame up to the higher layers. Figure 1 shows the normal upstream operation of the network.



**Figure 1 - Normal upstream operation**

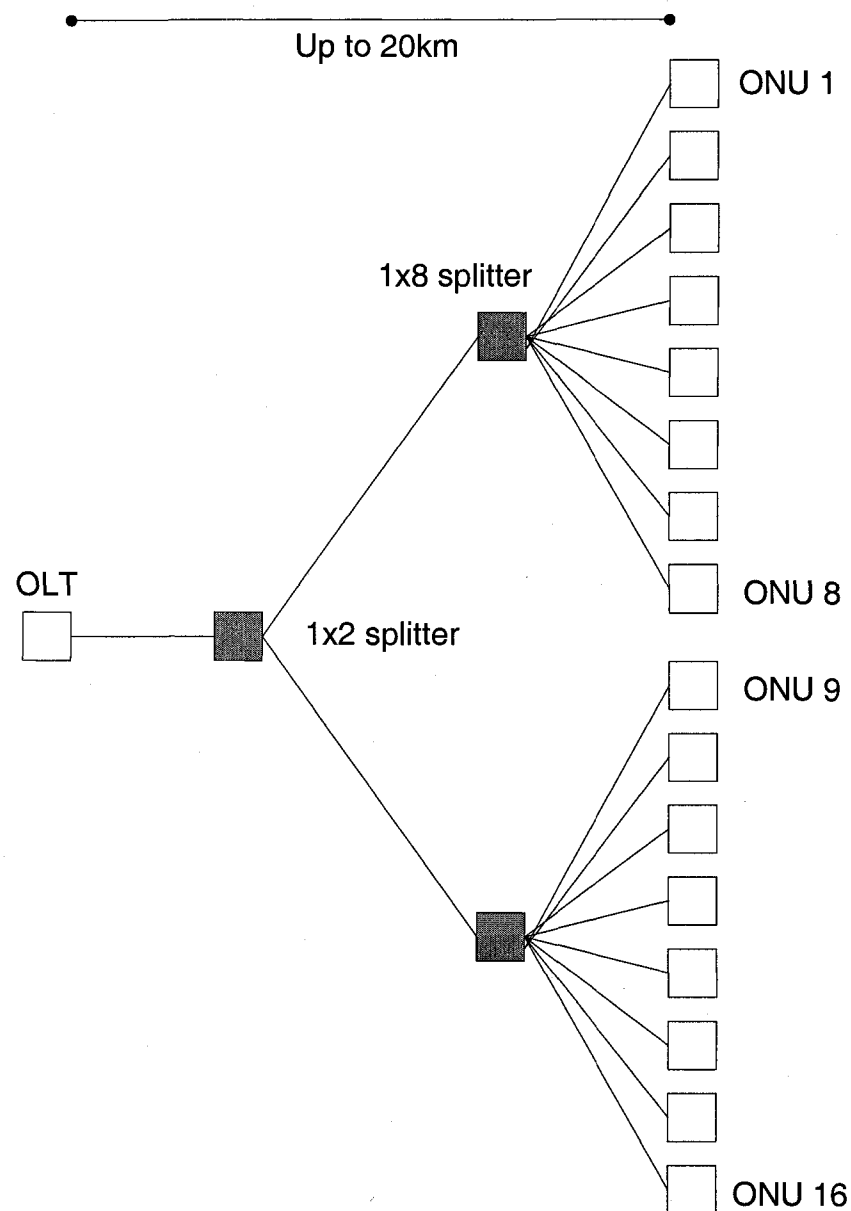
## **EPON architecture**

Figure 2, shown below, displays one example of EPON architecture. In this example, the OLT connects to a 1x2 splitter, and each output of this splitter connects to a 1x8 splitter. Those sixteen outputs each connect to a single ONU. The standard makes no comment on what type of architecture should be used (ring, bus, tree), how many splitters should be employed, and what types of fiber distances can be installed. Therefore, there exist almost an infinite number of potential architectures on which the EPON can function. In addition, by minimizing certain losses or using components with higher power transmitters or more sensitive receivers, extra lengths of fiber and/or additional splits may be possible other than 10 km/20 km and 1:16-split ratio.

However, once a service provider agrees upon and chooses to install a particular architecture, it is highly unlikely that this architecture will be modified in the future. The costs associated with training employees, maintaining specific inventories, and supporting an existing architecture require that the architecture be used for as long as possible and for as many upgrades as possible. As will be discussed later, one of the requirements that service providers are demanding for higher speed EPONs is the ability to re-use the existing infrastructure they have already deployed for traditional 1 gigabit EPON.

The flexibility associated with the EPON architecture makes it an ideal candidate for mass deployment in a variety of service provider networks. Some providers may choose to place all of the splitters within the central office, and others may choose to install the splitters closer to the location of the users.

Some providers may choose to use connectorized splitters, whereas others may choose to use fusion spliced splitters in order to minimize the loss.



**Figure 2 - EPON architecture example**

Finally, the architecture defined for EPON is very similar to the architecture defined for other PON technologies. This provides the potential for additional upgrading from non-EPON technologies to EPON.

## **CHAPTER IV**

### **EPON TEST TOOLS**

This chapter describes the test tools currently available within the UNH-IOL that can be used for EPON testing. Although there are a number of commercially available pieces of EPON test equipment, including an analyzer developed by Agilent, there are significant limitations with the built-in capabilities and functionality of this equipment. In order to comprehensively test EPON conformance, it is necessary to create specialized hardware and software that gives full control over the types of test sequences that can be transmitted and received.

In particular, there are three types of tools that have been created and are used for EPON conformance and interoperability testing. The first type is a tool that can emulate the functionality of either an OLT or ONU. This tool is then connected to the DUT and performs all of the necessary positive and negative testing for all protocol and coding layers. The second type of tool is a passive monitoring tool that can be used to monitor both the upstream and downstream traffic on the EPON. Analysis of the traffic can then be performed to help understand and eliminate interoperability problems between the OLT and ONU. The third tool is used to perform the optical PMD testing for both the OLT and ONU. This tool interfaces to standard test equipment and runs automated scripts to capture and analyze the analog waveforms from the DUT.

## OLT/ONU Emulation Tools

The ultimate wish for the OLT/ONU Emulator is that it will be a fully automated test system that can run user-defined test scripts, perform complex receiver triggering, collect all results, and generate a report. Figure 3, shown below, provides a high level description of the emulation tool.

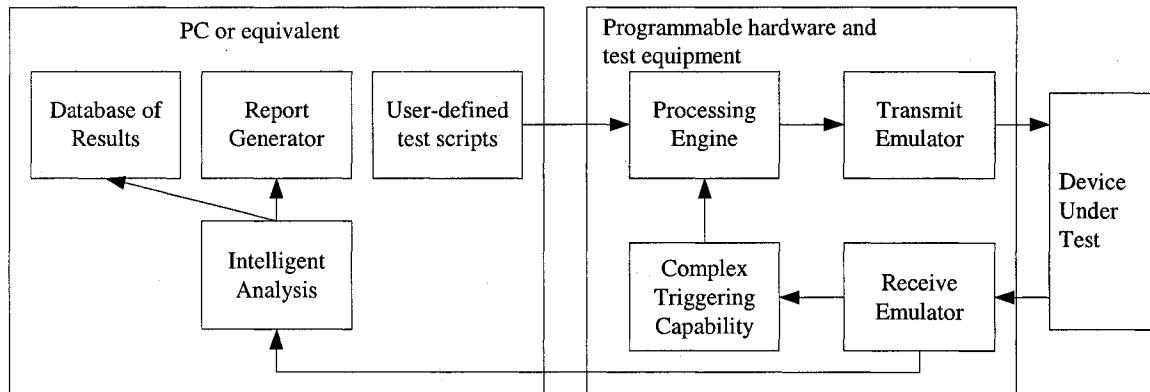


Figure 3 - High level EPON conformance tool

### Database of results

The database is used to store results obtained during testing. For each test that is performed, the database should contain the exact test vector transmitted as well as received so that there will be no confusion as to what was actually tested. Since it is expected that tests and test vectors will change over time, it becomes especially important to have access to the original results if new testing is performed on the same product that was previously tested with an earlier version of test scripts. The database can also store information about the test setup and equipment used during testing.

### **Report generator**

The report generator is used to take the test scripts, received patterns, and the results of the intelligent analysis to create a detailed technical report that provides all of the results captured during testing. The reports are then issued to the owners of the DUT and can be used for various marketing or technical purposes. The reports are based on the common UNH-IOL report format that is recognized within the networking industry.

### **Intelligent analysis**

Automated intelligent analysis of data is very important yet it is difficult to achieve. The module takes in the test script and looks at what has been received by the test tool. It then needs to make a determination of whether or not the correct behavior was observed. The difficulty lies in observing behavior that may be odd or behavior that is not normally expected. For these cases, it is often best to visually examine the received sequences to determine whether a test case has passed or failed.

### **User-defined test scripts**

The user-defined test scripts are one of the most important parts of the test tool, as they allow the user to define scripts at a number of different interfaces or layers. The test scripts can be used to define sets of frames to be transmitted, or they can be used to define specific binary bit patterns for the test tool to transmit to the DUT. The user has complete flexibility and control over the types of patterns and scripts that can be created within the limitations of the hardware platform provided.

### **Processing engine**

The processing engine takes the test scripts and prepares them for transmission on the fiber. The processing engine may also take information from the triggering module if it is necessary to wait for certain events at the receiver before beginning transmission. The processing engine can be setup to maintain synchronization by continuously generating an IDLE sequence, or it can be setup to maintain an MPCP link by continuously transmitting the appropriate MPCP frames while waiting for the test script to be created and downloaded.

### **Transmit emulator**

The transmit emulator is needed to convince the DUT that the emulator is actually a functioning OLT or ONU. It must be able to create the appropriate analog waveforms, having been fed the necessary digital waveforms from the processing engine and test script engine. This includes having a functioning optical transmitter module and the ability to generate validly formed and timed IDLE and data streams.

### **Complex triggering capability**

A logic analyzer that is monitoring the bit patterns in both the upstream and downstream directions provides the complex triggering capability. The capability exists to create a multi-state trigger with numerous variables and actions. Triggers can be made to look for specific frames, fields within frames, or even specific bit patterns. The logic analyzer also has the capability to selectively store different patterns and keep a very accurate timestamp associated with each set of bits that are captured.

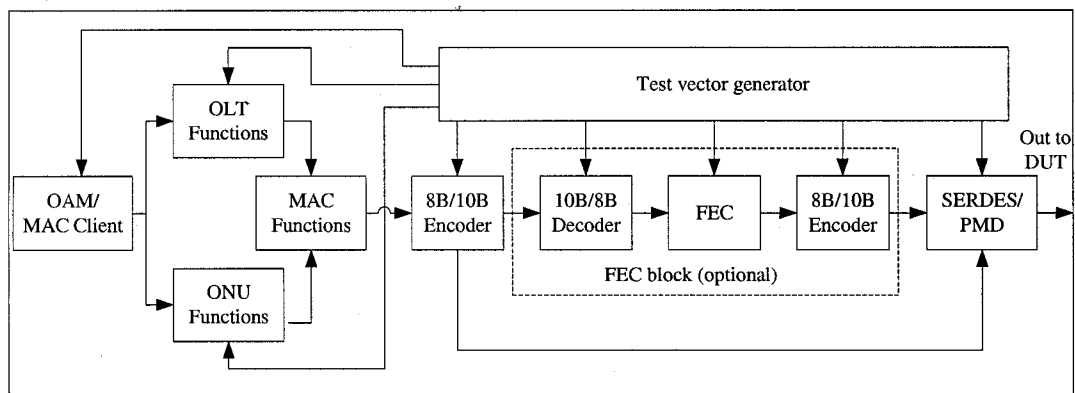


## Receive emulator

The receive emulator is used to provide the necessary receive functionality so that it is able to receive all transmissions from the DUT. For the EPON tool, the receive emulator must contain the appropriate optical module depending on whether the DUT is an ONU or OLT, and must have the necessary physical layer requirements, such as clock and data recovery suitable for burst mode and COMMA alignment.

## Detailed Transmitter tool

Figure 4, shown below, gives a more detailed look at the transmitter side of the EPON conformance emulator tool. It is clear from the figure that test vectors can be created and inserted at almost any interface. Test vectors can be created for OAM, MPCP, and MAC frames, or they can be created at the bit level to be transmitted directly by the SERDES.



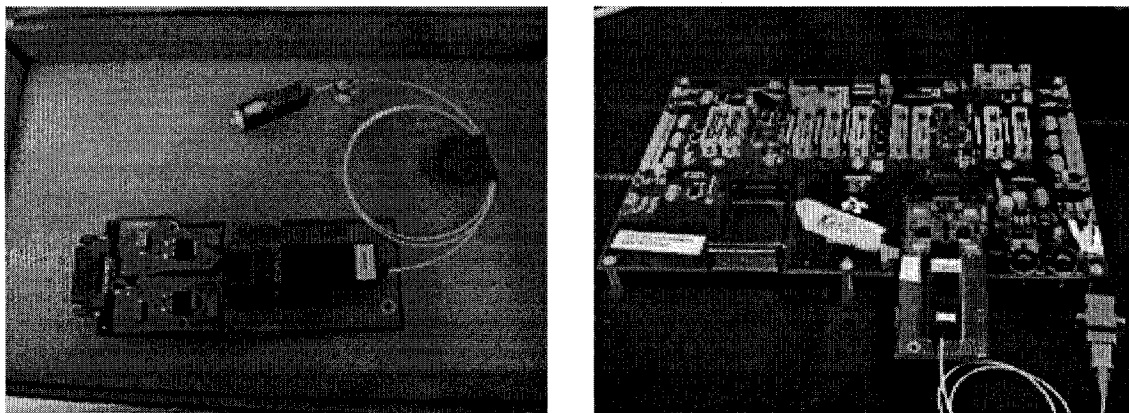
**Figure 4 - Detailed EPON conformance tool**

This type of flexibility gives the tester unique capabilities when it comes to creating many different types of test vectors. In particular, if the optional forward error correction (FEC) block is implemented, then the tester may insert specific

errors at any point in the bit stream in order to fully test the FEC mechanism and correction capabilities of the DUT.

### **Modified TIGER System**

Back in the early days of gigabit Ethernet, a test board was developed for UNH-IOL with the assistance of Texas Instruments. This board, shown in Figure 5, provides the tester with an optical interface to connect to the DUT, and a 20-bit wide interface to and from the SERDES chip that can connect to a logic analyzer and pattern generator. The TIGER board itself provides very little functionality, and primarily serves as a tool to convert an electrical data stream into the optical domain. Most of the intelligence of the test system is contained in the logic analyzer and pattern generator, and the code that controls them. In order to connect to a variety of optical port types, the TIGER board provides the user with a GBIC port so that any optical module can be used in order to test the correct interface on the DUT.



**Figure 5 - TIGER system and optics**

The fact that the TIGER system provides very little native functionality is very beneficial in a test environment. The tester can then create their own functionality, or can pull in functionality from other test tools, without having to rely on a single vendor's solution that may or may not provide the features needed for testing. Although it is common for vendors to provide certain special test modes or functions within their devices, it is not common that any single vendor will provide all of the necessary functionality for doing comprehensive testing, such as the UNH-IOL normally does. The TIGER system allows the tester to generate any bit pattern and to transmit this pattern on the fiber. On the receive side, the tester has access to every bit captured from the fiber.

In summary, the TIGER system provides a testing environment that allows complete control over the creation and transmission of arbitrary bit streams from the test system to the DUT. It also allows the test system to receive a complete and unaltered bit stream from the DUT.

### **Optical transceivers**

One of the parts necessary for developing an EPON test system is to have the correct optical module. When trying to create an OLT emulator, it is necessary to have a burst mode receiver, capable of receiving data streams from the ONU. It's also important to have the correct optical module in order to transmit to the ONU, as EPON systems use a different wavelength than other Ethernet optical devices running at the same data rate. An EPON company, Teknovus, has provided the UNH-IOL with a number of both ONU and OLT

optical modules that can easily be interfaced into the existing TIGER system. A picture of one of these modules can be seen in Figure 5.

### **Logic analyzer solution**

The logic analyzer used with the TIGER system is a Hewlett Packard 16500B system that contains both analyzer cards and a pattern generator. The analyzer connects to the TIGER system through its data pods and pattern generator pods. Since the TIGER board provides a 20-bit SERDES interface, the logic analyzer needs to support a 20-bit bus for both transmit and receive running at a rate of 62.5MHz (the actual bit rate of EPON is 1.25Gbps, due to the 8B/10B conversion).

The 16500B provides the ability to perform very complex triggers on the received signal, and has the ability to either capture an entire received data stream or to selectively filter and capture only certain events. It is this triggering capability that gives the TIGER system some of its most powerful resources.

Although there are certain limitations with both the triggers (limited number of variables and states) and with the pattern generator (limited memory space), the TIGER board and logic analyzer combine to form a very useful and valuable test tool. Control of the system is then offloaded to software so that the user can automate the transmission of test patterns, the reception of data, triggers, and other complicated logic analyzer functions.

### **LabVIEW code environment**

Virtually all of the code written to control the TIGER system and EPON test tool is written using National Instrument's LabVIEW software. The LabVIEW

software package provides an intuitive and easy to use (yet difficult to master) interface for the creation of LabVIEW virtual instruments (VI). Since the TIGER board was first introduced to the UNH-IOL, there have been many programs created and improvements made in order to fully control and best utilize the test system. For this project, a number of minor modifications were made to the base code of the system, and two primary VIs were used to create the set of scripts and test vectors necessary to perform the EPON conformance testing.

### **Test Vector Generator**

The “User Vector to 1652a v4.2.vi” program is shown below in Figure 6 (it should be noted that this VI was not modified during this project). This VI allows the tester to create a set of test vectors that can be uploaded to the pattern generator. These test vectors can be created in one of several ways, depending on the type of test vector that is necessary. The user may enter any 8-bit (uncoded) or 10-bit (coded) word, whether valid or invalid. It is also possible to take advantage of certain built-in features, such as an automated CRC-32 generator. The test vector may be any combination of bits of virtually any length, provided that the length is a multiple of 20-bits. Another useful feature of this VI is that it allows the user to save test vectors into a database so that they can be recalled at any time. As shown in Figure 6, there are a number of test vectors already written and stored in the database.

Although this VI is not helpful for automating testing, it is an invaluable tool when the user wants to manually generate and transmit a series of test vectors. Typically, when first developing a set of test vectors for a given technology, this

VI will be used as it provides the most flexibility to the user. As the test vectors begin to change less and less, they are then migrated into a more automated system. The VI is also incredibly useful when it is necessary to generate specific test vectors or bit patterns that may be requested by a vendor. In many instances, vendors simply do not have the capability to generate an arbitrary bit pattern and actually transmit it onto the fiber. This tool, combined with the logic analyzer, provides a unique solution for doing exactly that task.

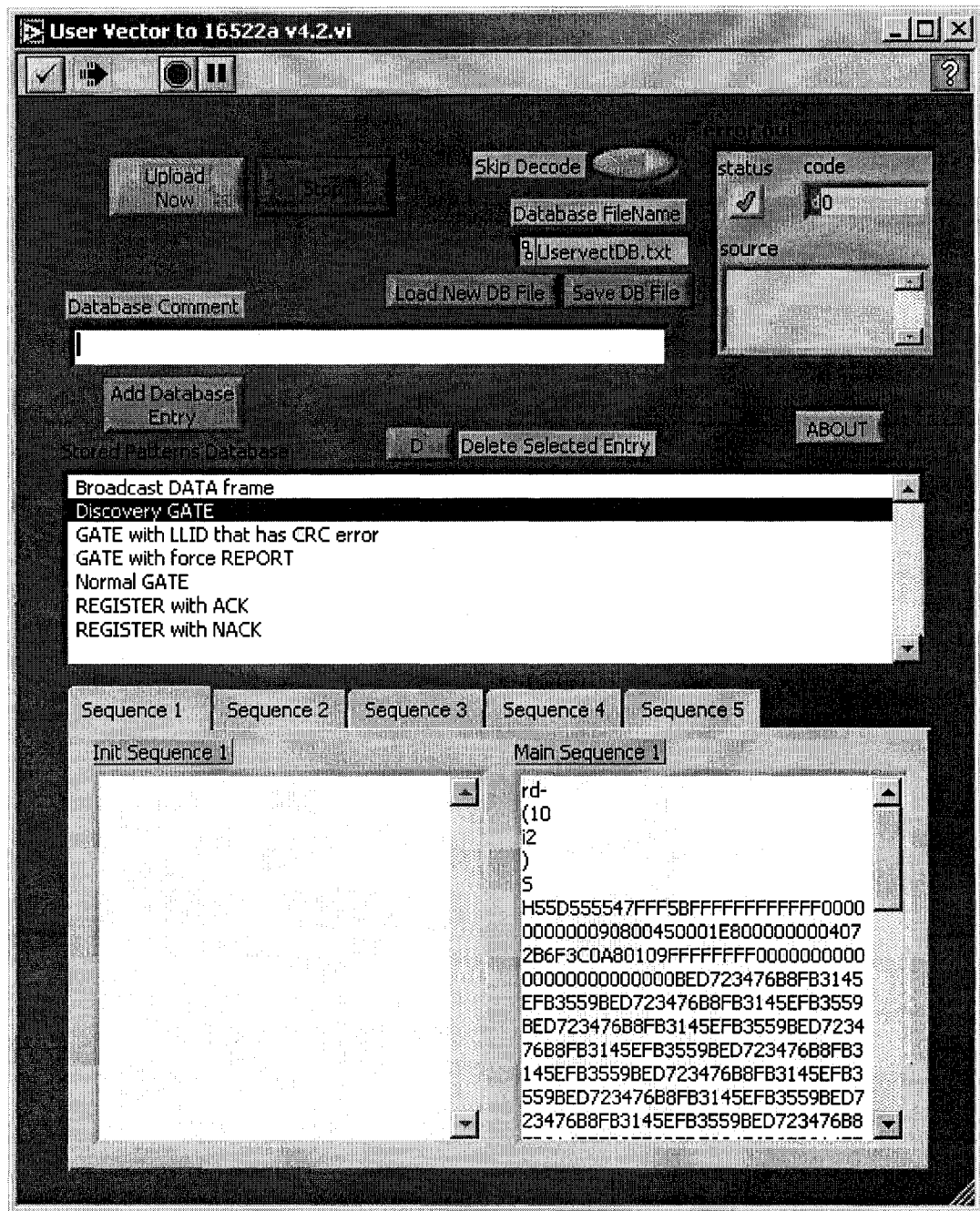
Throughout the course of the project, hundreds of test vectors were created using this tool. Where applicable, these test vectors are reproduced in later parts of this document in order to fully explain how the testing was performed. The full set of test vectors is available through the UNH-IOL.

