Optically Interconnected Computing Systems

Keith James Symington

Submitted for the Degree of

Doctor of Philosophy

Heriot-Watt University

Department of Physics

November 2001

Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank my supervisor John Snowdon for putting up with my constant questioning. For help and support, Mohammed Taghizadeh, Andrew Waddie and Rod Webb. Many thanks are also due to the technical efforts of those in the electronics workshop, namely Keith Logan, Graham Hunter and Gary Beveridge.

Special thanks must go to my parents, Jim and Christa Symington, for their constant support in many ways right through to the end of my Ph.D.

And last, but by no means least, I would like to thank my partner, Beatrice Alex, for keeping me relatively sane. ☺

Abstract

Future computing systems will offer greater processor performance using either conventional or emergent technologies. However, few of these systems address data locality issues such as moving information from processor to memory, storage or another processor. Electrical interconnection even across a large substrate is becoming prohibitive due to intrinsic bandwidth limitations. Optoelectronics is presently the only viable solution to alleviate this problem and has already superseded electronics for long-haul transmission. This thesis examines the benefits of optical interconnection at a short range chip-to-chip level, both in free space and in a waveguide. It shows that optoelectronics enables a connectionist approach to computing allowing the construction of architectures such as a neural network. Simulation maps the assignment problem to this type of architecture and underlines both its performance and startling scalability. The algorithm is then implemented, resulting in the construction of two optoelectronic neural network demonstrators, both of which use high density free-space optical interconnection. The respective performance of these demonstrators is then examined. Finally, the mutual benefits of integrating an optical interface to dynamically reconfigurable field programmable gate arrays (FPGAs) are considered.

Publications and Presentations

This list is in reverse chronological order.

1. K. J. Symington, A. J. Waddie, M. R. Taghizadeh and J. F. Snowdon, "A Neural Network Packet Switch Controller: Scalability, Performance and Network Optimization", submitted to *IEEE Trans. on Neural Networks*, May 2001.

2. K. J. Symington, A. J. Waddie, T. Yasue, M. R. Taghizadeh and J. F. Snowdon, "An Optoelectronic Neural Network Packet Switch Scheduler", presented at Department of Physics, Heriot-Watt University, Edinburgh, UK, January 2001.

3. K. J. Symington, A. J. Waddie, T. Yasue, M. R. Taghizadeh and J. F. Snowdon, "High Performance Optoelectronic Neural Network Scheduler", OSA Technical Digest of *Optics in Computing 2001*, Lake Tahoe, NV, pp. 12-14, January 2001.

4. K. J. Symington and J. F. Snowdon, "Optoelectronics and a Neural Network Packet Switch Controller", presented at Department of Electronic and Electrical Engineering, Strathclyde University, Glasgow, UK, August 2000.

5. J. F. Snowdon, K. J. Symington and B. Layet, "Optoelectronics in the Reconfigurable Environment (An Overview of Optics in Computing)", *Dynamically Reconfigurable Architectures*, Schloss Dagstuhl Seminar Report 00261, Schloss Dagstuhl, Germany, June 2000.

6. A. J. Waddie, K. J. Symington, M. R. Taghizadeh and J. F. Snowdon, "An Optoelectronic Neural Network: Implementation and Operation", OSA Technical Digest of *Optics in Computing 2000*, Quebec City, Canada, pp. 304-310, June 2000.

7. K. J. Symington, J. F. Snowdon, A. J. Waddie, T. Yasue, and M. R. Taghizadeh, "Optoelectronic Neural Networks", IEE Proc. of *Postgraduate Research in Electronics, Photonics and Related Fields*, Nottingham, UK, pp. 182-187, April 2000.

8. K. J. Symington, J. F. Snowdon and H. Schroeder, "High Bandwidth Dynamically Reconfigurable Architectures using Optical Interconnects", presented at IEEE-LEOS Meeting, Heriot-Watt University, Edinburgh, UK, March 2000.

9. R. P. Webb, A. J. Waddie, K. J. Symington, M. R. Taghizadeh and J. F. Snowdon, "An Optoelectronic Neural Network Scheduler for Packet Switches", *Appl. Opt.*, vol. 39, no. 5, pp. 788-795, February 2000.

10. K. J. Symington, J. F. Snowdon and H. Schroeder, "High Bandwidth Dynamically Reconfigurable Architectures using Optical Interconnects", 9th Int. Workshop on *Field Programmable Logic and Applications* (FPL), Glasgow, UK, Lecture Notes in Computer Science, Springer Verlag, vol. 1673, pp. 411-416, September 1999.

11. R. P. Webb, A. J. Waddie, K. J. Symington, M. R. Taghizadeh and J. F. Snowdon, "Optoelectronic Neural Networks for Packet Switching", IoP Technical Digest of *Quantum Electronics 14*, Manchester, UK, September 1999.

12. R. P. Webb, A. J. Waddie, K. J. Symington, M. R. Taghizadeh and J. F. Snowdon, "A Neural Network Scheduler for Packet Switches", OSA Technical Digest of *Optics in Computing 1999*, Aspen, CO, USA, pp. 193-195, April 1999.

13. M. R. Taghizadeh, J. F. Snowdon, A. J. Waddie and K. J. Symington, "Optoelectronic Neural Networks for Switching", presented at IEEE-LEOS Meeting, Heriot-Watt University, Edinburgh, UK, April 1999.

1 Introduction

"Leave the beaten track occasionally and dive into the woods. Every time you do so you will be certain to find something that you have never seen before. Follow it up, explore all around it, and before you know it, you will have something worth thinking about to occupy your mind. All really big discoveries are the result of thought."

Alexander Graham Bell, Inventor of the Telephone

The communications revolution began with the introduction of the telegraph which enabled long distance communication of information at high speed. Although work on the transmission of information by wire had been in progress since the beginning of the 19th century, the most successful implementations of the telegraph were made independently in 1837 by Cooke and Wheatstone in Britian and Morse in the United States. The telegraph was eventually superseded by the telephone in 1876 when Bell invented and patented his design. Interestingly, Bell also invented an optical telephone four years later which he referred to as the Photophone. Vibrations in a mirror caused by sound were transmitted on a focussed ray of sunlight to the receiver. Unfortunately, his idea proved impractical as not only did the device require sunlight to operate but the absorbtivity of air limited its range. Long distance communications were to remain exclusively in the electrical domain for many years to come.

Optical communications finally came of age with the invention of the laser, an application of Einstein's theory that a photon could stimulate an excited atom to emit another identical photon. Such a device was proposed by Schawlow and Townes [1] in 1958 but it was four years before semiconductor based devices were demonstrated, almost simultaneously, by four different research groups [2]-[5].

Today's versions of these components have evolved considerably since their inception. This is due not only to new and more efficient fabrication technologies but also to radical new ideas. Long haul fibre optic communications systems have become prevalent with almost every long distance call travelling through fibre-optic cabling at some point on its journey. Optic technologies have begun to encroach on areas that were once firmly considered to be the domain of electronics.

Although the advantages of long distance optoelectronic systems, such as fibre optics, are well known, the drastically increased information flow that they present electronic systems with is a serious problem as this information must be routed and switched in real time. This thesis shows that short distance optoelectronic systems enable the construction of novel architectures, not feasible in electronics, which can alleviate the very problems that long distance optoelectronic interconnects create.

The scope and layout of this thesis is summarised in Figure 1. Each chapter is described in detail in the following paragraphs.

Chapter 2: Optical Interconnects and Switching is a theoretical introduction to the areas of both optoelectronic interconnects and packet switching. It looks at the limitations electronic interconnects are facing and discusses how optical interconnects can alleviate this problem. Optics also has other properties that set it aside from electronics as it allows complex interconnection patterns to be created relatively easily. This chapter also discusses packet switch architectures in detail, presenting theory on the packet based transmission systems that voice and data communications are converging on. Basic concepts and measures are examined along with packet switch architectures such as the crossbar and multistage interconnection network (MIN).

Chapter 3: Optical Technologies and Switching examines the components used to transfer any electrical signal to the optical domain, guide it to its destination and return it to the electronic domain at the receiving end. Such technologies can be integrated at different levels, a few of which are examined here. This chapter also presents an optical highway architecture suited to high bandwidth multiprocessor systems, examining how and where optoelectronics could be inserted to interconnect multiple nodes.

Chapter 4: Neural Networks for Switching highlights the properties and uses of neural networks, examines a few common neural network architectures and defines their performance measures. Neural networks are well suited to solving optimisation problems and a network is derived to perform packet switch scheduling. This network is simulated and its characteristics compared and contrasted with conventional packet switch schedulers. Scalability is examined in depth with surprising conclusions drawn on its relationship to performance.

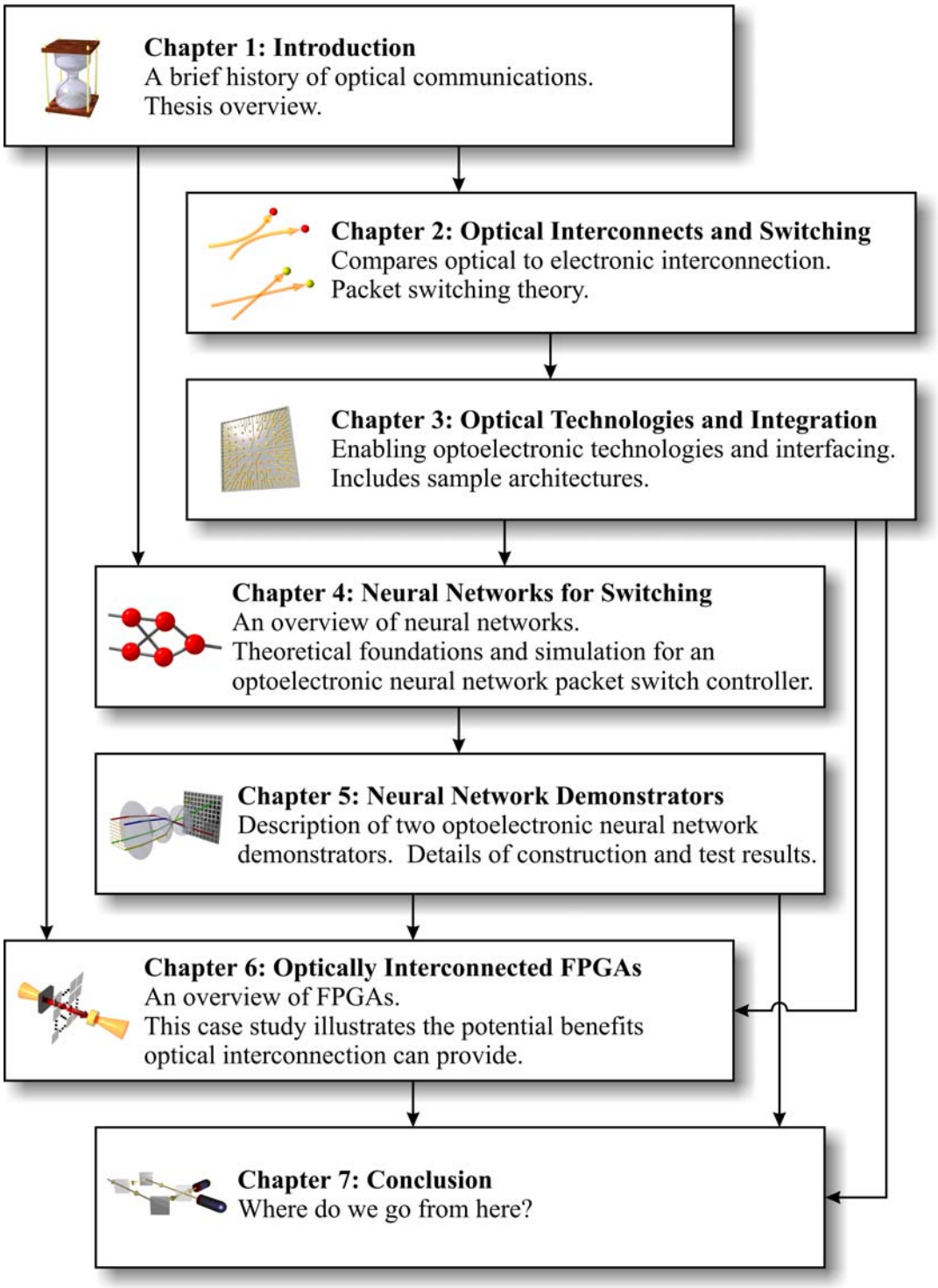

Figure 1: Thesis Overview
See text for a full description of each chapter.

Chapter 5: Neural Network Demonstrators examines two optoelectronic neural networks demonstrators, both of which can only be implemented in a scalable manner in optoelectronics, that algorithmically outstrip any other known purely electronic solution for packet scheduling. The first generation neural network demonstrator

provides proof of principle using discrete building blocks to construct a working system. The second generation provides enhanced functionality in the form of packet prioritisation and active calibration. The integration of digital signal processors (DSP) adds programmability resulting in a generalised neural network architecture. Results and analysis are presented for both of these demonstrators.

Chapter 6: Optically Interconnected FPGAs provides a case study detailing how optical interconnection provides an elegant method of circumventing a common electronic component's intrinsic limitations. The evolution of this device is examined. Initially, optoelectronic interconnection proved unnecessary. However as device size, complexity and performance increased, so did its need for the additional bandwidth gained using optoelectronic interconnects.

Chapter 7: Conclusion examines the future of short range optical interconnects and where the author sees them in the next few decades. The reasons why computer technologies on the horizon compound current interconnection problems are considered.

The author hopes to provide an up to date snapshot of state-of-the-art optoelectronic components with evidence to support chip-to-chip level optical interconnection complementing existing long distance links. Experiment follows theory, with the detailed examination of two fully operational optoelectronic neural network demonstrators exploiting short range interconnects.

2 Optical Interconnects and Networks

This chapter contains theory essential to the understanding of this thesis. It has two distinct threads which initially seem unrelated to each other. Their relationship becomes apparent in chapters 4 and 5.

2.1 Optoelectronic Interconnects

Optoelectronic devices act as an interface by converting an electrical signal into the optical domain or vice versa. The properties of the signal differ depending on domain. This section examines the motivation for converting a signal into the optical domain and the associated benefits that are brought to *very large scale integration* (VLSI) interconnection as a whole.

2.1.1 Electrons Versus Photons

The difference between electronics and optics is essentially the difference between the electron and the photon as shown in Figure 2. Since electrons carry mass and charge, they interact strongly with each other making them ideally suited to switching. Photons, on the other hand, do not carry mass or charge and are therefore non-interacting in free space. This makes them immune to magnetic interference, be it from man made sources such as electric motors or from environmental effects such as lightning. Therefore, photons are ideally suited as long distance data carriers [6] with the telecommunications industry having largely abandoned electrical interconnection in favour of fibre optics.

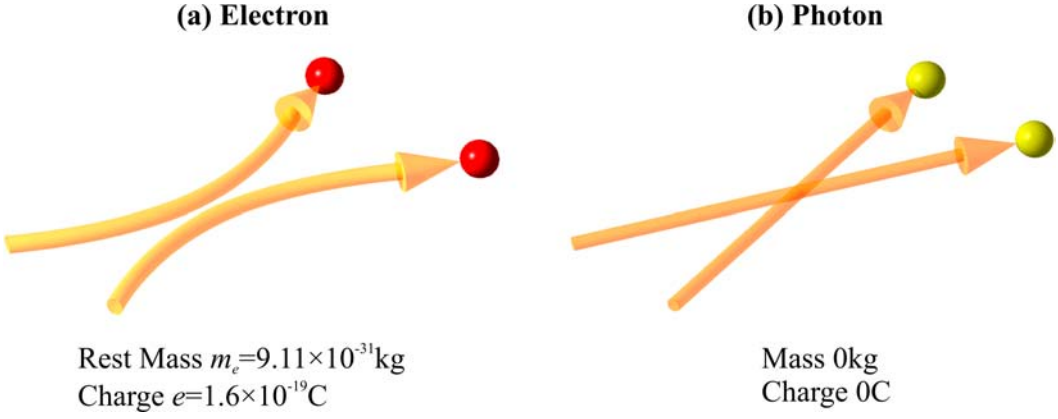

Figure 2: Properties of Electrons and Photons
Electrons interact strongly whereas photons do not interact at all.

The propagation speed of electrons compared to that of photons is dependent on the transmission medium refractive index n. Equation 1 describes the velocity v at which any signal propagates where c is the speed of light:

$$v = \frac{c}{n} \qquad \textbf{Equation 1}$$

Depending on transmission medium, the electron sees a refractive index of $n \approx 1.3\text{-}2.5$ [7]-[8]. In waveguides such as a fibre optic core, the photon sees a refractive index of $n \approx 1.4\text{-}1.6$ [9]. This indicates that the propagation speeds of electrons and photons are comparable, with electronics significantly slower in the worst case. The only major difference is when free space is used to transmit photons since $n \approx 1.0$. However absorbtivity in anything but a vacuum restricts all practical usage to short range only.

2.1.2 Transmission Lines

Optoelectronic systems attempt to make the best of both worlds by using electrons to switch data and photons to carry data. By this definition, optics attempts to replace electrical transmission lines.

An electrical signal is considered to be in the transmission line domain [10]-[11] when conductor length l is:

$$l >> \frac{vt_r}{\sqrt{\varepsilon_{eff}}} \qquad \textbf{Equation 2}$$

where v is the velocity of the signal in the transmission medium in ms^{-1}. The full scale signal rise time t_r indicates, but does not measure, the frequency of the signal. Normally, a signal will rise to its maximum and remain there for two or three times the duration of t_r. The other parameter in this equation is the relative dielectric constant of the line ε_{eff}, taken to be approximately 3.5 for a glass epoxy circuit board. Given that nanosecond rise times are common in electrical components, *printed circuit board* (PCB) tracks are considered to be transmission lines when their length is greater than a few centimetres.

Electrical transmission lines are modelled as shown in Figure 3. The inductance L_l, capacitance C_l and resistance R_l are all considered per unit length. The impedance of the cable can therefore be modelled over an infinite length using:

$$Z_l = \sqrt{\frac{R_l + j2\pi f L_l}{j2\pi f C_l}} \qquad \textbf{Equation 3}$$

where f is the frequency at which the system is modelled.

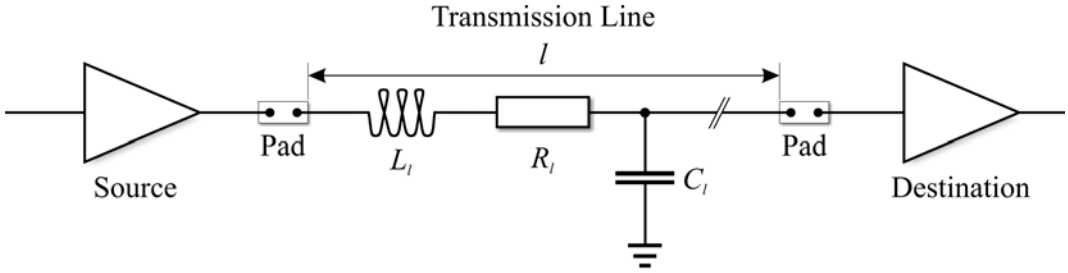

Figure 3: Electrical Transmission Lines
Signal transmitted from source to destination. The transmission line has L_l , C_l and R_l per unit length. Note that the pads also contribute to C_l.

Under most circumstances, the real resistance R_l of the conductor per unit length can be approximated as zero resulting in a characteristic impedance of:

$$Z_l = \sqrt{\frac{L_l}{C_l}} \qquad \textbf{Equation 4}$$

This information allows calculation of signal velocity in the medium using:

$$v = \frac{1}{\sqrt{L_l C_l}} \qquad \textbf{Equation 5}$$

Transmission line optimisation requires impedance matching as shown in Figure 4.

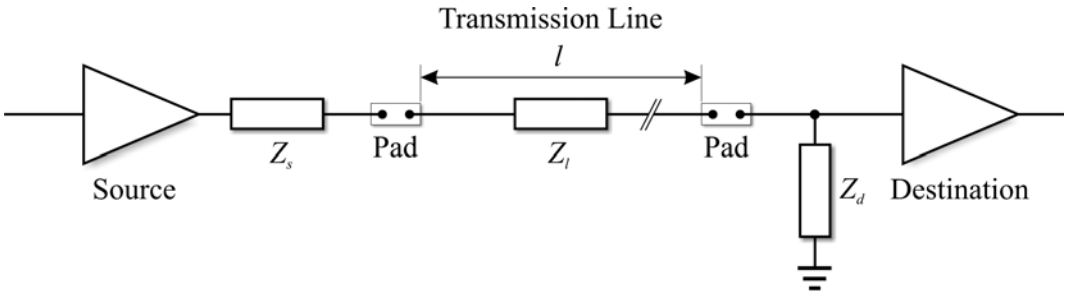

Figure 4: Impedance Matching
Signal transmitted from source to destination. Z_s is the source impedance, Z_l the transmission line impedance and Z_d destination load impedance.

To ensure that all power from the source is transferred into the transmission line, Z_s must equal Z_l. The same impedance matching criteria apply to power transfer from transmission line to destination, where Z_l must equal Z_d. If the destination is not correctly impedance matched then signal back reflection will occur, as shown in Figure 5. Here, a square wave signal with rise time t_r is applied to a transmission line. As not all of the signal's power is absorbed by the destination impedance, the remainder is

reflected back into the transmission line. This results in standing interference waves in the transmission line.

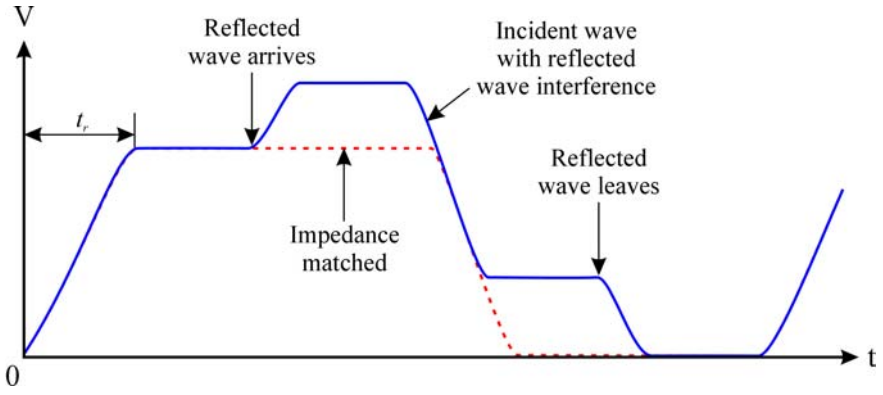

Figure 5: Reflection in Transmission Lines

No impedance matching results in reflected wave interference as shown by the solid blue line. Impedance matching transfers all power to the destination as shown by the dashed red line.

Electrical impedance matching can be achieved by using a resistive load with impedance Z_d that matches the impedance of the transmission line Z_l. Regardless of how a line is terminated, if correctly impedance matched no reflections will occur and all power will be absorbed within the terminator Z_d: giving rise to thermal concerns. At the transmitter, an appropriate drive current must be supplied resulting in further thermal concerns and considerably reducing physical packing density. To serve as an illustration, let us consider a 32-bit wide bus of $50\,\Omega$ terminated transmission lines with a 3.3V swing operating at 50% duty cycle. Equation 6 calculates the amount of power dissipated in the terminating resistors and Equation 7 the required drive current.

$$32 \times \frac{1}{2} \times \frac{V^2}{R} = 3.49\text{W} \qquad \textbf{Equation 6}$$

$$32 \times \frac{V}{R} = 2.11\text{A} \qquad \textbf{Equation 7}$$

This is quite a considerable figure to add to most power supply requirements. The problem is currently ignored by bus designers who limit their architectures to minimise reflections. An example of this is the PCI bus architecture [12] found in most personal computers where, if the 33MHz standard is used, up to 8 adjacent devices are allowed or, if the 66MHz standard is used, only 4 to 5 devices.

For both free-space and wave guided optics, the impedance matching criteria are very different since there is no electrical coupling between either end. This limits noise

transfer and ground contention issues as an optical link approximates a common component known as an optical isolator [13]. An optical transmission line is considered to be impedance matched when the full signal is absorbed without reflection at the receiving end. This can be done by fabricating a section of the transmission medium to be exactly one quarter of a wavelength thick or an integer multiple plus one quarter [14]. This results in destructive interference of any signal that would have been reflected back. Such optical impedance matching devices are called *anti-reflective* (AR) coatings. Figure 6 shows a sample optical transmission line.

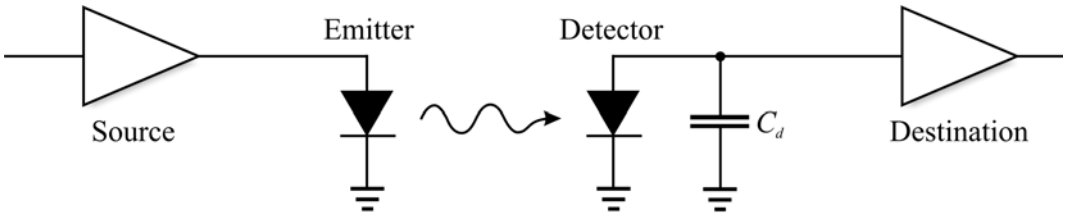

Figure 6: Optical Transmission Lines
Photons carry information from the source laser to destination detector. Detectors have an inherent capacitance C_d.

The major advantage is that an optical receiver operates in a non-dissipative manner [15]. Any photons incident on a photodetector are converted into charge based on the device's quantum efficiency η, with $\eta \approx 1$ feasible, resulting in little excess heat. Typical photodetectors generate 0.4 to 0.5 AW^{-1} of incident light but have an intrinsic capacitance C_d which limits speed of operation. The problem with heat dissipation in optical interconnection does not lie in reception but rather in emission. One of the most promising technologies, the vertical cavity surface emitting laser (VCSEL), still does not have the necessary power conversion efficiencies to eliminate excess heat: $\eta \approx 0.1$ (10%) in commercially available devices (can be $\eta > 0.5$ in lab situations). According to the MEL-ARI Optoelectronics Roadmap [17], a standard 64 element VCSEL array transmitter operating at 50% duty cycle with a 4mA operating current at 2.4V would thus produce excess heat of:

$$64 \times \frac{1}{2} \times V \times I \times (1 - \eta) = 0.28\mathrm{W} \qquad \textbf{Equation 8}$$

Each VCSEL in this illustration has $\eta = 0.1$ and produces 1mW of optical output power, orders of magnitude greater than the minimum power required for detection. This is still considerably less dissipated power than standard transmission lines generate, even when using a low efficiency VCSEL such as this. The difference is that excess thermal energy must be dealt with at the transmission end.

Optical transmission lines, whether guided or free space, are attenuated far less than electrical lines. Scaling an optical interconnect initially designed for a PCB from a few tens of centimetres to kilometres is feasible without significant, if indeed any, increase in driving power. Nor is bandwidth affected, since multiple pulses can be sustained en-route to the same destination. The only noticeable and unavoidable effect is increased delay between transmission and reception of information.

As data rates increase the breakeven length for optical communications decreases. If data rates exceed 1Gbs^{-1}, optical interconnects require less energy to transfer a single bit of information than electronics if transmission line lengths are more than a few centimetres long [16]. Nevertheless, there will always be a point at which conversion into the optical domain is unnecessary or even wasteful both in terms of transmission rates, additional expense incurred and power consumed. This breakeven point will always be application dependent. However, based on current technological trends, it is continually getting shorter.

2.1.3 On-Chip Interconnect

Traditionally, any interconnect not in the transmission domain as defined by Equation 2, has normally been considered as *RC* limited. However, technology has proven relentless in decreasing feature size, increasing die size and chip frequency. Commercial processor transistor feature sizes are currently 0.18μm and according to Intel [18] this progress will not hit any physical limit within the next 5 years. This has certainly increased the significance of *RC* charging, whose value increases exponentially as feature size diminishes, but it has also increased the problems associated with on-chip long lines. Inductance is beginning to become significant again in long lines [19] with effects such as overshoot, delay increase and inductive crosstalk.

Optics is a potential solution to this problem as it would allow distribution of signals such as the chip clock frequency in a non-electrically noisy manner. The author believes that on-chip optical long lines would be commonplace if it were not for the fact that no technology currently exists which can fabricate an optical emitter on silicon (Si).

2.1.4 Interconnection Bandwidth Limitations

As the length and/or density of electrical interconnection increases they begin to suffer from increased wire resistance, residual wire capacitance from fringing fields and inter-wire crosstalk.

The maximum bandwidth limit of any electrical system, such as that in Figure 7 and regardless of interconnection scheme, can be derived from physical principles and is described by:

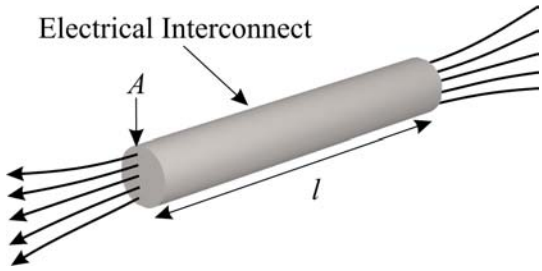

Figure 7: Electrical Interconnect
Cross sectional area *A* and length *l* are clearly indicated.

$$B_{\max} \approx B_0 \frac{A}{l^2} \qquad \textbf{Equation 9}$$

Here B_{max} is the maximum bandwidth, B_0 is a constant of proportionality, A is the interconnect's cross-sectional area and l is the interconnect length. B_0 has been independently estimated [20] to be approximately 5×10^{14} for electrical systems where A/l^2 is the aspect ratio. Thus the maximum bandwidth for a 0.1m off-chip electrical connection is around 150GHz.

When optics are used this limit simply does not apply. In free space, optics propagate by definition without a guiding medium and with an attenuation significantly smaller than that that seen in electronics. The parallelism available is estimated to be greater than 100,000 channels per cm^2, assuming energy density per unit volume does not exceed that required for ionisation of any gas present. Admittedly, the connections from data source to VCSEL or detector to data destination still need to be electrical: but when this distance is 10 to 100 microns in a flip-chip bonded system, as opposed to a few centimetres in an electrical system, the limitation is in practice irrelevant. The number of optical pins that can be driven depends primarily on thermal and real estate considerations. In 2001, 4,096 channels can be driven from $1cm^2$ with no real obstacle to reaching more than 10,000 channels. CMOS limited rates of $200Mbs^{-1}$ have been demonstrated giving a bandwidth of approximately $20Tbs^{-1}$. However, individual devices may routinely be driven at $10Gbs^{-1}$ so a system could essentially handle $1{,}000Tbs^{-1}$. The theoretical limit is actually much higher than this.

Optical channels can be driven at speeds much greater than electrical channels and take considerably less power to drive than pads and wire bonds. What should be

remembered though is that optical interconnection of a chip is an enhancement and does not preclude conventional electrical connection as well.

2.1.5 Optical Architectures

The use of optical interconnects enables the construction of novel system architectures not feasible in electronics. This section examines the optical properties that can be exploited rather than considering types of system architecture.

Optical information channels can be manipulated in many ways. Take for example Figure 8:

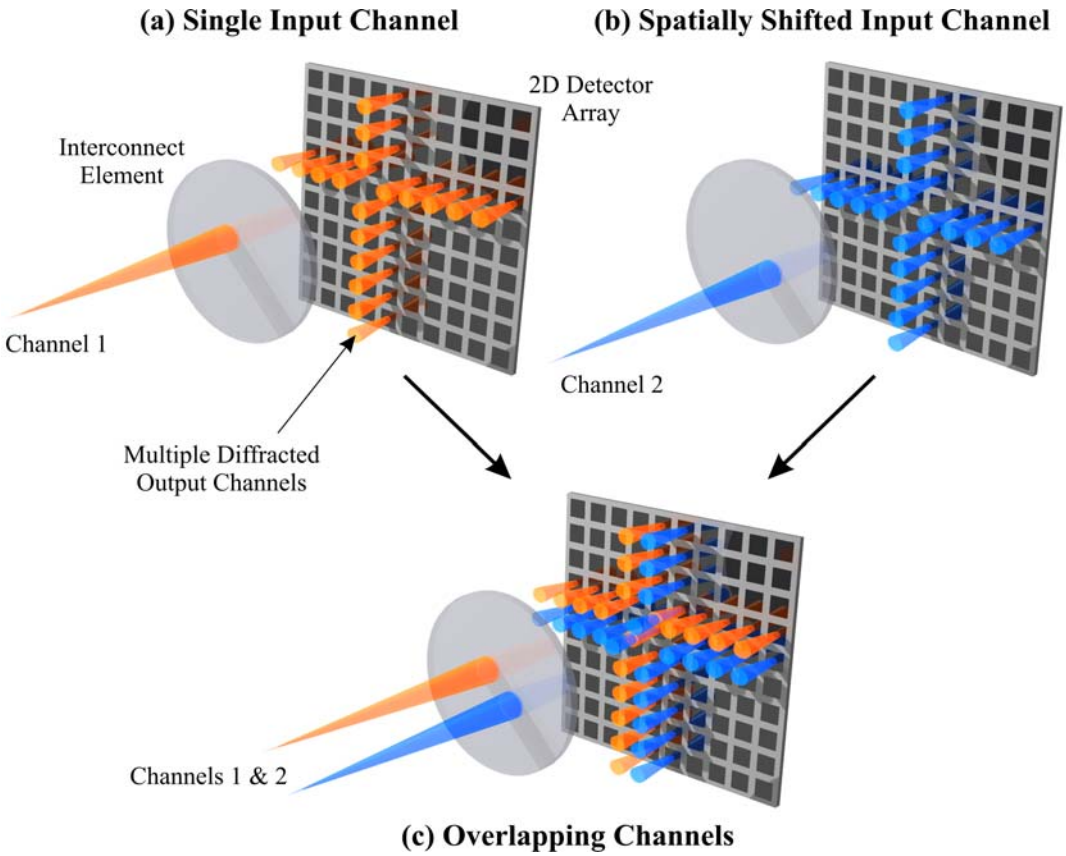

Figure 8: Manipulating Optical Channels
This interconnect architecture is used by both neural network demonstrators that are outlined in forthcoming chapters. It illustrates well the various beneficial properties of optics. Different colours are used to represent different channels and are not a reference to wavelength.

A single optical channel can be fanned-out to a few, or indeed many, destinations. In Figure 8(a) there are multiple destination detectors on a two dimensional detector array. Each destination detector receives an identical and diminished copy of any information on channel 1. Fan-out is limited by minimum detectable optical power. As

interconnect elements usually have no amplification, all incident power from channel 1 must be split up amongst the destination detectors. As fan-out increases, the power incident on each detector decreases - presuming channel power remains constant.

Figure 8(b) shows another channel, which we will refer to as channel 2, incident on a slightly different point of the interconnect element. Note the use of different colours for channels 1 and 2 is simply to distinguish them and does not represent a different optical frequency. Channel 2 produces an interconnect pattern identical to that of channel 1 but shifted in space by a distance equal to the input channel shift. Although the interconnect pattern is the same, an entirely different set of detectors are now receiving the signal.

Simultaneously applying both channels 1 and 2 gives Figure 8(c). As light has the property that it is non-interacting in free space, the interconnects can cross each other without any interference. Fan-in is also present in this configuration since there are two detectors that receive signal from both channels 1 and 2. This is undesirable in a digital information transmission system, but can be used in an analogue one to perform signal summation of both channels.

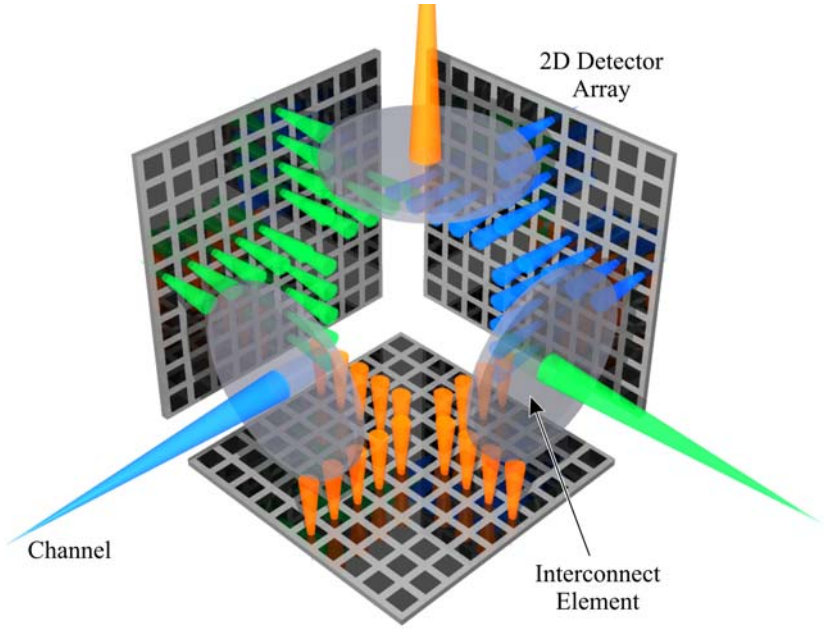

Figure 9: Overlapping in Multiple Dimensions
Optics allows use of the same transmission medium in multiple dimensions. The different colours represent different channels and are not a reference to wavelength.

Overlapping of channels can also be performed in multiple dimensions. Optics has the advantage that as long as nothing is physically blocking a path it can be reused, as

shown in Figure 9. This allows data densities to reach phenomenal values with interconnection complexity decades ahead of electronics.

The nature of optical interconnects also results in scaleable complex architectures. Such an example is a sorting algorithm for parallel computing developed by Stone in 1971 which has, to this day, not been surpassed as far as a minimal rate of growth of computational steps is concerned. The algorithm is based on work done by Batcher in 1968 called the bitonic merge-sort, Stone adapting Batcher's work for a shuffle exchange network [21]-[22]. This interconnection methodology, as shown in Figure 10, is generally known as Stone's perfect shuffle.

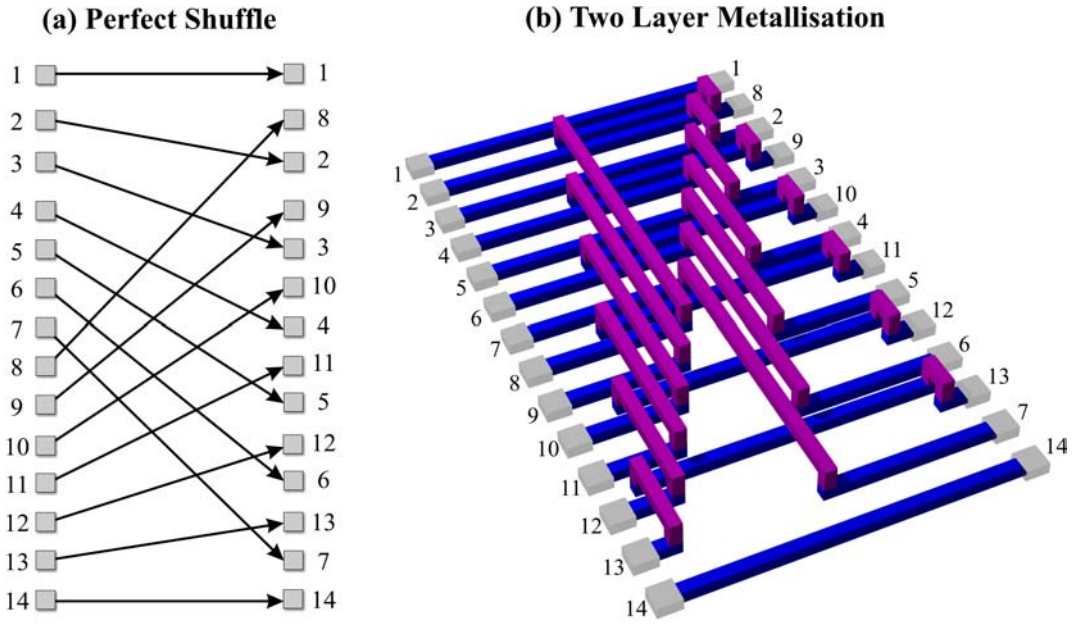

Figure 10: Stone's Perfect Shuffle
Part (a) shows the desired interconnect and (b) an implementation fabricated using a two layer metallisation process.

Any electronic implementation of this architecture at a large scale is prohibitive as the layout area required for interconnect scales quadratically with inputs and outputs. This is typical of the type of problem that limits any scaleable implementation of concurrent electronic systems and precisely where optical interconnection becomes beneficial.

The non-interacting nature of free space optical channels means that they can pass through each other to form any desired interconnection topology without cross-talk. Complex interconnects thus become relatively simple to implement with minimal skew as large wire length variation can be avoided. Figure 11 shows an optical implementation of Stone's perfect shuffle.

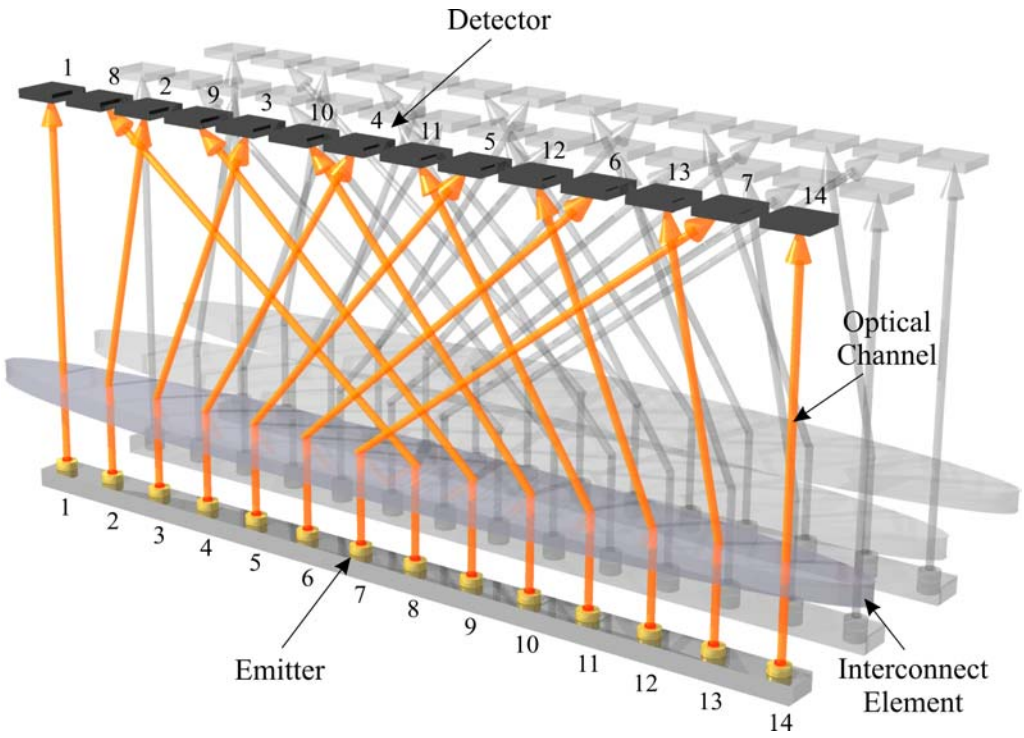

Figure 11: Optical Implementation of Stone's Perfect Shuffle
Shadow images indicate 2D extension of this architecture.

This example only uses one dimension for optical interconnection. The second dimension could be used to add either additional perfect shuffle elements, as shown, or to reduce maximum beam deflection by optimising interconnection towards a square array.

The architectures examined so far have extolled the advantages of what are referred to as free space systems. Such systems are usually short range architectures providing complex interconnection. Longer range systems require a guiding medium through which to travel, generally referred to as a waveguide. Commonly a fibre is used, but polymers and/or silicates are encroaching, providing the optical equivalent of printed circuit board tracks [23]. Such guiding almost always results in the traversal of corners with total internal reflection ensuring that photons do not leave the waveguide. Unfortunately, this does not preserve spatial coherence so only the total amount of incident optical power reaches the destination with no remaining relevance to initial spatial position.

Wavelength division multiplexing (WDM) uses different wavelengths of light to carry multiple channels on a single waveguide as shown in Figure 12. At the destination, these wavelengths are extracted and guided to the correct destination detector. Today's

WDM systems use individual channel data rates of up to $10Gbs^{-1}$ (OC–192) with just over 100 channels per waveguide at the high end. However, systems such as Essex Corp's Hyperfine WDM [24] are currently reaching production prototype stage and show the potential to carry 4,000 channels on a single waveguide with a spacing of 1GHz between carrier frequencies in the optical spectrum. This would result in data rates of $40Tbs^{-1}$ but this is still nowhere near the theoretical optical transmission limit.

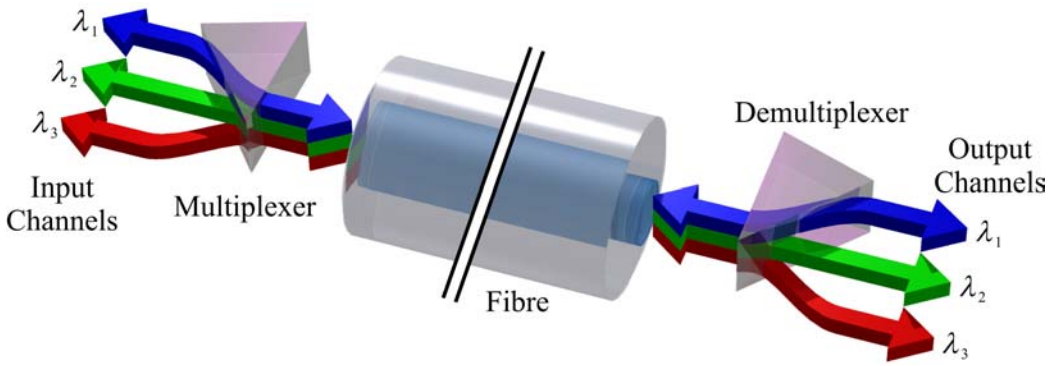

Figure 12: Wavelength Division Multiplexing
Three different wavelengths of light multiplexed down a single waveguide. Fibre not shown to scale.

Another technique to multiplex optical signals on a single waveguide is to polarise the light. Unfortunately, it is not as effective as WDM since reliable differentiation is currently only practical between two polarisations of light. A common component known as a *polarising beam splitter* (PBS) can be used to extract the signal as it only reflects light with a specific polarisation. However, polarisation state is not particularly stable in a waveguide and unless a specifically designed fibre with an elliptical cross section is used then polarisation will only be maintained for a few metres. This technique is problematic as light can also have multiple polarisation components simultaneously, for example circular or elliptical polarisation, giving it limited commercial interest.

