4

Operation, Administration and Maintenance in All Optical Networks

Prepared by:
Carol Baker

Supervised by:
Dr. D.M. Mashao

Department of Electrical Engineering
University of Cape Town
2001

This dissertation is submitted to the University of Cape Town in fulfillment of the academic requirements for the Degree of Master of Science in Electrical Engineering

Synopsis

The phenomenal growth rate of Internet traffic as well as the increasing demand for high bandwidth services such as video-on-demand (VoD), high definition TV (HDTV), video-conferencing etc., have created a new networking environment in which flexibility, scalability and high bandwidth capacity are of utmost importance. All Optical Networks has emerged as a promising technology to deliver the high capacity demands of current and envisaged future applications.

DWDM systems are currently being deployed in an attempt to increase the supply bandwidth of service providers. These systems have resulted in high traffic volumes (DWDM systems operating in excess of 1Tbps have already been demonstrated) being transported and network failure could lead to a considerable loss of data. Network management (NM) has therefore become an essential feature in these high capacity networks.

Operations, Administration and Maintenance (OAM) is the collection of processes that support the network management function. OAM incorporates performance monitoring, fault detection, fault localisation and configuration. The performance monitoring function monitors the performance parameters of the network elements and signal. If and when a fault is detected, the OAM procedure reports the occurrence to the operating system (OS) of the NM system.

IP-based services are becoming the dominant transmission technology and this trend is expected to continue as the multimedia age comes to the fore. IP traffic is inherently bursty in nature and the related bandwidth requirement for these services changes frequently. This could result in new managed objects (MOs) being added to the network, as lightpaths are set-up and torn-down to accommodate these changing demands. The NM system (NMS), operating systems (OSs) and managing elements must be able to cope with these changing conditions within the network, without overloading its own resources.

In this study a generic OAM model is proposed which takes into consideration these changing conditions. The model proposes that a network be logically divided into OAM subnetworks, which are further sub-divided into OAM domains. The performance of the proposed architecture is evaluated in terms of the notification load which each managing element (referred to as a gateway NE) or OS has to handle.

In order to do this a good knowledge of the optical networking components as well as the OAM procedures is required. This study focuses on the performance monitoring and fault detection functionality associated with OAM. The proposed architecture is explained with the aid of a point-to-point network. The explanation of the performance monitoring and fault detection techniques (SDH emulation) is also done using the same network example. The point-to-point network was then simulated in the optical network simulation environment and the optical performance parameters (optical channel power and OSNR) were monitored.

In an attempt evaluate the performance of the proposed OAM architecture, a simple NM tool which implements the performance monitoring and fault detection techniques, is developed. The input to the tool is the OptSim performance data into which periodic failures have been incorporated. The output is the number of notifications generated and sent to the gateway NEs and OS. The results are compared for the case where the proposed architecture is implemented with the case where no sub-division into OAM domains is considered. The results show that for the proposed scheme, the overall total number of notifications generated is increased. However, the number of notifications received by the OS decreases. Furthermore, the notifications generated are divided amongst the GNE and in so doing it distributes the load that they experience. This result is important because it means that in a network where demand changes frequently, OSs will not be overloaded in the event that network failures occur.

This study considered only static NEs and also, it did not take into account all the OAM procedures. It is recommended that the proposed architecture be evaluated for configurable NEs as well as the other OAM functions.

Acknowledgements

Firstly, I would like to thank God, for it is He who makes all things possible.

To my supervisors, Drs. Mashao and Nleya, thank you for your support and guidance throughout my work on this project. Also to Mr. Ventura, Head of the Communications Research Group (CRG) at the University of Cape Town as well as Professor Braae, Head of the Electrical Engineering Department at the University of Cape Town, thank you for your encouragement and excellent advice.

I would also like to express my sincere gratitude to my relatives, in particular my parents and siblings. Thank you for your support and for believing in me. Also to my boyfriend, Joseph Motaung, thank you for your love, support and patience. My dear friends, William, Rewayi, Tanger, Liesel and Angela, thank you for the encouragement during difficult times.

My colleagues at the CRG, in particular, André, Sven, Kevin, Stuart and José, thank you for your technical support, advice and encouragement. Also to the secretaries, Karen and Lesley, thank you for your assistance and encouragement.

Gabriella Gabino, Aldo Sion and the other support staff at Artis, thank you for your assistance with the implementation and testing of the OptSim model.

Dr. Bjarke Skjoldstrup (Tele Danmark), thank you for helping me to understand the Optical Layer Network structure. Also Dr. Ori Gerstel (Xros), thank you for taking the time to explain some network management concepts.

Finally, I wish to thank my bursar, Telkom for granting me the opportunity to work on this project. Also to Robert van Zyl, my Telkom mentor, thank you for your assistance and support throughout.

God bless you all!

Table of Contents

List of Illustrations

Chapter 1

Introduction

The last decade in the 20^{th} century has brought about an amazing change in networking technologies and communications as a whole. The phenomenal growth rate of Internet traffic as well as the increasing demand for high bandwidth services such as video-on-demand (VoD), high definition TV (HDTV), video-conferencing etc., have created a new networking environment in which flexibility, scalability and high bandwidth capacity are of utmost importance. All Optical Networks (AONs) have all this to offer!

This chapter serves as an introductory overview to optical communications as a whole and some of the topics introduced here will be re-visited in chapter 2. The first section begins with a discussion on the background of optical communications and briefly reviews the physics concepts on which optical communications is based. Thereafter, the spectrum of optical media and optical components are introduced. It then discusses the limitations that optical components impose on the performance of optical networks. A brief review is given of the three different approaches for implementing multiplexing in an optical network. Thereafter optical switching methods and challenges relating to optical buffering are discussed. Section 1.7 serves as an introduction to chapter three and compares the management of legacy networks to the structure that has been proposed for AONs. Finally, the thesis objectives are stated and the thesis outline, provided.

1.1 Background on Optical Communications

Optical communications has its origins back in the 1950s, when the possibility of transmitting data over optic fiber cable was first realised. The field of lightwave (optical) communication is based on the concepts of refraction and total internal reflection in physics. When light travels from one medium to another it experiences varying degrees of refraction and reflection, which depend on the angle of the incident light and the refractive indices of the media. The law of reflection, which states that the angle of incident light equals the angle of the reflected light and Snell's law, which is given by:

$$n_1 \cdot \sin(\theta_1) = n_2 \cdot \sin(\theta_2) \tag{1.1}$$

govern the phenomenon of refraction. In equation (1.1), n_1 and n_2 are the refractive indices of the media and, θ_1 and θ_2 are the angle of incidence and angle of refraction, respectively. The refractive index of a material (n_{mat}) is defined as:

$$n_{mat} = \frac{c}{v_{mat}} \tag{1.2}$$

where c is the speed of light in a vacuum and v_{mat} is the speed of light in the material. By using Planck's law, which is stated in equation (1.3), equation (1.2) can be re-written as

$$f = \frac{c}{\lambda} \tag{1.3}$$

$$n_{mat} = \frac{f \cdot \lambda}{v_{mat}} \tag{1.4}$$

from which it is clear that there exists a relationship between the frequency, wavelength, velocity of light in a given material and the refractive index of the material. This inter-dependency has inherent advantages and disadvantages. The advantage is the accessibility of a third dimension, namely the wavelength domain (in addition to time and space), for the implementation of communication concepts such as multiplexing, switching and routing. The wavelength-frequency-velocity inter-dependency has also proved useful in the design of optical components, for example filters. A disadvantage, however, is that it gives rise to the phenomenon of dispersion, which imposes a limitation on the performance of optical networking components and therefore also the performance of the network. In the next section the spectrum and bandwidth potential of optic fiber is examined.

1.2 Spectrum of Optical Media

Initially, optic fibers were deployed as "fast electric wires" [7]. Optical fiber transmission suffers from low attenuation (typically 0.2dB/km or less) and low noise power, which means that it is more immune to bit errors. Figure 1.1 below shows a plot of fiber attenuation vs. frequency.

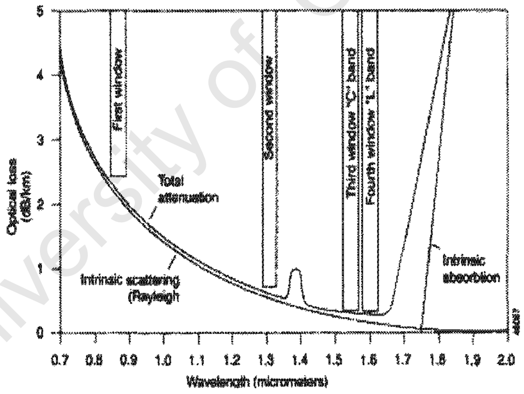


Figure 1. 1 Attenuation vs. Frequency curve for optical fiber

There are three passband windows, at 0.85, 1.3 and 1.55μm, which have been or are being used in optic fiber communication systems. The width of each of these passbands is approximately 0.2μm. The first commercial deployment of fiber optic systems operated in the first window (i.e. 0.85μm), but subsequent improvements in optic fiber fabrication resulted in systems operating in the second (i.e. 1.3μm), and the third and fourth (i.e. 1.55μm) passband windows.

The bandwidth of each of these passband windows can be calculated by taking the derivative of Planck's law, from which equation (1.5) is obtained.

$$\frac{\partial f}{\partial \lambda} = -\frac{c}{\lambda^2} \tag{1.5}$$

Then for a small interval $\Delta\lambda$ around λ_0 the corresponding window Δf can be calculated. Using this result it is found that the bandwidth associated with each of these passband windows is approximately 25THz. As a result of this huge bandwidth potential offered, optic fiber transmission has emerged as the dominant mode of transmission, particularly in long-haul networks. Since transmitters, which transmit at 25Tbps, are not yet realisable other means had to be devised to tap into this huge bandwidth potential offered by optic fiber. In the following section, three multiplexing schemes, which enable more of this potential bandwidth to be utilised, are presented.

1.3 Multiplexing Techniques for Optical Networks

Three multiplexing techniques have been identified for tapping into the huge bandwidth potential offered by optic fiber: Optical Time Division Multiplexing (OTDM), Optical Code Division Multiplexing (OCDM) and Dense Wavelength Division Multiplexing (DWDM).

OTDM is an extension and expansion of the traditional TDM. A schematic representation of TDM and OTDM is shown in figure 1.2 (*i*) and (*ii*), respectively. In TDM, the multiplexing operation is performed in the electrical domain. Digital data streams are multiplexed together by interleaving bits from the various data signals. Each bit is allocated a time slot in a digital frame. This electrically multiplexed signal then externally modulates a laser source before being transmitted onto an optic fiber link, as illustrated in figure 1.2(*i*).

OTDM is similar to TDM in that it also only uses one laser source. A train of narrow pulses, of width τ, is generated from this laser source (e.g. a mode-locked laser) and is then distributed to all the transmitters. At the transmitters the data is encoded onto the optical pulse streams using optical modulators. The data from each modulator is then multiplexed by allocating a time slot in the bit period (T), using fixed optical delay lines. So, in the case OTDM, the optical pulses are allocated to the various time slots. The delay lines ensure that data (which has been encoded onto the optical pulses) from a particular transmitter is always assigned or allocated to the same time slot. This process is illustrated in figure 1.2(*ii*).

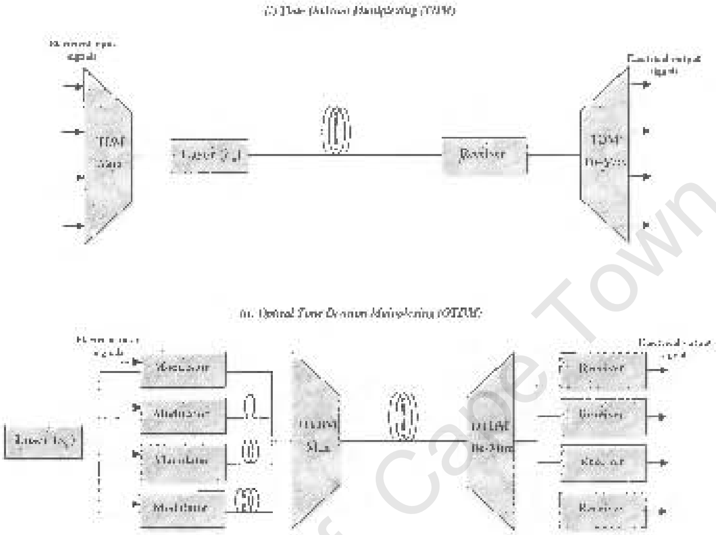


Figure 1.2 Illustration of (*i*) TDM and (*ii*) OTDM

OCDM draws from the spread-spectrum-based wireless CDMA protocol. In OCDM, the signals are transmitted simultaneously and on the same frequency. Each signal is optically encoded by an orthogonal code so that they do not interfere with each other. A block diagram of an OCDM system is shown in figure 1.3 below.

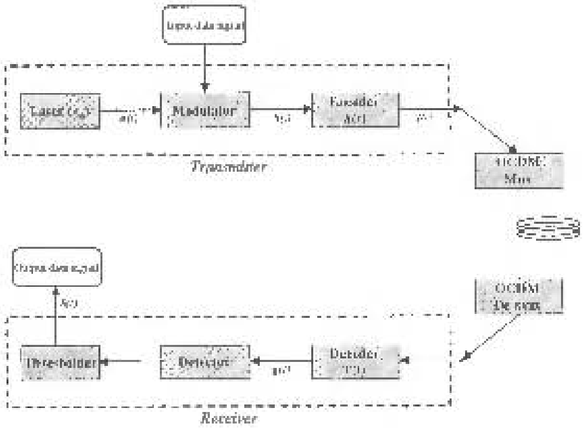


Figure 1.3 Block diagram of an OCDM system

In the transmitter, a laser source is externally modulated by a digital data stream to form $\mathbf{S}(t)$. The encoded signal is formed by the convolution product of $\mathbf{S}(t)$ and an orthogonal code $\mathbf{X}(t)$, which consists of a finite number of discrete elements $(\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_N)$. The process is represented by

$$\mathbf{f}(t) = \mathbf{X}(t) \otimes \mathbf{S}(t) \tag{1.6}$$

where $\mathbf{f}(t)$ is the encoded signal [41]. The encoded signal is then transmitted over the optic fiber link. At the receiver, the received signal is convolved with a correlation function to yield the decoded signal $\mathbf{g}(t)$. The process is represented by

$$\mathbf{g}(t) = \mathbf{f}(t) \otimes \mathbf{C}_{XY}(t) \tag{1.7}$$

where

$$\mathbf{C}_{XY}(t) = \int_0^{T_0} \mathbf{Y}(t')\mathbf{X}(t'+t)dt' \tag{1.8}$$

is the correlation function of $\mathbf{X}(t)$ and $\mathbf{Y}(t)$. The encoder and decoder pair operates as an optical phase-matched filter [41]. If $\mathbf{X}(t)$ and $\mathbf{Y}(t)$ are identical, then $\mathbf{C}_{XY}(t)$ is the auto-correlation function and $\mathbf{g}(t)$ will have peaks at the '1' bit positions. However, if $\mathbf{X}(t)$ and $\mathbf{Y}(t)$ are different, then $\mathbf{C}_{XY}(t)$ is the cross-correlation function and $\mathbf{g}(t)$ will not have peaks at any of the bit positions. The data signal can then be extracted from $\mathbf{g}(t)$ by means of a thresholding procedure.

DWDM differs from the previous two multiplexing techniques in that it uses multiple wavelengths. Each data signal externally modulates a laser source, which operates at a specific frequency. The modulated signals are then combined for transmission over an optic fiber link. The schematic representation of the process is shown in figure 1.4 below.

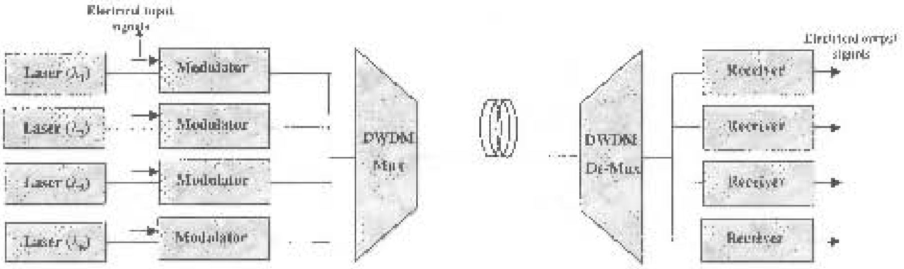


Figure 1.4 Illustration of a DWDM system

Each of these schemes has inherent advantages and disadvantages. In chapter two this topic will be re-visited and it will be shown that with the current device technology, DWDM delivers the largest bandwidth, especially for the core network.

1.4 Optical Components

The success of communications technologies depend heavily on the available device technologies as well as their capabilities and limitations. In this section the limitations that optical components impose on the communications systems are discussed. Thereafter, the basic components that enable optical communications are reviewed and the device specific limitations are discussed.

Bit Error Rate (BER) and Signal-to-Noise Ratio (SNR) are the parameters that are used to evaluate the performance of a communications system. The *BER* is defined as the ratio of the number of bit errors received to the total number of bits transmitted, usually expressed as ten to a negative power. It is an indication of how often a packet or data unit has to be re-transmitted because of errors. The SNR is defined as the ratio of the signal power level to the noise power level, normally expressed in dB. In general, it is desirable to maintain a low BER, which can be ensured if the SNR is maintained at a very high value. However, bit errors are not only caused by a degrading SNR, but could also result from other factors such as loss of synchronization, timing jitter etc. The desired BER and SNR are normally application specific. In the case of loss sensitive traffic such as electronic commerce for example, it would be desirable to maintain a very low BER (10E-11 or lower), whereas for voice traffic a BER of 10E-5 would be acceptable. In analog communications, as is the case with optical communications, SNR is a more meaningful parameter than BER. For these applications it is desirable to maintain high SNRs (at least 30dB), because in these systems, noise is additive. This is because signals are not regenerated at regular intervals.

The major factors that can cause BER degradation are the broadening of transmitted pulses, loss of optical signal power and crosstalk. These factors can in turn lead to SNR degradation. In section 1.1 it has been pointed out that there is an inherent relationship between the frequency, velocity and wavelength of light. This inter-dependency can give rise to the phenomenon of dispersion, which is the most significant factor causing pulse broadening in lightwave systems. Three types of dispersion can be distinguished:

- intermodal dispersion occurs only in multi-mode fibers where each mode propagates at a different velocity along the fiber. As a result, different rays of light from the same source arrive at the other end of the fiber at different times. The effect of this is a pulse that is spread out in the time domain. In this research only single mode fibers are considered and used for simulations;
- chromatic (or material) dispersion results from the inter-dependency between the refractive index of a material and wavelength. If the transmitted light signal consists of more than one wavelength then these different wavelengths will be refracted at different angles and will thus propagate at different velocities along the fiber, resulting in pulse broadening at the receiving end; and
- waveguide dispersion results from the refractive indices as well as other characteristics of the waveguide (such as the shape of the fiber core and cladding). This too leads to the propagation of different wavelengths at different velocities, which in turn leads to pulse broadening.

Significant dispersion can lead to inter-symbol interference which in turn limits the bit spacing and maximum bit rate in communication systems. Another phenomenon, which may lead to BER degradation in lightwave systems, is non-linearities in optic fiber. These fiber non-linearities that occur within a system are dependent on the optical intensity of the signal propagating through the fiber. The major non-linearities that occur are:

- self-phase modulation (SPM) and cross-phase modulation (XPM) are the result of the dependency of the refractive index on the optical intensity of the signal [9]. SPM is the change in phase of the light at a given frequency caused by variations in the signal power at the same frequency, whereas XPM is the change in phase of light caused by a variation in the power of a signal propagation at a different wavelength;
- stimulated raman scattering (SRS) occurs when incident light reacts with the molecular vibrations of the fiber material to create scattered light at longer wavelengths than the incident light. The result is that power is transferred from shorter to longer wavelengths. This transfer of signal power increases rapidly as the power of the incident light increases;
- stimulated Brillouin scattering (SBS) occurs when the incident light reacts with acoustic vibrations in the fiber. SBS differs from SRS in that the wave that results from the transfer of power from shorter to longer frequencies propagates in the opposite direction to that of the incident light; and
- four wave mixing (FWM) occurs when two wavelengths propagating at frequencies f_1 and f_2, mix to cause signals at $2f_1 - f_2$ and $2f_2 - f_1$. These "new" signals, which are called sidebands, can interfere with signals at other frequencies at which data is being transmitted.

These fiber non-linearities may limit the per channel optical power, the number of channels that can be transmitted on a fiber, the maximum bit rate of the system as well as the spacing between the different channels.

Loss of optical signal power can adversely affect the SNR of a system and consequently, the BER. Loss of signal power can result from the insertion loss associated with the optical components, fiber attenuation experienced by the propagating signal as well as the transfer of optical power between wavelengths (as in the case of SRS and SBS). Insertion loss varies from one component to another. The fiber attenuation is normally specified as αdB/km, from which it is clear that signal attenuation increases with increased transmission distance.

Crosstalk is simply the addition of unwanted optical power to a wavelength channel. In general, two types of crosstalk are identified: linear and non-linear. *Linear crosstalk* is further subdivided into [28]:

- *inter-band crosstalk*, which is the addition of optical power from different wavelength channels to a wavelength channel. This kind of crosstalk can easily be reduced by the use of narrow-bandwidth optical filters; and
- *intra-band crosstalk*, which is the addition of residual optical power from the same wavelength channels. This kind of crosstalk cannot be removed because the signal and crosstalk are at the same wavelength.

Both inter- and intra-band crosstalk results from imperfect filtering. *Non-linear crosstalk* arises from fiber non-linearities such as FWM, SRS and SBS. The extent of crosstalk that occurs in a system is dependent on the optical power levels, the wavelength spacing between optical channels and the transmission distance.

A basic optical communication system comprises of a transmitter, optic fiber, optical amplifiers and a receiver. Optic fiber has already been discussed in section 1.2; the other components as well as their limitations are discussed hereafter.

Transmitters

The transmitter comprises of a laser and a modulator. In order to transmit data across an optical fiber the data has to be encoded or modulated onto a laser signal. The preferred modulation technique for optical communications is binary ASK (also known as on-off-keying (OOK)) because it is so simple to implement. The laser can be directly modulated by turning the laser on and off, however, this could result in variations of the laser amplitude and frequency. In high bit rate systems, the laser is typically externally modulated, for example by using a Mach-Zehnder interferometer. The laser light is an important component in lightwave systems since it is this light onto which signals are encoded and which then propagates along the fiber. The characteristics of the laser that may affect system performance are the laser linewidth, frequency stability and the number of longitudinal modes. The laser linewidth refers to the spectral width of the light generated by the laser source. The laser linewidth can affect the amount of dispersion that occurs as the light propagates along a fiber. Ideally a laser with a narrow linewidth should be used for optic fiber systems. Frequency instabilities such as mode hopping, mode shifts and wavelength chirp can result in variations of the laser center frequency. The number of longitudinal modes refers to the number of wavelengths that are amplified by the laser. It has already been pointed out that different modes propagate at different velocities along a fiber, which results in pulse broadening. It is therefore desirable that the transmitter laser only amplifies a single longitudinal mode. Frequency instabilities such as mode hopping, mode shift and wavelength chirp can cause variations in the laser emission frequency. These frequency instabilities can affect the channel spacing and therefore the system capacity. This characteristic affects the placement and spacing of channels, so it affects only WDM systems in which multiple wavelengths are used. The number of longitudinal modes refers to the number of wavelengths amplified by the laser.

Receivers

The receiver comprises of a filter and photo detection device such as a PIN photodiode. In multi-wavelength systems the filter serves to separate out the multiple channels, which are then converted from the optical to electrical domain using the photo-detector. Filters can be either of a tunable or fixed type and the type of device selected is application specific. In a LAN, for example, a tunable filter is desirable to tune to the different frequencies that may have been transmitted. Filters are generally characterized by a transfer function, which determines the shape of the filter in the passband. The performance of filters is limited by the free spectral range (FSR) and finesse of the filter. The FSR is defined as the range after which the transfer function begins to repeat itself. FSR can impose a limitation on the number of channels since all the wavelength channels must fit within one FSR. The finesse of the filter is a measure of the width of the transfer function and normally defined as the ratio of the FSR to the 3-dB channel bandwidth. A high finesse indicates a filter with a narrow transfer function, which means that more channels can fit into one FSR, whereas a low finesse means a broader transfer function and consequently channels have to be spaced further apart to avoid crosstalk and hence signal degradation.

Receivers can employ either direct (non-coherent) or coherent detection. Direct detection is easy to implement. The incoming photonic stream is converted to a stream of electrons, which is then passed through a threshold device to determine whether the bit is a logical "1" or "0". Coherent detection uses the phase information of the signal for encoding and detection. It has already been pointed out that fiber non-linearities cause variations in the phase of a transmitted signal, thus limiting the performance of these kinds of detection devices.

Optical Amplifiers

In general, the optical amplifier (OA) takes as its input a wavelength multiplexed signal and increases the optical power level without affecting any of the other characteristics not related to the signal power level. Optical amplification is done entirely in the optical domain. It is possible that optical noise is added to the signal during the amplification process.

There are three types of amplification mechanisms that can be distinguished. The first type is called "1R (regeneration) or all-optical amplification" [9]; it acts only to boost the signal power, not to restore the shape and timing of the signal. It provides total data transparency i.e. the amplification is independent of the signal's modulation format, which is ideally what is required in all-optical networks. Today's digital networks (SDH) uses either 2R (regeneration and reshaping) or 3R (regeneration, reshaping and re-clocking). In both 2R and 3R techniques optical-to-electronic-to-optical conversion occurs, which also means that in the case of a wavelength multiplexed signal, the signal has to be demultiplexed and then later re-multiplexed. It therefore follows that 2R and 3R amplification is less transparent than 1R and thus not suitable for truly all-optical transmission.

The characteristics of the amplifier, which can impose a limitation on its performance, are the amplifier gain (efficiency), gain bandwidth, gain saturation and amplifier noise. The amplifier gain is simply a ratio of the output to input signal power and sometimes a more useful characteristic is the gain efficiency which measures the gain as a function of the input power in dB/mW. The gain bandwidth refers to the frequency range over which the amplifier operates. This is an important characteristic, particularly in multi-wavelength systems, since it limits the number of wavelength channels available for a given channel spacing. The gain saturation point refers to the value at which the output power no longer increases as the input power increases. It is defined as the output power at which there is a 3-dB reduction in the ratio of the output to input signal power. This parameter too is significant in multi-wavelength systems, since all wavelengths are not amplified equally. Those wavelength channels that are then amplified more than others can drive the amplifier into saturation. Finally, the amplifier noise, which in the case of optical amplification is dominated by amplified spontaneous emission (ASE). This parameter becomes significant in long-haul communication systems where 1R amplification is implemented and can lead to rapid degradation of the BER to unacceptable levels.

The extent to which each of these limitations affect performance varies with the system that is implemented. In chapter two the performance of OTDM, OCDM and DWDM systems will be compared in the context of these limitations.

1.5 Optical Switching Methods

The advent of optical communications has made it possible to transmit data in excess of 1 Tbps with current transmission gear. Electronic switching gear, however, cannot yet handle these high capacities. Consequently new switching methods and procedures are being investigated to alleviate the bottlenecks which are developing at electronic switching nodes.

In general, there are two types of switching paradigms which can be used to implement optical domain switching: optical circuit switching and optical cell (or packet) switching. Wavelength routing is a form of optical circuit switching. With this scheme a lightpath (on a dedicated wavelength) has to be set up on each physical link between the source and destination node before data can be transmitted. The lightpath is torn down when the transmission process has been completed. The drawback of this scheme is that it does not utilise the available bandwidth efficiently, especially in the case of bursty traffic, which is predominant in current communications networks. Furthermore, because of the limited number of bandwidths available it may not always be possible to set up a lightpath, which could lead to delays in routing of data from the source to the destination.

An alternative to circuit switching is packet switching. ATM, IP and Frame Relay are examples of layer 2 and 3 packet switching protocols. In packet switching the data payload is divided into packets, to which a header is appended. Information contained in the header is used to route the packet to its destination. Multi Protocol Label Switching (MPLS) is a packet switching mechanism that has been proposed to implement high-speed hardware switching in IP networks. MPLS is an adaptation and extension of the IP routing protocol. In the MPLS framework the hop-by-hop routing paradigm is replaced with a label-swapping paradigm, with a view to simplifying the routing engine [43]. An MPLS network is made up of Label Edge Routers (LERs), which form the ingress and egress nodes, and Label Switching Routers (LSRs), which form the core of the network. The function of the LER at the ingress node is to replace the header with an MPLS label. The label assignment is based on the forward equivalency class (FEC) to which the packet is allocated. The packet is then routed towards the egress node via the core LSRs and a label switched path. The latter is simply a concatenation of labels that form the path through the MPLS network. The core LSRs route the packet to the next hop based on the label and FEC to which it belongs. The LER at the egress node removes the label before forwarding the packet to the destination address in the access network. The advantage of MPLS is that the forwarding procedure at each intermediate node is simplified and is replaced by a label-swapping procedure which requires less processing overhead. Furthermore, MPLS can also support constraint based label switched paths (LSPs) from edge to edge, which will in turn enable load balancing, QoS provisioning and Virtual Private Networks (VPNs).

Optical Burst Switching (OBS) has been proposed as a possible approach for implementing a switched optical layer. OBS is essentially a packet switching paradigm, but it differs from traditional packet switching protocols in that the control packet (header) is de-coupled from the payload in the time domain [45]. In OBS, the control packet is sent first to set up the connection i.e. to reserve the appropriate amount of resources and configure the optical cross-connects along the path. The data packet is launched from the source after some time interval (referred to as the *offset time*), without waiting for an acknowledgement of connection establishment. This means that OBS implements a one-way reservation protocol similar to tell-and-go (TAG) and ATM's fast reservation protocol (FRP). The major design issue associated with OBS is the calculation of the *offset time*. The *offset time* should be calculated in such a manner so that it is large enough to allow for the processing of the control packet at the intermediate optical cross connects (OXCs) with the need for no or minimum buffering. In [45], Qiao et al. proposes a fixed *offset time* setting scheme called JET (*just-enough-time*). The scheme uses delayed reservation (DR) as a stand-alone feature or in conjunction with fiber delay line-based buffered burst multiplexers (BBMs) for efficient bandwidth utilization. Verma et al. [44] argue that the JET protocol is "not robust in distributed environments due to contention that may arise at intermediate nodes among bursts from geographically dispersed sources". As an alternative, [44] proposes a scheme in which the *offset time* generation process is randomised (similar to ATM's leaky bucket regulator). With this scheme it was shown that with a target blocking probability of 10^{-4} a burst switched DWDM network can be operated at 60% utilisation if 64 channels are supported. No buffering was implemented in the model.

Both MPLS as well as OBS are relatively new methodologies and much research effort is still required before successful commercial deployment.

1.6 Optical Buffering

In packet switches, the possibility exists that two or more packets may simultaneously access the same output port of a switch, thus creating a contention. In electronic switches a contention is resolved by a store-and-forward technique, in which packets in contention are stored in a queue and then forwarded to the desired output port one-by-one. Buffering (i.e. storing of packets) is one of the four fundamental tasks incorporated in electronic switch design; optical switch design is no exception. However, the implementation of an effective buffering strategy directly in the optical domain has been hampered, since optical random access memory (RAM) has not yet been realised.

A proposed technique for implementing optical buffering is by means of fiber delay lines (FDLs), which are simply fixed length fibers. The fundamental difficulty with this approach is that variable length buffers have to be implemented, but by their very nature FDLs implement fixed delays. Once a packet has entered an FDL it emerges after a fixed length of time and there is no means to remove the packet from the FDL before that time. The major limitation of FDLs is the optical splitting loss that the packet experiences when it is forwarded to the delay line for buffering.

Various switching architectures have been proposed that incorporate optical buffering using FDLs [46]. In these studies the focus was on the location of the buffers (i.e. input or output buffers) and the depth of the optical buffers. The switching paradigm discussed above i.e. OBS, has added another requirement in the design of FDL buffers: whether FDLs should implement equally spaced length delays. In [47], Callegati et al. argues that small spaced delays can lead to poor buffering capacity with poor time resolution. As an alternative, [47] proposes a scheme where the dimensioning of the delay is considered as part of the whole switch-dimensioning problem.

Optical switching and optical buffering are closely related issues and much research is still required in this area before the implementation of optimal switching techniques at the optical layer.

1.7 Management of Optical Networks

Optical communications has become the enabling technology to increase data transmission capacities in excess 1 Tbps. With such high traffic volumes at risk, NM is becoming an increasingly important feature of these high capacity networks. In [18], network management (NM) is defined as the process that "supports the activities of the network operators for the operation, administration, provisioning and maintenance of their networks". The NM function traditionally encompasses five functional areas (fault, configuration, accounting, performance and security management – also referred to as FCAPS) as specified by the

International Telecommunications Union – Telecommunications Standardisation Sector (ITU-T) in [32]. The NM process can only be achieved if it is possible to measure the validity, integrity and quality of the data being transmitted and if there is a means for transmitting this information to the network management system.

OAM is the collection of processes that support the NM function. The overall OAM actions include measuring the signal and NE performance, reporting on the degradation or failure of the signals or NEs and altering the configuration of NEs. In the case of degradation or failure, reporting is done when the event occurs, whereas the other OAM actions are performed on request. In digital networks, OAM procedures collect or derive data about the quality of signals and the state of the network equipment. It then transports this data to the NM system in the form of OAM-signaling in the transmission overhead. The network management system (NMS) is itself a software application that resides in the application layer of the Open Sytems Interconnect (OSI) protocol stack. However, it monitors, controls and manages the network resources, both software and hardware, that reside at the 4 lower layers of the stack. In the case of optical networks the NMS manages optical network elements (ONEs) and the optical signal i.e. it manages the optical layer.

In general, network management systems are based on the manager-agent model. A manager is defined as an entity that acts in a managing role. The role of the manager is to collect data from the agent. Data collection can be performed either synchronously or on request. An agent is defined as an entity that acts in a managed role. The agent collects and stores data that could be requested by the managing device, for use by the management system. The agent sends data to the manager via notifications.

The two network management paradigms that have evolved over the years were intended mainly for the networks for which they were developed. TMN was standardized by the ITU-T [31,32] and was intended for the management of telecom-based networks. In this approach, individual NEs have no knowledge of the network topology as a whole, their focus is mainly on their own operation. The management system is a hierarchy, with the lowest level of managers being the network element managers (referred to as the element management systems – EMS). The latter too does not have a global view of the network topology; in fact, only the NMS has a global view of the entire network. So, in TMN, more emphasis is placed on management, with the extent of the control plane being minimized as much as possible. This paradigm is illustrated in figure 1.5 (*i*), below.

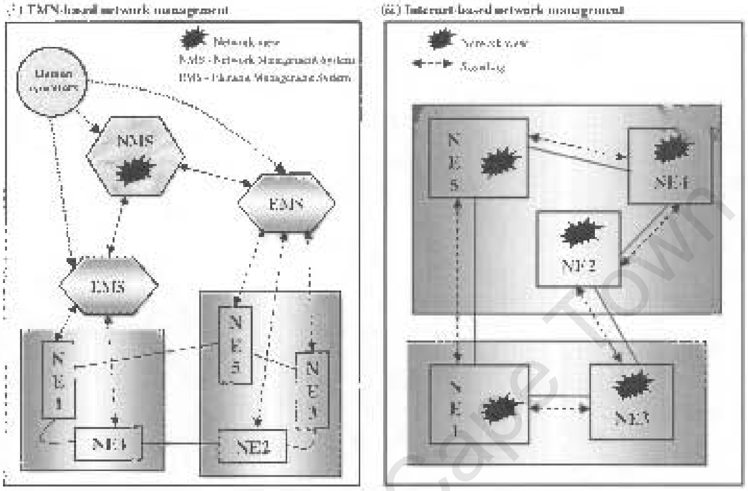


Figure 1.5 Illustration of the difference between the TMN- and Internet-based management paradigm

Internet-style management and control on the other hand, was standardized by the EITF [33,34,35] and was intended for the management of TCP/IP-based networks. This management style also has a hierarchical structure. The difference though, is that the NEs have a global view of the network topology. This is because each node runs a set of topology discovery and routing protocols (e.g. Open Shortest Path First - OSPF, Border Gateway Protocol – BGP), which discover its neighbours and the rest of the domain. So, the internet-based network management and control approach utilises a more extensive control plane, with less emphasis on management. This management paradigm is illustrated in figure 1.5 (*ii*) above.

1.8 Thesis Objectives

IP-based services are becoming the dominant transmission technology and this trend is expected to continue as the multimedia age comes to the fore. The bandwidth requirements for these services change frequently and new managed objects (MOs) will be added to the network as lightpaths are set-up and torn-down to accommodate these changing demands. The NM system (NMS), operating systems (OSs) and managing elements must be able to cope with these changing conditions within the network, without overloading its own resources. The major objective of this thesis is to propose a generic OAM model for the optical layer, which takes these changing networking conditions into consideration.

This study therefore investigates the implementation of a hierarchical OAM model architecture and examines its performance in terms of the notification load which each managing element or OS has to handle. The study focuses on the performance monitoring and fault detection functionality which forms part of the OAM procedures. It shows by means of simple networking examples how this model can be applied and then emulates those mechanisms in a simple NM tool. The study also proposes a framework for a core DWDM-based all optical network in the South African context.

1.9 Thesis Outline

The study of OAM procedures for the optical network layer requires an in-depth understanding of the functional architecture of the optical layer and also the entities that have to be managed. Knowledge of the optical layer and the optical networking entities will enhance the understanding of how the optical network elements (ONE) interact with one another and also how the layer networks will interact with each other. This will also help the understanding as to how, why and where the identified optical layer performance parameters are to be monitored.

Chapter two presents an overview into optical networking concepts. The chapter begins with a discussion of the optical multiplexing techniques that were introduced in chapter one and shows that with the available device technology, DWDM provides the best solution for tapping into the huge fiber bandwidth of optic fiber. Thereafter, it re-visits the subject of the spectrum of optic fiber and reviews the advances in technology that have resulted in increasing the available fiber optic bandwidth.

The ITU-T has recommended a protocol layer structure for the implementation of the DWDM optical layer; the layer structure is reviewed here. Thereafter, some optical networking components are discussed. The chapter concludes with a review of networking architectures for the implementation of core networks.

Chapter three presents a discussion on operation, administration and maintenance in optical networks. It begins with a discussion on the issues relating to protection and restoration i.e. survivability in optical networks. These issues are all related to NM and are presented here for the sake of completeness. A generic OAM model is then proposed and demonstrated with the aid of two simple network examples. Thereafter, the optical layer performance parameters are identified. These parameters are then used to demonstrate optical layer performance monitoring and fault detection.

Chapter four proposes a network structure to implement an AON national network in South Africa and shows by way of example how the existing network can be adapted to the proposed structure. It then gives a simple explanation of the relation between the number of wavelengths and number of nodes and demonstrates by means of a simple example how wavelength assignment can be implemented in a four-node ring network. Thereafter, it demonstrates how the proposed OAM model can be applied to the national network.

Chapter five presents the design, implementation and results of simulation of the network example illustrated in chapter three. It also presents the results of the OAM model testing. The chapter begins with an introduction to OptSim as well as a discussion of the simulation methodology used to implement the OptSim simulation environment and an evaluation of the accuracy of the simulation results generated by the simulator. It then discusses the design and implementation of the example network and reviews the simulation results. Thereafter it discusses the design and implementation of the NM tool, which implements the performance monitoring and fault detection techniques within the context of the proposed

OAM framework. The chapter concludes with a discussion and analysis of the results of this study.