### **Triggering capabilities**

The triggering capabilities of the TIGER system are showcased in Figure 7. Minor modifications were made to this program during the course of the project. The logic analyzer system used contained two independent “machines”. For some test setups, a single machine was used to observe either the upstream or downstream transmissions from the DUT. For other tests, it is necessary to use both machines in order to observe both the upstream and downstream transmissions at the same time.

This figure shows that there are up to 12 terms, or 10-bit variables, that can be made available between the two logic analyzer machines, and that each machine can build a trigger that has up to 12 levels. Within each level, the user must specify the following: what the logic analyzer should be storing (or not

storing), what the logic analyzer should be looking for (or not looking for), and what the logic analyzer should do when it finds what it is looking for (or does not find). Once the trigger has been created, the user must also specify the amount of memory for the logic analyzer to use, and several other options. The trigger is then uploaded to the logic analyzer. Once the trigger is loaded, it is possible to synchronize both the pattern generator and logic analyzer so that they may both begin at the same time. In addition, it is possible to launch the pattern generator upon reception of a certain trigger event.



**Figure 6 - Test Vector Generation Tool**

Throughout the rest of this document, certain triggers will be referred to in a fairly generic sense (example: Create a trigger to capture all GATE frames). The actual triggers are far more complicated and will not be described in this document.



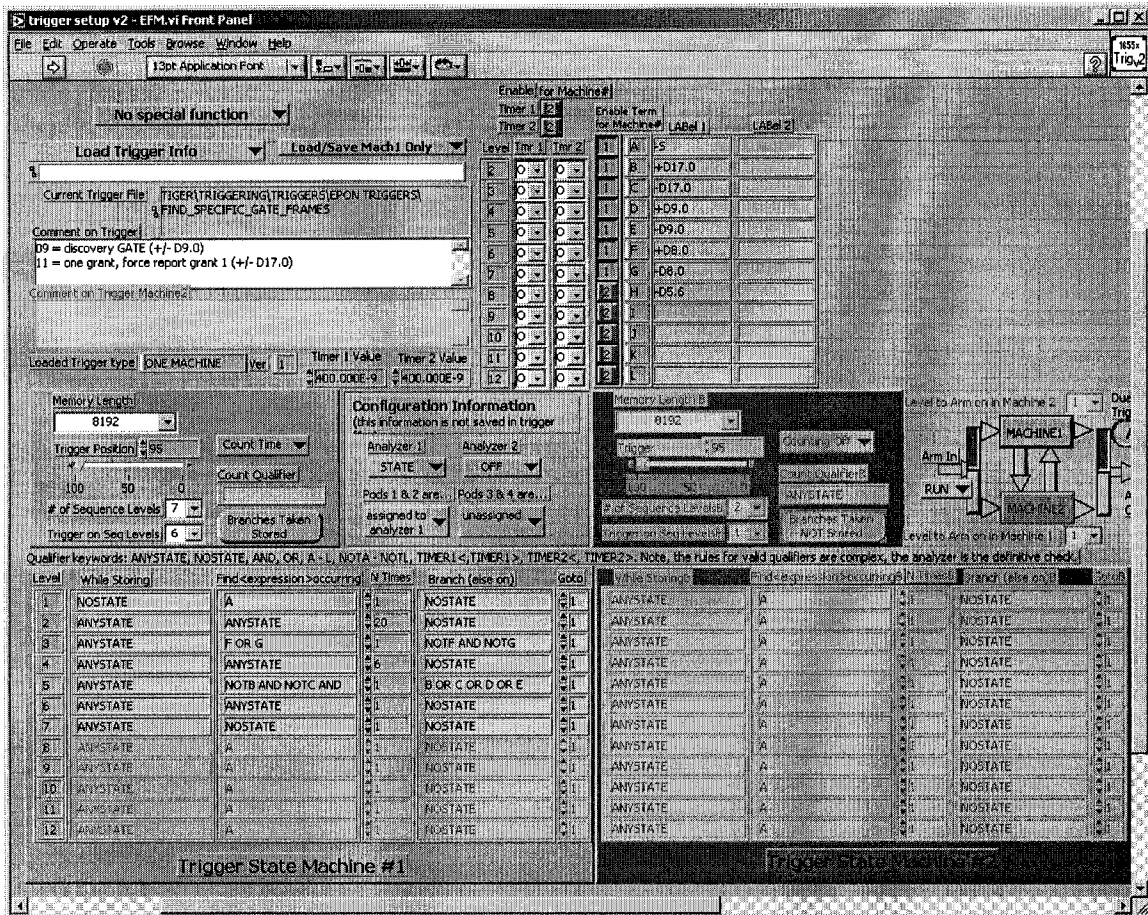


Figure 7 - Trigger program

### Additional EPON VIs

In addition to the previously discussed VIs, a number of LabVIEW programs were created throughout the course of this project that relate to the use of the TIGER system and to the analysis of data obtained by the TIGER system. Throughout the remainder of this document, some of these VIs will be referred to and discussed. Since these programs are primarily involved with the analysis of data, they will be discussed when relevant to specific tests.

### **Future improvements**

There are a number of improvements that can be made to the current EPON test system, and it is expected that these improvements will occur as the level of EPON testing begins to increase in the coming months and years. Since the current test system has not been used many times, it does not make sense to develop a fully compiled test system that will clearly need to undergo changes. Instead, future employees of the EFM consortium will make incremental improvements and changes to the EPON test system.

One improvement that will always be ongoing is that the existing test vectors and software will need to be updated. As more testing is performed and more experience is gained, the test vectors and software will be refined to test more efficiently and correctly. In addition, as more vendors receive the testing and reports, the test vectors will come under more scrutiny, and any invalid test vectors will be weeded out.

The second improvement involves porting the existing TIGER system into an FPGA based platform and providing additional transmission and reception capabilities. This type of project could greatly increase the number of types of test vectors and conditions that could be used to perform conformance testing. It is expected that this project will be a follow-on to the current project.

### **Passive Monitoring Tools**

One of the difficulties in testing EPON interfaces is that, until very recently, there was no commercially available equipment to passively monitor the transmissions in both the upstream and downstream directions. Although certain

equipment is now available, the cost is rather substantial, and therefore other types of monitoring solutions have been created. There are two monitoring solutions now available for EPON traffic within the UNH-IOL. The first uses the TIGER system that has already been described, and the second uses commercially available Ethernet equipment from Spirent Communications.

#### **TIGER System monitoring solution**

Although the TIGER system can be used to monitor EPON traffic in both the upstream and downstream directions, it does not provide the best solution if there is a significant amount of traffic to capture. The TIGER system is very good at capturing and analyzing bit streams, but it begins to falter when it is necessary to capture and analyze long streams of frames.

In order to use the TIGER system as a passive monitor, the user simply needs to connect the fiber to the correct optical module, and then needs to disable the optical transmitter of the TIGER system. By placing the optical transceiver in this mode, it will still be able to receive everything, but it will not transmit anything onto the fiber, and therefore will not interfere with other devices on the EPON. Depending on the types of traffic that the user wishes to analyze, the necessary trigger must be created and uploaded.

#### **SmartBits monitoring solution**

If a TIGER system or its equivalent is not available, it is possible to turn existing Ethernet test equipment into a passive monitoring tool for EPON. For this project, a Spirent Smartbits TeraMetrics card was used to create the monitoring station. This particular SmartBits card has four ports, and each port

may be configured to perform as either a 10/100/1000 Ethernet copper port or a gigabit Ethernet optical port. When operating as an optical port, an SFP module is used. Since most SFP modules require two fibers (one for transmit, one for receive) it is not necessary to disable the transmitters, as they will simply be left unconnected. Also, most 1000BASE-LX transceivers have a fairly wide optical bandwidth on their receiver and are able to receive signals from either the 1310 nm range (transmission window for ONU) or the 1490 nm range (transmission range for OLT).

Using this type of setup, the SmartBits should be able to establish a link and can begin to receive frames transmitted either in the downstream direction or in the upstream direction. In order to get the best performance out of the SmartBits tool, it helps to permanently enable the laser of the ONU, and to ensure that only a single ONU exists on the EPON.

The SmartBits will receive all frames as if they have errors in the CRC-32 field of the frame. This is due to the fact that the modified preamble on EPON frames will confuse the receivers of all traditional Ethernet ports. There is, unfortunately, no way to get around this, but since the SmartBits tool does capture and display frames that have CRC-32 errors, the frames can still be viewed and analyzed.

For this project, LabVIEW software has been written to interface to the SmartBits tool and download all of the captured EPON frames. The frames are then modified so that they look like valid frames, and the frames are saved in a format viewable by standard monitoring tools such as Ethereal. These files can

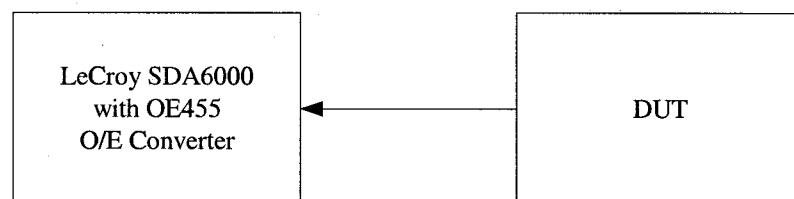
then easily be shared among vendors and provide a useful mechanism to share interoperability and conformance results.

### **Optical PMD Tools**

A number of Matlab and LabVIEW based tools have been developed to facilitate optical PMD testing for EPON. In addition to EPON, these tools may be used to test any NRZ-based optical Ethernet port. There are two main tools that have been developed, one for performing automated transmitter testing, and one for performing automated receiver testing.

#### **Automated transmitter testing**

The real-time oscilloscope used is the LeCroy SDA6000, equipped with an OE455 optical-to-electrical converter. The setup used for testing is shown in Figure 8. Traditionally, sampling oscilloscopes have been used to perform optical measurements, and until recently, these scopes have been limited in their ability to perform certain functions such as making pattern-based measurements. The real-time scope, on the other hand, has the advantage of being able to capture everything being transmitted by the DUT and is limited only by capture depth and bandwidth.



**Figure 8 - Transmit PMD test setup**

The primary tool used for transmitter testing is a Matlab program for which the user interface is software shown in Figure 9. This software runs in

conjunction with a real-time oscilloscope and optical-to-electrical module provided by LeCroy Corporation. The software is based on previous software written by UNH-IOL employees for the purpose of testing high-speed electrical NRZ technologies. The additions and modifications to the software will be described in the following sections.

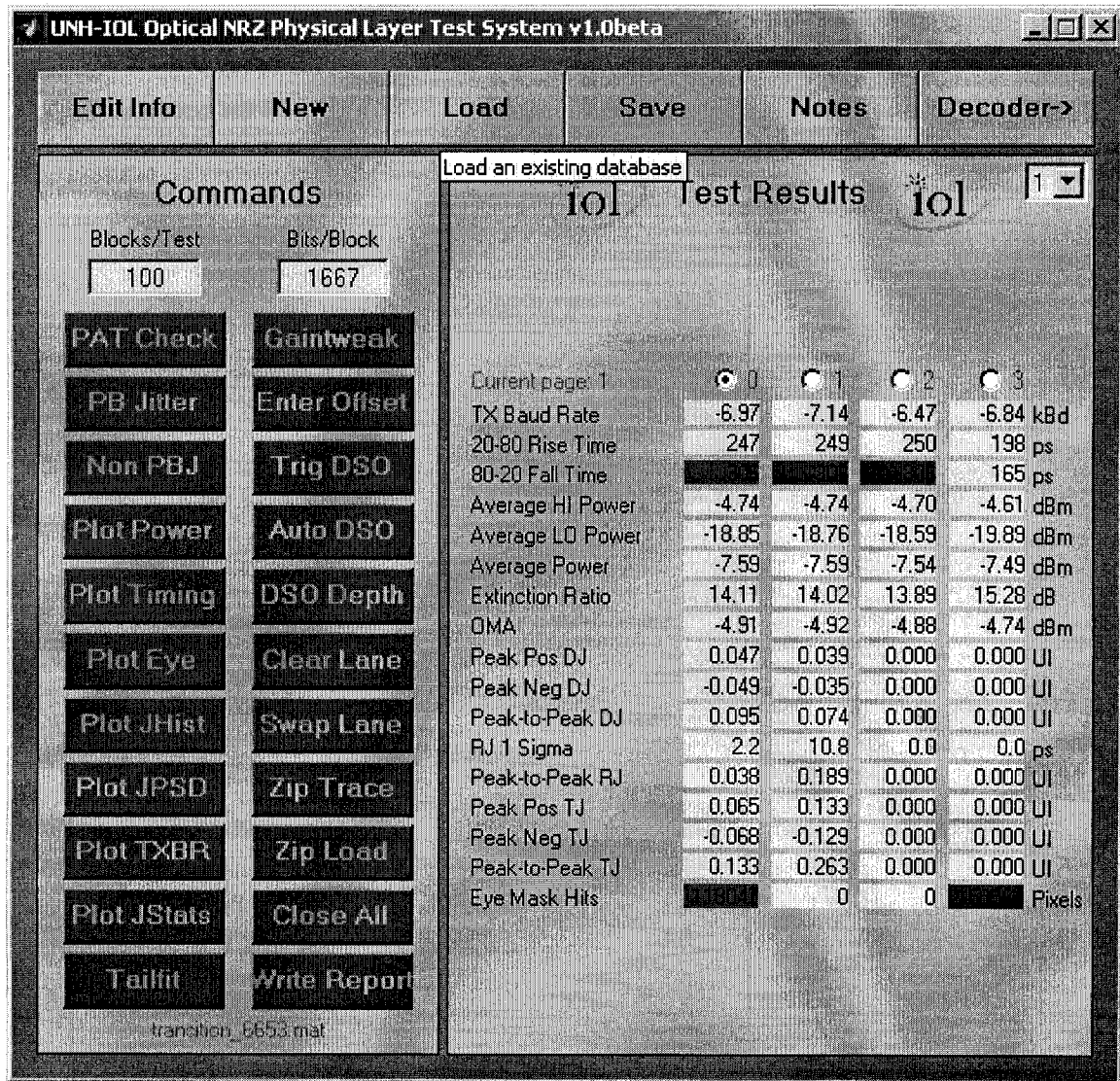
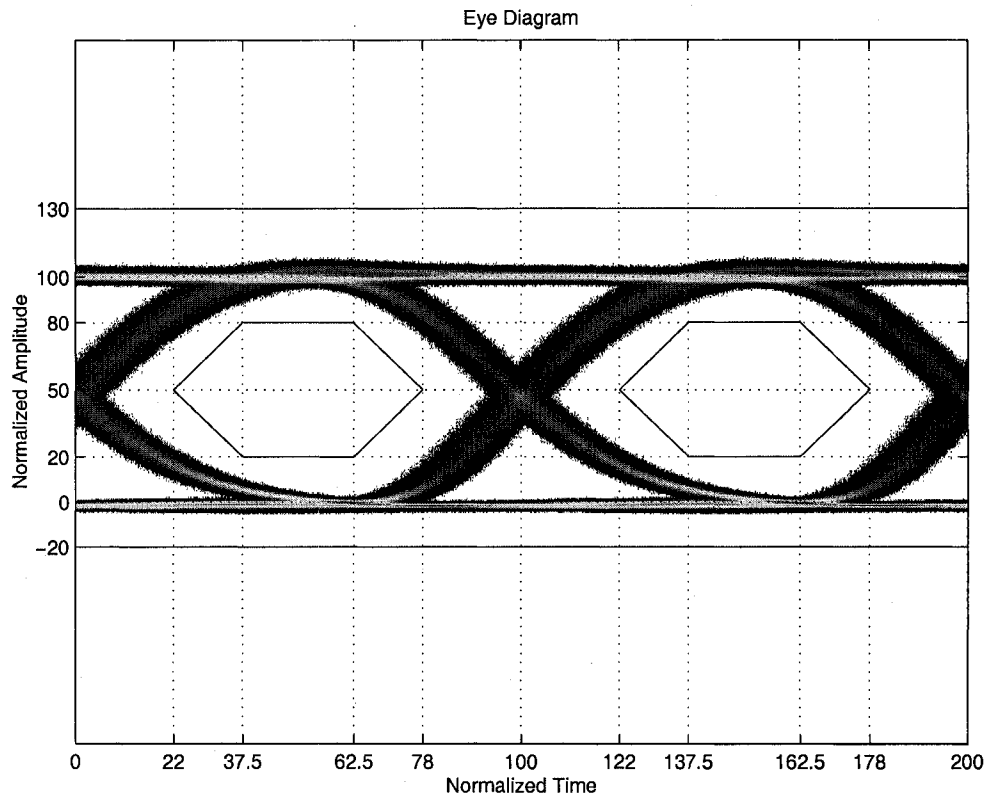


Figure 9 - Transmitter testing program

As seen in Figure 9, there are a number of “Commands” that may be issued, and a number of “Test Results” that may be displayed. For this project, algorithms were developed for the following results: scaled mask for eye diagram, 20-80 Rise Time, 80-20 Fall Time, Average HI Power, Average LO Power, Average Power, Extinction Ratio, and OMA. The eye diagram shown in Figure 10 can be used to perform most of the power and timing measurements. Generally speaking, an eye diagram shows all possible transitions that the signal may take. In the case of EPON, an optical NRZ modulation scheme is used, and therefore the laser will take on a HI or LO power.

Another important aspect of optical testing is mask testing. As shown in Figure 10, the interior of the eye is bounded by a hexagon. In order to achieve compliance, the DUT must not violate the mask at any point. As opposed to electrical testing, when the masks are often based on absolute voltages, optical eye testing is slightly more complicated. For optical technologies, the mask is based on the normalized power received by the test equipment. Instead of using absolute voltages or powers, the entire mask is relative to the actual HI and LO powers transmitted by the DUT. The mean of the HI power is given the relative amplitude of 100, and the mean of the LO power is given the relative amplitude of 0. The mask is then created around these powers and fit within the eye. Figure 10 shows a clean eye with no mask violations, and a decent amount of margin before any violations would occur.