2.1.6 Bandwidth Predictions

In 1965 Gordon Moore [25] observed that the number of transistors on an integrated circuit approximately doubled every eighteen months. His observation has held true to this day and is now commonly known as Moore's law. Prediction of future trends in component technologies based on recent research and development helps engineers plan for future designs. One respected group which makes such predictions for silicon chips

is the Semiconductor Industry Association (SIA) and every year it publishes a roadmap [26], an extract of which can be seen in Table 1.

Year	1999	2002	2005	2008	2011	2014
Process Size (nm)	180	130	100	70	50	35
Chip Size (mm^2)	450	509	622	713	817	937
On-Chip Clock (GHz)	1.2	1.6	2.0	2.5	3.0	3.6
I/O Bus Speed (MHz)†	480	885	1,035	1,285	1,540	1,800
I/O Pads††	368	464	584	736	927	1,167
Off-Chip Data Rate (Gbs^{-1})	**177**	**410**	**604**	**946**	**1,428**	**2,100**

Table 1: SIA Roadmap
†Chip to board speed for high performance off-chip buses. ††Number of chip package pads.

Examination shows that chip process size is continuing to decrease as overall chip area and clock rates increase. This leads to an overall growth in transistor density of $\times 1.3$ [18] per year, a trend which is not reflected by off-chip data rates and has not been for some time. Clever caching techniques and the addition of metal interconnect layers have reduced the impact of this problem, but we are reaching a point where this inconsistency cannot be circumvented any longer. Even transmitting a signal across a chip these days can be a daunting task.

Year	1997	2002	2007
VCSEL†+driver pitch (μm)	125	80	60
Chip Size (mm^2)	6.25	9	16
Optical Channels/Chip	256	1,024	4,096
VCSEL† Data Rate (Gbs^{-1})	0.6	1	2
I/O Bus Width	256	1,024	4,096
Off-Chip Data Rate (Gbs^{-1})	**154**	**1,024**	**8,192**

Table 2: MEL-ARI Opto Roadmap
†Vertical cavity surface emitting laser. An optical emitter.

A similar institution also exists for optoelectronics called MEL-ARI [17]. Table 2 is an extract from their roadmap showing the predicted growth of optical interconnect technologies.

Even when both of these roadmaps were published, optoelectronic off-chip bandwidth had exceeded electronic. As time progresses, the slow expected rise of electronic bandwidth is significantly outpaced by optoelectronics to the point where in 2007 optoelectronics has off-chip data rates one order of magnitude greater than electronics. Looking further into the future of electronics, in fact as far as the SIA go, we can see that electronics will not even have one quarter of the data rates in 2014 that optoelectronics is predicted to have in 2007.

2.1.7 Optical Alignment

The drawback with any free space optical system is alignment: all components must have a particular tolerance which is directly related to cost. These tolerances need to compensate for a variety of effects such as thermal expansion/contraction, optical aberrations, long term creep and environmental conditions such as mechanical vibration from cooling fans. One way round these problems is to use *adaptive optics* (AO) [27]-[28] which perform measurement and correction of focusing and positional error in real time. The commercial viability of such techniques is easily seen by looking at a CD player [29], generally regarded as a disposable piece of machinery, which maintains focus and position of a 1μm light spot in real time on a rapidly rotating optical disk.

2.1.8 Conclusion

Free space optical interconnects appear to offer tremendous advantages over electronics: they are attenuated far less than electrical signals, offer transmission lengths of the order of metres without significant driving powers, simplify complex interconnection structures and present levels of raw data throughput simply unattainable in electronics. Their very nature implies high parallelism around any implemented machine, such as to and from memory and/or peripherals. However, there are still many engineering issues to be confronted before such interconnects can be routinely deployed.

Perhaps optoelectronics will not reach the levels stated by MEL-ARI as quickly as predicted. Perhaps a new technology will emerge that supersedes optoelectronics. But with electronics rapidly approaching its physical boundaries, there is no option other than to find another method of enhancement. Optoelectronics is the next logical step

and provides orders of magnitude performance increases with ease of integration into existing technologies. Whether it likes it or not, electronics is going to be dragged, kicking and screaming, into the optoelectronic era.

2.2 Networks and Switching Theory

Technological advances are resulting in the convergence of the three previously distinct worlds of telephone networks, data networks and multiprocessor interconnection architectures. This section examines the evolution of today's networks from their initial emergence as the public telephone system and the complexity now involved in packet switched networks. It concludes with a brief look at architectures suited to packet switching, defining metrics and considering implementation issues.

2.2.1 Public Switched Telephone Network (PSTN)

The roots of networks and switching theory lie with the invention of the telephone. A single telephone is of no use unless there is a second telephone with which to connect. Construct more telephones and issues such as how do you connect one *subscriber's* telephone to another begin to arise. Figure 13 examines this switching problem.

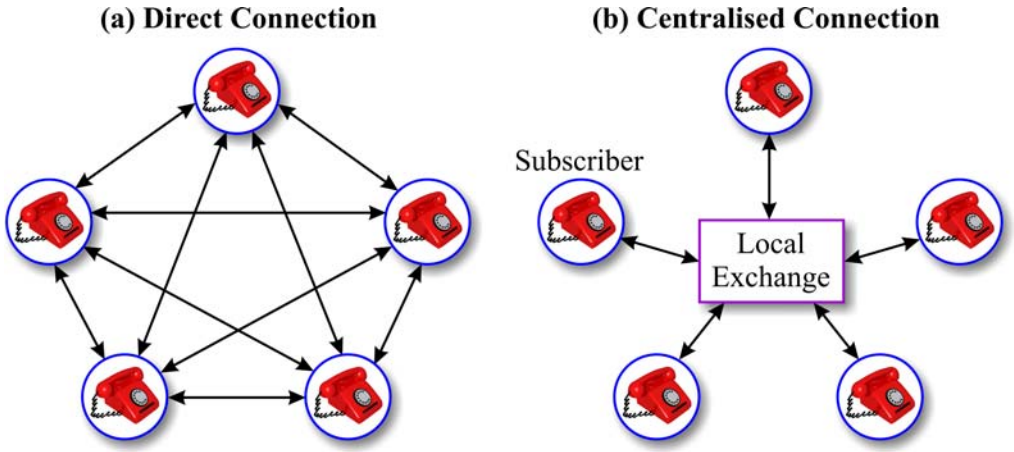

Figure 13: Local Network
Direct connection of subscribers requires significantly more lines than centralised connection, especially at higher orders.

Connecting every subscriber directly to every other in a network consisting of n_t telephones would require l_{dc} lines, as defined by Equation 10:

$$l_{dc} = \frac{n_t(n_t - 1)}{2} \qquad \textbf{Equation 10}$$

Therefore, the number of lines required increases quadratically with the number of subscribers. As network size continues to grow, direct connection will result in lines that are wasteful and may never even be used. This problem can be overcome by connecting every telephone within the local geographic area to a central point called a local exchange, as shown in Figure 13(b), thus greatly reducing the number of lines required to $l_{cc}=n_t$. However, such an architecture requires a switching mechanism at the local exchange.

For the same reason that subscribers are not fully interconnected, neither are local exchanges. Indeed they are part of a complex hierarchical structure [30] as shown in Figure 14. There are three standards used to name each level of this hierarchy: one for the USA, one for the UK and one international standard ITU-T [31] (formerly CCITT). The latter will be used here.

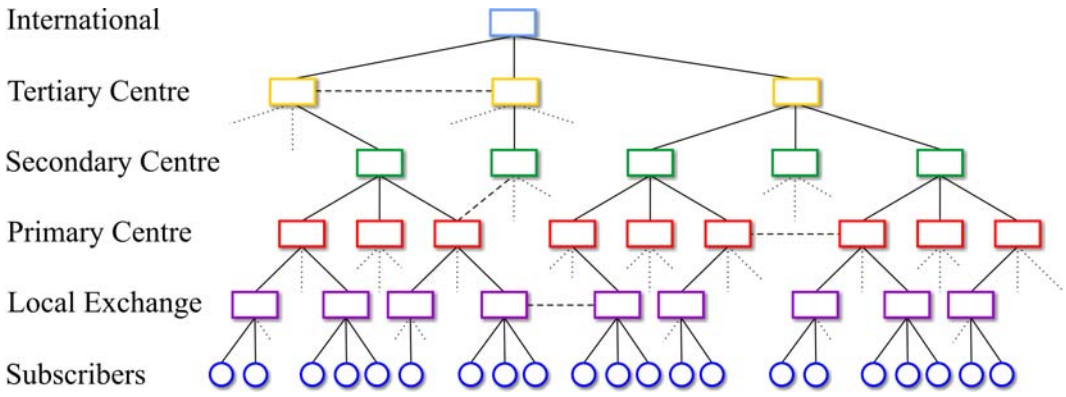

Figure 14: Hierarchical Routing System
Dashed lines indicate dedicated connections due to high demand between two particular nodes. Dotted lines indicate connections with no destination shown.

The whole point of the hierarchical architecture is to concentrate traffic from lower levels at the upper levels by multiplexing information onto higher data rate channels. This reduces overall transmission cost but results in expensive switching systems at the upper levels. The number of interconnections, exchanges and centres is also an important consideration and a network provider will install sufficient capacity to handle what they deem to be maximum load. Spare capacity does not generate revenue and is therefore a needless expense. The efficient use of all available network bandwidth is consequently a top priority.

The bandwidth usage of a channel is normally defined mathematically in the unit *Erlang* (E) [32], named after the Danish pioneer of teletraffic theory. If the channel is

continuously used for one hour it is said to have carried one Erlang of traffic, 75% usage would be 0.75E. Note that the USA uses a different unit of measurement called hundred call seconds per hour (CCS) where 1E=36CCS. To allow a network operator to determine appropriate bandwidth provision and ensure user satisfaction given economic constraints, a *quality of service* (QoS) [33] metric is used to describe the network. QoS defines the probability that any attempt to make a call by a subscriber during the busy period will be rejected by the network due to insufficient bandwidth. For example, a QoS of 0.02 means that 1 in every 50 calls placed at peak times would fail.

2.2.2 Circuit and Packet Switched Networks

Making a call using the public switched telephone network consists of three stages. First, a route is found from one subscriber to the other and bandwidth is allocated through the network. This process is called *call setup* and can take up to 10 seconds if mechanical exchanges and switching centres have to be traversed. Second, the call takes place and conversational traffic is transmitted. Third, the call is *cleared down* and bandwidth returned to the system. This type of network behaviour is called *circuit switching* as the circuit remains dedicated to a call for its entire duration, even when there are periods of inactivity between both users.

Although circuit switching is suited to telephone networks, data networks have a different set of requirements. As information transfer need not be continuous, a physical circuit does not have to be established between source and destination. This has resulted in the development of packet switched networks which were initially used in *local area networks* (LANs). Figure 15 compares circuit switching to packet switching.

Circuit switched networks transmit information within periodic time frames, a fraction of which is dedicated to a particular circuit allowing the link to handle up to a fixed number of circuits simultaneously. Spare capacity in any frame can only be used by a new circuit, and not by one that is already present, thus creating *idle slots* that have available but untapped bandwidth. Another disadvantage of circuit switching is that circuit inactivity, or *unused slots*, result in the allocation of bandwidth to a circuit that is not currently carrying information. Both of these situations waste bandwidth but are unavoidable due to the nature of the architecture.

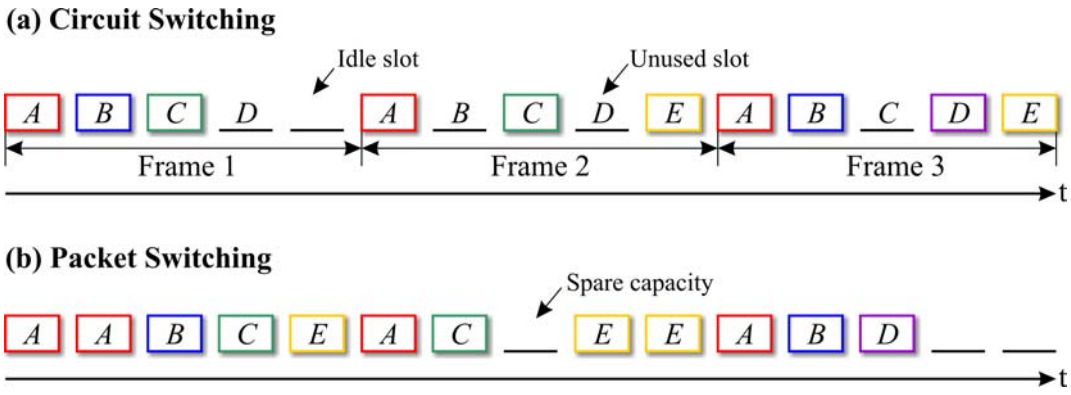

Figure 15: Circuit and Packet Switching

A letter indicates that a channel (*A* to *E*) has reserved a particular timeslot. A box indicates that data is being transmitted and a line that no information is present.

Packet switching differs from circuit switching in that bandwidth is allocated on demand. Information that travels on a packet switched network is broken up into chunks and transmitted whenever there is free bandwidth. Therefore, rather than decline a connection as a circuit switched solution would do, a packet switched network stores the packet and transmits it when a slot becomes available. This results in greater link efficiency at peak times, as shown in Figure 16, but unfortunately also in unpredictable packet delays. Although not a problem in computer networks, a user would not put up with a telephone call that keeps breaking up due to packets being delayed.

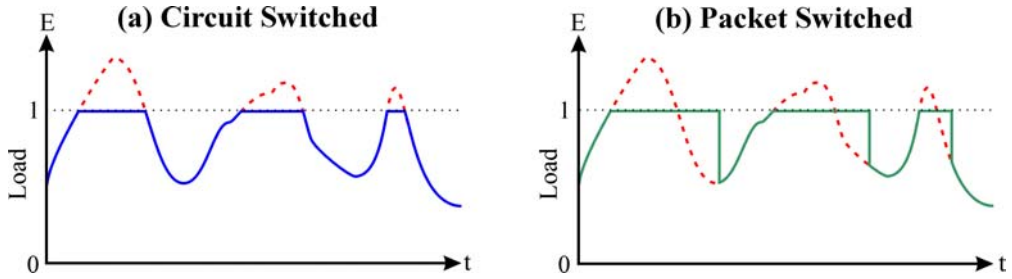

Figure 16: Link Traffic Carried

The red dashed line indicates traffic offered to the link. The circuit switched system simply routes as much as possible ignoring additional traffic. The packet switched system buffers excess packets and transmits them on a first in first out basis when link load goes below 1 Erlang.

As a packet switched network does not create a circuit from point to point and accepts packets immediately, without guaranteeing a path to the recipient, it is not possible to quantify quality of service and traffic flow in the same way as for circuit switched networks. The analysis of packet switched networks is normally statistical and quantified by the mean packet delay across the network. However, at peak times this

delay could theoretically be excessive. In practice, the finite capacity of network buffers prevents any more packets from being accepted thus capping theoretical delay limits.

A sample packet switched architecture is shown in Figure 17. It consists of a number of interconnected *packet switches* (PS) that can be connected directly to a computer system or LAN.

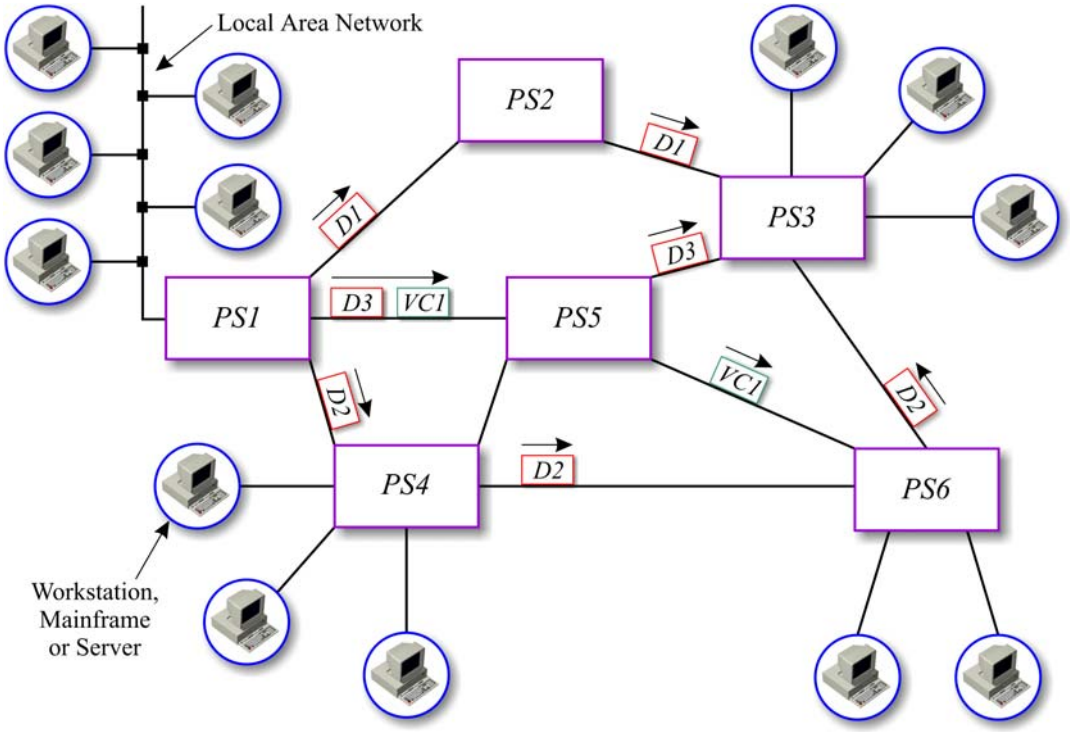

Figure 17: Packet Switched Network Architecture
Packet switched networks can have local area networks directly attached as they also operate by transmitting packets. A bridge or gateway may be required depending on LAN architecture.

The obvious difference between this type of network and the public switched telephone network is that there is no predefined architecture or hierarchy. Links are usually installed between packet switches based on estimated demand for the link. Tolerance to failed or overloaded packet switches is high as the system simply transmits on another link. In a hierarchical system, a failed node or link used to capacity could isolate large segments of the network.

When a packet switch receives a packet it stores it and then inspects the packet's header. This header contains sequencing information and may also detail the packet's destination. Each packet switch has a routing table that allows it to decode the packet's

header and choose the correct link(s) to retransmit it on. Figure 17 shows a piece of data that has been broken into three segments (*D1*, *D2* and *D3*) and transmitted onto the packet switched network. Due to high link traffic, the packets are transmitted via three different routes from *PS1* to *PS3*. Assuming that the delays incurred through each link and packet switch are equal, the packets will arrive out of sequence. Sequencing information contained within the packet header may allow reassembly of the information contained in the right order.

Packet switched networks are also able to support voice and video using a method called *virtual circuit switching* (VCS) [34]. In the same way that the public telephone network sets up a call, a virtual circuit is created between source and destination nodes through a specific set of nodes that does not change as long as the circuit remains active. Depending on implementation, the packet switched network will reserve sufficient bandwidth for the virtual circuit. This circuit is assigned a special *virtual channel identifier* (VCI) in the routing table of each packet switch traversed. After that only the VCI needs to be specified for a packet and information will automatically be transmitted to the correct destination, the virtual circuit ensuring they arrive in the order they were transmitted. Figure 17 shows a virtual circuit *VC1* set up between nodes *PS1* and *PS6*. Once transmission is complete, the virtual circuit is cleared down and resources are returned to the system.

2.2.3 Packet Switch Architectures

The task of a packet switch, to switch an input line to a particular output line, belies its complexity. Modern protocols require data to be transmitted seamlessly alongside real time information such as voice or video whose transmission has been prioritised using virtual circuits. Design decisions therefore prove critical and, if carelessly made, impact not only on the switches' performance but on that of the entire network. Figure 18 shows the important functional blocks in a packet switch.

Packets are received from transmission lines, not necessarily of the same data rate, and buffered temporarily at the packet switch input. Their destination addresses are extracted and passed to the control and routing unit which evaluates all information at its disposal and selects the packets to be transmitted. The control and routing unit then reconfigures the switch fabric if necessary and allows packet transmission to proceed. The output interface receives the routed packets, buffers them and retransmits them when possible. Again, output transmission lines do not need to be of the same data rate.

It is essential to minimise the delay incurred at every stage as a packet may travel through tens of switches before reaching its destination. If delay is not minimised, traffic such as voice could be subject to disconcerting delays as are evident in long distance communications.

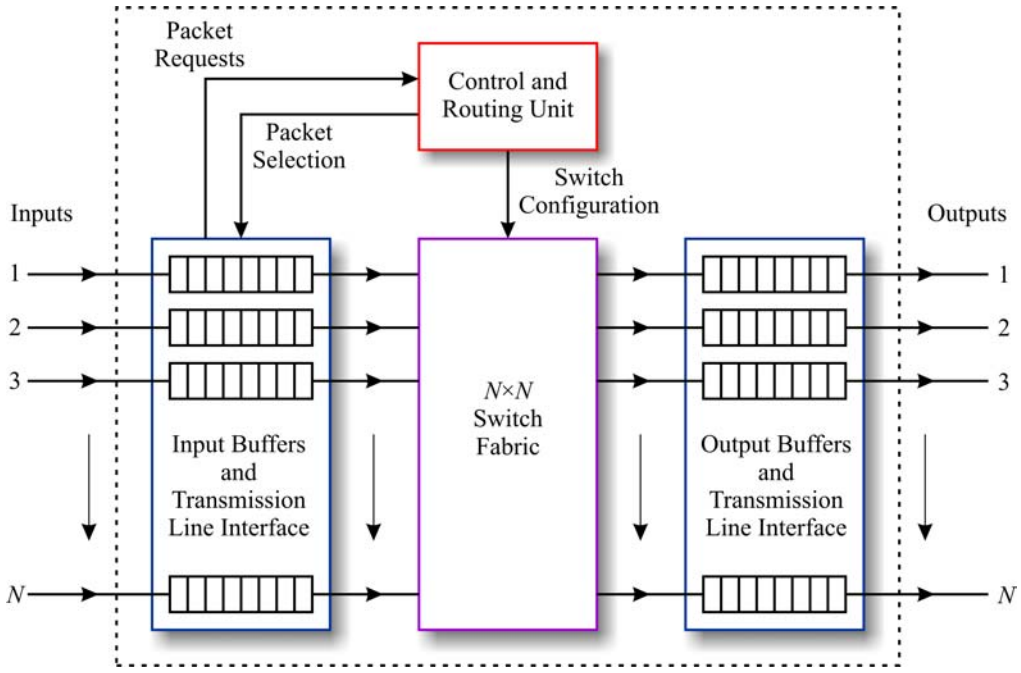

Figure 18: Packet Switch Architecture Overview
Packet switch functional blocks.

Packet switch fabrics [35] can and have been implemented in countless ways. The simplest of switching fabrics is called the crossbar switch and can be seen in Figure 19. The crossbar can achieve a connection from any input to any output by simply closing a single crosspoint switch. This results in high aggregate bandwidths as the datapath only needs to traverse a single switching stage. This type of switch is defined as *internally non-blocking* since it is always possible to connect a free input line to a free output line. However, the ability to choose any datapath also results in the requirement for complex control and routing algorithms.

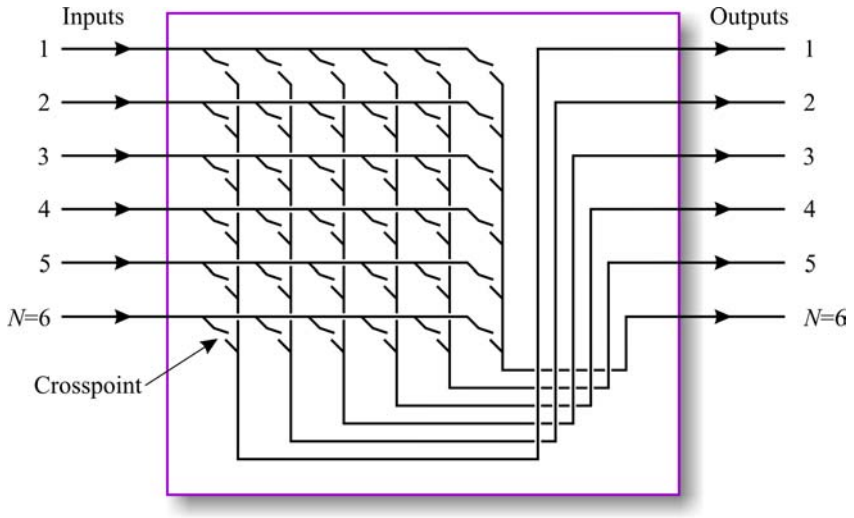

Figure 19: Crossbar Switch
Closing a crosspoint will connect an input line to the appropriate output line.

Unfortunately, the crossbar switch architecture has an N^2 growth function making it uneconomical at larger switch sizes. This problem was addressed by the arrangement of a structured network of smaller switches [36]-[37] in an attempt to find algorithms that could perform the same task using fewer crosspoints. This work resulted in the multistage interconnection network (MIN) of which the Banyan network [38] shown in Figure 20 is an example. However, MIN architectures incur a delay for each layer that a packet must traverse.

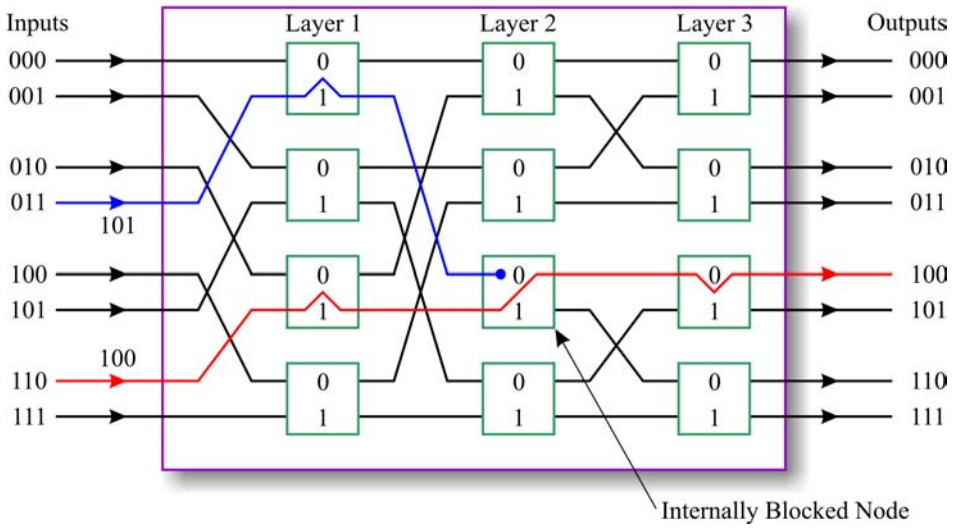

Figure 20: Banyan Network - A Multistage Interconnection Network
Each node is a 2×2 crossbar switch. A sample path routing a packet from 110 to 100 is shown. An internally blocked path from 011 to 101 is also shown. Numbers are in binary.

There are many kinds of Banyan network that can be classified into various sub-groups [39]. Unlike the example, they do not need to use the same size switch throughout each layer nor only be connected to adjacent layers. These types of networks are classified as self-routing networks since the packet's header contains all the information required to route it through the switch fabric. Figure 20 traces two paths through the switch. The first packet is input on channel 101 (binary) and is destined for output channel 100 (binary). The binary values are important as they are used to route a packet to its destination. The first node the packet encounters in layer 1 examines the *most significant bit* (MSB) of the destination address and routes it to the correct port, in this case down to 1. The node in the next layer examines the next most significant bit continuing on through layers and bits until the *least significant bit* (LSB) routes the packet out on the correct channel. Unfortunately, this network suffers from a problem that classifies it as *internally blocking*. Consider a second packet on input 011 and requesting to be output to 101. Transmission of the packet is blocked internally as it requires the use of a connection already in use by the previous packet even through the destination is different. Techniques exist for reducing and avoiding internal blocking using randomisation and sorting algorithms [40]. Unfortunately, these techniques require additional nodes which will nullify any reduction in switch elements that a MIN network may bring.

Comparison of Banyan and crossbar switches shows that for 8 inputs and outputs the crossbar requires 64 crosspoints whereas the Banyan requires 48. This illustrates that MIN networks can perform the same task as a crossbar but use fewer crosspoints. Unfortunately, improved scalability and self routing is offset by the probability of internal blocking and increased interconnection complexity [41].

The Banyan shown is an example of a *delta network* [42]. Delta networks encompass many of the better known multistage interconnection networks. A delta-b network is a delta network comprised of $b \times b$ switching elements. The number of input and output channels is $N=b^k$ where k is the number of layers the network has each with N/b nodes. The network in Figure 20 is a delta-2 network where $k=3$ and $N=8$. The sizes of these networks scale at a rate of $N\log_b N$ [42]-[43] making them suited to higher order applications than the crossbar switch. Figure 21 shows some common delta network architectures.

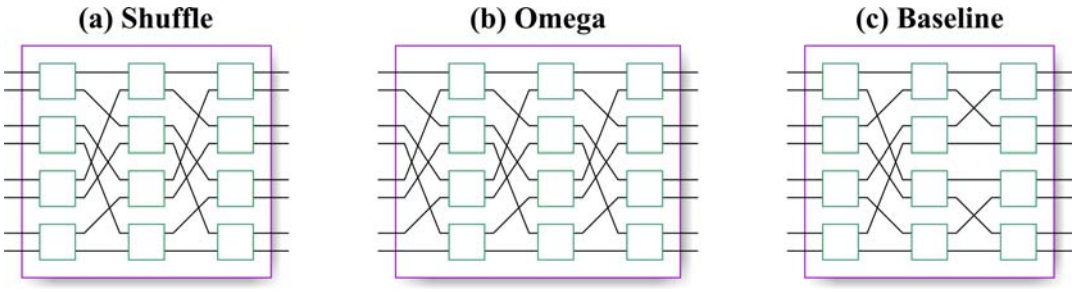

Figure 21: Delta Networks
Sample delta-2 network architectures.

Buffering strategies [44] exist to improve the QoS provided by a switch. If no buffering existed and two packets were contending for a single output, then one of the packets would be lost. Buffering normally exists at either switch input or output. It is either a dedicated buffer for each line or a shared buffer into which all packets are placed. Although a shared buffer can provide the same probability of blocking as dedicated buffers whose total size is far greater, it needs a bandwidth equal to the sum of all input or output speeds, whichever is higher.

However, buffering strategies also exist within the switch architecture itself, as shown in Figure 22.

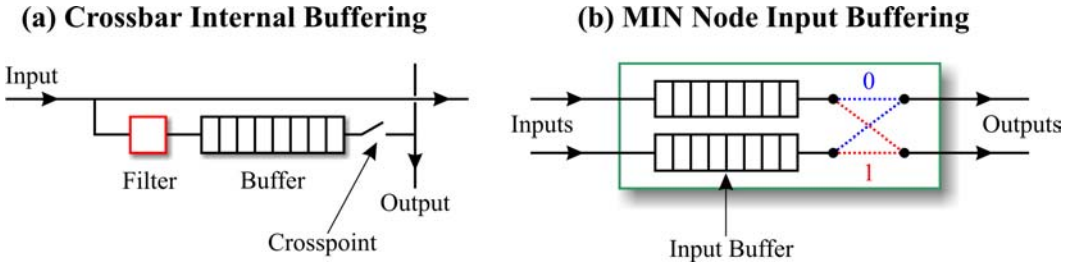

Figure 22: Switch Internal Buffering
Internal buffering allows distributed control in a crossbar switch and eases internal blocking in a MIN switch.

A crossbar switch crosspoint can be buffered internally with a filter selecting packets for its associated output line and buffering them before the crosspoint. An arbiter then decides which packet should be output first, distributing and simplifying the control function to the selection of a packet for an output line rather than the selection of a set of packets across the entire crossbar switch. As MIN architectures are internally blocking, buffering has a more pronounced effect. Integrating input buffers into a node will reduce the effect of internal blocking in a switch by storing packets until the required output is free.

2.2.4 Conclusion

The last decade has seen the development of packet switched broadband architectures that can cope with voice, data and video traffic on a single network. One such architecture is *asynchronous transfer mode* (ATM) [45] which focuses strongly on traffic management issues. The complete transition to such architectures is still a long way off.

Proliferation of network services and continuing technological developments have driven transmission line demand and capacity respectively in recent years. Traffic on the Internet backbone is doubling every three to six months [41] and optical interconnects in the form of wavelength division multiplexing over fibre have increased available capacity. However, packet switch architectures have not scaled accordingly. Unless new technologies are developed soon they will become a restricting factor in the convergence of telecommunication services onto a single, packet switched and broadband architecture.

3 Optical Technologies and Integration

This chapter examines the device technologies that enable optoelectronic interconnection, integration of these devices and the architectures in which they can be used. Attention will be paid specifically to 2D arrays of devices that allow spatial parallelism and therefore the creation of high-density optical interconnects or buses. It is essential to point out that this section is by no means exhaustive. It will illustrate some of today's key technologies paying particular attention to the types of emitter, detector and interconnect used to construct the demonstrators in section 5.

3.1 Semiconductor Theory

This section examines what a semiconductor is and the properties that allow it to be used to create optoelectronic devices. Differentiation is made between direct and indirect bandgap semiconductors thereby explaining the limited use of silicon (Si) and certain other materials in optoelectronic devices.

Unlike the widely spaced and distinct energy levels of a single atom, a crystal lattice has a large amount of similarly spaced energy levels known as bands. Figure 23 shows how these bands are configured in *insulators*, *conductors* and *semiconductors*.

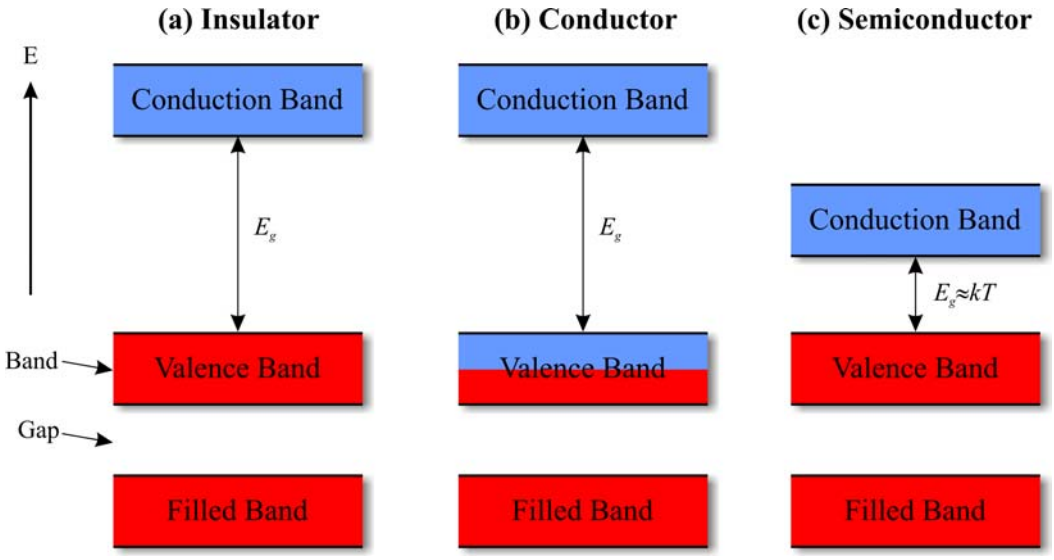

Figure 23: Energy Levels in a Lattice
Specific bands of energy levels exist in a crystal lattice which an electron can occupy. Red indicates electrons, blue indicates holes.

The *conduction band* describes delocalised electrons that are not associated with any specific atom and the *valence band* describes the outer electron layer which is involved in lattice bonding. The *filled bands* are the inner electrons of the atom that are not involved in bonding. An electron can have any energy level within a band but cannot take an energy level outside it.

An insulator, as shown in Figure 23(a), has a valence band completely filled with electrons and an energy gap between valence and conduction bands of typically >4eV [46]. The conduction band is empty and it is hard to alter the material's resistivity by either doping or any kind of external field. A conductor, as shown in Figure 23(b), has a partially filled valence band and an empty conduction band. Its resistivity is typically low and again it is hard to alter the resistivity by doping or through an external field. In a semiconductor, Figure 23(c), the gap between valence and conduction bands is relatively small, usually 1 or 2eV. At extremely low temperatures, a pure semiconductor will approximate an insulator as the valence band will be full and the conduction band empty. As temperatures increase, a semiconductor's small bandgap allows thermal energy kT to excite electrons into the conduction band. Unlike conductors and insulators, the resistivity of a semiconductor can be altered by doping or by applying an external field.

An electron becomes more energetic if it receives enough energy to move into a vacant state either in its present band or in a higher one. In an insulator, the lower states are completely filled and a significant amount of energy is required to get electrons into the conduction band. However, in a conductor there are many vacant states, including some in the valence band, so not much energy is needed to move an electron into one of these states thereby allowing it to participate in a current. In a semiconductor, controlled impurities are used to add charge carriers to the conduction band. This allows resistivity to be altered by up to 10 orders of magnitude [46]. The resistivity ρ of a material [47] is defined by:

$$\rho = \frac{m}{ne^2 t_c} \qquad \textbf{Equation 11}$$

where m is the mass of an electron, n is the number of charge carriers per m^3, e is the charge on an electron and t_c the mean time between charge carrier collisions. The values of m and e are constant and can therefore not be altered. In a metal, any increase in temperature will not appreciably effect the number of charge carriers. However, the

rate at which collisions occur will increase resulting in a smaller mean time between carrier collisions t_c. Therefore, the overall resistance of a metal increases with temperature. This situation is somewhat different in a semiconductor. Although the mean time between carrier collisions is analogous to that of a metal, this variable is more than offset by an increase in the number of charge carriers that become available (since $E_g \approx kT$) resulting in an overall decrease in resistivity with temperature rise. Put succinctly, high temperature applications using semiconductors will require a larger bandgap.

To improve current conduction in a semiconductor, dopants are added at rates of usually less than 1 part per million. The elements added must have a different number of electrons than the host. An *intrinsic* semiconductor is the undoped sample. Take for example an intrinsic group IV crystal such as silicon (Si). Adding a group III dopant to the lattice such as gallium (Ga) will capture one electron from the crystal, in which case it is called an *acceptor*. This missing electron creates a mobile *hole*, considered to be a positive charge, resulting in a positive or *p-type* material. Another electron can occupy the hole, thus creating another hole somewhere else and causing the positive charge carrier to move around. If a group V dopant such as antimony (Sb) is added to an Si lattice it donates an electron and is therefore called a *donor*. The additional electron creates a negative or *n-type* material. The holes in a p-type region or electrons in an n-type region are called the *majority carriers*. *Minority carriers* are holes in the n-type region or electrons in the p-type region.

A semiconductor is capable of both absorbing and emitting photons. Without this behaviour it would not be possible to create semiconductor emitters or detectors. Before we proceed any further though, it is important to remember that there are two types of bandgap as shown in Figure 24 - *direct* and *indirect* [48]. Note that these are E-k diagrams where E represents energy and k momentum wave vector.

First let us consider the absorption of a photon in a direct bandgap material. A photon will be absorbed if its energy is approximately equal to, but not less than, the bandgap energy E_g. If the photon energy is significantly smaller than the bandgap then the material appears transparent and the photon will pass through unhindered. If absorbed, the photon will excite an electron by giving it the energy it carries, transferring it from the valence band to the conduction band and allowing it to participate in any current. The amount of energy a photon imparts is equivalent to the amount it carries:

$$E = \frac{hc}{\lambda} = \hbar\omega \qquad \textbf{Equation 12}$$

where h is the Planck constant $6.63\times10^{-34}\,\text{Js}$, c the speed of light $3\times10^{8}\,\text{ms}^{-1}$, λ the wavelength of the photon and $\hbar$ the standard notation for $h/2\pi$. As a photon has no mass there will be no change in momentum and therefore no change on the k axis. Thus the transition is vertical only.

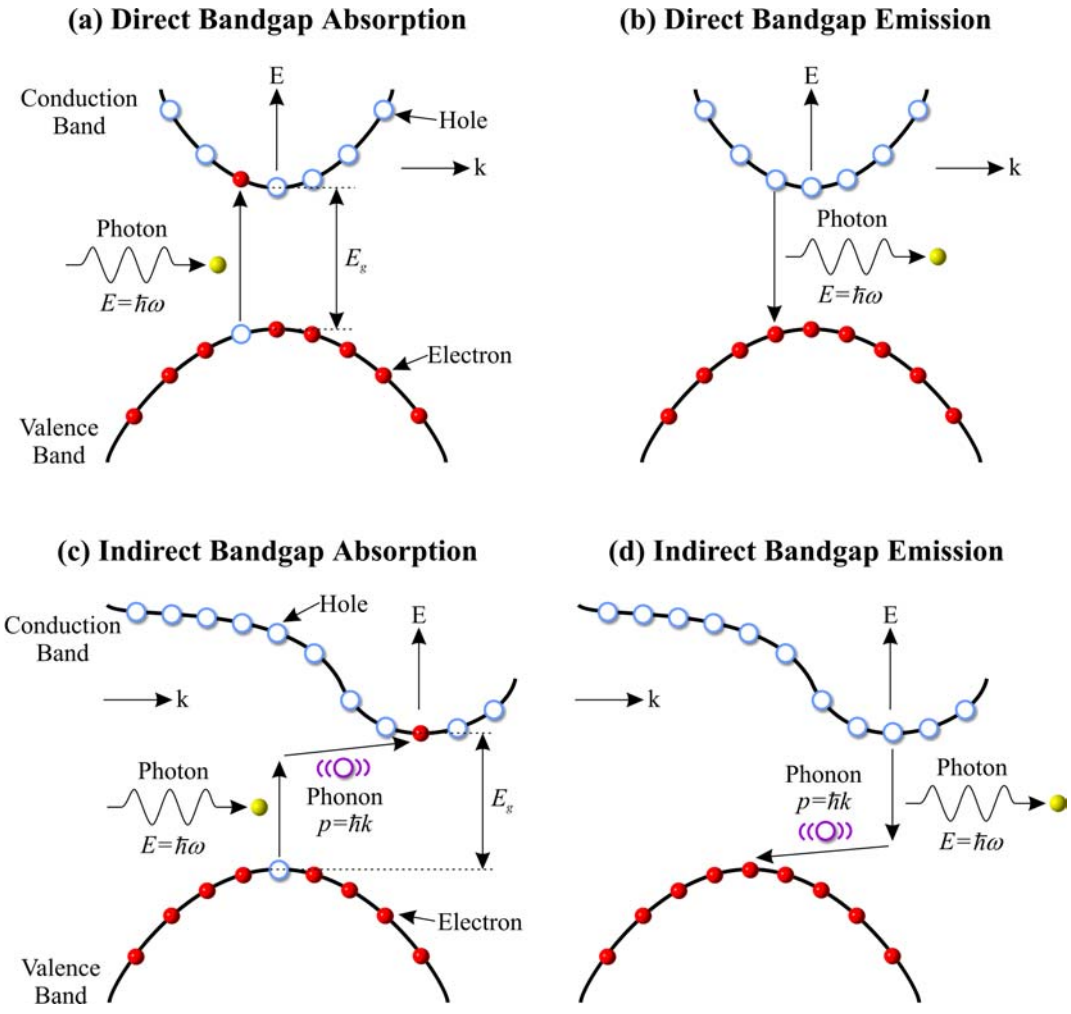

Figure 24: Direct and Indirect Bandgaps

Direct bandgap materials can easily absorb or emit photons. Indirect bandgap materials require both a photon and phonon for absorption and emission which decreases the probability that such an event will occur.

Emission occurs when there is an excited electron in the conduction band. Electrons are eager to lose any energy they carry and return to the lowest energy state either in their current band or the next. Inter-band transitions normally result in the emission of a photon, called spontaneous emission, as shown in Figure 24(b) where energy that the electron had is imparted to the photon.

Indirect bandgap materials are different in that the maximum valence band energy E and the minimum conduction band energy do not occur at the same wave vector k. Therefore, the interaction of a second particle called a phonon is required. A phonon is defined as a quantum of crystal lattice vibrational energy [49]-[50] and has a momentum p of:

$$p = \frac{hk}{2\pi} = \hbar k \qquad \textbf{Equation 13}$$

where h is the Planck constant 6.63×10^{-34} Js, k is the wave vector and $\hbar$ the standard notation for $h/2\pi$. A phonon also carries a small amount of energy E. Note that this momentum cannot be imparted by another photon as photons do not have any mass.

As both a phonon and photon of the correct energy are required, the probability that absorption or emission will occur is greatly reduced as can be seen in Figure 24(c) and (d). Emission is especially problematic as non-radiative recombination is dominant, such as Auger recombination [51], which returns the excited electron to the valence band by emission of multiple phonons before radiative recombination can emit a photon. This directly effects the quantum efficiencies η of devices made from indirect bandgap materials. For example, a typical indirect bandgap device made from silicon carbide (SiC) has a quantum efficiency of around $\eta = 0.0002$, whereas a direct bandgap device made from gallium nitride (GaN) is around $\eta = 0.12$ [52].

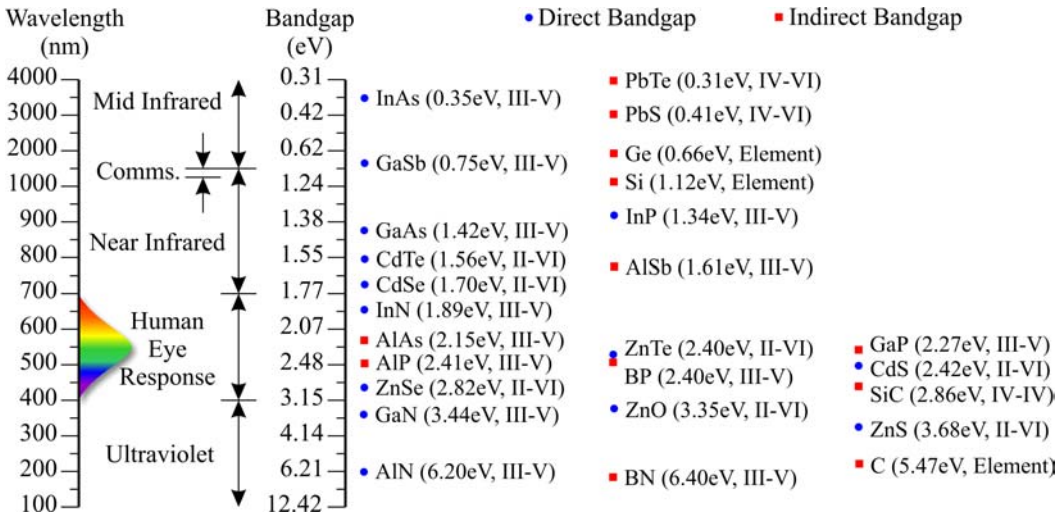

Figure 25: Bandgap Energies of Elements and Binary Compound Semiconductors
Bandgap energies and corresponding photon wavelengths for a variety of elements and binary compound semiconductors. Combinations of compounds give access to intermediate bandgaps. For example, $InAs_xP_{1-x}$ will have a bandgap of between 0.35eV and 1.34eV depending on the value of x. Data taken from [51]-[53].