Chapter six presents conclusions of this study as well as recommendations for future work in this area.

Chapter 2

All Optical Networking Concepts

All optical communications is emerging as a promising technology for implementing ultra high-speed networks, which can support a wide variety of high bandwidth services. In chapter one a broad overview of the fundamental networking concepts were presented. This chapter takes a closer look at optical networking concepts and serves as a basis for the investigation of performance monitoring and fault detection procedures, which will be discussed in chapter three. The first two sections re-visit subjects that were introduced in chapter one. Firstly, the advantages and disadvantages of the various optical multiplexing techniques are discussed with the view to showing that with the available device technology, DWDM currently presents the best solution to expanding bandwidth capacity. Thereafter, a more detailed analysis of the optical fiber spectrum follows.

The optical layer protocol structure, which has been proposed by the ITU-T, is reviewed hereafter and then some optical networking components and architectures are discussed. A thorough understanding of the components that make up the optical network is required in order to identify the performance parameters that have to be monitored at different points in the network. Furthermore, an understanding of the networking architectures are required, to fully understand the concepts of protection and restoration i.e. survivability that are discussed in chapter three.

2.1 Optical Multiplexing Techniques

Three optical multiplexing techniques (OTDM, OCDM and DWDM) have been identified in chapter one as possible candidates for accessing the huge fiber bandwidth capacity of optical fiber. All of these techniques have inherent advantages and disadvantages that will be discussed in this section.

OTDM is a method of carrying very high bit-rate data streams across a network by time interleaving ultra-short optical pulses onto which data is encoded. Laboratory systems operating at speeds of 100Gbps have already been demonstrated [52]. The major advantage of these systems is that they only require a single 1nm-bandwidth wavelength channel to transmit the entire OTDM data stream. Furthermore, these systems only require a single wavelength source and therefore do not place stringent requirements on the control of the filter and transmitter wavelengths. A further advantage is that OTDM also "lends itself naturally to optical digital processing techniques such as 3R regeneration and serial processing" [52].

The major disadvantage of OTDM systems is that the transmission of ultra-short pulses is limited to very short distances. Single-mode step-index fiber is commonly used and has been widely deployed by operators globally. Optical transmission centered on the 1.55μm band (the region in which EDFAs can be used) has been the focus of research attention. However, at this wavelength, ultra-short pulses launched into the step-index fiber suffer relatively large pulse broadening (as a result of group-velocity dispersion) as they propagate along the fiber. The effect of this is that the transmission distance of the pulses is severely limited (e.g. 40Gbps transmission at 1.55μm wavelength is limited to 5km), unless dispersion compensation techniques are employed [52]. The use of dispersion compensating fiber [] and soliton pulses [] has been proposed as possible solutions to this problem. Another disadvantage of OTDM systems is that the fiber delay lines (FDLs) allocate fixed time slots to transmitters, which means that this technique can not take advantage of statistical multiplexing gain, which may be significant in the case of bursty data traffic. Further disadvantages of this scheme include the stringent synchronization requirements as well as the complexity of the active de-multiplexers. The OTDM technique is still very much in the research phase and many of the components that are required to implement this technology, is still confined to the research labs.

OCDM is a technique, which uses optical orthogonal codes to access more of the available fiber capacity. As with OTDM systems, the major advantage of this approach is that it requires only one wavelength channel and therefore only one laser source. In [50], the author shows that the auto- and cross-correlation functions are insensitive to a wavelength change in the light source over a wide spectral range of 1.5nm. This means that the OCDM system does not place stringent requirements on the wavelength stability of the laser source. In contrast to OTDM systems, channels on different codes can share the same timeslot and therefore the system has no requirements in terms of synchronization. Another advantage of OCDM systems is that because it does not require dedicated time or wavelength slots, it can achieve high statistical multiplexing gain.

In order to obtain good auto- and cross- correlation properties, very long signature sequences are required. This places very strict requirements on the speed of the encoding and correlation hardware, which is the major disadvantage of OCDM systems [49]. The transmission capacity of OCDM systems are also limited by the number of unique signature codes available, which is in turn limited by the length of the codes. OCDM systems also limit the full transparency of the optical communication system because of the opto-electronic conversion, which forms an integral part of the optical conversion code procedure. The best performance of OCDM systems reported to date is 10Gbps [51], though it should be noted that most research effort has been on the hybrid approach of OCDM/DWDM.

DWDM is a technique, which uses multiple wavelengths to access the available capacity of optic fiber. The major advantage of DWDM is that it offers network operators the potential to increase the capacity of existing fiber links by up to three orders of magnitude; DWDM systems operating in excess of 1Tbps are already available. Furthermore, each wavelength channel acts as a virtual fiber, which can carry its own individual bit rate and data protocol. This is a very important attribute because it means that DWDM can support (simultaneously), various protocols operating at different bit rates i.e. it is protocol and bit rate transparent. Another advantage of DWDM systems is that the cost of manufacturing the optical components are also decreasing because components can now be monolithically integrated on semi-conductor substrates. DWDM technology also offers an advantage in terms of network survivability because multiple wavelengths can be simultaneously protected at the DWDM optical layer.

The technique however suffers from inherent disadvantages. Firstly, there is a limitation on the number of wavelength channels that can be employed in the system. This number is limited by the crosstalk requirements as well as the gain flatness of the fiber amplifiers. It has already been pointed out in chapter one that crosstalk limits the spacing between wavelength channels and thus affects the number of channels in the system. With regards to amplification, the EDFA (which is most widely used) has a 3dB-gain variation in its passband. This is acceptable if only one amplifier is used in the network. However, DWDM technology has mostly been deployed in long-haul communications networks, which requires the EDFAs to be cascaded. This cascading of amplifiers could result in a gain filtering effect, which could then reduce the available flat gain amplifier bandwidth.

The second major disadvantage of DWDM systems is the high power levels in long-haul systems, which are necessary to achieve longer transmission distances. Typically, 1mW of optical power is injected into the fiber per wavelength channel, so in a multi-wavelength system it means that several milliwatts are injected into the same fiber. Such high power levels give rise to diverse fiber non-linear effects such as SBS, SRS and FWM as have already been pointed out in chapter one. Other disadvantages of these systems are that it requires complex control schemes for wavelength stabilization and it places stringent requirements on laser sources. Though these disadvantages are of concern, DWDM research has already reached an advanced stage and the components required for implementing Tbps DWDM systems are already commercially available, which is not the case for OTDM and

OCDM. So, currently DWDM is the multiplexing technique of choice for implementing high-speed optical networks.

2.2 Optical Fiber and Spectrum

In the previous section, various multiplexing techniques were discussed that enable more of the available bandwidth of optic fiber to be utilized. Traditionally, this "available" bandwidth was found in the low attenuation regions in the 1.3- and 1.55μm band, as pointed out in chapter one. The 1.4μm band was "unavailable" because of the high attenuation (1dB/km or higher) over much of this spectral window; this is indicated by the peak in figure 1.1. The high attenuation is the result of the high hydroxyl ion (OH^-) impurities, i.e. water molecules that have been retained in the fiber. Recent advances in technology have resulted in a new optic fiber manufacturing process, which virtually eliminate water molecules in the glass fiber [56]. The result is a 100nm increase in the available bandwidth of optic fiber, which is manufactured using this new process.

2.3 Optical Layer Protocol Structure

It is commonplace for communications networks to be described in terms of a layered architecture (e.g. OSI or B-ISDN reference models). Optical networks are therefore also described using this layered approach and the ITU-T has recommended an architecture for the optical transport network (OTN) in G.872. This recommendation is only applicable to wavelength division multiplexed networks. This section reviews the recommended layered architecture, details of which can be found in [6]; it also includes material from [5].

The OTN comprises three layer networks: the optical channel, optical multiplex section and optical transmission section, as illustrated in figure 2.1. A client/server relationship exists between adjacent layers; each layer serves the layer immediately above it and acts as a client to the layer immediately below it. There are two classes of functions that occur within each layer network:

- the *adaptation function*, represented by the trapezium in figure 2.1, is responsible for adapting the received client signal (e.g. SDH or ATM signal) and performing the specific role of that layer network (e.g. amplification, multiplexing).
- the *termination function*, represented by the triangle, is responsible for guarding the "integrity of the signal through the underlying layers by using the overhead information"[5]. The overhead information is required for network management and control.

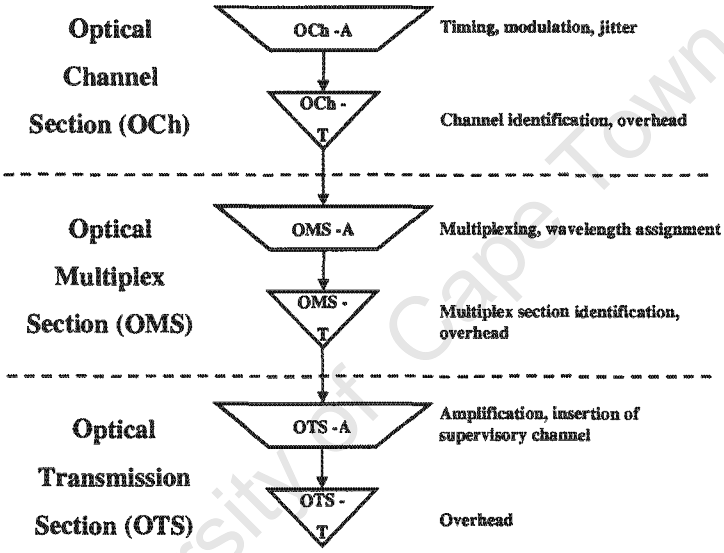


Figure 2.1 ITU-T recommended optical layer network protocol structure

The ***Optical Channel (OCh) Layer Network*** provides for the end-to-end networking of optical channels for transparently conveying client information of varying formats (e.g. SDH, PDH and ATM). The characteristics of this layer network include transparency, maximum number of cascaded optical nodes, optical cross-connect capability and optical protection. The ***Optical Multiplex Section (OMS) Layer Network*** provides the functionality for networking of a multi-wavelength signal, which also includes the case for just one wavelength. The characteristic of this layer network includes the number of available wavelengths, wavelength conversion and also optical protection. The ***Optical Transmission Section (OTS) Layer Network*** provides the functionality for the transmission of the optical signals over the optical media of various types. This layer network is characterized by the power budget, dispersion budget, accumulated noise, crosstalk, number of cascaded optical amplifiers.

2.4 Optical Networking Components

The widespread success of DWDM can be attributed directly to the optical device technologies that have overcome some of the inherent limitations associated with optics. Three important DWDM networking components, optical amplifiers, optical cross-connects and optical add-drop multiplexers that have (and will) impact heavily on the continued success of optical networking, are discussed in this section.

2.4.1 Optical Amplifier (OA)

Optical amplification is an optical transmission layer network function. The optical amplifier is responsible for amplifying the different wavelength channels simultaneously, as pointed out in chapter one. The most commonly used OA is the erbium doped fiber amplifier (EDFA), which implements 1R amplification. EDFA amplification was traditionally most efficient in the C-band (1.53 –1.57μm), but advances in OA technology has increased the gain bandwidth of EDFAs to the L-band (1.57-1.62μm) and the S-band (1.45-1.53μm) [56].

In chapter one it was also pointed out that there are three amplification techniques, namely 1R, 2R and 3R, which exist. The performance of 1R and 2R amplification was investigated in the OptSim simulation environment using a simple point-to-point network in which a 10Gbps signal was transmitted over five 50km spans of single mode fiber. The objective was to compare that the performance of a signal, which undergoes these two amplification techniques. A block diagram of the 1R and 2R network implementations are shown in figure 2.2 (*i*) and (*ii*), respectively. The difference between the two network implementations is that in the case of 2R amplification, the signal is converted from the optical to the electrical domain after each 50km span. The signal is then regenerated and converted back to the optical domain before transmission over the next span. After each span the power level as well as the electrical domain parameters are measured. It should be noted that the optical-to-electrical-to-optical (O-E-O) conversion in the 1R network implementation is required for measurement purposes only.

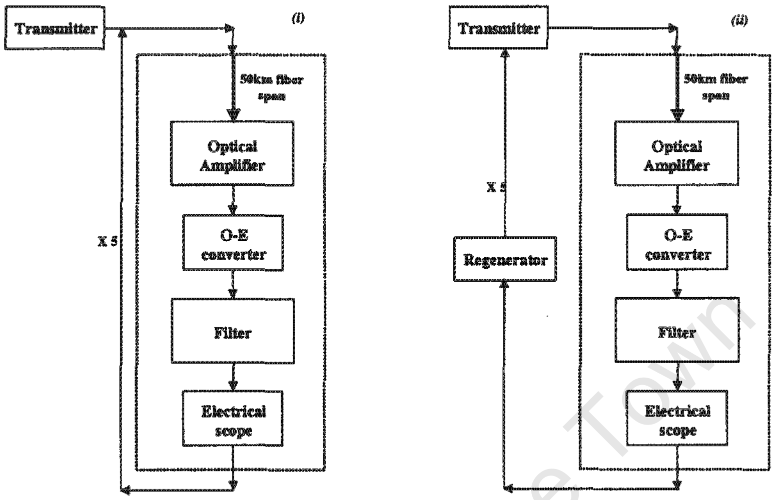


Figure 2.2 Block diagram of (*i*) 1R and (*ii*) 2R amplification techniques

The simulation results obtained indicate that for 1R amplification the signal degrades rapidly after transmission over 100km. In fact, after transmission over the entire range the eye is almost completely closed and the BER has degraded from the order 10E-12 to 10E-2 (see figure 2.3 below). In the case of 2R amplification, after transmission over 250km, the BER has degraded by only four orders of magnitude from10E-12 to 10E-7 (see figure 2.4 below).

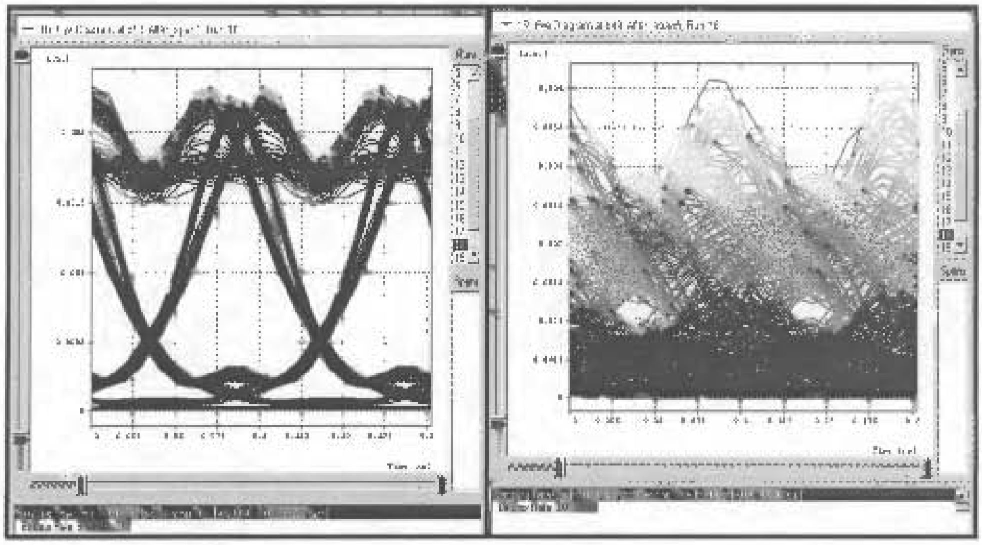


Figure 2.3 Comparative eye diagrams for 1R amplification after 50- and 250km

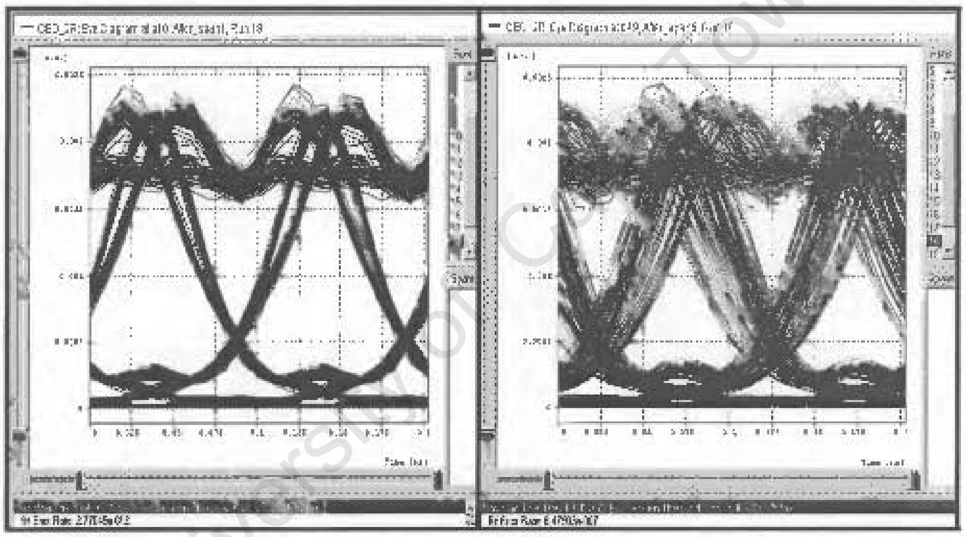


Figure 2.4 Comparative eye diagrams for 2R amplification after 50- and 250km

In the case of signal power levels as well as OSNR, there is no significant difference between the result obtained for 1R and 2R amplification. However, the results obtained for BER performance clearly suggests that 2R amplification provide for better signal performance over a longer transmission range.

2.4.2 Optical Add/Drop Multiplexer (OADM)

The OADM is a device that implements functions from all three of the layer networks. It allows for an optical signal at a specific wavelength (λ_j) to be the extracted or inserted at an intermediate site, without affecting the other wavelength channels (referred to as express channels). The basic node architecture for the OADM node is illustrated in figure 2.5 below. In general, pre- and post-amplifiers (which are not shown in the figure) are included as part of the OADM implementation and EDFAs were typically used for this. However, recent research has indicated that Raman pre-amplifiers in combination with EDFA post-amplifiers can extend the distance that the signal travels by "doing the amplification where the signal is already fairly weak, rather than counting on high amplifier powers to carry the signal all the way to the receiver" [56]. The result of this is that the spacing between amplifier stations are increased for a given or greater bit rate for existing installations. Furthermore, impairments are decreased because the maximum power level along the link can be lower. The (de-) multiplexer are implemented using arrayed waveguide gratings (AWGs). Typically the add and drop channels have the same wavelength.

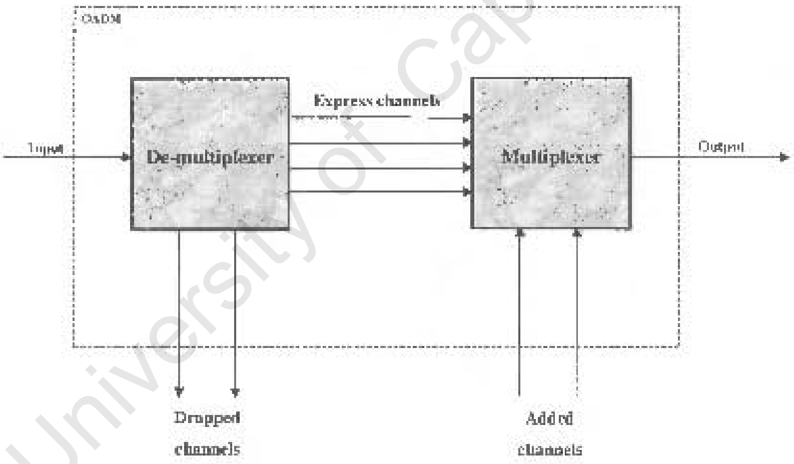


Figure 2.5 Illustration of optical add drop multiplexing

An alternative implementation of the OADM is to use filters to drop the required channel(s) and a combiner to add channel(s). The problem with this latter approach is the inadequate filtering could result in significant in-band crosstalk between the added and dropped channels.

The operation of the OADM and the effect of crosstalk in OADMs were investigated in the OptSim simulation environment. A fixed (i.e. non-configurable) eight-channel OADM, capable of adding/dropping one wavelength channel (channel one in this example) was implemented. Each channel operated at 10Gbps. No pre- and post-amplifiers were included in the model. The optical spectra observed at the input and output of the OADM is shown in figure 2.6 below.

Monte Carlo simulations were run for different values of crosstalk from –90 through –0.01dB. The observed signal power level and OSNR of all the channels, excluding the added (dropped) channel, remained relatively constant over the entire range. However, for the add (drop) channel, the signal power level at the output of the OADM increased by approximately 1dBm over the entire range the most significant change occurring from around a crosstalk value of –10dB. This power level increase was accompanied by degradation of the OSNR by approximately 0.5dB, which suggests that the increased power was in the form of noise.

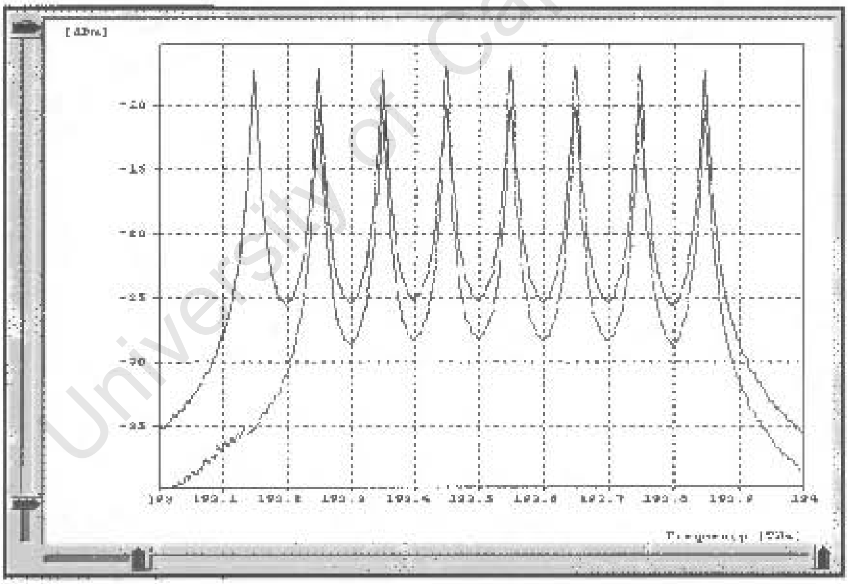


Figure 2.6 Optical frequency spectra at the input and output of the OADM

For the BER, the pattern was fairly similar for the express channels. For the add (drop) channel however, the BER performance was constant at 1E-40 for crosstalk values of –90 through-20dB. However, below this value the BER performance degrades sharply from 1E-40 at –19dB to 1.64E-11 at –14dB. These results suggest that there is indeed a correlation between crosstalk and the performance of the add (drop) channels of the OADM.

2.2.2 Optical Cross-Connect (OXC)

The cross-connect also implements functions from all three of the layer networks. An OXC is nothing more than a switching element that also has add (drop) and multiplexing (de-multiplexing) capability, as illustrated in figure 2.7 below. As with the OADM, which was discussed above, the OXC node also typically includes pre- and post-amplifiers (these are not shown in the illustration below).

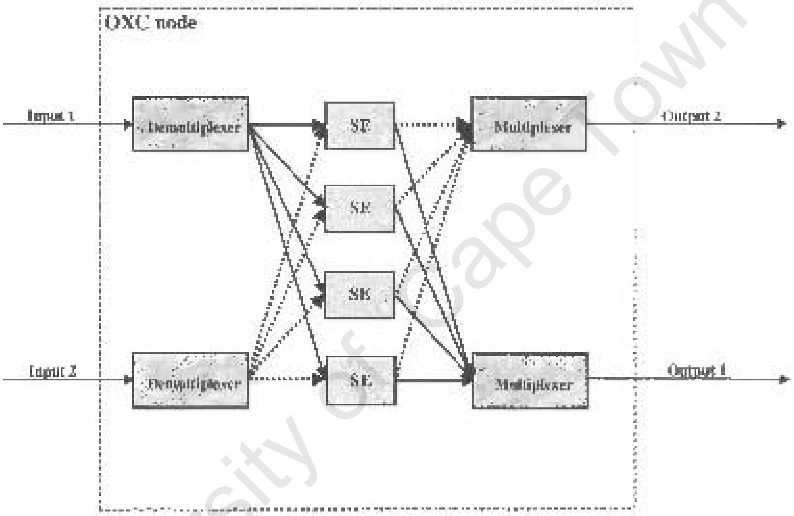


Figure 2.7 Illustration of optical cross connecting

In this configuration the optical switching elements have to perform high-isolation switching in order to limit the amount in-band crosstalk, which could result in BER degradation of the added and/or dropped channel [57].

The operation of the OXC and the crosstalk performance of the SEs were investigated in the OptSim simulation environment. The OXC takes at its input two 4-wavelength channel multiplexed signals, which are then de-multiplexed before being routed through the switching fabric. Each channel operates at 10Gbps and the channels are spaced 100 GHz apart. In this implementation the multiplex section at input port two is attenuated by 3dB. The reason for this is just to be able to distinguish between the switched channels at the output port, when the spectra of the input and output channels are compared. After switching, the signals are multiplexed together before being presented at the output port of the OXC.

The optical spectra observed at the input and output of the OXC are shown in figure 2.11 (*i*) and (*ii*), respectively. Note that there is a difference in the power levels of the input and output signals. This is because the power level of the signal at first input port is attenuated by 3dB, which makes it easier to differentiate the signals at the output ports.

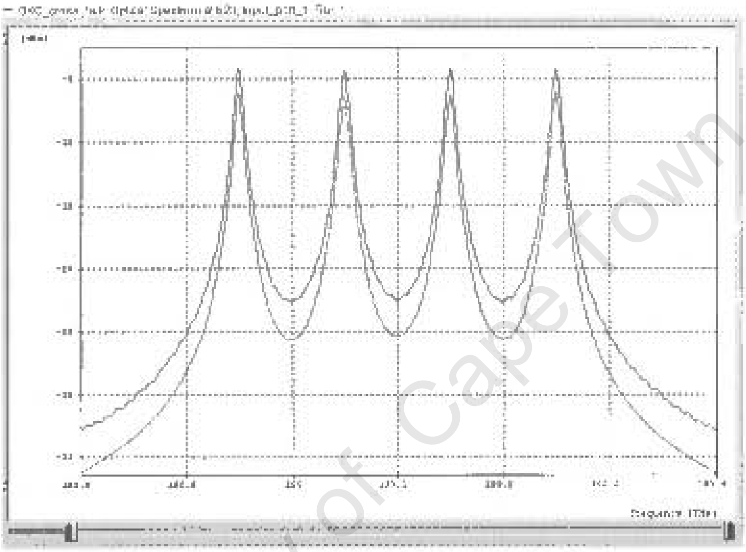


Figure 2.8 Spectra at input port 1 and output port 2 of the OXC

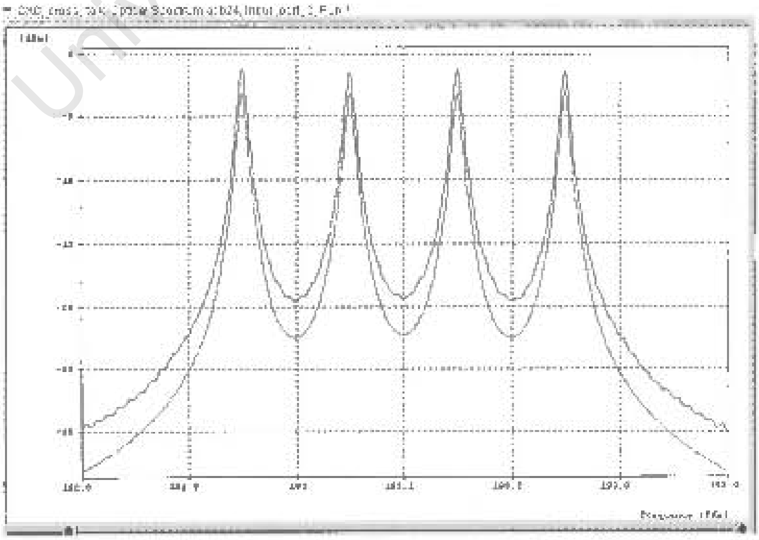


Figure 2.9 Spectra at input port 2 and output port 1 of the OXC

Monte Carlo simulations were run for different values of crosstalk from –90 through –1dB. It was observed that the crosstalk value does indeed affect the power levels as well as the OSNR. At the first output port in some instances the power level increased by as much as 6dBm with an associated degradation in the OSNR of approximately 2dB. At the other output port the maximum change in power was approximately 4dBm, with the OSNR degrading by approximately 1dB. Overall, the power level and OSNR degrades

2.5 Networking Architectures

The efficiency of a communications network depends on the design and architecture used to implement the network. In general, there are five architectures, which can be used to implement communications networks: point-to-point, common bus, star, ring and mesh.

The point-to-point is commonly used in access networks. The point-to-point topology is the simplest topology to implement. It provides a link between two nodes. This topology has been largely used recently with the first of DWDM deployment to upgrade existing fiber links. The common bus topology is largely used to implement LAN networks. In this architecture all nodes are connected to a common bus structure. The IEEE 802.4 Token Bus, IEEE 802.3 Ethernet and IEEE 802.6 Distributed Queue Dual Bus (DQDB) all implement this topology.

In the star topology all nodes are connected to a central node. Nodes communicate with each other through the central node. This architecture too is more common in LAN networks, though it was previously widely used in public switched telephone networks (PSTNs). The major advantage of this architecture is that for a given number of nodes the star topology requires the least number of links (when compared to the ring and mesh topologies which are discussed hereafter). For an N-node star network requires

$$N-1 \tag{2.1}$$

links. So for example, in a 5-node network as depicted in figure 2.10(*i*), four links are required. This means that the star topology minimizes the link installation cost. The architecture, however, also has many disadvantages. Firstly, all the nodes communicate via the central node, which places a higher capacity demand on this node to handle the transit traffic. Furthermore, if the central node fails then the whole system is brought down as no communication can be established. In order to provide network resilience a backup central node is required and this is undesirable because of the high cost implication. For the implementation of a core network, this architecture certainly does not provide an adequate solution.

In the ring architecture (illustrated in figure 2.10(*ii*)) the nodes are connected so that they form a closed loop. To effect communication in this architecture all nodes have to communicate through their adjacent neighbours. Consequently, there is a correlation between the number of nodes in the ring and the link capacity. The required link capacity increases as the number of nodes increase. Furthermore, the traffic flow on the link increases as more nodes communicate, which then also means higher throughput. However, from queuing theory it is known that as throughput increases the network utilisation increases, which in turn leads to an increase in network delay as the load approaches the total capacity of the link. Clearly there is a trade-off between throughput and network delay. So to maximise throughput and minimise delay the network designer has to limit the number of nodes that the ring contains.

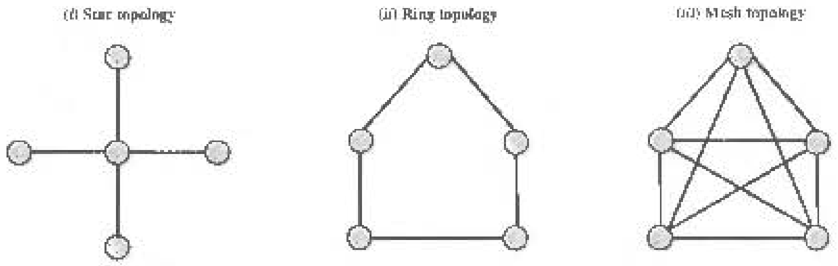


Figure 2.10 Illustration of topologies available for network implementation

For a given number of nodes, the ring architecture requires more links that the star network but less than the mesh (as will be shown later). An N-node ring network only requires N links. So in terms of link installation cost, the ring structure is more advantageous than the mesh. The IEEE 802.5 Token Ring, FDDI and SDH use this topology. In fact, the SDH ring structure has become very popular because of the high network resilience that its self-healing ring structure (SHR) offers.

In the mesh architecture (illustrated in figure 2.10(*iii*)) each node is connected to every other node (a full mesh). The advantage of this topology is that it provides excellent node-to-node connectivity, which means that nodes have the option of routing traffic along the shortest path. Furthermore, there are many alternate routes along which traffic can be forwarded thus providing for good network resilience without the need for redundant fibers. When compared to the ring architecture, the capacity required for each link is less for the same network throughput. Also, because in this architecture the shortest path is chosen, the delay experienced by the traffic is less, thus increasing the network performance. The major disadvantage of this architecture is the large number of links that it requires. A full mesh N-node network requires

$$\frac{N(N-1)}{2} \tag{2.2}$$

links. So, for example, for a 5-node network, ten links will be required. This large link requirement could lead to high network cost a result of the increased cost of installation of the fiber plant. Another disadvantage of the ring architecture is the complex routing algorithms that may be required to find the shortest path.

2.6 Discussions and Summary

In this chapter optical networking concepts were discussed. Firstly, the advantages and disadvantages of the three optical multiplexing techniques (OTDM, OCDM and DWDM) were compared. The major advantage of OTDM and OCDM is that they require only one laser source and therefore do not place stringent requirement on the frequency stability of the source. The major disadvantage of OTDM is the generation of pico-second pulses, which can propagate over longer distances without significant distortion as well as the complexity of the de-multiplexing devices. The performance of OCDM systems on the other hand is limited by the electronic processing speed of the encoders and decoders. DWDM too has many advantages and disadvantages, but systems operating in excess of 1 Tbps have already been demonstrated. So, with the available device technology, DWDM currently offers the best solution for the implementation of high-speed all optical networks.

A short discussion on the spectrum of optical fiber was then presented. It was pointed out that a new fiber manufacturing process has made available an extra 100nm of the optic fiber spectrum, thus eliminating some of the limitations of DWDM.

Thereafter, the optical layer protocol structure proposed by the ITU-T was reviewed. The optical layer is logically divided into three layer networks: the OCh, OMS and OTS layer networks. Each of these layer networks performs a specific function and together they are responsible for processing and transmission the optical signal. This layer structure will be referred to in chapter three when the performance monitoring and fault detection procedures are discussed.

Three optical networking components were then discussed: the OA, OADM and OXC. Optical amplification is an OTS layer network function. It is responsible for amplifying the entire multiplex section simultaneously. The performance of 1R and 2R amplification was investigated using EDFAs. It was shown that 2R performs better than 1R over a transmission distance of 250km. For the OADM and OXC, the effect of crosstalk on the signal quality was investigated. It was found that the signal quality (measured in terms of BER) deteriorated significantly.

Finally, the various networking architectures were discussed. The ring and mesh architectures present the best possible solution for the implementation of the core network. The advantage of the ring structure is that it minimises the fiber plant installation cost and offers good survivability features with its self-healing structure. The disadvantage of this architecture is the network delay introduced as a result of communications having to be done through the nodes adjacent neighbours. The mesh architecture on the other hand offers excellent node-to-node connectivity and the option of performing shortest path routing. However, this architecture requires a large number of links thus making it more expensive to install than the ring structure.

In chapter three, network survivability techniques for these architectures will be discussed. A generic model for optical layer OAM will also be proposed and the OAM functions of performance monitoring and fault detection will be discussed.

Chapter 3

Operation, Administration and Maintenance in All Optical Networks

Network management (NM) is becoming increasingly important as the communications industry is becoming more competitive. Furthermore, the deployment of dense WDM systems has resulted in high traffic volumes (in excess of 1Tbps per fiber is currently possible), being transported and network failure will therefore lead to a considerable loss of data. Network management, which includes functions to detect failures before they occur as well as functions that restore the network after a failure has occurred, has therefore become a desirable and essential feature of these high capacity networks. This chapter takes a closer look at these and the supporting NM functions. The chapter begins with a discussion of network survivability. Survivability techniques can be classified into protection and restoration. Various protection and restoration schemes for the network architectures discussed in chapter two will be presented.

The major objective of this research is to propose a generic OAM model for optical networks. In section two, the proposed model is introduced and discussed. Thereafter, the topics of performance monitoring and fault detection, which are the support function for survivability, are discussed. Performance monitoring is a very important OAM function, which assists the network in detecting faults that could result in network failure. These techniques are demonstrated for the optical layer with the aid of two simple network examples. The discussions are done within the framework of the proposed OAM model.

3.1 Survivability

Survivability refers to the ability of a network to provide continuous service in the presence of failures i.e. a survivable network can withstand and recover from failures. The basic types of network failures include link and node failures. Link failures are failures that occur as a result of a cable break whereas node failures are those that occur as a result of equipment failures at network nodes. Survivability techniques are generally classified as pre-designed protection or dynamic restoration; these mechanisms are discussed in the sections hereunder.

3.1.1 Protection

Protection is a pro-active survivability technique that uses pre-assigned capacity to ensure network survivability. Resources such as fibers, wavelengths, switches etc. dedicated to protection purposes are reserved for recovery from failures either at connection setup time or when the network is being designed. The major advantage of this approach is that it yields a 100 percent restoration guarantee, which means that a failed path will be restored when a failure has occurred. Furthermore, the restoration times for these schemes are very fast because the protection path already exists when the failure occurs. The major disadvantage of these techniques is that they make inefficient use of the available network capacity. The most common pre-designed protection schemes currently used in first generation optical networks (i.e. SDH) are Automatic Protection Switching (APS) and Self-Healing Rings (SHRs). The WDM layer protection schemes are very similar to those used in SDH systems. The reason for this is that currently WDM is being deployed to upgrade existing SDH networks, which means that the network topologies have remained relatively unchanged. It therefore seems reasonable to apply similar protection techniques at the WDM layer.

Protection for WDM point-to-point links

WDM technology has been largely deployed as point-to-point networks as pointed out in chapter two. APS presents a natural protection mechanism for these kinds of networks. It can however only handle link failures. APS can be implemented as 1+1, 1:1 or 1:N protection. In 1+1 APS, a protection link is provided for every working link. At the source node the signal is transmitted onto both the working and protection links. At the destination node, the receiver compares the two received signals and chooses the best one. In the case that a link failure occurs, the operational link carries the signal to the receiver. In 1:1 APS each working link is also provided with a protection link. The difference in this case is that at the source node the signal is only transmitted on the working link. When a fault occurs, the traffic is switched to the protection link. Under normal conditions, the protection link either remains idle or is used to transmit low priority traffic. In1: N APS, N working links are protected by a single protection link. When one of the working links fail then the traffic from that link is switched onto the protection link. The traffic is reverted to the failed working link once it has been restored so that the protection link can offer backup to the other working links. It should be noted that this scheme only offers a 100 percent restoration guarantee in the event that only one link fails. If two or more links fail simultaneously, this metric decreases below the optimum value. The scheme however, utilises the network resources more efficiently than for 1+1 and 1:1 APS.

Protection for WDM rings

Self-healing rings are more flexible than APS in that it can handle both link and node failures. In WDM rings, there are two kinds of protection mechanisms that can be implemented: OCh-Ring or OMS-Ring protection.

OCh-Ring protection is analogous to SDH path protection discussed in [12]. In this scheme the protection is implemented at the OCh layer network, as illustrated in figure 3.1 below. With OCh-Ring protection each wavelength channel is protected separately. There are two ways in which OCh-Ring protection can be implemented. Protection can be either *shared* or *dedicated.* In the case of shared protection half of the wavelengths in the fiber are allocated to working traffic and the other half to protection traffic. So if for example an 8-channel, 2-fiber implementation is considered then λ_{1-4} in the first fiber will be for working traffic and will be protected by λ_{1-4} in the second fiber.