**Figure 10 - Automated Eye diagram**

### **Power measurements**

The ONE power level is the measure of the mean logic one-level. It is found by constructing a vertical histogram in a narrow region about the center of the eye that includes only the one-level data. Often, a 40-60 window is used: the histogram starts at the 40% time point of the eye and stops at the 60% time point. The ONE level is defined as the mean of this histogram and the standard deviation is the RMS noise. The ZERO level is defined in an identical manner, but the histogram is taken around the zero-level data. As shown in Figure 10, the eye diagram contains data from two unit intervals (UI). Each eye diagram contains 1600 vertical columns and 512 horizontal rows. Each column

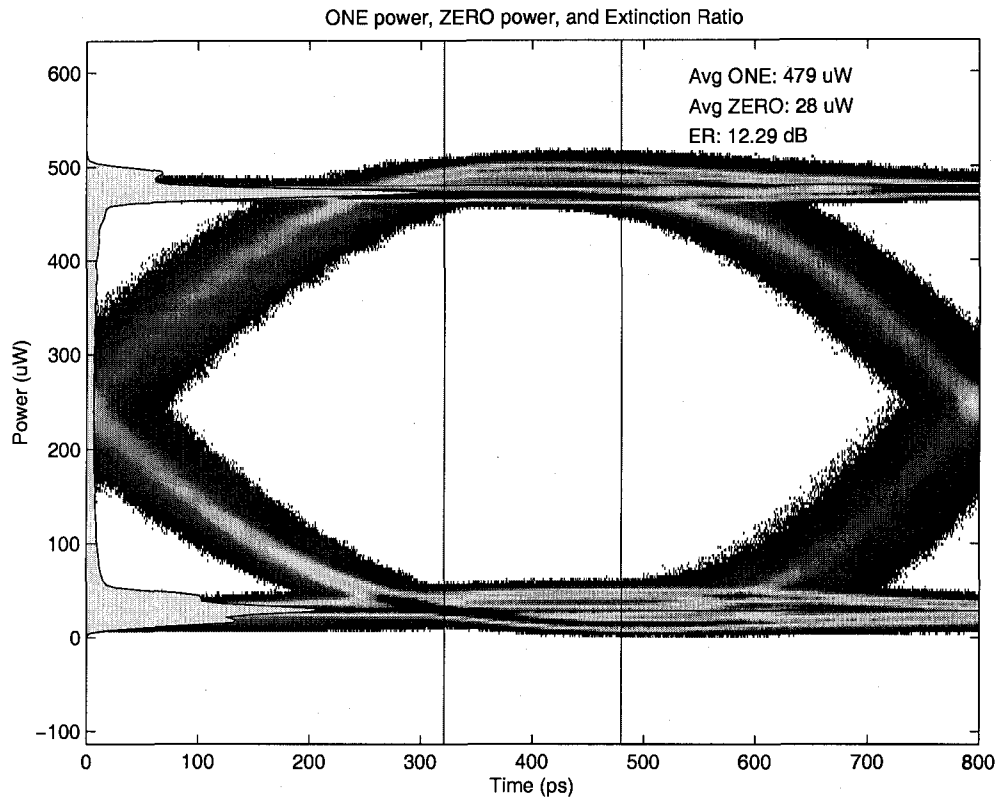


represents 1 picosecond, and each row represents a scaled power value. The first UI goes from column 1 to column 800, and the second UI goes from column 801 to column 1600. The actual columns for each 10% window are shown below, for the first UI.

**Table 1 - Eye diagram distribution**

<b>Percent window</b>	<b>Columns</b>	<b>Percent window</b>	<b>Columns</b>
1 to 10	1 to 80	51 to 60	401 to 480
11 to 20	81 to 160	61 to 70	481 to 560
21 to 30	161 to 240	71 to 80	561 to 640
31 to 40	241 to 320	81 to 90	641 to 720
41 to 50	321 to 400	91 to 100	721 to 800

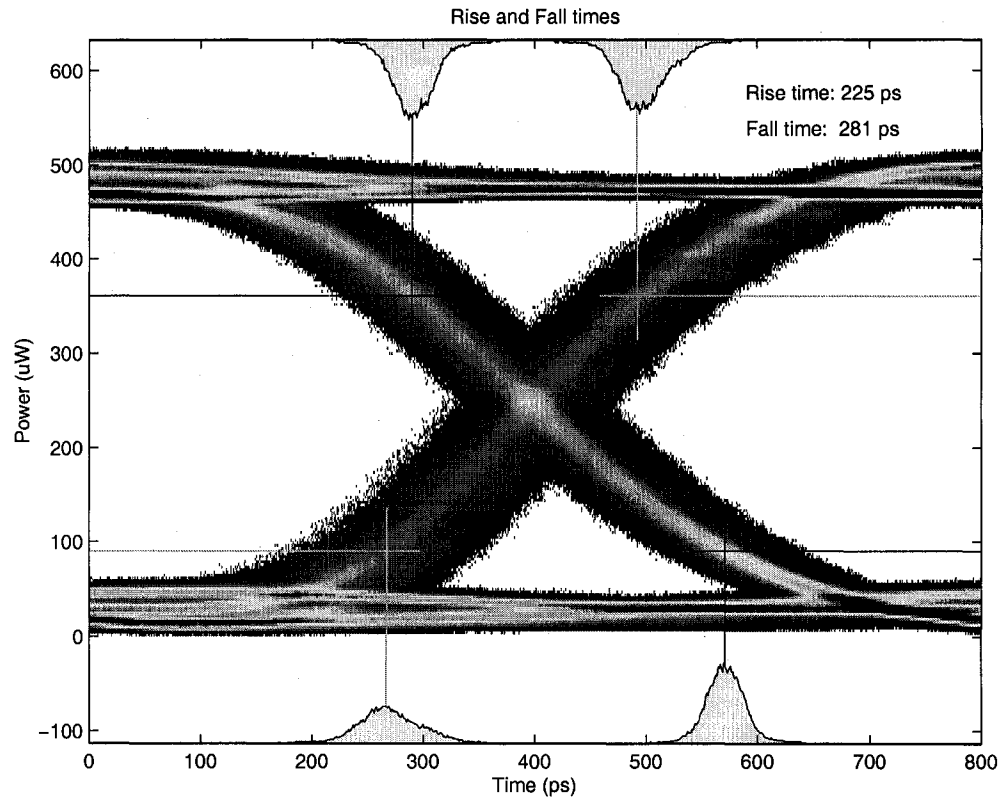
Once the windows are properly established, histograms can be built up around the correct points. Figure 11 shows the histograms around the 40% - 60% point on the diagram. The histograms, shown in yellow, are on the ordinate axis of the figure. For ease of viewing, the diagram contains only a single UI. Simply stacking both UIs of the previous eye diagram on top of each other can create this view. The vertical blue lines demarcate the area in question around which the histograms are based. The means of the histograms are taken to determine the average logical ONE and logical ZERO power levels. From these values, the extinction ratio (ER) can also be determined, as the ER is simply the ratio of the high power to the low power. The optical modulation amplitude (OMA) can also be calculated, as it is the difference between the high and low powers.



**Figure 11 - Automated power measurements**

### **Timing measurements**

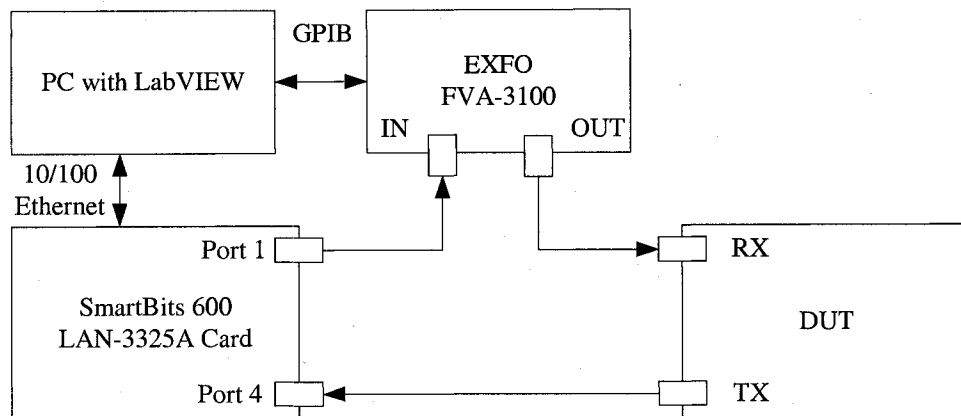
For optical signals, the rise time is defined as the time from when the rising edge reaches 20% of its final amplitude to the time when it reaches 80% of its final amplitude. The rise-time measurement begins by determining the one and zero levels of the eye. Once these levels are known, the time locations of the 20% and 80% thresholds can be determined. From previous power measurements, we already have the average ONE and ZERO powers. We can use this to find the 20% and 80% times and then make the rise time measurement. Figure 12 shows the same diagram as Figure 11, but this time the timing measurements and histograms of threshold crossing times are shown.



**Figure 12 - Automated timing measurements**

### **Automated receiver sensitivity testing**

In order to perform receiver sensitivity testing, a slightly more complicated setup is necessary. Figure 13 shows the setup used for this type of testing. A computer running LabVIEW controls all of the test equipment through GPIB and Ethernet connections. The SmartBits platform is used to generate the frames used for testing and to count the number of frames received. The EXFO device is a programmable optical attenuator that is used to vary the attenuation of the signal being provided to the DUT.



**Figure 13 - PMD Receiver Test Setup**

The methodology employed for receiver testing is as follows. The signal power entering the DUT is varied by adjusting the amount of optical attenuation, and the Bit Error Ratio (BER) is measured as a function of received power. As the power received by the DUT decreases, the BER will increase until the received signal is no longer able to sustain a link with the SmartBits. Through the collection of data points and application of proper statistics, it is possible to make a good prediction of the actual BER for any given power.

The frames used for the BER testing have been defined by the IEEE 802.3ah standard. Although these frames provide predefined stressful test patterns within the frames, they also provide a number of user-definable fields that allow for the transmission of these frames onto an actual Ethernet network. There are two types of frames that are defined: a random test frame and a jitter test frame.

The random test frame contains the following repeating pattern within its payload: 0xBED723476B8FB3145EFB3559. This pattern provides broad

spectral content within the frame. The jitter test frame contains the following repeating pattern, and is used to expose the receiver to phase jumps, and alternates between high-density transitions and low-density transitions.

- 0x7E7E repeated 96 times (10000111000111100011, low density)
- 0xF4EBF4EBF4EBF4AB (00101100011101001110..., phase jump)
- 0xB5 repeated 20 times (1010101010, high density)

The frame contains 18 bytes of header information, a 32-byte user defined field, and a 456-byte test pattern, giving the total frame a size of 506 bytes. These frames are sent continuously by the test station with a minimum interpacket gap. The DUT receives these frames and either forwards them out of another port, or loops them back to the transmitter. When the frames are received with errors, such as when the received power becomes too low, they are usually marked as having some sort of error and are not forwarded from any other ports. The second SmartBits port is used to count the frames that are forwarded by the DUT. All frames that do not make it through the DUT are assumed to be lost due to bit errors in the frame. The link between the DUT and the second SmartBits port is assumed to be error free, and if frames are lost across this link then they should be counted as valid frames in the BER calculation.

The software created to perform the automated receiver sensitivity testing was written in LabVIEW. Modules were written to control both the EXFO attenuator and the SmartBits device. The figures on the next several pages are screenshots taken from the program. Figure 14 shows the automated testing

screen where all of the information is captured. The primary table in the figure has three columns. The first column contains the number of frames transmitted, the second column contains the number of frames received, and the third column contains the power at which the test was run, as measured at the input to the receiver.

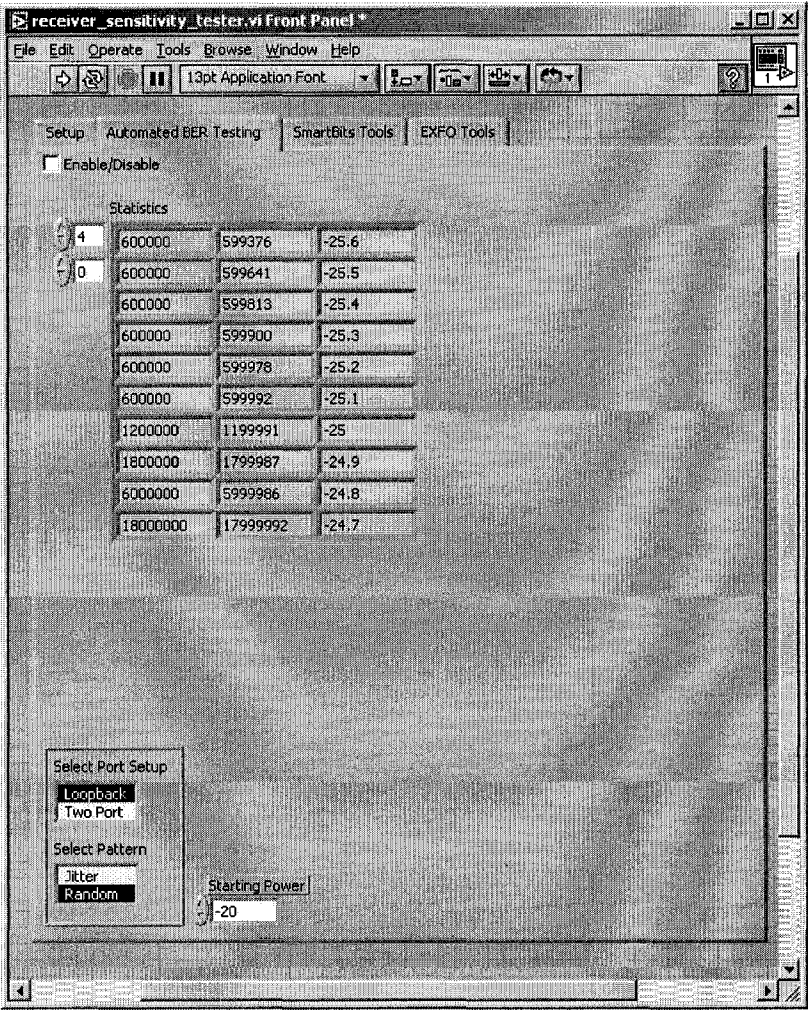


Figure 14 - Receiver tester Automated BER screen

Figure 15 and Figure 16 show the parts of the BER testing tool that can be run manually instead of automatically. In Figure 15, the user has the ability to

manually connect to the SmartBits and set up the SmartBits to transmit various numbers of frames at various rates, and to view the results directly on screen. In Figure 16, the user has the ability to manually control the optical attenuator. The step size, absolute attenuation, and reference wavelength, can all be controlled manually. The user will typically use these two sets of controls to do some quick tests of the setup and make sure everything is functioning properly.

For example, when running the test, it is easiest to start with very low powers and then slowly provide more and more power to the DUT. This has the effect of moving from a higher BER to a lower BER. However, when the power at the receiver becomes too low, then the DUT will no longer be able to maintain a link. Presently, it is quicker for the user to manually find this minimum power point instead of automatically finding it. This power is then manually input into the automated program before it is run.

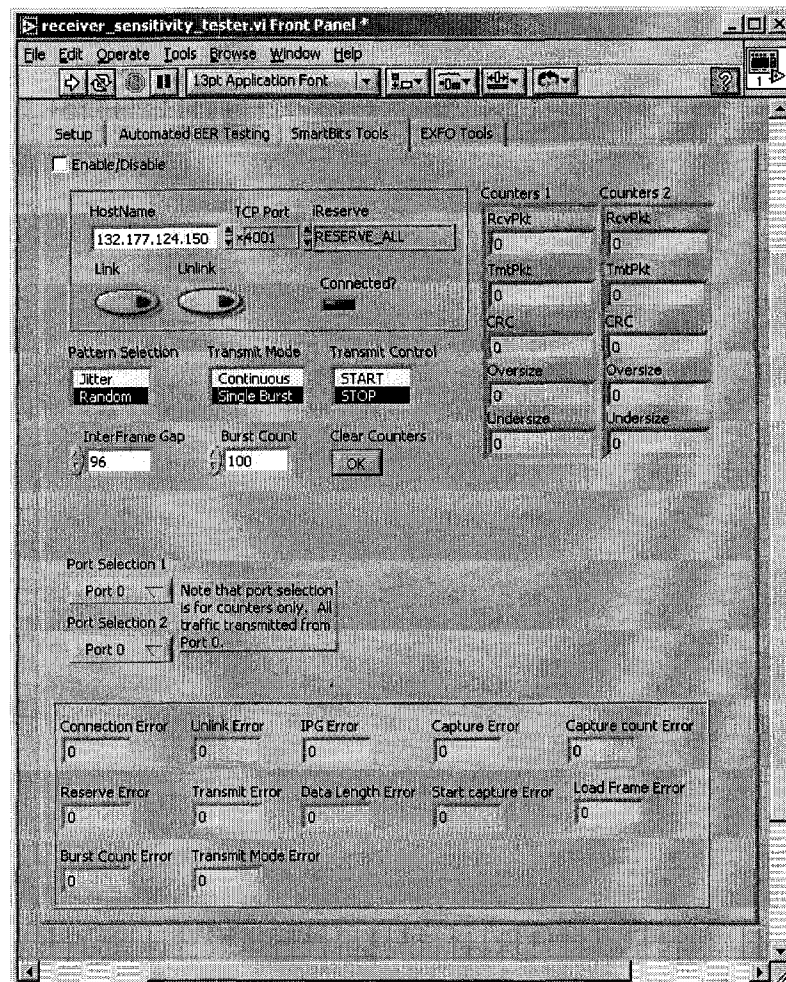


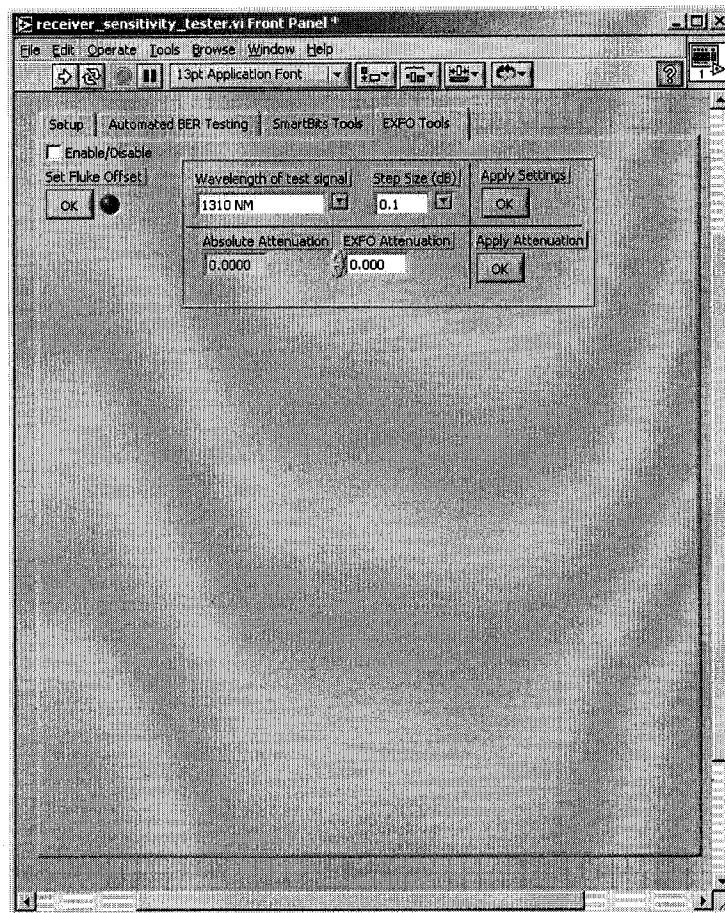
Figure 15 - Receiver tester manual test screen

### **BER testing algorithm**

As is shown in Figure 14, the received power is slowly increasing by 0.1 dB with each test run. In addition, the number of transmitted frames is also increasing. The reason behind this is that in order to accurately estimate the BER, it is important and necessary to see errors. When no errors are seen, it is difficult to accurately estimate what the BER actually is, and it is only possible to estimate that the BER is better than some other value. However, once errors are seen, it becomes easier to not only measure the BER at a particular received



power level, but to also assign levels of accuracy, or error bars, to each of those data points.



**Figure 16 - Receiver tester manual attenuation screen**

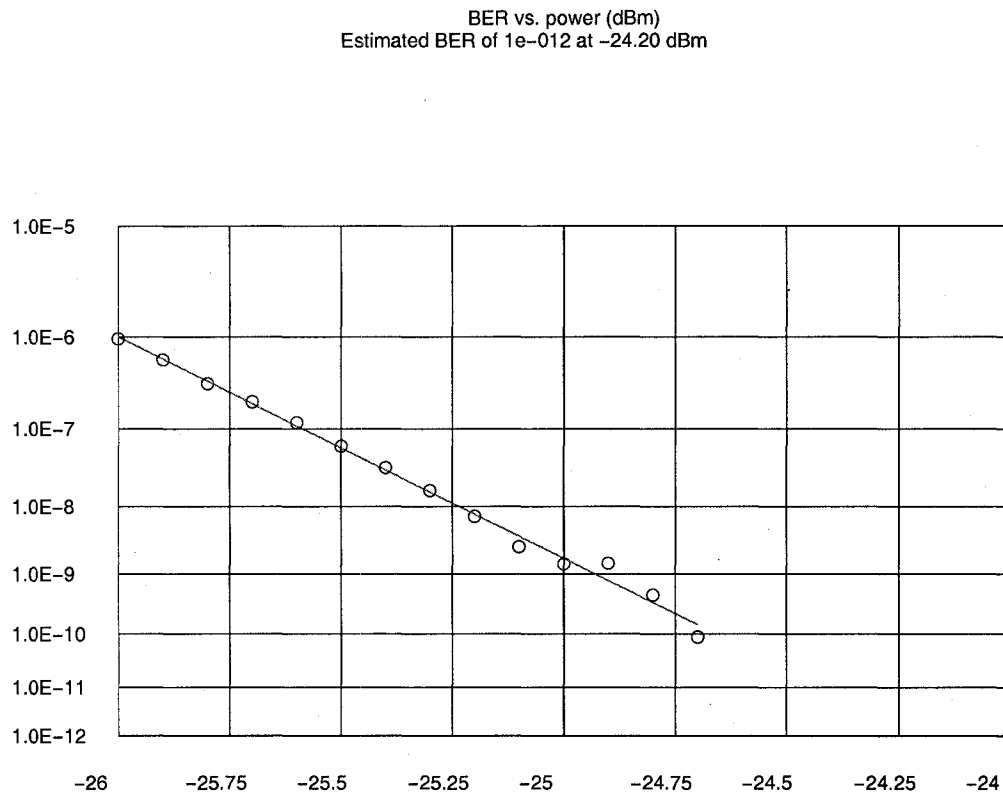
The generally accepted UNH-IOL algorithm for determining whether or not a DUT supports a given BER over a specific channel states that to test to a BER of  $10^{-X}$ , approximately  $3 \cdot 10^X$  bits must be sent [10]. When zero errors are observed after the test, there is a 95% chance that the DUT supports a BER of at least  $10^{-X}$ . If more than 7 errors are observed, then there is a 95% chance that the DUT does not support a BER of  $10^{-X}$ . When the number of errors observed is between 1 and 7, it is difficult to make a claim for meeting or not meeting the

BER. It should be noted that performing a test using these assumptions is very good when trying to determine whether or not a DUT meets a specific BER under a single set of static conditions. It is not a very useful method for determining the actual BER of the DUT using a given set of conditions, or for determining how the BER of the DUT will vary as the conditions vary.