The advantage of creating optoelectronic devices using semiconductors is that the component is solid state. Such components are compact, energy efficient, cost effective and relatively easy to produce in large quantities. Perhaps the most important characteristic is that semiconductors react quickly to a change in drive current, enabling modulation at comparatively high frequencies.

3.2 Emitters

A component is considered to be an emitter when photons are produced within the device. The wavelength of the photons is material dependent, a selection being shown in Figure 25. This section examines three semiconductor based optical emitters considering their construction, uses, limitations and modes of operation.

The first device which we will consider is the *light emitting diode* (LED), as it was the first to be discovered. In 1907, Round noticed photoluminescence in Carborundum (SiC) [54] when a current was applied essentially creating a Schottky device [48]. Today's LEDs and indeed most if not all optoelectronic devices rely on the p-n junction [55]. The junction shown here is classified as a *homojunction* structure as both sides are made of the same material with the same bandgap even though their electrical characteristics are different.

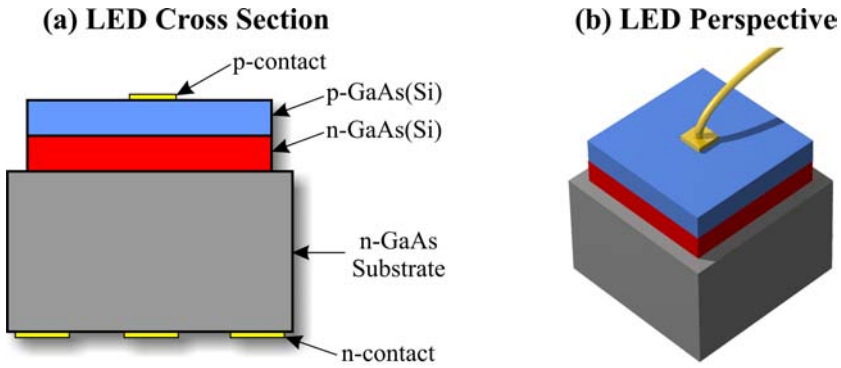

Figure 26: Near Infrared Light Emitting Diode (LED)
Typical NIR-LED structure. Photon emission is spontaneous and not in any particular direction. Doping behaviour of Si is controlled by the temperature of epitaxial growth [56]. Temperatures of above 820ºC form an n-type layer and below form a p-type layer. Patterned n-contact ensures minimum absorption of incident radiation by contact. Layers not to scale.

The first commercial LEDs became available in 1969 and were quickly incorporated as visual indicators on a wide variety of products. Production techniques improved with consequent increases in not only yield but also device efficiency. Today's LEDs are

almost in a position to take over from incandescent filament bulbs and fluorescent tubes as they provide a power efficient "cold" light. Not only that, but their lifetimes are an order of magnitude longer with graceful degradation rather than instantly burning out. The demand for LEDs has continued to grow since their inception and reached the $1billion mark in 2001. However, LEDs for communications purposes have been superseded by laser diodes. This is because the LED is highly divergent and slow by comparison. LEDs rely on spontaneous emission of light on application of a current to modulate their optical output. Unfortunately, spontaneous emission is a slow process and is not normally faster than 1ns [51]-[52], limiting throughput on an LED communications channel to around 100MHz.

The *laser diode* (LD) has become a fundamental building block in many of today's communication and storage systems. Again, it is a p-n junction device, but beam divergence is usually a few degrees with potential bandwidths in excess of 20Ghz [57]. This is because laser based devices use stimulated rather than spontaneous emission to produce light, thus the name *light amplification by stimulated emission of radiation* (LASER). Nevertheless, spontaneous emission and absorption will still occur, however probability remains heavily on stimulated emission's side.

A laser diode works by exciting the *electron population* in the active region to an upper energy band by application of a current. When the majority of electrons are excited, the population is said to be *inverted*, a state which is inherently unstable. If a photon passes through a medium with an inverted population, it can stimulate an electron in the upper energy band return to the lower one emitting another photon which is identical in every way to the first. Partially mirrored surfaces on the device, due to its refractive index, cause internal reflection returning photons back through the cavity and stimulating the emission of further identical photons in a cascade like manner. This causes light amplification within the resonant cavity, presuming there is sufficient current to sustain the inverted population, and is the basis of laser action.

Laser action was discovered almost simultaneously by four research groups in 1962 [2]-[5]. However, all of these groups created homojunction lasers which required high current densities and could not provide *continuous wave* (CW) operation at room temperature. One year later it was postulated that a *heterojunction* design could considerably improve semiconductor lasers [58], a theory which would later award Alferov and Kroemer the Nobel prize. Unfortunately, the technology to fabricate such

structures would not be available until 1969. It took just one year before room temperature continuous wave operation was demonstrated in 1970 [59]-[60]. Figure 27 shows the construction of a typical *double heterojunction* (DH) laser diode.

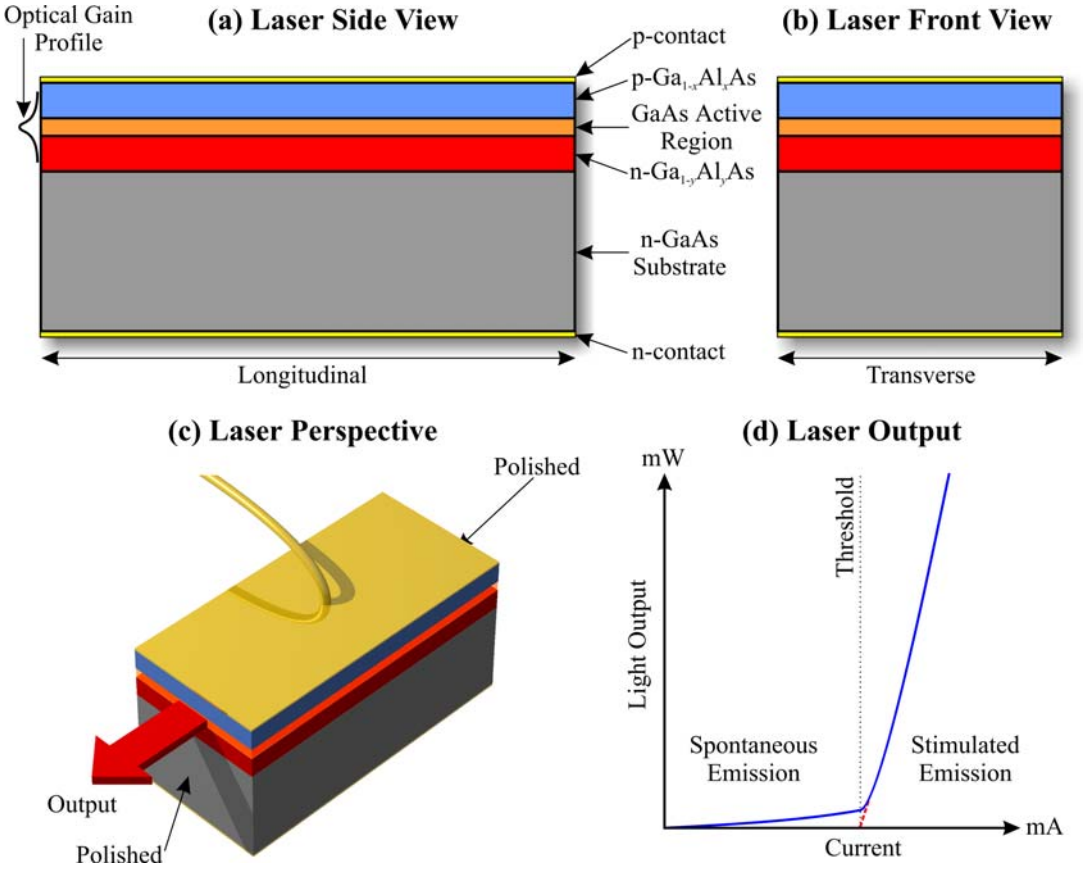

Figure 27: Stripe Laser Diode (LD)

Optical gain is supported in the active region (a). Polishing opposite ends (c) and leaving the remaining sides rough favours laser oscillation along this axis. Below laser threshold (d), spontaneous emission is dominant whereas stimulated emission is dominant above. Layers not to scale.

A heterojunction is a junction between two different materials, in this case GaAlAs and GaAs. Due to the bandgap differences at the two junctions, there is greater confinement of electrons and holes to the active region. In addition, the larger refractive index of GaAs also aids confinement of radiation to the active region. Most stripe diode lasers have an elliptical beam output profile, however this is not regarded as a problem since not only is the output stable but it is easier to couple into an optical fibre. For further information on laser diodes see [48], [57], [61]-[63].

The problem with creating laser diodes is *dicing*. A single substrate can contain hundreds of devices each being an edge emitting device. Thus the substrate must be

carefully cut using a precision diamond saw and the appropriate edges of each device polished. Considering that these devices are usually measured in microns, this can be an awkward and time consuming process. This problem is addressed by the third and final device that we will consider in this section, the *vertical cavity surface emitting laser* (VCSEL).

The VCSEL, as shown in Figure 28, emits perpendicular to the surface of the chip, simplifying fabrication and lowering production costs. Since there is no longer any need for dicing, smaller structures can be created that consume less power.

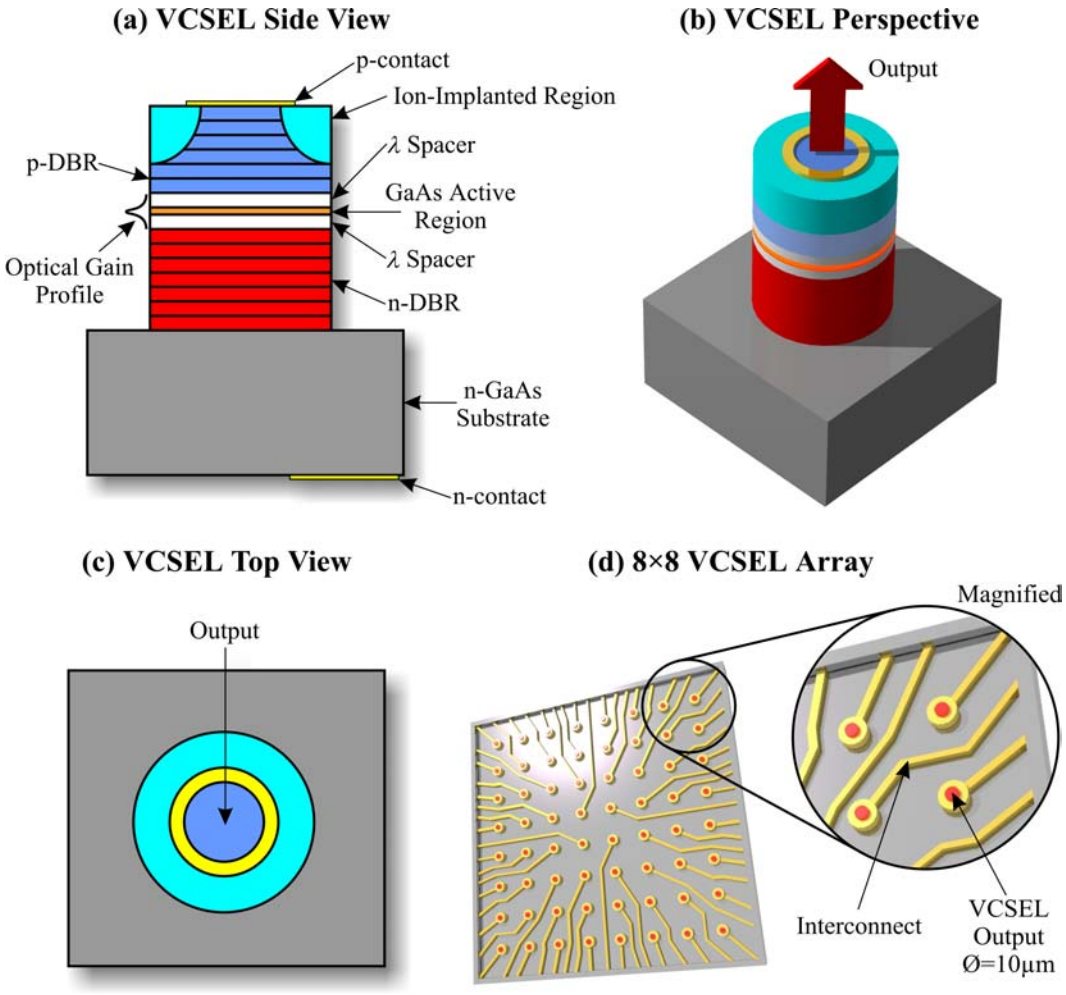

Figure 28: Vertical Cavity Surface Emitting Laser (VCSEL)
Distributed Bragg reflectors (DBR) act as mirrors in a VCSEL. Emission can either be through the top of the VCSEL, as shown, or through the substrate assuming it is transparent at emission wavelength or that a well has been etched. Layers not to scale.

The first VCSEL was built in 1979 [64]. It lased at a temperature of 77K, had a high threshold current and used metal mirrors which had substantial absorption. The construction of efficient mirrors was to remain a problem until 1989 when advances in

epitaxial growth enabled the construction of remarkably effective *distributed Bragg reflectors* (DBRs) and subsequently the first room temperature CW VCSEL [65]. DBRs are created by fabricating quarter wavelength $\lambda/4$ thick layers of alternating high and low refractive index substances such as $Ga_xAl_{1-x}As$ and $Ga_yAl_{1-y}As$ respectively. Well fabricated layers can have reflectivities approaching 99% [66], however poor fabrication results in poor reflectivity and prevents the VCSEL from lasing. The cavity in a VCSEL is fabricated such that a standing wave is formed between upper and lower DBRs, where the maximum is centred on the active region. Centring is ensured by adding spacers so that the cavity length is an integer multiple of wavelengths. Emission from the top of the VCSEL is forced by leaving the upper interface open to air as shown in Figure 28. To force emission from the bottom surface, the top layer is fully metallised with the contact layer. Both configurations have GaAs as the lower DBR interface.

A now common VCSEL enhancement called *oxide confinement* uses implantation of heavy ions, usually fluorine (F) or oxygen (O), to form a circular aperture in the upper Bragg reflector. This channels the carriers into the active region resulting in increased performance from the VCSEL. Unfortunately, ion-implantation is, by its very nature, imprecise. This results in crystal damage, creating fuzzy boundaries around the aperture in the active region with consequent increases in diffraction and divergence. For more information on oxide confined VCSEL arrays refer to [66]-[68].

VCSELs have already taken over from LDs at wavelengths around 850nm, however manufacturing techniques have so far limited their production in the communications wavelength window due to poor DBR reflectivities. This is about to change as improved epitaxial processes are approaching commercial viability which would allow efficient DBR construction for 1.3μm wavelengths [69].

VCSELs are ideal for high density optical interconnects, however array sizes are currently limited to 8×8 or 16×16 devices due to poor yield. Since there is no theoretical limit involved, investment in process technology will allow larger arrays to be fabricated with each VCSEL providing GHz bandwidths.

3.3 Modulators

Modulators differ from emitters in that they transfer information onto an incident optical channel either by changing transmission, reflection or re-routing the beam.

Figure 29 shows the construction of a typical *multiple quantum well* (MQW) modulator. Depending on the potential difference across the active region, this modulator will either absorb any input signal or reflect it along the modulated output path. This device is an example of an *absorptive* modulator.

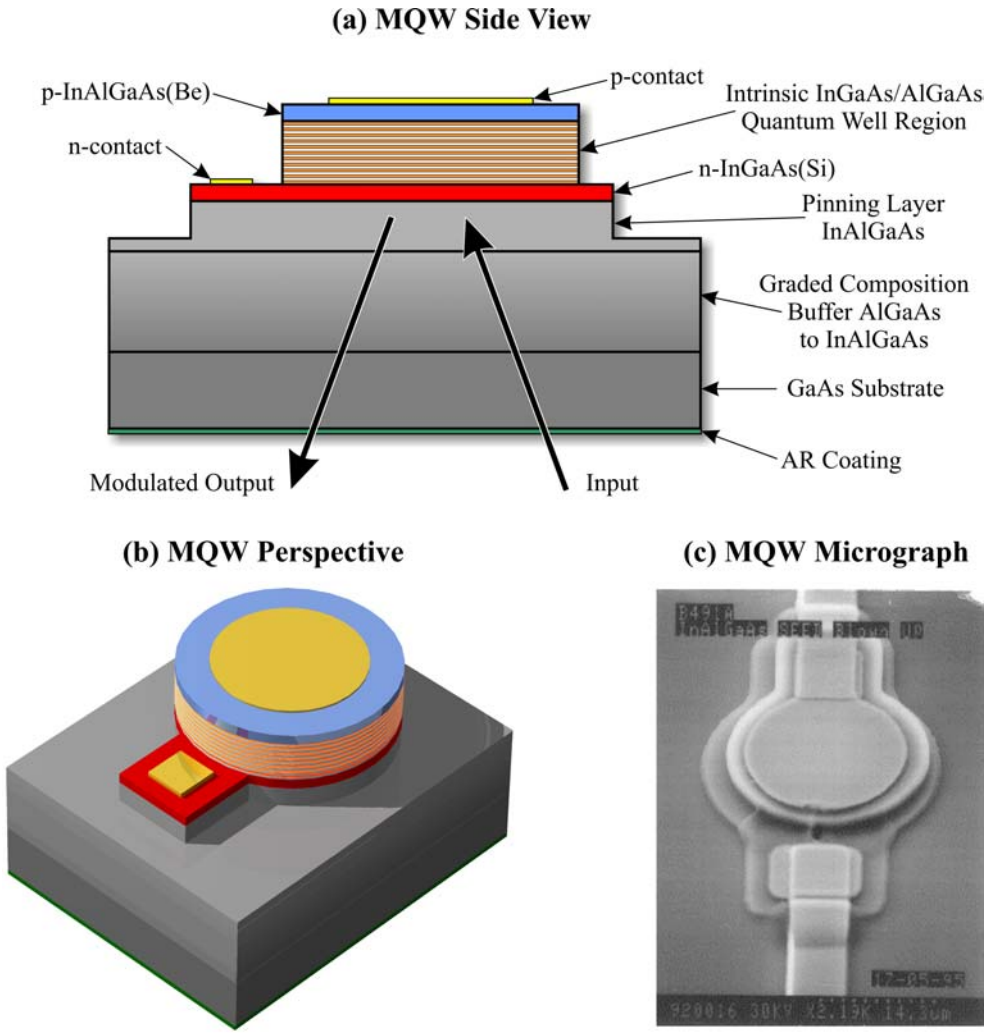

Figure 29: Multiple Quantum Well (MQW) Modulator
The quantum well region modulates information onto an incident optical beam. The substrate is transparent at operational wavelength ~1.04µm. Lattice constants of GaAs/AlGaAs are fairly closely matched, however a graded composition buffer eases lattice tension for the transition to indium (In). Ratios of elements are not included. For more information on the device pictured here see [70]. Layers not to scale.

Absorptive modulators can be created by using one of two effects, either the *Franz-Keldysh effect* (FKE) [71]-[72] or the *quantum confined Stark effect* (QCSE) [73]. The Franz-Keldysh effect states that in the presence of a field in bulk semiconductor, the wavefunctions of electrons and holes tunnel into the bandgap region allowing limited absorption of photons just below the bandgap energy. *Excitonic* interaction [74] adds

to, and possibly dominates, this effect allowing photons to be absorbed in a material that would normally appear transparent. Creating multiple quantum wells of alternating high and low bandgap materials allows use of the quantum confined Stark effect. This differs from the FKE in that application of a field shifts rather than broadens the peak excitonic absorption energy. This is because the quantum wells confine electron-hole pairs, preventing ionisation and allowing the application of larger fields resulting in an increased Stark shift. Note that the QCSE can be shown to be a quantised version of the FKE [75]. A typical MQW device has around 50 to 100 layers each of 5 to 10nm thick with a single chip able to sustain thousands of these devices [76]. Unfortunately, coupling light into a semiconductor is problematic at best leading to lower signal powers than with active emitters. Interestingly, absorptive modulators can be used as detectors if field polarity is reversed.

Reflective and refractive devices work by changing the optical properties of a structure such that the path length is altered [57]. Interference or propagation effects are then used to modulate the beam. Although the crystals in such devices are optimal for waveguides as they provide high transmission, they are comparatively large for semiconductor devices and therefore cannot be directly integrated.

The fact that modulators are not subject to carrier or photon build-up problems as seen in active emitters allows them to be used at higher speeds under certain circumstances. Unfortunately, this is also the principal drawback of modulators: they rely on an external optical source. Manufacturers generally wish to construct systems as efficiently as possible using a single process technology and, if feasible, on a single chip. The requirement of separate modulator and emitter, plus associated alignment, may add unnecessary complexity. The author believes that applications requiring extreme performance will push modulator technology with recent technological breakthroughs placing them well for future development. Specifically, Si compatible polymeric modulators have been demonstrated with a measured bandwidth of 110GHz given a meagre drive voltage of 0.8V [77].

3.4 Detectors

Although many types of optical detector exist, this section will concentrate specifically on semiconductor devices by examining their construction, defining characteristics and electrical properties.

Semiconductor detectors can be broadly categorised into three: *photoresistors*, *phototransistors* and *photodiodes*. The resistance of a photoresistor, such as a cadmium sulphide (CdS) cell, changes depending on the amount of incident light. Unfortunately, their response is normally in the millisecond range and composition not compatible with conventional substrate materials. Phototransistors, which can be fabricated in Si, control the flow of a current based on incident light intensity essentially providing amplification. However, at low light levels their amplification is poor and their frequency response limited to around 200kHz due to carrier diffusion times [55]. Finally we have the photodiode. It is an Si compatible device that converts any incident light into a current. Figure 30 shows a typical photodiode.

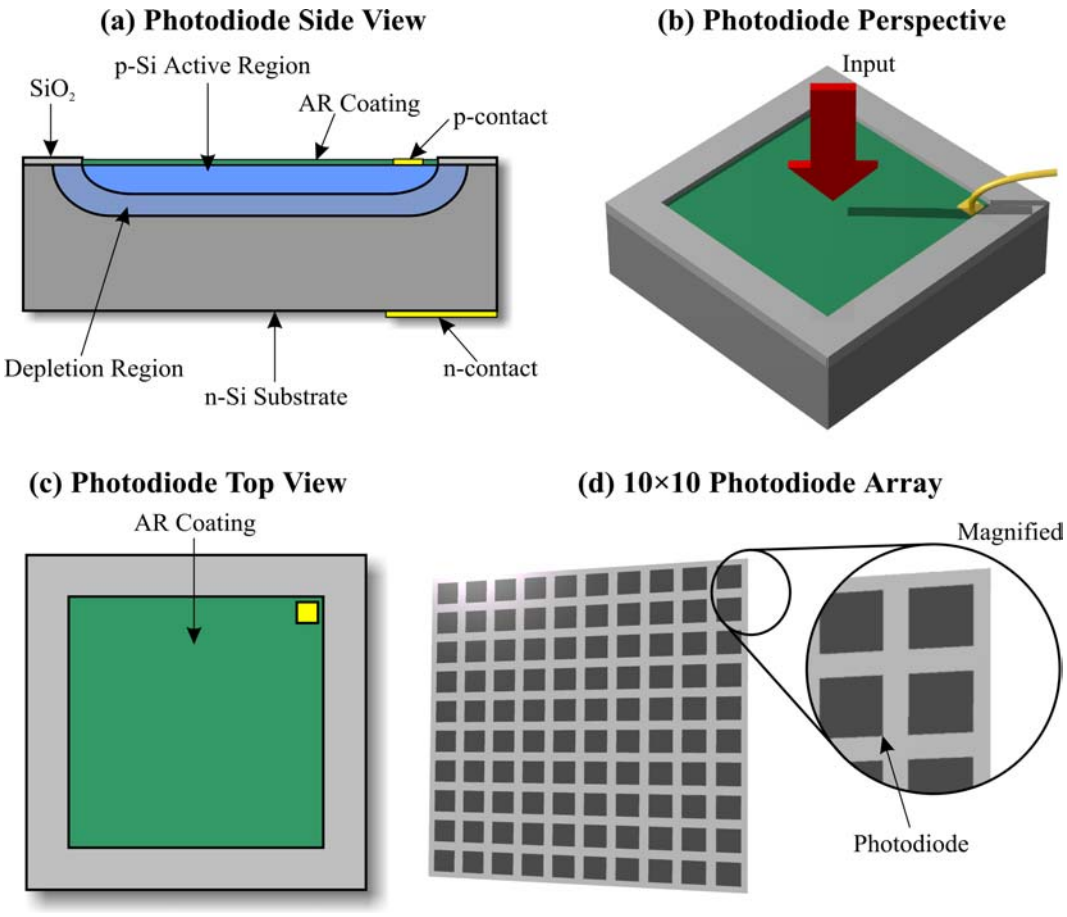

Figure 30: Photodiode Detector
SiO_2 layer masks all but the active region. Array shown in part (d) is not colour coded - each square represents a photodiode. Layers not to scale.

Photodiodes are the dominant optical detector technology. They can be easily fabricated in large arrays using existing technology, offer fast response times and can even count single photons at picosecond speeds in incarnations such as the *avalanche*

photodiode (APD) [78]. Photodiodes are by definition efficient, however recovering a useable signal requires power input which scales directly with bandwidth.

3.4.1 The Photodiode

This section examines the theory behind the photodiode array used in both neural network demonstrators in detail.

When light is incident on a photodiode, a current is produced through external circuitry which is proportional to the light intensity. The photodiode works by exploiting the *photovoltaic effect*. This current response is usually nearly, but not perfectly, linear. Electrons in the semiconductor junction of either p-n or p-i-n type are excited from the valence band into the conduction band by incident photons thus creating a current. This can only happen if the photons carry an energy greater than the bandgap of the detector material. At every wavelength the detector is therefore said to have a *responsivity* $\Re$. It is defined as the ratio of generated photocurrent I_p over incident optical power P_i:

$$\Re = \frac{I_p}{P_i} \qquad \textbf{Equation 14}$$

A real photodiode can be electrically modelled as shown in Figure 31.

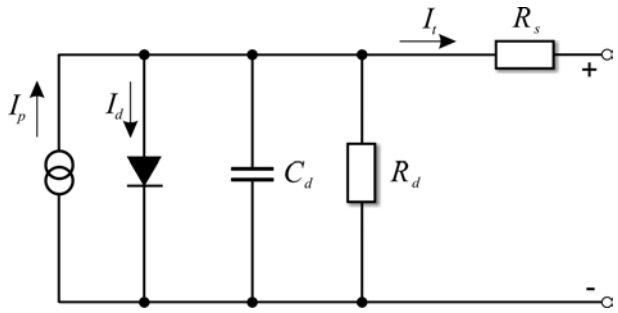

Figure 31: Photodiode Electrical Model
Arrows indicate flow of current.

The photodiode is considered to be an ideal current source, which produces a current I_p as examined above, electrically connected to components used to characterise the photodiode. C_d is the *junction capacitance* since the depletion region acts as the plates of a capacitor. This value limits the maximum detectable frequency and can be lowered by applying a reverse bias voltage. R_d is the *shunt resistance* of the photodiode which is used to determine the noise current in the photodiode when no light is incident. Ideally, the shunt resistance should be infinite, however typical values lie between $1\,\text{M}\Omega$ and $1{,}000\,\text{M}\Omega$. Shunt resistance can be measured by applying 10mV to the photodiode, reading the current and calculating the resistance. Finally, R_s is the *series resistance* of

the photodiode. It determines the linearity of the photodiode when operated in photovoltaic mode and should ideally be zero. A device's series resistance R_s can be calculated by adding together both junction and contact resistances. Typical devices have resistances of less than $1\,\text{k}\Omega$. Under normal circumstances R_s and R_d are considered negligible. C_d is the most important parameter when designing a photodiode receiver.

Photodiodes can be operated in a number of different ways depending on application, as shown in Figure 32.

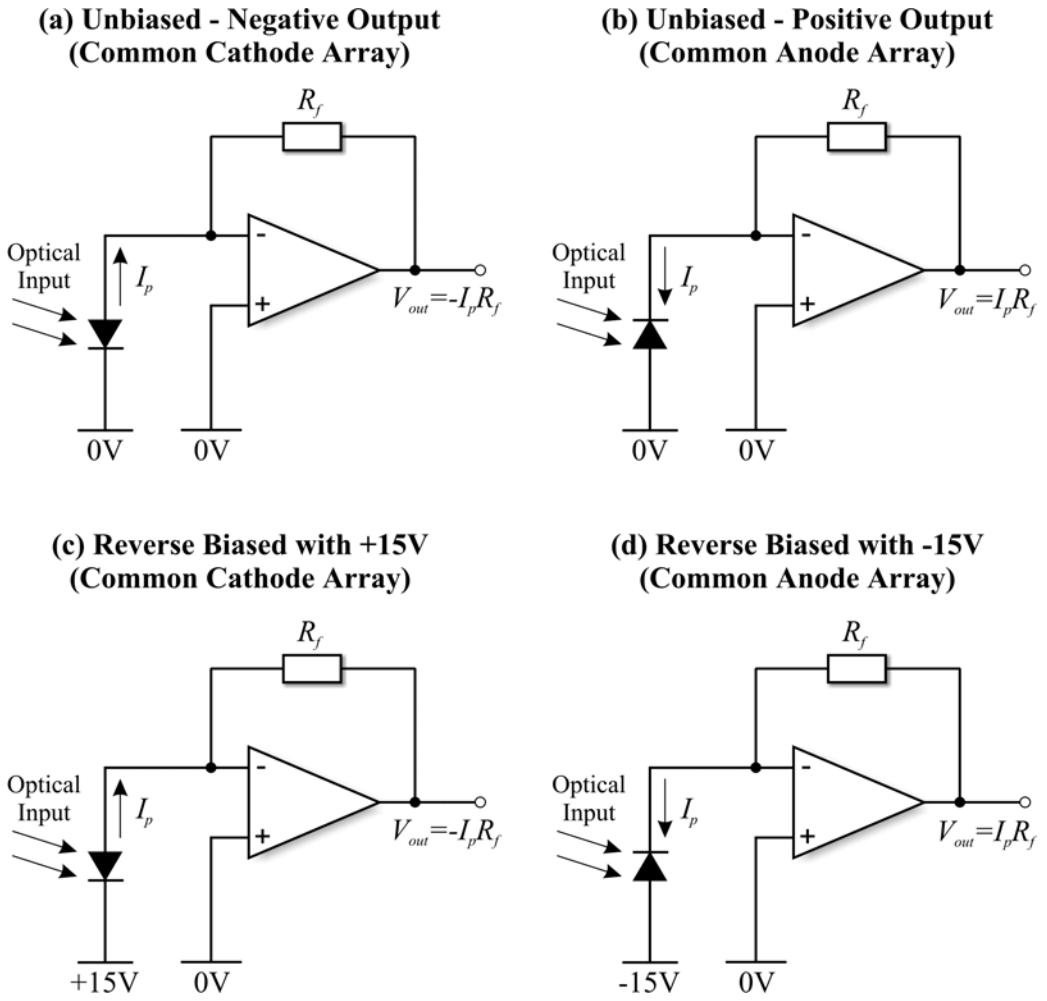

Figure 32: Photodiode Modes of Operation

Unbiased mode has high sensitivity but low bandwidth. Reverse biasing improves bandwidth response but adds to noise.

Firstly, there is *unbiased* or *photovoltaic* mode, as shown in Figure 32(a) and (b), where reversing the photodiode will invert the output signal's polarity. This mode offers low noise and high sensitivity but suffers from reduced bandwidth: such circuits are not normally operated above 350kHz, even when optimised. Secondly, there is *reverse*

biased or *photoconductive* mode, as shown in Figure 32(c) and (d). The reverse bias voltage sweeps electrons out of the junction improving both responsivity and bandwidth. Unfortunately, this also results in a *dark current* I_{dk} which adds to noise. Finally there is *forward bias*. Photodiodes are not normally used in this mode as they simply conduct. Returning to the photodiode electrical model in Figure 31,

$$I_t = I_p - I_d \qquad \textbf{Equation 15}$$

Under reverse or unbiased conditions I_d=0, so the total current is I_t=I_p. However, if a forward bias current is applied then there is a decrease in I_t. If I_t exceeds $-100\,\text{mA}$ the photodiode is usually destroyed as this is enough current to burn the contacts off.

Noise is intrinsic in almost all real systems and the photodiode is no exception. There are two main sources of noise in photodiodes. The first is *thermal noise*, referred to as *Johnson* [79] or *Nyquist* [80] noise. At absolute zero all electrons in the junction remain in the valence band but as the temperature increases they become excited and randomly elevate into the conduction band. This results in an r.m.s. current I_j of:

$$I_j = \sqrt{\frac{4kT\Delta f}{R_d}} \qquad \textbf{Equation 16}$$

where k is the Boltzmann constant $1.38\times10^{-23}\,\text{JK}^{-1}$, T is the absolute temperature in Kelvin, R_d is the photodiode shunt resistance and Δf the bandwidth over which the noise is measured (usually 1Hz). Note that thermal noise is also present in detector electronics and not exclusively in the photodiode.

The second source of noise is *shot noise* [81]-[82]. Shot noise (or *white noise*) is a statistical variation in the current generated by both incident optical power I_p and dark current I_{dk}.

$$I_s = \sqrt{2q\Delta f\left(I_p + I_{dk}\right)} \qquad \textbf{Equation 17}$$

This is again an r.m.s. value where q is electronic charge $1.60\times10^{-19}\,\text{C}$ and Δf the bandwidth over which the noise is measured (usually 1Hz). Dark current I_{dk} is the current that flows in a photodetector when no optical radiation is incident and an operating voltage is applied. It is a combination of surface leakage, generation and recombination of carriers within the depletion region and diffusion to the depletion region of thermally generated minority carriers.

The total r.m.s. noise I_n is a combination of both Johnson and shot noise:

$$I_n = \sqrt{I_j^2 + I_s^2} \qquad \textbf{Equation 18}$$

Noise dominance in this equation depends on the mode of operation. In reverse biased mode, the dark current I_{dk} increases so shot noise I_s becomes dominant over thermal noise I_j. Unbiased mode does not apply a potential difference over the junction, eliminating the dark current I_{dk} and causing Johnson noise I_j to become the dominant term. Therefore, unbiased mode is well suited to ultra-low light level applications as Johnson noise is significantly smaller than the dark current.

The total r.m.s. noise I_n is important as it helps to define the *noise equivalent power* (NEP) at a specific wavelength:

$$\text{NEP} = \frac{I_n}{\Re} \qquad \textbf{Equation 19}$$

This is the amount of incident optical power required ($\text{WHz}^{-\frac{1}{2}}$) to provide a *signal-to-noise ratio* (SNR) of 1. Average values range from $1\times10^{-11}\,\text{WHz}^{-\frac{1}{2}}$ for large area photodiodes to $1\times10^{-15}\,\text{WHz}^{-\frac{1}{2}}$ for small area ones. NEP can be used to compare two similar detectors.

Photodiodes are temperature sensitive but their response to temperature change is dependent on the mode of operation. In unbiased mode, an increase in temperature will result in a decrease of shunt resistance R_d. R_d is halved for every 6K increase in temperature [83]. In reverse biased mode, the dark current I_{dk} is doubled for every 10K increase in temperature [13]. Although the exact change is device dependent, the trend remains the same with unbiased mode being more sensitive to temperature increase. It should be noted that there is also a responsivity change with temperature but this is material dependent. For example, lowering an Si photodiode's temperature improves responsivity at shorter wavelengths (blue to ultra-violet) and increasing it improves responsivity at longer wavelengths (near infra-red). This change does not normally exceed a few percent under normal operating conditions.

Reverse biasing a photodiode is essentially a speed to noise trade-off. The demonstrators described in this thesis use low optical powers, however it is not necessary to minimise noise in a neural network as will be discussed later. Therefore both approaches were taken, with the first generation demonstrator using a biased photodiode configuration and the second generation an unbiased one.

3.4.2 Transimpedance Amplifiers

Transimpedance amplifiers, or current to voltage converters, take a current produced by a photodiode and convert it into a voltage. In so doing, they improve both the linearity and bandwidth of photodiode response. They are the basis of photodiode amplifiers and an essential, yet commonly overlooked, aspect of optical receiver design. This section examines the theoretical design limitations and considerations of a transimpedance amplifier. An understanding of this section is assumed in chapter 5.

The frequency response of any photodiode amplifier is a complex AC problem which depends on many different design parameters [84]-[85]. Figure 33 shows the unbiased photodiode receiver circuit examined in this section.

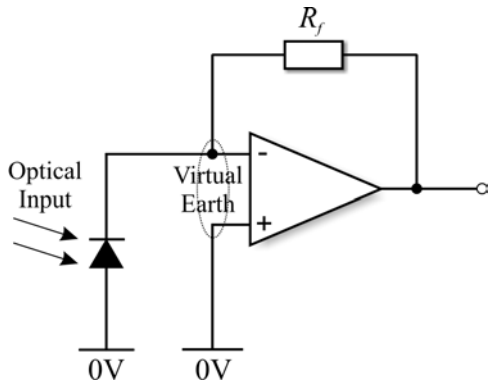

Figure 33: Unbiased Photodiode Transimpedance Amplifier
This configuration creates a virtual earth at both positive and negative amplifier inputs.

The bandwidth of a photodiode transimpedance amplifier is limited in one of three ways: *parasitic capacitance*, *op-amp bandwidth* or the requirement for *phase compensation*. Each of these three limitations will be examined in turn and design equations given which allow circuit optimisation.

Parasitic capacitance is inevitable in any resistor and is normally modelled by a capacitor C_p in parallel with the resistance affected, in this case R_f. The value of C_p is usually around 0.25pF for normal resistors but this is increased by poor PCB layout. When dominant, the -3 dB point has a frequency of:

$$f_p = \frac{1}{2\pi R_f C_p} \qquad \textbf{Equation 20}$$

This equation shows that it is usually dominant when R_f is a large value, i.e. circuit amplification is high.

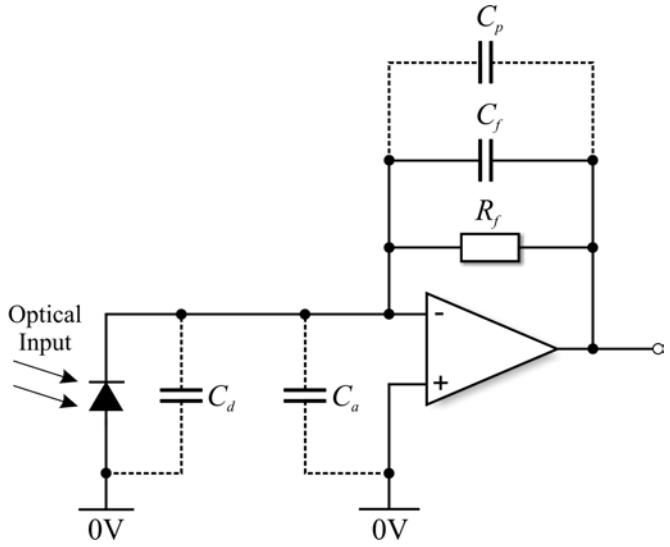

Figure 34: Operational Amplifier Bandwidth Limitations
Circuit showing photodiode junction capacitance C_d, op-amp input capacitance C_a, parasitic capacitance C_p and feedback capacitance C_f.

The next limitation is op-amp bandwidth. Normally, the gain bandwidth product of the amplifier imposes a limit in voltage mode operation. However, this is not the case in current to voltage mode. Figure 34 incorporates additional components which allow us to analyse this circuit's limitations. These additional components are C_a, which is the input capacitance of the operational amplifier (normally around 3pF), and C_d, the photodiode capacitance.

Sketching the response plot, as shown in Figure 35, aids frequency response analysis.

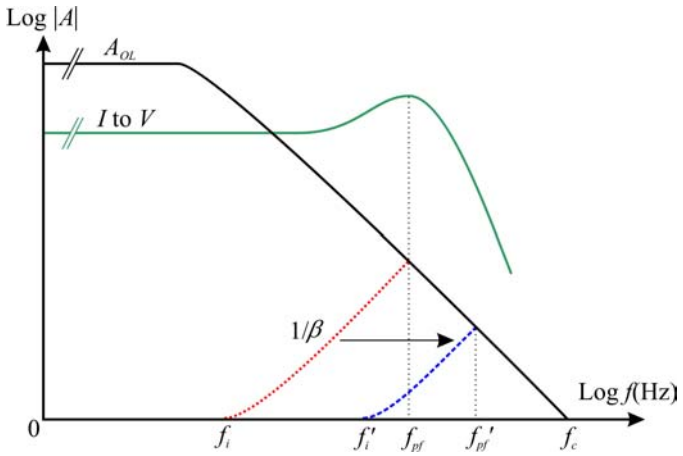

Figure 35: Transimpedance Amplifier Frequency Response Plot
Current to voltage response depends on open loop gain A_{OL} and feedback factor $1/\beta$.

The I to V curve (solid green line) indicates transimpedance amplifier gain which peaks at f_{pf}. This frequency is defined using two other parameters. Firstly by A_{OL}, which is the open loop gain of the amplifier. At lower frequencies this gain has a fixed value.

However, as operating frequency continues to rise this parameter hits a point at which it begins to drop steadily. This is because the product of gain and bandwidth is a fixed value, intrinsic to the amplifier chosen, which peaks at A_{OL} at low frequencies. At frequency f_c the amplifier has no more gain left to give as this is the unity gain bandwidth product. Secondly, the peak is also defined by the feedback factor $1/\beta$ (dotted red line), where β is the fraction of amplifier output that feeds back to the input. Initially, this value is zero, but it begins to rise at a rate of 20dB per decade from frequency f_i. This frequency can be calculated using:

$$f_i = \frac{1}{2\pi R_f (C_d + C_a)} \qquad \textbf{Equation 21}$$

The intercept of these two curves is important as it defines the point at which available gain is not enough to satisfy the feedback factor $1/\beta$. Peaking is usually exhibited at this frequency f_{pf} and it can be calculated by taking the geometric mean of f_c and f_i:

$$f_{pf} = \sqrt{f_c f_i} = \sqrt{\frac{f_c}{2\pi R_f (C_d + C_a)}} \qquad \textbf{Equation 22}$$

After this point frequency response drops off by approximately 40dB per decade. For a two pole system such as this, we therefore have a bandwidth of:

$$f_{bw} = 1.4 f_{pf} \qquad \textbf{Equation 23}$$

To improve this bandwidth response one can either choose an amplifier with a larger f_c parameter or increase f_i. As can be seen from the equations above, f_i is dependent on both input capacitance and feedback resistance. Any decrease in capacitance or resistance will increase frequency response, as shown in Figure 35 by the dashed blue line for f_i' and f_{pf}'. To maximise bandwidth rather than the gain-bandwidth product, these values must be moved to the right as far as possible. To do so, one selects a feedback resistor using the following equation:

$$R_f = \frac{1}{2\pi f_c (C_d + C_a)} \qquad \textbf{Equation 24}$$

This will move f_i' to the same value as f_c, utilising the entire amplifier bandwidth. Circuits using this value of feedback resistor are inherently stable and will not require any phase compensation.

Reverse biasing is a useful method for reducing photodiode capacitance C_d and increasing bandwidth. Capacitance C_d can easily be reduced, unlike that of the amplifier C_a, and since it is normally far greater and therefore dominant, any reduction

will have a significant impact on bandwidth. Applying a reverse bias voltage V_{rb} to the photodiode will result in a new capacitance of:

$$C_d' = \frac{C_d}{\sqrt{1 + \frac{V_{rb}}{\varphi_d}}} \qquad \textbf{Equation 25}$$

where C_d is the unbiased diode capacitance and φ_d the built-in voltage of the diode junction (usually ~0.6V). Unfortunately, this technique introduces a dark current noise, as discussed previously, making it unsuitable for small signal, high gain applications.

Resonance is a problem that seriously affects transimpedance amplifier performance and design. Resonance occurs at frequency f_r and is responsible for gain peaking in the frequency response plot. Frequency f_r can be determined using:

$$f_r = f_{pf} = \sqrt{\frac{f_c}{2\pi R_f (C_a + C_d)}} \qquad \textbf{Equation 26}$$

If no attention is paid to resonance then the amplifier design could be unstable or even oscillatory.

The requirement for phase compensation is the final limiting factor in transimpedance amplifier design. It is used to remove resonance and stabilise the circuit. Unfortunately, circuit stability is traded off against bandwidth. Phase compensation is achieved by adding a further feedback capacitor C_f in parallel with the feedback resistor R_f. The value of this capacitor can be approximated using:

$$C_f = \sqrt{\frac{C_a + C_d}{2\pi R_f f_c}} \qquad \textbf{Equation 27}$$

or calculated more precisely for applications sensitive to parasitic capacitance using:

$$C_f = \frac{1 + \sqrt{1 + 8\pi R_f (C_a + C_d) f_c}}{4\pi R_f f_c} \qquad \textbf{Equation 28}$$

If the capacitor value turns out to be zero or negative, then there is enough inherent parasitic phase compensation in the circuit to ensure stability. This is usually the case in high gain applications. A useful check to ensure amplifier stability with any value of compensation capacitor C_f is:

$$\sqrt{\frac{f_c}{2\pi R_f (C_d + C_a + C_f)}} > \frac{1}{2\pi R_f C_f} \qquad \textbf{Equation 29}$$

Unfortunately, the addition of this feedback capacitor C_f means that there are changes to the equations specifying both parasitic cut-off frequency f_p and $1/\beta$ feedback frequency f_i. Presuming that $C_f >> C_p$, f_p now becomes:

$$f_p = \frac{1}{2\pi R_f C_f} \qquad \textbf{Equation 30}$$

A reduction in f_i also decreases bandwidth. Again, presuming C_f dominates C_p, f_i becomes:

$$f_i = \frac{1}{2\pi R_f \left(C_d + C_a + C_f\right)} \qquad \textbf{Equation 31}$$

These equations must be used in any circuit that has a phase compensation capacitor. Presuming C_f has been matched adequately, the resonance frequency equations are now irrelevant as this capacitance prevents circuit oscillation, damped or otherwise, and peaking.