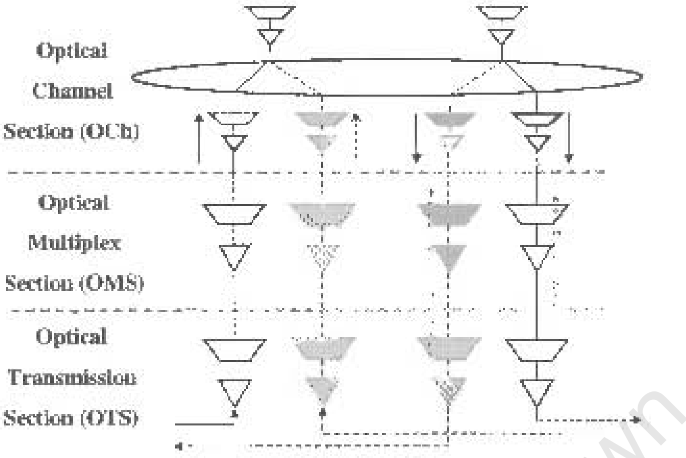


Figure 3.1 Illustration of OCh layer network protection

Similarly, λ_{5-8} in the second fiber will be dedicated to working traffic and will be protected by λ_{5-8} in the first fiber. In the case of dedicated protection, however, one fiber is utilized for working traffic and the other (redundant) fiber is the protection fiber. So then, considering the 8-channel example, λ_{1-8} in the one fiber will be for working traffic and λ_{1-8} on the other fiber will serve as the protection capacity. The implementation of dedicated and shared protection mechanisms for a 2-fiber 8 channel unidirectional case is illustrated in figure 3.2 below.

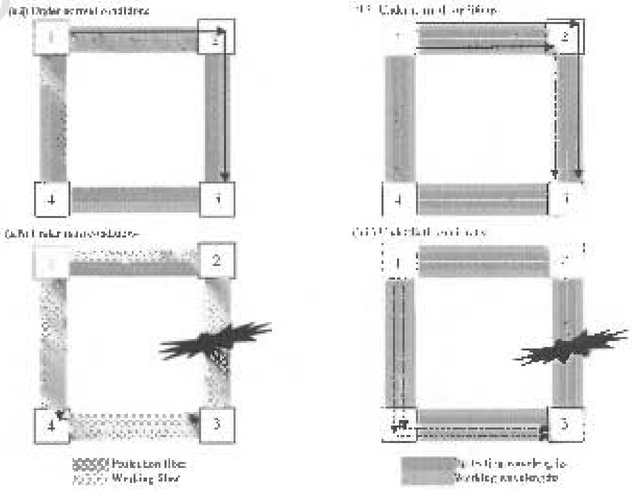


Figure 3.2 Illustration of a two-fiber (*a*) dedicated and (*b*) shared protection ring

OMS-Ring protection is implemented at the OMS layer network as illustrated in figure 3.3 below. In this scheme all the wavelength channels are simultaneously protected. This kind of protection can be quite advantageous because it could result in less optical switches being required for protection as compared to the case for protection at the optical channel layer. However, a disadvantage of this scheme is that it will only protect against signal degradation and cable-cuts, but not against higher network layer defects e.g. (de-)multiplexer failure. As was the case with OCh- Ring protection, OMS-Ring protection can also be implemented as either shared or dedicated as shown in figure 3.2 above.

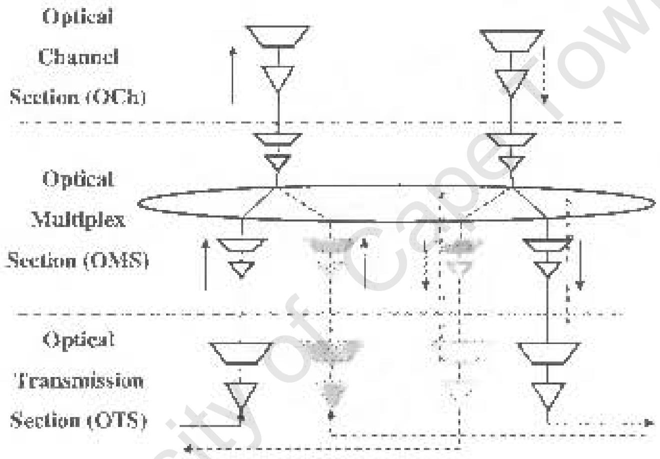


Figure 3.3 Illustration of OMS layer network protection

Protection for meshes

It is envisaged that as WDM deployment advances beyond the upgrading of existing first generation optical networks, mesh topologies using OXCs are likely to emerge. There are two ways in which pre-designed mesh protection can be implemented: link-based or path-based protection.

In link-based protection, a protection path is reserved for each link. Each working wavelength channel on a link has a protection wavelength path on the back-up path. When the link fails, traffic is re-routed around the failed link; the working part of the path is still used. An example of this scheme is illustrated in figure 3.4(*i*) below. Assume that the transmission path is 1-6-5. If a failure occurs between nodes 5 and 6, then the affected traffic can be re-routed either on protection paths 6-3-5 or 6-2-3-4-5 (link 1-6 is st8ill used). It should be noted that normally the shortest back-up path is chosen. In this kind of protection scheme the end nodes of the failed link (i.e. nodes 5 and 6 in this example) are responsible for initiating the recovery. Link-based protection may be either dedicated or shared. In the case of dedicated protection, protection wavelengths operating on overlapping protection paths must be different. This is illustrated in figure 3.4 (*ii*). In the example, the protection paths for links 2-3 and 5-6 are 2-6-3 and 6-3-5, respectively. The protection paths overlap on link 6-3, therefore the wavelengths allocated to these protection paths must be different. Lets assume that the protection paths are operating at λ_1 and λ_2 as indicated in the diagram. Assume further that λ_1 is the back-up wavelength for working wavelength λ_1 on link 5-6 and λ_2 is the back-up wavelength for λ_1 on link 2-3 (there can be no λ_1 protection wavelength on this path). Note that if link 2-3 fails that wavelength conversion (from λ_1 to λ_2) will be required. Dedicated link-based protection can provide a 100 percent restoration guarantee; the scheme however is inefficient and can be very complex because of the wavelength assignment constraint. Shared link-based protection differs from the previous scheme in that it allows wavelengths to be shared between overlapping protection paths. This scheme is illustrated in figure 3.4 (*iii*). The protection paths for links 2-3 and 5-6 are the same as for the above discussion; however, the protection wavelengths in both cases are λ_1. Shared link-based protection makes more efficient use of the wavelength resources and also provides 100 percent restoration guarantee for single link failures [54].

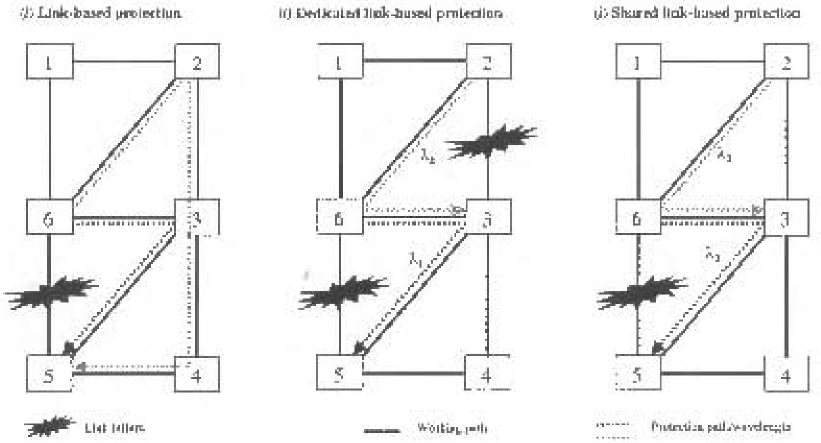


Figure 3.4 Illustration of link-based protection schemes for mesh networks

In path-based protection, protection is reserved for each path (and hence also all the wavelengths on that path). The protection path is normally disjoint from the working path. If a link on a particular working path fails, the source and destination nodes of that path will initiate the recovery mechanisms. This is illustrated in figure 3.5 (*i*) below. Assume that the working path is 1-6-5, where 1 is the source node and 6, the destination node. Assume further that a failure occurs on link 6-5. The affected traffic will then be re-routed along the protection path 1-2-3-5 as indicated in the diagram (note that the two routes are disjoint). As with link-based protection, path-based protection can also be either dedicated or shared. Dedicated path-based protection is similar to its link-based counterpart. In the case of dedicated path-based protection, the backup wavelength on the links of the protection path is reserved for a particular working path. Overlapping protection paths cannot use the same wavelength even if the working paths do not overlap. This scheme is illustrated in figure 3.5(*ii*). Lets assume that there are two working paths, 1-6-5 (connection 1) and 2-3-4 (connection 2), both operating at the same wavelength, lets say λ_1. The protection path for connection 1 is 1-2-3-5, which has to be allocated a wavelength other than λ_1, because this is a working wavelength on connection 2. The protection path for connection 2 is 2-6-3-5-4 and it has to be assigned a wavelength other than λ_2 (because it has already been allocated to the other protection path and the two wavelengths must be different). Note that the two protection paths overlap on link 3-5.

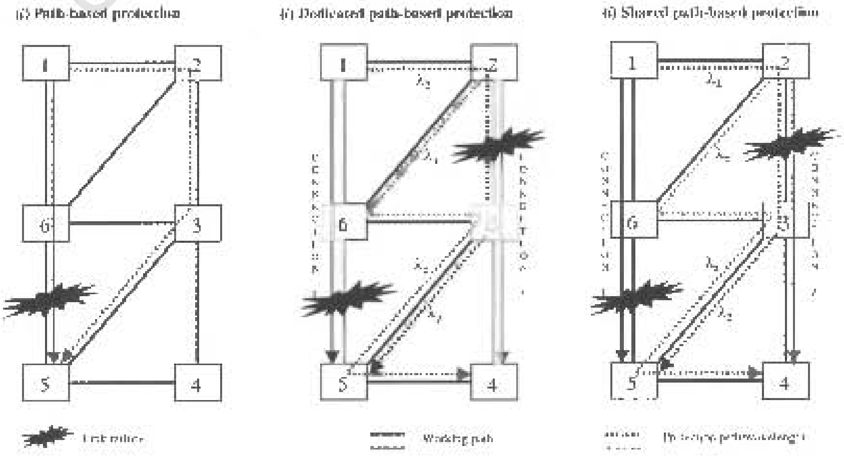


Figure 3.5 Illustration of path-based protection for mesh networks

Clearly, this protection scheme does not make very efficient use of the wavelength resources. However, it guarantees 100 percent recovery from single-link failures and in certain cases also from multi-link failures. Shared-path protection differs from dedicated path protection because it allows the wavelengths to be re-used on overlapping paths. This is illustrated in figure 3.5(*iii*), where both protection paths use the same wavelength, thus taking advantage of the overlap on link 3-5. This scheme makes efficient use of the wavelength resources (one wavelength used as opposed to 2 for the dedicated approach).

In [55], the authors evaluate the number of fibers that will be required to implement link- and path-based protection schemes. A virtual wavelength path (VWP) network model with single link failure is used to perform the evaluation. The authors show that path-based restoration schemes reduce the number of fiber required in the network by between 20 to 25 percent, when compared to the link-based approach. Ramamurthy et al. [58], compare the wavelength capacity requirements (the sum of the number of wavelengths required on each link) for dedicated and shared path- as well as shared-link protection that will guarantee 100 percent restoration. Their results suggest that shared path protection provides significant savings in capacity utilization over the other two methods. For example, "in a 15-node mesh network, the results show that 59 wavelength links suffice if no protection is needed for a 25-connection demand. The number of wavelength links required for dedicated path, shared path and shared link protection schemes in the same example is 163, 99, 189, respectively" [58]. They also evaluate and compare the switching times for the various schemes by using a protection switching time model, which they propose. Their results show that the switching times are lowest for shared link protection when the cross-connect configuration times are low (10μs), but when cross-connect configuration times are high (500μs), then dedicated path protection performs best and shared path protection worst.

3.1.2 Restoration

Restoration differs from protection in that the protection paths are not pre-assigned, but are determined and allocated dynamically if and when failures occur. The approaches used for dynamically allocating a restoration path are the same as those discussed for pre-designed protection above. What is critical in this case is the control protocol and reducing the restoration time, since the search for a backup path is only initiated after the failure has occurred. In [58] the authors propose a distributed control protocol that implements a two-way (each request sent, waits for an acknowledge message) backup path search scheme. They evaluate the performance of path- and link-based protection mechanisms using the proposed protocol. The performance metrics used are restoration efficiency (which is defined as the ratio between the number of lightpaths restored to the number of lightpaths that have failed) and restoration time. Their results show that the link-based restoration scheme has a shorter restoration time than path-based protection. They attribute this result to two reasons. Firstly, the link-based scheme does not need to signal messages to the end nodes, hence decreasing the propagation and processing delays for the control messages. Secondly, finding a back-up path around a link is likely to be shorter than a path-based backup path. Furthermore, their results also show that the restoration time in both cases are of the order of milliseconds. The results obtained for the restoration efficiency show that the path-based schemes perform better than the link-based schemes. This result is logical and can be attributed to the fact that for a path-based scheme, the backup path can be on any wavelength whereas for the link-based scheme the wavelength of the backup link must match the wavelength on the existing path (else wavelength conversion is required). It was also shown that the restoration efficiency decreases with increasing load. This can be attributed to the fact that there are less free channels at higher load hence more reservation conflicts will occur.

3.2 Operation, Administration and Maintenance

In chapter one OAM is defined as the collection of processes that support the NM function. Also, it was stated that OAM actions include measuring the signal and NE performance, reporting on the degradation or failure of the signals or NEs and altering the configuration of NEs. In this section a generic OAM model for the optical layer is proposed. Thereafter, performance monitoring and fault detection, which both form an integral part of the OAM process is discussed.

3.2.1 Proposed Generic OAM model

The proposed OAM model has a hierarchical structure and is based the OMN model proposed by the ITU-T in [27]. This hierarchical structure will act as a filter thus limiting the number of notifications processed and acted on by the NMS. It is envisaged that certain NM functions are "demoted" from the NMS to the subnetwork managers thereby implementing a distributed hierarchical control structure. At the lowest level of the hierarchy are the ONEs, which are either managers or agents. The ONEs are grouped together to form OAM-domains, which are the next higher level in the hierarchy. The OAM-domains are in turn grouped together to form OAM-subnetworks, which are the next higher level in the hierarchy. In each OAM-subnetwork or OAM-domain one ONE, which is also a manager, will act as a gateway ONE (GNE) to the next higher level of the hierarchy. For ease of reference, the GNE of the OAM-sunbetwork will be referred to as an S_GNE and that of the OAM-domain, as the D_GNE. All communication to an entity outside the OAM-subnetwork or OAM-domain shall proceed via the GNEs. At the OAM-subnetwork level, the GNE will communicate with the OSs of the NMS. The proposed structure is illustrated in figure 3.6 below.

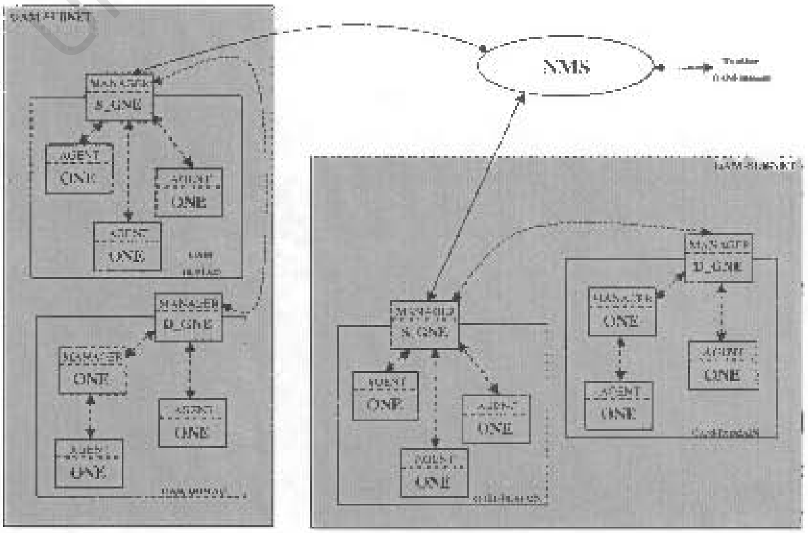


Figure 3.6 Illustration of the proposed generic OAM model

The division or sub-division of a network into OAM-domains and OAM-subnetworks is largely dependent on the network administrator. However, it is required that each domain has at least end-to-end optical channel functionality. It therefore follows that a subnetwork will consist of a concatenation of functional optical channels, the minimum being one.

Each level of the hierarchy is responsible for OAM monitoring and control; an anomaly shall be reported when it occurs and only if it requires maintenance attention. Within the OAM-domain all OAM-signaling will be done through the Embedded Communications Channel (ECC) as specified by the ITU-T in [27]. Within an OAM-subnetwork communication between the GNEs of the OAM-domains and that of the OAM-subnetwork, will be done through the overhead supervisory channel (OSC).

Figures 3.7 and 3.8 below illustrate how the scheme can be implemented for two simple network implementations. In figure 3.7, a simple point-to-point WDM network with OA functionality is illustrated.

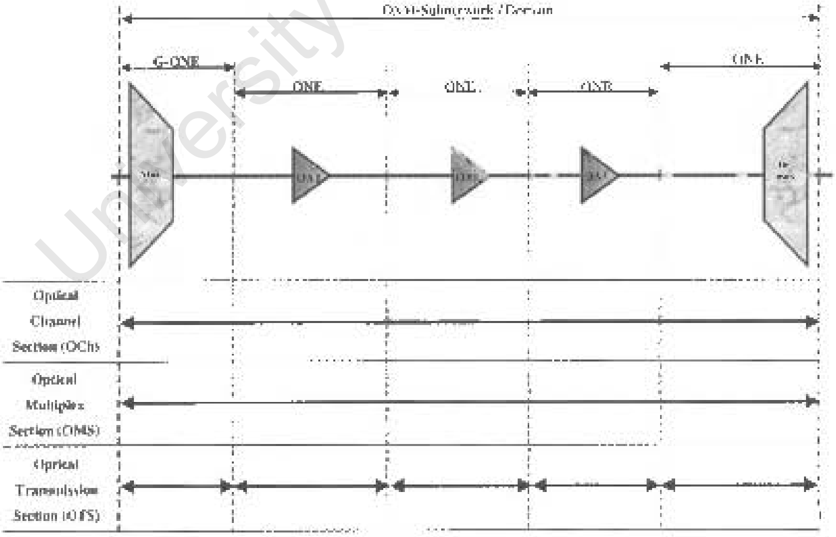


Figure 3.7 Application of the OAM model to a simple point-to-point network with OA functionality

This network can be considered to consist of either one OAM-subnetwork or one OAM-domain. The (de-)multiplexers terminate the optical channels, which makes it a valid subnetwork or domain. The example in figure 3.8 also implements a point-to-point link, however, in this instance, OXC and OADM functionality are included. Here, the link is considered to consist of one OAM-subnetwork and two OAM-domains, as indicated in the diagram. This too is a valid implementation of the OAM-model because the OXC terminates the optical channel, thus giving both domains end-to-end optical channel functionality.

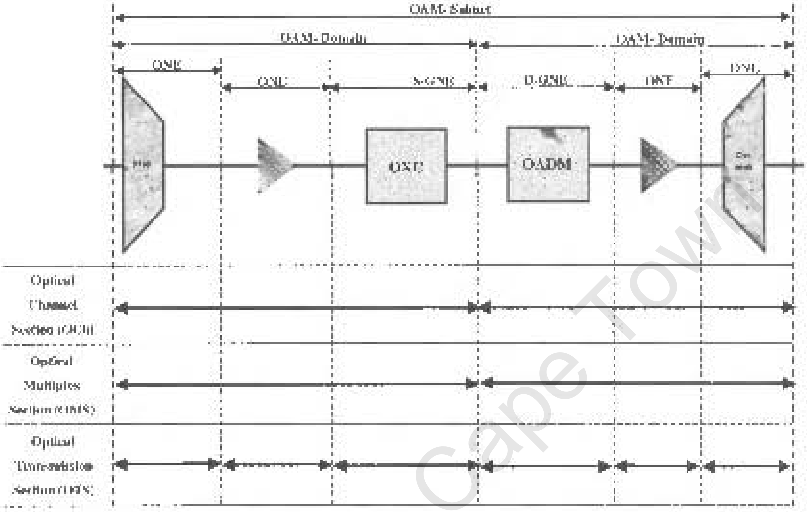


Figure 3.8 Application of the OAM model to a simple point-to-point network with OXC and OADM functionality

The reason for the sub-division of the OAM-subnetworks into OAM domains is to alleviate the processing burden at the GNEs and OSs. In today's networks, IP-based services are becoming the dominant transmission technology and this trend is expected to continue as the multimedia age comes to the fore. The bandwidth requirements for these services change frequently and new managed objects (MOs) will be added to the network as lightpaths are set-up and torn-down to accommodate these changing demands. The NM system (NMS), operating systems (OSs) and managing elements must be able to cope with these changing conditions within the network, without overloading its own resources. The objective of the proposed model is to distribute the processing load amongst the GNEs and to decrease then load at OSs.

3.2.2 Performance Monitoring

Performance monitoring is an OAM function, which supports and enables fault detection and hence network survivability. In this section, the performance parameters associated with the optical domain are firstly identified. Thereafter, the performance monitoring procedures, for the optical networking components discussed in chapter two, will be presented. It should be noted that the performance monitoring and fault detection techniques demonstrated here are not new, but are currently being used in SDH networks. They are merely adapted here in an attempt to demonstrate the validity of the proposed OAM model. The reason for adopting this approach is because of the similar layering structure of the SDH and optical layer.

Performance parameters

In chapter one, BER and SNR were identified as the parameters used to evaluate the performance of the communications systems. The BER is an electrical domain parameter and measurement thereof will require that the signal be converted from the optical to electrical domain. This is undesirable because it will limit the signal format transparency, which is one of the objectives of the AON. *SNR* on the other hand can be measured in the optical domain and is therefore considered as an optical domain performance parameter for the purpose of OAM. *Optical signal power* has also been identified as a performance parameter because it influences the operation of ONEs at all three of the layer networks. *Wavelength wander* (which is determined from the channel frequency) and *jitter* are the other parameters that have been identified for use as performance parameters. The discussion presented hereafter demonstrates how these performance parameters are utilized to perform performance monitoring and fault detection techniques.

Performance monitoring

This procedure incorporates functions dealing with *measurement, supervision* and *analyzing* of performance parameters of both the signal (i.e. at the network level) and the network equipment (i.e. at the NE level). For the signal, the measured performance parameters are the signal power level, OSNR as well as the channel frequency. The power level has to be measured at all the layer networks, the OSNR only needs to be measured at the OTS and OCh layer networks whereas the channel frequency only needs to be measured at the OCh layer network. Signal performance is normally determined by measuring the parameters at the input and output of the NEs. This also serves to confirm the integrity of the NEs in certain cases (e.g. OAs). The discussion hereafter considers the physical implementation of the optical networking componenets. The physical implementations of these devices are illustrated and it is also explained how, where and why the parameters are measured. Only static (i.e. non-configurable NEs are considered).

The first case to be considered is that of the wavelength (de-)multiplexer. Recall that the function of the multiplexer is to combine wavelength channels onto a single fiber whereas the de-multiplexer separates out the individual wavelengths. The most widely used physical implementation of the (de-)multiplexer device, is the arrayed waveguide grating (AWG). The channels are combined or separated out by the varying diffractive indices of the grating device. The wavelength (de-)multiplexer is a passive device, so measuring the power supply voltage is optional. At the input of the multiplexing device, the channel frequency (λ_k-f_{in}), input power (λ_k-P_{in}) and OSNR (λ_k-$OSNR_{in}$) have to be monitored. The channel frequency checks the integrity of the transmitter laser and is also required for channel identification. The input channel power checks the integrity of the laser (if the multiplexer is preceded by a laser) or the integrity of the preceding fiber as well as the integrity of the signal entering the OMS layer network. The OSNR checks the integrity of the input signal. The combined signal power at the output of the multiplexer also has to be monitored as this checks the integrity of the multiplexing device as well as the signal power level being passed to the OTS layer network. The physical implementation of a wavelength multiplexer with optical layer mapping as well as the performance parameters that have to be monitored is illustrated in figure 3.9 below.

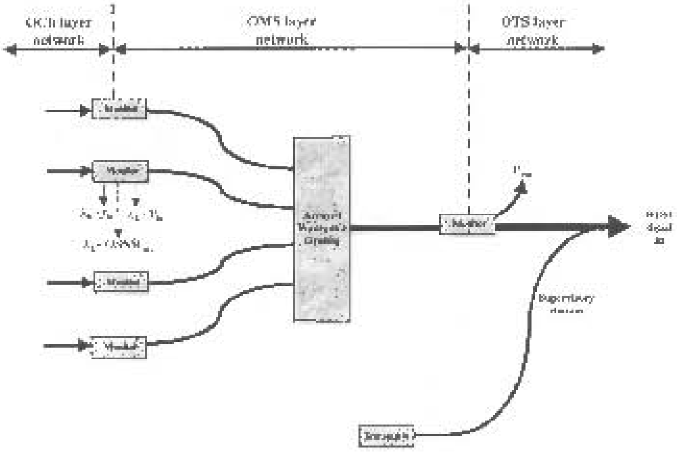


Figure 3.9 Physical implementation of a wavelength multiplexer illustrating the measured performance parameters

The de-multiplexing device is just the inverse of the multiplexing device. In this instance, the combined signal power level at the input of the device (P_{in}) has to be monitored. This parameter checks the integrity of the preceding fiber and also the signal power level that will be passed to the OMS layer network. For each of the channels, the frequency (λ_k-f_{out}), signal power (λ_k-P_{out}) and OSNR (λ_k-OSNR$_{out}$), have to be monitored. The channel frequency is required for channel identification and to check wavelength wander. The channel power at the output is used to check the power level of the signal that is passed to the OCh layer network and together with the OSNR, is required to check the end-to-end integrity of the channel. The physical implementation of a wavelength de-multiplexer as well as the performance parameters that have to be monitored is illustrated in figure 3.10 below.

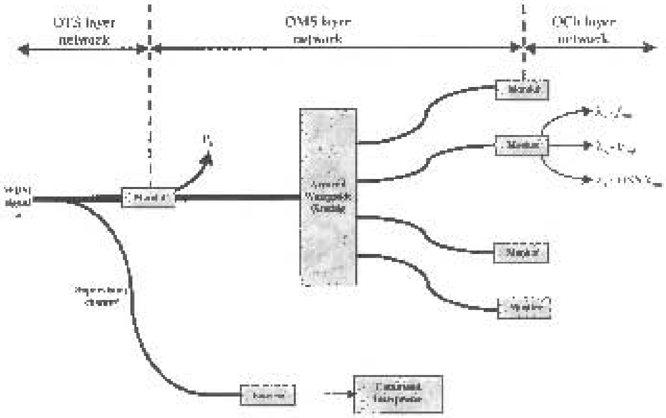


Figure 3.10 Physical implementation of the wavelength de-multiplexer illustrating the measured performance parameters

For the case of the OA, the EDFA implementation will be presented; the reason for this is that the EDFA is most widely used. Recall that the role of the OA is to increase the power level of the optical signal in order to compensate for power losses experienced along the optical path. In EDFAs, amplification is achieved by passing the signal through a length of fiber that has been doped with erbium. A pump laser is used to excite the erbium atoms to higher energy levels thus releasing photons, which increases the power level of the data signal [9]. For the laser the output power ($P_{L\text{-}out}$) and current (I_L), have to be measured. These parameters check that the performance of the laser is within specification. Another parameter, which also has to be monitored for the pump laser, is the laser temperature (T_L). The reason is that a change in temperature causes a shift in the laser center emission frequency, which will in turn reduce the gain of the OA. For the OA as a whole, other performance parameters are the amplifier gain (G), amplified stimulated emission (ASE), supply power and OSNR. The physical implementation of the EDFA as well as the performance parameters that require monitoring is illustrated in figure 3.11 below.

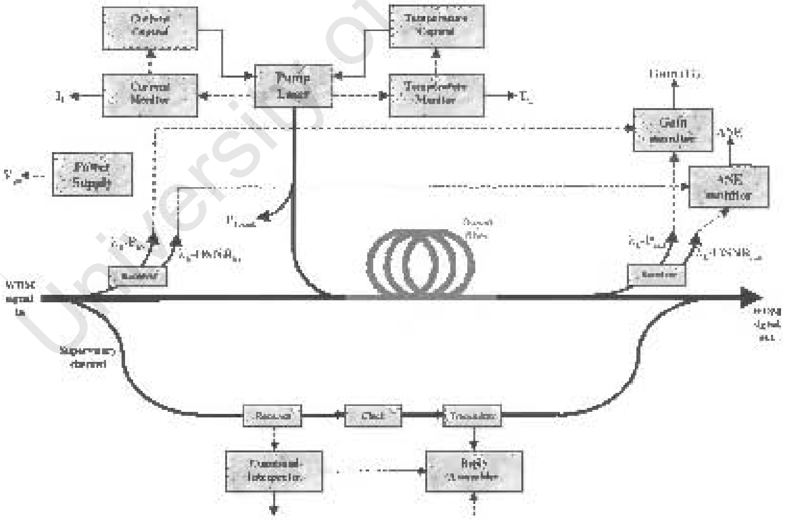


Figure 3.11 Physical implementation of the EDFA illustrating the measured performance parameters

In this implementation only one wavelength channel is used to determine the gain and OSNR, however, this is not compulsory and depends on the vendor. Ideally, the input and output powers of all the wavelength channels have to be measured, but because of the cost implication this is not yet feasible. The input-signal power level (λ_k-P_{in}) is not only used for the gain calculation but it simultaneously checks the integrity of the preceding fiber. Similarly, the output signal power level (λ_k-P_{out}) is used to calculate the gain of the OA as well as to check the integrity of the amplifier itself. One drawback of this approach is that it must be ensured that the channel being measured is always in use otherwise the results will be erroneous.

The OADM implementation, which includes an EDFA, wavelength multiplexers and de-multiplexers is considered. Recall that the OADM allows a certain number of wavelength channels to be added or dropped at an intermediate site. The performance parameters associated with the OA and (de-)multiplexer have already been discussed above. In addition to those parameters, for the added channel, the signal power (λ_{add}-P_{in}) and frequency (λ_{add}-f_{in}) should be monitored. These parameters check the integrity of the transmitter; the channel frequency is also required for channel identification. Similarly, for the dropped channel the signal power (λ_{drop}-P_{out}), frequency (λ_{drop}-f_{out}) as well as the OSNR (λ_{drop}-$OSNR_{out}$) has to be monitored. The channel power and OSNR check that the end-to-end channel integrity has been maintained and the channel frequency is required for channel identification. The physical implementations of the OADM as well as the monitored parameters are illustrated in figure 3.12 below.

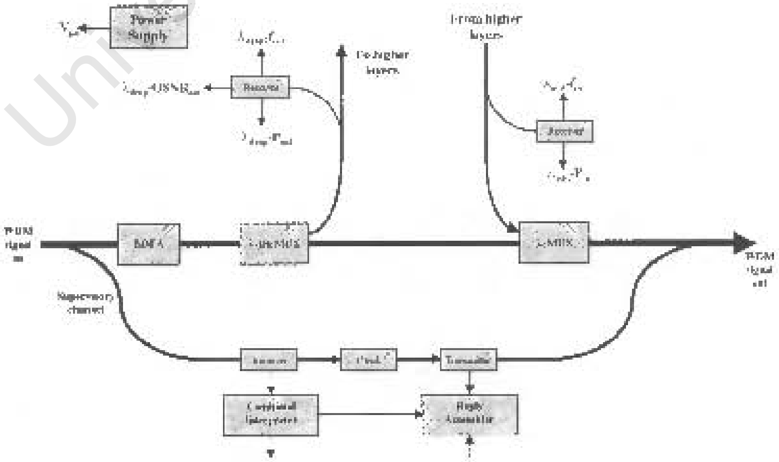


Figure 3.12 Physical implementation of the OADM illustrating the measured performance parameters

It should be noted that though only one channel is added or dropped in this example, if more than one channel is added or dropped then the performance parameters should be monitored for each of the channels.

Lastly, the OXC, which is implemented using OAs, wavelength (de-) multiplexers and SEs, will be considered. In addition to the performance parameters already mentioned for the OAs and wavelengths (de-)multiplexers, the optical signal power (λ_k-$P_{in/out}$), OSNR (λ_k-$OSNR_{in/out}$) as well as the channel frequency (λ_k-$f_{in/out}$) have to be monitored at both the input and output to the SEs. At the input of the SEs, the parameter values can be taken from the monitored de-multiplexer parameters. The signal power level and OSNR checks the integrity of the wavelength channels received from the OMS layer network, before passing them to the SE. The channel frequency at both the input and output is required for channel identification as well as checking the integrity of the switching operation. The OSNR and the output power are used to check the signal quality, the integrity of the SE as well as the integrity of the signal being passed to the OMS layer network. The physical implementation of the OXC as well as the performance parameters, which require monitoring, is illustrated in figure 3.13 below.

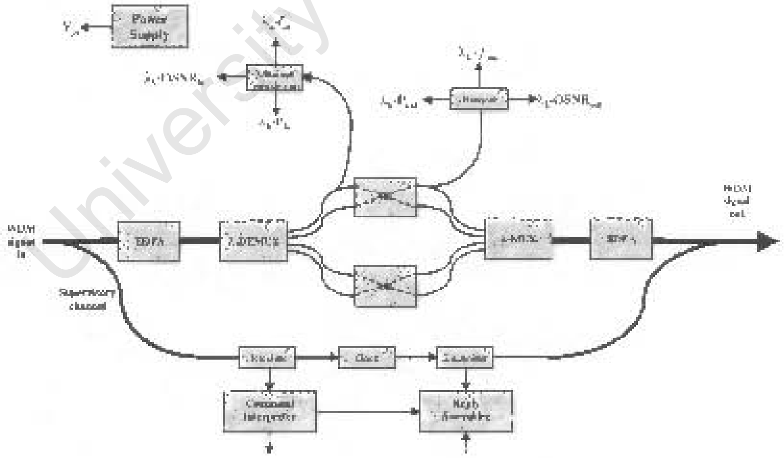


Figure 3.13 Physical implementation of the OXC illustrating the measured performance parameters

The measured performance parameters are summarised in the table below.

λ multiplexer & de-multiplexer	**Optical Add Drop Multiplexer (OADM)**
Input signal power (λ_k-P_{in})	Signal power (λ_{add}-P_a) & channel frequency (λ_{add}-f_a) of add channel
Combined signal power (P_{out})	Signal power (λ_{drop}-P_{out}), OSNR (λ_{drop}-OSNR$_{out}$) & channel frequency (λ_{drop}-f_{out}) of dropped channel
OSNR (λ_k-OSNR$_{in}$)	And also the parameters measured for wavelength mux, demux and OA
Channel Frequency (λ_k-f_{in})	
Optical amplifier (OA)	**Optical Cross Connect (OXC)**
Pump laser output power (P_{LD})	Input (λ_k-P_{in}) and output power (λ_k-P_{out}) to the SEs
Pump laser current (I_L)	Input (λ_k-OSNR$_{in}$) and output OSNR (λ_k-OSNR$_{out}$) to the SEs
Pump laser temperature (T_L)	Input (λ_k-f_{in}) and output channel frequency (λ_k-f_{out}) to the SEs
Amplifier gain (G) from (λ_k-P_{out}) - (λ_k-P_{in})	And also the parameters measured for wavelength mux, demux and OA
Amplifier Stimulated Emission (ASE)	
Power supply (P_{ps}) and OSNR (λ_k-OSNR$_{total}$)	

Table 3.1 Summary of the performance parameters to be measured for the signal and NEs

Performance monitoring can adopt one of two approaches: *threshold crossing alarm (TCA) surveillance* and *trend analysis*. The trend analysis approach looks at all the performance parameters continuously. It is costly to implement and is normally only applied to high bit rate and premium services. The TCA surveillance approach on the other hand only considers periods of poor performance. Typically, all the performance parameters have a two threshold values associated with it: degradation and failure threshold. When a parameter crosses any one of these thresholds an indication or notification is issued (normally only after a period of persistency checking). In this study only the TCA surveillance approach is considered.

3.2.3 Fault Detection

Performance monitoring is a pre-requisite for fault detection. The fault manager is invoked when it receives a notification from the performance monitoring process and it is responsible for raising the appropriate alarm signals. The fault manager also has to transmit information about the failure or degradation to the maintenance staff (which is not considered in this study) and the affected NEs. In general, there are two kinds of alarms that can be generated: a failure alarm and degradation alarm. One of these alarms can then be generated for any of the performance parameters and at any one of the layer networks. The possible alarm signals for each of the layer networks are tabulated in table3.2 below.

OTS	OMS	OCh
Signal Specific:	LOSU (Loss of Supervisory)	
LOS (Loss of Signal)	LOS (Loss of Signal)	LOS (Loss of Signal)
SD (Signal Degrade)	SD (Signal Degrade)	SD (Signal Degrade)
OSNR_F/D (OSNR Failure or Degrade)	LFA (Loss of Frequency Alignment)	LCID (Loss of Channel ID)
LOSU (Loss of Supervisory)	MOW (Mismatch of Wavelength ID	LOSU (Loss of Supervisory)
AIS (Alarm Indication Signal)	AIS (Alarm Indication Signal)	AIS (Alarm Indication Signal)
RDI (Remote Defect Indication)	RDI (Remote Defect Indication)	RDI (Remote Defect Indication)
Equipment Specific:		
OAF (OA Failure)	PSF (Power Supply Failure)	TXF (Transmitter Failure)
OAD (OA Degrade)	PSD (Power Supply Degrade)	TXD (Transmitter Degrade)
OSP (Optical Surge Protection)		PSF (Power Supply Failure)
PSF (Power Supply Failure)		PSD (Power Supply Degrade)
PSD (Power Supply Degrade)		

Table 3.2 Failure alarms that can be generated by the NEs

When a loss of optical signal power is detected or when the signal power level drops below the absolute minimum the loss of signal (LOS) and loss of signal supervisory (LOSU) alarms will be raised. A signal degradation (SD) alarm is raised if the signal power level drops below the degradation threshold value. Similarly, when the OSNR drops below the failure and degradation threshold values, the OSNR failure (OSNR_F) or OSNR degradation (OSNR_D) alarms will be raised. The monitored channel frequency will be used to determine frequency alignment as well as to perform channel identification and will subsequently initiate the relevant alarms. The alarm indication signal (AIS) and remote defect indication (RDI) alarms are used to notify upstream and downstream NEs that a fault has occurred; this will be demonstrated in the examples below. Regarding the equipment specific alarms, these will be initiated if or when one of the performance parameters for that equipment crosses a threshold.

The fault detection mechanism is demonstrated using two examples; only hard faults (LOS alarms caused by fiber breakage, in-line OA breakage, OXC breakage etc.) are considered. When an LOS is detected, AIS and RDI are generated. AIS is transmitted upstream on the user payload channel and its purpose is to inform the client layer that a serious alarm (fault) condition has occurred. RDI on the other hand is transmitted downstream on the return path and its purpose is to inform the source that the payload did not reach its destination. In general the reception of an AIS or RDI indicates that the layer network has lost its integrity and normally it initiates an automatic switching protocol (ASP) or network restoration.