As mentioned above, in order to make an actual measurement of the BER, it is necessary to see errors. When errors are present, a non-zero BER can be calculated by simply taking the ratio of the number of bits the DUT receives in error to the number of bits transmitted by the testing station. Therefore, the algorithm used by this project transmits frames at a constant power level to the DUT until a certain number of errors are observed. Once enough errors are seen, the power level is changed and the test is run again. As the power level increases, it becomes necessary to send additional frames so that errors can still be seen.

Since the specified minimum BER for EPON (and most Ethernet optical devices) is  $1 \cdot 10^{-12}$ , the algorithm sends a fixed number of frames to eventually transmit  $3 \cdot 10^{12}$  bits. Using the jitter test frames, this breaks down to approximately 600 million frames, which should take approximately 1 hour to send with a minimum IFG. Fortunately, for really low received powers, it is not necessary to send anywhere near this many frames. Once errors are observed it is possible to begin to make an estimate of the BER. Initially, 600,000 frames are transmitted. If more than X errors (X is user defined, but 7 is the nominal value) are observed, then the optical attenuation is adjusted and the process

repeats itself. If fewer than X errors, but more than 0 errors are observed, an additional 600,000 frames will be sent, and this will continue until at least X errors are observed. If 0 errors are observed, then additional frames are sent until the total number of frames climbs to 6 million. After sending 6 million frames, if 1 to (X-1) errors have been seen, then the program will send an additional 6 million frames and so on. If 0 errors are observed, then additional frames are sent until the total number of frames climbs to 60 million. Eventually, a total of 600 million frames will be sent, and the program will stop.



**Figure 17 - Automated BER plot example**

Once all of the frames have been transmitted, a table similar to the one shown in Figure 14 is created. This data can then be plotted using a specific set of axes, and the data points should fall on a straight line [11]. Figure 17 shows the previously mentioned set of data plotted on the BER axes, and extrapolates the BER up to  $10^{-12}$  (the extrapolated value is shown in the title of the graph). In addition, it is possible to assign error bars to each data point to estimate how accurate each measured BER data point is.

### **Individual contributions to existing tools**

As mentioned previously, modifications were made to the existing Test Vector Generator and Trigger programs in order to facilitate EPON testing. The modifications included allowing the code to properly utilize two logic analyzer channels instead of the one channel previously used. After making the modifications, all of the test vectors and triggers used to perform the testing were created by the author in support of this thesis.

The optical PMD software for performing transmitter measurements required significant code changes in order to support measurements on optical signals instead of electrical signals. This includes adding code for all of the power and timing measurements as well as the ability to scale the optical eye relative to its power levels. The optical PMD software for performing receiver measurements was created specifically for this project.

## **CHAPTER V**

### **CONFORMANCE TESTING OF CLAUSE 65 RS**

This chapter begins with an overview of the Clause 65 specification, and then goes into explicit detail on how to best perform conformance testing on a Clause 65 capable device. A discussion of the tools, methodology, and test suite are provided for the reader's consideration.

#### **Overview of Clause 65**

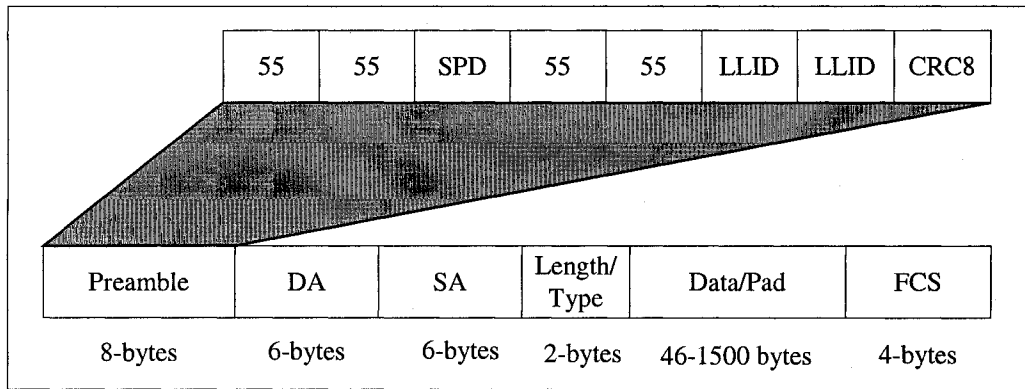
Clause 65 of IEEE Std. 802.3-2005 defines the Reconciliation Sublayer (RS) and Physical Coding Sublayer (PCS) / Physical Media Attachment (PMA) for 1000BASE-X multipoint links. The clause also delves into the forward error correction (FEC) scheme devised for EPON. The three main areas of importance of this clause are: Preamble for EPON devices, laser control for ONUs, and FEC for EPON devices.

#### **Preamble for EPON devices**

Even though an EPON device uses PCS and MAC sublayers that are identical to the ones used by 1000BASE-X devices, extensions have been added to the RS in order to allow for point-to-point emulation. Figure 18 shows the fields of the EPON frame. Although most of the fields are identical to those of a 1000BASE-X frame, there is a significant difference in the contents of the first part of the frame, the preamble. In a traditional device, this field would contain seven bytes of 0x55 and a single eighth byte of 0xD5. The preamble was

traditionally used for clock synchronization and to inform a device that a frame was on the way, and has been kept on with additions to IEEE 802.3 primarily for backwards compatibility issues.

For EPON devices, the preamble contains a significant amount of information. As shown in Figure 18, four bytes of the preamble have been left unaltered and will still be transmitted as 0x55. The third byte of preamble contains a Start of Packet Delimiter (SPD) that is transmitted as 0xD5. The sixth and seventh bytes are replaced with a Logical Link ID (LLID) that contains the LLID and mode bit associated with either an ONU or the OLT. A unique LLID is assigned by the OLT to each ONU once the registration process is complete. The RS of the ONU will filter frames based on the value of the LLID field in the preamble. This was a necessary feature to add to the EFM specifications in order to allow for the architecture of the PON. For example, in a typical shared Ethernet network, a device that transmits a frame will not receive the exact frame that it transmitted. A repeater or switch will forward a frame out all ports other than the port on which it was received. The nature of the PON makes this impossible. The OLT can be placed in a mode that will force it to forward all frames it receives from one ONU to all other ONUs. Doing this will mean that the initiating ONU will receive its own frame. Whereas this could potentially cause problems in a traditional Ethernet network, the filtering that takes place in the RS using the modified preamble will prevent the originating MAC from receiving its own frames.

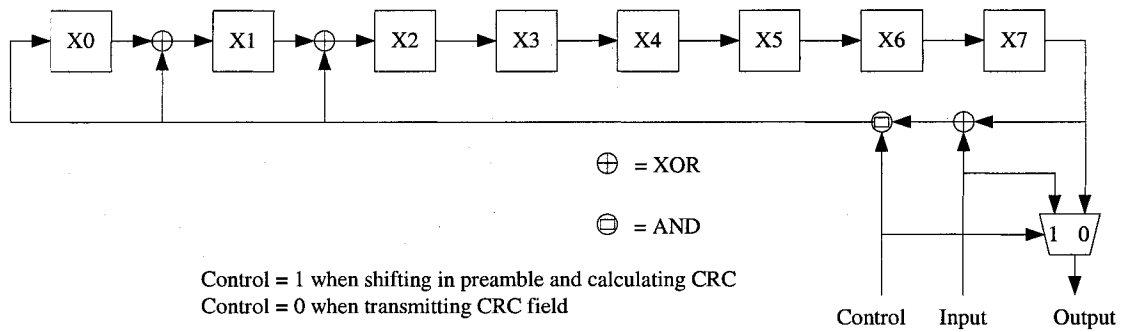


**Figure 18 - EPON Frame Structure**

Table 2 shows the values that are to be transmitted in the preamble fields. The mode bit is assigned to be a 1 for an OLT MAC and a 0 for an ONU MAC. The 15-bit logical\_link\_id value is set to a broadcast value of 0x7FFF for the ONU before it has registered with the OLT. The ONU is assigned a value, by the OLT, other than 0x7FFF after the registration process has completed. The OLT may use any value for this variable. Finally, the CRC8 field contains an 8-bit value that is computed as a function of the preamble from the first bit of the SPD through the last bit of the LLID. The receiver determines the integrity of the received preamble by calculating the CRC8 value on the frame and comparing the calculated value to the value placed in the frame by the remote transmitter.

**Table 2 - Preamble Values**

Preamble field	Value
SPD	0xD5
LLID[15:8]	<mode, logical_link_id[14:8]>
LLID[7:0]	<logical_link_id[7:0]>
CRC8	Calculated 8-bit CRC



**Figure 19 - Serial Implementation of Preamble CRC Generator**

Figure 19 shows a serial implementation of the CRC8 generator using an 8-bit shift register, a few other logic gates, and a multiplexer. The register is initialized to the value 0x00 every time a new frame is being transmitted. The calculation takes place on the 40 bits from the SPD to the end of the LLID fields, and the result is attached to the end of the preamble before the frame is passed on to the PCS.

### **Preamble Calculation Example**

The preamble is defined in IEEE Std. 802.3-2005 Clause 35.2.3.2.1 for 1000BASE-X devices. It is a seven-octet pattern, followed by a one-octet Start of Frame Delimiter (SFD), which indicates that the next bit following the SFD is the first bit of data in the frame. When transmitted serially from left to right, the preamble and SFD take on the following pattern (repeating 10101...):

10101010 10101010 10101010 10101010 10101010 10101010 10101010 10101011



If we assume that we are dealing with an OLT that is using the LLID value of 0x7FF, then the following would be the EPON preamble pattern, with the Xs representing the currently unknown CRC8 value:

10101010 10101010 10101011 10101010 10101010 11111111 11111111 XXXXXXXX

The CRC8 calculation takes place beginning with the third byte of preamble and continues through the seventh byte. The preamble bits will enter the shift register, shown in Figure 19, from left to right, exactly as they are shown here:

10101011 10101010 10101010 11111111 11111111

After 40 clock cycles, assuming the calculation is being done in a serial fashion, the following values will be in the shift register:

X0	X1	X2	X3	X4	X5	X6	X7
0	0	1	0	0	0	1	1

These bits will then leave the shift register serially from X7 down to X0 so that the CRC8 value to be transmitted is:

11000100

Putting all of the fields back together, the transmitted preamble will look like this (remember that each byte is transmitted LSB to MSB):

Binary	Hex
MSB      LSB	
01010101	0x55
01010101	0x55
11010101	0xD5
01010101	0x55
01010101	0x55
11111111	0xFF
11111111	0xFF
00100011	0x23

## **Laser Control for ONU**

As has been previously discussed, the EPON architecture makes it imperative that the ONU properly control its laser transmissions by turning the laser on and off at the appropriate times. In contrast to this, the OLT is allowed, and mandated, to always keep its laser on. The basic principle behind laser control is that the ONU will turn on its laser when it has data to transmit, and will turn off the laser when no more data is waiting to be transmitted. This is accomplished by the presence of a FIFO buffer, which has a length great enough such that when the laser is turned on, enough idle codes are transmitted by the ONU prior to data codes so that the OLT will be able to properly synchronize to the incoming bit stream.

Figure 20 depicts the laser control function and buffer for an ONU that is attempting to transmit two consecutive frames. Initially, the buffer is completely filled with idle codes, and the laser is turned off. When the first block of data enters the buffer, the laser begins to turn on and the ONU begins to transmit the idle codes within the buffer. The buffer has a large enough size such that there is plenty of idle transmitted before the data frame to permit the OLT to synchronize. Following the idle, the ONU will then begin to transmit the frame onto the fiber. Between the two frames, the ONU transmits a small amount of idle, as mandated by the standard. Once the second frame has been transmitted, the buffer begins to fill back up with idle, and when the buffer is completely full of idle, the laser will begin to turn off. In short, the laser will turn

on once data enters the buffer and will turn off when no more data is in the buffer.

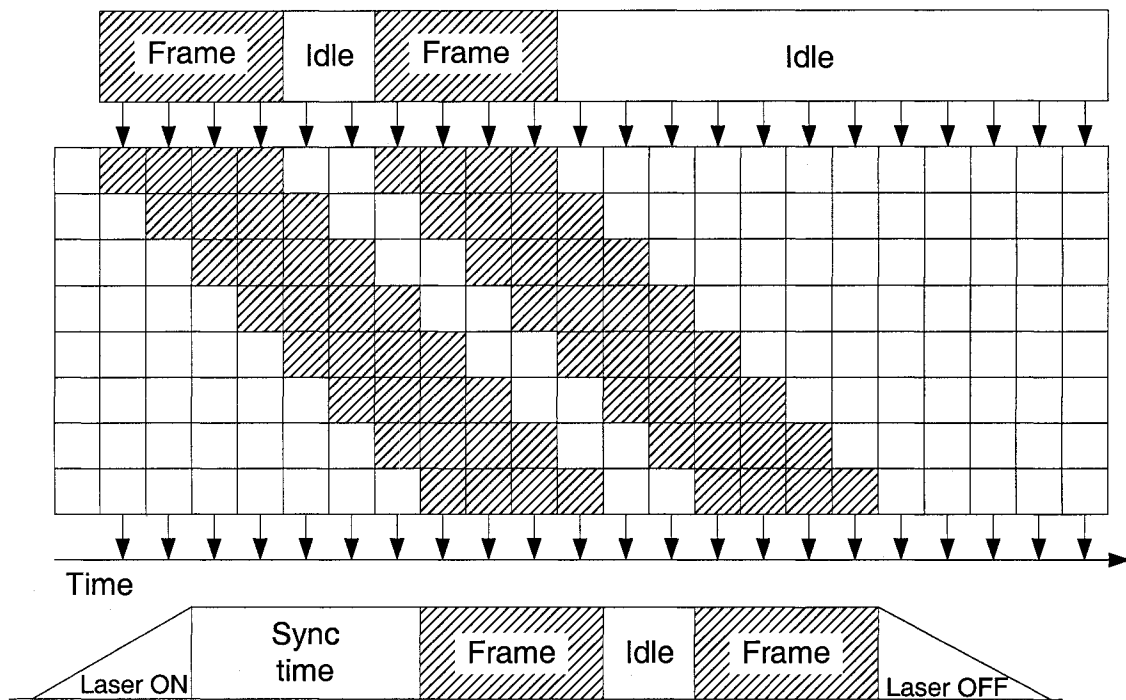


Figure 20 - Laser control for ONU

### **EPON Forward Error Correction**

Forward error correction was inserted into the standard in order to increase link distances and/or split ratios for the EPON. The mechanism chosen was a frame-based Reed-Solomon code (255, 239, 8). This code operates on 8-bit symbols and encodes blocks of 239 symbols into blocks of 255 symbols by adding 16 parity symbols. The particular code and scheme chosen support a Bit Error Ratio (BER) of  $10^{-4}$  at the physical layer, and can correct that ratio to  $10^{-12}$  at the PCS. In addition, since the FEC is frame-based (encoding and applying the parity symbols to just the frame) instead of stream-based (encoding and applying the parity symbols to the frame and idle) it allows for

optional functionality and backwards compatibility. One, both, or neither side of the link can choose to support FEC. On a given EPON, it is conceivable that some ONUs may support FEC and others, perhaps those closer to the OLT, may not support FEC.

### **Clause 65 Testing**

The remainder of this chapter deals with how to go about performing conformance testing of a device that supports Clause 65 functionality. Each of the optional and mandatory items will be covered individually. The reader should pay special attention to the grey “How to test” boxes.

**How to test:** Throughout the rest of this document, these shaded boxes provide detailed information on how to test the various items. In some cases, specific test vectors and setups are also provided.

There are just over twenty different mandatory items listed within Clause 65 and a small number of major options. In addition, a number of requirements that are not explicitly laid out in the PICS are also present. The first major option is whether the DUT is an OLT or ONU, as there are slightly different requirements for each. The second major option deals with whether or not the device supports the defined FEC algorithm. The descriptions that follow will state different procedures when necessary for both the OLT and ONU.

### **Protocol layer test setup**

Figure 21 shows the actual test setup used for performing the EPON protocol layer testing that will be defined in this and the following chapters of this

document. This setup incorporates many of the components mentioned in previous sections of this chapter, and shows all of the connections that are necessary to perform the Clause 64 and Clause 65 testing.

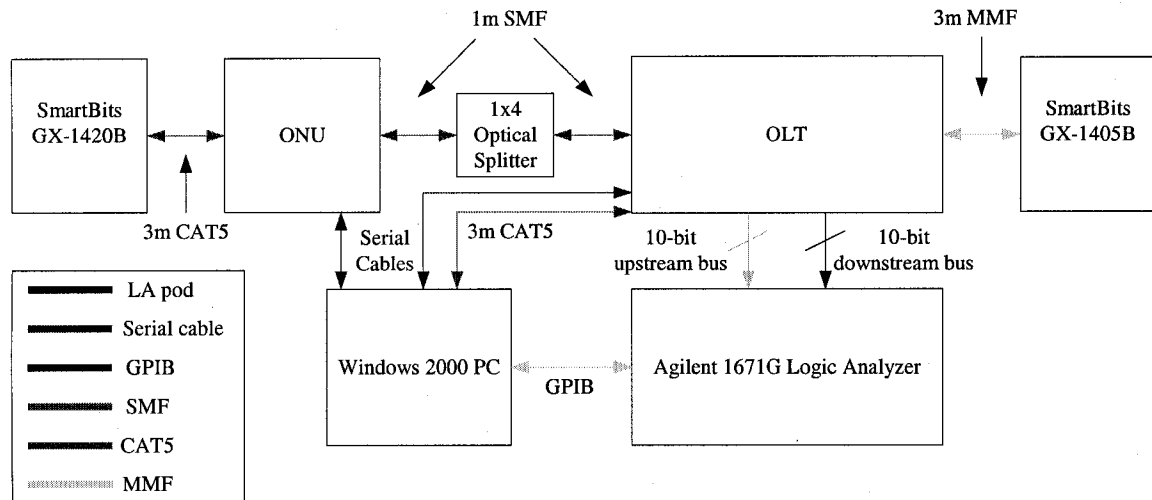


Figure 21 - EPON protocol layer test setup

### Unidirectional Mode

The Unidirectional mode requirement (Clause 65 PICS item OM1) states that the OLT must be able to transmit data frames while not receiving anything (data or idle) from the connected ONUs. At any given time, and due to the MPCP rules, the OLT may or may not actually be receiving data or idle from the connected ONUs. However, the OLT is required to keep its laser turned on continuously, and needs to be able to send GATE messages to the ONUs in order to register them. ONUs, on the other hand, are forbidden from operating in a unidirectional mode, as they may only transmit when granted by the OLT, and will always be receiving idle or data from the OLT.

**How to test:** In order to test this requirement, the tester needs only to connect

the transmitter of the ONU to a suitable monitoring device, and to make sure that no ONUs are currently connected to the EPON. If the OLT is observed to transmit any type of frame, then it is operating in unidirectional mode. If the OLT is not observed to transmit frames, then it is not operating in unidirectional mode.

### **Enable variable**

The enable variable requirement (Clause 65 PICS item FS1) states that the variable takes on a value of TRUE for an ONU and takes on a value of TRUE or FALSE for an OLT. For an OLT, the value will be TRUE when management has assigned a value to the mode and logical\_link\_id for the associated MAC, indicating that the MAC is now ready to receive data frames from the matching ONU. It will take on a value of FALSE when the MAC has not been enabled and does not have a mode or logical\_link\_id assigned to it, thus indicating that the MAC is not able to receive frames from the matching ONU.

Since no management register is defined to store the status of this variable, the tester must resort to transmitting frames to the individual MAC and observing whether or not the frames are received. Although this may not be the most obvious or easiest way to perform the test, the method is still valid.

### **How to test:**

*DUT is an ONU:* In order to test the ONU, the OLT emulator can be used to issue a Discovery GATE message and verify that the unregistered ONU responds with a REGISTER\_REQ message. The presence of the response indicates that the ONU MAC is enabled and capable of receiving frames from the OLT.

*DUT is an OLT:* In order to test the OLT, the ONU emulator can be used to transmit frames containing mode and logical\_link\_id values that have not been assigned by the OLT. The OLT should ignore these frames, and these frames should not bring about any response from the OLT if the OLT has properly set the enable variable. Conversely, the OLT should accept and act on frames coming from an ONU that has been assigned and is properly using the mode and logical\_link\_id values.

### **Mode Variable**

The mode variable requirement (Clause 65 PICS item FS2) states that the variable shall be a 0 for an ONU MAC and can take on the value of either 0 or 1 for an OLT. If the OLT is sending a frame on either the broadcast or multicast channel, then the mode bit takes on the value of 1, and if the OLT is sending a frame on the unicast channel, then the mode bit takes on the value of 0.