Noise analysis in this circuit is problematic at best. One new fixed source of current noise is thermal noise from the feedback resistor R_f:

$$I_f = \sqrt{\frac{4kT\Delta f}{R_f}} \qquad \textbf{Equation 32}$$

where k is the Boltzmann constant $1.38\times10^{-23}\,\mathrm{JK}^{-1}$, T is the absolute temperature in Kelvin and Δf the bandwidth over which the noise is measured (usually 1Hz). This noise current is combined with junction and shot noise to give a total amplifier input current noise of:

$$I_i = \sqrt{I_j^2 + I_s^2 + I_f^2 + I_a^2} \qquad \textbf{Equation 33}$$

where I_a is the amplifier input noise. Theoretical determination of amplifier input noise is complex as it consists of five different noise sources, each dominant between certain pole frequencies. Further information can be found in [84]-[86].

This section is by no means comprehensive. After initial design, the circuit will almost inevitably require experimental adjustment to deal with component and PCB layout issues.

3.5 Optical Interconnect Elements

This section examines optical components that can be used to create static or dynamic two dimensional interconnect patterns in free space.

The first component considered is the *diffractive optic element* (DOE), the operation of which was first demonstrated in 1967 [87]. In the same way that a diffraction grating divides an incident beam in one dimension, the DOE shapes a beam in two dimensions to create a desired intensity profile in the far field of a Fourier lens. These devices are planar elements consisting of areas which retard incident light. They can perform complex optical functions which may have previously required several optical components, that is if the function was at all feasible in any other manner.

DOEs are compact, can be constructed using robust materials such as silica and are simple to manufacture using existing VLSI fabrication techniques. They can be mathematically described by *kinoforms* [88]-[89] and are fabricated as either binary or multilevel. Binary elements use a single stage fabrication process resulting in good uniformity across the array. Multilevel structures, as shown in Figure 36, require as many fabrication steps as there are layers, leading to an increase in non-uniformity due to alignment mismatch of 1% to 2% per layer. However, multilevel structures result in substantially improved transmissions. Note that light which is not transmitted is normally scattered outside the diffraction window and not absorbed by the DOE.

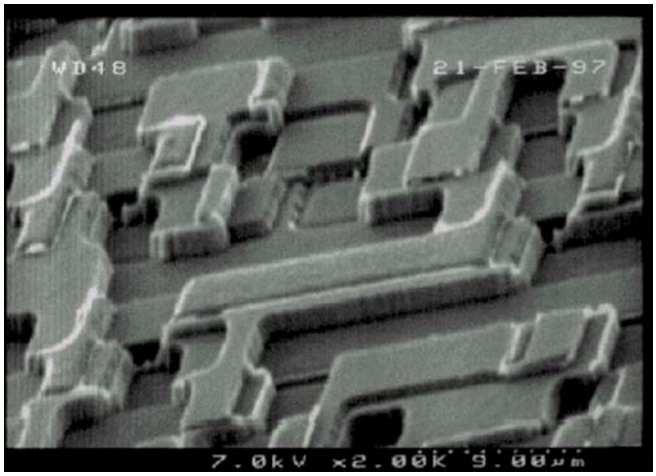

Figure 36: Multilevel Diffractive Optic Element
Surface micrograph of a multilevel DOE.

A repeating pattern exists in the DOE which is referred to as its period and defines the maximum angular divergence. As the grating period gets smaller, the maximum diffraction angle increases. DOEs are matched to a specific wavelength to minimise or eliminate the zero diffraction order. Complex and large scale interconnection patterns can be created using a DOE. Heriot-Watt University has fabricated devices in-house capable of fanning-out to 128×128 elements. Obviously such large scale interconnects

are limited by input beam intensity since every fanned-out channel must have a large enough fraction of the input beam power to make it detectable.

The second component examined is the *spatial light modulator* (SLM) [90] which works in the same way as the DOE except that it is programmable. These devices are based on *liquid crystal displays* (LCD) in that they contain a large number of individually addressable voltage controlled pixels. Figure 37(a) illustrates the operation of a *twisted nematic* (TN) device with no voltage applied. The liquid crystal cell rotates the polarisation state of any incident light thus allowing it to pass through the analyser. In Figure 37(b), a voltage is applied across the liquid crystal which prevents rotation of polarisation state. Since vertically polarised light cannot pass through a horizontal analyser there can be no transmission of light.

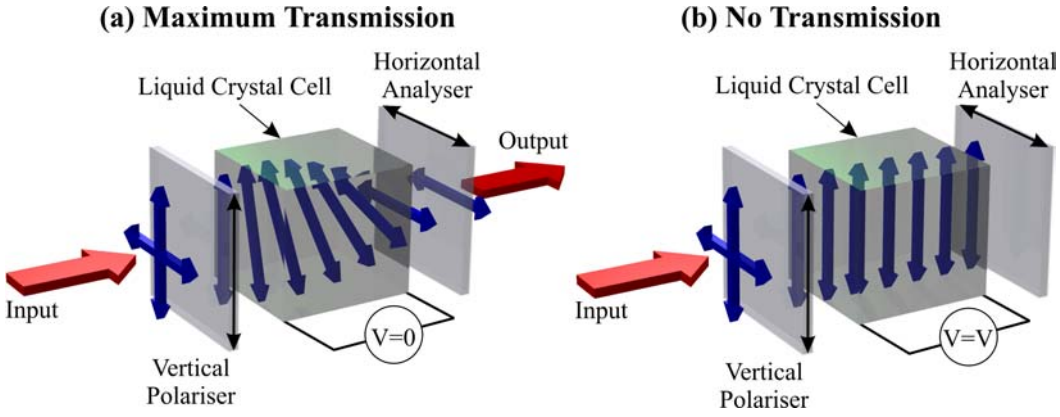

Figure 37: Spatial Light Modulator

Vertical polariser converts the input beam to a single vertical polarisation state. Horizontal analyser only allows transmission of horizontally polarised light. This device is configured to modulate amplitude. Replacing the horizontal analyser with a vertical one inverts the effect of any applied voltage.

When used as shown to modulate amplitude, the SLM can be used to control the routing of a specific transmission channel. Removal of both polariser and analyser generates phase lag rather than amplitude modulation allowing an SLM to be used as a programmable DOE.

The use of SLMs tends to be limited by their relative expense, complex control logic and slow refresh rates. If used to control phase, configurations need to be stored in memory as computation of a new configuration requires a large amount of processing power and is therefore not feasible in real time. The devices do not particularly suffer from fabrication limitations, indeed megapixel devices already exist [91], but rather from liquid crystal response times and serial reconfiguration. As SLMs are addressed in

a serial manner, larger arrays require longer reconfiguration times consequently reducing the refresh rate of the entire array. These disadvantages mean that SLMs rarely have frame refresh rates of greater than a few kilohertz.

3.6 Optoelectronic Integration

This section examines the degree to which optoelectronic interconnects can be integrated into existing systems [92] and the engineering issues that arise.

The extent of optoelectronic component integration will depend primarily on identified bottlenecks in any architecture. Applications range from low level integration of optical components directly onto a chip to high level integration of fibre ribbons or bundles connecting remote systems and boards. Figure 38 illustrates a few different levels of integration.

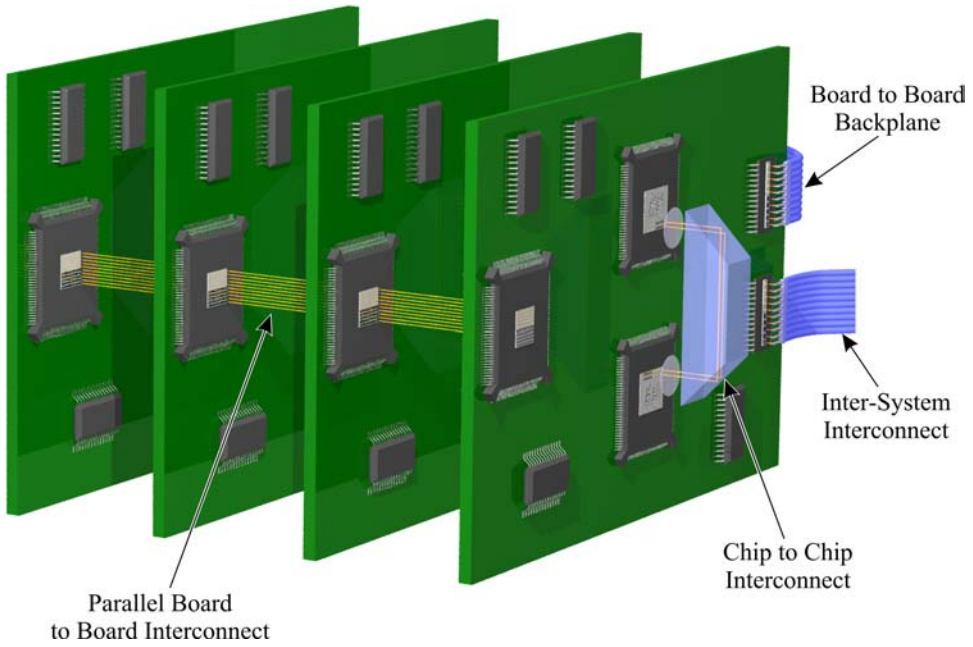

Figure 38: Optical Interconnection
Optical interconnects can be integrated at a low level interconnecting chips directly or at a high level connecting systems to an optical fibre based network.

Starting from the highest level and working down, fibre interconnects can be employed to integrate systems or boards using a dedicated component such as that shown in Figure 39. Fibres are directly butted up to an emitter or detector, or to a waveguide which channels the signal. Optical fibres can be used for board to board interconnection to create a backplane between multiple system boards with little attention to

arrangement or alignment. Few changes are needed to use the same components for connection of remote systems at the same data rates as local boards.

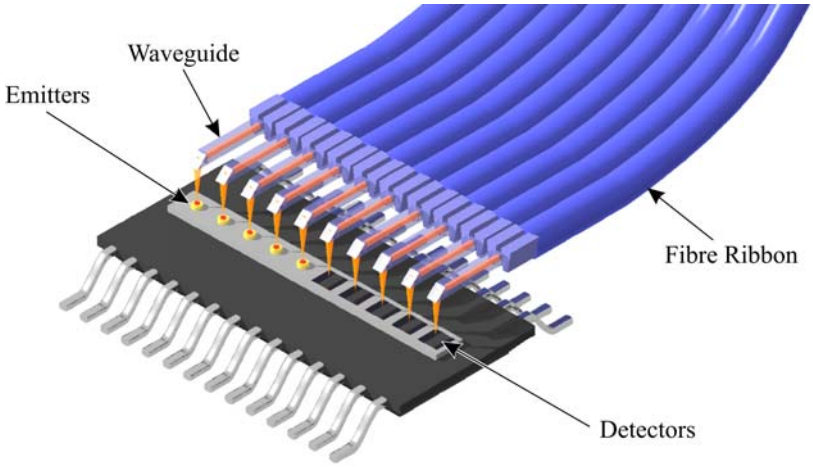

Figure 39: Fibre Ribbon Interconnect
This connector uses a waveguide to reflect the signal through 90º and into or out of a fibre ribbon.

Free space can also be used to interconnect boards in a highly parallel manner. This is a short range solution and cannot be practically implemented over more than a few tens of centimetres. Again, a dedicated interconnection chip is used but it must be kept in alignment with its counterpart as shown in Figure 40. The transmission media can be composed of a single block waveguide, sealing the system from potential environmental interference such as dust.

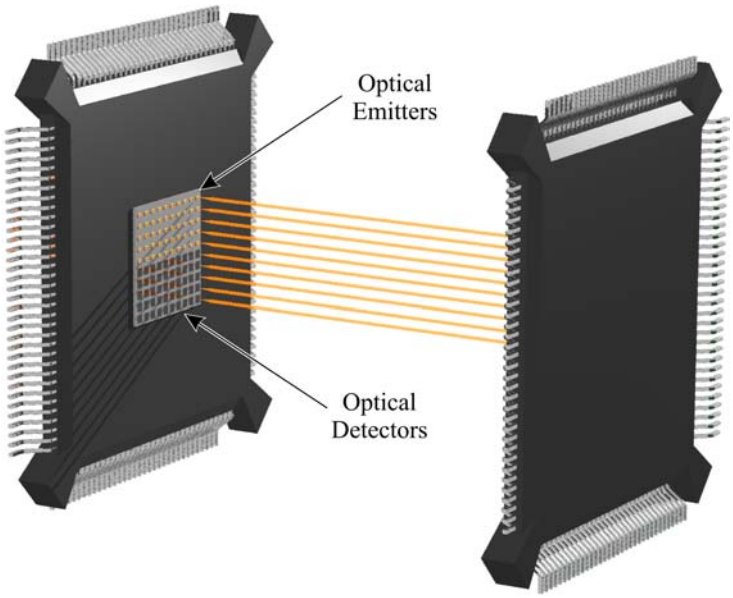

Figure 40: Board to Board Interconnect
Systems such as this usually require focussing. This is achieved using a bulk lens or micro-optic lenses fitted over every VCSEL, neither of which are shown.

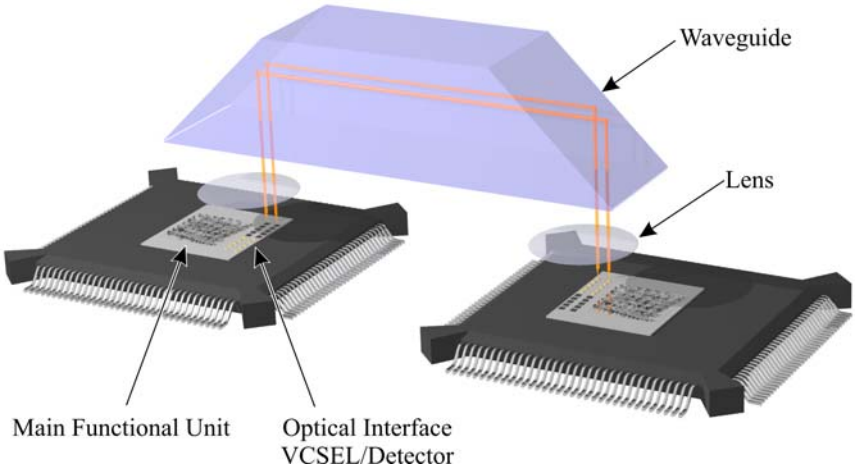

Figure 41: Chip-to-Chip Interconnect
Components directly integrated onto a die with unrelated functionality.

The ultimate goal of optical interconnection is to integrate components directly onto chips with high bandwidth requirements. Figure 41 illustrates a chip-to-chip interconnect using a single waveguide in a free space type configuration. The two chips shown here could be a processor and its memory, the connection between which is notoriously slow. Eventually optical interconnects may be fabricated across a single chip to replace long lines or improve clock signal distribution. Such routing can be achieved using polysilicon or SiO_2 waveguides as they can be fabricated directly onto the chip's surface.

Integration of optoelectronic components onto existing Si substrates is limited by the indirect bandgap in Si. The extensive use of Si is due to its low cost, robust nature and the ease with which it can be processed. Although detectors can be fabricated in Si, emitters and modulators require another direct bandgap material. However, such materials cannot be directly deposited onto Si as they typically have a different size of crystal lattice. This *lattice mismatch* causes strain within the device, resulting in cracks if the mismatch is too great, which can render connections and thereby entire components useless. There are two ways to ease or eliminate this problem using either *flip-chip bonding* or *strained layers*.

Flip-chip bonding attaches two intrinsically different substrate types using *bumps*. Figure 42(a) shows an example of *solder bump* flip-chip bonding. This technique attaches a direct bandgap material to Si by depositing solder balls onto bond pads, as seen in Figure 42(b), and then carefully positioning the complementary direct bandgap substrate on top. Controlled heating is used to reflow the solder balls thus making a

connection. Unfortunately this technique can result in strain between both layers as the solder balls are inflexible when cool and different substrates have different coefficients of thermal expansion. Note that since solder bumps are typically 15µm in diameter, the substrate must be transparent or etched through to transmit any optical signal.

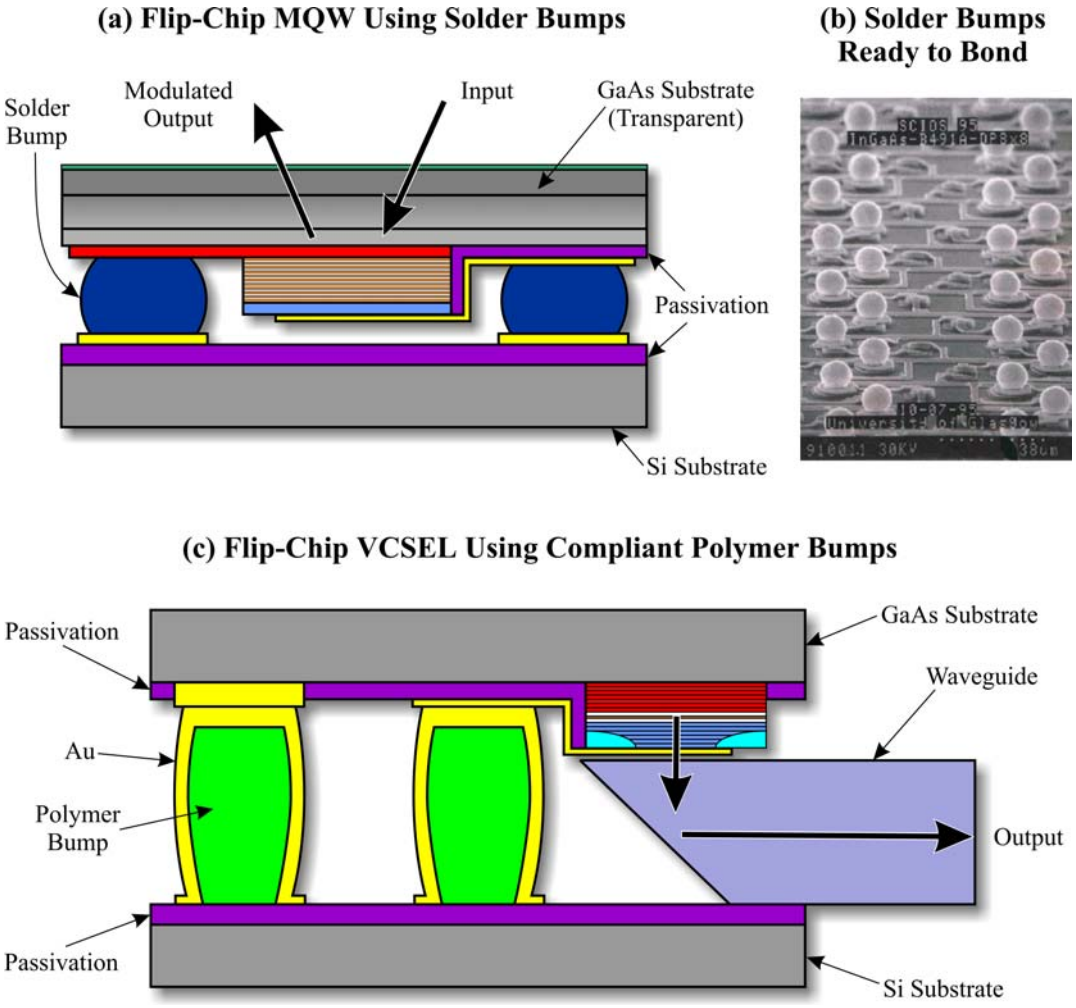

Figure 42: Flip-Chip Bonding of Incompatible Substrates
Solder bumps reflowed to make contact. Polymer bumps compressed to 80% of original size to make contact. Layer detail for MQW on page 40 and VCSEL on page 38. Layers not to scale.

An alternative flip-chip bonding technique, called *compliant polymer* flip-chip bonding [93], is shown in Figure 42(c). Elastic polymer bumps of around 100µm high and 50µm in diameter are created on an Si substrate and coated in gold so that they conduct. Their large dimensions allow waveguides to be integrated between both substrates. The complementary substrate is then lowered on to the bumps and pressure applied before the substrate is tacked in place. Compression of the bumps to 80% of their original size ensures a good pressure contact. Strain is considerably less than with solder bumps as

polymer bumps are elastic by definition thus allowing for movement through warpage and shrinkage.

Direct growth of an epitaxial layer on top of a dissimilar substrate, or *heteroepitaxy* [56], is sometimes used to integrate direct bandgap materials with Si. One method is deposition of a thick epitaxial layer to give a *strain relaxed* junction. The theory is that this layer is thick enough to ensure that few defects reach the surface. However, many defects still propagate into the epitaxial layer making successful manufacture of even simple circuits a difficult process. Deposition of a large number of epitaxial layers of alternating composition below a certain critical thickness results in a *strained layer superlattice* [94]. This allows strain to be dealt with gradually over many layers but does not prevent defects, it only reduces their occurrence. The final method is *pseudomorphic* growth. By ensuring that the thickness of an epitaxial layer is less than the critical thickness at a particular mole fraction the layer structure is forced to conform to that of the previous layer. Only the latter method showed any commercial potential in 2001, specifically for creating devices using SiGe [95].

Two recent developments indicate that both flip-chip bonding and heteroepitaxy may soon be superseded. The first development is the construction of a reasonably efficient Si LED [96]. This is done by creating a series of inverted pyramids which reflect light back into the semiconductor. This light trapping effect results in efficiencies of 1% at room temperature, an efficiency increase of between one to two orders of magnitude. The second and most promising development is the discovery that a buffer layer of the ceramic strontium titanate ($SrTiO_3$) eliminates strain and allows the efficient growth of III-V direct bandgap semiconductors onto an Si substrate [97]-[98]. GaAs electronic circuits on an Si substrate have already been constructed using this technique with dimensions which far exceed previous GaAs size limits.

3.7 Optical Highways

The concept of optical highways [99]-[100] envisages a high bandwidth low latency general purpose multiprocessor interconnection architecture. Figure 43 schematically shows such a highway which is used to connect nodes in an arbitrary topology where each node has access to more than 1,000 channels. A node is considered to consist of a processing element and shared memory space. The interconnect is point to point and hard wired, with several thousand channels being passed to and from each node via an

optoelectronic interface into a free space optical relay system which can hold several hundred thousand channels. The number of nodes that can be interconnected in a specific network topology is primarily limited by aberrations in bulk optic lenses.

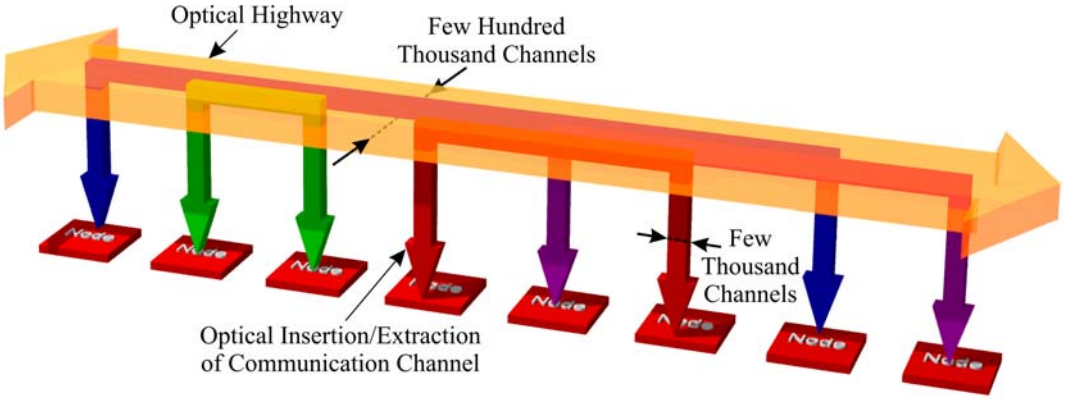

Figure 43: Optical Highway

Free space optical highways interconnect multiple nodes through a series of relays that are used to add or drop thousands of channels at a time.

Polarising optics are used to route channels as shown in Figure 44. A polarising beam splitter deflects channels of a specific polarisation to a node with each channel's polarisation state determined by patterned *half wave plates* (HWP).

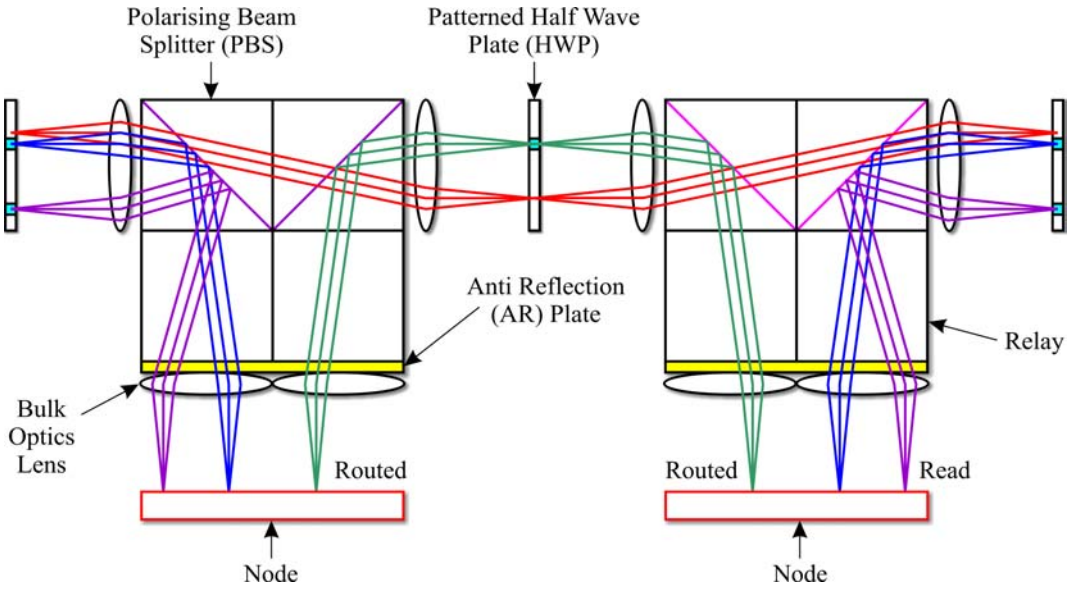

Figure 44: Optical Highway Construction

Polarising optics define a fixed network topology.

Optical highways can be made reconfigurable using an SLM in place of the patterned HWP. In order to maintain efficiency, run-time reconfiguration cannot be performed since it requires a reconfiguration controller with associated hardware reconfiguration delays. Compile time reconfiguration is desirable but not necessarily a requirement.

Regardless of when reconfiguration occurs, most algorithms will need to be adjusted for computation on a multiprocessor system. It would therefore be simpler to fit the algorithm to a fixed topology optical highway. Thus complex reconfiguration control hardware can be eliminated, including potential run time issues such as determination of the entire optical highway's current state.

Given components similar to those already constructed and in use by the SCIOS project in this research group [101], we can extrapolate the potential bandwidth of such a system. Considering that 2,500 MQW based optical channels off-chip are feasible at data rates of 250MHz each, a two node system has a bisection bandwidth in excess of $1Tbs^{-1}$. Depending on topology [99], the potential bisection bandwidth is therefore far in excess of any existing electronic architecture.

3.8 Conclusion

This chapter has examined the current technologies that enable optical interconnection and detailed the design issues involved in optical signal detection. It has outlined why the interface between optics and electronics is constructed using hybrid chip technologies which employ a direct bandgap optical substrate hosting emitters and detectors flip-chip bonded onto an Si substrate hosting detectors and processing logic. Systems such as this are referred to as *smart pixel arrays* (SPAs) and are defined as an optoelectronic device that may have memory, intra-pixel processing, inter-pixel communication and an optical input or output element.

The process technologies that enable smart pixels are beginning to mature in their own right. However, direct integration and exploitation of their full bandwidth potential is still some way off. This has previously been due to fabrication difficulties, however alternative methods of integration are beginning to emerge that can reduce integration complexity and improve yield. Viability and reliability of such systems has recently been clearly demonstrated by the company Terraconnect [102]. In 2001, it had just reached prototype stage with a single general purpose optical interconnect module that integrates 320 oxide confined VCSELs in a 16×20 array, 320 VCSEL driver amplifiers, 320 GaAs p-i-n photodiodes, 320 transimpedance amplifiers and all the coding and switching logic to ensure error free data transmission. At the time of writing, this system had been running continuously for just over eight weeks without so much as a

single bit error. This is further proof that not only are these systems feasible but they are becoming a reality.

4 Neural Networks for Switching

This chapter examines the concepts behind parallel systems and introduces a type of highly parallel architecture called the artificial neural network. It is shown that a neural network can be adapted to solve the assignment problem and that its very nature brings scalability to an otherwise impractical problem.

4.1 Parallel Systems

To sustain the rates of growth predicted by Moore [25] there is a need to develop new computational techniques since traditional *sequential* computers are rapidly approaching their physical performance limit. When this limit is reached, the only way forward will be to implement *parallel* architectures. The parallelism of an architecture is normally classified into one of four categories as defined by *Flynn's taxonomy* [103]:

- *Single instruction* stream, *single data* stream (SISD). This model represents a basic uniprocessor system. These systems do not support true parallelism but can emulate it if they support multitasking.
- *Single instruction* stream, *multiple data* streams (SIMD). A lot of scientific applications such as image processing or particle system simulations have an identical sequence of commands to be carried out on more than one data stream. Such a system would therefore execute one command on multiple data streams in lock-step as seen in vector and array processors.
- *Multiple instruction* streams, *single data* stream (MISD). This is the hardest type of system to envisage but involves one data stream which needs to be operated upon by more than one set of instructions. An example of such a system would be a pipeline or systolic array where one processor performs a set of operations on the data stream before passing it on to another processor which performs a different set of operations.
- *Multiple instruction* streams, *multiple data* streams (MIMD). This is the most powerful form of parallel processing system where multiple processors work on different data streams simultaneously.

These categories describe an increasing level of parallelism starting from none in SIMD architectures to highly parallel MIMD architectures. If we assume that any speedup is

linear, a MIMD system with two processors should be able to perform the same amount of operations in half the time that a similarly configured SIMD system would take. Unfortunately this is rarely the case as a number of fundamental limitations exist.

The most significant issue in parallel processing systems is computation sequentiality. Consider a parallel system executing an algorithm where no other processors can proceed until one specific calculation has been performed. This means that all other processors must remain idle until that particular processor is finished, effectively reducing optimum speedup. In 1967 *Amdahl* described a law [104] which states that the maximum speedup S possible is related to the number of processors p in a system and the fraction of the computation f that must be performed sequentially:

$$\frac{1}{f + \frac{1-f}{p}} \geq S \qquad \textbf{Equation 34}$$

To achieve linear speedup there can be no sequential element at all in the computation, with f=0. Such a case is not common in the real world, with the overall performance of a parallel system rarely equalling the combined computational power of all its processors.

Communication overheads can exacerbate sequentiality issues and are a matter that must be taken seriously in multi-processor architectures. If overheads are large, it may take more time to divide data and code amongst multiple processors and return it to the originating processor than it would to process the computation on a single processor in the first place. Although this case implies a small portion of parallelism before the computation returns to being sequential, consider the overheads in a system with thousands of processors. This results in what is commonly known as a *mortar shot* graph as shown in Figure 45. These graphs show a gradual, but tailing off, speedup as more processors are applied to a computation. Eventually there are so many processors working on the same computation that communications overheads become restrictive resulting in a reduction of overall system performance. Such behaviour is noticeable even with a large proportion of computation parallelism. This suggests that there is an optimal number of processors for a parallel system, however any figure is dependent on the fraction of sequentiality of the computation and inter-processor communication delay. Therefore most parallel systems are constructed as general purpose machines which require software optimisation to ensure best use of resources.

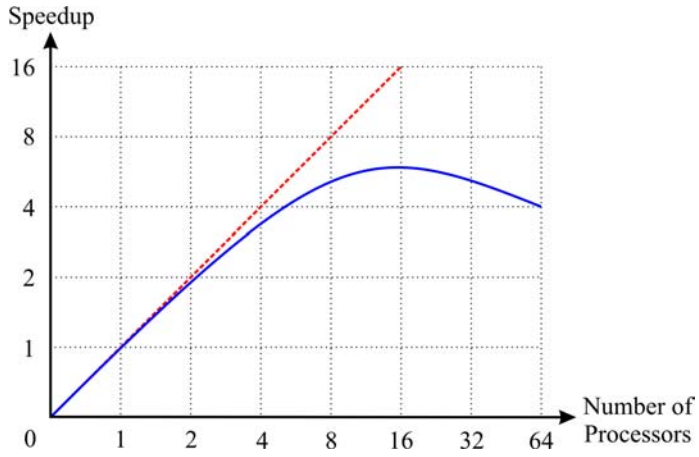

Figure 45: Mortar Shot Graph

Communications overheads can result in an increase in computation times if more processors are applied to a computation as shown by the solid blue line. The dashed red line indicates optimal speedup presuming there is no sequential component to the computation and communications overheads are negligible.

Increasing the overall performance of a parallel system can be achieved by boosting the performance of each processor, by adding to the total number of processors or by reducing communications overheads. If the performance of each processor is increased there is a reduction in sequentiality issues. Alternatively, additional processors provide a performance increase only if the computation is massively parallel in nature and communications overheads are low. Figure 46 shows the different approaches taken in an attempt to reach *El Dorado* [105]. Note that operating systems can balance the load on a parallel system by releasing idle processors to other processes and computations.

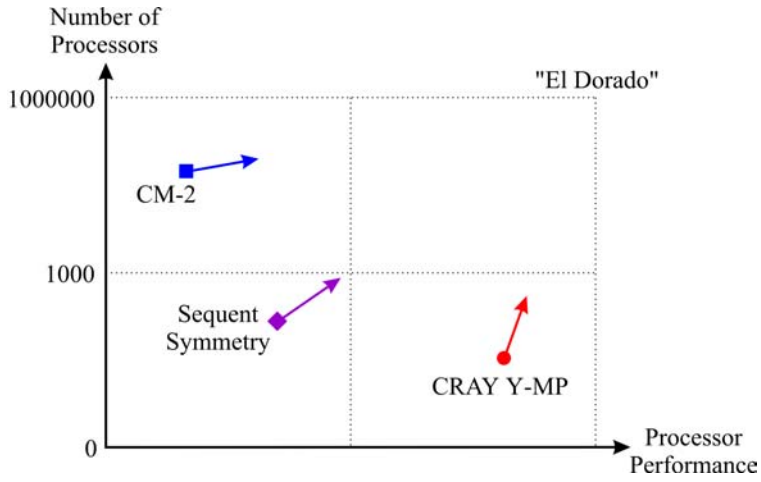

Figure 46: Performance versus Parallelism

Different systems trade processor performance against number of processors in an attempt to reach computational El Dorado.

It seems that no matter how hard we try all odds are stacked against a parallel computer system that achieves a linear performance increase with each additional processor. Doubling the number of processors in such a system should logically give, at best, twice the performance. This had been the view until a particular exception was discovered called *superlinear acceleration* [22]. It was first noticed on the highly parallel *Ocean* algorithm which simulates the flow of eddies and currents in the ocean. Addition of another processor to a single processor system slightly more than doubled computational throughput. The cause was determined to be that the processors shared their caches and therefore had effectively double the normal amount of cache available. This is an advocate, if a rather poor one, for parallel systems but there are still too few instances in which the perfect situation arises to make parallel systems unarguably advantageous.

The ultimate goal of parallel processing is to have a single unit which, without the aid of additional communications logic, can be attached seamlessly to another doubling available processing power. The user should not have to consider issues such as memory location or addressing, topology or communications efficiency but simply add another processor to enhance performance. Unfortunately, parallel implementations still tend to be application specific. The creation of a universal parallel system is plagued by architectural trade-offs. One promising architecture was the INMOS Transputer [106], however it was released with no software base and few potential customers wanted to start writing compilers. It did not take long before faster serial processors were released effectively putting an end to the Transputer before it even got started.

The use of optical interconnects can alleviate communications overheads in the short term but cannot solve sequentiality issues within a computation. Nevertheless, parallel computer systems are becoming ever more established as a viable alternative to sequential systems and indeed all of today's supercomputers are parallel machines [107]. However, there are hypotheses which suggest that computers are evolving in the wrong direction [108]. Although supercomputers exist that can perform in excess of 10^{13} floating point operations per second [109] they lack both a degree of intuition, which would allow them to rapidly extrapolate important information from raw data, and flexibility, which would allow them to adapt to a specific type of problem. These issues can be addressed by neural networks and field programmable gate array (FPGA) technologies respectively, both of which will be examined in this and later chapters.

4.2 Artificial Neural Networks

Artificial neural networks (ANNs) [110]-[113] were designed to be an analogy of the human brain but they are still a considerable way from rivalling its size and complexity. In computer science terms, the human brain is modelled as a highly interconnected MIMD architecture that contains around 10^{11} simple processing elements called *neurons*. It is the extreme interconnectivity within neural networks that is their attraction since noise, system errors or neuron failure are normally inconsequential. An entire calculation in a sequential system can be ruined by a single bit error whereas a neural system degrades gracefully. This fault tolerance creates a robust system and is the major advantage of artificial neural networks.

Even though the cycle times of silicon computer systems are 10^6 times faster than that of the human brain, the stochastic techniques used by computers are inefficient at performing tasks such as image recognition. Neural networks on the other hand have proven themselves at recognising patterns or trends in seemingly random data. Combining the speed of silicon and the abilities of neural architectures may one day lead to an evolution in computer systems.

4.2.1 The Artificial Neuron

A neuron is the basic building block of *neural networks* (NN) and can be seen in Figure 47. The artificial neuron approximates the behaviour of a neuron in nature.

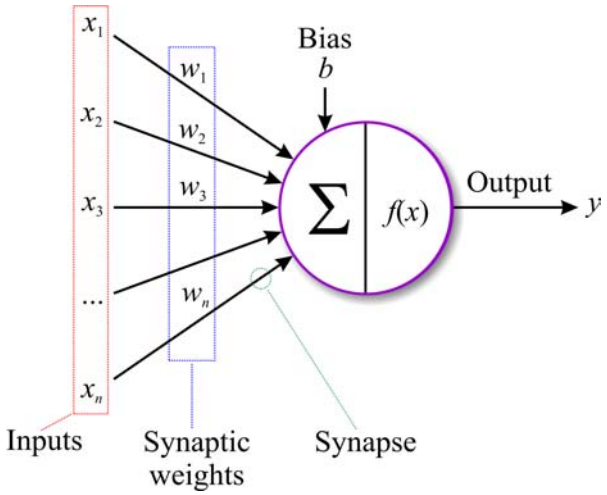

Figure 47: The Neuron

A neuron sums all incoming values x_1-x_n after multiplying them by an appropriate weight w_1-w_n , adds a bias b and finally determines its response based on the activation function $f(x)$.

A neuron takes n inputs x_1-x_n, multiplies their strength by a scalar weight w_1-w_n known as the synaptic weight, sums the products and then adds a bias b:

$$x = \sum_{k=1}^{n} x_k w_k + b \qquad \textbf{Equation 35}$$

Synaptic weights allow certain inputs to have greater significance than others. The transfer function $f(x)$ is then applied to the summation giving a final neuron activation level y. Transfer functions are neuron dependent and vary with application. Figure 48 shows some common transfer functions.

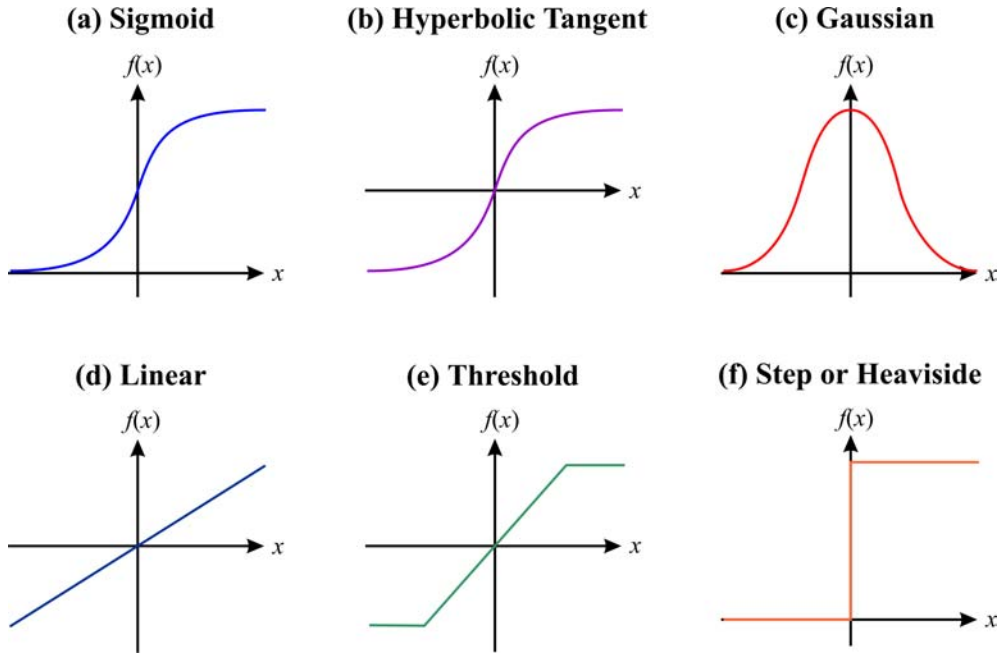

Figure 48: Sample Transfer Functions
Neuron transfer functions $f(x)$ determine the neuron's activation level y depending on the value of x.

One of the most commonly used transfer functions is the *sigmoid* function:

$$f(x) = \frac{1}{1 + e^{-\beta x}} \qquad \textbf{Equation 36}$$

where β determines the function's slope. The advantage of the sigmoid function is that it has smooth transition limits, unlike functions such as the *step* or *heaviside*, yet will converge asymptotically to a decision as the magnitude of x increases, unlike *linear* functions. Note that some applications use the *hyperbolic tangent* whose output is similar to that of the sigmoid function except that its value $f(x)$ ranges from -1 to $+1$ whereas the sigmoid function range is between 0 and +1.

The neuron is the basic building block of neural networks and by combining them in a specific manner, altering their transfer functions, input weights and biases, a variety of applications can be addressed, the computation of which proves prohibitive on conventional architectures.

4.2.2 Neural Network Types

There are many types of neural network with varying characteristics that suit them to specific applications. Table 3 shows some common networks specifying both the input taken and method used to train them.

Neural Network Name	Input Type	Training Method
Hopfield Net	Binary	Supervised
Hamming Net	Binary	Supervised
Carpenter/Grossberg Classifier	Binary	Unsupervised
Perceptron	Continuous	Supervised
Multi-Layer Perceptron	Continuous	Supervised
Kohonen Self-Organising Feature Maps	Continuous	Unsupervised

Table 3: Sample Neural Networks
Common neural networks including typical input type and training method used.

Input type defines whether binary or continuous valued input is normally taken. The output type is dependent on transfer function used. These networks can again be sub-classified dependent on procedure used to train them. *Unsupervised learning* or *self organisation* is where the neurons compete for data with no desired response applied since there is no particular target set. Such networks discover salient statistical features in information by themselves and are therefore suited to exploratory work. *Supervised* or *associative learning* is where there is a specific response that must be achieved. Networks trained in this manner are suited to modelling.

4.2.3 The Perceptron

This section examines a *feed forward* neural network called the *perceptron* in order to demonstrate the relationship between a neural network's size and the problem complexities that it is capable of solving. Rosenblatt [114] invented a variation on a simple neural network, which he called the perceptron, that is especially suited to

elementary pattern classification problems. Figure 49 shows three examples of perceptron networks each with differing levels of complexity.

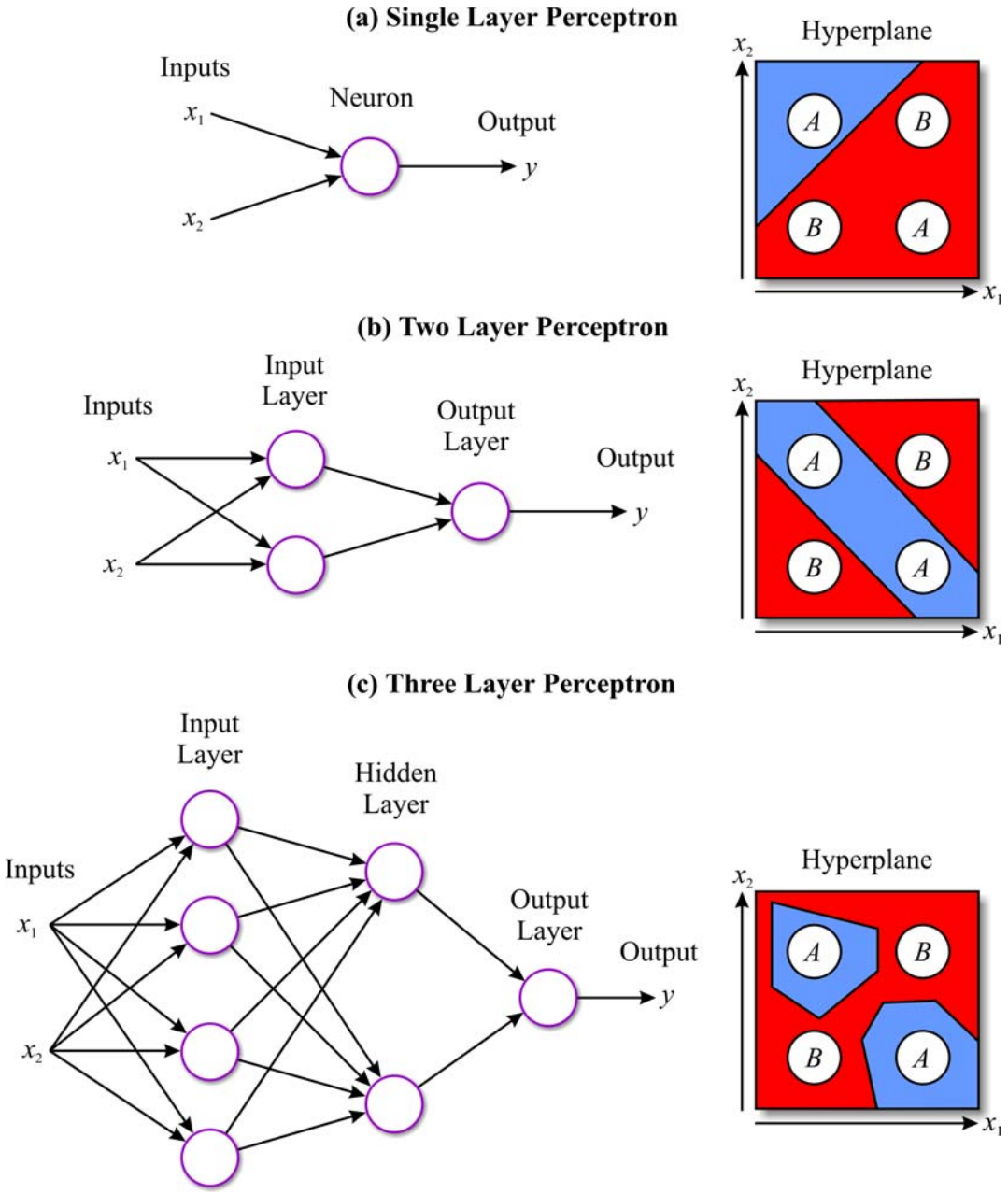

Figure 49: Perceptron Networks

Perceptron networks can be used to solve the XOR problem. The network in part (a) has insufficient complexity to represent the problem. Additional neurons and layers in part (b) allow the problem to be modelled correctly. Further neurons and layers in part (c) correctly represent the problem but with unnecessary complexity and detail. This illustrates that neural network size can be optimised to a specific application.