In the first example a simple point-to-point network with optical amplification functionality is considered, as illustrated in figure 3.14 below. The network is considered to consist of one OAM-domain in accordance with the proposed OAM-model. A fiber breakage occurs between OA1 and OA2 as indicated in the diagram. The first upstream NE, which in this case is OA2, detects the OTS-LOS. It then generates the OTS-AIS, which is sent upstream instead of the payload, as well as the OTS-RDI, which is returned downstream. The OTS-AIS is detected and regenerated by all the upstream NEs (OA3 and the wavelength de-multiplexer) that have OTS functionality along the path.

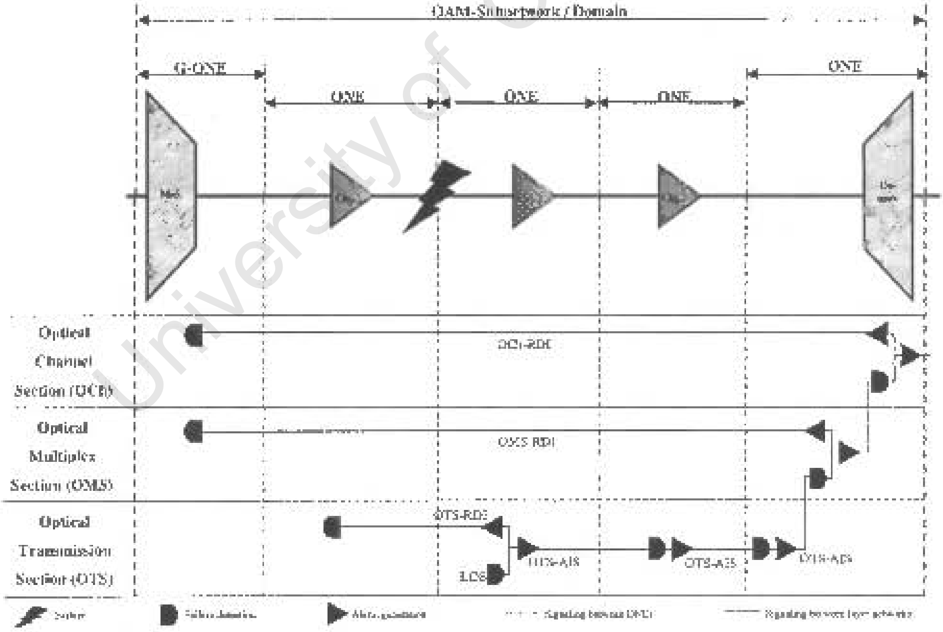


Figure 3.14 Illustration of the signaling which results from the fault detection mechanism in a simple point-to-point network with OA functionality

At the wavelength de-multiplexer, the OMS layer does not receive a payload or any AIS. It therefore has to generate the OMS-AIS and OMS-RDI to send upstream and downstream, respectively; similarly for the OCh layer network. The

RDI signals, which are sent downstream along the return fiber, travel in the path overhead until it reaches the first location at which the specific layer network was implemented. When the AIS is received in the OCh it triggers an OAM-subnetwork alarm, which it forwards via the OAM-Domain G-ONE to the NMS.

In the second example the point-to-point network is extended to include both OADM and OXC functionality as illustrated in figure 3.15 below. In this instance the network is considered to consist of one OAM-domain, which is further sub-divided into two OAM-subnetworks. The fiber breakage occurs between the wavelength multiplexer and the OA. As in the first example, the OTS-LOS is detected by the OA (it is the first NE it encounters upstream), which then generates the OTS-AIS and OTS-RDI and sends it upstream and downstream, respectively. The next NE encountered is the OXC, which implements all three layer networks. Since it does not receive any OMS or OCh payload, neither the OMS-AIS nor OCh-AIS, it generates the AIS and RDI for both the OMS and OCh, which it then sends upstream and downstream, respectively.

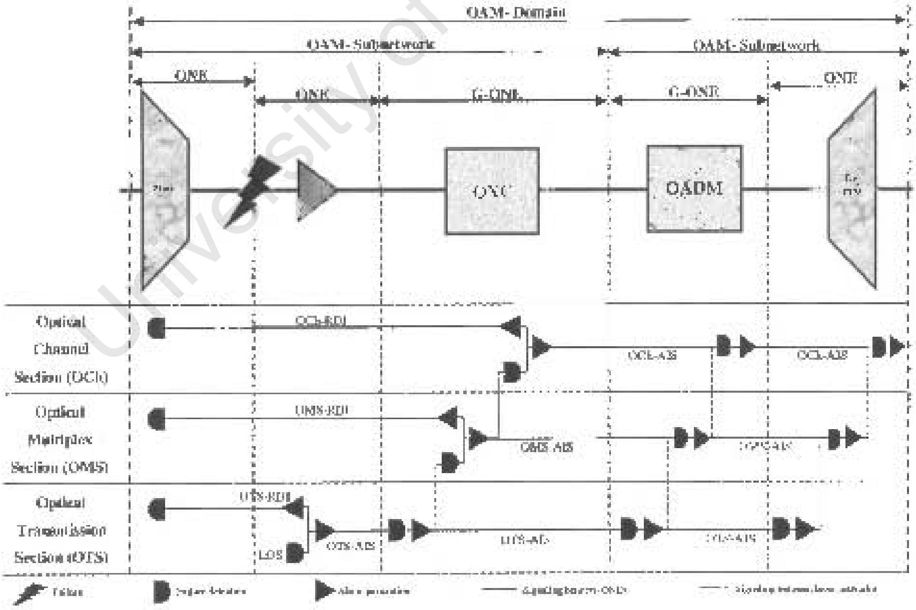


Figure 3.15 Illustration of the signaling which results from fault detection in a simple point-to-point network with OXC and OADM functionality

As was the case with the previous example, the AIS is detected and regenerated in each layer network and at each NE along the path. In this example however, the

OXC (which is the first NE to detect the OCh-AIS), triggers an alarm which is sent via the OAM-subnetwork G-ONE and OAM-Domain G-ONE to the NMS.
It should be noted that for both of these examples it is assumed that only one fiber breaks and that there is a path available on the return path. It should be noted however, that in the event of a fiber cable break, the RDI will be unavailable.

3.3 Discussions and Summary

In this chapter the issues pertaining to NM and OAM procedures were discussed. Firstly, survivability techniques for the DWDM optical were reviewed. In general the literature distinguishes between pro-active survivability i.e. protection and re-active survivability i.e. restoration. Various protection schemes for the different architectures discussed in chapter two were presented. The protection schemes proposed for WDM layer protection emulate the schemes implemented in first generation optical networks (e.g. SDH) networks.

A generic OAM model for the optical layer was then proposed and explained with the aid of two simple point-to-point network examples. The proposed model has a hierarchical structure. It sub-divides OAM subnetworks into OAM domains in an attempt to alleviate the processing overload that could potentially occur at OSs as a result of changing network conditions.

Performance monitoring for the optical layer was then discussed. The physical implementation of the optical network components were illustrated together with the performance parameters that have to be monitored. Finally, optical layer fault detection was discussed and demonstrated with the aid of simple point-to-point network examples.

In chapter four a network structure to implement an AON national network in South Africa is proposed. The application of the proposed OAM model to the national network is also demonstrated.

Chapter 4

Design Proposal of a DWDM-based All Optical Network

There are many factors that can influence the design and implementation of a DWDM network. The design approach is dependent on the main focus of the implemented network, for example, if the focus is the design of a management control plane then it is advisable to avoid optimization problems such as wavelength assignment. However, if the main focus were optimizing the network design, then it would be advisable to focus on problems relating to optimization. In the case of this research the key objective is to investigate OAM procedures at the optical layer and the focus is to implement a realistic DWDM optical network for testing the OAM procedures, therefore it will not focus on optimization issues.

The chapter begins with a proposal of a network structure to implement an AON national network in South Africa. It then explains the relation between the number of wavelengths and number of nodes and demonstrates by means of a simple example how wavelength assignment can be implemented in a four-node ring network. In section 4.3 it demonstrates how the OAM model, which was proposed in chapter three, can be applied to the national network.

4.1 Network Structure Proposal

In chapter two, the basic network topologies (i.e. ring and mesh structures) were introduced. In general, communications networks are complex structures. Their major aim is to support the current and future traffic demand by using as few network elements as possible. Many different combinations of these basic structures are possible, to implement a desired communications network. This research proposes a hierarchical interconnected OADM ring structure for the South African national network. Such an implementation is illustrated in figure 4.1 below. The reason for this network proposal is partly because the existing SDH infrastructure is implemented in this way. Furthermore, a hierarchical structure offers advantages in terms of routing, wavelength re-usability and manageability. The disadvantage of this structure is that signals have to traverse longer paths, but this is outweighed by the advantages.

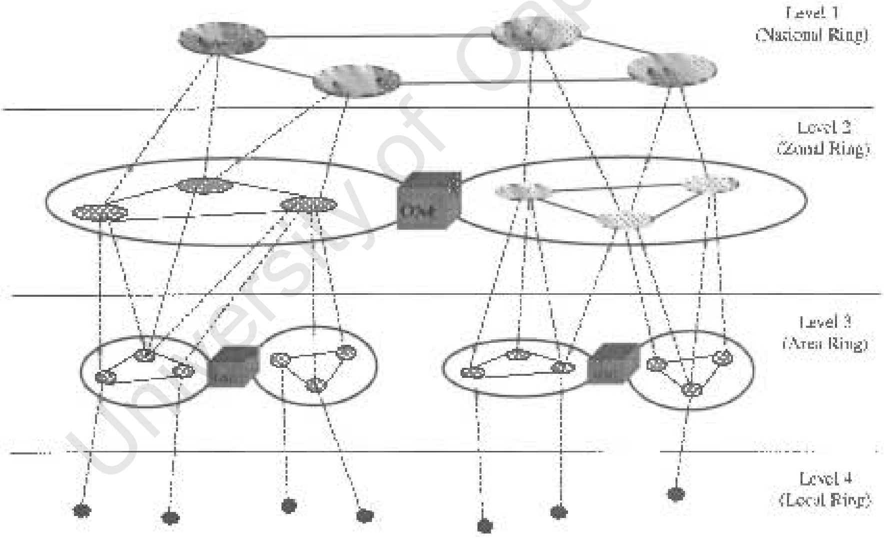


Figure 4.1 Hierarchical structure for all optical network implementation

In figure 4.1 a four-level hierarchical structure is depicted. At the highest level of the hierarchy (level 1) is the national core network, which is in itself, an OADM ring network. At the next level of the hierarchy are the zonal OADM rings. These are optically cross connected to the higher level core ring as well as to other rings at its own level. The third level of the hierarchy consists of the metropolitan or local rings, which are similarly cross-connected to the second level regional rings as well as to other metropolitan rings at its level. At the lowest level of the hierarchy (i.e. level 4) are the access networks.

In the South African (SA – international switching code 27) context it is proposed that the core ring be made up of OADM nodes at Cape Town (CPT), Port Elizabeth (PE), Durban (DBN), Johannesburg (JHB) and Bloemfontein (BFN), as illustrated in figure 4.2 below. These nodes correspond to the major switching codes 01, 02, 03, 04, and 05 for JHB, CPT, DBN, PE and BFN respectively.

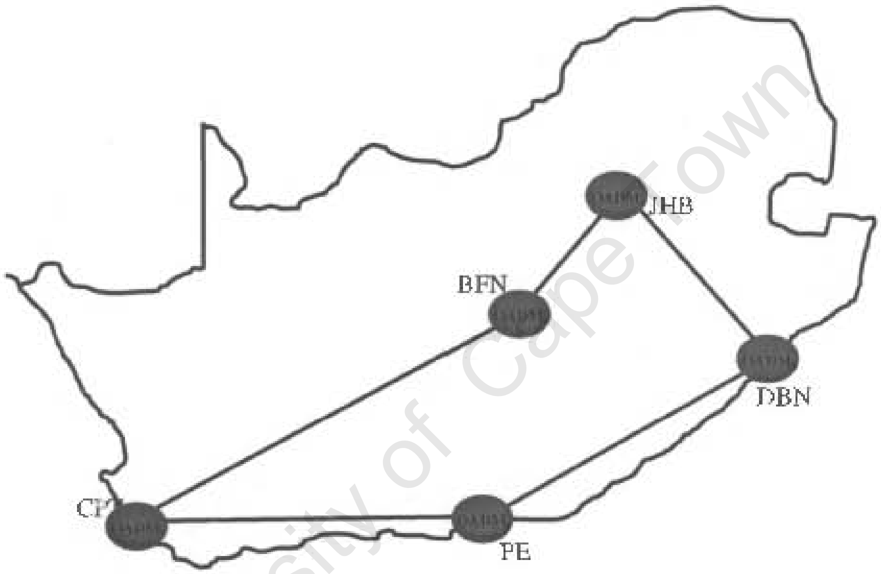


Figure 4.2 Proposed national (level 1) core ring for the envisaged AON

Considering now the second level, a possible zonal ring for the Western Cape region could consist of maybe four OADM nodes at CPT, Bredasdorp (BDP), Beaufort West (BFW) and Vredendal (VDL). These nodes correspond to switching code 021, 023, 027 and 028 for CPT, BFW, VDL and BDP, respectively. This zonal ring is illustrated in figure 4.3 below.

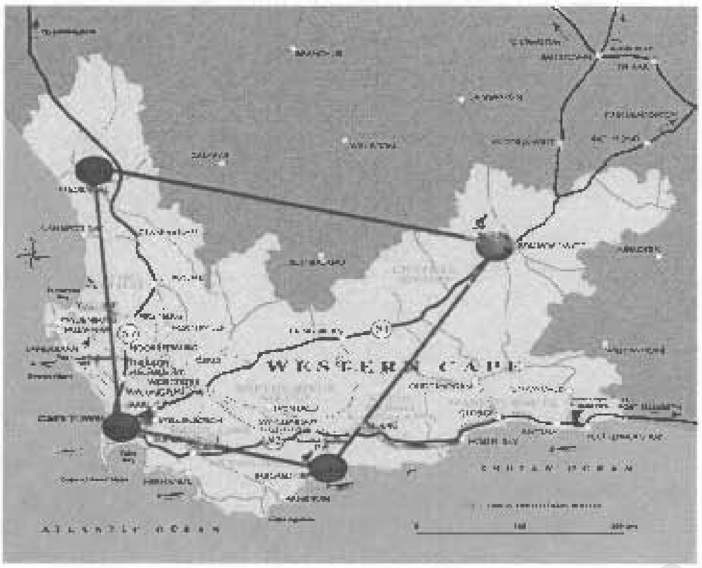


Figure 4.3 Proposed zonal (level 2) ring structure for the Western Cape

For the metropolitan (local) ring level an example ring could consist of OADMs at CPT, Fish Hoek (FH), Mitchells Plain (MP) and Goodwood (GWD). These nodes correspond to switching codes 21-3, 21-4, 21-5 and 21-7 for MP, CPT, GWD and FH, respectively. The example metropolitan OADM ring is illustrated in figures 4.4 below.

Figure 4.4 Proposed local (level 3) ring structure for the Cape Peninsula

4.2 Proposed Frequency Assignment

In [39] it is stated that the minimum number of non re-usable wavelengths (W) required for a ring structure, which will facilitate wavelength routing, can be calculated from

$$W = N(N-1) \tag{4.1}$$

where N is the number of nodes in the ring. It should be noted that this formula does not consider the case in which wavelengths between adjacent nodes are re-used. If a four-node ring, as proposed for the local ring in section 1 above, is considered then from equation 4.1 the minimum number of wavelengths required would be 12. The wavelength assignment table for this case is sown in Table 4.1 below.

	Node1	Node2	Node3	Node4
Node1	-	$\lambda_{1\text{-}2}$	$\lambda_{1\text{-}3}$	$\lambda_{1\text{-}4}$
Node2	$\lambda_{2\text{-}1}$	-	$\lambda_{2\text{-}3}$	$\lambda_{2\text{-}4}$
Node3	$\lambda_{3\text{-}1}$	$\lambda_{3\text{-}2}$	-	$\lambda_{3\text{-}4}$
Node4	$\lambda_{4\text{-}1}$	$\lambda_{4\text{-}2}$	$\lambda_{4\text{-}3}$	-

Table 4.1 Wavelength assignment for a 4-node ring

4.3 Application of Proposed OAM structure to envisaged SA AON

In chapter three an OAM model was proposed for application in an AON. The model proposed that a network be divided into OAM-subnets and OAM-domains and that management communication is between the domain and subnet proceeds via the GNEs. This section examines how this structure may be applied to the proposed SA AON.

It is proposed here that each OADM ring be managed as an OAM-domain and that OAM-subnets be made up of the OADM rings at the different levels. The reason for this is that there are many rings at the zonal and local level and these should be aggregated to reduce the management communication at these levels. At the national level then for example the core rings are also managed as OAM-domains, with only one OAM-subnet at the national level. The management information is then sent via the S-GNEs to the OSs of the NMS. In this case then the management information is not escalated up the network hierarchy, but is communicated to the management system from each of the different levels. The first advantage of this approach is that within the OAM-subnet itself the processing load will be distributed amongst the different GNEs. This precludes the occurrence of potential processing bottlenecks at the OS, which could result where many defects occur as the number of MOs changes to accommodate the changing network demand. The number of MOs changes as lightpaths are set-up or torn-down. As per example, lets consider the scenario in which at the local level there are 10 local rings in the peninsula region. Consider also that these rings are further sub-dived to form three OAM-domains, two domains consisting of three rings and one of four. If one defect should occur simultaneously in each of these rings, then the overall total number of notifications generated in the subnet is ten. Two of the domain GNEs would receive and process three notifications, whereas the other GNE would receive and process four notifications; the load is distributed amongst the domain GNEs. If the scenario is now considered in which the subnet is not sub-divided into OAM-domains, then all ten rings would report to one subnet GNE. As was considered, if a defect occurs simultaneously in each ring then the subnet GNE would receive and process ten notifications. This could also result in a longer time delay, since notifications cannot be processed simultaneously by the GNE and will thus have to be queued before processing.

4.4 Discussions and Summary

In this chapter a hierarchical interconnected OADM ring structure is proposed for the implementation of the South African national network. The advantages of this structure are that it provides a simple way of implementing a complex network and also, it simplifies the routing processes and promotes wavelength re-usability. The relation between wavelength assignment and the number of nodes is also briefly considered, though this is really an optimization problem. Finally, it demonstrates how the generic OAM model, which was proposed in chapter 3, can be implemented in the envisaged AON.

In chapter five we attempt to quantify the advantages to be gained from implementing the proposed OAM model.

Chapter 5

Network Implementation, Simulation and OAM Model Testing

In chapter three a generic OAM model for the optical layer was proposed and demonstrated with the aid of two network examples. In this chapter the implementation as well as simulation results of the network example from chapter three is presented. The chapter begins with an introduction to the OptSim simulation environment and a discussion of the methodology used to implement the software package. A short discussion about the accuracy of the OptSim performance calculations is also presented. Thereafter, the design and implementation of the point-to-point network example is presented. The OptSim network simulation results follows.

In an attempt to quantify the advantages to be gained by implementing the proposed OAM architecture, a simple NM tool was implemented. The design and implementation of the NM tool, which implements the performance monitoring and fault detection techniques discussed in chapter three is then discussed. The NM tool also implements the proposed generic OAM model. Finally, the results of the NM tool are analysed.

5.1 Introduction to OptSim

OptSim is a simulator for implementing optical communications systems. It is composed of three main parts: a graphical editor in which the system to be simulated is created, the software engine (hidden from the user) where all the numerical calculations are performed and the data post-processing and display system that allows for the display and manipulation of simulation results.

OptSim provides two different techniques under which simulations can be run. The Spectral Propagation Technique (SPT) is a spectral domain simulation and is quite useful if only the power spectra and OSNR is to be evaluated. In this method the power spectrum is propagated and it only simulates the optical components; only the linear effects are considered.

Alternatively, there is the Variable Bandwidth Simulation (VBS) technique, which simulates all the system components. It is a time domain simulation technique, which implements the TDSS method explained hereafter. In this case the full vector signal is propagated along the network as time domain samples and the linear and non-linear behaviour of both the electrical as well as the optical components are taken into account during the simulation process. The VBS technique allows the user to simulate the entire system bandwidth or only a portion of it, in which case only the channels included in the spectral slice, is simulated. The maximum VBS bandwidth must be set to a value of at least 20% smaller than the SPT bandwidth.

The OptSim simulation is based on a Time Domain Split-Step (TDSS) method, which takes into account all the linear and non-linear effects [39]. TDSS is one of the most accurate methods available for implementing optical fiber propagation systems. An alternative method for implementing optical communication system simulators is Frequency Domain Split-Step (FDSS); the two methods are compared below.

TDSS vs. FDSS

The split-step method is the most commonly used in optical system simulation tools to perform the integration of the fiber propagation equation (5.1)

$$\frac{\partial A(t,z)}{\partial z} = \{L + N\}A(t,z) \tag{5.1}$$

where $A(t,z)$ is the optical field, L is the linear operator, which takes into account all the linear effects (such as polarization mode dispersion and self-phase modulation) and N is the non-linear operator, which takes into account the non-linear effects (such as four-wave mixing and stimulated raman scattering). The algorithm separately applies L and N to $A(t,z)$ over small spans of the fiber, Δz.

There are two modes in which the split-step algorithm can be implemented: Time Domain Split-Step (TDSS) and Frequency Domain Split-Step (FDSS). The two methods differ only in the way in which the linear operator L is calculated; both methods calculate N in the same way. L is fully characterized by its impulse response $h(t)$ and TDSS calculates L in the time domain by computing its effect on $A(t,z)$ via a time domain convolution product. Equation 5.2 below represents the TDSS process.

$$TDSS \rightarrow A_L[n] = A[n] * h[n] = \sum_{k=-\infty}^{\infty} A[k]h[n-k] \tag{5.2}$$

FDSS on the other hand, calculates L in the frequency domain. Both $A(t,z)$ and $h(t)$ are sampled in time. The algorithm then multiplies the Fast Fourier transform (FFT) of $A[n]$ by the FFT of $h[n]$ and then converts the result back to the time domain by performing an inverse FFT. The FDSS process is represented by

$$FDSS \Rightarrow A_L'[n] = A[n] \otimes h[n] = FFT^{-1}(FFT(A[n]) \times FFT(h[n])) \tag{5.3}$$

The FDSS algorithm implements a circular convolution product (as indicated by the $\otimes$ symbol), which produces a signal fold-over error (i.e. aliasing) in the result. The effects of this aliasing error can be clearly observed at the output of a dispersive fiber span, which has as an input a flat envelope continuous wave (CW). The output signal will have amplitude oscillations in the envelope, which is the result of the intrinsic errors of FDSS. There are ways of avoiding this error, but these constraints in turn limit the transmission rates of the system being studied to 2.5Gbps and also limit the number of channels to one. This is not realistic for studying current DWDM systems.

OptSim, which uses the TDSS method, does not suffer from these aliasing errors. Furthermore, TDSS also accurately considers group delays. In chapter 1 it was pointed out that dispersion is one of the intrinsic effects of the frequency-wavelength-velocity inter-dependency. This effect causes time shifts among signals of different wavelengths. Because of the way in which the circular convolution is performed, the FDSS process introduces unrealistic noise sample correlation in the result. This is not the case for TDSS.

Accuracy of OptSim performance calculations

In chapter one it was also pointed out that the most commonly used performance parameters for measuring performance in digital systems are the BER and Q-factor. In OptSim the Q-factor is defined as

$$Q = \frac{m_1 - m_0}{\sigma_1 + \sigma_0} \tag{5.4}$$

where m_1, σ_1 (m_0, σ_0) are the mean and standard deviation of the received signal at the sampling instant when a logical "1" ("0") is transmitted. The BER is evaluated by

$$BER = \frac{1}{2} erfc\left(\frac{Q}{\sqrt{2}}\right) \tag{5.5}$$

which is only true if the noise at the receiver is Gaussian and inter-symbol interference (ISI) is negligible. In [39] it is reported that if the total number of bits simulated is at least 512, then the uncertainty of Q is of the order $\pm 1\,\text{dB}$. The uncertainty can further be reduced to approximately $\pm 0.77\,\text{dB}$ if 1024 bits are simulated.

The OSNR is evaluated by

$$OSNR_N = \frac{S_N}{\frac{N_{n1} + N_{n2}}{2}} \tag{5.6}$$

where S_N is the signal power which is calculated from the mean squared amplitudes of the Fourier transform, N_{n1} and N_{n2} are the absolute minima on either side of the spectrum of the detected channel.

5.2 Network Design and Implementation

Network example 2 (figure 3.8) implements a simple point-to-point network with OXC and OADM functionality. The simulation design is as follows: the center wavelength is placed at 193.1THz. The system implements four wavelength channels spaced 0.1nm apart. The system bit rate is set at 10Gbps.

The 4-to-1 wavelength multiplexer and transmitters are combined into one component. The component comprises of four NRZ transmitters and an ideal combiner for multiplexing the four signals together. The transmitter implements a CW laser source that is externally modulated by a Mach-Zehnder modulator. The modulator extinction ratio is 30dB and the insertion loss, 3dB. The lasers operate at center frequencies of 192.95, 193.05, 193.15 and 193.25THz for channels 1 through 4. The laser linewidth and output power is 10MHz and 0dBm, respectively.

The OA component simulates an EDFA. For a given input power $P_{in}(f)$ the output power of this implementation is evaluated using equation 5.6

$$P_{out}(f) = G(f) \cdot P_{in}(f) + P_{ASE}(f) \tag{5.6}$$

where $G(f)$ is the wavelength dependent amplifier gain and $P_{ASE}(f)$ is the ASE noise power spectrum at the output. The amplifier gain was set at 10dB and the noise figure F was set at 4.5dB. The output ASE power $P_{ASE}(f)$ is calculated for a specific bandwidth using equation 5.7 below

$$P_{ASE} = F(G-1)h\nu\Delta\nu \tag{5.7}$$

where h is Planck's constant and ν the optical frequency. The signal is transmitted over 50km spans of normal dispersion-shifted fiber. The fiber attenuation, PMD and non-linearity coefficient is set at $0.2\,dB/km$, $0.1\,ps/\sqrt{km}$ and $1.8\,W^{-1}km^{-1}$, respectively.

The OXC design is similar to the OXC node architecture illustrated in figure 2.7. The de-multiplexer is implemented using a splitter and Raised Cosine optical filters. The filter exponent, roll-off and bandwidth is set at α=1, 0.2 and 50GHz, respectively. The 2X2 optical switches have a crosstalk value of –90dB and an excess loss of 0.5dB. The multiplexer design is similar to that of the de-multiplexer except a combiner is used instead of a splitter.

The OADM uses a multi-stage Lorentzian optical filter to select the drop channel. The filter bandwidth is set at 50GHz and the crosstalk is –90dB. Channel 3 is the selected add/drop channel. At the receiver stage, the de-multiplexer and receiver are combined into one component. The de-multiplexer design is the same as that used in the OXC module. The receiver device is a PIN photodiode with a 5-pole Bessel post-detection filter. The OptSim design is shown in figure 5.1 below.

The total simulation time is 2048 bits. The optical channel power levels and OSNR are evaluated for each component. The BER is evaluated at the input and output of each component. The BER is evaluated at the output of the OXC, OADM and at the receiver; this is done to check whether the signal integrity is maintained throughout the simulation process. The measured values can be found in tables in appendix B.

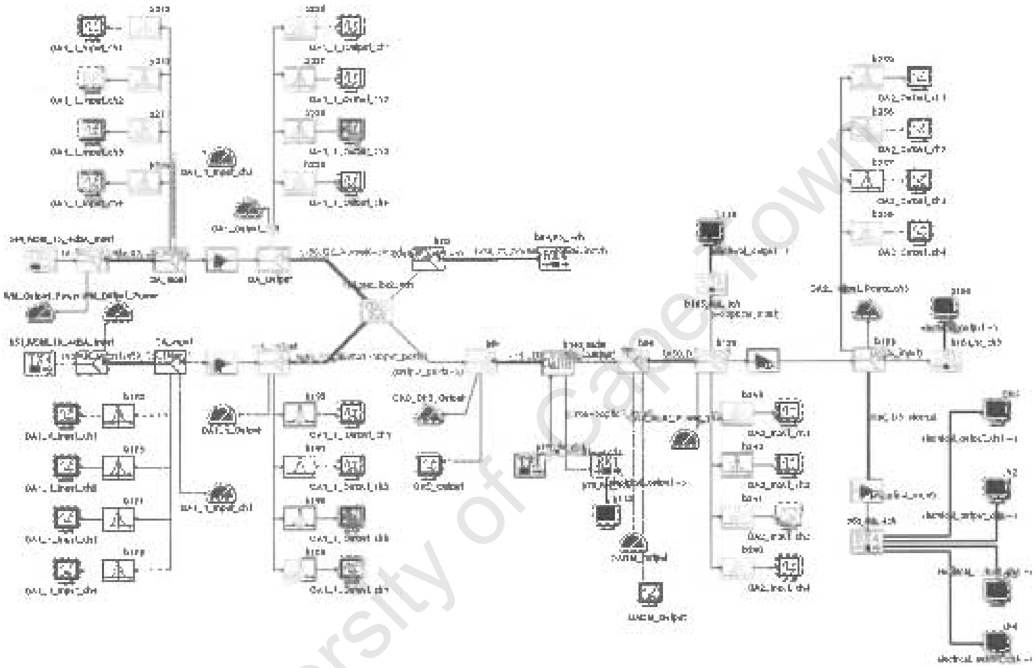


Figure 5.1 OptSim design of point-to-point network example

5.3 Simulation Results

The system was simulated for different random seed values. The output performance data for the NEs are tabulated in Appendix B. The optical spectra at the output of the wavelength multiplexer and the input of the wavelength de-multiplexer are shown in figure 5.2 and 5.3 below. The OSNR is maintained above 40dB throughout the simulation.

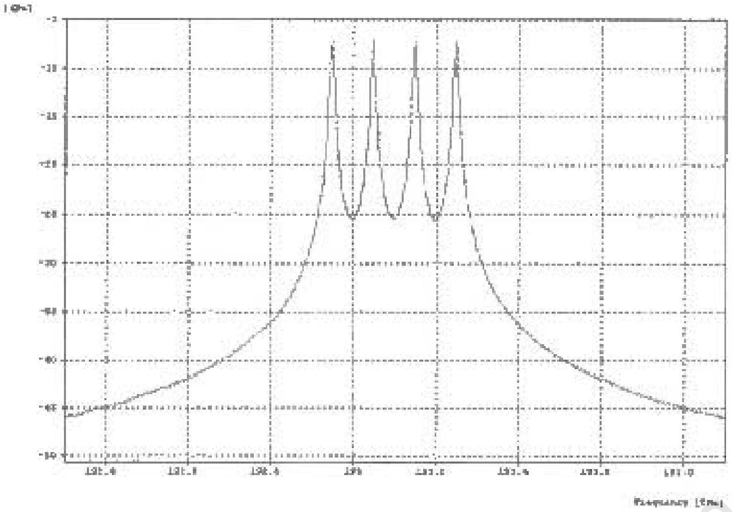


Figure 5.2 Optical sepctrum at the output of the wavelength multiplexer

Figure 5.3 Optical spectra at the input of the wavelength de-multiplexer

Similarly, the eye diagrams at the input to the wavelength multiplexer and at the receiver is shown below.

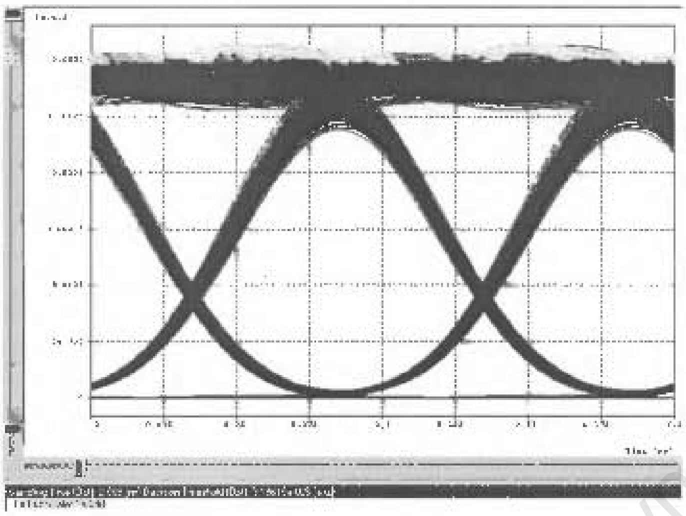


Figure 5.4 Channel 3 eye diagram at the output of the wavelength multiplexer

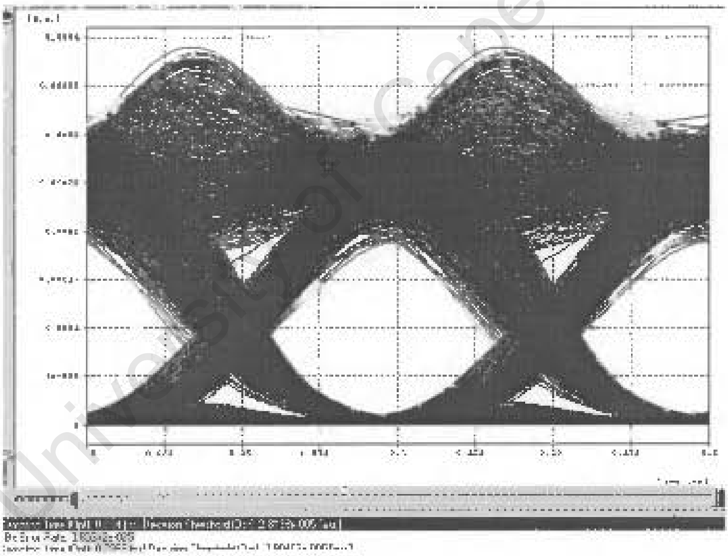


Figure 5.5 Channel 3 eye diagram at the receiver (before O-E conversion)

The BER degrades from 10E-40 at the input to 10E-25 at the output for the express channels. The BER of the add/drop channel degrades from 10E-40 at the input to 10E-16 at the output. This excess degradation can be attributed to the crosstalk at the OADM.

5.4 NM Tool Implementation

This section describes the implementation of an NM tool, which does performance monitoring and fault detection of networks that are logically divided into OAM domains and subnets as per the proposed OAM model architecture in chapter three. The NM tool takes as its input, the performance data of the performance parameters (as identified in chapter three) for each of the NEs. It then generates notifications and alarms as per the performance monitoring and fault detection algorithms, which are discussed hereafter. The output of the NM tool will be the number of notifications reported to and processed by each GNE and the NMS.

In [40], two approaches are identified for implementing the performance monitoring procedure: trend analysis and threshold crossing alarm (TCA) surveillance. "The trend analysis approach looks at all the performance parameter values all the time whereas the TCA surveillance method only considers periods of exceptionally poor performance". In this study, the threshold crossing technique is used to implement the performance monitoring procedure and thus detect degraded performance. Two thresholds are associated with each performance parameter: a degradation and failure threshold. A threshold crossing is reported when any of the monitored performance data reaches or exceeds the threshold values.

There are four stages in the performance monitoring process [40]; only the first stage will be considered in this study. In the first stage the status and behaviour of the NEs as well as the signal are monitored; the NE performs this stage. In real NMSs performance monitoring is done for 15-minute and 24-hour intervals. A threshold crossing alert is generated for each of these intervals if a threshold is reached or exceeded. It is also typical in NMSs to examine performance data over a period of time to ensure that the degraded performance persists, before alerting the next stage of the process. If the persistency check determines that an anomaly does not persist then the TCA is cleared.

The performance-monitoring algorithm that was implemented in this NM tool is illustrated in figures 5.6 and 5.7 below.

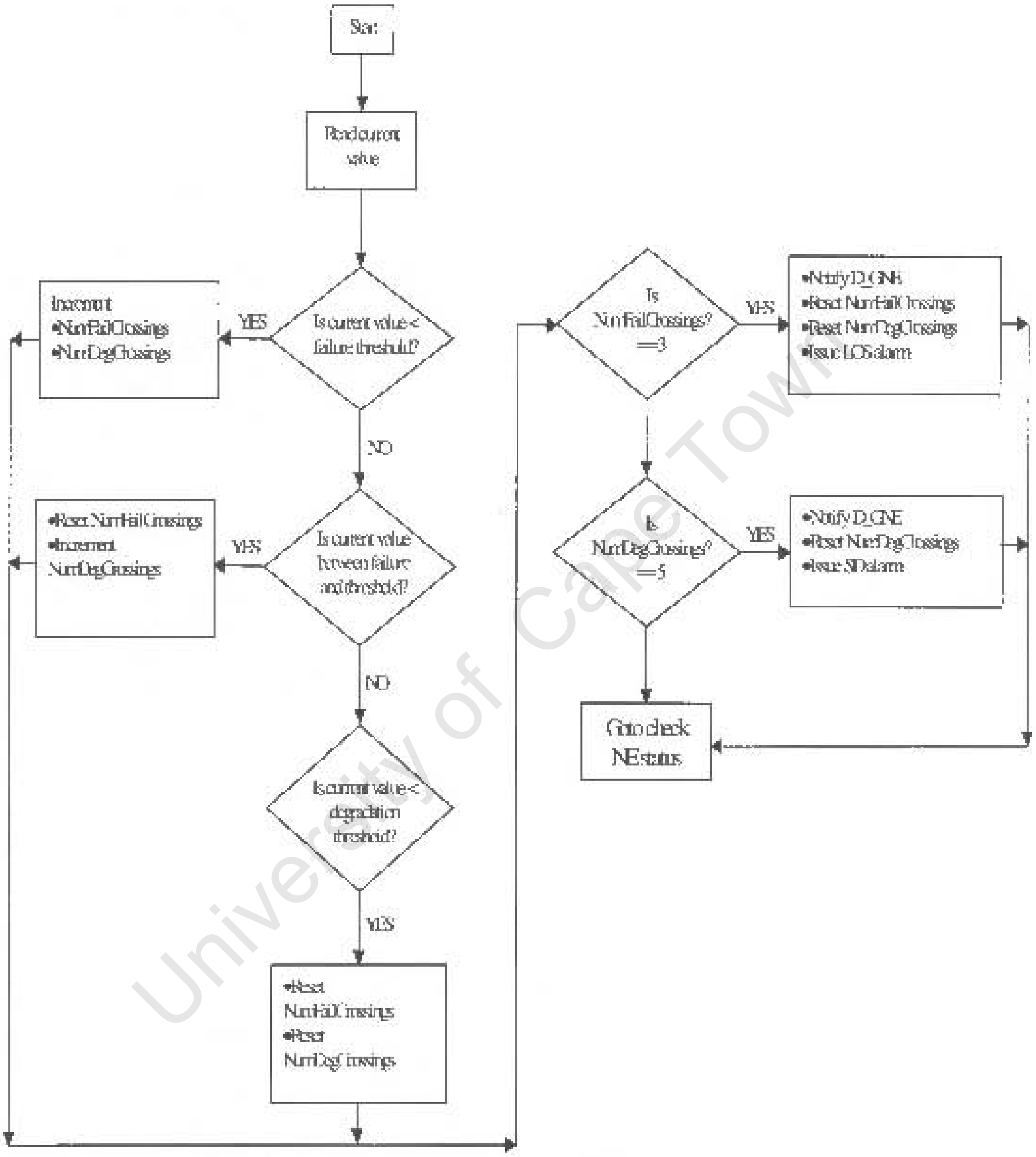


Figure 5.6 Algorithm for checking whether a threshold crossing has occurred

The algorithm illustrated in figure 5.6 works as follows (based on the TCA approach explained above): Each NE compares the current performance data with the threshold value to determine whether a threshold has been reached or exceeded. If a threshold crossing is detected, a flag is set which will be used in successive runs to check whether the threshold crossing condition persists. For serious alarm conditions (e.g. LOS), [40] recommends that notifications and alarms be generated within 3 seconds of the occurrence of the event. In this implementation, a serious alarm condition is signaled if a failure threshold is crossed on three consecutive cycles. In this case, a notification is sent to the NMS via the GNE i.e, no intermediate persistency checking is performed at the subnet GNE. If a degradation threshold crossing occurs for five consecutive cycles, then a notification is sent to the GNE and an alarm is raised. In this case though, the GNE will first check whether the degradation persists before notifying the NMS. If the condition does not persist, the alarm is cleared and no notification is sent to the NMS, otherwise a notification is sent to the NMS.