The purpose of this bit is to allow filtering to take place that is necessary to preserve the emulation of a point-to-point Ethernet network for an EPON. Traditionally, when a device sends a frame onto an Ethernet network, that frame will enter a switch or repeater and can potentially be forwarded out to all ports other than the port on which the frame is received. If an ONU wishes to send frames to another ONU on the same EPON, it must first send the frame to the OLT, and the OLT will turn around and send the frame downstream. However, on an EPON, since any frame sent in the downstream direction will be received

by all ONUs, the ONU that originated the frame will receive the same frame it transmitted. The mode bit, in combination with the logical\_link\_id, allows the ONU to properly filter the incoming frames and to discard and ignore any frames that would have originated from its local MAC.

**How to test:**

*DUT is an ONU:* Using a passive monitoring station, observe all REGISTER\_REQ, REGISTER\_ACK, and REPORT messages transmitted by the ONU. All of these MPCP frames should contain a mode bit with a value of 0. Using a standard traffic generator, such as a SmartBits, provide unicast, multicast, and broadcast DATA frames to the ONU so that they will be transmitted upstream. All of these frames should contain a mode bit with a value of 0.

*DUT is an OLT:* Using the passive monitoring station, observe all GATE and REGISTER messages transmitted by the DUT. All discovery GATE messages and REGISTER messages should contain a mode bit value of 1. All normal GATE messages should contain a mode bit value of 0. Using a traffic generator, cause the DUT to send unicast, multicast, and broadcast DATA frames. All unicast frames should contain a mode bit value of 0, and all broadcast or multicast frames should contain a mode bit value of 1.

**Logical Link ID Variable**

The logical\_link\_id variable requirement (Clause 65 PICS item FS3) states that the 15-bit variable is to be set to any value for an OLT, 0x7FFF for an



unregistered ONU MAC, and any value other than 0x7FFF for a registered ONU MAC. The value is set exclusively by the OLT, and is effectively an address for the individual ONUs. When attempting to transmit frames to multiple ONUs, the ONU uses the broadcast value (0x7FFF), and will use the assigned unicast value (ex: 0x0001, 0x0002) when attempting to communicate with a single ONU. All frames transmitted by the ONU, once it is registered, will contain the assigned value.

**How to test:**

*DUT is an ONU:* Using the passive monitoring station, observe all MPCP and DATA frames transmitted by the DUT. The REGISTER\_REQ messages should be transmitted with the logical\_link\_id value of 0x7FFF, when the ONU is not registered. All other MPCP messages, including REGISTER\_REQ messages once the ONU is registered, should be transmitted with the appropriate assigned logical\_link\_id value.

*DUT is an OLT:* Using the passive monitoring station, observe all MPCP and DATA frames transmitted by the DUT. Discovery GATE messages and REGISTER messages should use the broadcast logical\_link\_id of 0x7FFF. Broadcast DATA frames should also use the broadcast logical\_link\_id. Normal GATE messages and all other DATA frames should use logical\_link\_id values other than 0x7FFF. The actual values are implementation specific and up to the vendor.

## **CRC-8 Generation**

The CRC-8 generation requirement (Clause 65 PICS item PM1), states that the CRC8 field is to be calculated using the equation:  $G(x) = x^8 + x^2 + x + 1$ , which is also shown in Figure 19. Although the figure shows a serial implementation of the checksum generator, an actual hardware implementation is likely to use a parallel structure to compute the CRC8 field value. The standard often uses serial implementations as examples in order to easily and most clearly define the desired behavior of the function.

### **How to test:**

In order to test that the proper CRC-8 value has been calculated, the passive monitoring station is used to capture upstream and downstream traffic. Each frame, sent by either the ONU or OLT, is required to have a properly calculated CRC-8 value. The location of the field can be seen in Figure 18. After capturing the frames, the received CRC-8 value can be compared against the calculated CRC-8 value for the same frame contents. Both values must match in order for the device to pass the test. A simple LabVIEW utility, `epon_preamble.vi`, has been created to manually check the contents of the preamble. The tester simply needs to enter the preamble contents of one or more frames, and verify that the "CRC8" field contents match the calculated "Correct CRC8 value". The utility is shown in Figure 22.

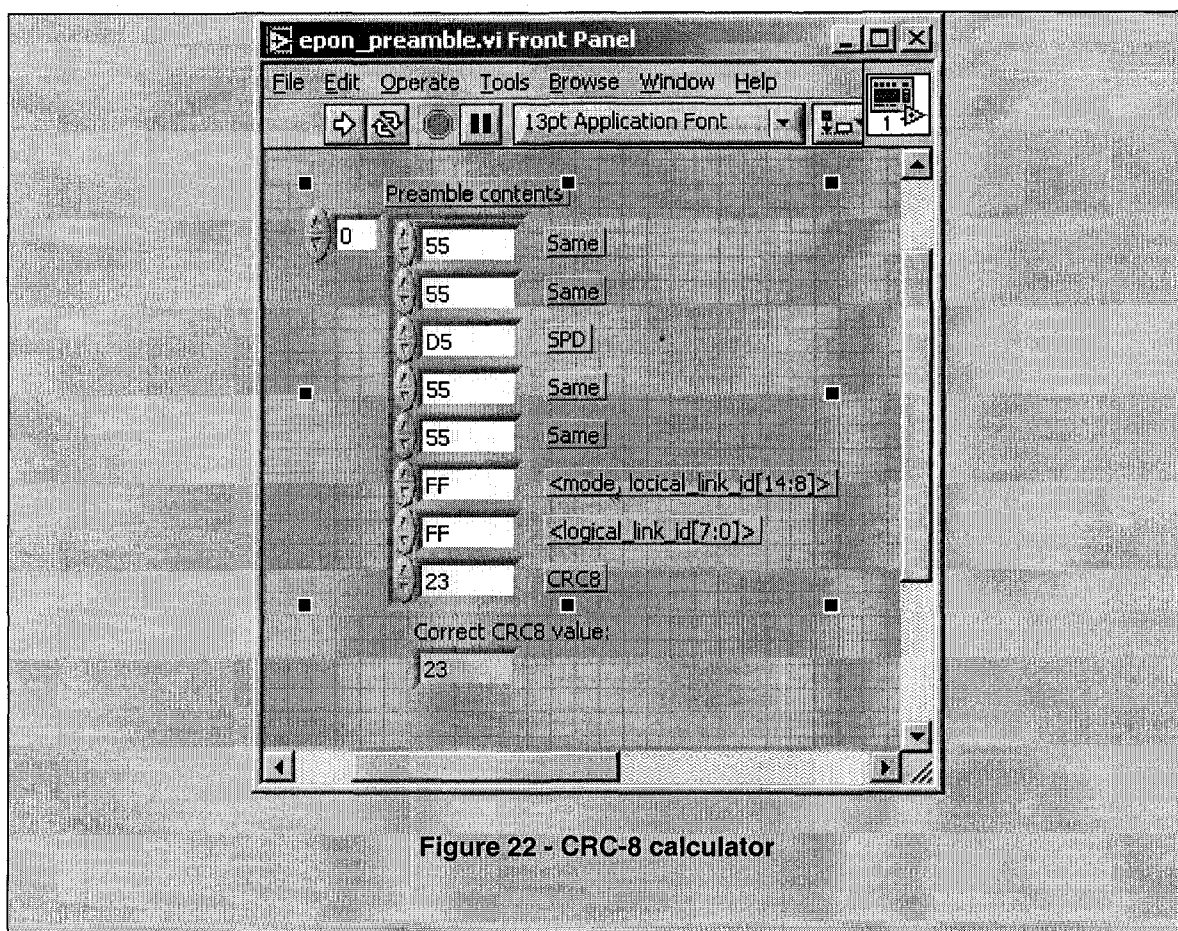


Figure 22 - CRC-8 calculator

### **CRC-8 Initial Value**

The CRC-8 initial value requirement (Clause 65 PICS item PM2) states that the shift register used to compute the CRC-8 value is initialized to the value 0x00 prior to every instance of preamble that needs the calculation performed. It is important to initialize the shift register prior to every calculation, so that there will always be the same initial shift register contents on both the transmitter and receiver. If every frame that the OLT were to send would be received in order by the ONU (and vice versa), it would not be necessary to do this. However, since the possibility exists that frames can be corrupted and lost throughout the

network, it is not possible to transmit and receive the frames without using a common initialization sequence.

**How to test:**

The initial value of the CRC-8 can be verified by using the same LabVIEW utility as previously mentioned. The utility also has a special mode so that the tester can manually enter in any initialization value (0x00 – 0xFF) for the shift register. The default settings use an initialization value of 0x00 to perform the calculation. For computed CRC-8 values using an initial value other than 0x00, the utility will detect a CRC-8 error.

**Start of LLID delimiter (SLD) Parsing**

The SLD Parsing requirement (Clause 65 PICS item PM3) states that there are only two positions in which the SLD may exist in a received frame. The SLD may exist in either the second or third octet of the preamble, depending on where it was placed by the transmitter of the link partner. Due to Clause 36 PCS requirements, it is possible for the transmitting device to delete one of the preamble octets in order to properly align the start of the frame to the correct position. Therefore, the receiver must be able to look for and receive the SLD in either position. If the SLD is found in either of these two locations, the DUT should continue to receive and parse the frame. If the SLD is not found in either of these locations, the DUT should discard the frame.

**How to test:**

A number of test vectors can be created to properly test the parsing of the



received SLD, as shown below. Test vectors 'a' and 'b' are valid test vectors that show the two possible positions of the SLD, which is represented by the value of 0xD5. Both of these test vectors should be properly parsed by the DUT. Test vectors 'c' and 'd' show otherwise valid test vectors that do not contain an SLD. Each of these test vectors should force the DUT to discard the frame. Test vector 'e' contains a valid SLD, but the SLD is in an invalid position within the preamble. This frame should also be discarded by the DUT.

a	b	c	d	e
55	55	55	55	55
55	D5	55	55	55
D5	55	55	55	55
55	55	55	55	D5
55	LLID	55	LLID	55
LLID	LLID	LLID	LLID	55
LLID	CRC8	LLID	CRC8	LLID
CRC8		CRC8		LLID
				CRC8

### **SLD Replacement**

The SLD Replacement requirement (Clause 65 PICS item PM4) states that if the SLD is found in a valid location, the DUT should replace the SLD with a normal preamble octet, and that the preamble octets preceding the SLD and the two octets following the SLD should be passed up without modification. It is important to replace the SLD with a normal preamble octet, because when the MAC receives the frame, it views the first octet following the value 0xD5 to be the first byte of the destination address. If the SLD is not replaced, then the

receiving MAC will improperly assume that the frame is beginning, and the frame will necessarily be received with an error.

#### **How to test:**

As with the previous test, there are a number of test vectors that can be sent to the DUT. Test vectors 'a' and 'b' are valid preamble patterns, and should be received by the DUT. If the DUT receives the frame without error, then it can be inferred that the SLD octets were replaced with normal preamble octets. Test vector 'c' can be used to verify that the octets preceding and following the SLD are passed up unmodified to the MAC. Since the receiving MAC ignores preamble octets, any octets that do not have the value of 0xD5 should be accepted at the receiver. It should be noted that it might be difficult to verify that the DUT actually passes up the octets without modification, since an observation point may not be available. However, the frame should be accepted by the DUT.

a	b	c
55	55	55
55	D5	01
D5	55	D5
55	55	02
55	LLID	03
LLID	LLID	LLID
LLID	CRC8	LLID
CRC8		CRC8

#### **LLID Matching**

The LLID Matching requirement (Clause 65 PICS item PM5) states that the device must look at the received mode and logical\_link\_id bits to determine whether or not the frame should be received. Different behavior is necessary

depending on whether the device is an ONU or an OLT. If the values are valid, then the frame should be received. If the values are invalid, then the frame should be discarded. If the device is an OLT, then the received mode bit is ignored. If the received logical\_link\_id value is the broadcast value or any value that is related to an enabled MAC, then the frame is accepted. If the received logical\_link\_id value does not have an enabled MAC, then the frame is discarded. If the device is an ONU, the frame is accepted if the mode bit is 0 and the logical\_link\_id value matches the logical\_link\_id of the ONU. The ONU will also accept the frame if the mode bit is 1 and the logical\_link\_id does not match the logical\_link\_id of the ONU, or if the logical\_link\_id has the broadcast value of 0x7FF.

**How to test:**

There are a number of test vectors, depending on whether the DUT is an OLT or an ONU. If the DUT is an OLT, then Case 1 test vectors 'a', 'b', 'c', and 'd' should be accepted. Test vectors 'e' and 'f' should be discarded. If the DUT is an ONU, then Case 2 test vectors 'a' and 'd' should be accepted, and test vectors 'b' and 'c' should be discarded.

**Case 1: The DUT is an OLT:**

- a. Valid frame with mode bit of 0 and logical\_link\_id of 0x7FF.
- b. Valid frame with mode bit of 1 and logical\_link\_id of 0x7FF.
- c. Valid frame with mode bit of 0 and logical\_link\_id of enabled MAC.
- d. Valid frame with mode bit of 1 and logical\_link\_id of enabled MAC.
- e. Valid frame with mode bit of 0 and logical\_link\_id of disabled MAC.



- f. Valid frame with mode bit of 1 and logical\_link\_id of disabled MAC.

Case 2: The DUT is an ONU:

- a. Valid frame with mode bit of 0 and logical\_link\_id of the DUT.
- b. Valid frame with mode bit of 0 and logical\_link\_id not that of the DUT.
- c. Valid frame with mode bit of 1 and logical\_link\_id of the DUT.
- d. Valid frame with mode bit of 1 and logical\_link\_id not that of the DUT.

### **LLID Replacement**

The LLID replacement requirement (Clause 65 PICS item PM6) states that if a valid match occurs on the received LLID, then the field should be replaced with normal preamble octets. As mentioned above, one of the main reasons for replacing these octets with normal preamble octets is to ensure that a 0xD5 value will not exist within the preamble, as this would improperly cause the MAC to assume the frame was starting.

#### **How to test:**

Each of the test vectors used in the previous test can also be used in this test. In addition, a test vector can be created that uses a logical\_link\_id of 0x0D5. If this frame is received, then it is possible to assume that the logical\_link\_id was replaced with normal preamble octets. It should be noted that it might be difficult to verify that the DUT actually passes up the octets without modification, since an observation point may not be available.



### **CRC-8 Checking**

The CRC-8 checking requirement (Clause 65 PICS item PM7) states that the DUT must calculate the CRC of the received preamble and compare it to the value contained within the CRC-8 field. If the values match, then the frame is accepted, and if the values do not match, the frame is discarded.

#### **How to test:**

The Emulation tool can be used to perform the CRC-8 checking requirement, as it can be used to transmit frames with an incorrect CRC-8 value. Frames are generated and sent to the DUT that contain both valid and invalid CRC-8 values. Frames that contain valid CRC-8 values should be accepted, and those frames that contain invalid CRC-8 values should be discarded.

### **CRC-8 Replacement**

The CRC-8 replacement requirement (Clause 65 PICS item PM8) states that if the preamble contains a valid CRC value, then the octet is replaced with an SFD (0xD5). The SFD is required by the MAC to determine where the preamble ends and where the frame begins. The MAC simply discards the preamble contents until it comes across the SFD, and considers the next octet to be the first octet of the destination address.

#### **How to test:**

Acceptance of validly formed frames is an indication that the CRC-8 is replaced with an SFD. If such a replacement did not occur, then the receiving MAC would either never find the SFD, or would find the SFD in the wrong location. In either

of these instances, the frame would be discarded.

### **Buffer Depth**

The transmitter of the ONU is required to turn itself off between transmissions. When starting a transmission, the ONU is required to send a specified number of idle code groups so that the OLT can properly synchronize itself to the ONU's transmissions. This amount of idle is determined by the OLT and communicated to the ONU during the discovery process. The ONU must transmit idle for  $T_{receiver\_settling} + T_{cdr} + T_{sync}$ . The first two parts of this equation are fixed constants, and the  $T_{sync}$  is determined by the OLT. Since the OLT never turns its laser off, there is no requirement to test here.

#### **How to test:**

Using the OLT emulator, go through the discovery process with the DUT, communicating a SyncTime value of 0x0000, and observe the transmissions from the DUT, measuring the amount of idle transmitted once the laser turns on until the first byte of preamble. The ONU should transmit a minimum number octets of idle before the frame, and this value can be considered IDLEmin. After this test case, vary the value of SyncTime and continue to observe transmissions from the ONU. All transmissions should contain IDLEmin + SyncTime bytes of idle prior to the frame.

## **Laser Control**

The OLT should never turn its laser off, and should always be transmitting validly formed signals to the ONUs. The ONUs, however, must turn their lasers on and off between transmissions. This test verifies that the ONU properly enables and disables its laser during the transmission window. As shown in Figure 20, the ONU maintains a buffer to gate its transmissions. When the buffer of the ONU is filled with idle, it should turn the laser off. As data enters the buffer, the laser turns on and the ONU transmits the proper amount of idle before beginning to transmit data. Upon completion of the transmission window, the ONU will stop transmitting frames and turn off its laser. In addition, if the gap between consecutive frames within a single transmission window is large enough that the buffer fills with idle, the ONU should also turn its laser off and then repeat the process.

### **How to test:**

In order to properly perform this test, it is first necessary to determine the size of the buffer ( $B_{size}$ ). When the ONU transmits consecutive frames with less than  $B_{size}$  of idle between them, the ONU should not turn its laser off between the frames. When the ONU transmits consecutive frames with more than  $B_{size}$  of idle between them, the DUT should turn its laser off between the frames. At the time of publication, all tested implementations have been observed to not transmit consecutive frames with more than a minimum interpacket gap between them. The reason for this is that the ONU waits until it has a number of frames queued up and then transmits only those frames. Additional frames are reserved until a

future transmission window. Therefore, there is no known mechanism to properly observe and verify that the ONU will turn its laser off between consecutive frames within the same transmission window.

### **FEC Coding Choice**

Forward Error Correction is an option available for both the ONU and OLT. As previously mentioned, it is not necessary for both devices to implement the FEC. However, should a device choose to use an FEC algorithm, there is only one algorithm defined and allowed in the standard. During the course of standardization, there were a number of proposals on different FEC mechanisms that could be implemented, including stream-based and frame-based. This test verifies that if FEC is implemented, the defined frame-based mechanism is used. Figure 23 shows the format used for frames that contain FEC. At the beginning of the frame is a special set of words, S\_FEC, used to signify to the receiver that an FEC encoded frame is about to follow (the S\_FEC will pass through a non-FEC device and the frame will be received as any other normal frame). Following the frame is a special set of words, T\_FEC, signifying that the parity bytes are about to follow. The T\_FEC is repeated following the parity bytes to signify the completion of the frame. The number of parity bytes varies as a function of the frame size, with 16 bytes added for every 239 bytes within the frame.

S_FEC 4 bytes	Preamble 8 bytes	Frame 64 - 1518	T_FEC 6 bytes	Parity bytes	T_FEC 6 bytes
------------------	---------------------	--------------------	------------------	-----------------	------------------

**Figure 23 - Frame format with FEC**



**How to test:**

If the DUT supports FEC, then enable the FEC and observe transmissions from the device. Frames should be verified through the FEC decoder program. Each frame should contain properly formed S\_FEC, T\_FEC, and parity bytes. At the time of publication, no FEC enabled devices have been tested at the UNH-IOL.

**Uncorrectable Block Replacement**

Occasionally, when implementing any type of FEC scheme, there may be some errors that cannot be corrected. When this condition occurs on an EPON, the device must replace that uncorrected word with a special symbol, /V/. This symbol is used within the PCS for error propagation. The device replaces the word with this symbol in order to clearly show that an error has occurred and to ensure that the frame is not received.

**How to test:**

In order to perform this test, the OLT emulator can be used to generate an FEC coded frame with deliberate errors. In order to do this, the parity bytes are calculated for a valid frame, and the entire FEC frame is created. One or more data words are then inverted, thus creating the maximum number of errors possible within a given word. A frame containing these errors cannot be corrected, and therefore the words should be replaced with /V/ and the frame should be discarded. It may be difficult to actually verify that the word has been replaced with a /V/, as a suitable observation point may not be present within the

DUT.

### **FEC state diagrams**

There are a number of state diagrams that define how the device should transmit, synchronize on the received signal, and receive FEC encoded frames. There are a number of tests that can be performed to properly validate the FEC operation of the DUT.

#### **How to test:**

Detailed tests and procedures have not yet been defined for these state diagrams. This is an area that needs more work and improvement. The main reason for this is that no FEC enabled devices have been tested yet, and therefore the testing has not been requested by any vendors. It is expected that a future update to the test suites will include these tests.

### **Loop Timing**

In order to simplify timing considerations for EPON, the ONU must recover the received clock from the OLT and then use this clock for its own transmissions. When the OLT and ONU use the same timebase, there is a much lower chance of any drifting, and it becomes easier for the OLT to receive signals from many different ONUs.