The *exclusive or* (XOR) [115] problem takes two inputs x_1 and x_2 and classifies them into either category *A* or *B* indicated by the value of y. This is considered to be the third dimension when graphed as a *hyperplane*. All three hyperplanes in Figure 49 indicate

the actual output from *y* using contrastingly shaded regions, the numeric value of which is irrelevant. Target classifications for *A* and *B* in particular regions are also shown here and, presuming that the network classifies successfully, both *A* targets must lie within the same shade of region whereas both *B* targets should be in the oppositely shaded region.

The simplest type of perceptron is shown in Figure 49(a) and consists of a single layer with one neuron. Such networks are only capable of classifying information using a single straight hyperplane line. This is not sufficient to successfully implement the XOR function with the neuron responding correctly to one occurrence of *A* but not the other.

The next increment in complexity is a two layer perceptron as shown in Figure 49(b). This network is capable of implementing the XOR function as the network reproduces more than a single hyperplane line. It can therefore model open and closed convex regions.

Perceptrons with more than two layers are referred to as *multi-layer perceptrons* (MLP) as shown in Figure 49(c). They differ from other perceptron networks in that they have *hidden layers*, the output of which is not externally observable. Each additional layer enables an extra, distinct and non-linear classification line thus allowing the creation of decision lines with almost any shape. This network can implement the XOR function, however the hyperplane reveals unnecessary complexity. The advantage of MLPs is their ability to model training sets that are not linearly separable.

Perceptron networks have two modes of operation: *learning* and *recall*. Learning mode trains the network to simulate appropriate data by altering each neuron's input weights until the system is modelled to a set of predefined limits called the *target set*. Recall mode applies a set of values which the perceptron will attempt to classify based on what it has previously learned.

4.2.4 Summary

Neural networks may exhibit a number of strengths when compared to conventional computational methods:

- They can be set up to learn through experience from the input data itself.
- Applications include classification, noise reduction or prediction problems.
- They can make a conclusion even though input data is not well defined.

- Patterns can be extracted even if differences are subtle.
- Decisions can be made even when data is chaotic by mathematical standards.

Neurocomputing is an immense theoretical field. This section has examined some neural network basics and complexity issues through the introduction of a common neural network type. Even though large scale neural computers do not yet exist their possible applications remain diverse from pattern recognition in both spatial and time domains to the implementation of digital logic with a degree of fault tolerance.

4.3 Solving the Assignment Problem

Current hardware and software systems suffer from an exponential increase in computational complexity when solving the *assignment problem*. This section considers the problem and proceeds to propose a solution using the inherent parallelism of a neural network to reduce computation times. Problem mapping is examined for both crossbar and Banyan packet switch architectures.

4.3.1 The Assignment Problem

As the complexity of modern communications and computational systems increases so does the need to develop techniques that deal with common assignment problems [116]-[117] in situations such as:

- Network and service management.
- Distributed computer systems.
- Work Management systems.
- General scheduling, control or resource allocation problems.

The common assignment problem is essentially optimising task allocation to all available resources thus maximising throughput. In a distributed computer system this results in a many-process computation being finished in the shortest possible time, whereas in a network management system packets are routed to optimise switch throughput and minimise blocking.

4.3.2 Neural Network Implementation

This section presents a neural network algorithm that solves the assignment problem in both crossbar and Banyan switches for packet routing [118], a problem known to be analogous to the *travelling salesman problem* (TSP). The TSP problem is a renowned

NP complete problem [119] which means that although it can be solved by integer programming techniques, such as the *Murnkes* algorithm [120], it is computationally intensive and complexity grows exponentially as its order increases. Thus, a simple single processor solution will not provide satisfactory scalability.

One alternative is to apply a neural network to the TSP problem [121]-[122]. The advantage of a neural net lies in the speed obtained through its inherent parallel operation, especially when dealing with large problem sets. Such implementations will easily outperform any other method at higher orders of network size [116], [123]-[128] providing a very good, but not always optimal, solution. It has been shown [116] that, at lower orders of network size, the average solution is within 3% of optimal. However, as the network size grows this figure improves slowly and begins to approach an optimal solution.

The crossbar switch controller described here arranges neurons in a two dimensional array representing all possible input to output connections such that each neuron corresponds directly to a crosspoint on the switch as seen in Figure 50. In order to choose a set of connections, the neurons representing all the requested connections are enabled simultaneously and set to the same intermediate level. Each has a bias that tends to increase its output, but also receives *inhibitory* inputs from those neurons which represent blocking connections. Crossbar switches can be blocked at their inputs and outputs only, so the neurons are arranged to be inhibited by others in the same row or column. All other possible connections are set to zero. The dynamics of the network resolve the conflicts between all the mutually excluded neuron pairs, leaving a valid set of neurons in the on state and the remainder off. The network is thus behaving as a *winner take all* (WTA) system with a particularly simple interconnect pattern - each neuron sees only its row and column neighbours, each of which are connected to it by a fixed, inhibitory weight. It is apparent that such a pattern is space invariant, in that it remains fixed regardless of where in space it is formed, and therefore highly suitable for implementation in a diffractive optical system.

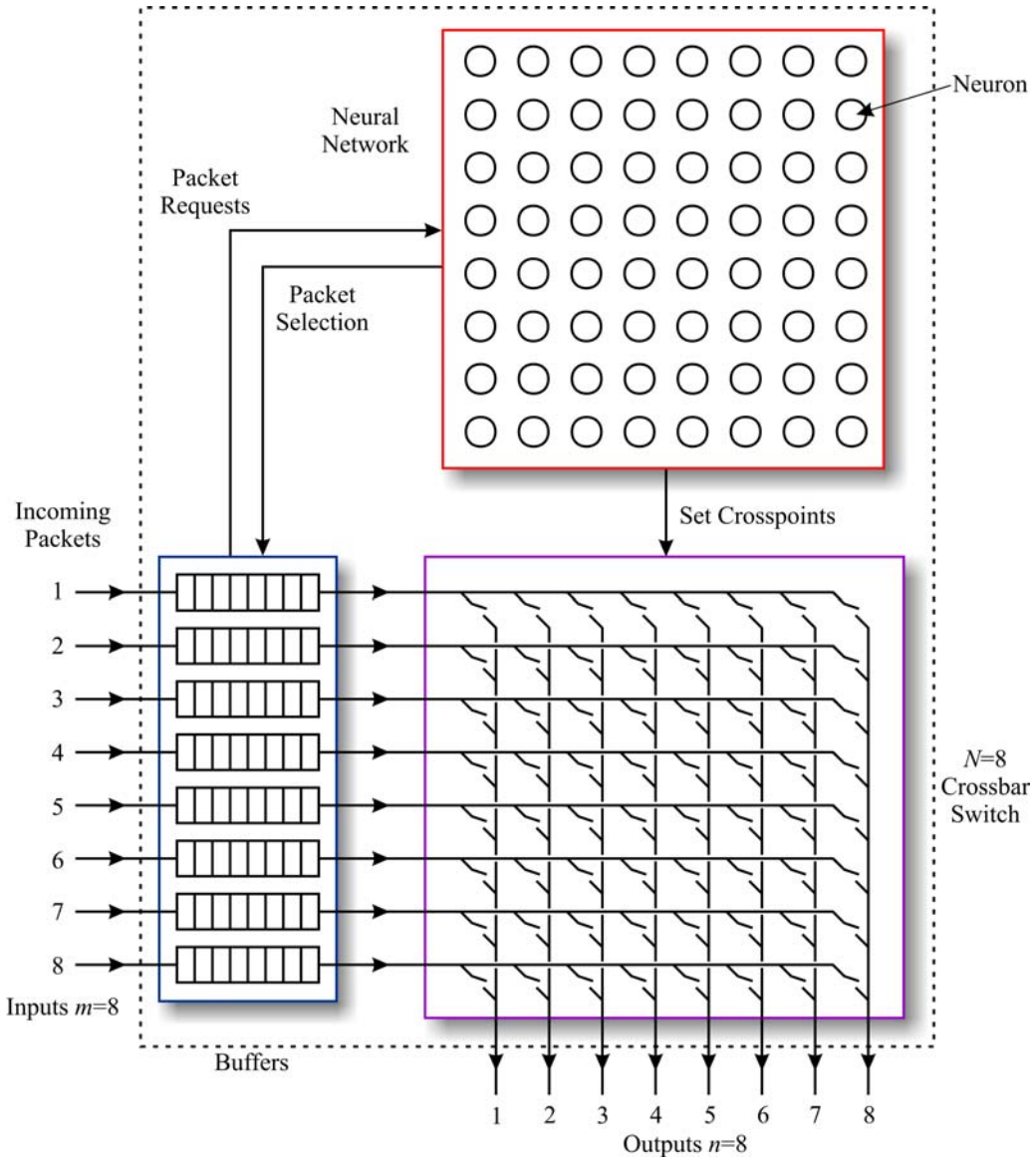

Figure 50: Neural Network Crossbar Switch Controller

Based on the connections requested by incoming packets, the neural network chooses an optimal solution, sets the appropriate crossbar switches and then selects the chosen packets.

It is worth taking into consideration how other types of switch might be controlled in this way, particularly with regard to exploiting shift invariant interconnection patterns. Consider the scheduler shown in Figure 51. In this case the incoming packets have a header address which determines their path through the Banyan network. The penalty is of course that the Banyan type of switch shown here is internally blocking and a more complex task must be performed by the scheduler. The neural network scheduler copes well with this added complexity in that the correct functionality may be attained by merely providing additional inhibitory paths to provide contention between requests for

these blocking configurations. How this may be done can be seen by considering the grid of neurons with the left hand vertical edge corresponding to the input port of a packet and the lower horizontal edge corresponding to an output port. Since the destination and input positions are known for any packet, the internally blocking connections are clear and may be explicitly programmed in advance [129]. The resultant pattern will perform a WTA optimisation and allow the switch to operate.

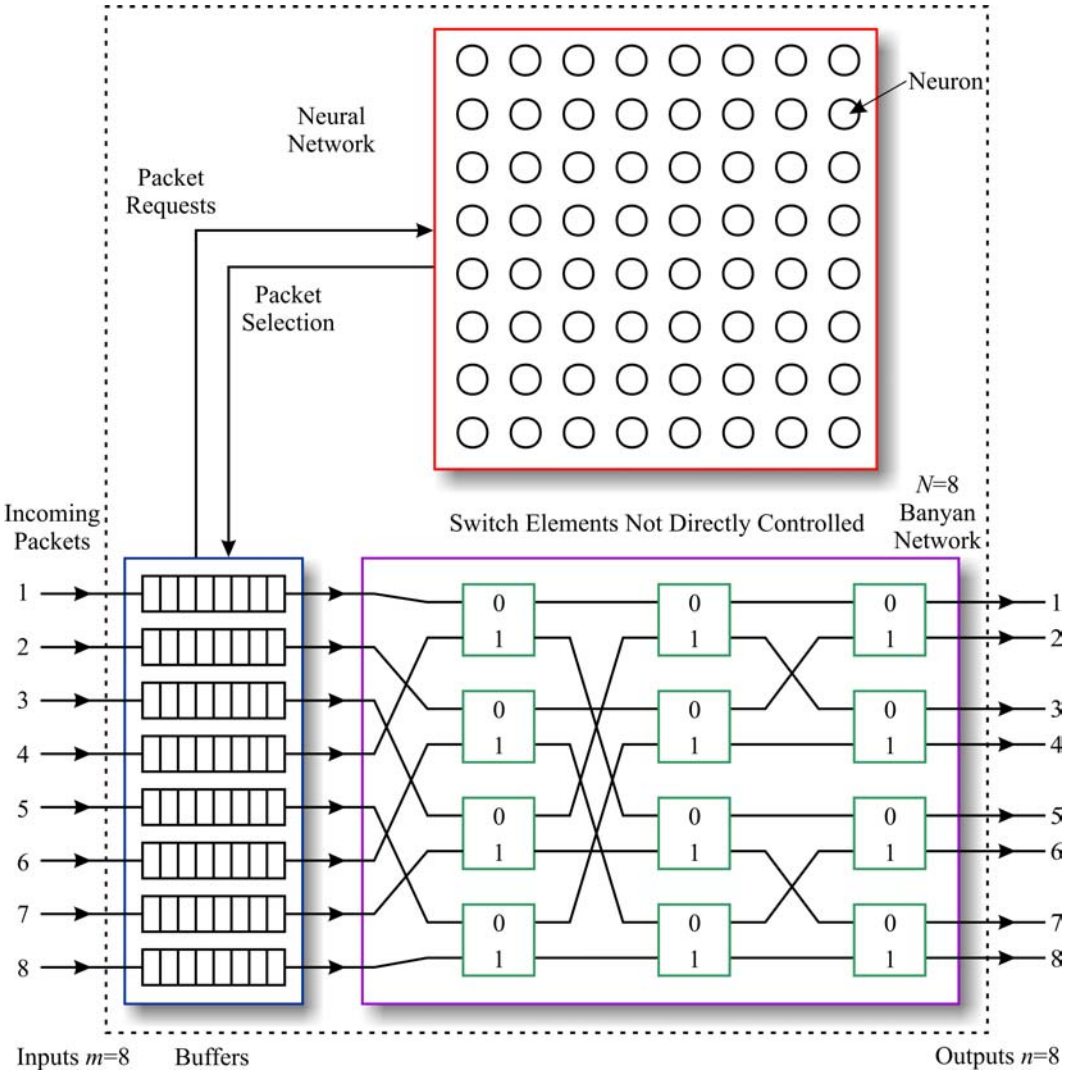

Figure 51: Neural Network Controller for a Self Routing Multistage Banyan Switch
Here the neural network selects an optimal packet solution and notifies the input buffers. It does not directly control the switching network.

A novel feature in this work is making such patterns amenable to optics. What is most important from the perspective of a diffractive optical implementation is that the final pattern of optics that is generated is space invariant. The interpretation of one of the edges of the neuron grid in bit-reversed addressing order makes the inhibitory interconnection pattern once more space invariant and consequently suitable.

4.4 Neural Network Algorithm

The key to utilising the parallelism of a neural network is matching the network as closely as possible to the problem. This section examines the algorithm used by our system in detail with specific reference to the crossbar implementation. Adjusting the algorithm to the Banyan network simply requires a different interconnect pattern.

4.4.1 Crossbar Switch Notation

A crossbar switch can be abstracted as a set of *m* inputs and *n* outputs where each input can be switched to any output by closing the correct crosspoint switch. Figure 52 details how a matrix may be mapped onto the crossbar switch, each crosspoint having a corresponding matrix element.

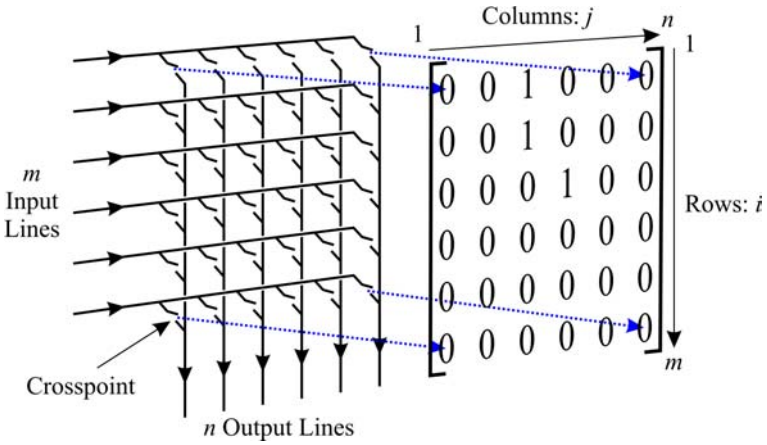

Figure 52: Matrix Representation of Crossbar Switch
A matrix can be used to represent the crossbar switch mathematically. Each matrix element represents a neuron. Neurons can take values that vary continuously between the on and off levels of one and zero respectively.

A specific element *y* in any matrix can therefore be referenced using y_{ij}, where *i* is the input line and *j* the output line. Every element in the matrix can take on a continuous range of values. Normally a value of 1 is used to represent a *connection* or a *connection request* and 0 for off. These values and their legality is dependent on situation. Equations 37 and 38 both represent the crossbar switch in Figure 52 but are taken from different points of view. Equation 37 represents a set of desired connections where three input lines have *requested* connection to two different output lines. Such a matrix is legal regardless of the matrix element values. Here, input *i*=1 has requested connection to output *j*=3 and both inputs *i*=2 and *i*=3 have requested connection to output *j*=4. One request is obviously going to have to wait.

$$y = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$ **Equation 37**

Equation 38 on the other hand shows a sample neural network *response* to Equation 37. One request has been discarded in favour of another since only one input line can be connected to one output line at any time. A response is considered legal if no other connections on the same input row or output column has been selected.

$$y = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$ **Equation 38**

Real optimisation problems arise when a system with buffered input is considered. In such systems there can be multiple packets waiting on a single input line for various output lines as shown in Equation 39. Requests for multiple connections can be seen in the left matrix and the unique optimal solution which maximises throughput on the right. This request matrix proves useful for testing crossbar systems.

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$ **Equation 39**

If the request values are made continuous it becomes possible to implement prioritisation strategies. For example, if the range of request values lay between 0 and 2 then a requested connection would be half active with a value of 1 and fully active with a value of 2. Thus a packet whose weight was 2 would have an advantage over a packet whose weight was 1. Indeed, request matrix values could be, for instance, between 0 and 10 to represent the number of packets waiting for each connection. As the number of packets waiting increased, so would the importance of the connection.

4.4.2 Neuron Interconnection

The neural network described here has been theoretically derived from the *Hopfield* neural network [121], [130]-[131] but behaves similarly to a winner take all network. It is a recurrent network in that it evolves to a stable state where neuron activation levels do not change any more. Adapting the neural network to the assignment problem required redefinition of the *updating rule* and thereby the network interconnection structure. Updating rules determine the next value that a neuron will take, with respect to time, based upon the previous outputs of other neurons. Figure 53 shows the required interconnect structure graphically.

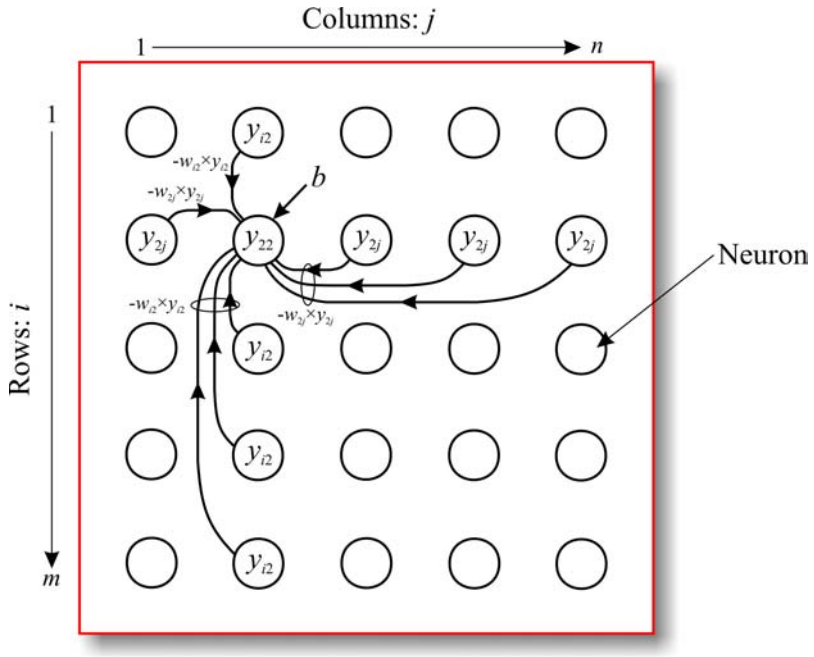

Figure 53: Neural Network Interconnection
Neural network mapped to a 5×5 element crossbar switch. Neuron y_{22} receives inhibitory input from all other neurons in the same row and column.

We can therefore define the network's updating rule:

$$x_{ij}(t) = i_{ij}\left(x_{ij}(t-1) + \lambda_{ij}\left(-A\sum_{k\neq i}^{m} w_{kj}\, y_{kj} - B\sum_{k\neq j}^{n} w_{ik}\, y_{ik} + b \right) \right) \qquad \textbf{Equation 40}$$

where:

- x_{ij} is the summation of all inputs to the neuron referenced by ij including the bias b.
- i_{ij} determines whether a neuron referenced by ij is allowed to evolve.
- λ_{ij} is the time constant for the neuron referenced by ij.
- w_{ij} is the synaptic weight for neuron input ij.

- y_{ij} is the output from a neuron referenced by *ij*.
- *A* is the neuron optimisation value weighting input from any element in the same column.
- *B* is the neuron optimisation value weighting input from any element in the same row.
- *b* is the optimisation value representing an external bias supplied to each neuron.

All inputs are summed by the neuron along with addition of the bias *b* to find x_{ij}. The neuron's output y_{ij} can then be determined using a monotonic activation function $f(x_{ij})$:

$$y_{ij} = f(x_{ij}) = o_{\min} + \frac{o_{\max} - o_{\min}}{1 + e^{-\beta x_{ij}}} \qquad \textbf{Equation 41}$$

Here β is used to control the gain of the sigmoid function, a higher value resulting in a steeper transition, and $o_{\min}$ and $o_{\max}$ to determine the minimum and maximum output values for y_{ij} respectively. The exact form of $f(x_{ij})$ is not particularly important, indeed any appropriate non-linear monotonically increasing function could be used. The preferred embodiment is however the sigmoid function.

To illustrate the operation of this updating rule we can use Figure 53. Here the neuron marked with output y_{22} has inputs from all the other neurons in the same row $-w_{2j} \times y_{2j}$ and column $-w_{i2} \times y_{i2}$. The important point to note here is that the neural network operates in an inhibitory fashion thus any active input will inhibit y_{22}. Note that the external bias *b* supplied to each neuron is *excitatory*.

The idea behind this interconnection strategy is that any active neuron will try to turn all the others off, eventually resulting in only one of the requests remaining active in each row and column. To demonstrate its ability to find an optimal solution, the example in Figure 53 needs to be extended slightly as shown in Equation 42:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad \textbf{Equation 42}$$

The left matrix here represents a set of requested connections and the right its best case solution with y_{22} switched off. Careful consideration leads us to conclude that the network must converge to the solution shown since both y_{24} and y_{42} are inhibiting y_{22}

resulting in it being switched off before the others and essentially losing. If y_{22} had won then it would have resulted in a poor solution since y_{24} and y_{42} would be inactive therefore not maximising potential throughput.

It has been shown by Hopfield that with symmetric connections and a monotonically increasing activation function $f(x)$, the dynamical system described by a neural network possesses a *Lyapunov* energy function [115] which continually decreases with time. The existence of such a function guarantees that the system converges towards equilibrium, often referred to as a *point attractor*. In any system with a continually reducing energy function there is always a risk that the system will become trapped in *local minima*. In this system, local minima can be represented as a solution which satisfies the switching constraints but is not a global optimal solution. This problem can be avoided by introducing noise and was done in initial mathematical modelling by perturbing β. This alteration of the activation curve's gradient is significant enough to provide successful convergence to a global minimum. However, the technique has proven necessary only in simulation as the optical and electronic systems used in both demonstrators provides sufficient intrinsic noise to ensure convergence to *global minima*.

4.4.3 Determination of Optimisation Parameters

The optimisation parameters A, B and b [132] have been determined purely by trial and error in previous work [133]. If these parameters are not chosen carefully then Equation 40 will converge either slowly or not at all. Indeed the system may even converge to an invalid solution.

It is possible to determine limits for optimisation parameters methodically before using trial and error. First, consider a solution for Equation 40 when the system is in equilibrium:

$$\frac{dx_{ij}}{dt} = 0 \qquad \textbf{Equation 43}$$

This results in the equation:

$$x_{0,ij} = -A\sum_{k\neq i}^{m} f\left(x_{0,kj}\right) - B\sum_{k\neq j}^{n} f\left(x_{0,ik}\right) + b \qquad \textbf{Equation 44}$$

where $x_{0,ij}$ is the value x_{ij} at equilibrium.

Further restricting the parameters, we know that in the final solution to the switching problem each neuron will settle to either zero or one. Presuming that a valid solution has been found then there should be at most one active neuron for each row and column. This information allows us to establish that if *ij* is a zero position then the equilibrium condition becomes:

$$x_1 = -A - B + b \qquad \textbf{Equation 45}$$

where x_1 denotes the first equilibrium solution. However, we also know that since we are at equilibrium the associated *y* value must be close to zero and that *y* tends towards zero as *x* tends towards minus infinity. Accordingly, we can rewrite Equation 45 as an inequality:

$$-A - B + b << 0 \qquad \textbf{Equation 46}$$

This solution is referred to as the *negative attractor*. There must be N^2-N positions in the network satisfying this condition, presuming a square matrix of N^2.

The next consideration must be the *ij* positions which tend towards one. Under equilibrium, the condition then becomes:

$$x_2 = b \qquad \textbf{Equation 47}$$

where x_2 represents the second equilibrium solution. Examining Equation 41 it can be seen that *y* tends to one as *x* tends to infinity. This allows us to rewrite the second equilibrium solution as an inequality:

$$b >> 0 \qquad \textbf{Equation 48}$$

This is a *positive attractor* and has to be satisfied at *N* positions in the network.

The final equilibrium conditions mean that *N* neurons in the network have converged to one of the two attractors and N^2-N neurons to the other. Combining Equations 46 and 48 gives the overall inequality:

$$0 < b < A + B \qquad \textbf{Equation 49}$$

This inequality can be refined since a symmetric matrix is desired where $A=B$:

$$0 < b < 2A \qquad \textbf{Equation 50}$$

This information has been used to simulate the perfect theoretical case in Matlab [139] enabling the determination of the significance of each optimisation parameter including the neuron's activation function. Analysis of the model indicated that:

- The value of β should lie within the region 0.08 to 0.16 for optimal performance. β is effectively linked to *b* in the following manner:

$$\beta b \approx 1 \qquad \textbf{Equation 51}$$

- The bias b should remain within the limits of 20 to 75 for optimal operation. Increasing the value of b encourages the neurons to choose quickly leading to sub-optimal solutions.
- Presuming that the network operates using a symmetric matrix, A should be at least ten times greater than b.

A set of preferred values were discovered during simulation with A=1,250 and b=50. We can therefore use Equation 51 to calculate that β=0.02. This value lies slightly out with the stated optimal range but does not prevent or hinder correct network operation. The longer decision times associated with lower values of b are justified due to improvements in quality of solution.

4.4.4 Conclusion

Simulation of the neural network proves that the algorithm can be mapped successfully on to a packet switch scheduler. It has also underlined two important points:

- Noise plays a significant role in the model. As the noise level increases, the time taken for network stabilisation decreases. Calculations indicate that an experimental implementation will receive enough background noise to stimulate convergence. However, when noise reaches unity the network becomes unstable and does not provide a valid or steady solution.
- Network size plays an important role in convergence. The larger it is the longer it takes to converge.

What makes this system so interesting is its diversity with packet switch scheduling only one of many applications.

4.5 Neural Network Hardware

The neural network studied here is unique in that it contains optoelectronic components to provide the high degree of connectivity required and uses algorithms specially designed to exploit this additional connectivity. This section examines the hardware used to create both first and second generation demonstrators discussing the limitations that directly influence network operation. Abstracted hardware descriptions of both optical and electronic systems enable an understanding of further simulations examined later in the chapter.

4.5.1 Optical System

In the optical domain, a DOE provides fixed and evenly weighted interconnection between the neurons in the system, as is required by the algorithm. Increasing the size of the neural network requires an increase in DOE fan-out. Increasing DOE fan-out decreases the signal incident on a detector with a consequent decrease in SNR. Achievable network size is therefore tightly bound to DOE fan-out. The pattern of neurons inhibited by any given active neuron, two of which are shown in Figure 54, is shift invariant. That is, the pattern remains the same relative to the position of the active neuron. An electrical system would require a separate wiring network for each output leading to a quadratic increase in routing complexity as order increases. Such intractability hinders the construction of neural networks as an *application specific integrated circuit* (ASIC). Indeed, replacing the DOE with either a different DOE or a reconfigurable SLM allows the construction of many types of neural network.

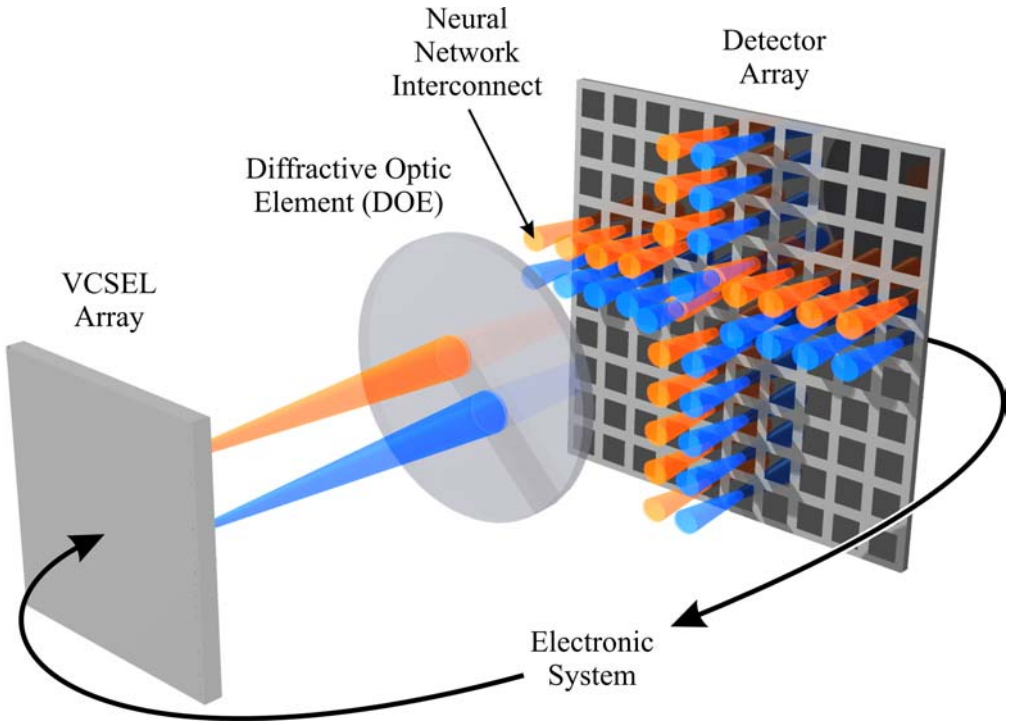

Figure 54: Neural Network Optical System
Scalar domain DOE fans out inhibitory signal providing a shift invariant interconnect pattern as defined in section 4.4.2. Neuron input summation is performed by measuring the total optical power incident on a detector.

The far-field intensity profile generated by the DOE can be quantified using two parameters: the *diffraction efficiency* (T_{DOE}) and the *non-uniformity* or *reconstruction error* (Δr). The diffraction efficiency is the proportion of the incident light intensity

diffracted into the required orders and is typically around 60 to 70%. The reconstruction error is a measure of the fidelity of the measured far-field intensity profile to the desired far-field intensity profile. It is defined as:

$$\Delta r = \max_{i,j \in M} \left| 1 - \frac{100 S_{DOE} P_{ij}}{T_{DOE}} \right| \qquad \textbf{Equation 52}$$

where M is the desired set of diffraction orders, S_{DOE} is the number of orders in M, P_{ij} is the intensity in the ij^{th} diffraction orders and T_{DOE} is the diffraction efficiency as a percentage. The value of Δr is dependent upon both the non-uniformity inherent in the design process and the additional non-uniformity added during fabrication. The inherent reconstruction error is due to the non-bandwidth limited nature of the desired far-field intensity profile as well as the error introduced by quantisation of the phase profile [134]. Typically, this unavoidable error will be of the order of 0.1% provided that there is a sufficient space bandwidth product to specify the far-field intensity profile with reasonable accuracy [90]. The fabrication reconstruction error is introduced by a number of different mechanisms. These are feature rounding and elimination during photolithographic transfer as shown in Figure 55, layer misalignment for multilevel phase profiles, etch depth inaccuracy and etch non-uniformity across the element.

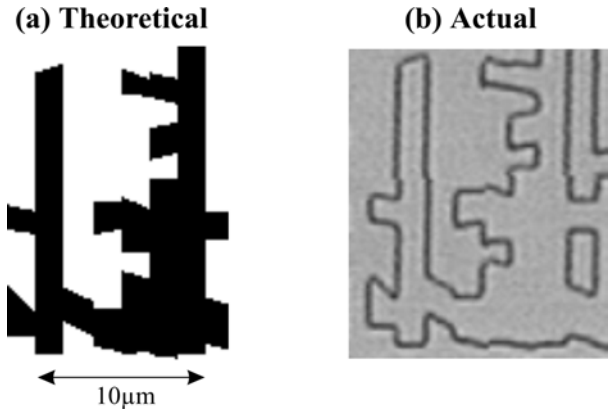

Figure 55: Photolithographic Transfer Error
Features are rounded or even eliminated during transfer from the desired photolithographic pattern in (a) to actual DOE element in (b).

The inter-layer alignment accuracy is of the order of 0.1µm with the etch depth inaccuracy being typically less than 1%. Etch non-uniformity across the element is the least significant of all fabrication non-uniformity mechanisms and can generally be ignored for the scalar domain elements considered. The overall effect of different non-uniformity mechanisms is an additional reconstruction error of between 1% and 2% per mask level. This means that a 16 phase level element which has 4 mask layers would

typically have between 4% and 8% reconstruction error after fabrication. The DOE used in this demonstrator was designed to have a signal to noise ratio >10 and a fabricated element reconstruction error of <2%. The optical system constrained DOE design such that some diffraction efficiency had to be sacrificed to ensure that the signal-to-noise ratio and reconstruction error criteria were satisfied. The final efficiency of the binary DOE was 50% with a fabricated reconstruction error of 1.4%. It contained 32 rows of 32 trapezoids with a minimum feature size of 1.8μm and a total DOE period of 96μm.

Experimental measurements of DOE output and associated reconstruction error were used to simulate the operation of the neural demonstrator for different sizes of network. By altering the total reconstruction error of the DOE and observing the variations in neural network operation, a measure of the scalability of the neural network can be gained from an optical viewpoint. The operation of the different electronic subsystems was assumed to be noise free although this is not the case in practice. Figure 56 shows the change in response of an N=8 network with increasing DOE reconstruction error. Each value of reconstruction error was simulated 10,000 times.

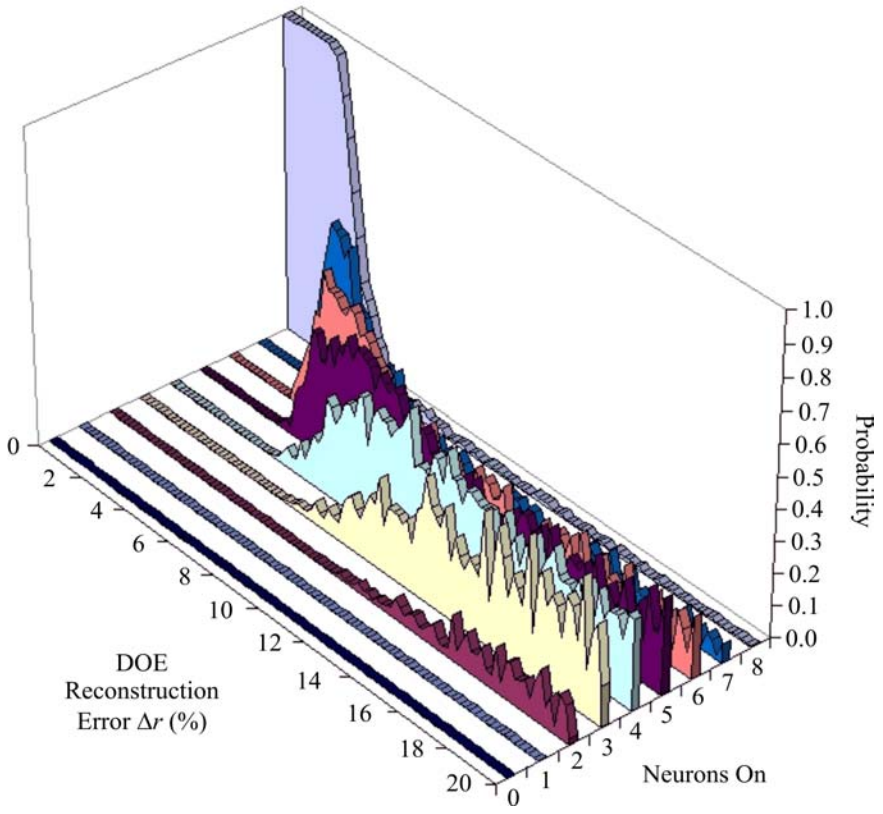

Figure 56: Reconstruction Error
Variation in neural network optimisation with DOE reconstruction error Δr.

The network operates optimally when the number of neurons on is 8. As the size of the reconstruction error increases, the probability that an optimal solution will be achieved is decreased. Unacceptable reconstruction error is defined as the point at which the proportion of optimal test runs drops below 90% of the total number of test runs. Variation of this quantity with network size is shown in Figure 57.

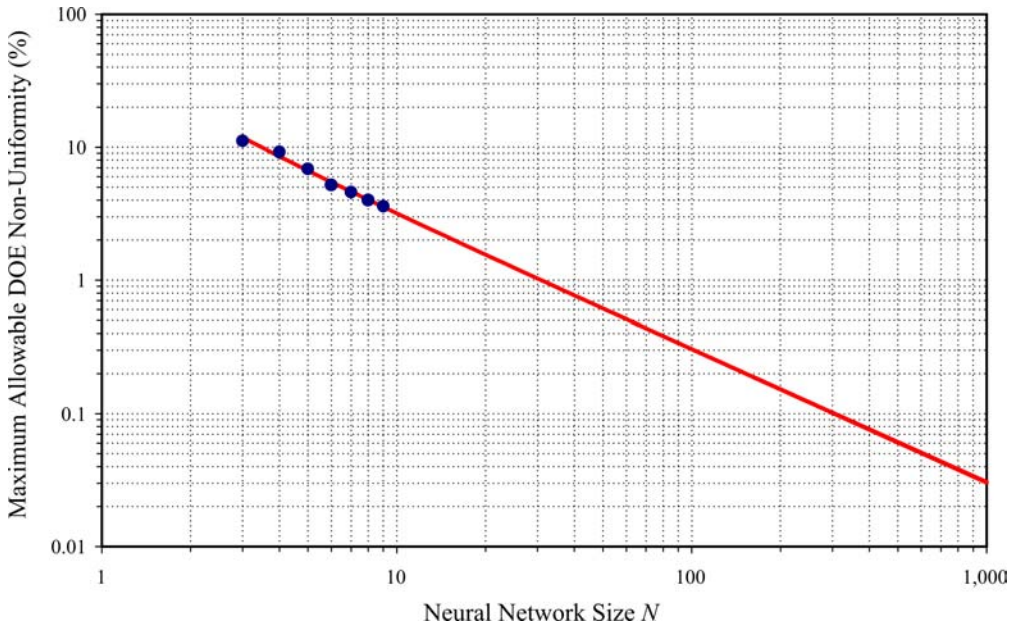

Figure 57: Maximum Allowable Non-Uniformity
Extrapolation of variation of maximum allowable DOE non-uniformity. If non-uniformity exceeds the maximum for a given network size then solution optimality will be compromised.

The solid line in Figure 57 is a fit to simulated discrete points allowing extrapolation of maximum non-uniformity for larger network sizes. This line has the equation:

$$\Delta r_{\max} = \frac{1}{uS_{DOE} + v} \qquad \textbf{Equation 53}$$

where u=3.288 and $v = -1.435$. Assuming that the minimum achievable reconstruction error in DOE fabrication is 1%, the maximum size of a neural network that can be implemented using this optical interconnection technology is N=30. Currently both demonstrators use an N=8 fan-out element.

Non-uniformity in the DOE is not normally considered to be noise since it does not affect the network in the same manner. Assuming that system noise is approximately the same across all channels if time averaged to infinity, non-uniformity provides certain neurons with a distinct advantage over others due to their improved ability to inhibit. Put succinctly, every neuron would not be equal.

4.5.2 Electronic System

The first generation electronic system, of which a single channel from 48 is shown in Figure 58, used analogue components to approximate the neural network.

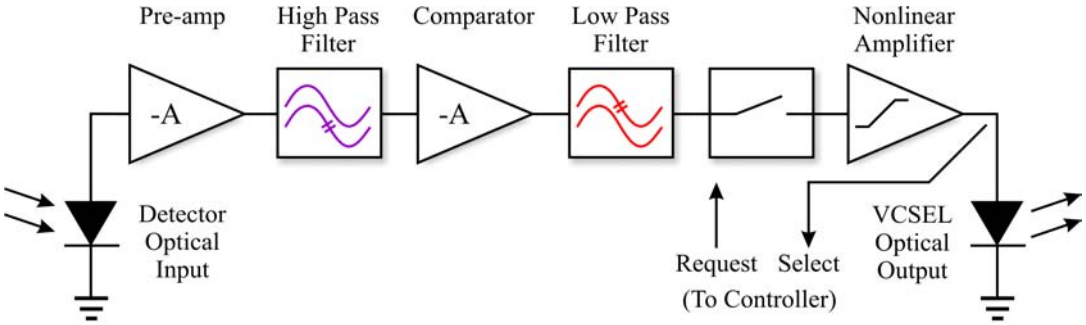

Figure 58: Single Electronic Channel from the First Generation Neural Network
Discrete electronic building blocks approximate the neural transfer function.

Speed of convergence can be adjusted by altering resistive and capacitive component values in both high and low pass filters. Adjustment of the transfer function is only possible by replacing electronic components, however component tolerances appeared to reduce solution optimality prompting a redesign of the electronic system.

The second generation system shown in Figure 59 uses off-the-shelf digital signal processors to provide the neural transfer function which allows network operation to be reprogrammed to tackle new applications with minimal, if any, adjustment to hardware. Non-linearities in the system can be compensated for at system startup using calibration routines. However, the most significant enhancement in this demonstrator is the ability to prioritise connection requests, a function ubiquitous in packet switch controllers.

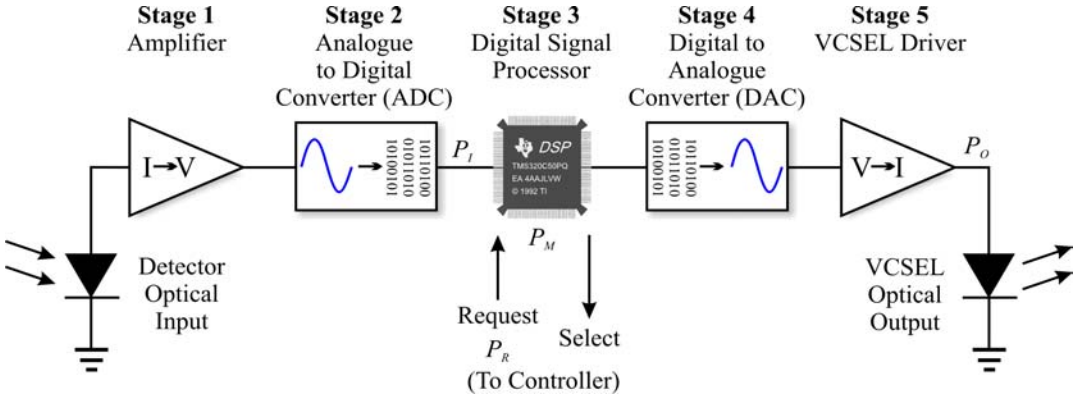

Figure 59: Single Electronic Channel from the Second Generation Neural Network
One DSP handles 16 channels. Its operation is modelled using a set of matrices to represent neuron states at specific stages. The stages are indicated by P_I, P_M, P_R and P_O.

The electronic system can be considered as consisting of five stages, each performing a specific task. At the optical input end there is the detection system which converts a

current generated by incident light into a voltage of magnitude specified by the amplification of the transimpedance amplifier. This is then converted by the second stage using an analogue-to-digital converter (ADC). The third stage consists of a DSP [135] which takes the digital information and performs a transfer function on it based on previous and requested values. The fourth stage consists of digital-to-analogue converters (DACs) that are fed the new activation levels from the DSP. The fifth and final stage takes the voltages from the DAC and converts them into an appropriate drive current for the VCSELs thus returning the signal into the optical domain. A single chip solution has been fabricated for stage five.

4.6 Simulation and Results

Simulation software was written that closely modelled the hardware system. This allowed analysis of potential problems and the determination of hardware requirements with respect to accuracy, tolerable noise and efficiency. This section outlines the simulator and discusses parameter limitations. Hardware minimisation is illustrated that not only improves convergence times but significantly enhances scalability and network robustness.

4.6.1 Simulator Algorithm

The simulator divides the demonstrator up into four *planes*. Each plane contains information about a neuron at a certain point and is therefore represented by a matrix of the same size as the neural network. A single loop iteration is illustrated and, defining variables as we proceed, the computations required to move from one plane to the next are examined. There are three stages in an iteration with all planes initially containing zero values, except for the *request plane* P_R. This plane specifies a set of requested connections and is sent to the DSP by an external controller. If the value of a request is zero then a neuron cannot evolve and will never turn on. The higher the value here, the more priority a neuron has. Next there is the *output plane* P_O which describes neuron output levels from stage 5 in Figure 59. These levels are voltages quantised to the resolution of the DAC. Their values reflect the current state of the neural network with a set of maxima and minima indicating network convergence. The *input plane* P_I measures the light input from the optical system and is a voltage level from stage 2 in Figure 59 quantised to the bit depth of the ADC. Linearly distributed noise is added to this value representing both ADC quantisation error and optical system noise. Every

neuron stores an internal value from the previous state. This value is held in the *memory plane* P_M and is a real number. Figure 60 illustrates information flow between these planes in the simulation.