Once the performance data has been checked for threshold crossings, the GNE will check whether it has received any notifications from NEs in its domain or subnet; this is illustrated by the flow-chart in figure 5.7 below. The algorithm is explained as follows; the GNE checks whether it has received any notifications. If a notification had been received by the GNE, then it firstly determines the cause of the notification. If the notification is as the result of a serious alarm condition, then a notification will be sent immediately to the NMS via the subnet GNE. The GNE also checks whether there are any persisting alarms conditions; non-persistent alarms will be cleared whereas persistent alarm conditions will result in further notifications being sent to the subnet GNE. A similar procedure is followed by the subnet GNE, however, in this case notifications are sent to the NMS.

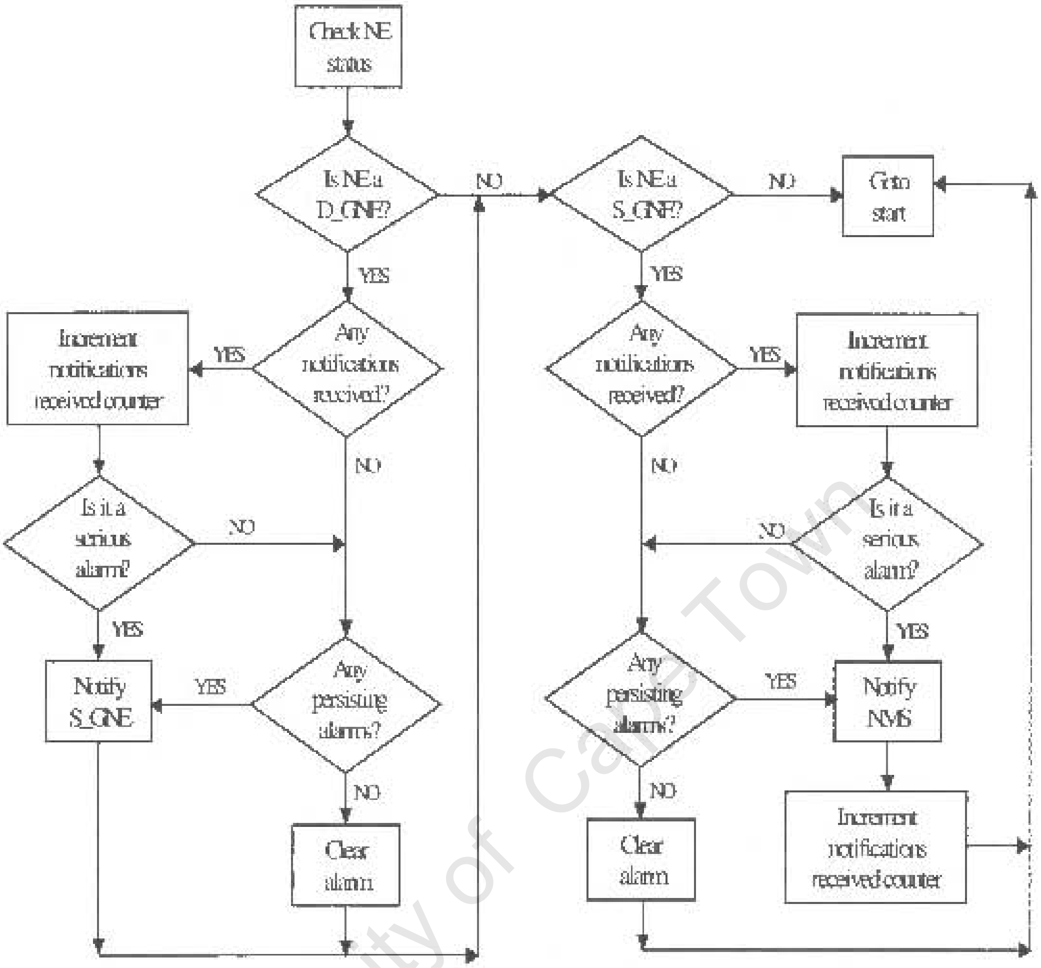


Figure 5.7 Algorithm for checking whether any notifications have been sent by the NEs

The NM tool was implemented in C++. A code listing can be found in appendix C.

5.5 Analysis of Results

The performance data from the point-to-point network was used as the input to the NM tool in an attempt to quantify the usefulness (i.e. advantages to be gained by) of the proposed OAM model. The reason for this is that it is envisaged that a ring network be managed as an OAM domain. So, a comparison of the results using the proposed OAM scheme and the ITU-T recommendation will show no difference. This is because in both cases the ring network will be considered as one domain or subnetwork. However, in the case point-to-point example, the network can be logically divided into two OAM domains, which then form an OAM subnetwork as illustrated in figure 5.8 below. Each domain contains three NEs.

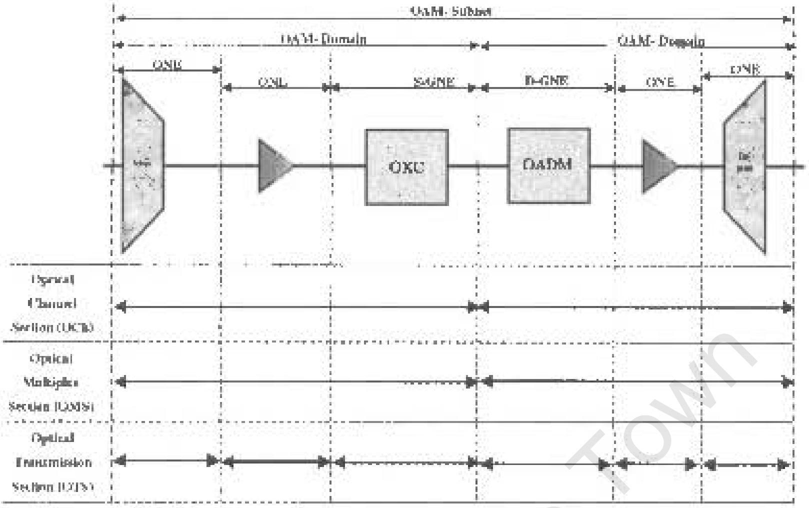


Figure 5.8 Application of the OAM model to a simple point-to-point network with OXC and OADM functionality

The performance data was obtained from the OptSim simulations. Periodic degradations and failures were inserted into the data, which was then input to the NM tool. The results were compared for the case where the point-to-point network is considered as one domain or subnetwork (which will be referred to as the ITU-T approach) and alternatively where it is sub-divided into two domains (which will be referred to as the proposed OAM model approach). Three different scenarios were tested. Firstly, the scenario where no serious or persistent alarm conditions occur are considered. An example of a serious alarm condition is detection of an LOS alarm. For the purpose of this study an alarm will be considered to be persistent if the current value read exceeds the threshold in more than seven consecutive cycles in the case of the proposed OAM model. For the ITU-T case, an alarm is persistent after six consecutive cycles. The reason for this is that the subnetwork GNE performs an intermediate persistency check before notifying the OS. In the ITU-T case the intermediary persistency checking is performed by the OS itself.

Scenario1 - Test 1: Two domains one subnetwork

Overall, seven degradations occurred in the network: four in domain one and three in domain two. Each of these degradations generated a notification to the GNE in its domain. The results are plotted in figures 5.9 below.

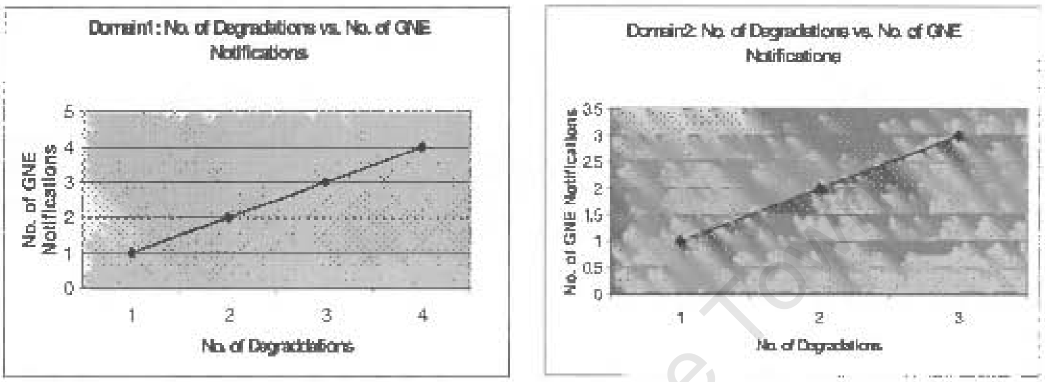


Figure 5.9 Number of GNE notifications generated as a result of degradation threshold crossings (proposed OAM model)

The results suggest that if only non-persistent degradations are considered then there is a linear relationship between the number of degradations that occur and the number of notifications that are sent to the GNE. No notifications were sent to the OS.

Scenario1 - Test 2: One domain or subnetwork

In this case the number of degradations and number of notifications generated by the NEs is just the sum as for test case 1 above. The results obtained are plotted in figure 5.10 below.

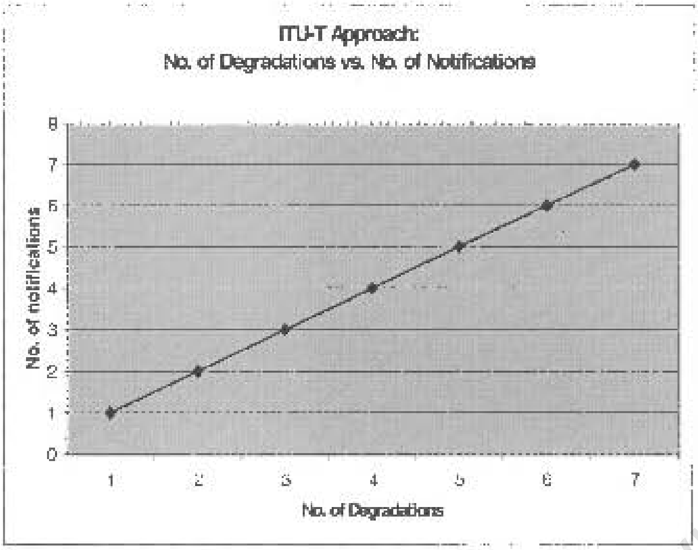


Figure 5.10 Number of GNE notifications generated as a result of degradation threshold crossings (ITU-T Approach)

The results of test 2 also suggest a linear relationship between the number of degradations and the number of notifications to the GNE. The difference between the two approaches is that in this case all the notifications are sent to one GNE, whereas with the proposed model the load is split almost evenly between the two domain-GNEs. It should be noted that this result is only obtained because the periodic degradations occurred in both domains. If all the degradations occurred in one domain, then there is no advantage to be gained from sub-dividing the subnetwork into domains.

Scenario 2 considered the case in which persistent as well as non-persistent degradations occurred. It should be noted that the degradations that occurred are the same, in some cases the periods were merely extended to make them persistent. So, the overall total number of degradations that occurred is still seven: four in domain 1 and three in domain 2.

Scenario 2 - Test 1: Two domains and one subnetwork

The results obtained for the case in which persistent and non-persistent conditions are considered within the proposed OAM framework is plotted in figure 5.11 below.

As noted above, the number of degradation crossings in both domains remain unchanged from the first case. The graphs show that there is no longer a linear relationship between the number of degradations that occur and the number of notifications generated.

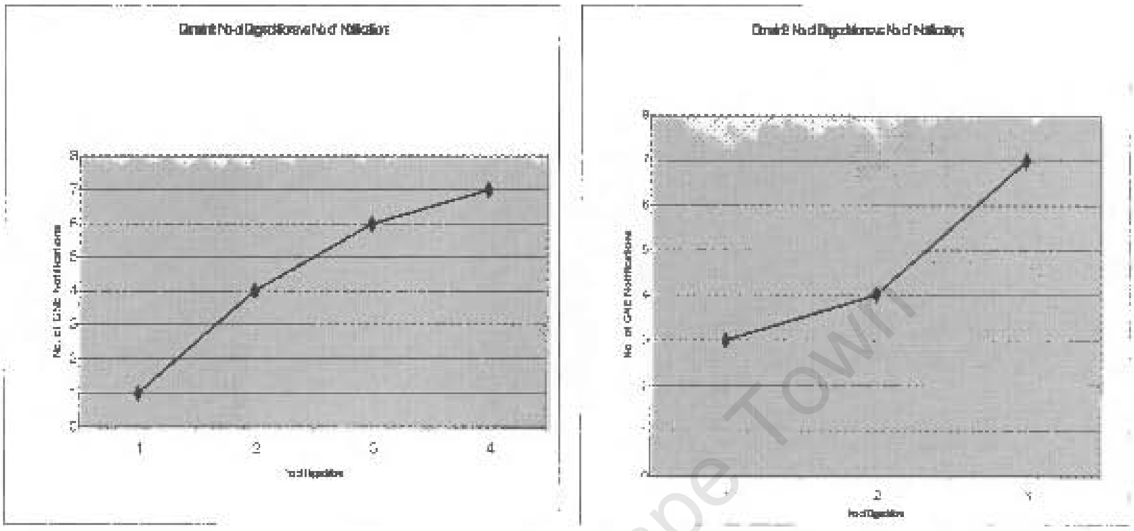


Figure 5.11 Number of notifications generated as a result of degradation threshold crossings (proposed OAM model Approach)

In figure 5.12 below, the results obtained for the two different scenarios are compared for both domains. In both instances the number of notifications generated as a result of including persistent degrading conditions, is increased.

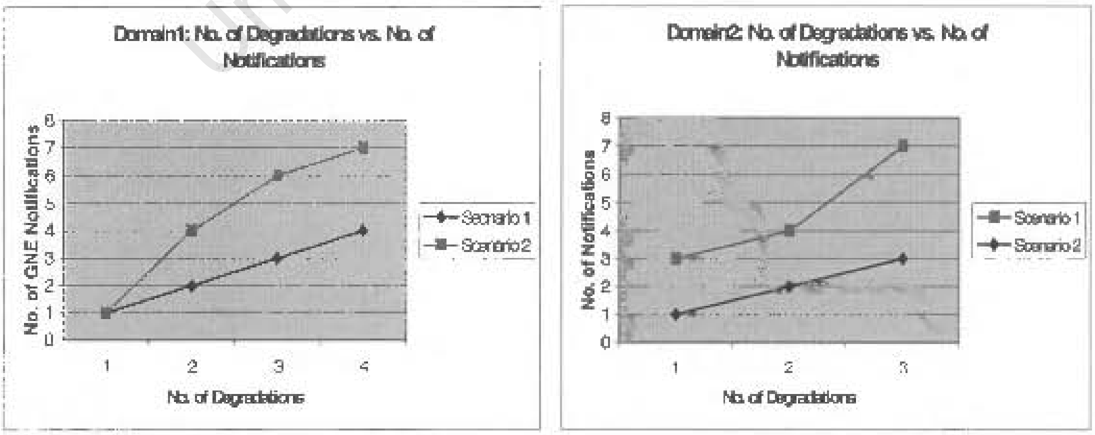


Figure 5.12 Comparison of the number of notifications generated for scenarios 1 and 2

In this scenario, two notifications were sent to the OS.

Scenario 2 - Test 1: One domain or subnetwork

The results obtained for this scenario is plotted in figure 5.13 below. As was the case with scenario 1, when persistent conditions are included in the test data the number of notifications generated in the network increases. Furthermore, there is no longer a linear relationship between the number of degradations that occur and the number of notifications generated. The number of notifications sent to the OS in this case is three.

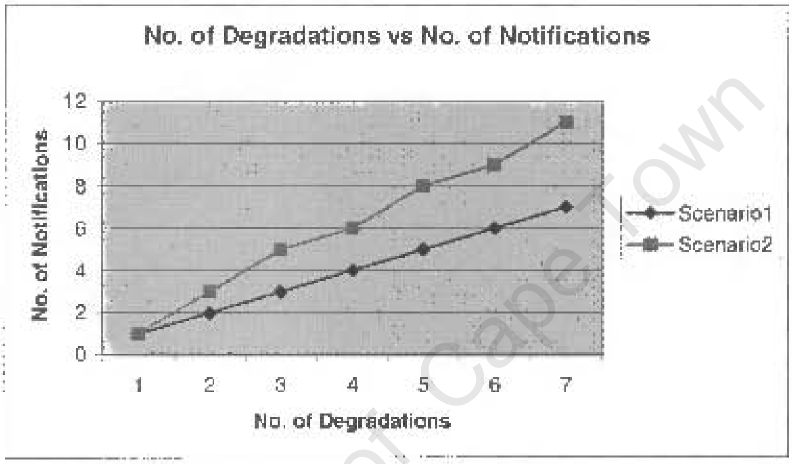


Figure 5.13 Comparison of the number of notifications generated for scenarios 1 and 2

In figure 5.14 below, the overall total number of notifications generated for the two different schemes (ITU-T vs. proposed OAM model approach) are compared. Case 1 refers to the ITU-T approach and case 2, to the proposed model architecture. It is noted that the overall total number of notifications generated in the proposed model is higher than that for the ITU-T case. This can be attributed to the fact that for each persistent alarm condition, an extra notification is sent from the domain GNE to the subnetwork GNE i.e. three notifications per persistent alarm condition for the proposed model case. For the ITU-T case, however, only two notifications are generated as a result of the persistent alarm condition. It should be remembered though that in the proposed scheme, the notifications are processed by two GNEs, whereas in the ITU-T case, only one GNE handles all the notifications. Also, the number of notifications sent to the OS is less for the proposed scheme than for the ITU-T scheme.

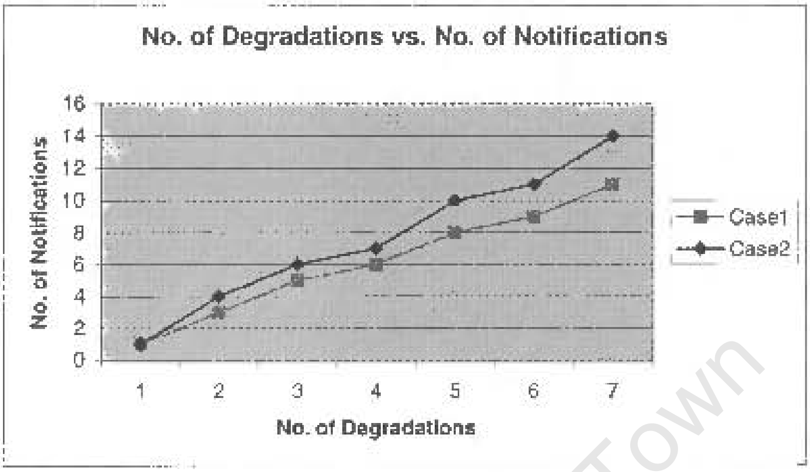


Figure 5.14 Comparison of the overall total number of notifications generated for the two schemes

Scenario 3 considered the case in which persistent, non-persistent as well as serious alarm conditions occurred. The total number of alarms generated in this case increased to ten.

Scenario 3 - Test 1: Two domains and one subnetwork

In this test two serious, two persistent and two non-persistent alarm conditions occurred in domain 1, whereas in domain 2 there was one serious, two persistent and one non-persistent alarms. The test results for the two domains are plotted in figure 5.15 below. As was the case with scenario 2, the relationship between the number of degradations and the number of notifications generated is not linear. This is because in the case of persistent and serious alarm conditions, three notifications are generated per occurrence.

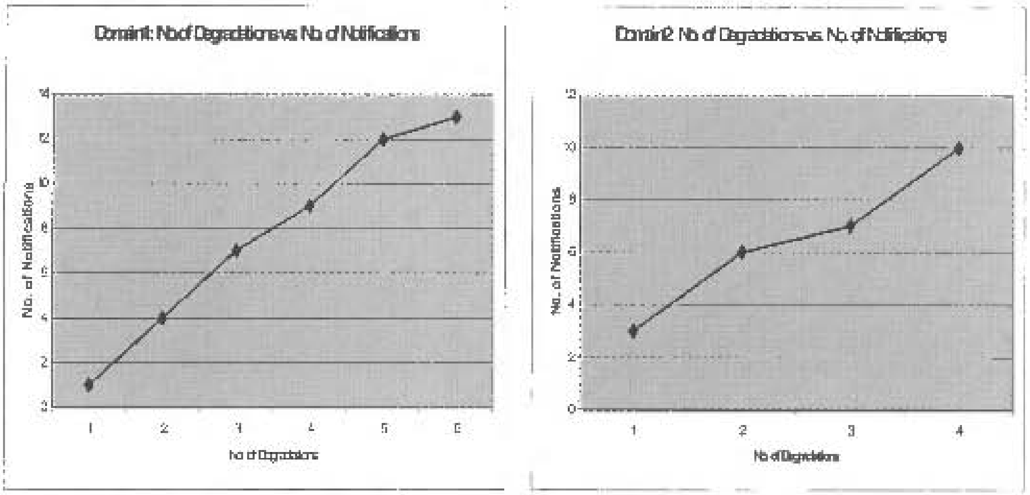


Figure 5.15 Number of notifications generated in domains 1 and 2

Scenario 2 - Test 1: One domain or subnetwork

In this test there were also ten alarms conditions reported. The test results are plotted in figure 5.16 below. The results are compared with the overall total number of notifications, which occur in the two cases.

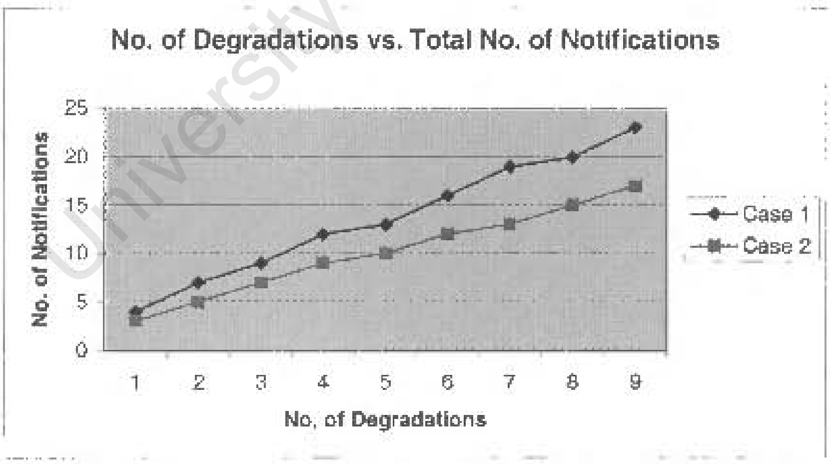


Figure 5.16 Comparison of the number of notifications sent for case 1 and 2

Again, it is clear that the total number of notifications sent in the case of the proposed model exceeds the number sent for the case of the ITU-T approach. However, as pointed out before, in the case of the proposed architecture the notifications are processed by two GNEs. The number of notifications sent to the OS in the case of the proposed scheme is less that for the ITU-T approach.

5.6 Discussions and Summary

In this chapter the design and implementation of the network examples were discussed. Firstly, some background information was given about the optical network simulation environment that was used to simulate the networks namely, OptSim. Thereafter, the network design and implementations was discussed and the simulation results demonstrated.

In an attempt to quantify the advantage to be gained from implementing the OAM model architecture that was proposed in chapter three, a simple NM tool was implemented and three different scenarios were tested. The results were compared for the proposed scheme and the ITU-T approach. It was found that the total number of notifications generated in the proposed scheme exceed the number of notifications generated in the ITU-T scheme. However, in the ITU-T scheme all notifications are processed by one GNE, whereas for the proposed scheme the load is divided between the GNEs. Furthermore, it was found that the number of notifications sent to the OS in the ITU-T scheme exceeds the number of OS-notifications in the proposed scheme.

As was pointed out in chapter three, the number of managed objects in current communication networks are constantly changing as a result of the bursty nature of IP traffic, which has become the dominant source of communication. Therefore, there is some advantage to be gained by implementing an OAM scheme, which relieves or distributes the processing load at the OS and GNEs that results from network failure or degradation conditions.

Chapter 6

Conclusions and Recommendations

The main objective of this thesis was to propose a generic model for operations, administration and maintenance (OAM) at the optical layer. Furthermore, it had to quantify the advantages to be gained by implementing the proposed architecture. OAM is a collection of processes that support the Network Management (NM) function. The overall OAM actions include measuring the signal and NE performance, reporting on the degradation or failure of the signals or NEs and altering the configuration of NEs. In this study only performance monitoring and fault detection was considered.

A point-to-point optical network with OXC and OADM functionality was designed and implemented in the OptSim simulation environment. Only static network components were implemented, the reason being that OptSim does not support re-configurable components and networks. The design and implementation ensures that the signal integrity is maintained throughout the simulation process: the OSNR is maintained above 40dB and the BER below 10E-16. The performance parameters were measured as specified in chapter three.

A generic OAM model, which implements a hierarchical structure and sub-divides OAM subnetworks into OAM domains was proposed. To test the validity of this approach and to quantify the advantages to be gained from implementing the proposed model architecture, a NM tool was implemented. The performance data from the OptSim simulations were used as input to the NM tool. Periodic failures are inserted in the test data before it is input to the NM tool. The results are compared for a network, which is logically sub-divided into two domains (as per the proposed model), and a network that considers the network as one domain or subnetwork (as per the ITU-T recommendations).

Three different scenarios were tested: firstly, the case where only non-persisting degradations occurred. Secondly, the case in which persisting and non-persisting degradations occurred. Finally, all failure and degradation conditions were included. It was found that in the case that no persisting and serious alarm conditions occurs in the network, the number of notifications generated in both cases are the same. However, in the case of the proposed architecture the number of notifications is processed by two GNEs, whereas in the ITU-T case, only one GNE processes the entire load. When persisting. When persisting and serious alarm condition are included in the test data, the number of notifications generated by the proposed scheme exceeds that for the ITU-T case. The advantage of this scheme though is that the notifications are processed by more than GNE thus decreasing the processing load experience GNEs. Furthermore, in the case of the proposed scheme the number of notifications sent to the OS is less than that for the ITU-T case. This is a very important result especially in view of the fact that todays network traffic is very bursty in nature, thus resulting in the number of managed objects constantly changing. The OSs must be able to cope with these changing network conditions without overloading its resources.

Therefore, there is some advantage to be gained by implementing the proposed OAM scheme, which relieves or distributes the processing load at the OS and GNEs that results from network failure or degradation conditions

In this study only static network elements were considered. Also, only the functions relating to performance monitoring and fault detection were considered and implemented. During the course of this research several areas of interest have surfaced that could not be included in the scope of this thesis. The following list outlines possible further work based on this study:

- investigate signaling protocols that can be implemented in the proposed OAM framework;
- investigate the performance of other OAM functions (e.g. configuration) within the proposed framework;
- identify the performance parameters associated with configurable NEs;
- investigate the performance monitoring and fault detection procedures associated with networks implementing re-configurable devices;
- implement a robust NM tool for optical layer management.

References

[1.] B. Mukherjee, "WDM Optical Communication Networks: Progress and Challenges", *IEEE J. Selected Areas in Commun.*, Vol. 18 No. 10 pp1810-1823, October 2000.

[2.] F-X. Ollivier, C. Zugno, S. Thompson, "Evolution of high speed DWDM backbone networks", *Alcatel Commun. Review*, pp181-193, 3rd Quarter 2000.

[3.] H. Zang, L. Sahasrabuddhe, J.P.Jue, S, Ramamurthy, B. Mukherjee, "Connection Management for Wavelength Routed WDM Networks", *Proc. IEEE Globecom '99*, pp1428-1432.

[4] S.Yao, S. Dixit, B. Mukherjee, "Advances in Photonic Packet Switching: An Overview", *IEEE Commun. Mag.*, pp84-94, Feb. 2000.

[5] M.O.Van Deventer et al., "Evolution towards an Optical Network Layer", *EURESCOM Project report*, P615, 1998.

[6] ITU-T Recommendation G.872, "Architecture of Optical Transport Networks", Feb. 1999.

[7] H.J.R. Dutton, "Understanding Optical Communicatios", *Prentice_Hall*, 1998.

[8] J.P. Ryan, "WDM: North American Deployment Trends", *IEEE Commun. Mag.*, pp40-44, Feb. 1998.

[9] M. Borella, J. Jue, D. Banerjee, B. Ramamurthy, B Mukherjee, "Optical Components for WDM Lightwave Networks", *Proceedings of the IEEE*, Vol 85 no. 8, Aug. 1997.

[10] B. Mukherjee, "Optical Communications Networks", *McGraw-Hill*, 1997.

[11] J. Chawki et al., "Planning of Full Optical Networks", *EURESCOM Project report*, P709, 1999.

[12] ITU-T Recommendation G.841, "Types and Characteristics of SDH Network Protection Architectures", Oct. 1998.

[13] J. Manchester, P. Bonenfant, C. Newton, "The Evolution of Transport Network Survivability", *IEEE Commun. Mag.*, pp44-51, Aug. 1999.

[14] L. Kant, H. Kim, D. Hsing, T. Wu, "Modelling and Simulation Study of Survivability Mechanisms in WDM-based High-Speed Networks", *IEEE Globecom 1999*, pp1028-1034.

[15] D. Schupke, "Reliability Models for WDM Self-Healing Rings", www.lkn.ei.tun.de/~ds/papers/final_corrected.pdf

[16] M. Medard, S. Finn, R. Barry, "WDM Loop-Back Recovery in WDM Mesh Networks", *Infocomm*, pp752-59, Mar 1999.

[17] R. Tucker, W. Zhong, "Photonic Packet Switching: An Overview", *IEICE Trans. on Commun.*, pp254-264, Feb. 1999.

[18] G. Ferraris, K. Oguchi, "Management of Optical Networks", *ECOC 1998*, pp695-696.

[19] D. Udupa, "TMN: Telecommunications Management Network"*McGraw-Hill*, 1999.

[20] R. Wagner, R. Alferness, A. Saleh, M. Goodman, " MONET: Multiwavelength Optical Networking", *J. of Lightwave Technology*, Vol. 14 No. 6, pp1349-1355, June 1996.

[21] R. Wagner, "Realizing the Vision of Multiwavelength Optical Networking", *IEICE Trans. On Electron.*, Vol. E81-C No.8, pp1159-1166, Aug. 1998.

[22] M. Lehdorfer et al., "AC084 PHOTON: Paneuropean Photonic Transport Overlay Network", http://www.infowin.org/acts/rus/projects/ac084.htm

[23] A. Mcguire, "Management of Optical Transport Networks", *Electron. & Commun. Engineering Journal*, pp155-163, Jun 1999.

[24] S. Tomic, "Management of the WDM all-optical network", *Proc. Of SPIE*, Vol. 3230, pp114-124, 1997.

[25] O. Gerstel, "Optical Layer Signalling: How Much is Really Needed?", *IEEE Commun. Mag.*, pp154-160, Oct. 2000.

[26] M. Berger et al., "Pan-European Optical Networking using Wavelength Division Multiplexing", *IEEE Commun. Mag.*, pp82-88, Apr 1997.

[27] ITU-T Draft Recommendation G.874, "Management Aspects of the Optical Transport Network Element", Feb. 2001.

[28] I. Monroy, E. Tangdiongga, "Performance Evaluation of Optical Cross-Connects by Saddlepoint Approximation", *IEEE J. of Lightwave Tech.*, pp317-323, Mar 1998.

[29] H. Dorren, H. De Waardt, I. Monroy, "Statistical Analysis of Crosstalk Accumulation in WDM Networks", *IEEE J. of Lightwave Tech.*, pp2425-2430, Dec. 1999.

[30] I. Monroy, "Performance Evaluation of Optical Communication Networks", Phd thesis, Sept. 1999.

[31] ITU-T Recommendation M.3010, "Principles for a Telecommunications Management Network", May 1996.

[32] ITU-T Recommendation M.3400, "TMN – Management Functions", Feb. 2000.

[33] RFC 1155

[34] RFC 1156

[35] RFC 1157

[36] H. Foisel et al., "Integration of IP over Optical Networks: Networking and Management", *EURESCOM Project report*, P918, Sept. 2000.

[37] J.Hayes, "A Practical Guide to Testing Fiber Optic Components and Networks", *Fotec's Fiber Optic Testing Textbook*, http://www.fotec.com/testbook/book3.htm

[38] M. Kasima, N. Minato, S. Kutsuzawa, S. Oshiba, "A Realistically Architecture of WDM Ring Network Using OXC and OADM", *IEICE Trans. On Commun.*, pp2253-9, Oct. 2000.

[39] Artis Software Corporation, "OptSim™ 3.0 User Manual", July 2000.

[40] ITU-T Recommendation M.2140, "Transport Network Event Correlation", Feb. 2000.

[41] N. Wada, K. Kitayama, "10Gbps Optical Code Division Multiplexing Using Chip Optical Bipolar Code and Coherent Detection", *IEEE J. of Lightwave Tech.*, pp1758-65, Oct. 1999.

[42] K. Kitayama, "Code Division Multiplexing Lightwave Networks Based upon Optical Code Conversion", *IEEE J. on Selected Areas on Commun.*, pp1309-19, Sept. 1998.

[43] A. Viswanathan, "Evolution of Multiprotocol Label Switching ", *IEEE Commun. Mag.*, pp165-73, May 1998.

[44] S. Verma, H. Chaskar, R. Ravikanth, "Optical Burst Switching: A Viable Solution for Terabit IP Backbone", *IEEE Network*, pp48-53, Dec 2000.

[45] C.Qiao, M. Yoo, "Optical Burst Switching (OBS) – A New Paradigm for an Optical Internet", *J. of High Speed Networks*, pp79-90, 1999.

[46] D. Hunter, M. Chia, I. Andonovic, "Buffering in Optical Packet Switches", *IEEE J. Lightwave Tech.*, pp2081-94, Dec. 1998.

[47] F. Callegati, H. Canyaka, Y. Xiong, M. Vandenhoute, "Design Issues of Optical IP Routers for Internet Backbone Applications", *IEEE Commun. Mag.*, pp124-28, Dec. 1999.

[48] D. Hunter, I. Andonovic, "Approaches to Optical Internet Packet Switching", *IEEE Commun. Mag.*, pp116-22, Sept. 2000.

[49] A. Stok, E. Sargent, "Lighting the Local Area: Optical Code Division Multiple Access and Quality of Service Provisioning", *IEEE Network*, pp42-46, Nov/Dec 2000.

[50] K. Kitayama, "Code Division Multiplexing Lightwave Networks Based on Optical Code Conversion", *J. Select Areas on Commun.*, pp1309-19, Sept. 1998.

[51] N. Wada, K. Kitayama, "A 10Gbps Optical Code Division Multiplexer using 8-chip Optical Bipolar Code and Coherent Detection", *IEEE J. Lightwave Tech.*, pp1758-65, Oct. 1999.

[52] D. Cotter, J. Lucek, D. Marcenac, "Ultra-High-Bit-Rate Networking: From the Transcontinental Backbone to the Desktop", *IEEE Commun. Mag.*, pp90-95, Apr. 1997.

[53] G. Mohan, C. Murthy, "Lightpath restoration in WDM Optical Networks", *IEEE Network*, pp24-32, Dec 2000.

[54] D. Zhou, S. Subramaniam, "Survivability in Optical Networks", *IEEE Network*, pp16-23, Dec. 2000.

[55] S. Kuroyanagi, T. Nishi, "Optical Path Restoration Schemes and Cross-Connect architectures for Photonic Transport Networks", *IEEE Globecom '98*, pp2282-88, Nov. 1998.

[56] P. Green, "Progress in Optical Networking", *IEEE Commun. Mag.*, pp54-61, Jan. 2001.

[57] K.Hattori, M. Fukui, M. Jinno, M. Oguma, K. Oguchi, "PLC-Based Optical Add/Drop Switch with Automatic Level Control", *IEEE J. Lightwave Tech.*, pp2562-71, Dec. 1999.

[58] S. Ramamurthy, B. Mukherjee, "Survivable WDM Mesh Networks: Part II - Protection", *Proceedings of the ICC*, pp2023-30, 1999.

Appendix A

Wavelength (De-) Multiplexing Demonstration

The objective is to demonstrate the wavelength (de-)multiplexing functionality. An eight-channel multiplexer was implemented using ideal combiners, which combine the wavelength channels onto a single fiber. The OptSim design is shown in figure A.1 below.

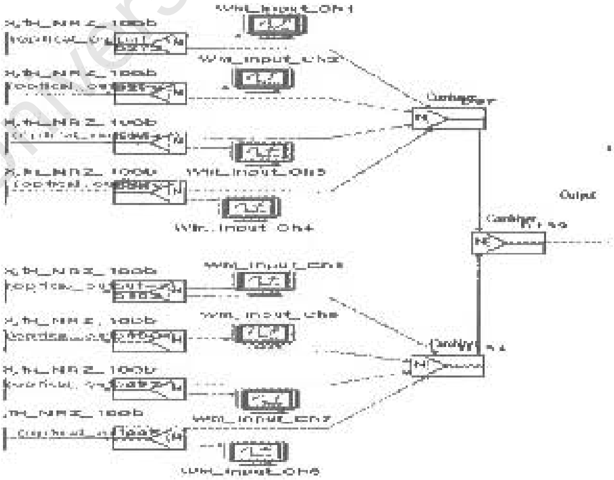


Figure A.1 OptSim design of a wavelength multiplexer

The VBS bandwidth for the test is set to 1.25THz and the SPT bandwidth, to 1THz. The center frequency of the multiplexer is at 193.1THz and the channels are spaced at

0.1nm (i.e. 100GHz).Each channel is de-multiplexed by means of a first order Raised Cosine filter. The filter bandwidth is 50THz and the roll-off factor (R_0) is 0.2. The OptSim design of the wavelength de-multiplexer is shown in figure A.2 below.