#### **How to test:**

In order to perform this test, it is first necessary to accurately measure the signaling rate of the OLT, and then measure the signaling rate of the ONU. It is

extremely helpful if the ONU provides a mode that allows the laser to be turned on continuously, as this greatly simplifies the measurement process. Also, if such a mode is supported by the ONU, it is possible to measure the signaling rate of the ONU both before and after it is added to the EPON. If the ONU is truly using the clock of the OLT, then a noticeable change in signaling rate should be observed in these two instances. All that is required is to capture the analog waveforms generated by the ONU and extract the signaling rate from them.

### **Delay Variation**

Since EPON operation relies heavily on distribution of timestamps between the ONU and OLT, it is important to maintain as small a delay variation as possible through the multiple sublayers. The absolute delay through the device is not as important as the variation in delay through the device. The amount of allowable variation has been defined to be no more than 16 bit times.

#### **How to test:**

At the time of publication, this test and its procedures are still under development and will be added to a future version of the test suite.

## **CHAPTER VI**

### **CONFORMANCE TESTING OF CLAUSE 64 MPCP**

This chapter begins with an overview of the Clause 64 specification, and then goes into explicit detail on how to best perform conformance testing on a Clause 64 capable device. A discussion of the tools, methodology, and test suite are provided for the reader's consideration.

#### **Overview of Clause 64**

Clause 64 of IEEE Std. 802.3-2005 defines the Multipoint MAC Control Protocol (MPCP) for EPON. The MPCP utilizes certain MAC Control messages to setup, maintain, and tear down the links between the OLT and the various ONUs. This clause contains the definitions of all of the MPCP messages, along with a number of state diagrams that define the behavior of the protocol and how the messages are to be used for both transmission and reception. The following sections describe the MPCP messages along with a brief overview of the EPON discovery process and its operational modes.

#### **GATE messages**

GATE messages are transmitted by the OLT and received by the ONU. There are two types of GATE messages, discovery GATE messages, and normal GATE messages. Both message types are used to inform the ONU when it may transmit in the upstream direction and for how long it may transmit.



Discovery GATE messages are broadcast messages that are received by every ONU that is physically connected to the network, whether or not the ONU has been registered on the PON. These messages inform the ONUs of three items: the current time, the time the next discovery window opens, and the length of the discovery window.

Normal GATE messages are unicast messages that are only received and acted on by registered ONUs. Each of these messages may contain multiple grants, where each grant contains a start time and length of time for transmission to occur. An ONU may only transmit during the time indicated by the grants contained within the GATE messages, and must be silent at all other times. The OLT is required to periodically allow the ONU to transmit, although the frequency with which this happens and the length of time that must be granted is left up to the implementer.

### **REGISTER\_REQ messages**

REGISTER\_REQ messages are transmitted by the ONU during the discovery process. Once the discovery window has opened, an unregistered ONU may transmit a REGISTER\_REQ message to request access to the EPON. The REGISTER\_REQ message also contains information regarding the number of pending grants that it can support if it is to become part of the network. The OLT will use this information when sending future GATE messages, as some ONU implementations may be able to support more grants than other implementations.

### **REGISTER messages**

REGISTER messages are transmitted by the OLT during the discovery process, and are transmitted in response to a REGISTER\_REQ from the ONU. The REGISTER message contains a flags field that informs the ONU of a successful or unsuccessful registration attempt, along with the amount of idle that the ONU needs to send prior to each transmission in order for the OLT to acquire receiver synchronization. The message also contains the assigned logical\_link\_id that will be used in all future frame transmissions by the ONU. Finally, this message echoes the number of grants that the ONU had previously sent in the REGISTER\_REQ message.

### **REGISTER ACK messages**

REGISTER\_ACK messages are transmitted by the ONU to acknowledge the REGISTER message and to echo back important information to the OLT. The REGISTER\_ACK message echoes both the sync time and the assigned logical\_link\_id back to the OLT. Two reasons for this handshaking process are to ensure that the OLT is able to keep track of which ONUs it is in the process of registering, and to permit the OLT to verify that the specific ONU is transmitting during the appropriate window.

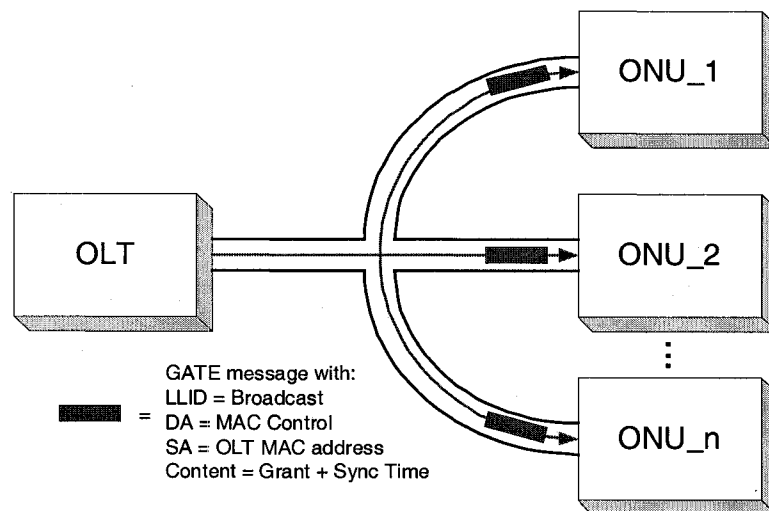
### **REPORT messages**

REPORT messages are transmitted by the ONU, and are used to indicate to the OLT how many bytes the ONU has queued up to transmit. Each REPORT message may contain several queue sets, with each set containing up to seven separate queues, which correspond to priority levels. Although the number of

queues and queue sets are determined by the ONU, it is up to the OLT to actually grant time. The OLT is not required to grant the ONU as much time as it requests.

### **EPON discovery process**

The EPON discovery process is a multi-step process that enables the OLT to register new ONUs by allowing them on the network and allowing them to transmit data frames. Over the next several pages, the discovery process is depicted with a number of different color figures. Although different colors are used in the figures to indicate traffic destinations, all upstream transmissions occur on a single wavelength, and all downstream transmissions occur on a single wavelength. To simplify the viewing of the figures, only one direction is shown at a time. In an actual network, there will be upstream and downstream traffic (each on a separate wavelength) at the same time.

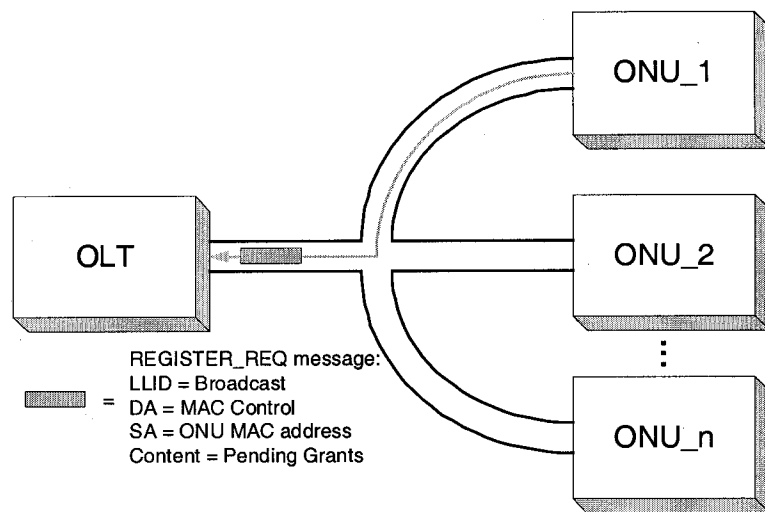


**Figure 24 - Transmission of discovery GATE**

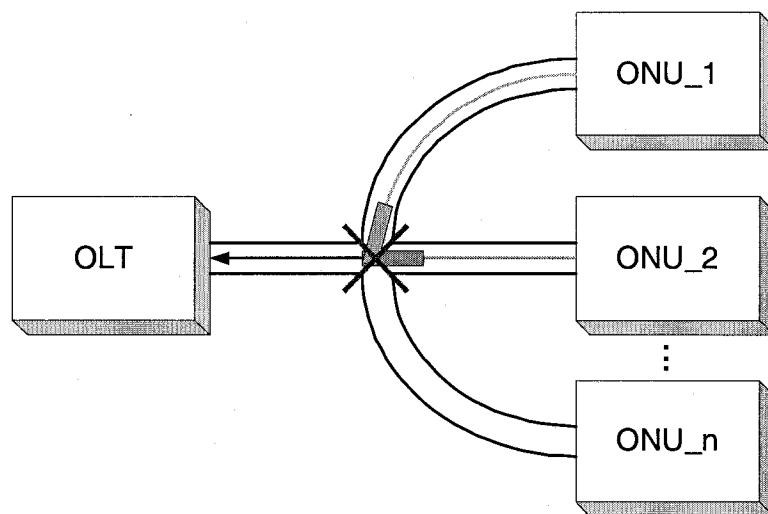
The first step in the discovery process occurs when the OLT sends a discovery GATE message to all of the ONUs, as shown in Figure 24. This is the

only GATE message that is broadcast to each ONU. The OLT is required to open periodic discovery windows so that new ONUs can automatically be added to the network.

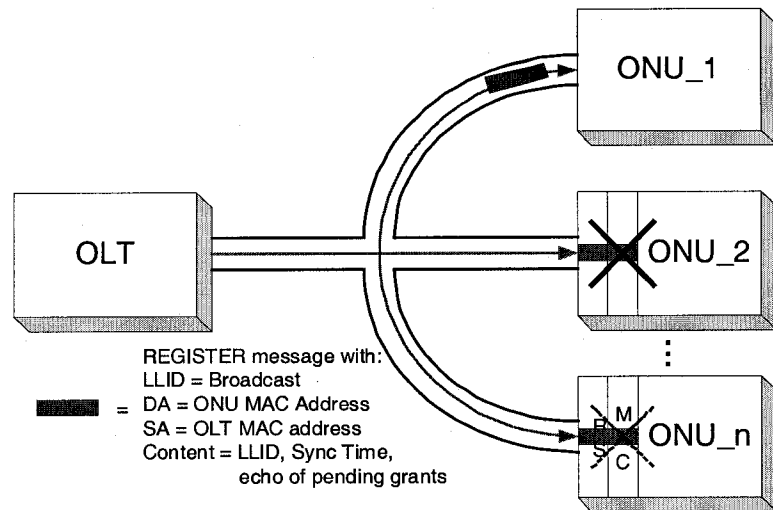
The second step in the discovery process has the ONU sending a REGISTER\_REQ message back to the OLT, as shown in Figure 25. Except under special circumstances, the REGISTER\_REQ message is only sent by an unregistered ONU. This part of the discovery process is also the only time on the network that, in the absence of errors or problems, collisions will occur. For all other upstream transmissions, only a single ONU is allowed to transmit at any given time. Figure 26 shows an example of the case when two unregistered ONUs attempt to transmit their REGISTER\_REQ messages at the same time. To help reduce the likelihood of a collision, each ONU is required to wait a random amount of time from the beginning of the discovery window until the REGISTER\_REQ is generated. If a collision does occur, and the ONU is not registered during the discovery window, then it simply tries again during the next window.



**Figure 25 - Transmission of REGISTER\_REQ**



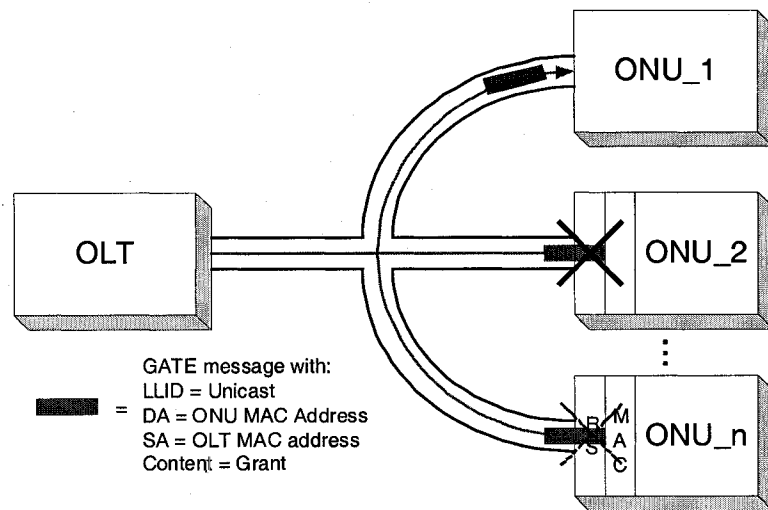
**Figure 26 - ONU collision**



**Figure 27 - Transmission of REGISTER**

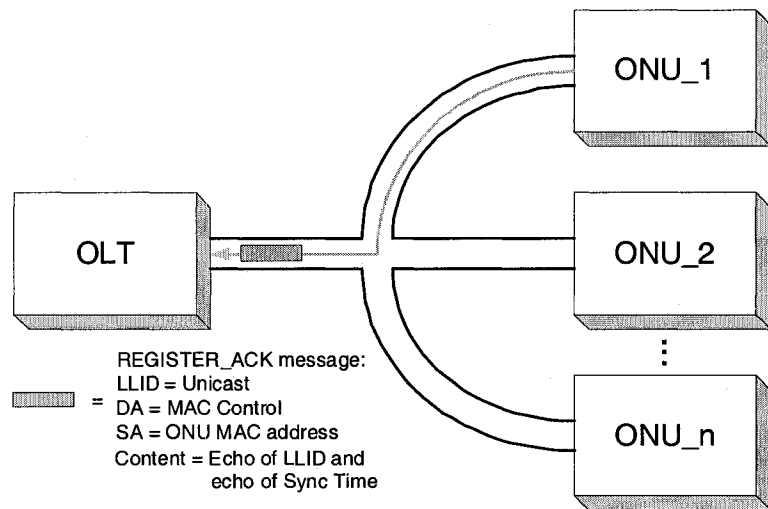
Following the reception of the REGISTER\_REQ message, the OLT will send a REGISTER message to the ONU. This message, although it uses the broadcast LLID, uses a unicast MAC address, and will therefore be filtered out by all MACs other than the one to which it is directed. Figure 27 shows ONU\_1 receiving the REGISTER message, while the other ONUs discard the message once it reaches the MAC sublayer.

Since the ONU is not allowed to transmit frames outside of specifically granted windows, the OLT must send an initial GATE message to the ONU so that the discovery process can complete. Figure 28 shows this process in action. This is the first frame received by the ONU that has both a unicast LLID and a unicast MAC address. The unicast LLID means that the frame will be filtered at the RS layer by the other ONUs.



**Figure 28 - Transmission of initial GATE**

After receiving the initial GATE message, the ONU transmits the REGISTER\_ACK message to let the OLT know that it is joining the EPON. Figure 29 shows this message being sent. This is the first message transmitted by the ONU with the unicast LLID, and all other frames transmitted by the ONU while it is attached to the network will contain this LLID.



**Figure 29 - Transmission of REGISTER\_ACK**

### **Normal EPON operation**

Once the ONU has been registered and the discovery window has closed, the normal operation of the EPON resumes. During this mode of operation, the OLT uses an implementation dependent dynamic bandwidth allocation (DBA) algorithm to determine when to allow each of the individual ONUs to access the PON. The OLT maintains complete control over all ONUs, and therefore takes on the responsibility of making sure that no more than one ONU is scheduled to transmit at the same time. The OLT transmits normal GATE messages to grant the ONUs individual transmission windows, and the ONUs transmit the REPORT messages to request additional bandwidth.

### **Clause 64 Testing**

The remainder of this chapter deals with how to go about performing conformance testing of a device that supports Clause 64 functionality. Each of the optional and mandatory items will be covered individually. The reader should pay special attention to the grey “How to test” boxes.

There are just over forty different mandatory items listed within Clause 64 and one major option. In addition, a number of requirements that are not explicitly laid out in the PICS are also present. The primary option is whether the DUT is an OLT or ONU, as there are slightly different requirements for each. The descriptions that follow will state different procedures when necessary for both the OLT and ONU. The bulk of the requirements within this clause are contained within detailed state machines. In many instances, the OLT and ONU



must comply with different state machines, and each state machine may contain many individual requirements.

### **Generic MPCPDU description**

The generic structure of a multipoint control protocol data unit (MPCPDU) is shown in Table 3. These frames are simply normal 64-byte Ethernet frames that are not tagged, and have a number of fields fully specified by the standard. The destination address of these frames is the MAC Control Multicast address, 0x0180C2000001. The source address is the individual MAC address associated with the port through which the MPCPDU is transmitted. The length/type field is type encoded and should have the value 0x8808. The opcode is dependent on the type of frame being transmitted, and should take on a value between 0x0002 and 0x0006. The timestamp field contains the local time for when the frame is transmitted. The Data/Pad field contains the payload of the MPCPDU and should be no more than 40 bytes in length. Finally, the frame check sequence (FCS) field contains the CRC-32 checksum. All of the specific MPCP messages described later in this chapter are simply a subset of the generic MPCPDU.

**Table 3 - Generic MPCPDU**

<b>Field/Value</b>	<b>Length (bytes)</b>
Destination Address	6
Source Address	6
Length/Type = 0x8808	2
Opcode	2
Timestamp	4
Data/Reserved/Pad	40
FCS	4

**How to test:** The passive monitoring station is connected to observe transmissions from either the OLT or ONU. Each MPCPDU should contain an LLID that will be generated by the RS. The destination address of all MPCPDU should be 01-80-C2-00-00-01 or the unicast address of the port to which it is destined. The source address of all MPCPDU should be a unicast address associated with the port through which the frame was transmitted. The length/type field should contain a value of 0x8808. The opcode should contain a value between 00-02 and 00-06. The timestamp field should contain a valid timestamp generated by the local device. The data/PAD field should be no more than 40 bytes long. The DUT should never transmit a tagged MPCPDU. Each received MPCPDU is analyzed to make sure it follows these rules.

### **GATE description**

The GATE message is used to grant transmission to an ONU. A GATE message with the discovery flag set is used to grant transmission windows to ONUs that have not been registered on the EPON. A GATE message that does not have the discovery flag set is used to grant transmission windows to ONUs that have been registered on the EPON. Up to four grants can be contained within a single GATE message. A description of the contents of the GATE message is shown below in Table 4 and Table 5.

**Table 4 - Content of GATE**

Field/Value	Length
LLID = broadcast for discovery, otherwise unicast	
DA = MAC Control	6 bytes
SA = OLT MAC address	6 bytes
Length/Type = 88-08	2 bytes
Opcode = 00-02	2 bytes
Timestamp	4 bytes
Number of grants/flags	1 byte
Grant #1 start time	4 bytes
Grant #1 length	2 bytes
Sync time	2 bytes
Pad	31 bytes
FCS	4 bytes

**Table 5 - Content of GATE flags field**

Bit	Flag field	Values
0-2	Number of grants	0-4
3	Discovery	0/1, normal/discovery
4	Force Report Grant 1	0/1, no action/report
5	Force Report Grant 2	0/1, no action/report
6	Force Report Grant 3	0/1, no action/report
7	Force Report Grant 4	0/1, no action/report

**How to test:** Connect the passive sniffer between the OLT and an ONU, and monitor all GATE messages transmitted by the OLT. GATE messages issued for opening a discovery window should use the broadcast LLID. All other GATE messages should use the unicast LLID of the ONU. All GATE messages shall use an opcode of 00-02. The flags field should contain the necessary information as shown in Table 5, and all discovery GATE messages shall contain a single grant. The lengths of each grant are properly included in the GATE message. The start times of each grant are properly included in the GATE message, and the values shall satisfy the condition that Grant #n Start Time < Grant #n+1 Start

Time. The SyncTime is properly included in discovery GATE messages and absent from other GATE messages. The OLT should properly encode all other fields of the GATE messages as shown above.

### **REPORT description**

The REPORT message is used by the ONU to request one or more transmission windows so that data can be transmitted on to the EPON. The message contains the registration request and the number of grants the ONU can accomodate. A description of the contents of the REPORT message is shown below in Table 6 and Table 7.

**Table 6 - Contents of REPORT**

Field/Value	Length
LLID = unicast	
DA = OLT MAC address	6 bytes
SA = ONU MAC address	6 bytes
Length/Type = 88-08	2 bytes
Opcode = 00-03	2 bytes
Timestamp	4 bytes
Number of queue sets	1 bytes
Report bitmap	1 byte
Queue #0 report	0/2 bytes
Queue #1 report	0/2 bytes
...	...
Queue #7 report	0/2 bytes
PAD/Reserved	0-39 bytes
FCS	4 bytes

**Table 7 - Contents of REPORT bitmap field**

Bit	Flag field	Values
0	Queue 0	0 – queue 0 report not present 1 – queue 0 report present
1	Queue 1	0 – queue 1 report not present 1 – queue 1 report present
...	...	...
7	Queue 7	0 – queue 7 report not present 1 – queue 7 report present

**How to test:** Use the OLT emulator to register the ONU or the passive sniffer, and examine all REPORT messages generated by the ONU. The REPORT message should be transmitted with a unicast LLID and only by an registered ONU. The opcode should contain the value of 00-03. The number of queue sets contains the number of requests in the REPORT message. The report bitmap field indicates which queues are represented, as shown above. The Queue #n report represents the length of the queue. The DUT should properly encode all other fields of the REPORT messages as shown above.