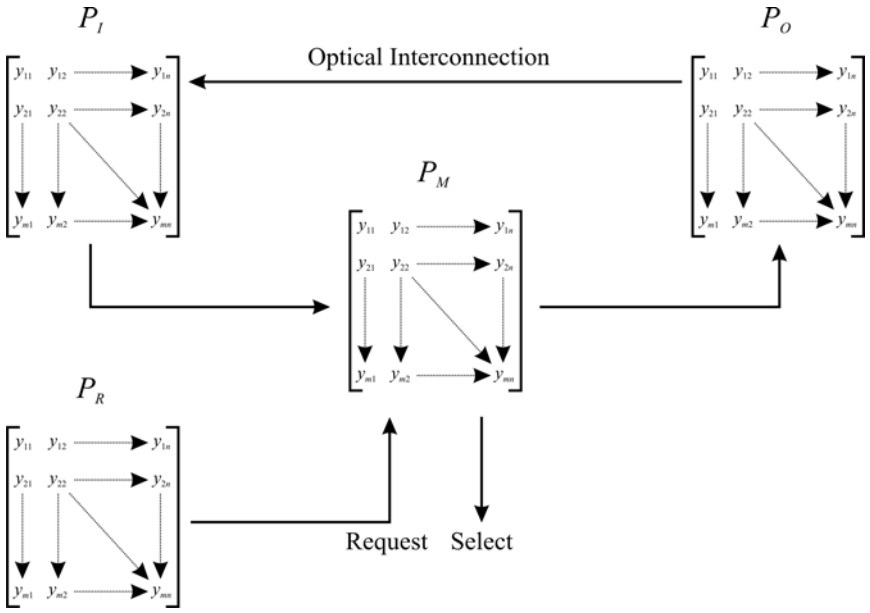

Figure 60: Simulator Planes

Simulator planes are interrelated by a predefined information flow.

A single iteration is considered to be complete when computation returns to the starting point. To illustrate the complexity involved, we will examine the computation required to complete a single iteration. Starting from the memory plane, the first stage in an iteration is to calculate the new values for the output plane. This is done using:

$$P_{O_{ij}} = P_{R_{ij}} \frac{V_{o\max} - V_{o\min}}{1 + e^{-\beta P_{M_{ij}}}} \qquad \textbf{Equation 54}$$

where $V_{o\min}$ specifies the voltage used to switch a VCSEL off, $V_{o\max}$ the voltage level when a VCSEL is fully on and β the gradient of the neuron's sigmoid activation function. The output values in P_O are quantised to Q_{DAC} bits thus representing the DAC.

To calculate the input plane we require a temporary plane which we will call P_{temp}. This is used to determine the VCSEL optical output power presuming a linear relationship between minimum and maximum values:

$$P_{temp_{ij}} = \left(P_{O_{ij}} - V_{o\min}\right) \times \left(\frac{P_{o\max} - P_{o\min}}{V_{o\max} - V_{o\min}}\right) + P_{o\min} \qquad \textbf{Equation 55}$$

where $P_{o\min}$ is the optical power generated by the VCSEL when $V_{o\min}$ is applied to the DAC and $P_{o\max}$ is the optical power generated by the VCSEL when $V_{o\max}$ is applied.

From this we can calculate the amount of power in each spot produced by the VCSELs presuming that the DOE distributes the incident optical power evenly:

$$P_{temp_{ij}} = \frac{P_{temp_{ij}} T_{DOE}}{100 S_{DOE}} \qquad \textbf{Equation 56}$$

Where S_{DOE} is the number of spots created by the DOE symmetrical cross, calculated using $S_{DOE} = 2(m+n-2)$, and T_{DOE} is the percentage transmission of the DOE.

Next, the total power incident on each detector needs to be calculated:

$$P_{I_{ij}} = \sum_{k \neq i}^{m} P_{temp_{kj}} + \sum_{k \neq j}^{n} P_{temp_{ik}} \qquad \textbf{Equation 57}$$

This power value can then be translated into a voltage using:

$$P_{I_{ij}} = \left(P_{I_{ij}} - P_{i\min}\right) \times \left(\frac{V_{i\max} - V_{i\min}}{P_{i\max} - P_{i\min}}\right) + V_{i\min} \qquad \textbf{Equation 58}$$

where $V_{i\min}$ is the minimum input voltage from the detectors produced when $P_{i\min}$ watts or less is incident and $V_{i\max}$ is the maximum input voltage from the detectors produced when $P_{i\max}$ watts or more is incident.

Random noise from the system between N_- and N_+ is then added since the neural network thrives on noise and therefore it must be simulated. The value used is generally ±½ least significant bit of Q_{ADC} since this is normally larger than the optical noise. Once noise has been added, the voltage held in P_I is then quantised to Q_{ADC} bits.

The final stage in the loop is to calculate the new values in the memory plane P_M. This is performed using:

$$P_{M_{ij}} = P_{M_{ij}} + \lambda_{ij}\left(-A \times P_{I_{ij}} + B\right) \qquad \textbf{Equation 59}$$

where A is the neural network optimisation parameter, b is the neuron bias and λ_{ij} a virtual time increment since the discrete system models an analogue one and therefore must have a time interval at which network weights are recalculated.

Figure 61 shows neural network conversion in a sample simulation run. Each trace represents the output from a single neuron. The system parameters used reflect hardware construction for the second generation system where N=8. The network optimisation parameters are A=1,250, b=50, β=0.02 and $\lambda_{ij} = 1/30$ with request values ranging from 1 to 10. The DACs output voltages from 0 to 5V at 8-bit resolution resulting in a VCSEL optical output power of between 0.1 and 1.5mW. The DOE

creates a symmetrical cross with 28 inhibitory signals at 60% transmission. Each detector therefore receives a signal ranging from 30 to 450μW of incident optical power. This is converted back into a voltage between 0 and 5V at 8-bit ADC resolution. Noise is present in this conversion and ranges from −10 to +10mV.

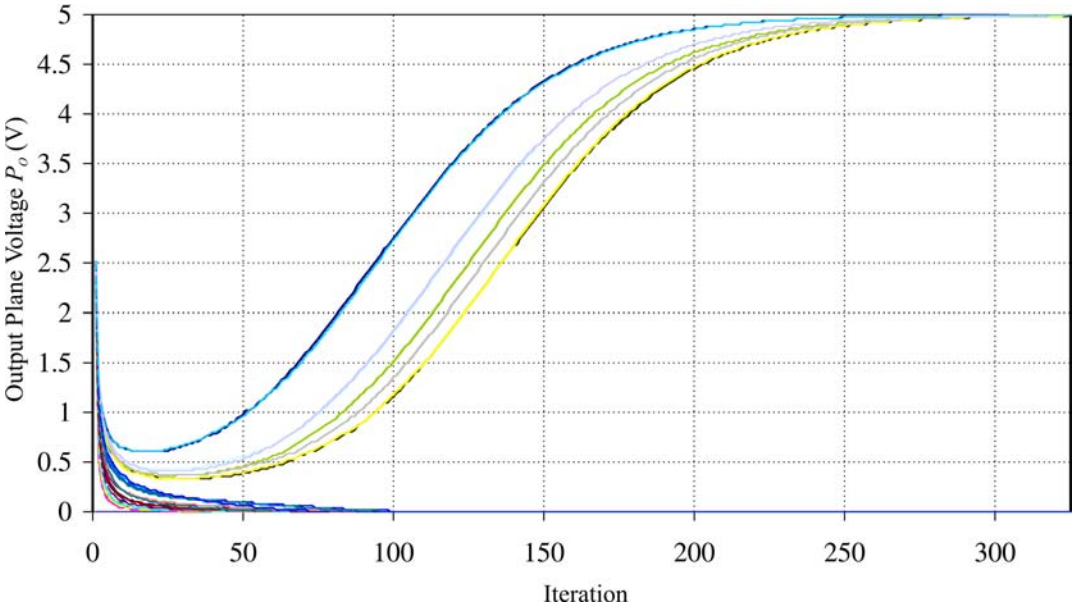

Figure 61: Neuron Evolution

Monitoring the output plane allows us to determine if neural network convergence is complete. It is clear that a stable decision has been reached after around 300 iterations.

An important figure of merit is the number of *iterations to convergence*. Figure 62 plots 1,000 network convergence times.

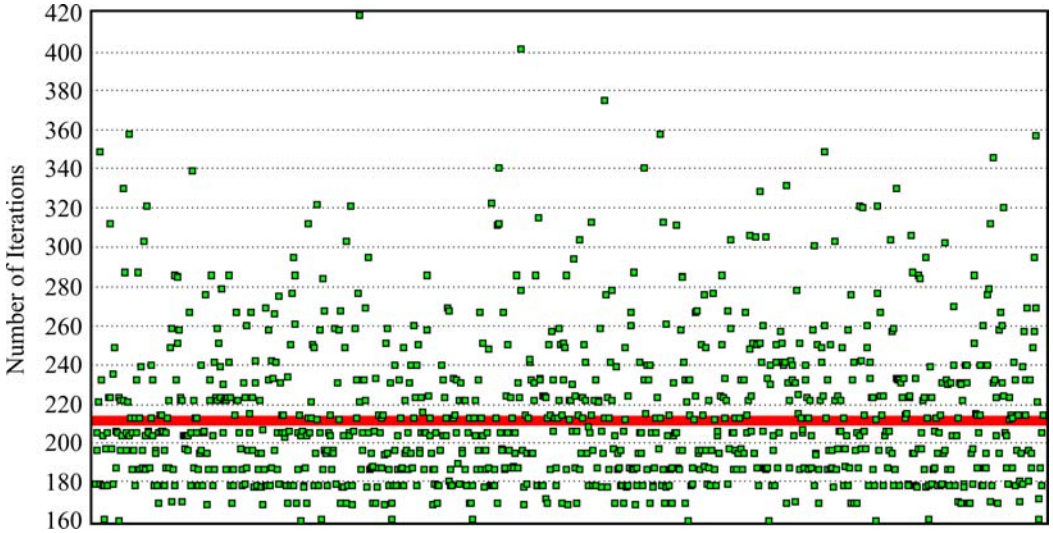

Figure 62: Iterations to Network Convergence

Plot of convergence times for 1,000 random routing requests all with a load of 0.5625. The majority of tests indicated convergence in under 300 iterations. These test runs did not produce any invalid switch configurations.

Iterations to convergence represents the number of complete system cycles required for the neural network to converge to a steady state and is essentially a measure of how fast the system can be run. The distribution shown in Figure 62 indicates that a point attractor exists for the network as there is a skewed gaussian distribution around a median number of iterations as indicated by the bold red line. There also appears to be a set of local minima clearly represented by rows of solutions at particular numbers of iterations.

4.6.2 Simulator Results

The first simulations performed examined the total number of iterations required for the network to converge given different request plane values P_R. Normally request values are set at 1 to allow a neuron to evolve or 0 to prevent it from evolving. However, the addition of prioritisation requires the use of a range of values. Simulation was used to establish the range of P_R values that would allow evolution and their consequent effect on system performance. Two studies were made - one on a lower triangular test matrix which has a single valid and optimal solution:

$$P_R = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \qquad \textbf{Equation 60}$$

and one on a fully loaded network which has no distinct solution:

$$P_R = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \qquad \textbf{Equation 61}$$

Figure 63 clearly illustrates that a completely loaded square network requires longer to converge than the more lightly loaded triangular one. As expected, the error margin in a square request matrix is larger as it has no clearly defined attractor.

It was discovered that if the value of the request is less than 1 then the network would not converge reliably. From 1 to 10 there is continual improvement and beyond 10 negligible performance difference.

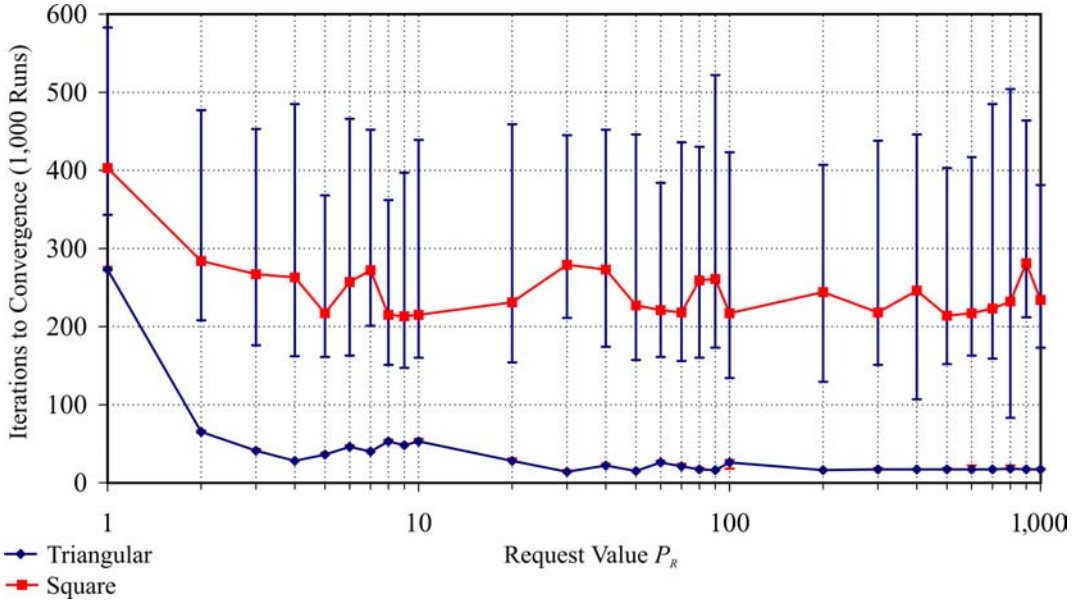

Figure 63: Effect of P_R on Iterations to Convergence
Varying the magnitude of the request value has a similar impact on two differently loaded networks.

Any discrete implementation has the exclusive parameter λ_{ij} that had not been optimised in previous work. Therefore simulation was undertaken to find an optimal value. It was discovered that to enable network convergence $0.003 < \lambda_{ij} < 0.08$ must hold true. Outwith these limits the network does not provide a valid solution. As can be seen in Figure 64, the best average conversion time that is not near limiting values, and therefore possibly unstable, is when $\lambda_{ij} = 0.03$.

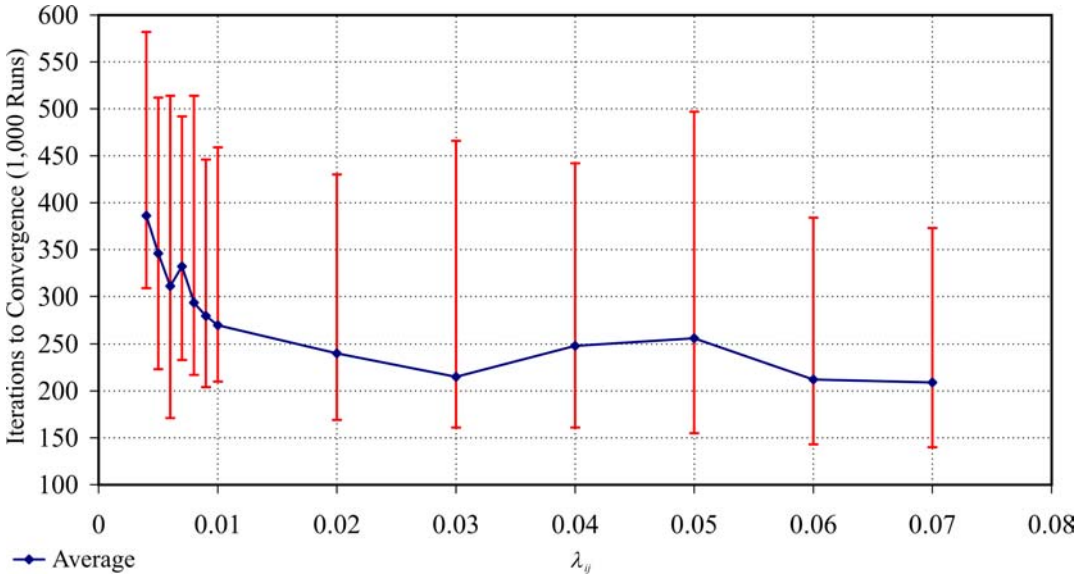

Figure 64: Effect of λ_{ij} on Iterations to Convergence
Variation of digital simulation step size to find an optimal value. This value is a time constant that approximates the analogue system.

The next aspect examined was network scalability. Initially Q_{ADC}=Q_{DAC}=8 thus modelling the preferred bit depth implementation. As expected, the number of iterations to convergence scaled in a linear manner with network size *N*. Based on current component tolerances, the upper maximum network size proved to be *N*=21, or 441 neurons, as shown in Figure 65. Degradation of solutions began after this point as the system started to produce invalid results. These simulations were performed under full network load conditions.

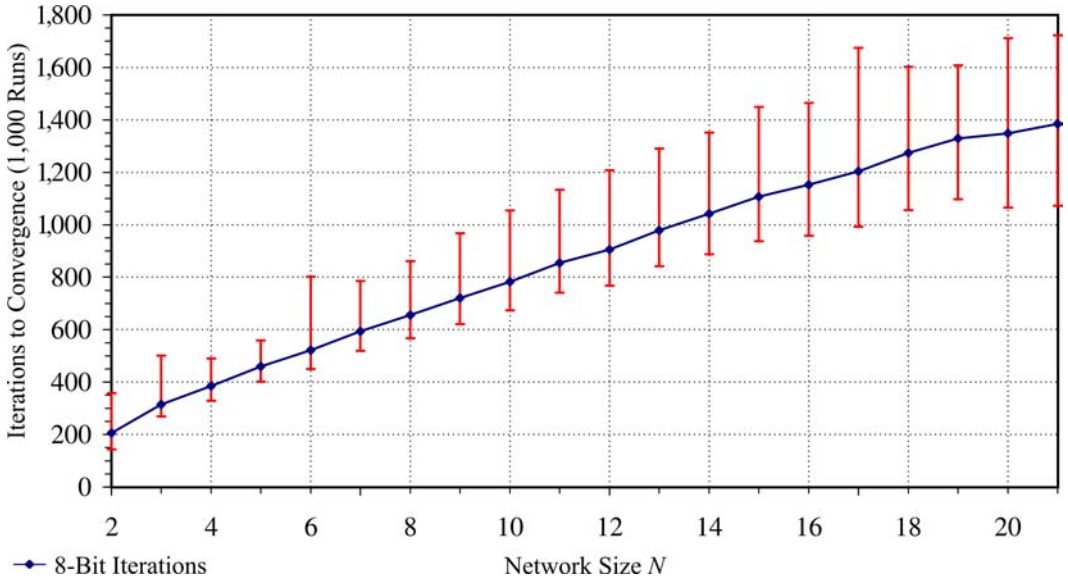

Figure 65: Effect of Network Size *N* on Iterations to Convergence with Q_{DAC}=8
Scalability of an 8-bit system displays a linear increase in convergence time that is proportional to network size *N*.

Apart from proof of principle, the goals of simulation were to minimise hardware and improve system performance. Reduction of Q_{DAC} to 1, thus thresholding neural output, provided some interesting results. It confirmed that a digitally driven system was not only feasible but that it improved both scalability and convergence times. Simulation has shown that a network with Q_{DAC}=1 will scale up to *N*=63 or 3,969 neurons. This is graphed in Figure 66 which also shows that the average number of iterations increases only slightly with network size. However, at *N*=62 this number begins to increase dramatically with invalid results appearing at *N*=64 and beyond. These simulations were again performed under full network load conditions.

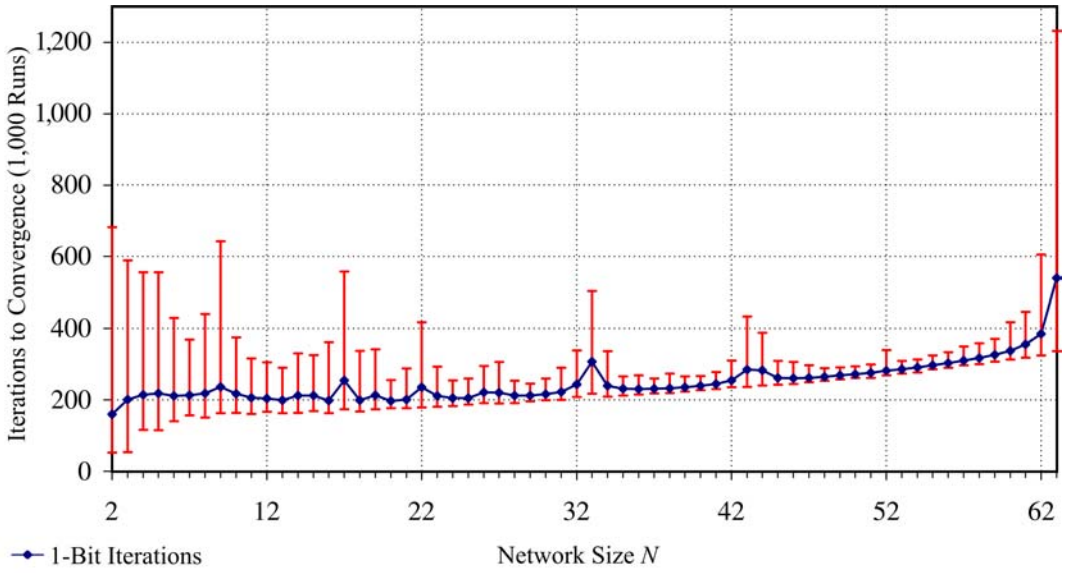

Figure 66: Effect of Network Size *N* on Iterations to Convergence with Q_{DAC}=1
Examination of the scalability of a digitally driven system shows that convergence time is not greatly effected by network size.

Therefore removal of hardware complexity, in this case digital to analogue converters, improved both scalability and performance dramatically.

To add a third dimension, both systems were also simulated under varying network load conditions. This was started at no load and increased until all connections were requested. The 8-bit network response is shown in Figure 67 and it copes reasonably well with the number of iterations required increasing in accordance to load, only falling off slightly under full load conditions.

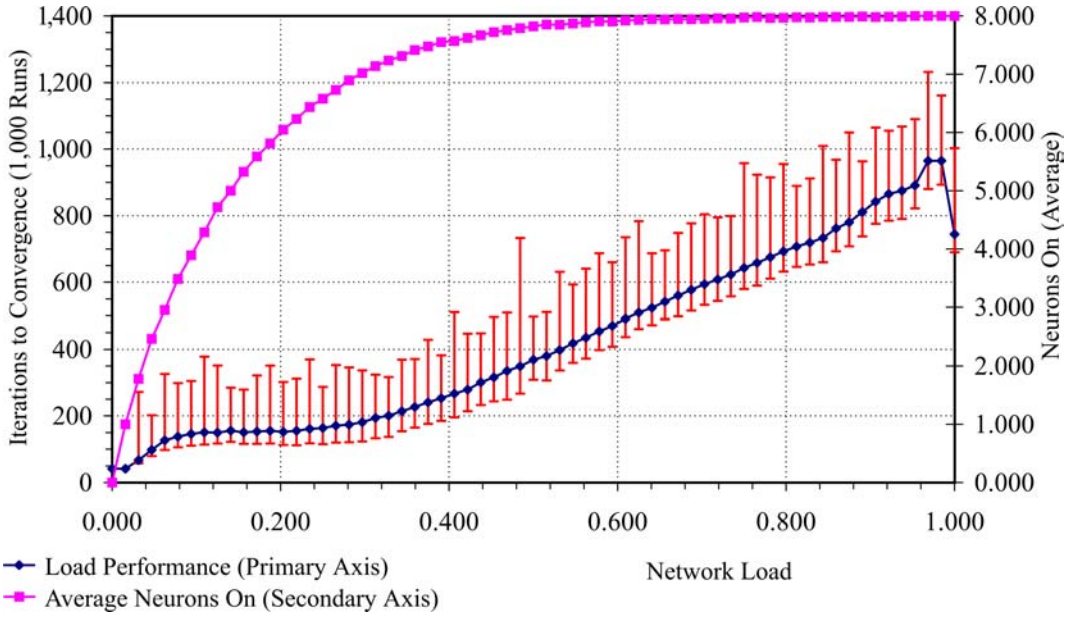

Figure 67: Effect of Load on Iterations to Convergence with Q_{DAC}=8
Performance of an 8-bit driven *N*=8 network under varying load conditions.

The *Average Neurons On* axis details the number of neurons in an on state at convergence and is averaged from 1,000 test runs. It is plotted against the secondary axis and is an indication of solution optimality.

In comparison, a thresholded network copes particularly well as load increases. Figure 68 shows that the number of iterations required increases slowly and hardly breaches the 500 iterations mark even under maximum load. Conversely, the 8-bit system has achieved this point when load reaches 0.4. Indeed, no convergence time is ever less than 500 iterations if network load exceeds 0.7, even if outliers are included.

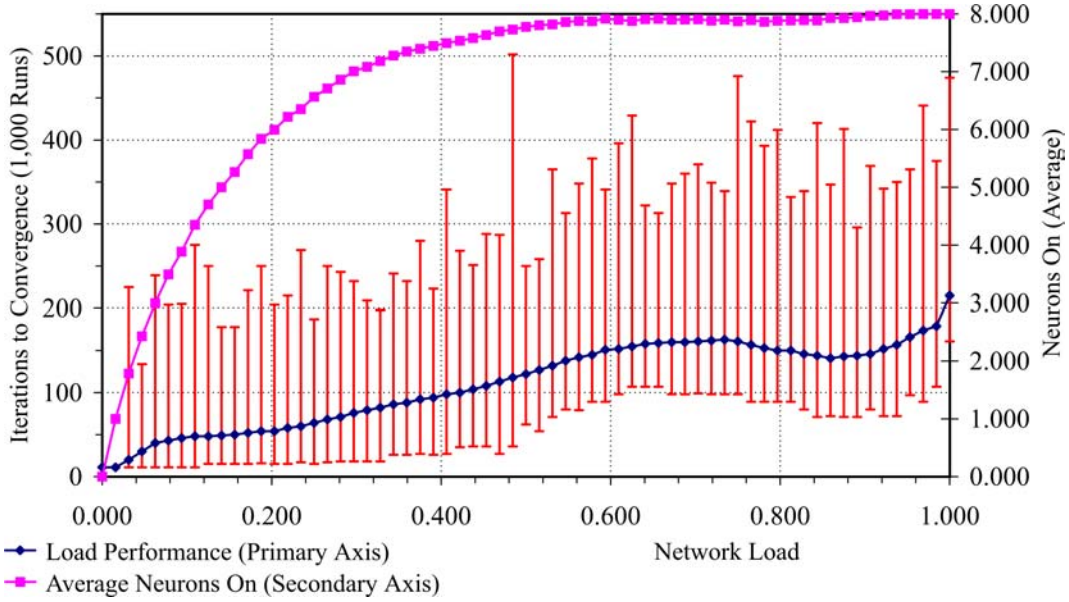

Figure 68: Effect of Load on Iterations to Convergence with Q_{DAC}=1
Performance of a digitally driven N=8 network under varying load conditions.

The digital thresholded system outperforms an 8-bit one in all ways as far as convergence is concerned. Unfortunately, speed of convergence and scalability is traded off against quality of solution. Expanding the secondary axis in Figure 67 and Figure 68 to give Figure 69 shows that, on average, sub-optimal solutions were generated slightly more often by the digitally driven system than by the 8-bit one. Under the high load conditions shown in Figure 70, an 8-bit system gives a solution that is below optimal approximately 3% of the time whereas the digital system is below optimal about 9% ±1% of the time.

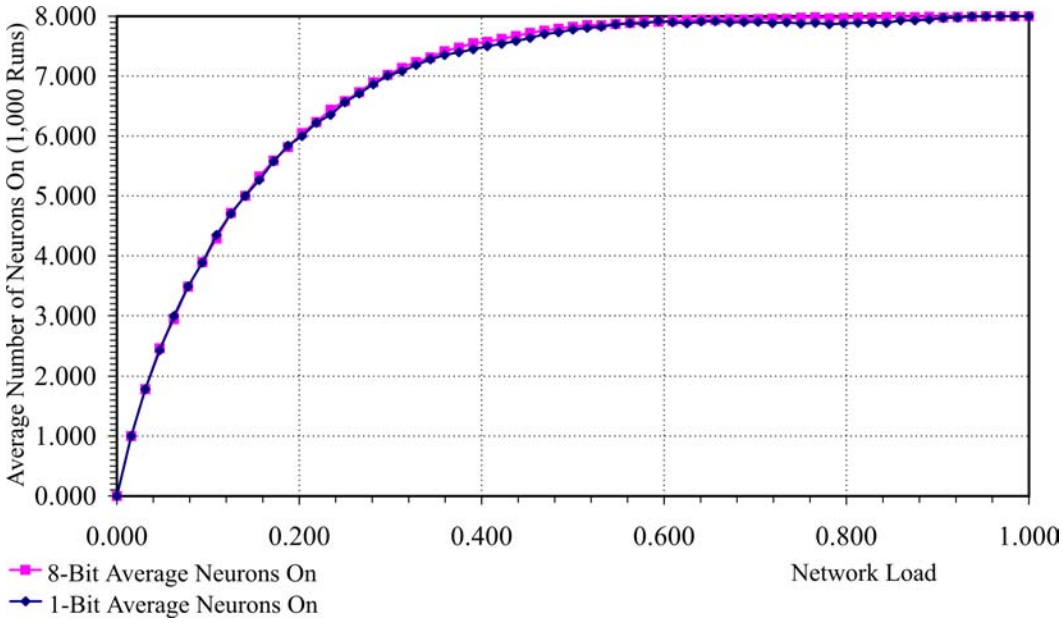

Figure 69: Comparison of Digital and Analogue Drivers

Comparison of quality of solution between 1-bit and 8-bit systems.

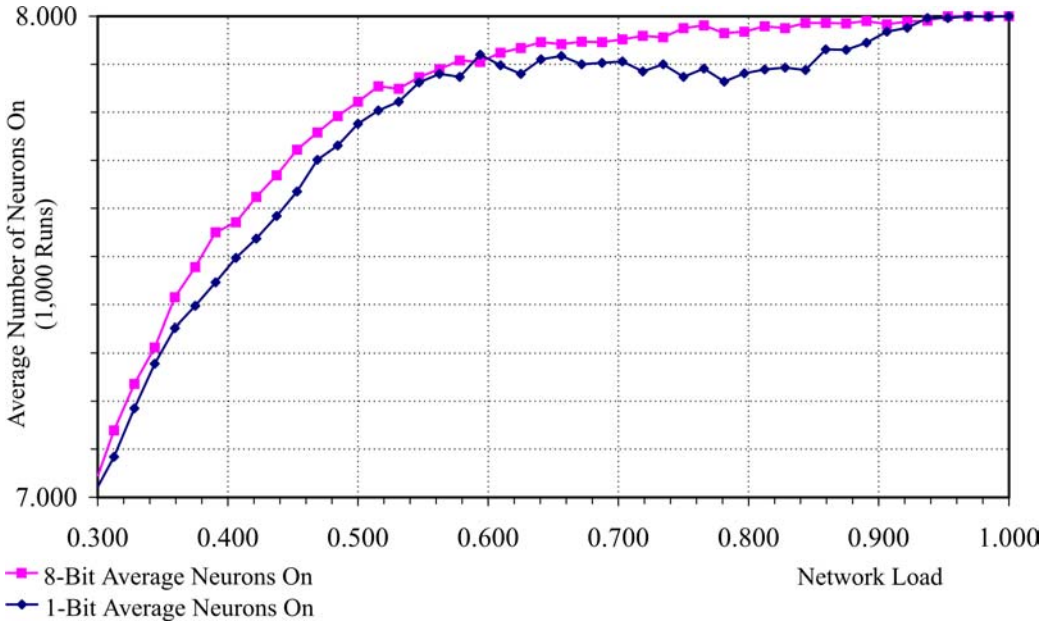

Figure 70: Comparison of Digital and Analogue Drivers (Expanded)

Comparison of quality of solution between 1-bit and 8-bit systems under high load levels. The optimality of a 1-bit system solution is slightly below that of the 8-bit system.

In a similar attempt to minimise input hardware, the minimum analogue to digital converter bit resolution Q_{ADC} required to provide convergence was determined. Obviously digital input is not possible in this situation, as was verified during simulation, since there would only be one value of inhibitory input to a neuron and consequently no discrimination between single or multiple incident inputs. Simulation involved varying the value of Q_{ADC} and examining the number of iterations required for

convergence. Some rather unexpected results were observed. For example, a 6-bit ADC produced valid and optimal results beyond what was believed to be its limiting bit resolution and a 16-bit ADC managed to activate only a few neurons and did not produce near optimal solutions. Additional noise was inserted into the 16-bit ADC system which resulted in it beginning to converge again. Further simulation supported the following hypotheses:

- If a system starts off balanced with a full request matrix, noise is an important factor for convergence. As bit resolution increases, LSB noise decreases so the system will not converge within the maximum number of simulated iterations, in this case 5,000. Therefore the introduction of a lower resolution ADC adds sufficient noise to guarantee convergence regardless of optical system characteristics.
- If a system starts off unbalanced such that there is a distinct solution or descent gradient then noise is no longer dominant. Such systems are capable of converging with ADCs that do not have sufficient resolution to determine whether a single neuron is active. This is because quantisation error pushes the reading into the discernable regime with a calculable probability. As the descent gradient is in a well defined direction, this is sufficient to cause network conversion characterised by increasing convergence times and decreasing probability of error due to quantisation.

These results are interesting in that they show that a digitally driven implementation can outperform an analogue-like 8-bit system and can be scaled to a considerably greater extent. A digital implementation can be considered as essentially digitally thresholded neurons that are either on and winning or off and losing. The reason that a digital system converges faster is that instead of gradually turning on, as the analogue system does, neurons switch from off to on in a single step. This adds impetus to convergence but unfortunately also a hastiness that can lead to sub-optimal results more often than in the 8-bit case. Nevertheless, the possible hardware savings by bypassing DACs and analogue VCSEL drivers for a more rapid decision and increased scalability are very tempting provided that a sub-optimal solution of N-1 neurons on can be tolerated less than 10% of the time under higher load conditions.

The system's major speed limitation is processing power. However, if the system were to use a hardware implemented transfer function then we could see its speed approaching 1GHz. Given a digitally driven system at this frequency, a fully loaded N=8 neural network with defined convergence in a generous 400 cycles could produce

2.5 million solutions per second. The impressive thing about this system is that we could scale up the size of the problem to *N*=60 and there would be no difference in the number of cycles required for the system to converge.

Scalability is currently limited in 2001 by VCSEL array size. Poor process technologies limit any system implementation to *N*=16 presuming the use of commercially available components. This is closely followed by DOE non-uniformity which begins to influence solution optimality at *N*=30. However, process technologies continually advance and as tolerances improve so will the maximum implementable network size.

4.6.3 Scheduler Performance Comparison

It is interesting to note that this neural solution when applied to the scheduling problem [127] can, in terms of switch cycles, outperform state-of-the-art digital solutions. Simulations of an *N*=16 neural network scheduler were undertaken in order to make performance comparisons with other scheduler designs. The simulations were performed under uniform traffic conditions and the mean delay, measured in packet periods, was plotted against offered load which is the probability of a packet arriving at each input. Figure 71 summarises the results of this exercise.

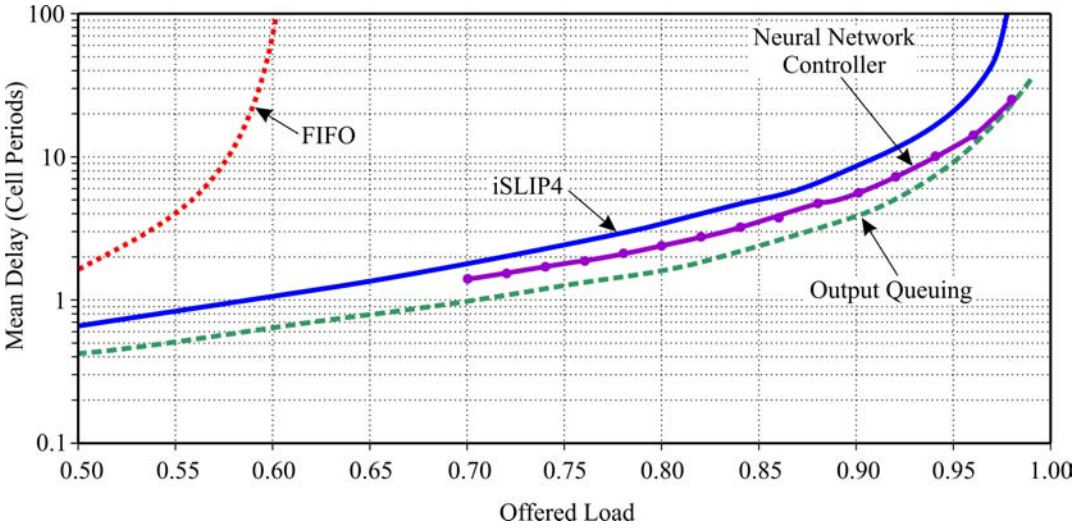

Figure 71: Performance Comparison
Comparison of neural network scheduler with a state-of-the-art scheduler iSLIP4. The advantage of the neural network is clearly visible under high levels of load. Output queuing is the theoretical optimal.

The uppermost dotted curve shows the situation when the inputs are simply buffered in a *first in first out* (FIFO) fashion. FIFO queues suffer from the problem of *head of line* (HOL) blocking in that if the foremost packet in the queue is blocked by another

request, it also blocks all the packets behind it even if their destinations are not in contention. As might be expected, a scheduler based on FIFO buffering suffers severe performance degradation under increasing load. The lowest dashed curve represents the theoretical best that can be achieved. This is described as *output queuing* and is calculated assuming an ideal, and infeasible, switch fabric where packets have only to wait for a vacant slot on the output line. The solid line represents an algorithm called iSLIP4 [136] which could be implemented in CMOS electronics for a high speed switch of this size. The line drawn through a series of simulated points shows the neural network scheduler performance and its favourable throughput at loads from 0.7 upwards.

Although our current hardware is not built for the high speed of an individual switch cycle, this is purely due to our system being designed as proof of principle. Indeed, the simplicity of the system is such that extremely high speed versions of the hardware are easily conceivable. These facts, combined with encouraging results on the scalability of a digital system, shows that a neural network with an optoelectronic interconnect will provide an excellent solution to the packet switch control problem or indeed to any variation on the quadratic assignment problem.

4.7 Conclusions

This chapter has examined the issues that currently limit conventional sequential and parallel architectures and has shown how neural networks combine many simple processing elements in a highly interconnected manner thus providing a *connectionist approach* to problem solving. A neural network has been mapped to a common variation on the quadratic assignment problem found in packet switch schedulers and solution optimality demonstrated which exceeds that of conventional algorithms, approaching theoretical optimal under extreme load. Hardware minimisation has been demonstrated that not only removes system complexity but introduces both performance and scalability to an algorithm normally intractable to scale. Regardless of whether the system is just N=2 or a massive N=60 it will still require the same computation time. However, the key to this unprecedented scalability lies with optics as it is not feasible to implement in electronics due to a quadratic increase in interconnection requirements.

Optics enables this system and puts within reach a viable alternative to linear programming techniques. The system examined here can essentially be used to solve

any quadratic assignment problem where time is of the essence. Its ability to handle larger order problems without serious performance degradation emphasises the contribution that such systems could make to the field of highly parallel computing.

5 Neural Network Demonstrators

This chapter will examine the strengths and weaknesses of two generations of the neural network demonstrator. It considers system construction, nuances, limitations and performance. The design decisions behind the second generation demonstrator are outlined in detail and performance assessments are made comparing it with previous hardware neural networks. Note that algorithmic and application issues for both neural networks are discussed in chapter 4, with which this chapter presumes familiarity.

5.1 First Generation Neural Network

The first generation demonstrator [137]-[139] was constructed as a discrete component analogy of the neural network model. Although the system served as an initial proof of principle, the author was not involved in the initial stages of specification and construction. Therefore this section provides an overview, examination of implementation issues and series of results but does not include detailed design specifications.

A simplified overview of both demonstrators with regard to functionality and information flow was presented in section 4.5. This divided the demonstrators into two distinct systems, optical and electronic, a simplification which will again be used to address complexity.

Although the first generation demonstrator used optical components capable of implementing an N=8 system, only 48 neurons were actually employed due to interfacing constraints. Therefore, this neural network is considered to be a 6×8 system.

5.1.1 Optical System

Given certain intrinsic values and basic lens formulae, the optical system dimensions can be calculated. These intrinsic values consist of overall system magnification and DOE working distance. They were considered intrinsic as their alteration requires the replacement of either VCSEL array, DOE or detector array. Modelling was performed using Matlab V4.2.1c which was given these intrinsic values and a set of available lens sizes with differing focal lengths.

This system relies heavily on the properties of the DOE which splits up incoming light and diffracts it on to the appropriate detectors. For optimal results from the DOE,

incoming light must be nearly, but not quite, collimated. However, each VCSEL has a divergence of approximately 8°, thus requiring collimation. In addition, if the 250µm spaced VCSEL array is to be focussed onto the 1.5mm spaced detector array, a magnification of ×6 is required. Therefore, the lens system must perform collimation before the DOE and magnification after.

The Matlab simulation produced hundreds of values on each test pass as component positions were gradually varied. A metric was required to grade each result which would represent both overall optical system size d and beam divergence through the DOE. As system size was considered more important than beam divergence, it was given twice the weight in the metric. Thus, the quality of any valid solution could be estimated while the program exhaustively tried different lens combinations and positions.

Given a maximum optical system size of 1,000mm, 5mm as the minimum distance between components and a maximum deviation of ±50% of f_1 between VCSEL and lens 1, the dimensions shown in Figure 72 were calculated as optimal given that only lenses of specific sizes were available.

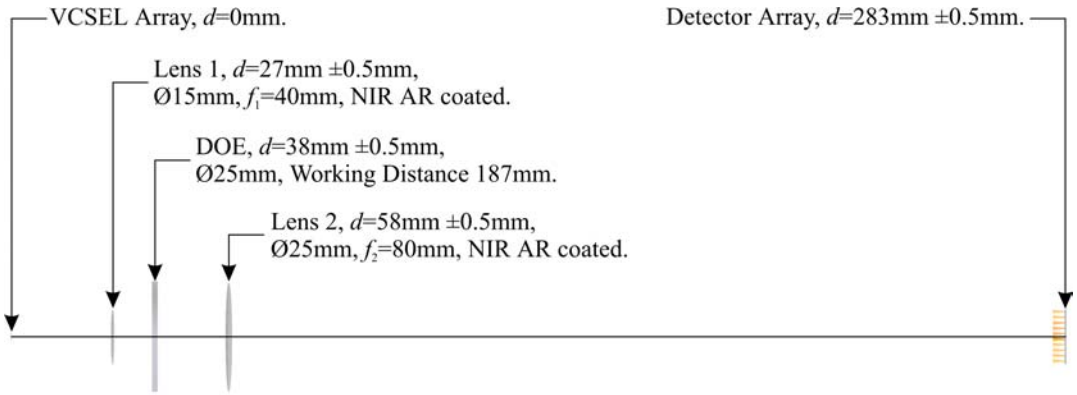

Figure 72: First Generation Optical System
The components in this diagram are scaled relative to each other and to the overall system size. Distance d is measured from the VCSEL array on the left at d=0mm. The detector array on the right is at d=283mm which is the overall system size. Data flows optically from left to right. Lens focal lengths are represented by f_x where x is the lens number.

The final system has an overall size of 283mm and magnification of ×6. Subsequent implementation and alignment proved this to be a valid solution.

The VCSEL array used in the first generation system is a Lase-Array SN 2640 [140]. It has 64 VCSELs in an 8×8 configuration operating at λ_{1g}=760nm. The array's optical output power has been profiled and the results are shown in Figure 73.

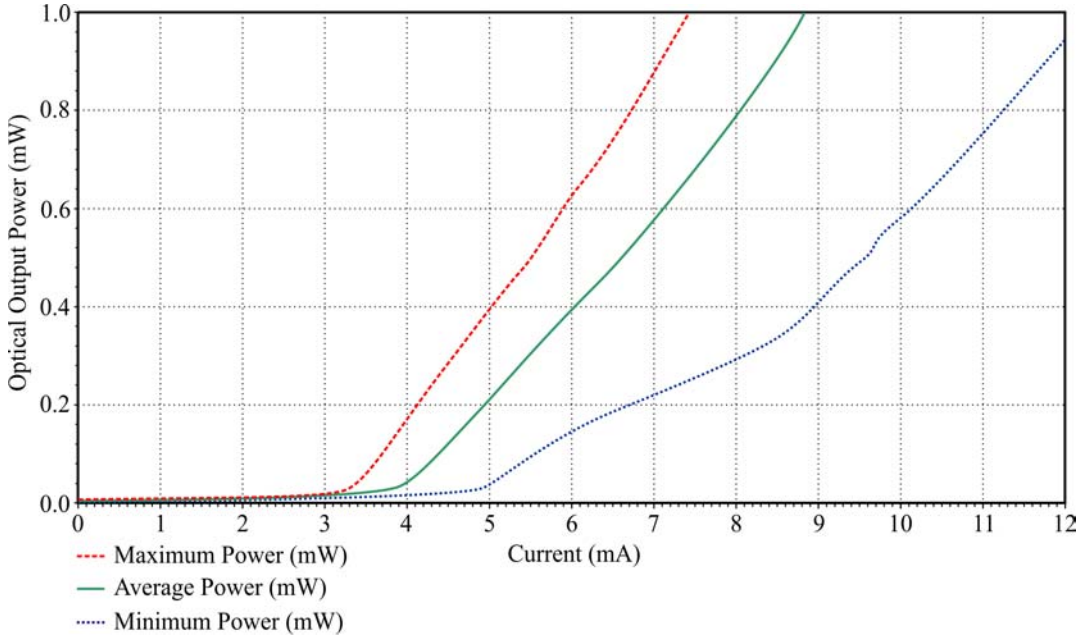

Figure 73: Lase-Array SN 2640 VCSEL Optical Output Power Characteristics
Optical power generated through application of a current. This diagram shows minimum and maximum optical output powers sampled across all 64 VCSEL elements.

Unfortunately, there is a large optical power variance across the array. However, as the gradient of slope remains constant, the addition of a current offset will allow for the calibration of individual VCSELs. Current to voltage curves are an important characteristic when designing driver circuitry and can be seen for this array in Figure 74.