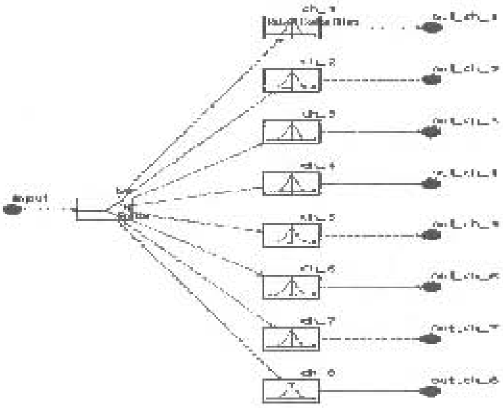


Figure A.2 OptSim design of a wavelength de-multiplexer

The input to the multiplexer is eight NRZ signals. The transmitter is implemented using a continuous wave laser, which is externally modulated by a Mach Zehnder modulator; the bit rate is 10Gbps. At the output of the de-multiplexer the signal is detected using a PIN photodiode, which is then low pass filtered using a 6-pole Bessel filter. The wavelength multiplexer and de-multiplexer are connected in a back-to-back configuration. The reason for this is that the functionality of the multiplexing and de-multiplexing operation is to be examined without the added effect of transmission over fiber. The optical power spectrum and OSNR is evaluated at the input and output of the multiplexer. At the output of the de-multiplexer, the channel performance is evaluated using the optical spectra, eye diagrams and the measured BER and Q-factor. The OptSim test model that was used is shown in figure A.3 below.

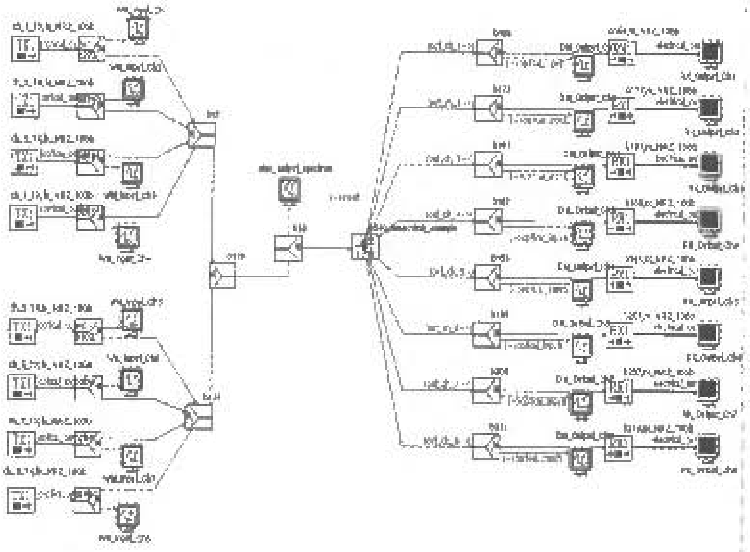


Figure A.3 Back-to-back wavelength multiplexing/de-multiplexing

Firstly, to demonstrate the multiplexing operation the optical spectra at the input and output of the multiplexer as well as the spectra at the output of the de-multiplexer was measured. The spectral diagrams are shown in figures A.4-13 below.

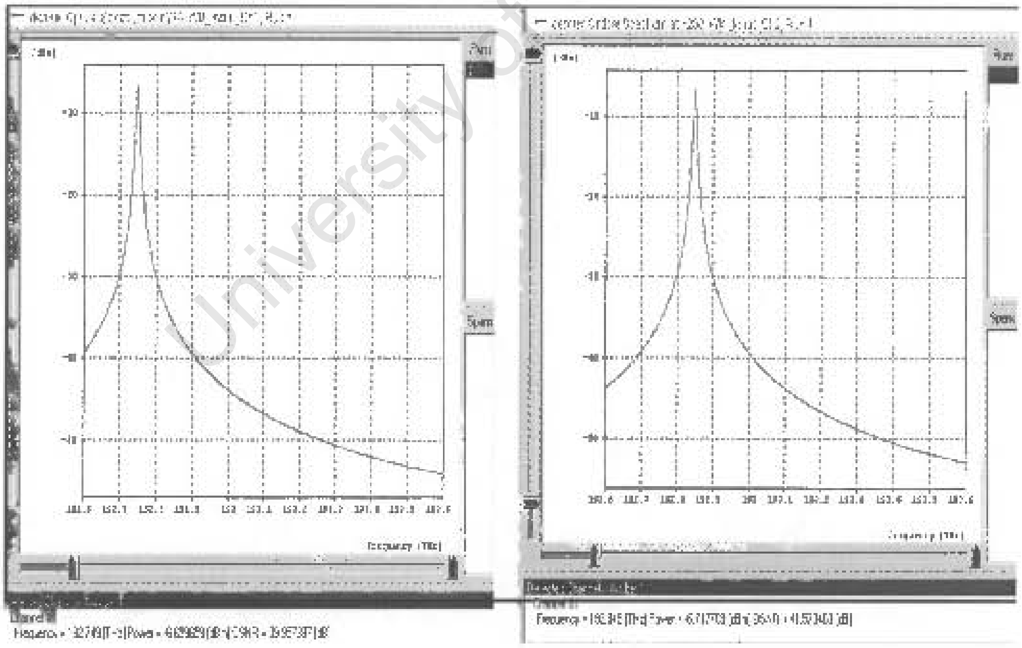


Figure A.4 Wavelength multiplexer input: Channels 1 and 5

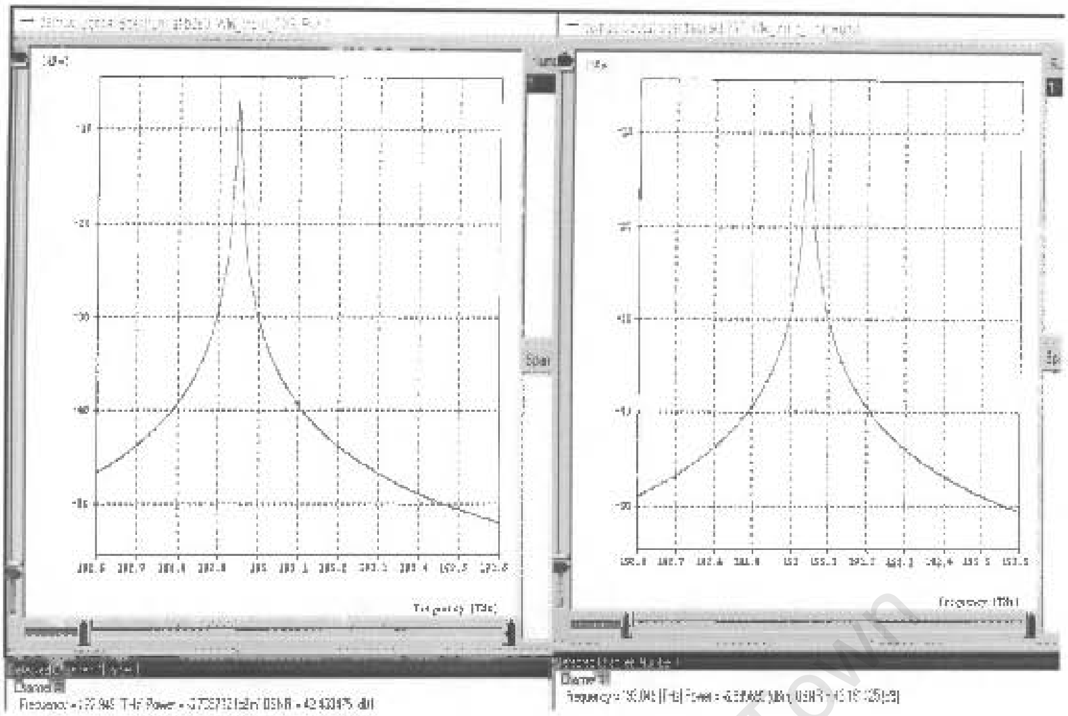


Figure A.5 Wavelength multiplexer input: Channels 3 and 4

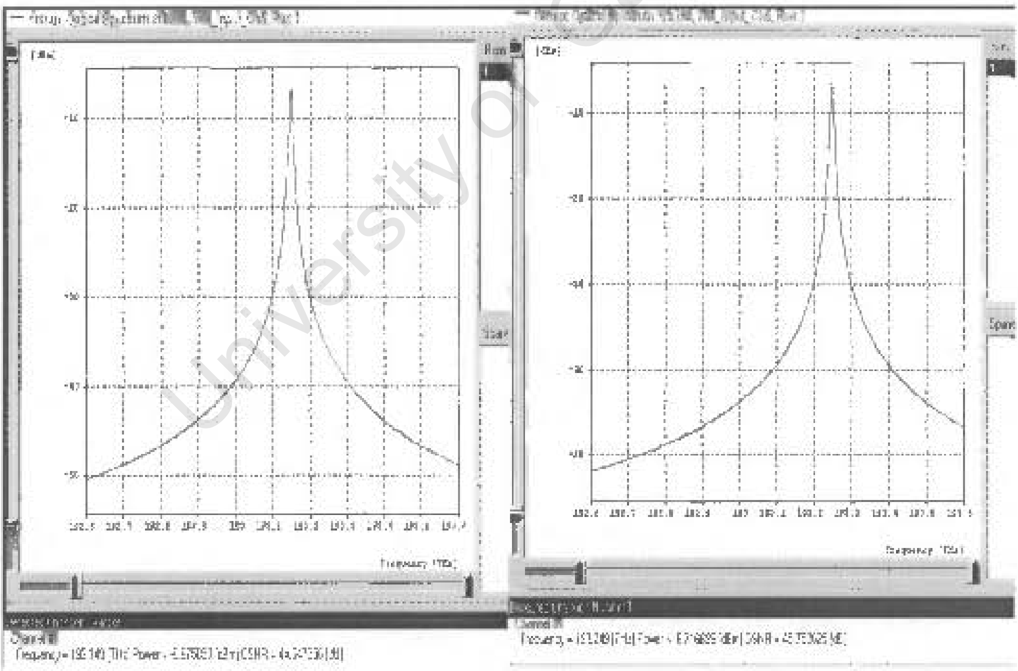


Figure A.6 Wavelength multiplexer input: Channels 5 and 6

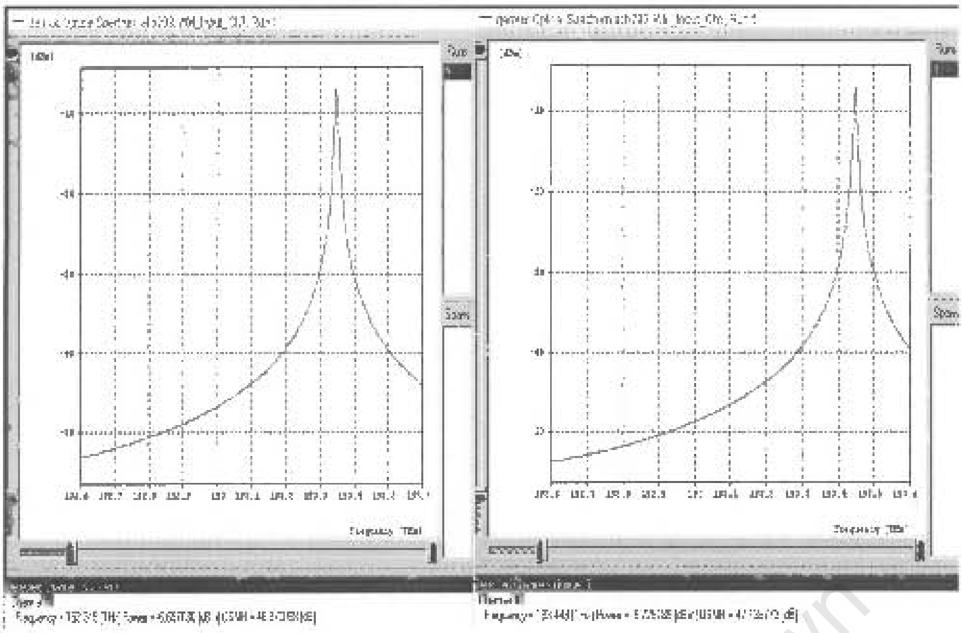
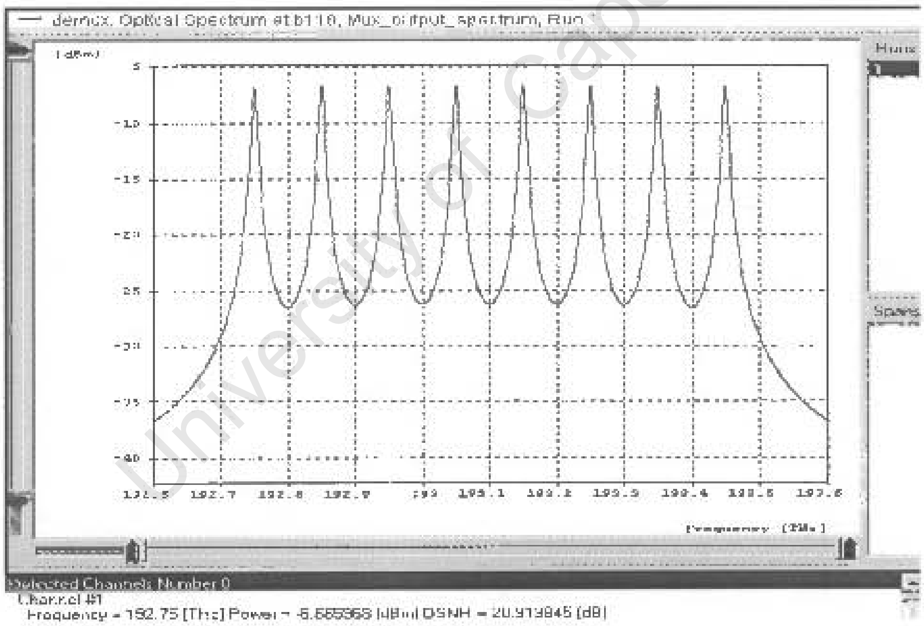


Figure A.7 Wavelength multiplexer input: Channels 7 and 8



```
Selected Channels Number 0
Channel #1
Frequency = 192.75 [Thz] Power = -6.665368 [dBm] OSNR = 20.919845 [dB]
```

Figure A.8 Wavelength multiplexer output

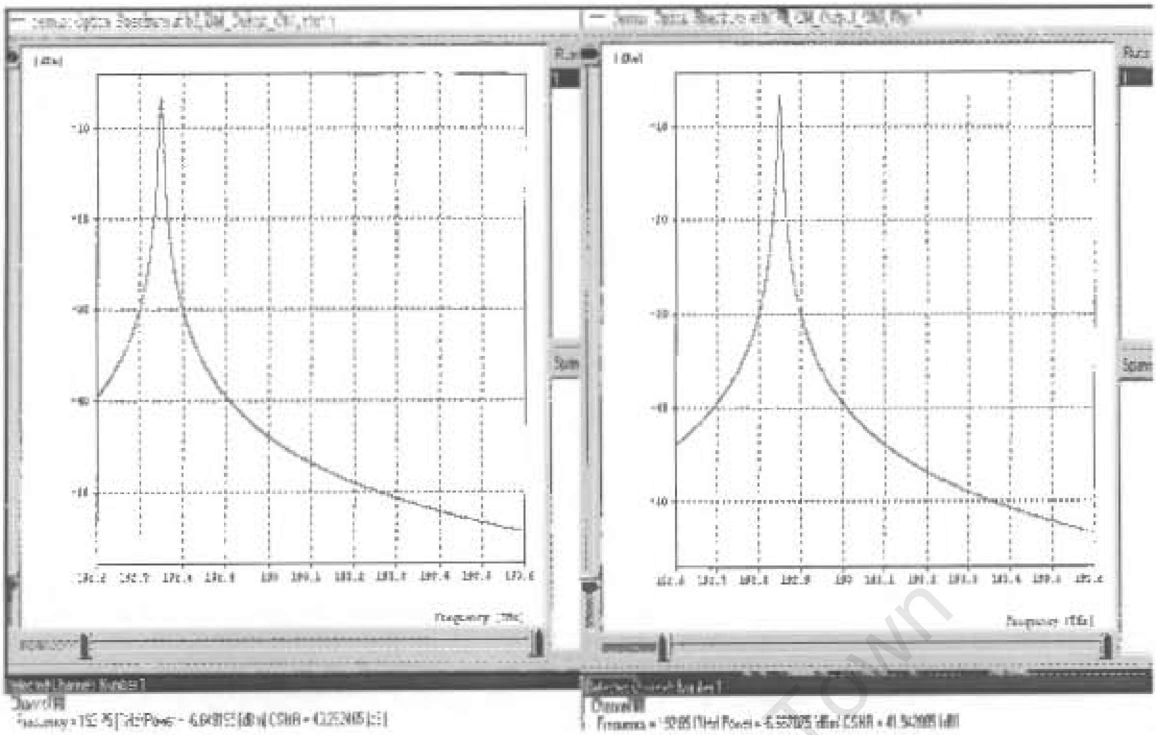


Figure A.9 De-multiplexer output: Channels 1 and 2

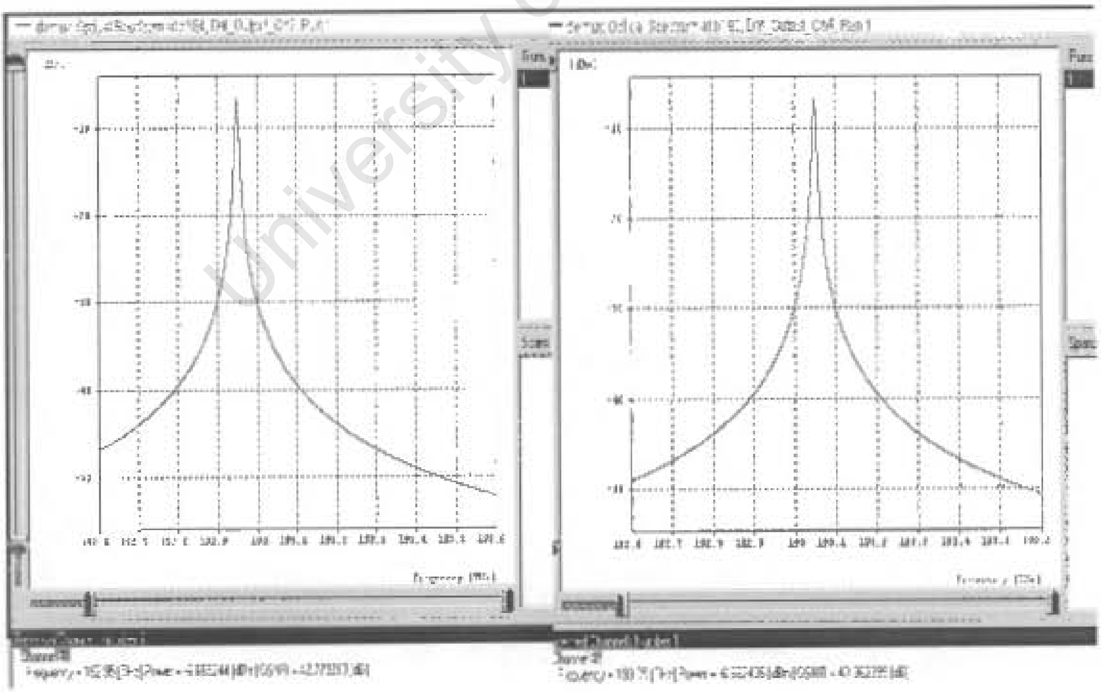


Figure A.10 De-multiplexer output: Channels 3 and 4

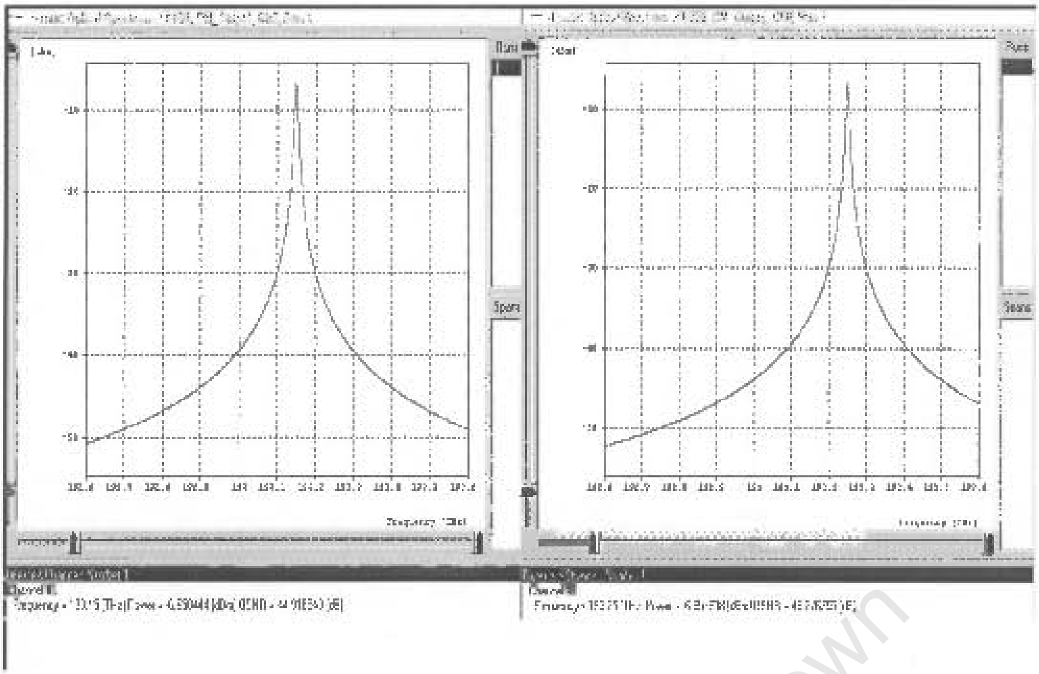


Figure A.10 De-multiplexer output: Channels 5 and 6

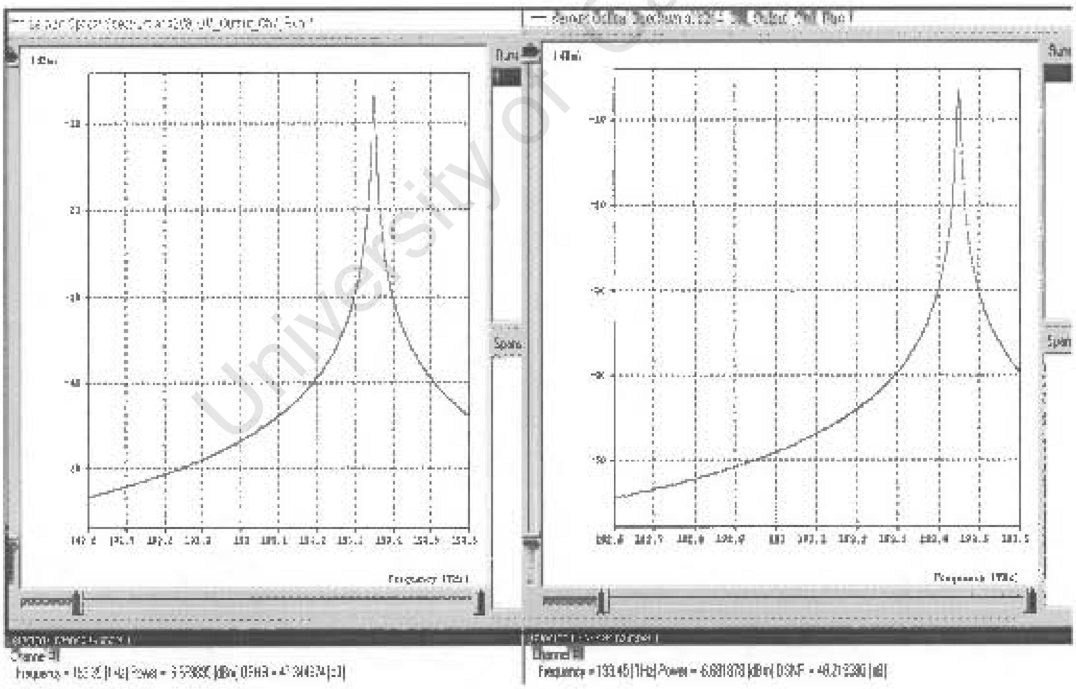


Figure A.11 De-multiplexer output: Channels 7 and 8

Appendix B

Simulation Results for Point-to-Point Network

Performance data for the wavelength multiplexer (WM1)

Output combined power										
	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
	0.038	0.043	0.054	0.053	0.059	0.044	0.045	0.054	0.053	0.064
	Run 11	Run 12	Run 13	Run 14	Run 15	Run 16	Run 17	Run 18	Run 19	Run 20
	0.047	0.054	0.049	0.045	0.054	0.051	0.047	0.057	0.055	0.053
	Run 21	Run 22	Run 23	Run 24	Run 25	Run 26	Run 27	Run 28	Run 29	Run 30
	0.054	0.056	0.059	0.048	0.047	0.053	0.048	0.041	0.038	0.044
	Run 31	Run 32	Run 33	Run 34	Run 35	Run 36	Run 37	Run 38	Run 39	Run 40
	0.043	0.049	0.047	0.045	0.047	0.049	0.043	0.048	0.036	0.033
	Run 41	Run 42	Run 43	Run 44	Run 45	Run 46	Run 47	Run 48	Run 49	Run 50
	0.051	0.045	0.047	0.042	0.046	0.043	0.039	0.038	0.051	0.047

Wavelength Mux										
Input channel power										
	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
ch 1	-7.107	-7.107	-7.146	-7.126	-7.054	-7.165	-7.106	-7.101	-7.154	-7.073
ch 2	-7.069	-7.128	-7.124	-7.141	-7.11	-7.085	-7.1	-7.132	-7.144	-7.13
ch 3	-7.11	-7.155	-7.098	-7.082	-7.143	-7.06	-7.126	-7.128	-7.055	-7.14
ch 4	-7.124	-7.023	-7.044	-7.04	-7.068	-7.059	-7.107	-7.083	-7.108	-7.109
	Run 11	Run 12	Run 13	Run 14	Run 15	Run 16	Run 17	Run 18	Run 19	Run 20
ch 1	-7.072	-7.129	-7.135	-7.114	-7.173	-7.104	-7.115	-7.132	-7.127	-7.082
ch 2	-7.084	-7.058	-7.107	-7.148	-7.14	-7.075	-7.059	-7.089	-7.133	-7.154
ch 3	-7.171	-7.107	-7.147	-7.111	-7.087	-7.141	-7.065	-7.12	-7.136	-7.073
ch 4	-7.136	-7.128	-7.138	-7.143	-7.141	-7.148	-7.125	-7.142	-7.129	-7.1
	Run 21	Run 22	Run 23	Run 24	Run 25	Run 26	Run 27	Run 28	Run 29	Run 30
ch 1	-7.138	-7.107	-7.074	-7.167	-7.104	-7.106	-7.137	-7.138	-7.093	-7.169
ch 2	-7.095	-7.043	-7.068	-7.174	-7.147	-7.112	-7.047	-7.059	-7.137	-7.154
ch 3	-7.146	-7.133	-7.076	-7.138	-7.09	-7.094	-7.142	-7.097	-7.089	-7.142
ch 4	-7.099	-7.061	-7.065	-7.049	-7.071	-7.028	-7.117	-7.066	-7.114	-7.133
	Run 31	Run 32	Run 33	Run 34	Run 35	Run 36	Run 37	Run 38	Run 39	Run 40
ch 1	-7.091	-7.079	-7.135	-7.079	-7.089	-7.152	-7.1	-7.089	-7.164	-7.139
ch 2	-7.149	-7.113	-7.064	-7.133	-7.108	-7.133	-7.129	-7.08	-7.1	-7.129
ch 3	-7.116	-7.135	-7.129	-7.063	-7.155	-7.119	-7.121	-7.164	-7.121	-7.145
ch 4	-7.135	-7.146	-7.149	-7.145	-7.152	-7.151	-7.146	-7.165	-7.144	-7.147
	Run 41	Run 42	Run 43	Run 44	Run 45	Run 46	Run 47	Run 48	Run 49	Run 50
ch 1	-7.09	-7.135	-7.096	-7.1	-7.144	-7.122	-7.105	-7.163	-7.119	-7.074
ch 2	-7.157	-7.136	-7.094	-7.061	-7.106	-7.173	-7.164	-7.098	-7.046	-7.099
ch 3	-7.132	-7.067	-7.132	-7.146	-7.066	-7.124	-7.148	-7.07	-7.158	-7.117
ch 4	-7.126	-7.117	-7.105	-7.132	-7.07	-7.091	-7.061	-7.083	-7.042	-7.109
Input channel OSNR										
	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
ch 1	45.051	44.988	45.009	45.054	45.034	44.965	44.935	45.127	44.984	45.047
ch 2	45.524	45.514	45.546	45.566	45.463	45.429	45.457	45.468	45.362	45.39
ch 3	46.403	46.265	46.409	46.29	46.219	46.319	46.489	46.46	46.461	46.122
ch 4	46.713	46.726	46.709	46.715	47.065	47.078	46.708	46.724	46.701	46.979
	Run 11	Run 12	Run 13	Run 14	Run 15	Run 16	Run 17	Run 18	Run 19	Run 20
ch 1	44.957	44.911	44.94	45.085	44.938	45.09	44.946	44.991	45.008	45.044
ch 2	45.854	45.765	45.707	45.604	45.594	45.667	45.827	45.763	45.728	45.737
ch 3	46.148	46.07	46.495	46.613	46.237	46.104	46.288	46.457	46.346	46.31
ch 4	46.958	47.109	47.098	47.1	46.813	46.82	46.629	46.616	46.615	46.814
	Run 21	Run 22	Run 23	Run 24	Run 25	Run 26	Run 27	Run 28	Run 29	Run 30
ch 1	45.017	45.079	44.936	45.023	45.071	44.943	45.028	45.001	45.003	45.026
ch 2	45.831	45.324	45.467	45.418	45.49	45.603	45.624	45.533	45.474	45.448
ch 3	46.297	46.277	46.315	46.192	46.271	46.206	46.466	46.506	46.542	46.17
ch 4	46.816	47.195	47.184	47.213	47.167	47.188	46.721	46.742	46.734	47.034
	Run 31	Run 32	Run 33	Run 34	Run 35	Run 36	Run 37	Run 38	Run 39	Run 40
ch 1	45.009	44.825	44.958	44.919	44.979	44.927	44.988	45.029	44.93	44.879
ch 2	45.445	45.475	45.669	45.644	45.669	45.532	45.526	45.759	45.746	45.736
ch 3	46.138	46.508	46.537	46.635	46.138	46.13	46.275	46.424	46.354	46.299
ch 4	47.046	46.69	46.693	46.682	46.682	46.677	46.982	46.956	46.988	47.064
	Run 41	Run 42	Run 43	Run 44	Run 45	Run 46	Run 47	Run 48	Run 49	Run 50
ch 1	45.104	44.973	45.001	44.983	44.987	44.925	45.087	44.974	44.928	45.129
ch 2	45.723	45.747	45.463	45.564	45.54	45.537	45.579	45.495	45.436	45.492
ch 3	46.276	46.294	46.26	46.196	46.381	46.504	46.453	46.606	46.147	46.155
ch 4	47.087	46.872	46.87	46.864	46.754	46.729	46.872	46.867	46.878	47.12

Performance data for the optical amplifier (OA1)

Optical Amplifier 1										
Input channel power										
	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
ch 1	-17.055	-17.157	-17.074	-17.079	-17.131	-17.102	-17.111	-17.173	-17.099	-17.08
ch 2	-17.097	-17.09	-17.13	-17.133	-17.137	-17.093	-17.086	-17.107	-17.128	-17.127
ch 3	-17.079	-17.127	-17128	-17.053	-17.134	-17.126	-17061	-17.139	-17.106	-17.099
ch 4	-17.171	-17.132	-17.144	-17.134	-17.141	-17.125	-17.108	-17.105	-17.106	-17.064
	Run 11	Run 12	Run 13	Run 14	Run 15	Run 16	Run 17	Run 18	Run 19	Run 20
ch 1	-17.148	-17.08	-17.125	-17.157	-17.129	-17.102	-17.156	-17.099	-17.097	-17.118
ch 2	-17.08	-17.053	-17.084	-17.133	-17.132	-17.093	-17.062	-17.083	-17.135	-17.152
ch 3	-17.166	-17.09	-17.09	-17.156	-17.052	-17.122	-17.121	-17.046	-17139	-17.126
ch 4	-17.077	-17.047	-17.051	-17.046	-17.064	-17.071	-17.106	-17.112	-17.103	-17.126
	Run 21	Run 22	Run 23	Run 24	Run 25	Run 26	Run 27	Run 28	Run 29	Run 30
ch 1	-17.103	-17.092	-17.123	-17.095	-17.049	-17.161	-17.059	-17.103	-17.15	-17.113
ch 2	-17.104	-17.064	-17.066	-17.153	-17.157	-17.133	-17.063	-17.09	-17.103	-17.154
ch 3	-17.092	-17.139	-17.095	-17.108	-17.135	-17.041	-17.108	-17.147	-17.038	-17.131
ch 4	-17.139	-17.135	-17.141	-17.136	-17.167	-17.118	-17.157	-17.108	-17.154	-17.124
	Run 31	Run 32	Run 33	Run 34	Run 35	Run 36	Run 37	Run 38	Run 39	Run 40
ch 1	-17.11	-17.149	-17.076	-17.1	-17.176	-17.102	-17.082	-17.135	-17.122	-17.11
ch 2	-17.133	-17.112	-17.097	-17.099	-17.11	-17.139	-17.148	-17.074	-17.088	-17.127
ch 3	-17.149	-17.085	-17.085	-17.109	-17.117	-17.132	-17.087	-17.126	-17.17	-17.098
ch 4	-17.131	-17.089	-17.093	-17.081	-17.074	-17.078	-17.079	-17.1	-17.086	-17.128
	Run 41	Run 42	Run 43	Run 44	Run 45	Run 46	Run 47	Run 48	Run 49	Run 50
ch 1	-17.139	-17.086	-17.072	-17.144	-17.079	-17.097	-17.161	-17.111	-17.068	-17.142
ch 2	-17.145	-17.136	-17.103	-17.038	-17.087	-17.169	-17.169	-17.136	-17.044	-17.062
ch 3	-17.115	-17.131	-17.081	-17.169	-17.146	-17.078	-17.15	-17.072	-17.144	-17.17
ch 4	-17.119	-17.167	-17.146	-17.166	-17.15	-17.164	-17.137	-17.162	-17.124	-17.141
Input channel OSNR										
	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
ch 1	46.453	46.362	46.41	46.452	46.357	46.385	46.317	46.363	46.37	46.336
ch 2	47.104	47.073	46.895	46.95	46.919	47.013	47.166	46.895	46.958	46.938
ch 3	48.096	47.763	47.83	48.121	47.843	47.766	48.072	47.849	47.794	47.802
ch 4	48.635	48.636	48.646	48.642	48.61	48.627	48.537	48.538	48.549	48.737
	Run 11	Run 12	Run 13	Run 14	Run 15	Run 16	Run 17	Run 18	Run 19	Run 20
ch 1	46.354	46.358	46.343	46.356	46.318	46.37	46.361	46.315	46.383	46.376
ch 2	46.945	47.219	46.909	46.957	46.985	46.917	46.197	46.963	46.939	46.937
ch 3	47.856	47.998	47.808	47.842	48.142	47.821	47.786	48.149	47.855	47.783
ch 4	48.722	48.984	48.991	48.998	48.79	48.786	48.557	48.548	48.551	48.603
	Run 21	Run 22	Run 23	Run 24	Run 25	Run 26	Run 27	Run 28	Run 29	Run 30
ch 1	46.333	46.408	46.364	46.445	46.452	46.362	46.415	46.456	46.374	46.365
ch 2	46.885	47.126	47.059	46.917	46.948	46.921	47.045	47.167	46.981	46.927
ch 3	47.876	47.863	47.884	47.789	47.865	48.152	47.78	47.82	48.158	47.866
ch 4	48.593	48.651	48.63	48.63	48.637	48.662	48.567	48.59	48.587	48.569
	Run 31	Run 32	Run 33	Run 34	Run 35	Run 36	Run 37	Run 38	Run 39	Run 40
ch 1	46.328	46.365	46.365	46.331	46.344	46.339	46.361	46.361	46.328	46.383
ch 2	46.942	46.906	47.087	47.071	46.91	46.94	46.919	47.011	47.16	46.86
ch 3	47.802	47.988	47.988	47.802	47.792	47.866	48.068	47.783	47.823	48.101
ch 4	48.568	48.833	48.841	48.961	48.968	48.689	48.66	48.676	48.546	
	Run 41	Run 42	Run 43	Run 44	Run 45	Run 46	Run 47	Run 48	Run 49	Run 50
ch 1	46.363	46.326	46.39	46.362	46.457	46.407	46.356	46.421	46.452	46.37
ch 2	46.956	46.937	46.92	47.218	46.93	46.934	46.934	46.889	47.211	46.971
ch 3	47.839	47.775	48.097	47.847	47.781	47.827	47.861	47.952	47.798	47.842
ch 4	48.556	48.627	48.624	48.624	48.65	48.626	48.641	48.639	48.633	48.542

Optical Amplifier 1

Output channel power

	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
ch 1	-7.056	-7.156	-7.074	-7.079	-7.132	-7.101	-7.11	-7.173	-7.088	-7.082
ch 2	-7.099	-7.089	-7.13	-7.134	-7.135	-7.092	-7.086	-7.108	-7.129	-7.127
ch 3	-7.08	-7.127	-7.127	-7.062	-7.134	-7.125	-7.058	-7.138	-7.107	-7.099
ch 4	-7.17	-7.132	-7.144	-7.133	-7.142	-7.125	-7.109	-7.104	-7.107	-7.063
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
ch 1	-7.149	-7.091	-7.125	-7.156	-7.129	-7.101	-7.156	-7.1	-7.099	-7.118
ch 2	-7.079	-7.053	-7.082	-7.131	-7.132	-7.091	-7.087	-7.084	-7.134	-7.151
ch 3	-7.164	-7.09	-7.088	-7.155	-7.062	-7.122	-7.119	-7.044	-7.137	-7.123
ch 4	-7.077	-7.049	-7.051	-7.044	-7.065	-7.071	-7.104	-7.112	-7.102	-7.125
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
ch 1	-7.102	-7.09	-7.121	-7.094	-7.051	-7.161	-7.058	-7.103	-7.153	-7.112
ch 2	-7.104	-7.064	-7.065	-7.152	-7.156	-7.132	-7.054	-7.089	-7.102	-7.154
ch 3	-7.093	-7.137	-7.096	-7.107	-7.134	-7.041	-7.109	-7.147	-7.037	-7.13
ch 4	-7.138	-7.135	-7.139	-7.134	-7.168	-7.119	-7.158	-7.108	-7.152	-7.123
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
ch 1	-7.11	-7.15	-7.077	-7.101	-7.718	-7.105	-7.082	-7.137	-7.122	-7.109
ch 2	-7.134	-7.111	-7.096	-7.099	-7.109	-7.142	-7.148	-7.075	-7.088	-7.126
ch 3	-7.149	-7.084	-7.149	-7.11	-7.117	-7.133	-7.085	-7.125	-7.171	-7.093
ch 4	-7.123	-7.087	-7.092	-7.08	-7.074	-7.075	-7.079	-7.1	-7.087	-7.128
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
ch 1	-7.14	-7.087	-7.071	-7.142	-7.079	-7.087	-7.161	-7.112	-7.068	-7.14
ch 2	-7.145	-7.136	-7.104	-7.037	-7.087	-7.087	-7.158	-7.136	-7.043	-7.063
ch 3	-7.115	-7.131	-7.081	-7.169	-7.147	-7.078	-7.148	-7.071	-7.144	-7.17
ch 4	-7.118	-7.168	-7.148	-7.164	-7.151	-7.164	-7.136	-7.161	-7.125	-7.142