### **REGISTER\_REQ description**

The REGISTER\_REQ message is used by the ONU to request registration on the EPON. The message contains the registration request and the number of grants it can be assigned. A description of the contents of the REGISTER\_REQ message is shown below in Table 8 and Table 9.

**Table 8 - Contents of REGISTER\_REQ**

<b>Field/Value</b>	<b>Length</b>
LLID = broadcast	
DA = OLT MAC address	6 bytes
SA = ONU MAC address	6 bytes
Length/Type = 88-08	2 bytes
Opcode = 00-04	2 bytes
Timestamp	4 bytes
Flags	1 bytes
Pending grants	1 byte
PAD/Reserved	38 bytes
FCS	4 bytes

**Table 9 - Contents of REGISTER\_REQ flags field**

Value	Indication	Comment
0	Reserved	Ignored on reception
1	Register	Registration attempt for ONU
2	Reserved	Ignored on reception
3	Deregister	Request to deregister ONU
4 - 255	Reserved	Ignored on reception

**How to test:** Use the OLT emulator to transmit various discovery GATE messages to the ONU, and monitor all REGISTER\_REQ messages that are transmitted by the ONU. The REGISTER\_REQ message should be transmitted with a broadcast LLID, and only by an ONU that has not yet been discovered. The opcode of the REGISTER\_REQ message should contain a value of 00-04. The flags field of the REGISTER\_REQ message should be encoded as shown above. The pending grants field should contain the maximum number of grants the ONU can handle. The DUT should properly encode all other fields of the REGISTER\_REQ messages as shown above.

### **REGISTER description**

The REGISTER message is used to formally register the ONU on the EPON. The message contains the assigned LLID and sync time that must be used by the ONU for all of its transmissions. The flags field either informs the ONU that the registration attempt was successful or that some problem exists with the process. A description of the contents of the REGISTER message is shown below in Table 10 and Table 11.



**Table 10 - Contents of REGISTER**

Field/Value	Length
LLID = broadcast	
DA = ONU MAC address	6 bytes
SA = OLT MAC address	6 bytes
Length/Type = 88-08	2 bytes
Opcode = 00-05	2 bytes
Timestamp	4 bytes
Assigned LLID	2 bytes
Flags	1 byte
Sync Time	2 bytes
Echoed pending grants	1 byte
Pad	34 bytes
FCS	4 bytes

**Table 11 - Contents of REGISTER flags field**

Value	Flag field	Comment
0	Reserved	Ignored on reception
1	Reregister	ONU should reregister
2	Deregister	Request to deallocate port
3	Ack	Registration successful
4	Nack	Registration attempt denied
5-255	Reserved	Ignored on reception

**How to test:** The DUT should transmit REGISTER messages using the broadcast LLID. The destination address of the REGISTER message shall be the unicast address of the ONU. The opcode of the REGISTER message shall be 00-05. The assigned port will contain a unique 16-bit value that is not 0x7FFF. The flags field of the REGISTER message will be properly encoded as shown above. The sync time will meet all of the proper requirements. The echoed pending grants shall match the value received from the ONU in the REGISTER\_REQ message. The DUT should properly encode all other fields of the REGISTER messages as shown above.

## **REGISTER\_ACK description**

The REGISTER\_ACK message is used by the ONU to acknowledge the registration message sent by the OLT. This frame is identified by the 00-06 opcode, and contains an echo of both the sync time of the OLT and of the assigned LLID. The flags field may either contain an acknowledgement or a negative acknowledgement. A description of the contents of the REGISTER\_ACK message is shown below in Table 12 and Table 13.

**Table 12 - Contents of REGISTER\_ACK**

<b>Field/Value</b>	<b>Length</b>
LLID = unicast	
DA = OLT MAC address	6 bytes
SA = ONU MAC address	6 bytes
Length/Type = 88-08	2 bytes
Opcode = 00-06	2 bytes
Timestamp	4 bytes
Flags	2 bytes
Echoed assigned port	1 byte
Echoed Sync Time	2 bytes
Pad	35 bytes
FCS	4 bytes

**Table 13 - Contents of REGISTER\_ACK flags field**

<b>Value</b>	<b>Indication</b>	<b>Comment</b>
0	Nack	Registration denied
1	Ack	Registration successful
2 – 255	Reserved	Ignored on reception

**How to test:** Use OLT emulator to register the ONU by following the proper discovery procedures. Observe management indicators and transmissions from the ONU. Repeat this process, varying the assigned LLID and sync time values. The DUT should transmit all REGISTER\_ACK messages using a unicast LLID. The DUT should properly transmit an opcode of 00-06 in the REGISTER\_ACK



message. The DUT should set the proper flags bits in the REGISTER\_ACK message. The DUT should properly echo the assigned LLID in the REGISTER\_ACK message. The DUT should properly echo the sync time in the REGISTER\_ACK message. The DUT should properly encode all other fields of the REGISTER\_ACK messages as shown above.

### **Timestamp verification**

The localTime variable is responsible for holding the value of the timestamp that is placed in all transmitted MPCPDU frames. It is a 32-bit value that is incremented in units of time\_quanta, where each time\_quanta is 16ns. At the OLT, the timer is based on the transmit clock and runs continuously. At the ONU, the timestamp is taken from each MPCPDU message received from the OLT and made to become the local time of the ONU. Following this procedure, the counter then increments synchronously with of the recovered data clock at the receiver. This means that the ONU updates its localTime variable with every received MPCPDU from the OLT.

**How to test:** Use the passive monitor to observe transmissions from either an OLT or an ONU, and examine the transmitted timestamps in outgoing MPCP messages. The timestamps should increment properly such that for a given frame size and amount of idle between frames, the next observed timestamp should be identical to the calculated timestamp. An independent timer can also be used to verify that the timestamp is incrementing properly.

### **OLT delay requirements**

Due to processing time, the OLT is responsible for not granting the ONU a time slot less than 1024 time\_quanta into the future. In addition, the OLT shall not issue more than one MPCP message every 1024 time\_quanta to a single ONU. The ONU is required to act on a received MPCP message within 1024 time\_quanta, and therefore the OLT cannot violate this requirement. It is interesting to point out that even if an ONU is able to process the MPCP messages very quickly (much less than 1024 time\_quanta), it is still not conformant for the OLT to break this rule. Some vendors have chosen to ignore this requirement in favor of improving performance of the network.

**How to test:** Instruct the OLT to transmit a number of GATE messages to the ONU and observe all transmissions from the OLT. The start of each time slot should be greater than 1024 time\_quanta into the future. No more than one MPCP message should be sent to the ONU every 1024 time\_quanta.

### **Implementation of mpcp timer**

The mpcp\_timer is a watchdog timer that will reset the MPCP on either the OLT or the ONU if the one-second timer expires without seeing any traffic from the link partner. At the ONU, the timer is reset every time the ONU receives a GATE message from the OLT. If the timer expires, the ONU will indicate to its client that it has been deregistered. Under normal conditions, the OLT will be generating up to 20 GATE messages per second.

#### **How to test:**

*Case 1: DUT is an ONU:* Proceed through the discovery process until an ONU

has registered with the OLT. Transmit two GATE messages to the ONU that have less than 1 second gap between them. Observe management indicators and transmissions from the ONU. Repeat these steps, increasing the gap between the GATE messages until the ONU stops transmitting frames to the OLT and deregisters itself. The time between these frames is the value of `mpcp_timer`.

*Case 2: DUT is an OLT:* Proceed through the discovery process until an ONU has registered with the OLT. Transmit two REPORT messages to the OLT that have less than 1 second gap between them. Observe management indicators and transmissions from the OLT. Repeat these steps, increasing the gap between the REPORT messages until the OLT transmits a REGISTER message with the deregister flag set. The time between these frames is the value of `mpcp_timer`.

### **Transmission in proper timeslot**

When the ONU receives a grant from the OLT, it must then prepare to transmit in the next assigned timeslot. To be as efficient as possible, the ONU should commence its transmission at the very beginning of the timeslot, or as close to the beginning as possible.

**How to test:** Instruct the ONU to transmit various frames to the OLT. Observe all transmissions from the ONU. During normal timeslots, the ONU should initiate transmission at the beginning of the timeslot. Although data will not begin until sometime after the timeslot starts, the ONU should turn on its laser and begin transmitting idle at the beginning of the timeslot.

## **RANDOM WAIT**

During the discovery window, each ONU will wait a random amount of time before transmitting the REGISTER\_REQ message. This random delay helps to minimize the number of collisions that will occur when multiple ONUs transmit at the same time during the discovery window. When a broadcast discovery GATE is sent, the DUT determines the value of maxDelay, and then chooses a uniformly distributed random integer between 0 and maxDelay. The DUT then waits that amount of time before beginning its transmission. When a unicast discovery GATE is sent, the DUT does not have to wait, and should begin transmitting the REGISTER\_REQ message at the beginning of the window.

### **How to test:**

*Part A: Random delay between 0 and maxDelay:* Instruct the OLT to transmit broadcast discovery GATE messages to the DUT such that the calculated value of maxDelay is 0. Observe management indicators and transmissions from the ONU. Repeat the above steps, using the same value of maxDelay, multiple times. Repeat, increasing the grant length so that the value of maxDelay will increment.

*Part B: Uniformly distributed random delay:* Instruct the OLT to transmit broadcast discovery GATE messages to the DUT such that the calculated value of maxDelay is 0. Observe management indicators and transmissions from the ONU. Repeat, using the same value of MaxDelay, multiple times. Repeat,



increasing the grant length so that the value of maxDelay will increment.

*Part C: Reception of unicast discovery GATE:* Instruct the OLT to transmit unicast discovery GATE messages to the DUT. Observe management indicators and transmissions from the ONU.

In Part A, the DUT should commence transmission following a random wait between 0 and maxDelay. In Part B, the length of the timer should be uniformly distributed between 0 and maxDelay. In Part C, the DUT should commence transmission at the beginning of the discovery window after reception of a unicast discovery GATE message.

### **Recognition of timestampDrift**

Due to the time sensitive nature of the EPON, it is necessary for both the OLT and ONU to adhere to a strict timing scheme. Although the OLT maintains the clock on which the EPON is based, the ONU is responsible for updating it's timestamps with information from the OLT. When the OLT receives a message from the ONU that has a timestamp, it parses the timestamp and calculates a new roundtrip time for that particular ONU (newRTT) and computes the difference between this value and the previously calculated roundtrip time (RTT). If the received timestamp from either the ONU or OLT is outside the bounds set by guardThresholdOLT or guardThresholdONU, respectively, then the timestampDrift variable should be set to true.

#### **How to test:**

*Case 1: DUT is an OLT:* Configure the ONU to transmit an MPCPDU message

with a timestamp such that  $\text{abs}(\text{newRTT} - \text{RTT}) > \text{guardthresholdOLT}$ . Observe management and transmissions from the OLT.

*Case 2: DUT is an ONU:* Configure the OLT to transmit an MPCPDU message with a timestamp such that  $\text{abs}(\text{timestamp} - \text{localTime}) > \text{guardthresholdONU}$ . Observe management and transmissions from the ONU.

The OLT should indicate timestampDrift when  $\text{abs}(\text{newRTT} - \text{RTT}) > \text{guardThresholdOLT}$ . The ONU should indicate timestampDrift when  $\text{abs}(\text{timestamp} - \text{localTime}) > \text{guardthresholdONU}$ .

### **OLT transmission of discovery GATE**

The discovery process is controlled by the OLT, specifically by the Multi-point MAC Control Client. During this handshaking process, an unregistered ONU may attempt to join the EPON by exchanging several MAC Control frames back and forth with the OLT. At the end of the process, the OLT assigns the ONU a unique identifier (LLID) and registers the ONU so that it may transmit MAC Client frames. The OLT is required to periodically open a window of time during which ONU registration may happen. Although the frequency of these windows and their length are implementation dependent, all ONUs should be allowed to join the EPON in a reasonable amount of time.

**How to test:** Observe GATE transmissions from the OLT with the discovery flag enabled, and log their frequency and length. Register multiple ONUs with the OLT and repeat step 1. The OLT should periodically transmit properly formed GATE messages on the broadcast channel with the discovery flag enabled. As

the number of registered ONUs increases, the frequency and length of the discovery windows may vary. For each discovery window, the OLT should transmit exactly one GATE message with the discovery flag set, and should not transmit any GATE messages with the discovery flag set outside of the discovery window.

### **OLT reception of REGISTER\_REQ**

The OLT accepts REGISTER\_REQ messages from ONUs that attempt to register during the discovery window. These messages are sent by the ONU to the broadcast MAC address of the OLT and should only be accepted during the discovery window. When these messages are received at other times, the OLT should not accept and act on them. When parsing the received REGISTER\_REQ, the OLT will pass up to its client the following information: status, flags, pending\_grants, and RTT. It is also possible for the ONU to transmit a REGISTER\_REQ to the OLT with an indication that it wants to be deregistered from the EPON. Under these circumstances, the OLT will deregister the ONU but will still only accept the REGISTER\_REQ during the discovery window.

**How to test:** Wait for or force the OLT to open a discovery window. Transmit a REGISTER\_REQ message to the OLT during a normal discovery window and observe management indicators and transmissions from the OLT. Repeat these steps outside a discovery window. The OLT should accept and act on any REGISTER\_REQ messages received within the discovery window. The OLT should not accept or act on any REGISTER\_REQ messages received outside

the discovery window.

### **OLT transmission of REGISTER**

Upon reception of a REGISTER\_REQ message from an ONU, the OLT will respond to the ONU with a REGISTER message. The REGISTER message will be sent to the unicast MAC address of the ONU on the broadcast channel. It will also echo the maximum number of pending grants that the ONU can handle based on the value received in the REGISTER\_REQ message from the ONU. During discovery, the flags field will contain a value of 0x03 (ack) or 0x04 (nack) to signify whether or not the requested registration has been successful or has been denied, respectively.

Outside of the discovery process, the OLT may transmit a REGISTER message to an ONU to explicitly ask it to reregister or deregister by using a flags field value of 0x01 (reregister) or 0x02 (deregister). This could happen if the ONU requests to be deregistered, if the OLT client makes a similar request, the mpcp\_timer expires, or there is too much drift in the timestamps.

#### **How to test:**

*Part A: Initial registration:* Wait for or force the OLT to open a discovery window. Transmit a REGISTER\_REQ message to the OLT during a normal discovery window. Observe management indicators and transmissions from the OLT.

*Part B: Deregistration due to mpcp\_timer timeout:* Completely register an ONU. Withhold all transmissions from the ONU to the OLT for at least mpcp\_timer in length. Observe management indicators and transmissions from the OLT.



*Part C: Deregistration due to ONU request:* Completely register an ONU. During a discovery window, force the ONU to send a REGISTER\_REQ message to the OLT with the deregister flag set. Observe management indicators and transmissions from the OLT.

*Part D: Deregistration due to OLT client request:* Completely register an ONU. Attempt to force the OLT to deregister the ONU. Observe management indicators and transmissions from the OLT.

*Part E: Deregistration due to timestamp drift:* Completely register an ONU. Force the ONU to transmit frames to the OLT such that it will set the timestampDrift variable. Observe management indicators and transmissions from the OLT.

*Part F: Deregistration due to lack of REGISTER\_ACK:* Proceed through the discovery process until the OLT transmits the initial GATE to the ONU. Prevent the ONU from responding with a REGISTER\_ACK message. Observe management indicators and transmissions from the OLT.

*Part G: Deregistration due to OLT Nack:* Proceed through the discovery process until the OLT transmits the initial GATE to the ONU. Force the OLT client to indicate a status of Nack. Observe management indicators and transmissions from the OLT.

In Part A, after reception of a REGISTER\_REQ message, the OLT should transmit a properly formed REGISTER message to the ONU, on the broadcast channel, that contains an echo of the pending grants, assigned LLID, and the proper flags. In Part B, upon completion of mpcp\_timer, the OLT should send a REGISTER message with the deregister flag to the ONU. In Part C, upon

reception of a REGISTER\_REQ during the discovery window with the deregister flag set, the OLT should send a REGISTER message with the deregister flag set to the ONU. In Part D, given an indication by the client to deregister the ONU, the OLT should transmit a REGISTER message with the deregister flag set to the ONU. In Part E, if the difference between timestamps becomes too great, the OLT should transmit a REGISTER message with the deregister flag set to the ONU. In Part F, upon completion of the discovery window without having received a REGISTER\_ACK, the OLT should transmit a REGISTER message with the deregister flag set to the ONU. In Part G, upon reception of a Nack from the client, the OLT should transmit a REGISTER message with the deregister flag set to the ONU.

### **OLT completion of discovery process**

After the OLT transmits the REGISTER message to the ONU, it will then transmit a normal GATE message on the unicast channel to prompt the ONU to send a REGISTRATION\_ACK message. Upon reception of a valid acknowledgement, the OLT will register the ONU and normal data transmission can begin.

#### **How to test:**

*Part A: Discovery NACK:* Proceed through the discovery process until the indication comes from the client to transmit the initial GATE to the ONU. During the appropriate time period, have the ONU transmit a REGISTER\_ACK message to the OLT that does not have the ACK flag set. Observe management indicators

and transmissions from the OLT.

*Part B: Discovery ACK:* Proceed through the discovery process until the indication comes from the client to transmit the initial GATE to the ONU. During the appropriate time period, have the ONU transmit a REGISTER\_ACK message to the OLT that does have the ACK flag set. Observe management indicators and transmissions from the OLT.

In Part A, upon reception of a REGISTER\_ACK with the flag != ACK, the OLT should discontinue attempting to register the ONU. This may be observed by the fact that no future unicast GATE messages will be sent to the ONU.

In Part B, upon reception of a REGISTER\_ACK with the flag = ACK, the OLT may complete the registration of the ONU. This may be observed by indication of additional GATE messages sent to the ONU.

### **ONU transmission of REGISTER\_REQ**

When an ONU needs to register itself on the EPON, it must first wait for a discovery window to open. Once the discovery window is open, the ONU can send a REGISTER\_REQ message to the OLT. This message contains the status of the process (whether the ONU is attempting to register or deregister) and the maximum number of future grants that the ONU can handle. It is possible when the ONU sends the REGISTER\_REQ message that it will collide with a message from a different ONU. When this happens, the ONU may choose to resend the REGISTER\_REQ message at the next opportunity it gets.

### **How to test:**



*Part A: REGISTER\_REQ during discovery window:* Instruct the OLT to open a discovery window at a given time in the future. Force the ONU client to begin the registration process. Observe management indicators and transmissions from the ONU.

*Part B: REGISTER\_REQ to deregister:* Complete the discovery process with the ONU so that it is registered with the OLT. Force the client to indicate that it wishes to deregister from the EPON. Observe management indicators and transmissions from the ONU.

In Part A, the ONU should transmit a REGISTER\_REQ message with the register flag set, during a discovery window. In Part B, The ONU should transmit REGISTER\_REQ message with the deregister flag set.

### **ONU reception of REGISTER**

After the ONU transmits the REGISTER\_REQ message, it waits to receive a REGISTER message from the OLT. This message will arrive on the unicast channel and will contain a positive or negative acknowledgement of the registration request along with an assigned LLID.

#### **How to test:**

*Part A: Lack of receipt of REGISTER:* Proceed through the discovery process until the ONU sends the REGISTER\_REQ message. Prohibit the OLT from sending a REGISTER message to the ONU. Before the mpcp\_timer expires, instruct the OLT to open another discovery window. Observe management indicators and transmissions from the ONU.

*Part B: Denial of registration request:* Proceed through the discovery process until the ONU sends the REGISTER\_REQ message. Instruct the OLT to send a REGISTER message to the ONU with the Nack flag set. Observe management indicators and transmissions from the ONU. Repeat while forcing the OLT to open a new discovery window before the ONU receives the REGISTER message.

*Part C: ONU wants to stop registration process:* Proceed through the discovery process until the ONU sends the REGISTER\_REQ message. Force the ONU client to issue a new request with a status set to deregister. Observe management indicators and transmissions from the ONU. Repeat while forcing the OLT to open a new discovery window before the ONU client issues its deregister request.

*Part D: Acceptance of registration request:* Proceed through the discovery process until the ONU sends the REGISTER\_REQ message. Instruct the OLT to send a REGISTER message to the ONU with the Ack flag set. Observe management indicators and transmissions from the ONU. Repeat, forcing the OLT to open a new discovery window before the ONU receives the REGISTER message.

In Part A, the ONU should transmit another REGISTER\_REQ during the next discovery window. In Part B, the ONU should indicate that its request has been denied and return to the WAIT state when the REGISTER message is received outside of the discovery window. In Part C, the ONU should abort the registration process and return to the WAIT state when the client request is



received outside of the discovery window. In Part D, the ONU should continue with the registration process when the REGISTER message is received outside of a discovery window.

### **ONU transmission of REGISTER\_ACK**

Upon reception of a REGISTER message from the OLT, the ONU will be assigned an LLID and will be given its initial GATE message after which it needs to acknowledge the registration with a REGISTER\_ACK message. The REGISTER\_ACK message will also contain an echo of the assigned LLID and syncTime variable that were received in the REGISTER message.

#### **How to test:**

*Part A: Negative acknowledgement:* Proceed through the discovery process until the ONU has received the REGISTER and initial GATE messages. Force the ONU client to issue a REGISTER\_ACK request with a Nack status. Observe management indicators and transmissions from the ONU.