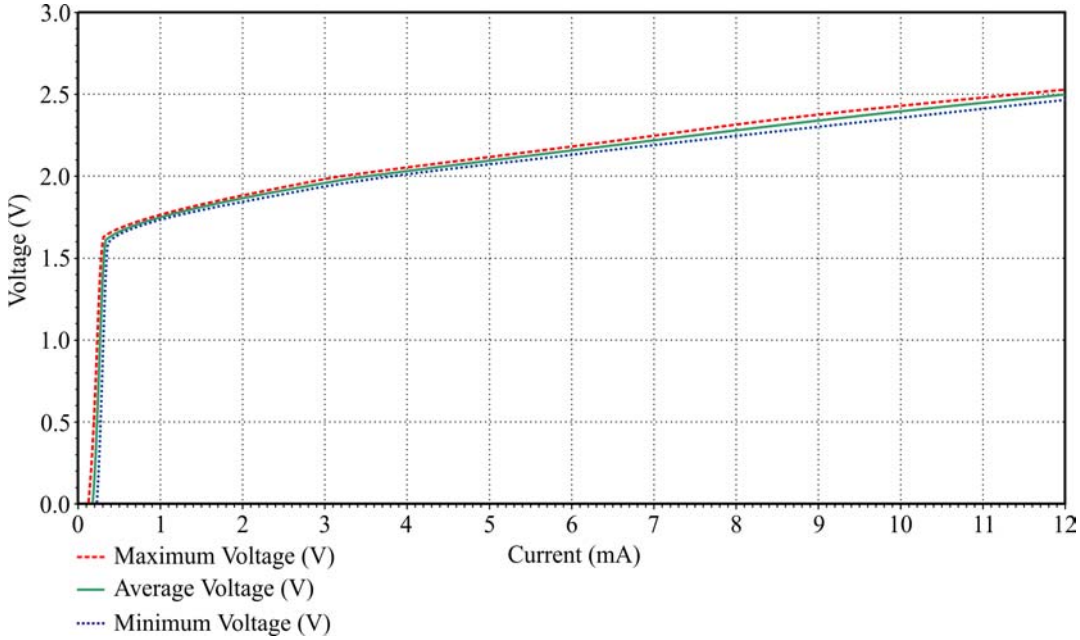

Figure 74: Lase-Array SN 2640 VCSEL Current to Voltage Characteristics
Voltage developed across VCSEL through application of a current. Minimum and maximum values generated by sampling across all 64 elements in the array.

The current to voltage characteristics of each VCSEL across the array are comparable and, according to Ohms law, have a similar resistance. This similarity is desirable as any variance adds another parameter that needs to be taken into account during system design and construction.

An issue which became apparent during the latter stages of the project was sensitivity of the VCSEL array to ambient temperature, as shown in Figure 75. This was noticed as the VCSEL array was originally profiled in a distinctly colder environment than that in which it was operated. Previously profiled output powers no longer matched those measured during construction and alignment leading to a complete recalibration of the array.

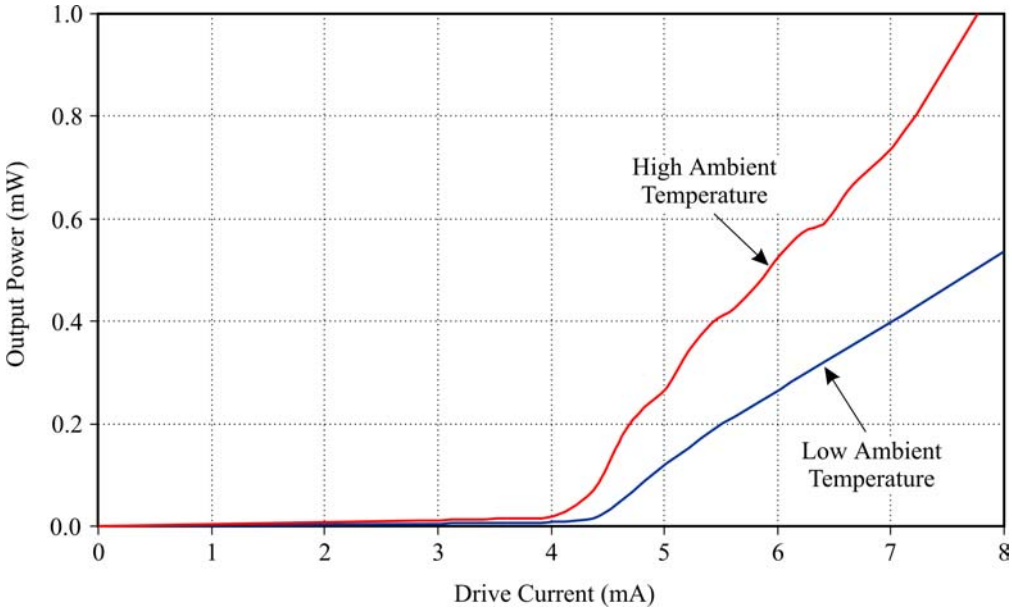

Figure 75: VCSEL Array Temperature Stability
The Lase-Array SN 2640 proved sensitive to ambient temperature. VCSEL responses are plotted for both low and high ambient temperatures to illustrate how drastic this variance is.

Any rise in temperature decreases a VCSEL's threshold current and increases its power output gradient. This variation, both in absolute terms and particularly in distribution across the array, provides one source of noise to aid convergence. However, the response displayed by each VCSEL to any change in temperature is similar but not identical to that observed in others.

Repositioning the VCSEL array with relation to lens 1 alters focussing on the image plane. To maintain a sharp image and thereby minimise an order's spot size, the

VCSEL array must remain at the focal point of lens 1. Therefore, relative displacement must be approximately equal to f_1.

The massively parallel interconnection seen in this system is enabled by the DOE where the pattern generated matches the inhibitory interconnect patterns derived in chapter 4. Figure 76 shows the interconnect patterns required to implement the crossbar packet switch scheduler (a) and the Banyan scheduler (b). Note that the additional orders in the Banyan network interconnect represent the internally blocking connections which must be inhibited.

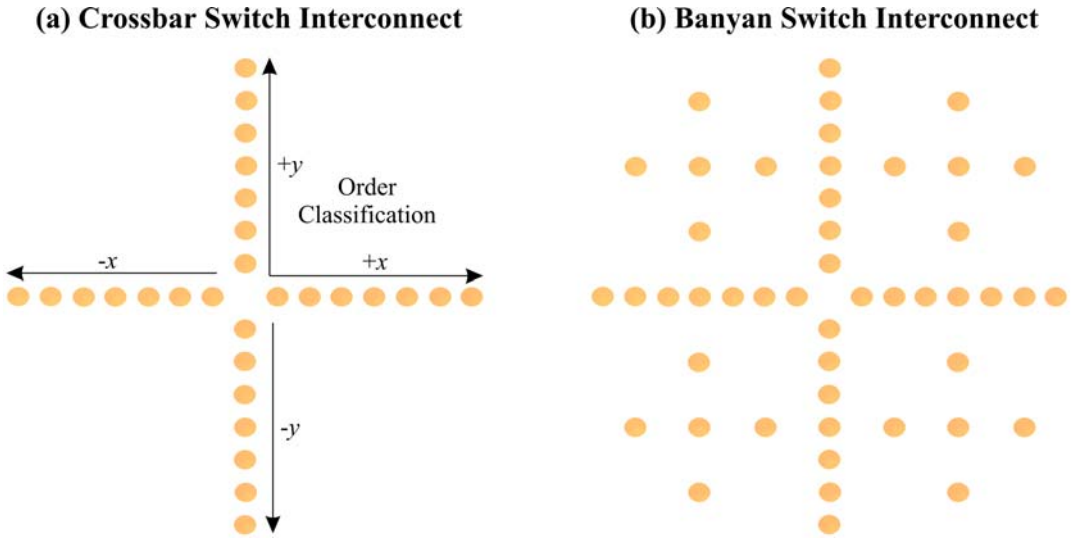

Figure 76: Diffractive Optic Element Interconnect Pattern
Each spot represents an order. Since this is a two dimensional pattern, each order must be classified in both *x* and *y* directions with the centre of the pattern representing the origin. Both patterns are specifically designed for an *N*=8 system.

The pattern of orders is classified depending on spatial position where the *x* axis is the horizontal axis viewed at the detector array and *y* the vertical. The origin of both axes is found at the centre of the pattern and would be the position at which the VCSEL would be imaged if the DOE were to be removed.

To minimise reconstruction error Δr, a binary process was used to fabricate the DOE. Figure 77(a) shows the e-beam mask used to etch the DOE on a fused silica substrate where the reactive ion used for etching was CHF_3. This process produces sharp vertical etches without undercutting and a uniformity of etch depth, and correspondingly phase, of 0.5% to 1% over the entire 25mm element. However, as the feature size in this element was approaching the minimum resolvable, rounding was observed in the fabricated element as seen in Figure 77(b).

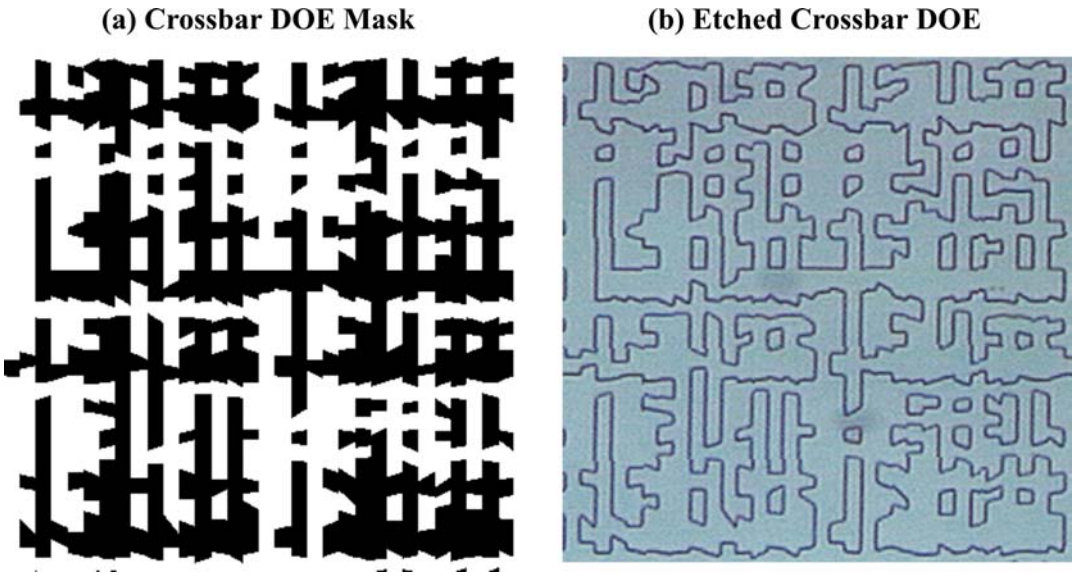

Figure 77: Diffractive Optic Element Phase Profile
Part (a) shows the theoretical phase profile where white represents no phase shift and black a 90º phase shift. Part (b) is an image of the fabricated element where the dark blue lines represent the changing phase shift.

If this system is correctly aligned it should exhibit the behaviour shown in Figure 78. Theoretically, crosses created by 4 VCSELs incident on a DOE should overlap perfectly with no spots lying on the grid lines between detectors.

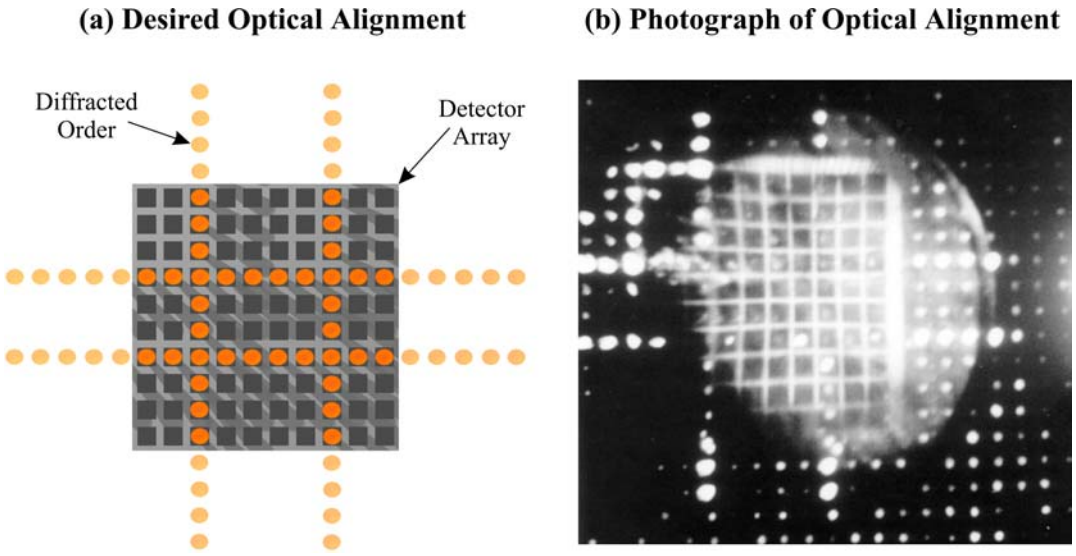

Figure 78: Optical System Alignment
Part (a) illustrates four VCSEL channels diffracted onto the detector array. Part (b) shows this pattern implemented in the aligned system. Although image quality is poor, the pattern can be distinctly seen.

The DOE's position was found to be extremely sensitive to change, including rotation. Movement away from lens 1 resulted in an increasing number of visible orders between

two different VCSELs' zero order positions. Rotation of the DOE on any axis will also alter the projected image. Probably the most useful is in plane rotation which rotates the interconnect pattern projected by a VCSEL around its own origin. Such movement does not rotate the position of the origin in relation to other origins.

After fabrication, DOE transmission efficiency T_{DOE} and array non-uniformity Δr were measured and can be seen in Table 4. Previous calculations indicate that the non-uniformity seen in the crossbar DOE is sufficient only for systems of $N<7$. Its use in the current system does not prevent operation but will compromise solution optimality.

DOE	T_{DOE} (%)	Δr (%)
Crossbar Switch	60.0	4.7
Banyan Switch	60.0	2.9

Table 4: First Generation DOE Efficiency and Non-Uniformity
Non-uniformities across both elements are reasonable but not sufficient in the crossbar element to guarantee optimality.

The power in each spot on both *x* and *y* axes was profiled for the crossbar DOE to help assess the magnitude of any potential problem and can be seen in Figure 79.

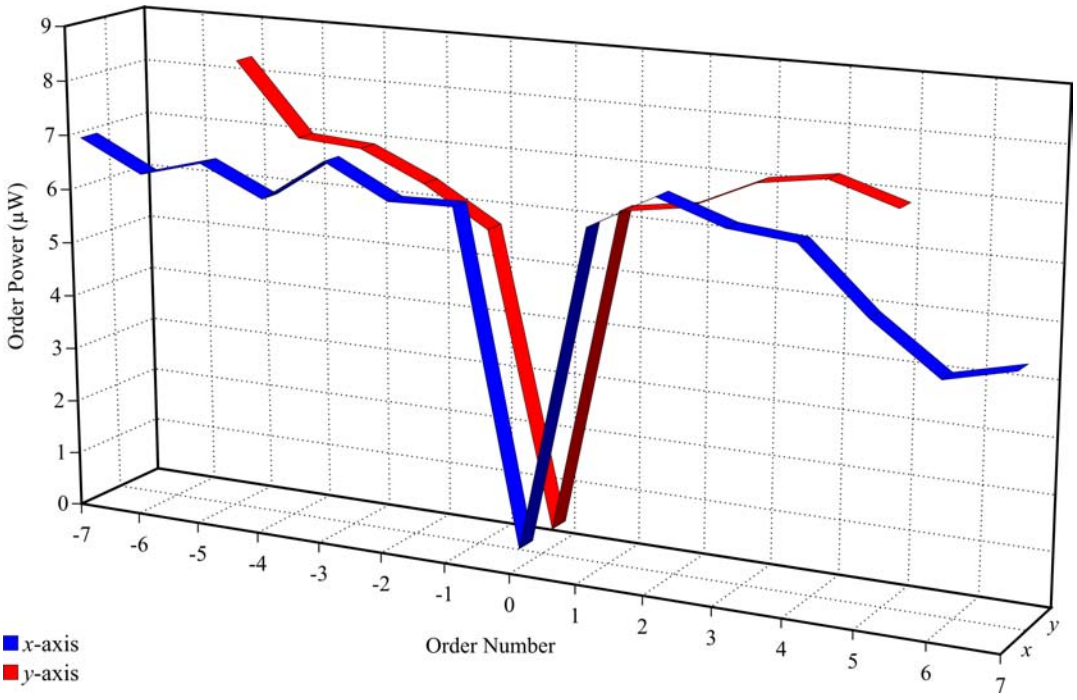

Figure 79: First Generation Diffractive Optic Element Profile
Measured optical power in each spot given a total input power of 300μW ±25μW. Measurements are normalised against the zero order and averaged over six sample cases.

This graph shows an average value. It can be clearly seen that most of the orders are reasonably stable at around 20 times the magnitude of the zero order, except for in the horizontal x direction where x=5 and x=6 prove consistently low. This has been traced to a fabrication error in this version of the DOE. Unfortunately, these measurements are subject to many sources of error ranging from imprecision in VCSEL and driver output to detector and transimpedance amplifier non-linearity. The errors were reduced by sampling using a different set of devices each time, normalising the results with the zero order and then averaging. The combined error is approximately ±0.7μW per spot.

The photodiode detector array used in both first and second generation neural network demonstrators was a Centronic MD100-5T, as shown in Figure 80, and consisted of an array of 10×10 individually addressable photodiodes. The first generation demonstrator did not use every element in the array so only the central 6 rows and 8 columns were actually employed.

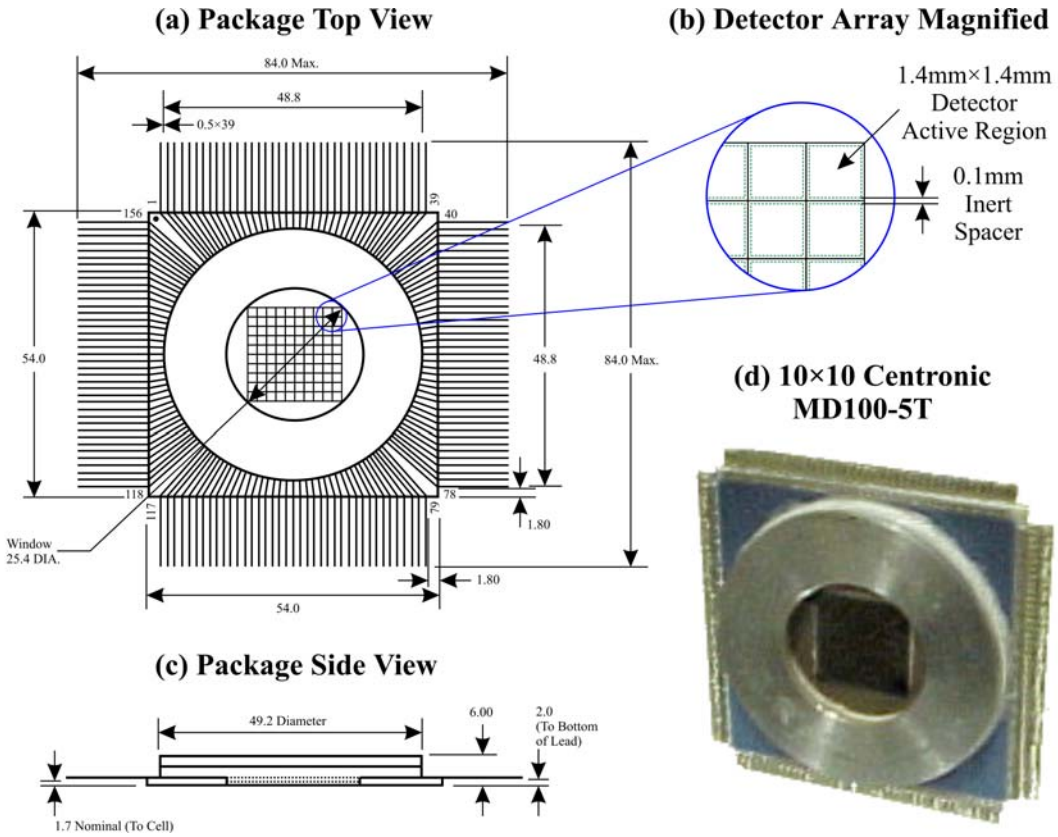

Figure 80: 10×10 Centronic MD100-5T Photodiode Array
All dimensions indicated are in millimetres. Each photodiode is spaced from any other adjacent photodiode by a inactive region of 0.1mm as shown in part (b).

The centre of each neighbouring photodiode is spaced by 1.5mm, 0.1mm of which is inactive and used to distinguish adjacent detectors. The detectors are relatively large

which could potentially cause capacitive problems thus limiting the overall speed of operation, in this case to 100MHz. However, the ease of alignment brought about by using larger detectors more than compensates for this rather generous limitation given that the system is primarily a demonstrator. Table 5 shows the characteristics of the detector array with responsivity over its entire spectrum in Figure 81. The operating wavelengths of both first λ_{1g} and second λ_{2g} generation systems are indicated.

Parameter	Value	Parameter	Value
Elements	10×10	NEP at 436nm ($WHz^{-\frac{1}{2}}$)	4.15×10^{-14}
Width (mm)	1.4	Length (mm)	1.4
Area (mm^2)	1.96	Separation (mm)	0.1
Typ. Shunt Res. (MΩ)	400	Min. Shunt Res. (MΩ)	1
Max. Unbiased Cap. (pF)	55	Max. 12V Biased Cap. (pF)	12
Max. Dark Current (nA)	200	Min. Dark Current (nA)	1
Max. Resp. 760nm (AW^{-1})	0.455	Min. Resp. 760nm (AW^{-1})	0.430
Max. Resp. 956nm (AW^{-1})	0.360	Min. Resp. 956nm (AW^{-1})	0.310

Table 5: Centronic MD100-5T Photodiode Array Characteristics

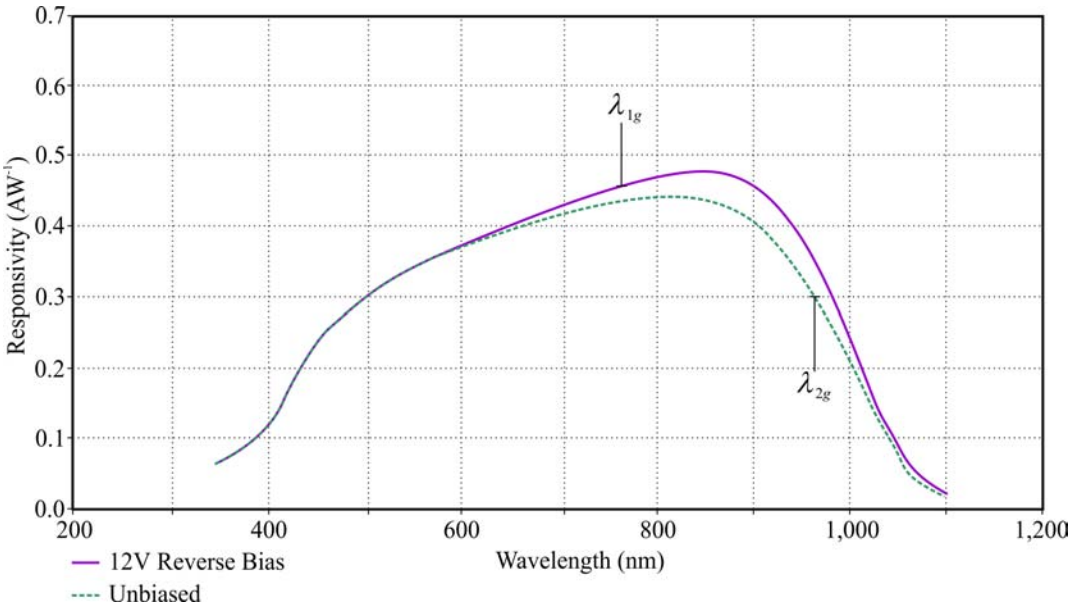

Figure 81: Centronic MD100-5T Spectral Response
Reverse biasing improves responsivity in the NIR range. The operating wavelengths with respect to configuration are indicated for both first and second generation demonstrators.

The detector array is made of Si which has a bandgap of 1.12eV, thus only responding to wavelengths of ~1.1μm or less. It is enhanced to operate well at the blue end of the spectrum but this does not preclude operation at NIR wavelengths. Reverse biasing the photodiode slightly improves NIR responsivity, a technique used in the first generation demonstrator but not the second.

A diffracted image of the VCSEL array is focussed at a specific distance from lens 2 and this must also be the position of the detector array. Movement of lens 2 with relation to lens 1 allows alteration of the image size on the detector array.

5.1.2 Electronic System

In the first implementation, each of the 48 neurons has an input photodiode detector followed by a capacitor-coupled inverting amplifier chain and low-pass filter, the output of which drives a VCSEL. Figure 82 shows the electronic modular layout and Table 6 the associated component values.

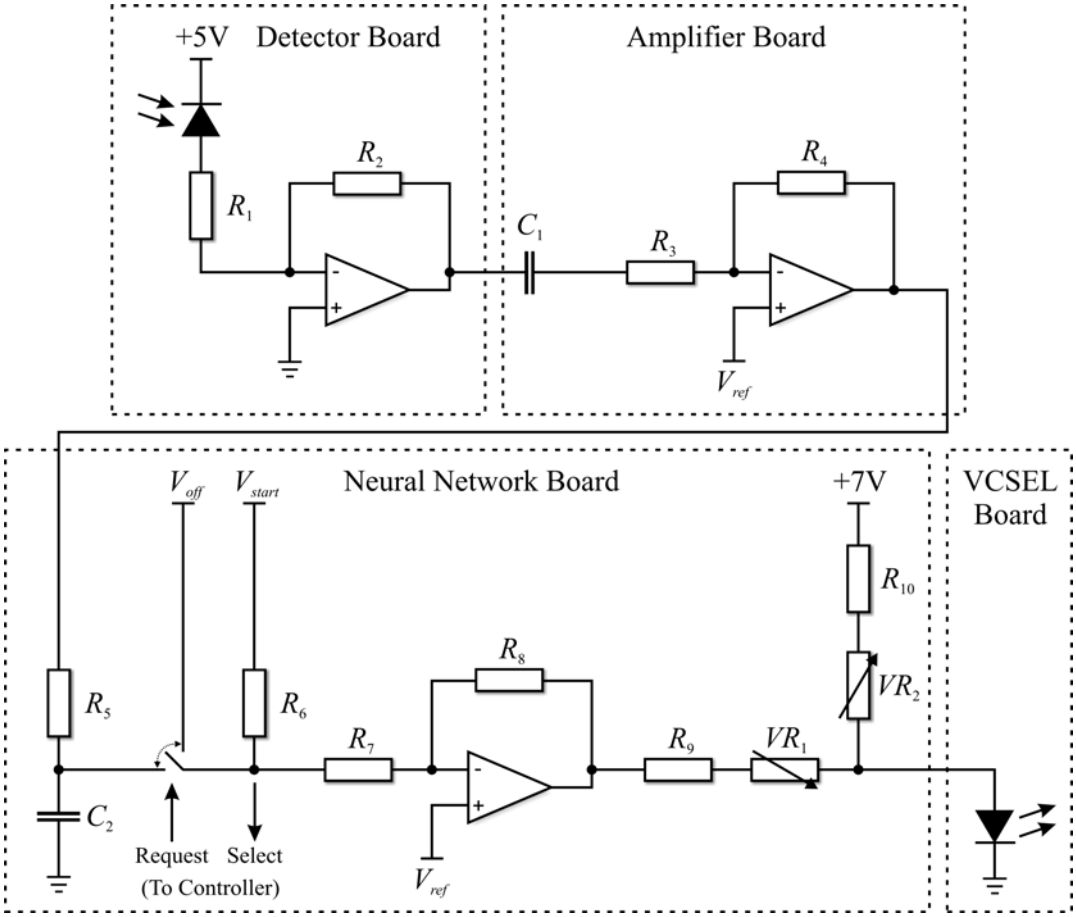

Figure 82: First Generation Electronic System Circuit Diagram
Electronic layout of a single neural channel, 48 of which exist in the completed system.

Component	Value	Component	Value
Resistor R_1	100Ω	Resistor R_2	100kΩ
Resistor R_3	470Ω	Resistor R_4	1kΩ
Resistor R_5	3.3kΩ	Resistor R_6	100kΩ
Resistor R_7	100kΩ	Resistor R_8	100kΩ
Resistor R_9	470Ω	Resistor R_{10}	1kΩ
Variable Resistor VR_1	500Ω	Variable Resistor VR_2	1kΩ
Capacitor C_1	47pF	Capacitor C_2	10nF

Table 6: First Generation System Component Values
Component types and values specifically for Figure 82.

The electronic system is divided up over different boards with images of the two most important shown in Figure 83. The first stage is the *amplifier board* which is designed to amplify the detector input and integrates a high pass filter to remove any DC component. The second stage is the *neural network board* which provides neural functionality in the system.

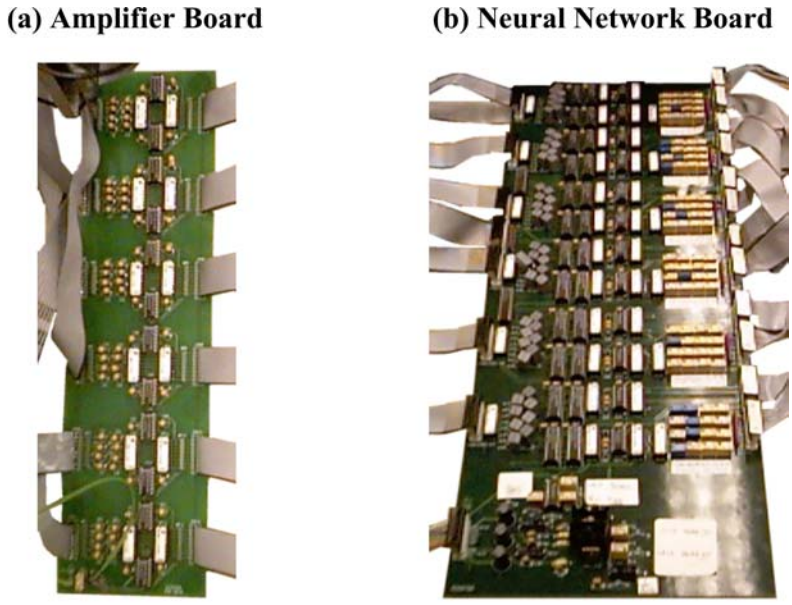

Figure 83: First Generation Electronic System Images
Images of discrete component circuit boards, each board supporting 48 neural channels. Left is the amplifier board, right the neural network.

Before testing could commence, it was also necessary to calibrate the neural network board. However, the first task was to set up the correct reference voltages as shown in Table 7. These voltages had been previously calculated in [138] and promote evolution as predicted during simulation.

Reference Voltage	Value	Error
V_{start}	5.01V	±0.001V
V_{ref}	0.75V	±0.150V
V_{off}	3.92V	±0.001V

Table 7: Reference Voltages
These reference voltages define neuron behaviour. Incorrect values can compromise solution quality or prevent network convergence altogether.

The next step was to calibrate all VCSELs using the available optical output power versus drive current data. A square wave with a frequency of 0.5Hz was applied to the channel under adjustment so that the full range of neuron input voltages was swept. By observing the drive current, minimum and maximum values were determined thus allowing adjustment of the variable resistors VR_1 and VR_2 to give an appropriate output power. These powers were deemed to be 0.05mW representing an off state and 0.8mW designating on. This method of testing also had the advantage that the electrical circuitry for each neuron was simultaneously tested. Before VCSEL calibration, it was ensured that both variable resistors were at absolute minimum power out. Even though such precautions were taken, damage to previously working VCSELs became apparent during the calibration process. This was traced to both a hardware fault and the idling of the neural network card at -5V when disconnected from the amplifier board. Application of a negative potential to a VCSEL can cause permanent damage or complete failure if it exceeds $\sim -2\text{V}$. Avoidance of this situation prevented further damage to any more VCSELs.

Calibration revealed that ten VCSEL elements had failed along with two detectors. Channels with failed VCSELs were set to minimum current drain so that unnecessary power was not drawn. During testing, the channels with failed VCSELs or failed detectors were not used. The reason for detector failure is not clear, however the array was integrated with amplification circuitry which was believed to be the problem.

Three other channels had malfunctioning components which were located and replaced allowing channel recalibration.

Optical alignment was again checked to ensure accuracy. It was found that the total optical power present was too high, but instead of recalibrating every part of the system a beam splitter was inserted. Useable channels were chosen at random and their output examined. It rapidly became clear that certain neurons exhibited unexpected priority over others. Re-examination of the system flagged that the VCSELs were not correctly calibrated and switching on certain VCSELs induced a photocurrent twice that of others. This threw the accuracy of the initial calibration data into question. The optical power from a random set of VCSELs was again measured and found to exhibit considerable variance. Two sample measurements taken showed that one VCSEL produced 1.29mW while another produced 2.19mW. Both of these channels were supposedly calibrated at 0.8mW and such powers were beyond safe operational limits.

It was at this point that VCSEL sensitivity to ambient temperature was discovered, as shown previously in Figure 75. The calibration data used was taken in a low ambient temperature environment however the high ambient temperature now present meant that the array had to be re-profiled and recalibrated before any further results were taken.

During testing it became apparent that V_{ref} plays a critical role in network convergence. Its value had to be adjusted to that shown in Table 7 and had to remain within the tolerance shown as the error. Due to issues with component tolerances, namely the DOE, no universal value was found that provided a completely optimal solution under all circumstances. Fine adjustment was required occasionally to improve solution quality.

5.1.3 Results

After recalibration of the VCSELs, a set of fully operational channels was chosen which had similar characteristics. Six request matrices were generated randomly, as shown by the next six equations, and results obtained by allowing the system to evolve to a steady state.

$$\text{Request } 1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$ **Equation 62**

$$\text{Request } 2 = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$ **Equation 63**

$$\text{Request } 3 = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$ **Equation 64**

$$\text{Request } 4 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$ **Equation 65**

$$\text{Request } 5 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$ **Equation 66**

$$\text{Request } 6 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$ **Equation 67**

Each request matrix was run 10,000 times with the scheduler never once producing an invalid switch configuration. The results for the crossbar switch can be seen in Figure 84. DOE non-uniformity was not sufficient to ensure continually optimal results and indeed the results for trial run request matrices 1 to 4 are close to those predicted in Figure 56. For trial run request matrices 5 and 6 almost all results proved to be optimal. This indicated that these matrices have a distinct solution thus limiting the effects of non-uniformity.

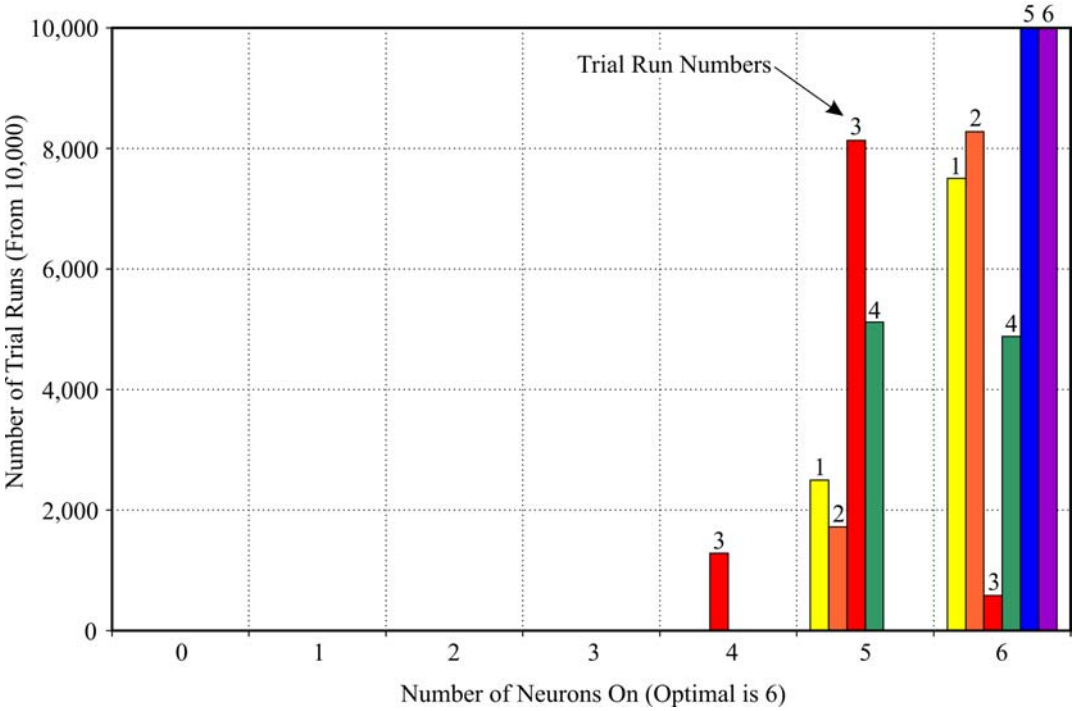

Figure 84: Crossbar Switch Results

Optimal number of neurons on in all trial run cases is 6.

In the Banyan switch controller, DOE non-uniformity was adequate and provided the results shown in Figure 85. Trials 2, 4 and 5 ran optimally the vast majority of the time, however, trial 3 continually settled on 5 neurons rather than the optimal 6. This behaviour indicates the presence of a local minima suggesting that overall system noise was not sufficient. Note that system noise should not be confused with non-uniformity as noise is an evenly distributed time random factor rather than a constant difference in a particular neuron's characteristics.

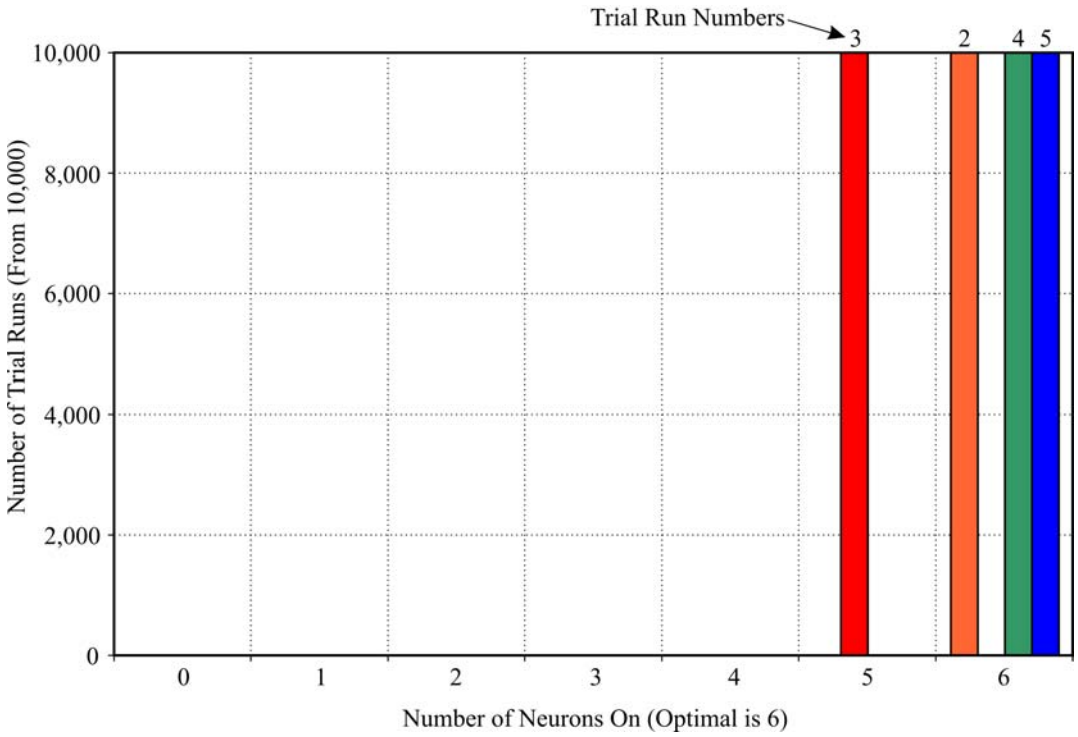

Figure 85: Banyan Switch Results
Optimal number of neurons on in all trial run cases is 6.

No attempt has been made to optimise demonstrator performance. Decision times are determined by the time constant of the low-pass filter t_{lpf}:

$$t_{lpf} = R_5 C_2 \qquad \textbf{Equation 68}$$

which in this case was 33µs. Through reduction of this time constant it should be possible to obtain a decision in tens of nanoseconds using only off-the-shelf electronics, thus achieving scheduling decisions at a rate compatible with the latest router requirements.

5.1.4 Conclusion

The first generation system successfully demonstrated the feasibility of an optoelectronic neural network, however any subsequent design will require a redesign of the electronic system. Issues in the electronic system were not helped by optical component non-linearities in VCSEL, DOE and detector arrays. There was insufficient configurability in the demonstrator to compensate adequately. Nevertheless, this system showed potential in its implemented form but the inability to support prioritisation due to a discrete controller, essential in any modern day packet switched network, will prove debilitating.

5.2 Neural Network Hardware Analysis

The first generation demonstrator exhibited a number of problems which are solvable if thought is given to the next generation demonstrator. Any subsequent system should incorporate sufficient flexibility to solve other assignment problems by allowing rapid reconfiguration and enable optical component calibration to improve uniformity. Prioritisation support is essential and a performance on a par with both commercial and research neural networks is mandatory. Bearing these points in mind, this section defines neural network performance metrics, evaluates existing neural networks and compares and contrasts them against the second generation neural network. It concludes by assessing implementation strategies for the next demonstrator.

5.2.1 Neural System Classification

There are many ways of classifying a neural network be it through architecture, network type, the number of external inputs and outputs it has or its number of neurons. However, to evaluate performance this thesis will use the accepted metric of *connections per second* (CPS) as defined by Holler in 1991 [141].

A *connection* is defined as the calculation of the product of a synaptic input and its associated weight. It is the basic unit of computation in a neural network and the number of connections executed every second is a measurement of its performance. Thus CPS is directly related to how fast a network can perform mappings from inputs to outputs. This rating does not include the calculation time incurred during neuron summation and consequent application of the transfer function as both of these must have occurred before any other connections can take place. Such performance metrics are useful in that they set a target for the demonstrator.

Hardware neural networks tend to be classified into one of three categories: *digital*, *analogue* or *hybrid*. A digital neural network is a complete digitalisation of the system with all internal parameters stored and calculated digitally. Although digital summation can be relatively slow, especially with regard to synapses, it is an extremely flexible technology with mature fabrication processes. Table 8 summarises a few digital neural networks and quotes their performance.

Digital Network	Architecture	Neurons	Synapses	Ref.	CPS
NeuraLogix NLX-420	FF, ML	16	Off-Chip	[142]	300
HNC 100-NAP	SIMD, FP	100 PE	512K Off-Chip	-	250×10^6
Hitachi WSI	SIMD, Hopfield	576	32K	[143]	138×10^6
Inova N64000	SIMD, Int.	64 PE	128K	[144]-[145]	870×10^6
MCE MT19003	FF, ML	8	Off-Chip	[146]	32×10^6
Micro Devices MD	FF, ML	1 PE	8	[147]	8.9×10^6
Philips Lneuro-1	FF, ML	16 PE	64	[148]	26×10^6
Siemens MA-16	Matrix Ops.	16 PE	256	[149]-[150]	400×10^6

Table 8: Digital Neural Networks
Examples of digital neural network hardware as of December 1998. Abbreviations can be found in the glossary on page 180.

The second type of neural network hardware is analogue which exploits physical properties to perform operations thereby obtaining high performance and integration densities. For example, in a current based analogue system a common electrical line could sum the currents from several synapses to give the neuron's input. However, the major problem with such systems tends to be component tolerances as they can be nigh impossible to compensate for during manufacture. Such problems were noticeable in the first generation neural network demonstrator. For this reason analogue neural networks are a rare breed with only one network shown in Table 9.

Analogue Network	Architecture	Neurons	Synapses	Ref.	CPS
Intel ETANN	FF, ML	64	10,280	[151]	2×10^9

Table 9: Analogue Neural Networks
Examples of analogue neural network hardware as of December 1998. Abbreviations can be found in the glossary on page 180.

The hybrid implementation was created by taking the best characteristics from both analogue and digital systems. It combines the summation potential of analogue with the noise resistance of digital. Table 10 shows a few hybrid neural networks and quotes their performance.

Hybrid Network	Architecture	Neurons	Synapses	Ref.	CPS
AT&T ANNA	FF, ML	16-256	4,096	[152]	2.1×10^9
Bellcore CLNN-32	Boltzmann	32	992	[153]	100×10^6
Mesa Research Neuroclassifier	FF, ML	6	426	[154]	21×10^9
Ricoh RN-200	FF, ML	16	256	[155]	3×10^9

Table 10: Hybrid Neural Networks
Examples of hybrid neural network hardware as of December 1998. Abbreviations can be found in the glossary on page 180.

Hybrid systems achieve the highest level of performance among all types of hardware neural networks. Systems with a performance of 21×10^9 CPS have been demonstrated by Mesa Research [154]. The second generation neural network demonstrator described in this thesis adopts a hybrid approach.

It is also possible to implement neural networks using alternative methods. This usually involves a generic processor such as a Transputer, an Intel i860 or a DSP which simulates the network and its interconnects purely in software. However, such systems are not examined here as they are not strictly hardware neural networks, rather software ones.

5.2.2 System Performance

To compare the performance of the crossbar switch neural network to that of existing networks, its CPS rating must be calculated with relation to iteration frequency and network size. Given that sufficient hardware is available to correctly interconnect the system, we can define its connection density C_{dn} in terms of the number of inputs m and outputs n:

$$C_{dn} = mn(m+n-2) \qquad \textbf{Equation 69}$$

The resulting number of connections with respect to number of inputs and outputs is graphed in Figure 86.