Output channel OSNR

	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
ch 1	46.45	46.362	46.408	46.451	46.355	46.382	46.315	46.352	46.367	46.336
ch 2	47.103	47.072	46.893	46.947	46.918	47.012	47.164	46.891	46.957	46.936
ch 3	48.094	47.762	47.828	48.12	47.841	47.763	48.072	47.849	47.793	47.799
ch 4	48.634	48.636	48.643	48.639	48.611	48.628	48.535	48.536	48.549	48.734
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
ch 1	46.352	46.358	46.341	46.362	46.317	46.357	46.361	46.313	46.381	46.374
ch 2	46.944	47.217	46.909	46.956	46.961	46.915	47.195	46.951	46.937	46.936
ch 3	47.854	47.996	47.807	47.843	48.141	47.819	47.786	48.148	47.854	47.783
ch 4	48.719	48.982	48.988	48.997	48.791	48.787	48.557	48.546	48.55	48.603
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
ch 1	46.333	46.407	46.363	46.443	46.449	46.36	46.415	46.455	46.371	46.364
ch 2	46.885	47.126	47.056	46.917	46.947	46.919	47.045	47.166	46.979	46.927
ch 3	47.874	47.863	47.883	47.788	47.861	48.153	47.778	47.817	48.157	47.863
ch 4	48.589	48.649	48.629	48.627	48.635	48.651	48.565	48.591	48.588	48.558
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
ch 1	46.326	46.362	46.363	46.33	46.344	46.336	46.36	46.359	46.328	46.312
ch 2	46.939	46.905	47.086	47.068	46.91	46.939	46.918	47.009	47.157	46.878
ch 3	47.8	47.988	47.854	47.799	47.792	47.864	48.067	47.783	47.821	48.099
ch 4	48.556	48.832	48.832	48.841	48.96	48.966	48.689	48.66	48.675	48.543
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
ch 1	46.361	46.323	46.391	46.361	46.456	46.405	46.354	46.419	46.482	46.37
ch 2	46.954	46.936	46.919	47.219	46.928	46.928	46.937	46.888	47.21	46.971
ch 3	47.837	47.774	48.095	47.849	47.779	47.826	47.859	47.951	47.795	47.839
ch 4	48.552	48.627	48.621	48.624	48.648	48.624	48.637	48.636	48.632	48.541

Performance data for the optical cross connect (OXC1)

Optical Cross Connect										
Wavelength Demux										
Output channel power										
	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
ch 1	-13.161	-13.104	-13.083	-13.067	-13.109	-13.158	-13.147	-13.135	-13.074	-13.116
ch 2	-13.049	-13.139	-13.046	-13.144	-13.058	-13.148	-13.067	-13.144	-13.084	-13.135
ch 3	-13.106	-13.098	-13.061	-13.056	-13.11	-13.118	-13.144	-13.12	-13.168	-13.16
ch 4	-13.105	-13.094	-13.092	-13.089	-13.064	-13.083	-13.116	-13.078	-13.125	-13.111
	Run 11	Run 12	Run 13	Run 14	Run 15	Run 16	Run 17	Run 18	Run 19	Run 20
ch 1	-13.154	-13.171	-13.11	-13.092	-13.053	-13.126	-13.145	-13.134	-13.089	-13.087
ch 2	-13.051	-13.129	-13.062	-13.162	-13.1	-13.146	-13.06	-13.142	-13.074	-13.143
ch 3	-13.16	-13.156	-13.176	-13.155	-13.151	-13.111	-13.138	-13.144	-13.117	-13.099
ch 4	-13.133	-13.15	-13.137	-13.151	-13.141	-13.124	-13.096	-13.072	-13.098	-13.087
	Run 21	Run 22	Run 23	Run 24	Run 25	Run 26	Run 27	Run 28	Run 29	Run 30
ch 1	-13.089	-13.111	-13.134	-13.137	-13.089	-13.071	-13.115	-13.124	-13.16	-13.116
ch 2	-13.068	-13.13	-13.126	-13.139	-13.117	-13.136	-13.108	-13.133	-13.108	-13.131
ch 3	-13.111	-13.094	-13.098	-13.083	-13.096	-13.057	-13.078	-13.064	-13.076	-13.061
ch 4	-13.044	-13.089	-13.101	-13.077	-13.091	-13.133	-13.163	-13.134	-13.149	-13.153
	Run 31	Run 32	Run 33	Run 34	Run 35	Run 36	Run 37	Run 38	Run 39	Run 40
ch 1	-13.101	-13.092	-13.127	-13.136	-13.142	-13.104	-13.06	-13.07	-13.157	-13.151
ch 2	-13.115	-13.125	-13.154	-13.099	-13.137	-13.09	-13.102	-13.087	-13.154	-13.098
ch 3	-13.064	-13.079	-13.056	-13.1	-13.125	-13.127	-13.131	-13.111	-13.122	-13.141
ch 4	-13.15	-13.129	-13.111	-13.081	-13.087	-13.07	-13.087	-13.074	-13.099	-13.102
	Run 41	Run 42	Run 43	Run 44	Run 45	Run 46	Run 47	Run 48	Run 49	Run 50
ch 1	-13.119	-13.058	-13.076	-13.155	-13.146	-13.142	-13.079	-13.086	-13.097	-13.134
ch 2	-13.115	-13.09	-13.159	-13.106	-13.109	-13.086	-13.141	-13.095	-13.153	-13.057
ch 3	-13.148	-13.17	-13.136	-13.124	-13.132	-13.115	-13.144	-13.149	-13.125	-13.141
ch 4	-13.142	-13.115	-13.153	-13.153	-13.172	-13.133	-13.129	-13.135	-13.087	-13.082
Optical Cross Connect										
Wavelength Demux										
Output channel OSNR										
	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
ch 1	46.33	46.311	46.476	46.537	46.336	46.376	46.321	46.308	46.362	46.483
ch 2	47.197	46.943	47.22	46.989	47.232	46.945	47.196	46.925	47.226	46.969
ch 3	48.008	47.984	47.982	47.786	47.787	47.761	47.743	47.741	47.841	47.836
ch 4	48.527	48.744	48.736	48.982	48.94	48.934	48.68	48.685	48.515	48.538
	Run 11	Run 12	Run 13	Run 14	Run 15	Run 16	Run 17	Run 18	Run 19	Run 20
ch 1	46.316	46.377	46.321	46.526	46.64	46.305	46.314	46.31	46.344	46.549
ch 2	47.209	46.934	47.149	46.911	47.076	46.87	47.182	46.911	47.112	46.897
ch 3	47.849	47.857	47.841	47.798	47.781	47.792	47.793	47.761	47.787	47.826
ch 4	48.598	48.582	48.637	48.633	48.506	48.53	48.595	48.837	48.821	48.994
	Run 21	Run 22	Run 23	Run 24	Run 25	Run 26	Run 27	Run 28	Run 29	Run 30
ch 1	46.475	46.296	46.377	46.299	46.534	46.629	46.442	46.35	46.369	46.341
ch 2	47.061	46.868	46.954	46.87	46.897	46.885	47.011	46.867	46.933	46.883
ch 3	47.86	47.871	47.893	47.925	48.131	48.149	48.161	48.155	48.161	48.033
ch 4	48.998	48.87	48.86	48.616	48.59	48.518	48.521	48.648	48.621	48.61
	Run 31	Run 32	Run 33	Run 34	Run 35	Run 36	Run 37	Run 38	Run 39	Run 40
ch 1	46.544	46.448	46.278	46.348	46.289	46.355	46.617	46.428	46.269	46.375
ch 2	46.888	46.937	46.863	47.022	46.899	46.906	46.877	46.975	46.87	47.037
ch 3	48.03	48.006	47.987	47.962	47.767	47.76	47.755	47.771	47.764	47.845
ch 4	48.611	48.512	48.656	48.652	48.916	48.915	48.977	48.985	48.763	48.77
	Run 41	Run 42	Run 43	Run 44	Run 45	Run 46	Run 47	Run 48	Run 49	Run 50
ch 1	46.301	46.592	46.609	46.267	46.362	46.359	46.397	46.6	46.392	46.292
ch 2	46.883	47.132	46.915	47.004	46.892	47.088	46.912	47.166	46.943	47.216
ch 3	47.849	47.842	47.871	47.881	47.807	47.801	47.775	47.778	47.778	47.749
ch 4	48.519	48.544	48.553	48.566	48.646	48.557	48.571	48.53	48.552	48.751

Input combined power										
	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**Run 6**	**Run 7**	**Run 8**	**Run 9**	**Run 10**
	-2.925	-2.921	-2.916	-2.924	-2.924	-2.923	-2.934	-2.921	-2.924	-2.928
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
	-2.923	-2.929	-2.924	-2.939	-2.929	-2.92	-2.191	-2.923	-2.922	-2.932
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
	-2.921	-2.916	-2.927	-2.918	-2.928	-2.925	-2.939	-2.92	-2.929	-2.929
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
	-2.927	-2.928	-2.931	-2.922	-2.931	-2.921	-2.921	-2.916	-2.935	-2.928
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
	-2.922	-2.929	-2.928	-2.93	-2.924	-2.914	-2.915	-2.939	-2.916	-2.925
Wavelength Mux										
Input channel power										
	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**Run 6**	**Run 7**	**Run 8**	**Run 9**	**Run 10**
ch 1	-13.66	-13.602	-13.582	-13.567	-13.61	-13.658	-13.646	-13.634	-13.574	-13.617
ch 2	-13.55	-13.638	-13.546	-13.644	-13.558	-13.648	-13.565	-13.645	-13.554	-13.635
ch 3	-13.607	-13.599	-13.562	-13.607	-13.611	-13.619	-13.645	-13.621	-13.638	-13.65
ch 4	-13.604	-13.593	-13.591	-13.588	-13.564	-13.583	-13.618	-13.579	-13.626	-13.612
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
ch 1	-13.675	-13.671	-13.668	-13.591	-13.553	-13.627	-13.646	-13.634	-13.588	-13.583
ch 2	-13.551	-13.629	-13.551	-13.662	-13.599	-13.646	-13.56	-13.642	-13.573	-13.644
ch 3	-13.66	-13.655	-13.676	-13.651	-13.65	-13.611	-13.637	-13.643	-13.616	-13.598
ch 4	-13.633	-13.651	-13.636	-13.651	-13.64	-13.623	-13.593	-13.571	-13.597	-13.587
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
ch 1	-13.589	-13.613	-13.634	-13.636	-13.588	-13.571	-13.616	-13.625	-13.66	-13.614
ch 2	-13.567	-13.631	-13.624	-13.64	-13.615	-13.637	-13.607	-13.634	-13.607	-13.633
ch 3	-13.61	-13.592	-13.597	-13.582	-13.595	-13.567	-13.578	-13.564	-13.576	-13.562
ch 4	-13.543	-13.56	-13.602	-13.579	-13.592	-13.634	-13.664	-13.635	-13.65	-13.652
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
ch 1	-13.601	-13.593	-13.625	-13.636	-13.641	-13.602	-13.56	-13.571	-13.638	-13.65
ch 2	-13.614	-13.627	-13.653	-13.601	-13.636	-13.592	-13.6	-13.588	-13.653	-13.599
ch 3	-13.565	-13.58	-13.559	-13.602	-13.627	-13.628	-13.632	-13.612	-13.623	-13.643
ch 4	-13.65	-13.628	-13.61	-13.58	-13.586	-13.599	-13.587	-13.574	-13.6	-13.603
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
ch 1	-13.618	-13.557	-13.567	-13.655	-13.646	-13.642	-13.577	-13.586	-13.598	-13.635
ch 2	-13.614	-13.591	-13.658	-13.608	-13.608	-13.587	-13.64	-13.596	-13.653	-13.557
ch 3	-13.648	-13.671	-13.636	-13.624	-13.632	-13.614	-13.613	-13.648	-13.624	-13.64
ch 4	-13.643	-13.616	-13.654	-13.653	-13.672	-13.633	-13.628	-13.634	-13.586	-13.581

Input channel OSNR										
	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**Run 6**	**Run 7**	**Run 8**	**Run 9**	**Run 10**
ch 1	46.326	46.315	-46.485	46.529	46.331	46.377	46.318	46.313	6.31	46.475
ch 2	47.19	46.939	47.218	47.939	47.235	46.947	47.202	46.928	47.223	46.97
ch 3	47.997	47.973	47.971	47.781	47.784	47.759	47.742	47.742	47.845	47.839
ch 4	48.531	48.756	48.748	48.969	48.934	48.926	48.669	48.673	48.514	48.536
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
ch 1	46.32	46.376	46.326	46.534	46.638	46.301	46.318	46.307	46.35	46.557
ch 2	47.213	46.937	47.157	46.915	47.066	46.872	47.187	46.915	47.12	46.901
ch 3	47.857	47.859	47.842	47.793	47.777	47.768	47.789	47.758	47.784	47.835
ch 4	48.603	48.587	48.634	48.63	48.502	48.527	48.603	48.848	48.833	48.995
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
ch 1	46.467	46.293	46.376	46.296	46.542	46.626	46.433	46.353	46.357	46.348
ch 2	47.07	46.889	46.962	46.878	46.903	46.888	47.02	46.867	46.941	46.878
ch 3	47.87	47.881	47.903	47.936	48.137	48.154	48.164	48.157	48.162	48.023
ch 4	48.999	48.859	48.849	48.607	48.581	48.532	48.525	48.65	48.625	48.601
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
ch 1	46.551	46.439	46.276	46.351	46.289	46.362	46.613	46.42	46.269	46.372
ch 2	46.892	46.931	46.864	47.013	46.905	46.9	46.89	46.967	46.869	47.027
ch 3	48.02	47.986	47.977	47.952	47.764	47.759	47.754	47.771	47.763	47.849
ch 4	48.606	48.512	48.657	48.664	48.925	48.924	48.973	48.98	48.75	48.759
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
ch 1	46.301	46.598	46.603	46.266	46.366	46.356	46.405	46.605	46.384	46.292
ch 2	46.879	47.124	46.911	46.995	46.88	47.08	46.909	47.158	46.94	47.212
ch 3	47.851	47.845	47.873	47.882	47.803	47.797	47.771	47.774	47.775	47.747
ch 4	48.514	48.539	48.558	48.572	48.645	48.551	48.565	48.535	48.556	48.763

Output combined power										
	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**Run 6**	**Run 7**	**Run 8**	**Run 9**	**Run 10**
	-9.541	-9.537	-9.531	-9.538	-9.539	-9.538	-9.55	-9.536	-9.54	-9.543
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
	-9.538	-9.544	-9.54	-9.533	-9.545	-9.536	-9.534	-9.538	-9.537	-9.549
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
	-9.537	-9.531	-9.543	-9.534	-9.544	-9.541	-9.555	-9.535	-9.544	-9.544
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
	-9.543	-9.544	-9.547	-9.538	-9.547	-9.537	-9.536	-9.533	-9.551	-9.544
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
	-9.537	-9.545	-9.544	-9.545	-9.54	-9.53	-9.53	-9.564	-9.531	-9.541

OXC overall										
Input combined power										
	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
	0.075	0.079	0.084	0.076	0.076	0.077	0.083	0.079	0.076	0.072
	Run 11	Run 12	Run 13	Run 14	Run 15	Run 16	Run 17	Run 18	Run 19	Run 20
	0.077	0.071	0.076	0.057	0.071	0.08	0.081	0.077	0.078	0.068
	Run 21	Run 22	Run 23	Run 24	Run 25	Run 26	Run 27	Run 28	Run 29	Run 30
	0.079	0.084	0.073	0.082	0.072	0.075	0.061	0.08	0.071	0.071
	Run 31	Run 32	Run 33	Run 34	Run 35	Run 36	Run 37	Run 38	Run 39	Run 40
	0.073	0.072	0.069	0.078	0.069	0.079	0.079	0.082	0.065	0.072
	Run 41	Run 42	Run 43	Run 44	Run 45	Run 46	Run 47	Run 48	Run 49	Run 50
	0.078	0.071	0.072	0.07	0.076	0.086	0.085	0.061	0.084	0.075
Output combined power										
	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
	0.472	0.476	0.482	0.476	0.473	0.475	0.463	0.477	0.473	0.468
	Run 11	Run 12	Run 13	Run 14	Run 15	Run 16	Run 17	Run 18	Run 19	Run 20
	0.475	0.469	0.473	0.46	0.467	0.476	0.478	0.475	0.475	0.464
	Run 21	Run 22	Run 23	Run 24	Run 25	Run 26	Run 27	Run 28	Run 29	Run 30
	0.476	0.481	0.469	0.479	0.469	0.473	0.458	0.477	0.469	0.47
	Run 31	Run 32	Run 33	Run 34	Run 35	Run 36	Run 37	Run 38	Run 39	Run 40
	0.47	0.47	0.467	0.475	0.466	0.476	0.477	0.48	0.462	0.469
	Run 41	Run 42	Run 43	Run 44	Run 45	Run 46	Run 47	Run 48	Run 49	Run 50
	0.475	468	0.469	0.467	0.473	0.483	0.482	0.459	0.481	0.472

Performance data for the optical add drop multiplexer (OADM1)

Optical Add Drop Multiplexer										
Input channel power										
	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**Run 6**	**Run 7**	**Run 8**	**Run 9**	**Run 10**
ch 1	-6.588	-6.578	-6.585	-6.632	-6.669	-6.604	-6.585	-6.592	-6.629	-6.67
ch 2	-6.57	-6.624	-6.552	-6.652	-6.581	-6.649	-6.575	-6.653	-6.558	-6.628
ch 3	-6.608	-6.588	-6.547	-6.556	-6.561	-6.566	-6.606	-6.563	-6.62	-6.599
ch 4	-6.594	-6.653	-6.461	-6.668	-6.637	-6.648	-6.628	-6.603	-6.591	-6.58
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
ch 1	-6.627	-6.614	-6.574	-6.618	-6.62	-6.653	-6.618	-6.574	-6.556	-6.627
ch 2	-6.556	-6.626	-6.549	-6.662	-6.569	-6.641	-6.563	-6.653	-6.569	-6.643
ch 3	-6.614	-6.617	-6.629	-6.638	-6.651	-6.628	-6.645	-6.641	-6.629	-6.634
ch 4	-6.553	-6.582	-6.568	-6.571	-6.603	-6.57	-6.602	-6.632	-6.642	-6.657
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
ch 1	-6.658	-6.632	-6.574	-6.579	-6.612	-6.647	-6.663	-6.588	-6.615	-6.58
ch 2	-6.558	-6.643	-6.608	-6.632	-6.61	-6.649	-6.621	-6.631	-6.593	-6.644
ch 3	-6.669	-6.635	-6.636	-6.628	-6.633	-6.617	-6.621	-6.605	-6.622	-6.582
ch 4	-6.629	-6.631	-6.65	-6.595	-6.595	-6.575	-6.605	-6.561	-6.579	-6.585
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
ch 1	-6.632	-6.653	-6.647	-6.586	-6.571	-6.58	-6.611	-6.625	-6.665	-6.58
ch 2	-6.608	-6.63	-6.614	-6.608	-6.609	-6.597	-6.589	-6.591	-6.631	-6.594
ch 3	-6.57	-6.577	-6.568	-6.593	-6.596	-6.582	-6.579	-6.56	-6.563	-6.604
ch 4	-6.587	-6.609	-6.643	-6.62	-6.645	-6.657	-6.647	-6.649	-6.632	-6.645
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
ch 1	-6.558	-6.603	-6.65	-6.659	-6.601	-6.577	-6.563	-6.643	-6.652	-6.654
ch 2	-6.6	-6.601	-6.662	-6.615	-6.603	-6.594	-6.637	-6.605	-6.647	-6.57
ch 3	-6.591	-6.62	-6.597	-6.587	-6.627	-6.62	-6.631	-6.647	-6.651	-6.656
ch 4	-6.619	-6.6	-6.594	-6.597	-6.595	-6.589	-6.587	-6.617	-6.596	-6.64
Input channel OSNR										
	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**Run 6**	**Run 7**	**Run 8**	**Run 9**	**Run 10**
ch 1	46.606	46.592	46.444	46.301	46.281	46.454	46.611	46.579	46.321	46.289
ch 2	47.139	46.876	47.191	46.857	47.212	46.844	47.217	46.844	47.202	46.856
ch 3	48.083	48.1	48.107	48.15	48.13	48.124	48.108	48.119	47.918	47.925
ch 4	48.649	48.532	48.514	48.465	48.51	48.493	48.626	48.618	48.813	48.799
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
ch 1	46.34	46.465	46.553	46.417	46.311	46.28	46.352	46.617	46.585	46.403
ch 2	47.213	46.857	47.213	46.848	47.179	46.877	47.208	46.846	47.193	46.866
ch 3	47.907	47.886	47.878	47.773	47.76	47.752	47.743	47.754	47.744	47.755
ch 4	48.979	48.96	48.948	48.946	48.756	48.78	48.62	48.503	48.495	48.499
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
ch 1	46.28	46.287	46.514	46.624	46.397	46.301	46.283	46.379	46.511	46.562
ch 2	47.166	46.861	47.08	46.903	47.023	46.861	47.108	46.895	47.063	46.891
ch 3	47.723	47.75	47.754	47.756	47.843	47.84	47.845	47.887	47.878	48.05
ch 4	48.479	48.547	48.531	48.684	48.684	48.881	48.867	48.986	48.976	48.891
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
ch 1	46.388	46.283	46.299	46.39	46.628	46.539	46.319	46.291	46.294	46.538
ch 2	47.003	46.951	46.974	47.004	47.044	46.94	46.922	46.975	46.943	47.013
ch 3	48.08	48.085	48.115	48.102	48.118	48.133	48.124	48.118	48.105	47.912
ch 4	48.895	48.701	48.553	48.549	48.48	48.467	48.478	48.479	48.588	48.0565
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
ch 1	46.621	46.384	46.295	46.301	46.397	46.552	46.524	46.344	46.281	46.301
ch 2	46.912	47.085	46.871	46.984	46.937	47.056	46.887	47.109	46.875	47.16
ch 3	47.924	47.904	47.872	47.882	47.768	47.767	47.755	47.749	47.742	47.754
ch 4	48.722	48.727	48.904	48.92	48.957	48.82	48.824	48.654	48.652	48.523

Optical Add Drop Multiplexer										
Input combined power										
	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**Run 6**	**Run 7**	**Run 8**	**Run 9**	**Run 10**
	0.48	0.483	0.5	0.481	0.483	0.484	0.473	0.484	0.483	0.48
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
	0.486	0.488	0.489	0.474	0.482	0.489	0.492	0.485	0.491	0.477
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
	0.483	0.489	0.484	0.488	0.479	0.482	0.47	0.491	0.477	0.483
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
	0.48	0.48	0.482	0.486	0.482	0.485	0.495	0.494	0.475	0.484
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
	0.493	0.479	0.478	0.48	0.482	0.491	0.492	0.467	0.486	0.482
Output channel power										
	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**Run 6**	**Run 7**	**Run 8**	**Run 9**	**Run 10**
ch 1	-9.596	-9.585	-9.592	-9.639	-9.677	-9.613	-9.593	-9.6	-9.636	-9.678
ch2	-9.585	-9.642	-9.568	-9.659	-9.597	-9.665	-9.592	-9.638	-9.594	-9.645
ch3	-9.858	-10.128	-10.299	-10.152	-10.036	-10.206	-9.87	-10.561	-10.318	-10.311
ch4	-9.61	-9.669	-9.657	-9.684	-9.653	-9.664	-9.644	-9.619	-9.607	-9.606
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
ch 1	-9.636	-9.622	-9.582	-9.625	-9.627	-9.671	-9.627	-9.582	-9.563	-9.634
ch2	-9.572	-9.642	-9.56	-9.678	-9.587	-9.657	-9.58	-9.669	-9.597	-9.668
ch3	-10.148	-10.328	-9.85	-10.219	-10.354	-10.538	-10.027	-10.025	-10.02	-9.882
ch4	-9.569	-9.598	-9.574	-9.587	-9.619	-9.586	-9.618	-9.648	-9.658	-9.673
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
ch 1	-9.665	-9.641	-9.582	-9.587	-9.618	-9.654	-9.671	-9.598	-9.623	-9.588
ch2	-9.586	-9.657	-9.626	-9.647	-9.629	-9.664	-9.639	-9.646	-9.612	-9.658
ch3	-10.038	-10.149	-10.083	-10.232	-9.991	-10.319	-10.301	-10.033	-10.421	-10.027
ch4	-9.645	-9.647	-9.666	-9.611	-9.611	-9.591	-9.621	-9.577	-9.596	-9.601
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
ch 1	-9.629	-93.661	-9.656	-9.595	-9.579	-9.587	-9.618	-9.633	-9.674	-9.588
ch2	-9.627	-9.644	-9.634	-9.623	-9.627	-9.612	-9.588	-9.605	-9.65	-9.609
ch3	-10.01	-10.173	-10.241	-10.253	-10.088	-10.049	-10.031	-10.018	-10.03	-10.187
ch4	-9.604	-9.625	-9.659	-9.636	-9.661	-9.673	-9.663	-9.665	-9.648	-9.661
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
ch 1	-9.566	-9.61	-9.657	-9.668	-9.61	-9.586	-9.57	-9.649	-9.66	-9.663
ch2	-9.618	-9.616	-9.68	-9.629	-9.622	-9.609	-9.665	-9.62	-9.665	-9.598
ch3	-10.061	-10.391	-10.053	-10.132	-10.017	-10.246	-10.343	-10.404	-9.944	-9.967
ch4	-9.636	-9.616	-9.61	-9.613	-9.611	-9.606	-9.603	-9.633	-9.611	-9.656

Output channel OSNR										
	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**Run 6**	**Run 7**	**Run 8**	**Run 9**	**Run 10**
ch 1	46.601	46.596	46.45	46.3	46.279	46.446	46.606	46.583	46.322	46.288
ch 2	47.152	46.874	47.199	46.856	47.215	46.844	47.215	46.846	47.209	46.806
ch 3	48.14	48.296	48.105	48.033	48.123	48.248	48.131	47.937	48.139	48.151
ch 4	48.65	48.533	48.516	48.467	48.51	48.493	48.624	48.619	48.815	48.8
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
ch 1	46.335	46.458	46.587	46.421	46.312	46.277	46.348	46.613	46.589	46.407
ch 2	47.213	46.868	47.209	46.85	47.17	46.881	47.207	46.847	47.187	46.869
ch 3	48.027	48.056	48.327	48.054	47.958	47.997	48.293	48.079	48.081	48.308
ch 4	48.979	48.96	48.948	48.946	48.756	48.78	48.621	48.504	48.497	48.5
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
ch 1	46.279	46.285	46.506	46.821	46.402	46.301	46.282	46.374	46.503	46.555
ch 2	47.156	46.866	47.067	46.909	47.011	46.864	47.086	46.9	47.05	46.899
ch 3	48.189	47.99	48.073	48.235	48.036	47.946	48.116	48.269	47.963	48.129
ch 4	48.48	48.518	48.532	48.685	48.685	48.861	48.868	48.987	48.977	48.891
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
ch 1	46.392	46.282	46.297	46.384	46.626	46.514	46.319	46.289	46.292	46.531
ch 2	46.992	46.961	46.965	47.016	47.031	46.949	46.982	46.986	46.936	47.027
ch 3	48.316	48.123	47.983	48.09	48.314	48.086	48.091	48.286	48.249	48.018
ch 4	48.895	48.701	48.554	48.55	48.481	48.468	48.478	48.48	48.589	48.586
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
ch 1	46.62	46.387	46.294	46.298	46.391	46.545	46.529	46.347	46.279	46.301
ch 2	46.908	47.098	46.868	46.986	46.931	47.07	46.881	47.122	46.873	47.171
ch 3	48.064	48.208	48.104	47.952	48.167	48.243	47.985	47.968	48.294	48.238
ch 4	48.723	48.728	48.906	48.971	48.967	48.82	48.823	48.654	48.652	48.524
Output combined power										
	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**Run 6**	**Run 7**	**Run 8**	**Run 9**	**Run 10**
	-2.638	-2.698	-2.715	-2.698	-2.683	-2.715	-2.647	-2.769	-2.723	-2.727
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
	-2.689	-2.728	-2.634	-2.716	-2.728	-2.764	-2.689	-2.673	-2.657	-2.66
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
	-2.674	-2.697	-2.686	-2.716	-2.667	-2.723	-2.74	-2.675	-2.747	-2.677
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
	-2.681	-2.707	-2.716	-2.713	-2.691	-2.673	-2.66	-2.672	-2.689	-2.699
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
	-2.671	-2.745	-2.688	-2.731	-2.676	-2.714	-2.721	-2.752	-2.66	-2.672

Performance data for the optical amplifier (OA2)

Optical Amplifier 2										
Input channel power										
	Run 1	Run2	Run3	Run4	Run5	Run6	Run7	Run8	Run9	Run10
ch1	-19.644	-19.651	-19.649	-19.605	-19.596	-19.619	-19.635	-19.66	-19.629	-19.627
ch2	-19.623	-19.632	-19.585	-19.659	-19.599	-19.65	-19.595	-19.662	-19.61	-19.666
ch3	-19.892	-20.153	-20.278	-20.138	-20.079	-20.202	-19.847	-20.569	-20.348	-20.311
ch4	-19.591	-19.607	-19.602	-19.599	-19.614	-19.611	-19.639	-19.626	-19.665	-19.659
	Run 11	Run 12	Run 13	Run 14	Run 15	Run 16	Run 17	Run 18	Run 19	Run 20
ch1	-19.59	-19.606	-19.659	-19.658	-19.615	-19.596	-19.577	-19.64	-19.636	-19.679
ch2	-19.585	-19.668	-19.592	-19.686	-19.591	-19.665	-19.576	-19.672	-19.587	-19.666
ch3	-20.113	-20.351	-19.858	-20.202	-20.329	-20.559	-20.04	-19.983	-20.015	-19.917
ch4	-19.662	-19.66	-19.641	-19.628	-19.624	-19.608	-19.591	-19.59	-19.581	-19.633
	Run21	Run22	Run23	Run24	Run25	Run26	Run27	Run28	Run29	Run30
ch1	-19.6	-19.561	-19.601	-19.648	-19.649	-19.625	-19.603	-19.574	-19.623	-19.665
ch2	-19.59	-19.663	-19.601	-19.665	-19.617	-19.672	-19.612	-19.651	-19.606	-19.683
ch3	-20.013	-20.124	-20.116	-20.245	-19.963	-20.234	-20.326	-20.052	-20.385	-20.062
ch4	-19.591	-19.625	-19.623	-19.633	-19.646	-19.652	-19.676	-19.644	-19.677	-19.648
	Run31	Run32	Run33	Run34	Run35	Run36	Run37	Run38	Run39	Run40
ch1	-19.948	-19.588	-19.588	-19.599	-19.624	-19.664	-19.617	-19.59	-19.6	-19.609
ch2	-19.608	-19.653	-19.609	-19.643	-19.643	-19.626	-19.573	-19.636	-19.622	-19.618
ch3	-20.05	-20.147	-20.225	-20.266	-20.111	-20.032	-20.049	-20.039	-20.02	-20.083
ch4	-19.636	-19.615	-19.602	-19.587	-19.603	-19.611	-19.61	-19.601	-19.641	-19.636
	Run41	Run42	Run43	Run44	Run45	Run46	Run47	Run48	Run49	Run50
ch1	-19.621	-19.65	-19.638	-19.594	-19.594	-19.608	-19.649	-19.656	-19.599	-19.592
ch2	-19.611	-19.624	-19.66	-19.64	-19.609	-19.616	-19.631	-19.637	-19.648	-19.604
ch3	-20.158	-20.399	-20.035	-20.311	-20.044	-20.256	-20.337	-20.043	-19.97	-19.963
ch4	-19.66	-19.658	-19.673	-19.676	-19.683	-19.637	-19.639	-19.627	-19.61	-19.606

Input channel OSNR										
	Run 1	Run2	Run3	Run4	Run5	Run6	Run7	Run8	Run9	Run 10
ch1	46.383	46.197	46.176	46.511	46.595	46.507	46.363	46.187	46.379	46.519
ch2	47.097	46.937	47.147	46.868	47.174	46.801	47.184	46.737	47.146	46.821
ch3	48.06	48.099	48.246	48.09	48.047	48.244	48.243	47.922	48.015	48.228
ch4	48.878	48.936	48.938	48.934	48.864	48.857	48.72	48.704	48.439	48.419
	Run 11	Run 12	Run 13	Run 14	Run 15	Run 16	Run 17	Run 18	Run 19	Run 20
ch1	46.598	46.513	460183	46.225	46.404	46.587	46.591	46.328	46.183	46.229
ch2	47.177	46.752	47.177	46.748	47.191	46.742	47.187	46.747	473.191	46.714
ch3	48.171	47.978	48.282	48.232	47.999	47.889	48.231	48.181	48.046	48.146
ch4	48.292	48.316	48.508	48.507	48.768	48.763	48.898	48.931	48.949	48.925
	Run21	Run22	Run23	Run24	Run25	Run26	Run27	Run28	Run29	Run30
ch1	46.528	46.61	46.484	46.31	46.226	46.406	46.542	46.586	46.462	46.158
ch2	47.193	46.74	47.178	46.183	47.151	46.723	47.188	46.781	47.171	46.821
ch3	48.308	48.087	48	48.172	48.214	47.9	47.932	48.231	48.098	48.013
ch4	48.918	48.833	48.836	48.637	48.625	48.634	48.335	48.347	48.348	48.6
	Run31	Run32	Run33	Run34	Run35	Run36	Run37	Run38	Run39	Run40
ch1	46.235	46.543	46.586	46.563	46.308	46.152	46.438	46.556	46.6	46.45
ch2	47.143	46.898	47.112	46.981	47.154	46.884	47.135	46.943	47.087	47.009
ch3	48.153	48.279	48.033	48.002	48.171	48.202	48.037	48.101	48.317	48.043
ch4	48.604	48.821	48.927	48.924	48.953	48.923	48.896	48.909	48.768	48.77
	Run41	Run42	Run43	Run44	Run45	Run46	Run47	Run48	Run49	Run50
ch1	46.293	46.263	46.445	46.603	46.552	46.437	46.156	46.285	46.558	46.606
ch2	47.042	47.067	46.978	46.958	47.075	47.028	47.013	47.076	46.944	47.131
ch3	48.15	48.132	48.291	48.021	48.043	48.19	48.051	47.959	48.14	48.361
ch4	48.507	48.516	48.269	48.293	48.397	48.694	48.697	48.856	48.856	48.927

Optical Amplifier 2

Output channel power

	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
ch 1	-9.644	-9.652	-9.648	-9.605	-9.596	-9.621	-9.634	-9.659	-9.628	-9.629
ch2	-9.621	-9.632	-9.584	-9.659	-9.599	-9.652	-9.597	-9.663	-9.611	-9.664
ch3	-9.893	-10.154	-10.276	-10.137	-10.079	-10.202	-9.845	-10.563	-10.347	-10.312
ch 4	-9.589	-9.607	-9.602	-9.598	-9.615	-9.611	-9.638	-9.626	-9.653	-9.656
	Run 11	Run 12	Run 13	Run 14	Run 15	Run 16	Run 17	Run 18	Run 19	Run 20
ch 1	-9.591	-9.605	-9.659	-9.657	-9.615	-9.597	-9.575	-9.638	-9.638	-9.677
ch 2	-9.584	-9.668	-9.593	-9.684	-9.596	-9.66	-9.577	-9.673	-9.584	-9.664
ch3	-10.112	-10.362	-9.866	-10.198	-10.326	-10.539	-10.044	-9.996	-10.016	-9.919
ch4	-9.664	-9.668	-9.64	-9.627	-9.623	-9.607	-9.589	-9.589	-9.58	-9.632
	Run 21	Run 22	Run 23	Run 24	Run 25	Run 26	Run 27	Run 28	Run 29	Run 30
ch 1	-9.6	-9.559	-9.603	-9.645	-9.648	-9.263	-9.6	-9.574	-9.624	-9.652
ch2	-9.591	-9.663	-9.602	-9.686	-9.619	-9.672	-9.611	-9.662	-9.606	-9.681
ch3	-10.016	-10.124	-10.114	-10.244	-9.963	-10.323	-10.328	-10.049	-10.383	-10.052
ch4	-9.59	-9.62	-9.62	-9.629	-9.644	-9.65	-9.677	-9.642	-9.677	-9.647
	Run 31	Run 32	Run 33	Run 34	Run 35	Run 36	Run 37	Run 38	Run 39	Run 40
ch 1	-9.657	-9.598	-9.588	-9.589	-9.626	-9.622	-9.618	-9.588	-9.602	-9.608
ch2	-9.606	-9.654	-9.607	-9.643	-9.614	-9.626	-9.572	-9.638	-9.62	-9.616
ch3	-10.053	-10.145	-10.225	-10.264	-10.111	-10.033	-10.049	-10.037	-10.019	-10.157
ch4	-9.639	-9.615	-9.601	-9.586	-9.601	-9.61	-9.609	-9.601	-9.64	-9.638
	Run 41	Run 42	Run 43	Run 44	Run 45	Run 46	Run 47	Run 48	Run 49	Run 50
ch 1	-9.621	-9.65	-9.637	-9.593	-9.596	-9.607	-9.647	-9.658	-9.598	-9.592
ch2	-9.609	-9.623	-9.669	-9.641	-9.61	-9.614	-9.63	-9.636	-9.65	-9.606
ch3	-10.08	-10.398	-10.033	-10.309	-10.042	-10.253	-10.337	-10.403	-9.968	-9.961
ch4	-9.659	-9.656	-9.674	-9.674	-9.679	-9.637	-9.638	-9.627	-9.608	-9.607

Output channel OSNR

	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6	Run 7	Run 8	Run 9	Run 10
ch 1	46.38	46.193	46.175	46.51	46.593	46.503	46.359	46.186	46.377	46.517
ch 2	47.095	46.933	47.144	46.866	47.173	46.797	46.179	46.735	47.144	46.821
ch3	48.059	48.094	48.246	48.088	48.043	48.241	48.239	47.917	48.013	48.228
ch 4	48.878	48.935	48.935	48.929	48.861	48.853	48.72	48.704	48.44	48.415
	Run 11	Run 12	Run 13	Run 14	Run 15	Run 16	Run 17	Run 18	Run 19	Run 20
ch 1	46.596	46.511	46.182	46.222	46.397	46.584	46.591	46.326	46.18	46.23
ch2	47.173	46.751	47.174	46.747	47.186	46.743	47.184	46.475	47.19	46.712
ch3	48.17	47.972	48.259	48.23	47.994	47.884	48.229	48.176	48.044	48.144
ch4	48.293	48.312	48.507	48.505	48.759	48.759	48.896	48.927	48.947	48.92
	Run 21	Run 22	Run 23	Run 24	Run 25	Run 26	Run 27	Run 28	Run 29	Run 30
ch 1	46.527	46.606	46.482	46.308	46.224	46.406	46.539	46.583	46.459	46.158
ch2	47.192	46.735	47.175	46.809	47.149	46.723	47.186	46.777	47.172	46.819
ch3	48.305	48.086	47.996	48.168	48.209	47.898	47.93	48.23	48.096	48.009
ch4	48.916	48.828	48.836	48.634	48.621	48.359	48.333	48.344	48.394	48.599
	Run 31	Run 32	Run 33	Run 34	Run 35	Run 36	Run 37	Run 38	Run 39	Run 40
ch 1	46.231	46.54	46.594	46.562	46.307	46.151	46.437	46.553	46.597	46.449
ch2	47.142	46.896	47.109	47.109	47.152	46.981	47.13	46.939	47.086	47.006
ch3	48.15	48.277	48.031	48.031	48.168	48.2	48.032	48.1	48.314	48.147
ch4	48.598	48.818	48.925	48.925	48.951	48.921	48.882	48.907	48.786	48.776
	Run 41	Run 42	Run 43	Run 44	Run 45	Run 46	Run 47	Run 48	Run 49	Run 50
ch 1	46.29	46.259	46.445	46.599	45.561	46.435	46.15	46.283	46.568	46.603
ch2	47.041	47.065	46.997	46.954	47.072	47.026	47.01	47.075	46.94	47.128
ch3	48.04	48.128	48.287	48.017	48.04	48.187	48.046	47.955	48.136	48.359
ch4	48.503	48.514	48.267	48.259	48.393	48.688	48.695	48.853	48.864	48.928