*Part B: Acceptance of registration request:* Proceed through the discovery process until the ONU has received the REGISTER and initial GATE messages. Force the ONU client to issue a REGISTER\_ACK request with an Ack status. Observe management indicators and transmissions from the ONU.

In Part A, the ONU should transmit a REGISTER\_ACK message on the unicast channel during its initial GATE period that has the flag set to Nack and then return to the WAIT state. In Part B, the ONU should transmit a REGISTER\_ACK message on the unicast channel during its initial GATE period that contains the

assigned LLID, Ack flag, and echoed syncTime and proceed to the REGISTERED state.

### **ONU deregistration**

After registration is complete, the ONU will remain in the REGISTERED state until it is deregistered. There are several reasons that the ONU may become deregistered. It may decide on its own that it chooses to deregister or the OLT may inform the ONU that it is required to deregister.

#### **How to test:**

*Part A: Remote deregister:* Proceed through the discovery process until the ONU has been registered. Instruct the OLT to transmit a REGISTER message to the ONU with the deregister flag set. Observe management indicators and transmissions from the ONU

*Part B: Deregister due to timestamp drift:* Proceed through the discovery process until the ONU has been registered. Force the received timestamps to drift greater than the allowable margin. Observe management indicators and transmissions from the ONU.

*Part C: Remote reregister:* Proceed through the discovery process until the ONU has been registered. Instruct the OLT to transmit a REGISTER message to the ONU with the reregister flag set. Observe management indicators and transmissions from the ONU.

*Part D: Local deregister:* Proceed through the discovery process until the ONU has been registered. Instruct the ONU to transmit a REGISTER\_REQ message

to the ONU with the deregister flag set. Observe management indicators and transmissions from the ONU.

In Part A, the ONU should indicate that it has been deregistered and return to the WAIT state. In Part B, the ONU should indicate that it has been deregistered and return to the WAIT state. In Part C, the ONU should return to the REGISTER\_PENDING state and send a new REGISTER\_ACK message to the OLT. In Part D, the ONU should remain in the REGISTERED state after transmitting the REGISTER\_REQ message.

### **Generation of grants**

In order to keep the EPON alive, the OLT must periodically grant the ONU an available time slot, even if the ONU has not requested any bandwidth. During this timeslot, the ONU will generate a periodic REPORT message. The combination of these GATE and REPORT messages serve the purpose of resetting the watchdog timers in both the OLT and ONU and preventing the ONU from being deregistered. When issuing grants, the OLT must not issue more than the number of grants supported by the ONU. This value is relayed to the OLT during the discovery process in the REGISTER message.

#### **How to test:**

*Part A:* Configure the ONU such that it has no data to transmit and observe all transmissions from the OLT.

*Part B:* Configure the ONU such that it is attempting to transmit various amounts of data and observe all transmissions from the OLT.



The OLT should periodically grant a timeslot to each registered ONU so that the mpcp\_timer does not expire. These GATE messages may have the force report bits set in them. The OLT should not issue more grants than the number supported by the ONU.

### **Transmission of unsolicited REPORT**

Once an ONU has become registered on the EPON, it must periodically give a status update so that the OLT knows the ONU is still alive. These status updates come in the form of REPORT messages. Additionally, each time that the ONU wishes to transmit data frames, it must request this through a REPORT message that indicates how much data the ONU has to transmit.

#### **How to test:**

*Part A: Periodic transmission of REPORT:* Proceed through the discovery process until an ONU has registered with the OLT and prohibit the ONU from requesting the transmission of data frames.

*Part B: Normal operation transmission of REPORT:* Proceed through the discovery process until an ONU has registered with the OLT, and force the ONU to request the transmission of data frames.

*Part C: ONU becomes deregistered:* Proceed through the discovery process until an ONU has registered with the OLT and force the ONU to become unregistered.

In Part A, the ONU should transmit an empty REPORT message every 50 ms. In Part B, the ONU should transmit REPORT messages that contain the

appropriate number of queue status reports. In Part C, the ONU should stop transmitting REPORT messages as soon as it becomes deregistered.

### **GATE processing by ONU**

Upon reception of a GATE message from the OLT, the ONU needs to process the message and store the next set of grant start times and lengths. There are a number of conditions that the ONU needs to check in order to accept and act on a grant. According to Figure 64-28 of the IEEE 802.3-2005 standard, the ONU will process at least one grant even if the number of grants, as determined by the flag bits, is zero. This could potentially create a number of problems. Devices should ignore the contents of the GATE message if the number of grants is set to zero. Additionally, the grant start time cannot be too soon or too far in the future in order to be acted upon. Finally, the size of the grant needs to be a minimum size in order to be acted upon.

#### **How to test:**

*Part A: Number of grants is zero:* Instruct the OLT to transmit a GATE message to the ONU. The message contains one or more grants, but the number of grants in the flag variable is set to zero.

*Part B: Grants too far ahead in the future:* Instruct the OLT to transmit a GATE message to the ONU. The message contains one or more grants that have a start time of greater than max\_processing\_time.

*Part C: Grants too soon:* Instruct the OLT to transmit a GATE message to the ONU. The message contains one or more grants that have a start time of less

than min\_processing\_time.

*Part D: Grants not long enough:* Instruct the OLT to transmit a GATE message to the ONU. The message contains one or more grants that have a length less than laserOnTime + syncTime + laserOffTime + tailGuard.

The DUT should not accept and act on grants included in a GATE message if the number of grants is zero. The DUT should not accept and act on grants that are greater than 1 second in the future. The DUT should not accept and act on grants that are less than 16.384 us in the future. The DUT should not accept and act on grants that are less than the minimum required time.

## **CHAPTER VII**

### **CONFORMANCE TESTING OF CLAUSE 60 PMD**

This chapter begins with an overview of the Clause 60 specification, and then goes into explicit detail on how to best perform conformance testing on a Clause 60 capable device. A discussion of the tools, methodology, and test suite are provided for the reader's consideration. Due to the test equipment and procedures that are necessary to perform comprehensive verification of the optical transmitter and receiver, a number of tests have not been included in this document. The reader is urged to consult the standard as the definitive guide to the tests that are necessary.

#### **Overview of Clause 60**

Clause 60 of IEEE Std. 802.3-2005 defines the Physical Medium Dependent (PMD) sublayer and medium for 1000BASE-PX10 and 1000BASE-PX20 networks. The four main areas of importance of this clause are: optical transmitter specifications, optical receiver specifications, measurement methodologies, and the optical channel. Although this is one of the shorter clauses, there are still quite a few requirements that need to be met. As opposed to other clauses, the PMD clauses typically contain information regarding accepted testing methods, and that text takes up a substantial portion of the clause.

### **Transmitter specifications**

The transmitter specifications are almost entirely specified in two tables within Clause 60, Table 60-3 for 1000BASE-PX10 transmitters, and Table 60-6 for 1000BASE-PX20 transmitters. These two tables contain separate specifications for both the upstream and downstream lasers. The transmitter requirements include a number of timing and power characteristics that must be met by the DUT.

### **Receiver specifications**

The receiver requirements are specified in Table 60-5 for 1000BASE-PX10 devices and in Table 60-8 for 1000BASE-PX20 devices. Although there are a number of important characteristics, the stressed receiver sensitivity test is arguably the most important. This test creates a worst-case test signal that is presented at the receiver of the DUT. The DUT should be able to operate with a BER of  $10^{-12}$  or better while receiving this signal. However, due to the complexity of this measurement, it is not often performed and it is very difficult to find dedicated test equipment to perform the measurement.

### **Measurement methodologies**

Much of Clause 60 is devoted to text describing how to perform the various optical transmitter and receiver tests. In instances where accepted procedures are already well defined and documented, such as with optical power testing, these international standards and documents are referenced within the standard. For new tests or tests that require modified procedures, the complete methodologies are presented in the text. The standard makes every attempt to

be agnostic with regard to the test equipment used and never states a preference for a specific company's test equipment. However, in some instances, there may only be a very small number of choices when purchasing equipment for some of these tests.

### **Optical channel**

There are two optical channels defined in Clause 60, a 10 km channel and a 20 km channel. The channel is specified as including everything from the patch cords connecting the ONU and OLT all the way through the 10 km or 20 km of optical fiber as well as the splitter. The fiber is allowed to have an attenuation of up to 0.4 dB/km at 1310 nm and 0.35 dB/km at 1550 nm, with a zero dispersion wavelength between 1300 nm and 1324 nm. Most of the insertion loss, therefore, comes from the passive optical splitter(s) that may be inserted in the link between the ONU and OLT.

### **Clause 60 Testing**

The remainder of this chapter deals with how to go about performing conformance testing of a device that supports Clause 60 functionality. The reader should pay special attention to the grey "How to test" boxes. When necessary, specific test setups or equipment that were first examined in Chapter 4 are referenced here.

### **Transmitter testing**

As mentioned above, the transmitter specifications are contained within Table 60-3 and Table 60-6 of the standard. A subset of the transmit characteristics are covered in the following test procedures. For all transmitter



tests, the DUT is connected to the real-time digital oscilloscope as specified in Chapter 4 and using the setup shown in Figure 8 or an optical spectrum analyzer (OSA).

Many modern oscilloscopes contain built-in automated test modes or measurements for general optical signals, and even for selected specific technologies. However, since different manufacturers may use slightly different algorithms in performing their measurements and calculations, the UNH-IOL attempts to perform all measurements using their own algorithms developed with Matlab.

### **Signaling speed**

The signaling speed is specified to be 1.25 GBd  $\pm$  100 ppm. This means that the clock associated with the serial bit stream transmitted by the DUT must be within a certain range. As the range becomes tighter, the cost of the clock increases. EPON uses a 100 ppm clock to allow the developer freedom to choose a less expensive clock and to be compatible with other one-gigabit optical Ethernet technologies. The actual range is calculated as follows:

$$\begin{array}{rccccccccc} \text{Serial data rate} & / & 1e6 & * & 100 & = & \pm \text{signaling range} \\ 1.25e+9 & / & 1e6 & * & 100 & = & \pm 125 \text{ kHz} \end{array}$$

**How to test:** The DUT is configured to transmit the jitter test pattern, as referenced in Chapter 4 of this document. The jitter test pattern will provide the transmitter with one of the more difficult patterns, and will help to stress the clocking of the transmitter. The waveform is captured by the DSO and then processed to extract the clock and compute an average baud rate over the

captured signal. Any device that has an average baud rate inside the range of 1.25 GBd  $\pm$  100 ppm passes the test. Note that for ONU devices, it may be necessary to force the ONU to turn its laser on continuously so as to facilitate the measurement.

### **Average launch power**

The average launch powers for each of the PMDs are defined in Table 59-3 and Table 59-6 of the standard for three instances: max, min, OFF. The maximum and minimum launch powers are defined to be the highest and lowest powers, respectively, that can be measured at the transmitter of the DUT. The OFF power is the highest power that can be measured at the transmitter of the DUT when the laser is disabled. Table 14 shows the defined average launch powers for each of the PMDs defined in Clause 60.

**Table 14 - EPON Launch powers**

	max (dBm)	min dBm)	OFF (dBm)
1000BASE-PX10-D	+2	-3	-39
1000BASE-PX10-U	+4	-1	-45
1000BASE-PX20-D	+7	+2	-39
1000BASE-PX20-U	+4	-1	-45

**How to test:** Configure the DUT so that it is transmitting any valid 8B/10B data pattern, and connect the DUT to the DSO. Measure the average launch power with the transmitter enabled and with the transmitter disabled by creating an eye diagram from the captured signal. The methods used for obtaining the measurement are described in Chapter 4. The average launch power with the transmitter enabled shall be between the appropriate limits, as shown in Table 14



for each PMD. The average launch power with the transmitter disabled shall be less than the values specified in Table 14 for each PMD.

### **Extinction ratio and optical modulation amplitude**

The extinction ratio, or the power ratio between the average HIGH and LOW levels, is specified to be a minimum 6 dB for each of the PMDs defined in Clause 60 of the standard. The extinction ratio is one of the most important transmit parameters, because it helps to define how much room will be between the HIGH and LOW levels on an eye diagram, which is also how the receiver will sense the signal. Transmitters that have higher extinction ratios will perform much better than those with low extinction ratios. The optical modulation amplitude is a relatively new way of looking at the transmitted power and the difference between the HIGH and LOW levels. Table 15 shows the OMA requirements for the four different EPON PMDs. The extinction ratio and OMA values are related, as the extinction ratio is the ratio of HIGH and LOW levels, and the OMA is the LOW level subtracted from the HIGH level.

**Table 15 - EPON OMA levels**

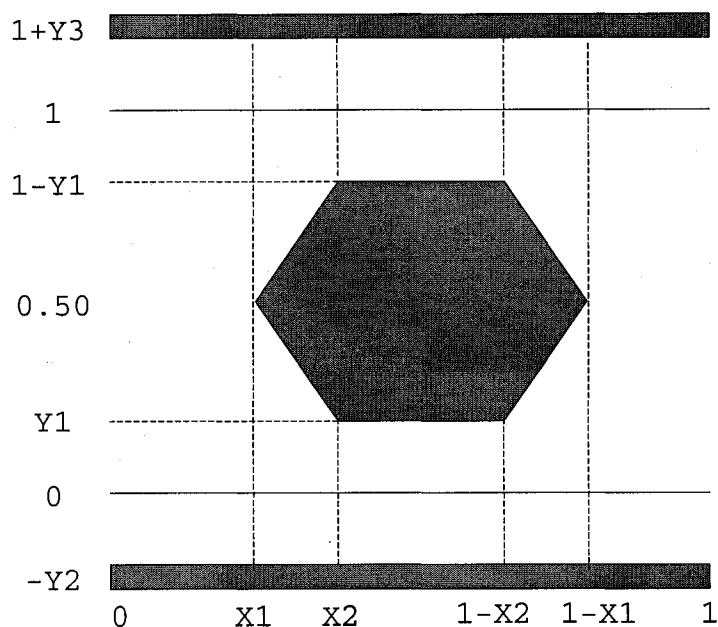
<b>PMD</b>	<b>Minimum OMA (dBm)</b>
1000BASE-PX10-D	-2.2
1000BASE-PX10-U	-0.22
1000BASE-PX20-D	+2.8
1000BASE-PX20-U	-0.22

**How to test:** The extinction ratio and OMA levels are performed on the DUT while it is transmitting the /12/ pattern. As with the previous power

measurements, an eye diagram is generated on the oscilloscope and analyzed using the methods described in Chapter 4. The measured extinction ratio should be greater than 6 dB, and the measured OMA value should be greater than the appropriate value shown in Table 15.

### **Transmitter eye mask**

The transmitter eye mask for EPON is the same for both the 10 km and 20 km devices, and is shown below in Figure 30. Additionally, the coordinates for the figure are shown in Table 16. As previously noted, the optical eye diagram is relative to the average HIGH and LOW powers produced by the transmitter of the DUT, and must be scaled to meet these powers. In the figure, there are grey areas above and below the mask, and within the mask itself. For a device to successfully meet the transmitter eye mask requirements, none of the grey areas may be violated.



**Figure 30 - Transmitter eye mask**

**Table 16 - Eye mask coordinates**

Variable	Value
X1	0.22
X2	0.375
Y1	0.20
Y2	0.20
Y3	0.30

**How to test:** The DUT is configured to transmit any valid 8B/10B pattern. Since a number of tests already require the jitter test pattern to be used, and since this pattern is one of the more stressful patterns, it is typically used for this test, too. The waveform is captured by the DSO and analyzed such that the clock is extracted and an eye diagram is built-up from the captured bits. Any violations of the mask will mean that the DUT does not pass the test. Many current

oscilloscopes have automated filters and masks already built into the scope software.

## **Receiver specifications**

### **Receiver sensitivity**

There are two types of receiver sensitivity tests defined in Clause 60, one for normal sensitivity, and one for stressed receiver sensitivity. The stressed sensitivity test applies a worst-case signal at the receiver of the DUT, whereas the sensitivity test applies a signal with a minimal extinction ratio and power at the receiver. The values for both tests are shown below in Table 17.

**Table 17 - Receiver sensitivity values**

<b>PMD</b>	<b>Sensitivity</b>	<b>Stressed Sensitivity</b>
1000BASE-PX10-D	-24 dBm	-22.3 dBm
1000BASE-PX10-U	-24 dBm	-21.4 dBm
1000BASE-PX20-D	-27 dBm	-24.4 dBm
1000BASE-PX20-U	-24 dBm	-22.1 dBm

**How to test:** Configure the DUT to transmit the jitter test pattern using the test setup as specified in Figure 13. Configure the test station to begin transmitting the jitter test pattern to the DUT at an initial power of X dBm. Observe all management status indications on the DUT, taking note of CRC errors and other frame errors. Decrease the power and repeat the measurement of BER at each power level. At the completion of the test, a plot will be created from which the power level that supports a BER of  $10^{-12}$  can be extrapolated. This value must be at least the value specified in Table 17.

## **CHAPTER VIII**

### **CONCLUSION**

This document has laid the groundwork for a comprehensive test solution that can be provided by the UNH-IOL to the EPON community. Test suites derived from the EPON standard according to the methodologies described in this thesis can be defined and implemented to test the protocol and physical layers of EPON devices. The code written throughout this project can be used and modified to create hundreds or thousands of additional test vectors to perform even more comprehensive testing, or specialized testing that may be desired by particular vendors.

Although EPON is a relatively new technology, it has received wide support in many areas of the world and is undergoing mass deployment. To date, there has been no public forum in which EPON vendors could come together to perform conformance and interoperability testing. This project establishes the UNH-IOL as such a forum. Even though UNH-IOL has not been fully utilized for the testing of EPON up until now, there are signs that this is about to change. There are more and more EPON silicon vendors now making themselves heard, and the push from industry and the service providers will continue to move companies away from proprietary EPON solutions and toward standards compliant and interoperable EPON solutions. As more and more companies come out with products, there will be a desire to create and use uniform test plans and methodologies for everyone to follow. With the complete

development of the flexible and extensible EPON test system, as described within this document, the UNH-IOL is ready and willing to provide EPON test services to those vendors that desire it. Through the development of public test suites and industry-reviewed methods and procedures, the UNH-IOL presents the perfect forum for the EPON industry to come together and test its products.

As previously mentioned, the IEEE 802.3 Working Group has recently commissioned a study group to investigate higher speed EPON physical layers. At the time of publication, there have been several proposals presented to the study group regarding a 10 Gb/s downstream and 10 Gb/s upstream solution as well as a 10 Gb/s downstream and 1 Gb/s upstream solution. In both instances, the desire is to support both the existing MPCP and the installed fiber infrastructure. One of the primary areas under investigation for new technology is that of burst mode lasers operating at 10 Gb/s. Although this project will likely not become an official standard for a number of years, it is time to start investigating and thinking about how this technology will be tested once it is available. Careful consideration of testing procedures and methodologies now will help to accelerate both the development and deployment of 10 Gigabit EPON, later.

Finally, the advent of higher speed EPON will certainly drive the market to create additional legacy EPON devices at lower cost or with additional services. The market for testing EPON, at all speeds, is just beginning to grow and will be around for a long time to come.

## **LIST OF REFERENCES**

- [1] IEEE Std 802.3<sup>TM</sup>-2005. New York: IEEE, 2005.
- [2] G. Kramer, B. Mukherjee, and G. Pesavento, "Interleaved Polling with Adaptive Cycle Time (IPACT): A Dynamic Bandwidth Distribution Scheme in an Optical Access Network", *Photonic Network Communications*, vol. 4, no. 1 pp. 89-107, January 2002.
- [3] H. Frazier, G. Pesavento, "Ethernet takes on the first mile", *IT Professional*, vol. 3, issue 4, pp. 17-22, Jul-Aug 2001.
- [4] G. Kramer, G. Pesavento, "Ethernet passive optical network (EPON): building a next-generation optical access network", *IEEE Communications Magazine*, vol. 40 issue 2, pp. 66-73, February 2002.
- [5] J. R. Stern, C.E. Hoppitt, D.B. Payne, M.H. Reeve, and K.A. Oakley, "TPON – A passive optical network for telephony," *Fourteenth European Conference on Optical Communication (ECOC'88)* vol.1, pp. 203-206, Brighton, UK, September 1988.
- [6] Glen Kramer. Ethernet Passive Optical Networks. New York: McGraw-Hill, 2005.
- [7] Michael Beck. Ethernet in the First Mile. New York: McGraw-Hill, 2005.
- [8] Wael Diab and Howard Frazier Ethernet in the First Mile: Access for Everyone. New York: IEEE, 2006.
- [9] 10GEPON Study Group,  
[http://grouper.ieee.org/groups/802/3/10GEPON\\_study/index.html](http://grouper.ieee.org/groups/802/3/10GEPON_study/index.html)
- [10] "Fast Ethernet Consortium Clause 25 Physical Medium Dependent Test Suite",  
[ftp://ftp.iol.unh.edu/pub/ethernet/test\\_suites/CL25\\_PMD/PMD\\_Test\\_Suite\\_v3.4.pdf](ftp://ftp.iol.unh.edu/pub/ethernet/test_suites/CL25_PMD/PMD_Test_Suite_v3.4.pdf)
- [11] "Straight Line BER Plots",  
<http://circadiant.com/resources/techbriefs/pdf/tb010.pdf>