Performance data for the wavelength de-multiplexer (DM1)

Wavelength Demux										
Output channel power										
	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**Run 6**	**Run 7**	**Run 8**	**Run 9**	**Run 10**
ch 1	-9.645	-9.611	-9.564	-9.598	-9.627	-9.657	-9.624	-9.606	-9.566	-9.67
ch 2	-9.627	-9.638	-9.593	-9.645	-9.617	-9.643	-9.619	-9.646	-9.631	-9.65
ch 3	-9.895	-10.202	-10.311	-10.097	-10.075	-10.22	-9.836	-10.525	-10.357	-10.342
ch 4	-9.64	-9.669	-9.645	-9.652	-9.639	-9.617	-9.609	-9.624	-9.603	-9.605
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
ch 1	-9.656	-9.662	-9.612	-9.612	-9.583	-9.638	-9.662	-9.635	-9.6	-9.607
ch 2	-9.587	-9.653	-9.604	-9.667	-9.616	-9.649	-9.602	-9.674	-9.6	-9.641
ch 3	-10.094	-10.353	-9.907	-10.206	-10.306	-10.562	-10.061	-9.971	-9.997	-9.92
ch 4	-9.611	-9.599	-9.616	-9.6	-9.639	-9.628	-9.647	-9.666	-9.666	9.669
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
ch 1	-9.608	-9.621	-9.655	-9.626	-9.381	-9.59	-9.6	-9.63	-9.562	-9.618
ch 2	-9.615	-9.655	-9.606	-9.669	-9.611	-9.678	-9.611	-9.668	-9.614	-9.673
ch 3	-10.042	-10.086	-10.102	-10.27	-9.953	-10.27	-10.343	-10.077	-10.391	-10.035
ch 4	-9.636	-9.636	-9.622	-9.599	-9.603	-9.594	-9.613	-9.59	-9.617	-9.636
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
ch 1	-9.597	-9.62	-9.65	-9.641	-9.61	-9.606	-9.593	-9.607	-9.643	-9.651
ch 2	-9.607	-9.662	-9.6	-9.649	-9.603	-9.646	-9.569	-9.637	-9.604	-9.641
ch 3	-10.09	-10.158	-10.188	-10.267	-10.13	-9.996	-10.036	-10.048	-10.062	-10.117
ch 4	-9.62	-9.646	-9.666	-9.653	-9.664	-9.659	-9.639	-9.616	-9.621	-9.616
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
ch 1	-9.611	-9.596	-9.615	-9.634	-9.651	-9.649	-9.585	-9.589	-9.621	-9.642
ch 2	-9.605	-9.641	-9.624	-9.651	-9.609	-9.617	-9.633	-9.636	-9.631	-9.605
ch 3	-10.052	-10.464	-10.032	-10.266	-10.053	-10.291	-10.239	-10.385	-9.984	-9.955
ch 4	-9.616	-9.609	-9.611	-9.596	-9.597	-9.623	-9.619	-9.633	-9.668	-9.672
Output channel OSNR										
	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**Run 6**	**Run 7**	**Run 8**	**Run 9**	**Run 10**
ch 1	46.32	46.523	46.386	46.557	46.416	46.095	46.358	46.538	46.581	46.564
ch 2	47.08	47.064	47.12	47.001	47.143	46.896	47.172	46.806	47.141	46.958
ch 3	47.989	47.904	48.063	48.264	48.041	48.036	48.337	48.032	47.94	48.018
ch 4	48.442	48.171	48.212	48.447	48.757	48.767	48.927	48.906	48.917	48.925
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
ch 1	46.143	46.099	46.543	46.582	46.578	46.383	46.111	46.374	46.55	46.572
ch 2	47.174	46.83	47.159	46.75	46.75	46.664	47.164	46.8	47.175	46.7
ch 3	48.298	47.977	48.077	48.227	48.177	47.909	48.052	48.347	48.133	48.061
ch 4	48.933	48.947	48.904	48.896	46.684	48.674	48.289	48.206	48.189	48.559
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
ch 1	46.545	46.36	46.134	46.412	46.58	46.57	46.537	46.111	46.158	46.552
ch 2	47.162	46.664	47.163	46.646	47.165	46.666	47.174	46.631	47.169	46.674
ch 3	48.181	48.291	48.058	48.022	48.306	48.131	47.928	48.187	48.187	48.028
ch 4	48.552	48.841	48.833	48.92	48.926	48.931	48.931	48.939	48.939	48.845
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
ch 1	46.579	46.526	46.333	46.097	46.432	46.557	46.578	46.529	46.3	46.165
ch 2	47.179	46.742	47.183	46.843	47.173	46.694	47.183	46.794	47.171	46.904
ch 3	47.964	48.13	48.157	47.974	47.98	48.347	48.137	48.042	48.134	48.303
ch 4	48.844	48.554	48.207	48.216	48.292	48.671	48.671	48.697	48.901	48.955
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
ch 1	46.458	46.592	46.574	46.305	46.076	46.212	46.583	46.589	46.503	46.269
ch 2	47.141	46.989	47.128	46.839	47.158	46.951	47.133	47.046	47.089	47.112
ch 3	48.122	47.922	48.268	48.179	48.061	47.997	48.245	48.015	48.04	48.235
ch 4	48.891	48.935	48.936	48.945	48.942	48.928	48.774	48.773	48.391	48.198

Input combined power										
	Run 1	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**Run 6**	**Run 7**	**Run 8**	**Run 9**	**Run 10**
	-2.593	-2.65	-2.664	-2.645	-2.633	-2.655	-2.594	-2.713	-2.68	-2.682
	Run 11	**Run 12**	**Run 13**	**Run 14**	**Run 15**	**Run 16**	**Run 17**	**Run 18**	**Run 19**	**Run 20**
	-2.639	-2.68	-2.598	-2.666	-2.675	-2.711	-2.622	-2.623	-2.616	-2.609
	Run 21	**Run 22**	**Run 23**	**Run 24**	**Run 25**	**Run 26**	**Run 27**	**Run 28**	**Run 29**	**Run 30**
	-2.63	-2.637	-2.634	-2.663	-2.612	-2.67	-2.679	-2.634	-2.696	-2.629
	Run 31	**Run 32**	**Run 33**	**Run 34**	**Run 35**	**Run 36**	**Run 37**	**Run 38**	**Run 39**	**Run 40**
	-2.633	-2.663	-2.662	-2.661	-2.636	-2.626	-2.617	-2.619	-2.629	-2.646
	Run 41	**Run 42**	**Run 43**	**Run 44**	**Run 45**	**Run 46**	**Run 47**	**Run 48**	**Run 49**	**Run 50**
	-2.624	-2.698	-2.637	-2.671	-2.632	-2.662	-2.678	-2.692	-2.614	-2.627

Appendix C

Code Listing for NM Tool Implementation

```
//The class structure for the NE class. It's the parent class, which has all the common
//attributes and functions. Compiled by Carol Baker on 25 June 2001.
#ifndef _NE_h
#define _NE_h
#include <iostream.h>
#include <string.h>

//global variables
int FailAlarms[5];
int DegAlarms[5];
int DGNE_Notifications[5];
int SGNE_Notifications[5];
int NMS_Notifications;

//Failure and degradation thresholds
const float ChPowerDegThreshold = -25.0;
const float ChPowerFailThreshold = -30.0;
const float ChOSNRDegThreshold = 25.0;
const float ChOSNRFailThreshold = 20.0;
const float CombPowerDegThreshold = 0.0;
const float CombPowerFailThreshold = -2.0;
const int WAVELENGTHS = 8;
```

```
class NE
{
        public:
                NE (char *n, char *d, char *s);
                void Initialize();
                void ReadData();
                void ChPowerPerfTest();
                void ChOSNRPerfTest();
                void ChFreqPerfTest();
                void CombPowerPerfTest();
                void Print();
                void SetChFreq();
                void UpdateAlarms();
                char* ShowDomainName();
                char* ShowSubnetName();
                char* ShowName();

        protected:
                char Name[6];           //holds the name of the NE
                char DomainName[8];     //holds the domain name of the NE
                char SubnetName[8];     //holds the subnet name of the NE
                int DGNE_Flag;          //to indicate which NE is the domain GNE
                int SGNE_Flag;          //to indicate which NE is the subnet GNE
                double ChPower[WAVELENGTHS];
                double ChOSNR[WAVELENGTHS];
                double ChFreq[WAVELENGTHS];
                double CurrentChFreq[WAVELENGTHS];
                float CombPower;
                //to indicate that an SD alarm has been raised
                bool SD_Flag[WAVELENGTHS];
                //to indicate that an LOS alarm has been raised
                bool LOS_Flag[WAVELENGTHS];
                //to indicate that an OSNR_D alarm has been raised
                bool OSNRD_Flag[WAVELENGTHS];
                //to indicate that an OSNR_F alarm has been raised
                bool OSNRF_Flag[WAVELENGTHS];
                //to indicate that an SD alarm has been raised
                bool FAD_Flag[WAVELENGTHS];
                //to indicate that an LOS alarm has been raised
```

```
bool FAF_Flag[WAVELENGTHS];
//to indicate that an SD alarm has been raised
bool C_SD_Flag;
//to indicate that an LOS alarm has been raise
bool C_LOS_Flag;
//will be used for persistency analysis checking at the domain level
bool PPersist_Flag1[WAVELENGTHS];
//will be used for persistency analysis checking at the subnet level
bool PPersist_Flag2[WAVELENGTHS];
//will be used for persistency analysis checking at the domain level
bool OPersist_Flag1[WAVELENGTHS];
//will be used for persistency analysis checking at the subnet level
bool OPersist_Flag2[WAVELENGTHS];
//will be used for persistency analysis checking at the domain level
bool FPersist_Flag1[WAVELENGTHS];

//will be used for persistency analysis checking at the subnet level
bool FPersist_Flag2[WAVELENGTHS];
//will be used for persistency analysis checking at the domain level
bool CPersist_Flag1;
//will be used for persistency analysis checking at the subnet level
bool CPersist_Flag2;
//tracks the number of power failure threshold crossings
int NumChPFailCrossings[WAVELENGTHS];
//tracks the number of power degradation threshold crossings
int NumChPDegCrossings[WAVELENGTHS];
//tracks the number of failure threshold crossings
int NumOSNRFailCrossings[WAVELENGTHS];
//tracks the number of degradation threshold crossings
int NumOSNRDegCrossings[WAVELENGTHS];
//tracks the number of frequnecy failure threshold crossings
int NumChFFailCrossings[WAVELENGTHS];
//tracks the number of frequency degradation threshold crossings
int NumChFDegCrossings[WAVELENGTHS];
//tracks the number of power failure threshold crossings
int NumPFailCrossings;
//tracks the number of power degradation threshold crossings
int NumPDegCrossings;
//keeps a count of the number of SD alarms raised
```

```
        int SD_Count;
        //keeps a count of the number of LOS alarms raised
        int LOS_Count;
        //keeps a count of the number of OSNR_F alarms raised
        int OSNRF_Count;
        //keeps a count of the number of OSNR_D alarms raised
        int OSNRD_Count;
        //keeps a count of the number of FAD alarms raised
        int FAD_Count;
        //keeps a count of the number of FAF alarms raised
        int FAF_Count;
        //keeps a count of the number of SD alarms raised
        int C_SD_Count;
        //keeps a count of the number of LOS alarms raised
        int C_LOS_Count;
        //int NotSentCounter;
}; #endif
//This is the definition file for the NE class. Compiled by Carol Baker, last updated 26 June 2001

//# include <iostream.h>
//# include <string.h>

# include "NE.h"
# include "WMux.h"
# include "DMux.h"



//The class definitions for the parent class
NE::NE(char *n, char *d, char *s)
{
    strcpy(Name, n);
    strcpy(DomainName, d);
    strcpy(SubnetName,s);
}


//This function initializes the channel frequency depending on the number of wavelengths
//in the system. Channels are spaced 100Ghz apart and are centered at 193.1Thz
void NE::SetChFreq()
```

```
{
        if (WAVELENGTHS == 4)
        {
                ChFreq[1] = 192.95;
                ChFreq[2] = 193.05;
                ChFreq[3] = 193.15;
                ChFreq[4] = 193.25;
        }

        if (WAVELENGTHS == 8)
        {
                ChFreq[1] = 192.75;
                ChFreq[2] = 192.85;
                ChFreq[3] = 192.95;
                ChFreq[4] = 193.05;
                ChFreq[5] = 193.15;
                ChFreq[6] = 193.25;
                ChFreq[7] = 193.35;
                ChFreq[8] = 193.45;
        }

        if (WAVELENGTHS == 16)
        {
                double ChFreq[WAVELENGTHS];
                ChFreq[1] = 192.35;
                ChFreq[2] = 192.45;
                ChFreq[3] = 192.55;
                ChFreq[4] = 192.65;
                ChFreq[5] = 192.75;
                ChFreq[6] = 192.85;
                ChFreq[7] = 192.95;
                ChFreq[8] = 193.05;
                ChFreq[9] = 193.15;
                ChFreq[10] = 193.25;
                ChFreq[11] = 193.35;
                ChFreq[12] = 193.45;
                ChFreq[13] = 193.55;
                ChFreq[14] = 193.65;
                ChFreq[15] = 193.75;
                ChFreq[16] = 193.85;
```

```
    }
}//end setting channel frequencies



//Calculates the total number of failure and degradation alarms that were generated in each
//domain or subnet
void NE::UpdateAlarms()
{
    //calculate number of degradation alarms generated in each domain
    int NumDegAlarms = SD_Count + OSNRD_Count + FAD_Count + C_SD_Count;
    if (NumDegAlarms != 0)
    {
        if (strcmp(DomainName, "Domain1") == 0)
                DegAlarms[0] = DegAlarms[0] + NumDegAlarms;

        if (strcmp(DomainName, "Domain2") == 0)
            DegAlarms[1] = DegAlarms[1] + NumDegAlarms;

        if (strcmp(DomainName, "Domain3") == 0)
            DegAlarms[2] = DegAlarms[2] + NumDegAlarms;

        if (strcmp(DomainName, "Domain4") == 0)
            DegAlarms[3] = DegAlarms[3] + NumDegAlarms;

        if (strcmp(DomainName, "Domain5") == 0)
            DegAlarms[4] = DegAlarms[4] + NumDegAlarms;
    }//end calculate degradation alarms


    //calculates number of failure alarms generated in each domain
    int NumFailAlarms = LOS_Count + OSNRF_Count + FAF_Count + C_LOS_Count;
    if (NumFailAlarms != 0)
    {
        if (strcmp(DomainName, "Domain1") == 0)
            FailAlarms[0] = FailAlarms[0] + NumFailAlarms;

        if (strcmp(DomainName, "Domain2") == 0)
            FailAlarms[1] = FailAlarms[1] + NumFailAlarms;
```

```
		if (strcmp(DomainName, "Domain3") == 0)
			FailAlarms[2] = FailAlarms[2] + NumFailAlarms;

		if (strcmp(DomainName, "Domain4") == 0)
			FailAlarms[3] = FailAlarms[3] + NumFailAlarms;

		if (strcmp(DomainName, "Domain5") == 0)
			FailAlarms[4] = FailAlarms[4] + NumFailAlarms;
	}//end calculate failure alarms
}//end clacluate number of alarms



//Prints the number of alarms generated in each domain as well as the number of
//notifications sent
void NE::Print()
{
	cout << "NEs in Domain1 generated " << DegAlarms[0] << " degradation alarms." <<
endl;
	cout << "NEs in Domain2 generated " << DegAlarms[1] << " degradation alarms." <<
endl;
	cout << "NEs in Domain1 generated " << FailAlarms[0] << " failure alarms." << endl;
	cout << "NEs in Domain2 generated " << FailAlarms[1] << " failure alarms." << endl;
	cout << "Domain1 GNE received " << DGNE_Notifications[0] << " notifications." <<
endl;
	cout << "Domain2 GNE received " << DGNE_Notifications[1] << " notifications." <<
endl;
	cout << "Subnet1 GNE received " << SGNE_Notifications[0] << " notifications." <<
endl;
	cout << "Subnet2 GNE received " << SGNE_Notifications[1] << " notifications." <<
endl;
	cout << "The NMS received" << NMS_Notifications << "notifications." << endl;
}


//This functions is just used to identify which domain an NE belongs to
char* NE::ShowDomainName()
{
	return DomainName;
}
```

```
//This function is used to identify the subnetwork to which the NE belongs
char* NE::ShowSubnetName()
{
	return SubnetName;
}


//This function identifies the name of the NE
char* NE::ShowName()
{
	return Name;
}



//Just initailizing variables, this is done only once at the beginning of the simulation
void NE::Initialize()
{
	NMS_Notifications = 0;
	SD_Count = 0;
	LOS_Count = 0;
	OSNRF_Count = 0;
	OSNRD_Count = 0;
	FAD_Count = 0;
	FAF_Count = 0;
	LOS_Count = 0;
	C_LOS_Count = 0;
	C_SD_Count = 0;
	NumPFailCrossings = 0;
	NumPDegCrossings = 0;
	C_SD_Flag = false;
	C_LOS_Flag = false;
	CPersist_Flag1 = false;
	CPersist_Flag2 = false;
	CombPower = 0.0;

	for (int i = 0; i < 5; i++)
	{
```

```
        DGNE_Notifications[i] = 0;
}

for (int j= 0; j < 5; j++)
{
        SGNE_Notifications[j] = 0;
}

for (int k = 0; k < WAVELENGTHS; k++)
{
        NumChPFailCrossings[k] = 0;
}

for (int l = 0; l < WAVELENGTHS; l++)
{
        NumChPDegCrossings[l] = 0;
}

for (int m = 0; m < WAVELENGTHS; m++)
{
        SD_Flag[m] = false;
}

for (int n = 0; n < WAVELENGTHS; n++)
{
        LOS_Flag[n] = false;
}

for (int o = 0; o < WAVELENGTHS; o++)
{
        PPersist_Flag1[o] = false;
}

for (int p = 0; p < WAVELENGTHS; p++)
{
        PPersist_Flag2[p] = false;
}

for (int q = 0; q < WAVELENGTHS; q++)
{
```

```
            NumOSNRDegCrossings[q] = 0;
}

for (int r = 0; r < WAVELENGTHS; r++)
{
            NumOSNRFailCrossings[r] = 0;
}

for (int s = 0; s < WAVELENGTHS; s++)
{
            OSNRF_Flag[s] = false;
}

for (int t = 0; t < WAVELENGTHS; t++)
{
            OSNRD_Flag[t] = false;
}

for (int u = 0; u < WAVELENGTHS; u++)
{
            OPersist_Flag1[u] = false;
}

for (int v = 0; v < WAVELENGTHS; v++)
{
            OPersist_Flag2[v] = false;
}

for (int w = 0; w < WAVELENGTHS; w++)
{
            NumChFDegCrossings[w] = 0;
}

for (int x = 0; x < WAVELENGTHS; x++)
{
            NumChFFailCrossings[x] = 0;
}

for (int y = 0; y < WAVELENGTHS; y++)
{
```

```
        FAD_Flag[y] = false;
    }

    for (int z = 0; z < WAVELENGTHS; z++)
    {
        FAF_Flag[z] = false;
    }

    for (int a = 0; a < WAVELENGTHS; a++)
    {
        FPersist_Flag1[a] = false;
    }

    for (int b = 0; b < WAVELENGTHS; b++)
    {
        FPersist_Flag2[b] = false;
    }

    for (int c = 0; c < 5; c++)
    {
        FailAlarms[c] = 0;
    }

    for (int d = 0; d < 5; d++)
    {
        DegAlarms[d] = 0;
    }
}//end initialize variables



//Read-in in performance data from the keyboard
void NE::ReadData()
{
    void NE::ReadData()
{
        cout << "Key in channel power performance data!" << endl;
        for (int i = 0; i < WAVELENGTHS; i++)
        {
            cin >> ChPower[i];
```

```
                }

                cout << "Key in input channel OSNR performance data!" << endl;
                for (int j = 0; j < WAVELENGTHS; j++)
                {
                        cin >> ChOSNR[j];
                }

                cout << "Input the combined power." << endl;
                cin >> CombPower;
}



//Tests the power performance for each channel. This is at the input of the multiplexer, but
//I'll check later whether they can be used for other NEs.
void NE::ChPowerPerfTest()
{
        for(int i = 0; i < WAVELENGTHS; i++)
        {
                //checks whether channel power failure or degradation has occurred
                if(ChPower[i] < ChPowerFailThreshold)
                {
                        NumChPFailCrossings[i] += 1;
                        NumChPDegCrossings[i] += 1;
                }

                if((ChPower[i] > ChPowerFailThreshold) && (ChPower[i] <
ChPowerDegThreshold))
                {
                        NumChPFailCrossings[i] = 0;
                        NumChPDegCrossings[i] += 1;
                        if (SD_Flag[i] == true)
                                PPersist_Flag1[i] = true;

                        if ((PPersist_Flag1[i] == true) && (SD_Flag[i] == false))
                                PPersist_Flag2[i] = true;
                }

                if (ChPower[i] > ChPowerDegThreshold)
                {
```

```
        NumChPFailCrossings[i] = 0;
        NumChPDegCrossings[i] = 0;
        LOS_Flag[i] = false;
        SD_Flag[i] = false;
        PPersist_Flag1[i] = false;
        PPersist_Flag2[i] = false;
}//end check for channel power failure or degradation



//generates alarms and sends notifications
if (NumChPFailCrossings[i] == 3)
{
        NumChPFailCrossings[i] = 0;
        NumChPDegCrossings[i] = 0;
        if (strcmp(DomainName, "Domain1") == 0)
                DGNE_Notifications[0] += 1;

        if (strcmp(DomainName, "Domain2") == 0)
                DGNE_Notifications[1] += 1;

        if (strcmp(DomainName, "Domain3") == 0)
                DGNE_Notifications[2] += 1;

        if (strcmp(DomainName, "Domain4") == 0)
                DGNE_Notifications[3] += 1;

        if (strcmp(DomainName, "Domain5") == 0)
                DGNE_Notifications[4] += 1;

        LOS_Flag[i] = true;
        LOS_Count += 1;
}

if (NumChPDegCrossings[i] == 5)
{
        NumChPDegCrossings[i] = 0;

        if (strcmp(DomainName, "Domain1") == 0)
                DGNE_Notifications[0] += 1;
```

```
        if (strcmp(DomainName, "Domain2") == 0)
                DGNE_Notifications[1] += 1;

        if (strcmp(DomainName, "Domain3") == 0)
                DGNE_Notifications[2] += 1;

        if (strcmp(DomainName, "Domain4") == 0)
                DGNE_Notifications[3] += 1;

        if (strcmp(DomainName, "Domain5") == 0)
                DGNE_Notifications[4] += 1;

        SD_Flag[i] = true;
        SD_Count += 1;
}//end generate alarms and send notifications


//checks whether channel power degradation persists
if ((SD_Flag[i] == true) && (PPersist_Flag1[i] == true))
{

        if (strcmp(SubnetName, "Subnet1") == 0)
                SGNE_Notifications[0] += 1;

        if (strcmp(SubnetName, "Subnet2") == 0)
                SGNE_Notifications[1] += 1;

        if (strcmp(SubnetName, "Subnet3") == 0)
                SGNE_Notifications[2] += 1;

        if (strcmp(SubnetName, "Subnet4") == 0)
                SGNE_Notifications[3] += 1;

        if (strcmp(SubnetName, "Subnet5") == 0)
                SGNE_Notifications[4] += 1;

        SD_Flag[i] = false;
        PPersist_Flag2[i] = false;
}
```

```
        if ((SD_Flag[i] == true) && (PPersist_Flag2[i] == true))
        {
            NMS_Notifications += 1;
            SD_Flag[i] = false;
        }//end channel power degradation persistence check
    }//end for loop
}//end channel power performance checking



//Tests the OSNR performance for each channel. This is at the input of the multiplexer, but
//I'll check later whether they can be used for other NEs.
void NE::ChOSNRPerfTest()
{
    for(int i = 0; i < WAVELENGTHS; i++)
    {
        //checks whether OSNR failure or degradation has occurred
        if(ChOSNR[i] < ChOSNRFailThreshold)
        {
            NumOSNRFailCrossings[i] += 1;
            NumOSNRDegCrossings[i] += 1;
        }

        if((ChOSNR[i] > ChOSNRFailThreshold) && (ChOSNR[i] <
ChOSNRDegThreshold))
        {
            NumOSNRFailCrossings[i] = 0;
            NumOSNRDegCrossings[i] += 1;
            if (SD_Flag[i] = true)
                OPersist_Flag1[i] = true;

            if ((OPersist_Flag1[i] = true) && (SD_Flag[i] = false))
                OPersist_Flag2[i] = true;
        }

        if (ChOSNR[i] > ChOSNRDegThreshold)
        {
            NumOSNRFailCrossings[i] = 0;
            NumOSNRDegCrossings[i] = 0;
            OSNRF_Flag[i] = false;
            OSNRD_Flag[i] = false;
```

```
        OPersist_Flag1[i] = false;
        OPersist_Flag2[i] = false;
}//end check for OSNR failure or degradation



//generates alarms and sends notifications
if (NumOSNRFailCrossings[i] == 3)
{
        NumOSNRFailCrossings[i] = 0;
        NumOSNRDegCrossings[i] = 0;
        if (strcmp(DomainName, "Domain1") == 0)
                DGNE_Notifications[0] += 1;

        if (strcmp(DomainName, "Domain2") == 0)
                DGNE_Notifications[1] += 1;

        if (strcmp(DomainName, "Domain3") == 0)
                DGNE_Notifications[2] += 1;

        if (strcmp(DomainName, "Domain4") == 0)
                DGNE_Notifications[3] += 1;

        if (strcmp(DomainName, "Domain5") == 0)
                DGNE_Notifications[4] += 1;

        OSNRF_Flag[i] = true;
        OSNRF_Count += 1;
}

if (NumOSNRDegCrossings[i] == 5)
{
        NumOSNRDegCrossings[i] = 0;

        if (strcmp(DomainName, "Domain1") == 0)
                DGNE_Notifications[0] += 1;

        if (strcmp(DomainName, "Domain2") == 0)
                DGNE_Notifications[1] += 1;

        if (strcmp(DomainName, "Domain3") == 0)
```

```
                DGNE_Notifications[2] += 1;

        if (strcmp(DomainName, "Domain4") == 0)
                DGNE_Notifications[3] += 1;

        if (strcmp(DomainName, "Domain5") == 0)
                DGNE_Notifications[4] += 1;

        OSNRD_Flag[i] = true;
        OSNRD_Count += 1;
}//end generate alarms and send notifications


//checks whether OSNR degradation persists
if ((OSNRD_Flag[i] == true) && (OPersist_Flag1[i] == true))
{

        if (strcmp(SubnetName, "Subnet1") == 0)
                SGNE_Notifications[0] += 1;

        if (strcmp(SubnetName, "Subnet2") == 0)
                SGNE_Notifications[1] += 1;

        if (strcmp(SubnetName, "Subnet3") == 0)
                SGNE_Notifications[2] += 1;

        if (strcmp(SubnetName, "Subnet4") == 0)
                SGNE_Notifications[3] += 1;

        if (strcmp(SubnetName, "Subnet5") == 0)
                SGNE_Notifications[4] += 1;

        OPersist_Flag1[i] = false;
}

if ((OSNRD_Flag[i] == false) && (OPersist_Flag2[i] == true))
{
        NMS_Notifications += 1;
        OSNRD_Flag[i] = false;
        PPersist_Flag2[i] = false;
```

```
        }//end OSNR persistence check
    }//end for loop
}//end OSNR performance checking



//Tests the frequency performance for each channel. This is at the input of the multiplexer, but
//I'll check later whether they can be used for other NEs.
void NE::ChFreqPerfTest()
{
    for(int i = 0; i < WAVELENGTHS; i++)
    {
        //checks whether channel frequency failure or degradation has occurred
        if((CurrentChFreq[i] < (ChFreq[i] - 0.01)) || (CurrentChFreq[i] > (ChFreq[i] +
0.01)))
        {
            NumChFFailCrossings[i] += 1;
            NumChFDegCrossings[i] += 1;
        }

        if((CurrentChFreq[i] < (ChFreq[i] - 0.005)) || (CurrentChFreq[i] > (ChFreq[i] +
0.005)))
        {
            NumChFFailCrossings[i] = 0;
            NumChFDegCrossings[i] += 1;
            if (FAD_Flag[i] == true)
                FPersist_Flag1[i] = true;

            if ((FPersist_Flag1[i] == true) && (FAD_Flag[i] == false))
                FPersist_Flag2[i] = true;
        }

        if ((CurrentChFreq[i] > (ChFreq[i] - 0.005)) && (CurrentChFreq[i] < (ChFreq[i]
+ 0.005)))
        {
            NumChFFailCrossings[i] = 0;
            NumChFDegCrossings[i] = 0;
            FAF_Flag[i] = false;
            FAD_Flag[i] = false;
            FPersist_Flag1[i] = false;
```

```
          FPersist_Flag2[i] = false;
}//end check for channel frequency failure or degradation



//generates alarms and sends notifications
if (NumChFFailCrossings[i] == 3)
{
          NumChFFailCrossings[i] = 0;
          NumChFDegCrossings[i] = 0;
          if (strcmp(DomainName, "Domain1") == 0)
                    DGNE_Notifications[0] += 1;

          if (strcmp(DomainName, "Domain2") == 0)
                    DGNE_Notifications[1] += 1;

          if (strcmp(DomainName, "Domain3") == 0)
                    DGNE_Notifications[2] += 1;

          if (strcmp(DomainName, "Domain4") == 0)
                    DGNE_Notifications[3] += 1;

          if (strcmp(DomainName, "Domain5") == 0)
                    DGNE_Notifications[4] += 1;

          FAF_Flag[i] = true;
          FAF_Count += 1;
}

if (NumChFDegCrossings[i] == 5)
{
          NumChFDegCrossings[i] = 0;

          if (strcmp(DomainName, "Domain1") == 0)
                    DGNE_Notifications[0] += 1;

          if (strcmp(DomainName, "Domain2") == 0)
                    DGNE_Notifications[1] += 1;

          if (strcmp(DomainName, "Domain3") == 0)
                    DGNE_Notifications[2] += 1;
```

```
        if (strcmp(DomainName, "Domain4") == 0)
                DGNE_Notifications[3] += 1;

        if (strcmp(DomainName, "Domain5") == 0)
                DGNE_Notifications[4] += 1;

        FAD_Flag[i] = true;
        FAD_Count += 1;
}//end generate alarms and send notifications


//checks whether channel power degradation persists
if ((FAD_Flag[i] == true) && (FPersist_Flag1[i] == true))
{

        if (strcmp(SubnetName, "Subnet1") == 0)
                SGNE_Notifications[0] += 1;

        if (strcmp(SubnetName, "Subnet2") == 0)
                SGNE_Notifications[1] += 1;

        if (strcmp(SubnetName, "Subnet3") == 0)
                SGNE_Notifications[2] += 1;

        if (strcmp(SubnetName, "Subnet4") == 0)
                SGNE_Notifications[3] += 1;

        if (strcmp(SubnetName, "Subnet5") == 0)
                SGNE_Notifications[4] += 1;

        FAD_Flag[i] = false;
}

if ((FAD_Flag[i] == false) && (FPersist_Flag2[i] == true))
{
        NMS_Notifications += 1;
        FAD_Flag[i] = false;
        FPersist_Flag2[i] = false;
}//end channel frequency degradation persistence check
```

```
    }//end for loop
}//end channel frequency performance checking



//Tests the combined power performance for each fiber. This is at the output of the multiplexer, but
//I'll check later whether they can be used for other NEs.
void NE::CombPowerPerfTest()
{
    //checks whether combined channel power failure or degradation has occurred
    if(CombPower < CombPowerFailThreshold)
    {
        NumPFailCrossings += 1;
        NumPDegCrossings += 1;
    }

    if((CombPower > CombPowerFailThreshold) && (CombPower <
CombPowerDegThreshold))
    {
        NumPFailCrossings = 0;
        NumPDegCrossings += 1;
        if (C_SD_Flag == true)
            CPersist_Flag1 = true;

        if ((CPersist_Flag1 == true) && (C_SD_Flag == false))
            CPersist_Flag2 = true;
    }

    if (CombPower > CombPowerDegThreshold)
    {
        NumPFailCrossings = 0;
        NumPDegCrossings = 0;
        C_LOS_Flag = false;
        C_SD_Flag = false;
        CPersist_Flag1 = false;
        CPersist_Flag2 = false;
    }//end check for channel power failure or degradation


    //generates alarms and sends notifications
```

```
if (NumPFailCrossings == 3)
{
        NumPFailCrossings = 0;
        NumPDegCrossings = 0;
        if (strcmp(DomainName, "Domain1") == 0)
                DGNE_Notifications[0] += 1;

        if (strcmp(DomainName, "Domain2") == 0)
                DGNE_Notifications[1] += 1;

        if (strcmp(DomainName, "Domain3") == 0)
                DGNE_Notifications[2] += 1;

        if (strcmp(DomainName, "Domain4") == 0)
                DGNE_Notifications[3] += 1;

        if (strcmp(DomainName, "Domain5") == 0)
                DGNE_Notifications[4] += 1;

        C_LOS_Flag = true;
        C_LOS_Count += 1;
}

if (NumPDegCrossings == 5)
{
        NumPDegCrossings = 0;

        if (strcmp(DomainName, "Domain1") == 0)
                DGNE_Notifications[0] += 1;

        if (strcmp(DomainName, "Domain2") == 0)
                DGNE_Notifications[1] += 1;

        if (strcmp(DomainName, "Domain3") == 0)
                DGNE_Notifications[2] += 1;

        if (strcmp(DomainName, "Domain4") == 0)
                DGNE_Notifications[3] += 1;

        if (strcmp(DomainName, "Domain5") == 0)
```

```
                    DGNE_Notifications[4] += 1;

                C_SD_Flag = true;
                C_SD_Count += 1;
            }//end generate alarms and send notifications



            //checks whether channel power degradation persists
            if ((C_SD_Flag == true) && (CPersist_Flag1 == true))
            {

                if (strcmp(SubnetName, "Subnet1") == 0)
                    SGNE_Notifications[0] += 1;

                if (strcmp(SubnetName, "Subnet2") == 0)
                    SGNE_Notifications[1] += 1;

                if (strcmp(SubnetName, "Subnet3") == 0)
                    SGNE_Notifications[2] += 1;

                if (strcmp(SubnetName, "Subnet4") == 0)
                    SGNE_Notifications[3] += 1;

                if (strcmp(SubnetName, "Subnet5") == 0)
                    SGNE_Notifications[4] += 1;

                C_SD_Flag = false;
            }

            if ((C_SD_Flag == true) && (CPersist_Flag2 == true))
            {
                NMS_Notifications += 1;
                C_SD_Flag = false;
                CPersist_Flag2 = false;
            }//end channel power degradation persistence check
        }//end channel power performance checking



        //the executable program
        void main()
```

```
{

        //Creating the NEs that make up the differnet domains and subnets
        NE NE1("Mux1", "Domain1", "Subnet1");
        NE NE2("OA1", "Domain1", "Subnet1");
        NE NE3("OXC1", "Domain1", "Subnet1");
        NE NE4("OADM1", "Domain2", "Subnet1");
        NE NE5("OA2", "Domain2", "Subnet1");
        NE NE6("DMux1", "Domain2", "Subnet1");

        /*if (strcmp(NE1.ShowDomainName(), "Domain2") == 0)
                cout << "In this domain" << endl;
        else
                cout << "Not in this domain" << endl;
        */

        //Initializing all the varaibles associated with the NEs
        NE1.Initialize();
        NE2.Initialize();
        NE3.Initialize();
        NE4.Initialize();
        NE5.Initialize();
        NE6.Initialize();

        //initializes the channel frequencies
        NE1.SetChFreq();
        NE2.SetChFreq();
        NE3.SetChFreq();
        NE4.SetChFreq();
        NE5.SetChFreq();
        NE6.SetChFreq();

        int Runs = 50;

        for (int i = 0; i < Runs; i++)
        {
                //read the current input data for each NE
                NE1.ReadData();
                NE2.ReadData();
                NE3.ReadData();
```

```
NE4.ReadData();
NE5.ReadData();
NE6.ReadData();


//check for threshold crossings and alarms for each NE
//checks for channel power failures or crossings
NE1.ChPowerPerfTest();
NE2.ChPowerPerfTest();
NE3.ChPowerPerfTest();
NE4.ChPowerPerfTest();
NE5.ChPowerPerfTest();
NE6.ChPowerPerfTest();

//checks for channel OSNR failures or degradations
NE1.ChOSNRPerfTest();
NE2.ChOSNRPerfTest();
NE3.ChOSNRPerfTest();
NE4.ChOSNRPerfTest();
NE5.ChOSNRPerfTest();
NE6.ChOSNRPerfTest();


//checks for failure or degradation of the multiplex signals
NE1.CombPowerPerfTest();
NE2.CombPowerPerfTest();
NE3.CombPowerPerfTest();
NE4.CombPowerPerfTest();
NE5.CombPowerPerfTest();
NE6.CombPowerPerfTest();


//reads the current output data
NE1.ReadData();
NE2.ReadData();
NE3.ReadData();
NE4.ReadData();
NE5.ReadData();
NE6.ReadData();
```

```
//check for threshold crossings and alarms for each NE
//checks for channel power failures or crossings
NE1.ChPowerPerfTest();
NE2.ChPowerPerfTest();
NE3.ChPowerPerfTest();
NE4.ChPowerPerfTest();
NE5.ChPowerPerfTest();
NE6.ChPowerPerfTest();

//checks for channel OSNR failures or degradations
NE1.ChOSNRPerfTest();
NE2.ChOSNRPerfTest();
NE3.ChOSNRPerfTest();
NE4.ChOSNRPerfTest();
NE5.ChOSNRPerfTest();
NE6.ChOSNRPerfTest();


//checks for failure or degradation of the multiplex signals
NE1.CombPowerPerfTest();
NE2.CombPowerPerfTest();
NE3.CombPowerPerfTest();
NE4.CombPowerPerfTest();
NE5.CombPowerPerfTest();
NE6.CombPowerPerfTest();


//updating the number of alarms generated
NE1.UpdateAlarms();
NE2.UpdateAlarms();
NE3.UpdateAlarms();
NE4.UpdateAlarms();
NE5.UpdateAlarms();
NE6.UpdateAlarms();


//printing the number of alarms generated in each domain and also the number
//of notifications sent
NE1.Print();
```

```
                NE2.Print();
                NE3.Print();
                NE4.Print();
                NE5.Print();
                NE6.Print();

        }



        cout << "The end." << endl;
}
//end of executable program
```