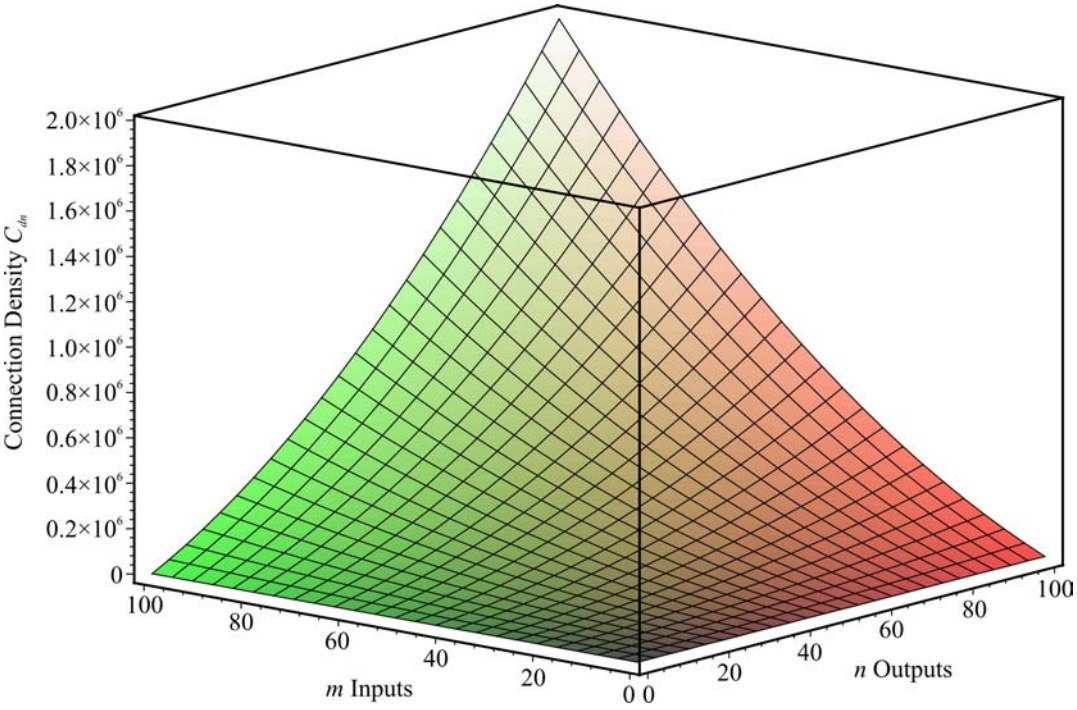

Figure 86: Neural Network Connection Density
Growth in network interconnectivity of the crossbar switch neural network scheduler.

Once the number of connections have been calculated, the neural network's CPS rating can be determined given frequency of iterations f_{it} in Hz. Note that this value is not directly related to solutions per second as network convergence requires multiple iterations. The following relationship can be derived for a square network where N=m=n:

$$CPS = f_{it} N^2 (2N - 2) \qquad \textbf{Equation 70}$$

This can be graphed as shown in Figure 87.

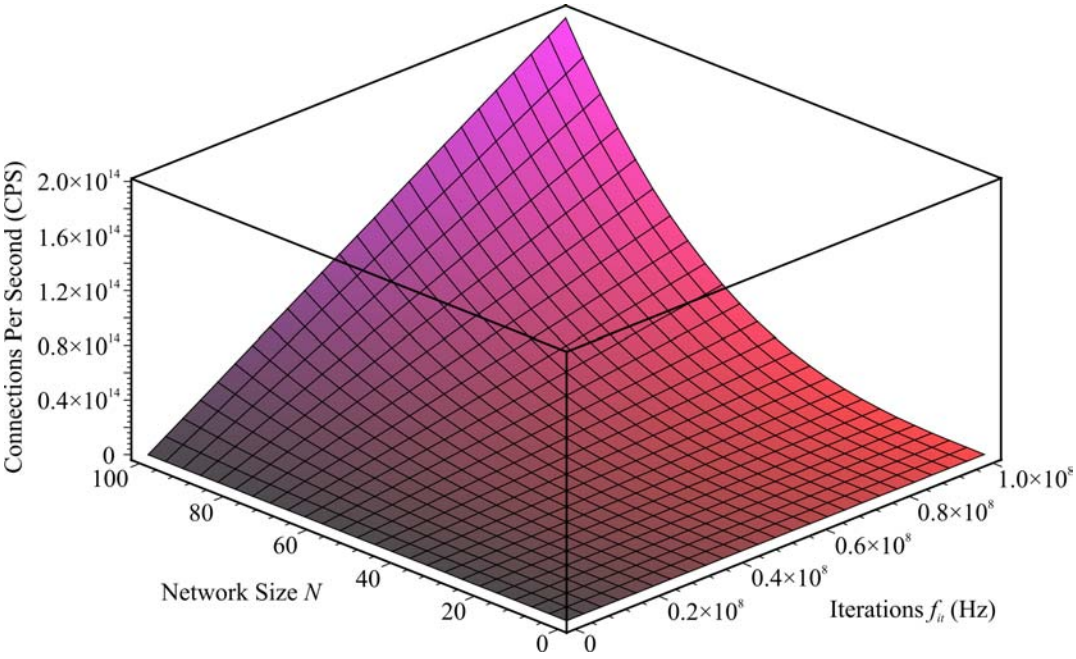

Figure 87: Neural Network Performance

The CPS rating increases quadratically with network size N. This increase is linear when related to iterations per second f_{it}.

A square neural network enables optimum use of the crossbar switch since a full set of inputs m can be simultaneously connected to a full set of outputs n, assuming $m>n$. If this were not the case, then a situation would inevitably arise where certain inputs cannot be connected to any output as all existing lines are busy.

These calculations allow determination of the performance of the second generation neural network demonstrator. At the time of writing, the network is of order N=8, however the iteration frequency f_{it} was not determined in the initial specification. It is hoped that values from 100kHz up to perhaps 2.0MHz would be possible in the first instance giving respective ratings of 89.6×10^6 CPS and 1.08×10^9 CPS. Although not as fast as some neural systems in section 5.2.1, the comparison is favourable and indicates a relatively high performance.

However, the crossbar switch interconnect is only partial and if the interconnect pattern were altered to accommodate a fully interconnected Hopfield network [121] then the CPS rating could be determined using:

$$CPS = f_{it}\left(N^4 - N^2\right) \qquad \textbf{Equation 71}$$

This would give a performance of 403×10^6 CPS at 100kHz and 8.06×10^9 CPS at 2.0MHz. Given such low operational speeds and only 64 neurons, these performance ratings carry considerable weight.

5.2.3 Second Generation Implementation

Since there is a predefined optical solution, the only unspecified aspect of the demonstrator is the electronic neural network. This section compares and contrasts the aspects of implementation in either a digital or analogue manner, thus highlighting the design decisions behind the second generation neural network demonstrator.

Implementation using solely analogue electronics would mean that the entire system would be classified as an analogue one, as was the first generation. The simplest, and cheapest, solution is to use an operational amplifier chain to act as neurons. Such an analogue implementation is advantageous as digital systems require a higher transistor count to perform the equivalent analogue function. In addition, analogue systems usually offer better performance at lower cost for approximately the same functionality. However, component tolerances are critical and have already proven to be a problem in the first generation demonstrator. Network convergence is highly dependent on the analogue component values used, thus correct hardware design becomes tricky. This also results in a lack of flexibility. Once designed and built, it is hard to alter any system parameters. Nevertheless, this problem could be circumnavigated through the use of components such as the *electrically programmable analogue circuit* (EPAC).

Since there is already an analogue component in the form of the optical system, adding any digital hardware would change its classification to hybrid. Although integration of digital components may sound out of place, the benefits they bring far outweigh the drawbacks. Unfortunately, the problem with any digital neural implementation is conversion from analogue to digital at the input and digital to analogue at the output. This can not only be slow but can require expensive components.

After careful consideration, it was decided that a microprocessor solution was promising for a range of reasons. These include that since such systems are usually a plug-in solution, electronic design can be kept to a minimum. Their flexibility also allows a neuron's activation function to be reconfigured to anything that can be programmed. In addition, a microprocessor is capable of judging when the network has converged, therefore the result can be output when the system is finished rather than after a predefined period of time that could be either insufficient or excessive. Look-up tables can be used for system calibration allowing the microprocessor to adjust a VCSEL's output until it reached a predefined level on the photodiode. This would prevent saturation of the photodiode and enable active calibration. In a similar manner,

alignment also becomes active as a microprocessor could examine the light intensities falling on a range of detectors to ensure that light is only present on nominated active channels and not on neighbouring ones. For evaluation purposes, a microprocessor could replicate characteristics from another proposed implementation or simulate a fault to assess tolerance in the current system. Finally, workload could be divided across multiple processors to allow ease of scaling, an issue eased by the fact that the number of calculations per iteration is directly proportional to the number of neurons.

However, a microprocessor solution does have its drawbacks. Digital to analogue conversion can be slow without expensive hardware, but this would not be an issue in the hardware minimised neural network as the neural outputs are digitally thresholded. Multiple processors would also be required to prevent bottlenecking. Unfortunately, multiple processors increase both hardware size and cost, thereby rapidly limiting any implementation. However, this problem could be circumvented by constructing a SIMD type architecture.

Interfacing of the neural network must also be carefully considered. There are three interface points that must be examined regardless of system design.

Firstly, there is the interface between the detector array and electronic neural network. If an analogue neural network is implemented then this is not an issue. However, if it is digital then analogue to digital conversion of N^2 channels will be required. This can be performed using multiplexed components but would result in serial processing of parallel data.

Secondly, between the electronic neural network and its controller, such as a PC. Presuming that any system built will not just support binary requests, an analogue electronic neural network requires digital to analogue conversion from the PC and potentially during the return of solutions. A digital network would minimise this problem with microprocessors capable of direct connection to a PC's communications ports.

Third and finally, there is the interface from the electronic neural network to the VCSEL array. All 64 VCSELs need to be driven by the neural network. An analogue network would not prove difficult to interface but, again, a digital network would. Digital to analogue conversion would clearly be required, which can be slow and could result in the multiplexing of several channels. However, minimisation of the neural

network algorithm has verified that in this case direct digital driving of the VCSELs provides not only a viable but beneficial alternative interface.

Implementing an analogue solution at this stage would have been taking a leap towards a project goal without necessarily understanding the problem completely. Since it is a demonstrator that was being designed, flexibility is of the utmost importance. This encompasses the ability to alter network configuration, activation functions and to compensate for component tolerances. As an analogue demonstrator had already been successfully constructed, it was concluded that a hybrid system using microprocessor hardware should be next. Simulation has already shown that such a system with appropriate hardware minimisation enables enhanced network stability and scalability, a prospect that must be examined fully.

5.3 Second Generation Neural Network

The second generation demonstrator attempted to enhance on the first by reducing system size, curtailing non-linearity and improving performance. Its operation is based around hardware minimisation as outlined in chapter 4 which removes the digital to analogue output conversion stage. Functionality is also improved through the addition of packet prioritisation. The optical systems in both generations remain virtually the same, however the electronic system in the second generation has been subjected to a complete overhaul.

5.3.1 System Overview

To aid understanding it is pertinent to first examine the system as a whole. Subsequent sections break the demonstrator down into its component modules while detailing both their construction and performance. This process follows data flow through the demonstrator with results presented as appropriate either during the description or in the results section at the end.

An image of the completed system can be seen in Figure 88. Note that dust and RF interference shields have been removed. Surface mount technology has been used throughout to reduce system size.

At the beginning of a computation, a set of request values are downloaded from a PC to four TMS320C5x *digital signal processors* (DSPs) using serial RS232 ports. A reusable software module called DSK Controller V2.00 was written to provide communications under Microsoft Windows™ 9X/ME. Information interchange is

possible between a single PC and up to four DSPs simultaneously. The DSPs can be remotely reprogrammed and blocks of data of a user defined size can be transparently uploaded or download between any memory addresses in both DSP and PC. All commands can be executed at up to 115k baud per DSP.

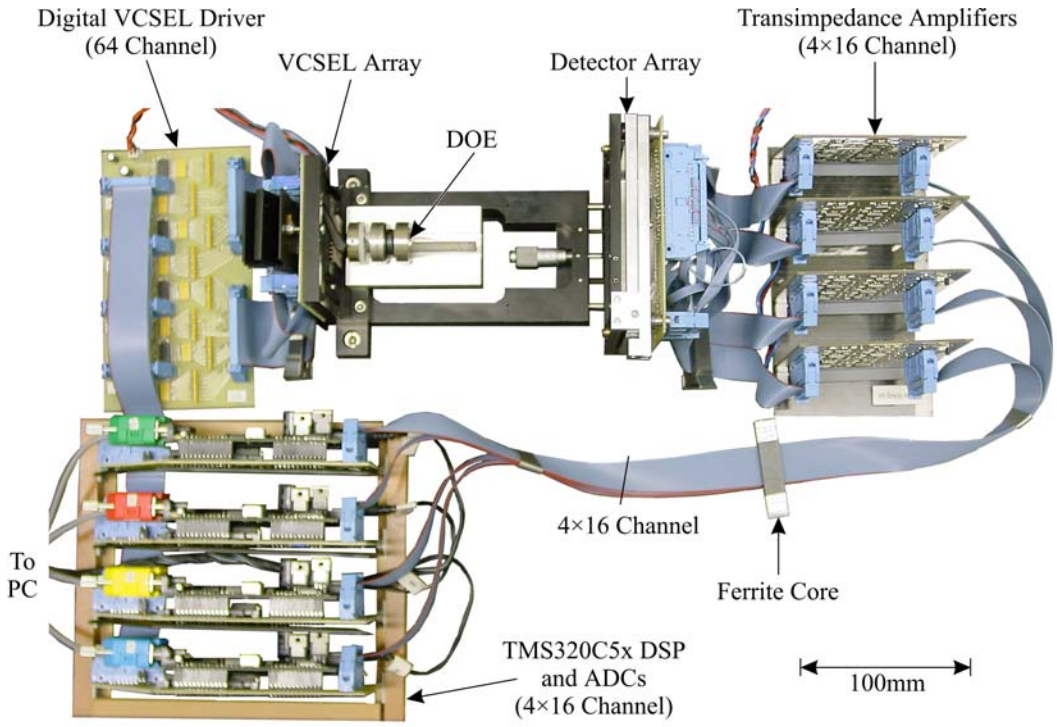

Figure 88: Second Generation System Overview
This is an *N*=8 or 64 neuron optoelectronic network. The ferrite cores suppress RF interference.

In this implementation, each DSP handles 16 neurons and thus receives one quarter of the requests. They perform a transfer function based on these requests and use the result to switch a neuron's associated VCSEL on or off via the digital VCSEL driver. An interesting point for any future development is that the DSPs perform the same instruction on sixteen different data channels. This mode of operation is analogous to that of a SIMD processor. Such processing would undoubtedly be preferable to the current architecture which partially serialises parallel data in an attempt to reduce hardware complexity and cost.

The digital VCSEL driver module controls the current supplied to the VCSEL array. It is attached to the DSP memory bus and configured by writing a value to a specific memory location. The values stored in the driver do not change until specifically rewritten by the DSP.

Neural network interconnection occurs in the optical domain where the DOE diffracts the VCSEL image to form an interconnection pattern on the detector array. The second generation optical system shows greatly improved uniformity in a package two thirds the size of the first generation.

Amplification of any detected optical signal is performed by 64 dual element transimpedance amplifiers distributed across four modules. The transimpedance amplifiers take a current generated by the photodiodes and convert it into a voltage thus improving its magnitude, bandwidth and linearity all at the same time. They output a voltage range compatible with the ADCs on the DSP module input.

Combined with the DSPs on the same circuit board are two octal ADCs. These simultaneously return any two of sixteen values converted from the detector by accessing a particular memory location. The values read are used by the DSPs in the activation function to determine the next neural network state.

This iterative process continues until a solution has been found to a given problem. Once solved, the DSPs can either signal to the PC that they have finished or continue on to solve another problem. Since the DSPs have internal memory, thousands of problems can be downloaded at a time with completion only signalled once computation is complete.

As with all parallel systems, synchronisation is a serious issue. In this case, no electrical interconnects are necessary since the system is already optically interconnected. An optical signal is broadcast which indicates the readiness of each DSP to begin computation. When all DSPs signal ready, network evolution begins.

5.3.2 Digital Signal Processor Module

This module is designed to interface with a 40MHz TMS320C5x DSP starter kit [156] enabling analogue to digital conversion from 16 different channels through dual multiplexing ADCs and 8-bit digital output on two different channels via the normal DSP address and data buses. A logical overview showing data and control flow is given in Figure 89. The module has been designed around existing components on the DSK board.

Analogue signals are sampled from the detector using two Analog Devices AD7829 8-bit octal flash ADCs [157]. They are connected to a 20 pin input header of which 16 channels are used for analogue detector input and one for the analogue ground *AGND*,

the function of which is controlled by jumper *JP*1. This jumper allows *AGND* to be connected to the DSP module's digital ground *DGND* and when shorted noise suppression in the ADC is improved. When open, the analogue ground is isolated from the digital one.

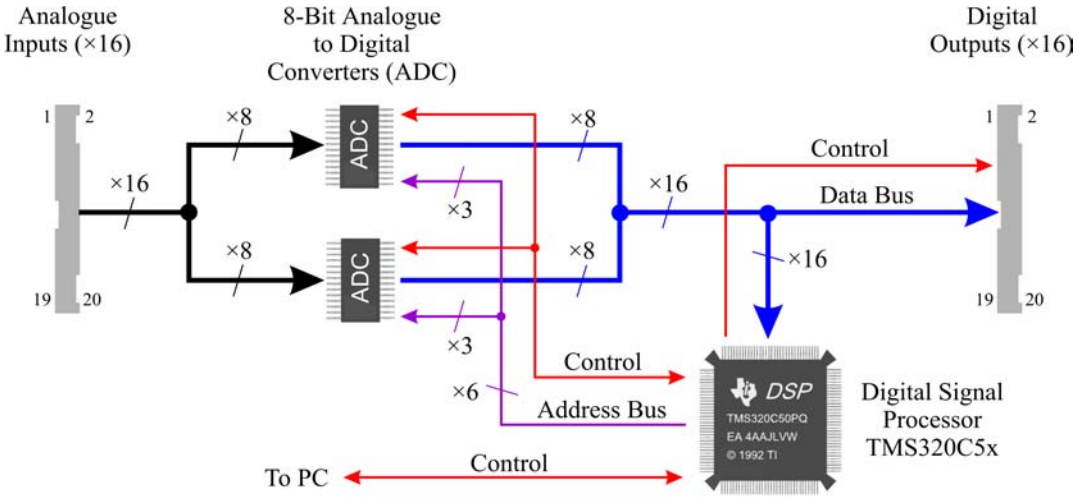

Figure 89: Digital Signal Processor Module Logical Overview
The DSP controls two separate systems, one an analogue input and the other a digital output, both of which are integrated on the DSP address and data buses.

The analogue inputs have a voltage span of 2.5V centred around a specified voltage V_{mid}. To ensure maximum linearity of readings, V_{mid}=2.5V was chosen such that all readings would be within a standard operational amplifier's region of optimal linearity, presuming that the supply voltage V_{cc}=5V.

The 16 input channels V_{in1} to V_{in16} are connected to two eight-channel sample-and-hold ADCs. These are numbered *ADC*1 and *ADC*2 with each responsible for V_{in1} to V_{in8} and V_{in9} to V_{in16} respectively. Both ADCs convert the sampled input to 8-bit values and transfer their information to the DSP data bus on request. *ADC*1 controls the 8 least significant digits on the data bus and *ADC*2 the 8 most significant digits.

Both ADCs should not power down but remain in a high impedance state unless information is requested from I/O space. Sixteen of the DSP's available 64k I/O ports are mapped to data memory using *memory mapped I/O* (MMIO) at address locations 50h-5Fh using address lines *A*0 to *A*4 and *A*6. Therefore, any lines that access these ports should actively be avoided when interfacing with the ADCs.

Controlling the ADCs requires the execution of a specific sequence of actions as outlined in Figure 90. To start conversion of any input, chosen previously during the

read cycle, $\overline{CONVST}$ must be pulled low for at least 20ns. If $\overline{CONVST}$ is not returned to a high state soon afterwards then the ADC will go into power down mode. Conversion will then begin on the selected input channel. To apply the $\overline{CONVST}$ pulse, read from I/O address 100h. The combination of $\overline{IS}$ and $A8$ determine the state of $\overline{CONVST}$ as shown in Table 11. Note that this read command will not return any useable data.

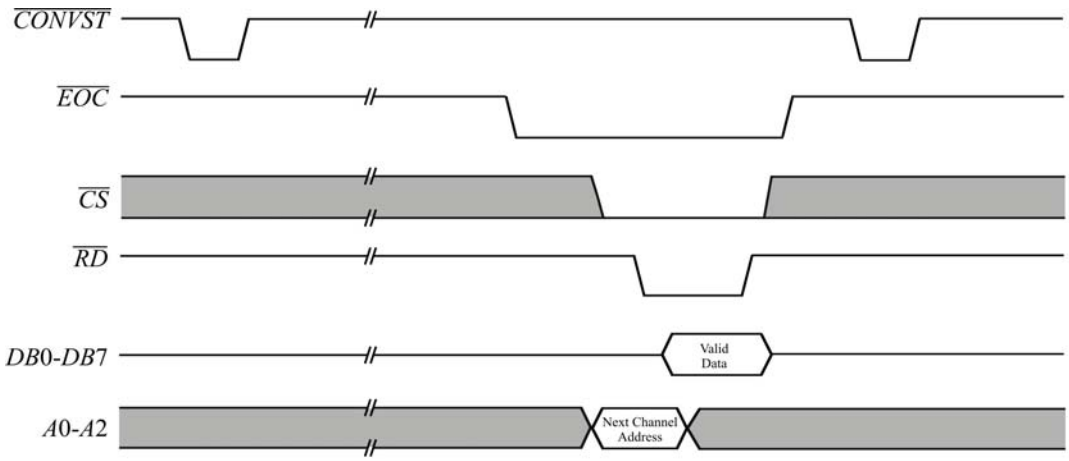

Figure 90: ADC Conversion Flow and Data Readout
A $\overline{CONVST}$ pulse begins conversion. After 420ns the conversion is completed and $\overline{EOC}$ is driven low. The value can then be read from the ADC.

$\overline{IS}$	$A8$	$\overline{A8}$	$P1$	$\overline{CONVST}$
0	0	1	0	1
0	1	0	1	0
1	0	1	0	1
1	1	0	0	1

Table 11: Conversion Start Interface Logic
Address line $A8$ in I/O space is used to control conversion start.

Figure 91 shows the circuitry required to construct the logic table. This system uses only NOR gates so that just one type of logic gate need be used ensuring optimal use of resources on multi-gate components. The device used was a National MM74HC02 quad 2-input NOR gate [158]. It operates switches within 8ns and uses a standard 7402 pin configuration.

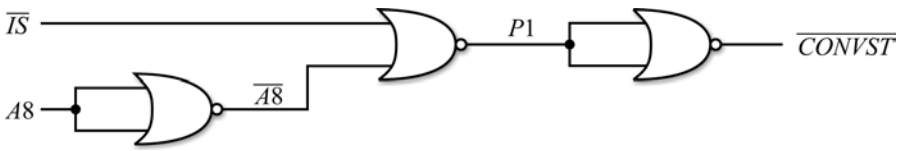

Figure 91: Conversion Start Logic Circuitry
Logic implementation of Table 11. Design converted to only NOR gates.

When conversion is complete $\overline{EOC}$ is pulled low. This signifies the end of conversion and is at most 420ns from when the $\overline{CONVST}$ pulse was applied. The $\overline{EOC}$ signals are used to generate interrupts to the DSP with *ADC*1 connected to $\overline{INT3}$ and *ADC*2 to $\overline{INT4}$. This allows software to be notified when conversion is complete.

Data is read from the ADCs by issuing a read command to I/O space. This enables $\overline{CS}$ line on both ADCs but must be executed with *A*9 high (200h). The interface logic required is examined in Table 12 and logic circuitry shown in Figure 92.

$\overline{IS}$	*A*9	$\overline{A9}$	*P*1	$\overline{CS}$
0	0	1	0	1
0	1	0	1	0
1	0	1	0	1
1	1	0	0	1

Table 12: Chip Select Interface Logic
Address line *A*9 in I/O space is used to control chip select.

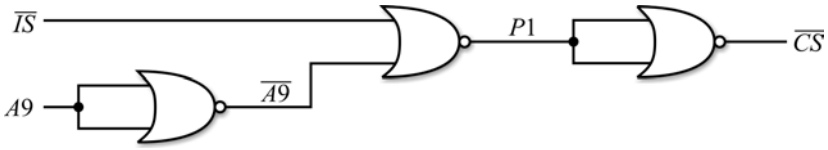

Figure 92: Chip Select Logic Circuitry
Logic implementation of Table 12. Design converted to only NOR gates.

The values applied to an ADC's address lines during read specify the next channel to be sampled on application of the $\overline{CONVST}$ pulse. For *ADC*1, *A*0-*A*2 are mapped to address bus bits *A*10-*A*12 and *ADC*2 lines *A*0-*A*2 mapped to address bus bits *A*13-*A*15. The data returned during a read cycle is a word containing the previous converted value from *ADC*1 in the least significant byte and from *ADC*2 in the most significant byte. To allow a standard read operation to acquire data, both ADC $\overline{RD}$ lines must be directly connected to $\overline{RD}$ on the DSP. Table 13 summarises this address bus usage.

	MSB															LSB
DSP	*A*15	*A*14	*A*13	*A*12	*A*11	*A*10	*A*9	*A*8	*A*7	*A*6	*A*5	*A*4	*A*3	*A*2	*A*1	*A*0
HEX	8000h	4000h	2000h	1000h	800h	400h	200h	100h	80h	40h	20h	10h	8h	4h	2h	1h
ADC	*A*2	*A*1	*A*0	*A*2	*A*1	*A*0	$\overline{CS}$	$\overline{CONVST}$	NC	MMIO	NC	MMIO	MMIO	MMIO	MMIO	MMIO
			*ADC*2			*ADC*1										

Table 13: DSP Address Bus Usage

The table header row contains the DSP address bus lines, the next row the hex value of that bit and the final row to where the line is connected on an ADC. Note that NC means *no connection* and MMIO memory mapped I/O.

To ensure accurate sample tracking of subsequent readings, conversion of the next value must not be initiated until 30ns after the last read operation.

The completed DSP to ADC interface circuitry can be seen in Figure 93.

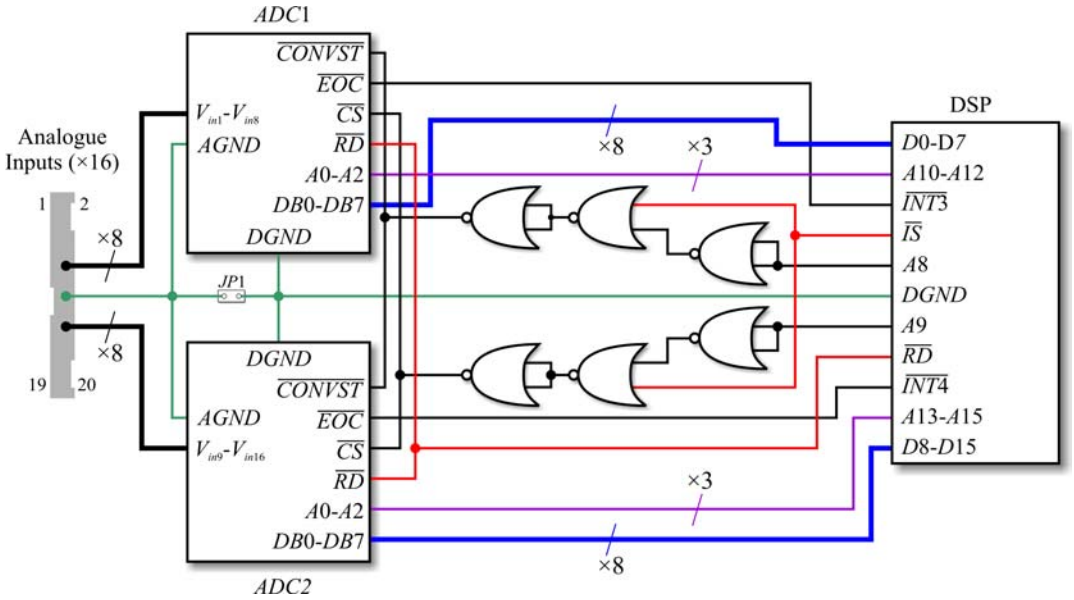

Figure 93: ADC to DSP Interface Circuitry

This diagram shows the additional circuitry necessary to interface the DSP with two ADCs.

Additional circuitry is also required to handle digital output. The system to which information is written is an extension of the entire 16 bit data bus with writes occurring when a latch enable (*LE*) line is driven high. Pins V_{out1} to V_{out16} are TTL digital logic outputs direct from the data bus where V_{out1} corresponds to *D*0 and V_{out16} to *D*15. *DGND* is an optional ground connector that can be used to connect the external latch's ground plane to that of the DSP by shorting *JP*2.

As long as *LE* remains low, the external module will ignore any information on the data bus outputs V_{outx}. When *LE* goes high, the information currently on outputs will be stored in the external latch. *LE* must remain high for a minimum of 20ns. This state

occurs when both $\overline{WE}$ and $\overline{IS}$ lines are low. This results in a NOR logic function as shown Table 14. Thus to activate *LE* we must write to I/O space with an address of 0h, thereby preventing any interference with the ADC interface circuitry.

$\overline{IS}$	$\overline{WE}$	*LE*
0	0	1
0	1	0
1	0	0
1	0	0

Table 14: Latch Write Control
Implemented using a single NOR gate.

Minimal circuitry is required and can be seen in Figure 94.

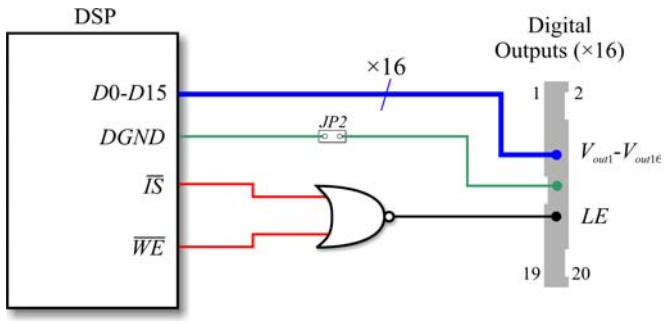

Figure 94: DSP to Digital Output Interface Circuitry
The digital output is an extension of the data bus with a dedicated read control line.

Power consumption of these additional components is critical as the DSK starter kit can only supply 50mA at a V_{cc} of 5V. The necessary additional components are:

- 2×ADC at 12mA maximum each.
- 2×Quad NOR gates at 20µA maximum each.
- 1×ZRA250 V_{mid}=2.5V voltage reference source at 150µA maximum.

This gives a total additional power consumption of 24.19mA which is less than half the maximum available and well within tolerable limits.

A *printed circuit board* (PCB) was designed using Ranger XL to connect the necessary components in the correct manner, the final module shown in Figure 95. This is a two layer PCB where red lines are tracks on the upper layer and blue on the lower.

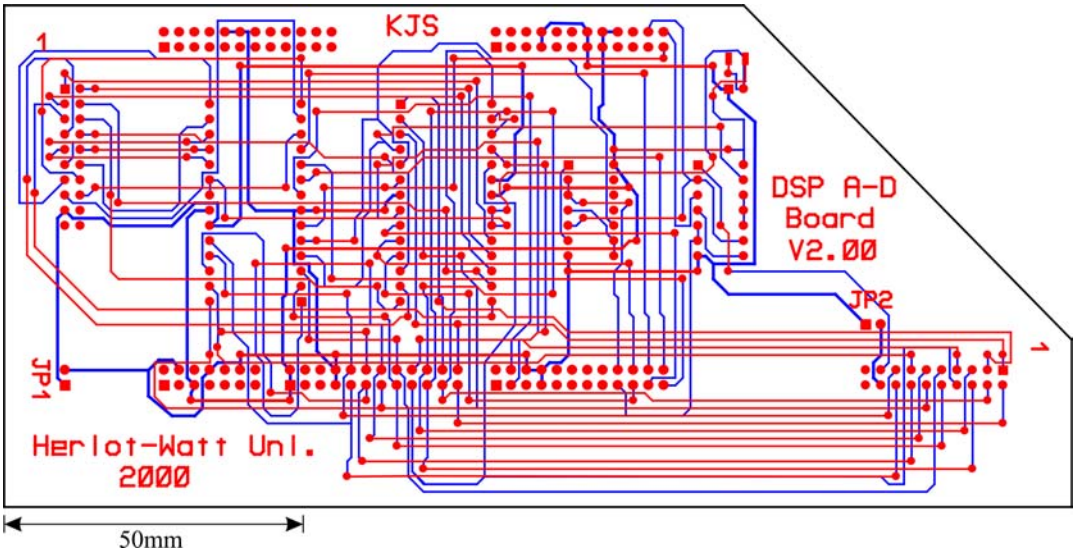

Figure 95: DSP Module PCB Layout
DSP module PCB layout where red is the upper layer and blue the lower.

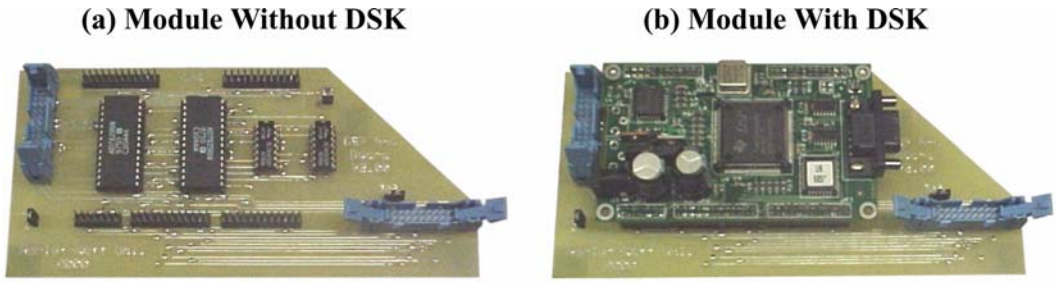

Figure 96: DSP Module Images
Part (a) shows the module without the DSK attached, component designations from left to right being *ADC*2, *ADC*1, *NOR*2 and *NOR*1. Part (b) shows the fitted DSK.

Four DSP modules have been constructed and undergone extensive testing and use. They have proven reliable and their flexibility has sparked interest from third parties as a tool for other experiments.

5.3.3 Digital VCSEL Driver Module

The VCSEL driver module examined here is designed to be directly driven by the DSP. It has 64 channels, divided into four blocks of 16 VCSELs, where each channel can take either an on or off state. The current state is altered by writing a new state to the data bus and pulling the *LE* pin high which is associated with the same block of 16 VCSELs. The bus state is then transferred to the latches and will be maintained until a new state is written.

A current sinking method is used to drive each VCSEL from a standard octal transmission line driver. The driver part chosen is a Texas Instruments SN74HC573AN

octal transparent D-type latch [159] with high current outputs. The VCSEL array has a common cathode configuration so each pin is used to drive the anode as shown in Figure 97.

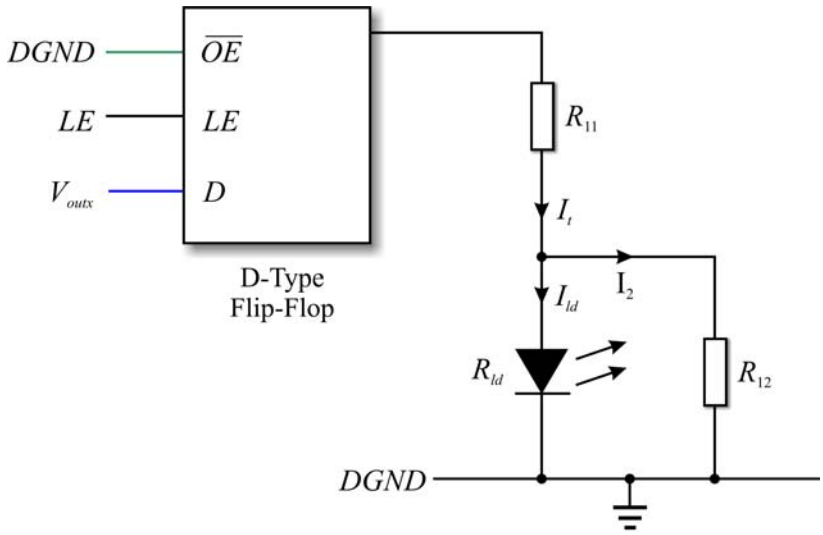

Figure 97: Digital VCSEL Driver Circuitry
64 identical driver circuits exist on the module. The D-type flip-flop holds the current state until the LE signal is applied.

The value of R_{11} was calculated to ensure that the correct current was supplied to the VCSEL. To give a maximum drive current, the operating voltage of the latches was set to 6V. It was also known that the maximum load which could be applied to the driver was 70mA so each VCSEL could get, at absolute maximum, $70/8 = 8.75\,\text{mA}$. Extrapolation of known VCSEL characteristics allowed a standard resistor value to be chosen such that is was not possible to overload the driver. This value was calculated to be $R_{11} = 470\Omega$.

To prevent any stray currents reverse biasing the VCSEL during power up and power down of the module, a protective resistor R_{12} was fitted in parallel with the VCSEL. To prevent invalidating all previous calculations, this resistor needed to have a resistance where $R_{12} >> R_{ld}$. The value of $R_{12} = 100\text{k}\Omega$ was chosen giving minimal interference to the current going through R_{ld}.

Under normal driver board operation the *LE* pin on each of the four neural channels should be set to 0V. This holds the latch values regardless of the data present on the data bus. To set the VCSELs to a different configuration, one sets all the pins to the desired levels of 0V for off or +5V for on and activate the *LE* pin by pulling it high to +5V for at least 17ns. The neuron's new values must remain constant whilst *LE* is high.

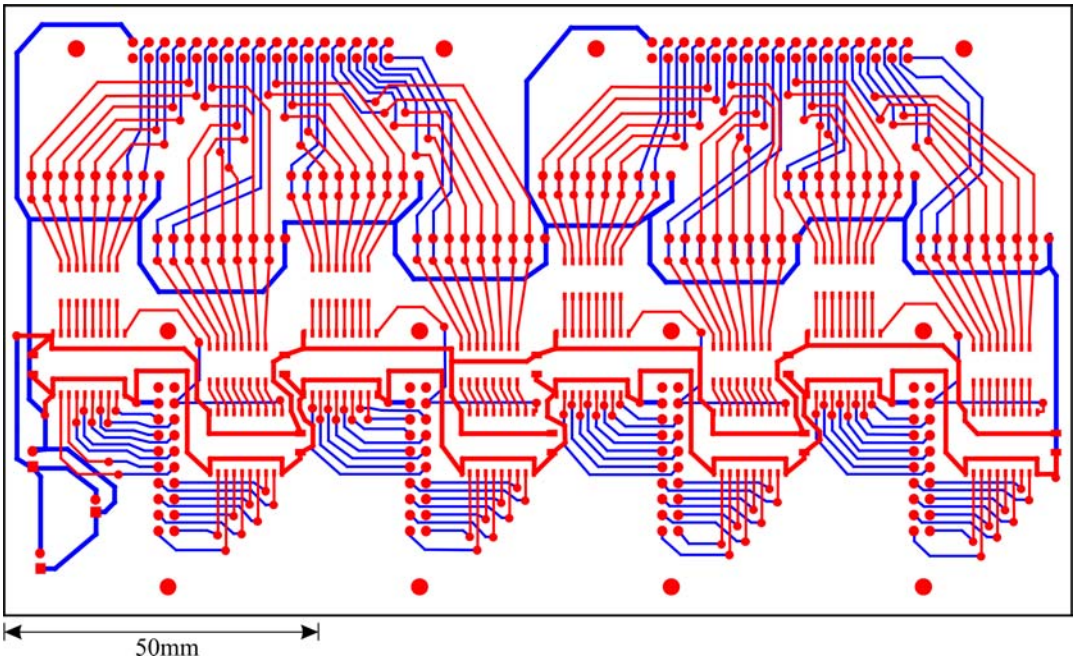

Figure 98: Digital VCSEL Driver Module PCB Layout

Digital VCSEL driver module PCB mask where red is the upper layer and blue the lower.

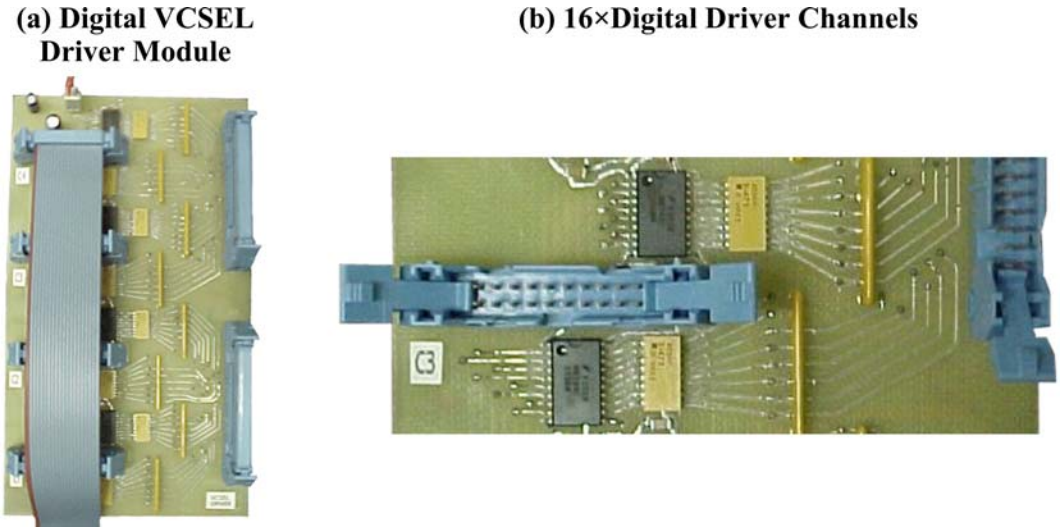

Figure 99: Digital VCSEL Driver Module Images

Part (a) shows the complete digital VCSEL driver and part (b) 16 channels that would be controlled by a single DSP module.

Two versions of the digital VCSEL driver were constructed. Only the second device is detailed here, however fabrication of a single chip solution to replace this entire module was undertaken during the project. Due to power constraints, a single chip was only capable of driving 16 VCSELs, therefore four devices were required.

5.3.4 Optical System

Again the optical system can be calculated given certain intrinsic values and basic lens formulae. The principle behind the optical system did not change from the first generation demonstrator; however it was packaged into a smaller area. Figure 100 details the new optical system size and includes images of the optically aligned components.

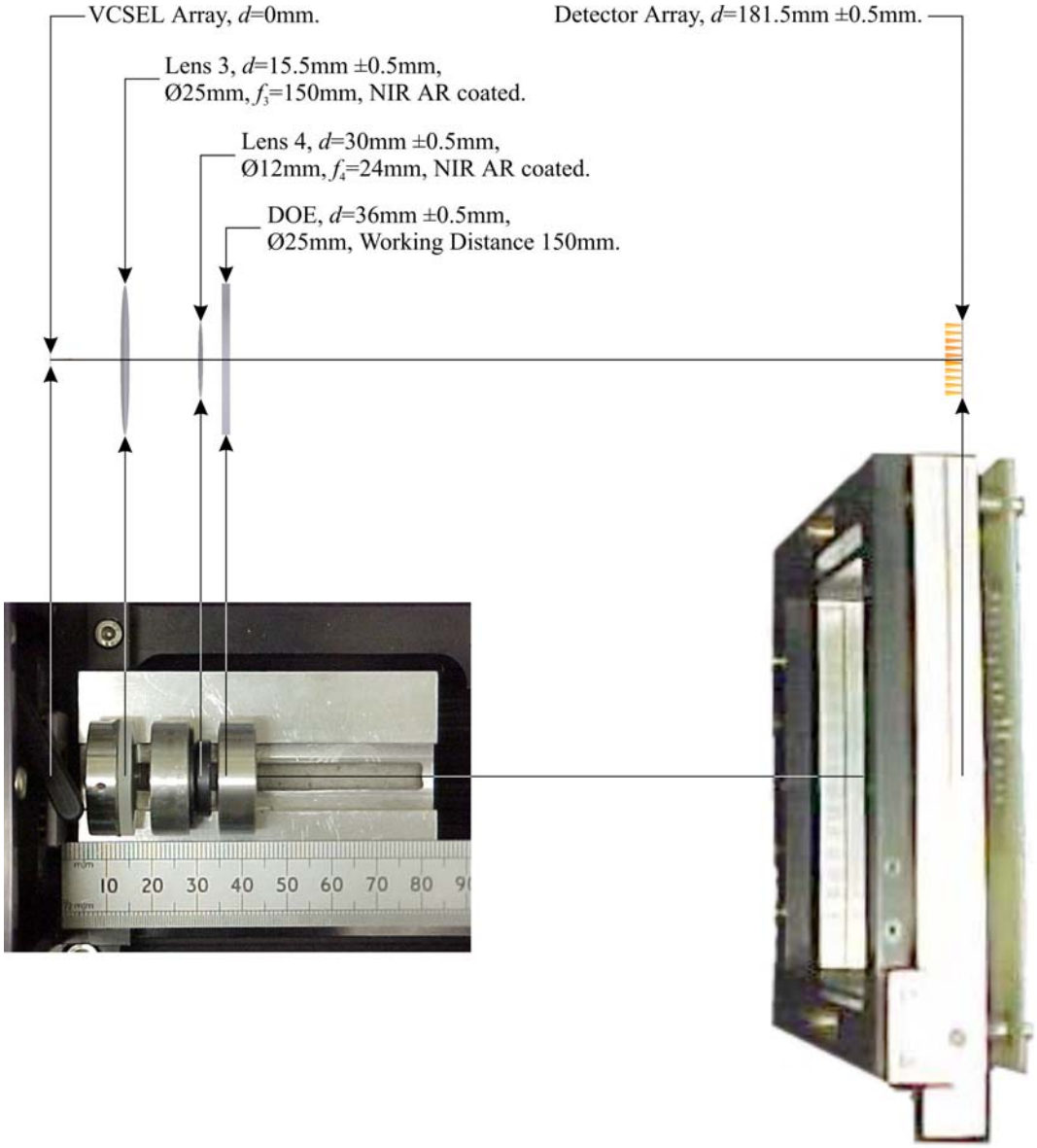

Figure 100: Second Generation Optical System

The components in this diagram are scaled relative to each other and overall system size. Distance *d* is measured from the VCSEL array on the left at *d*=0mm. The detector array on the right is at *d*=181.5mm which is the overall system size. Data flows from left to right in this figure. Note that component alignments seem slightly out on the diagram. This is due to the perspective of the camera lens.

The first generation demonstrator was based around a free-space interconnection held in place by optical bench components. In general, such systems have a negative effect on alignment stability and physically occupy a large space. Therefore the second generation used a compact optomechanical baseplate with groove and magnet technology. Furthermore, the addition of *x*, *y* and *z* axis translation mechanisms simplified the focusing of diffracted VCSEL outputs onto the photodetector array. Figure 101 shows the device schematics and Figure 102 an image of the completed baseplate.

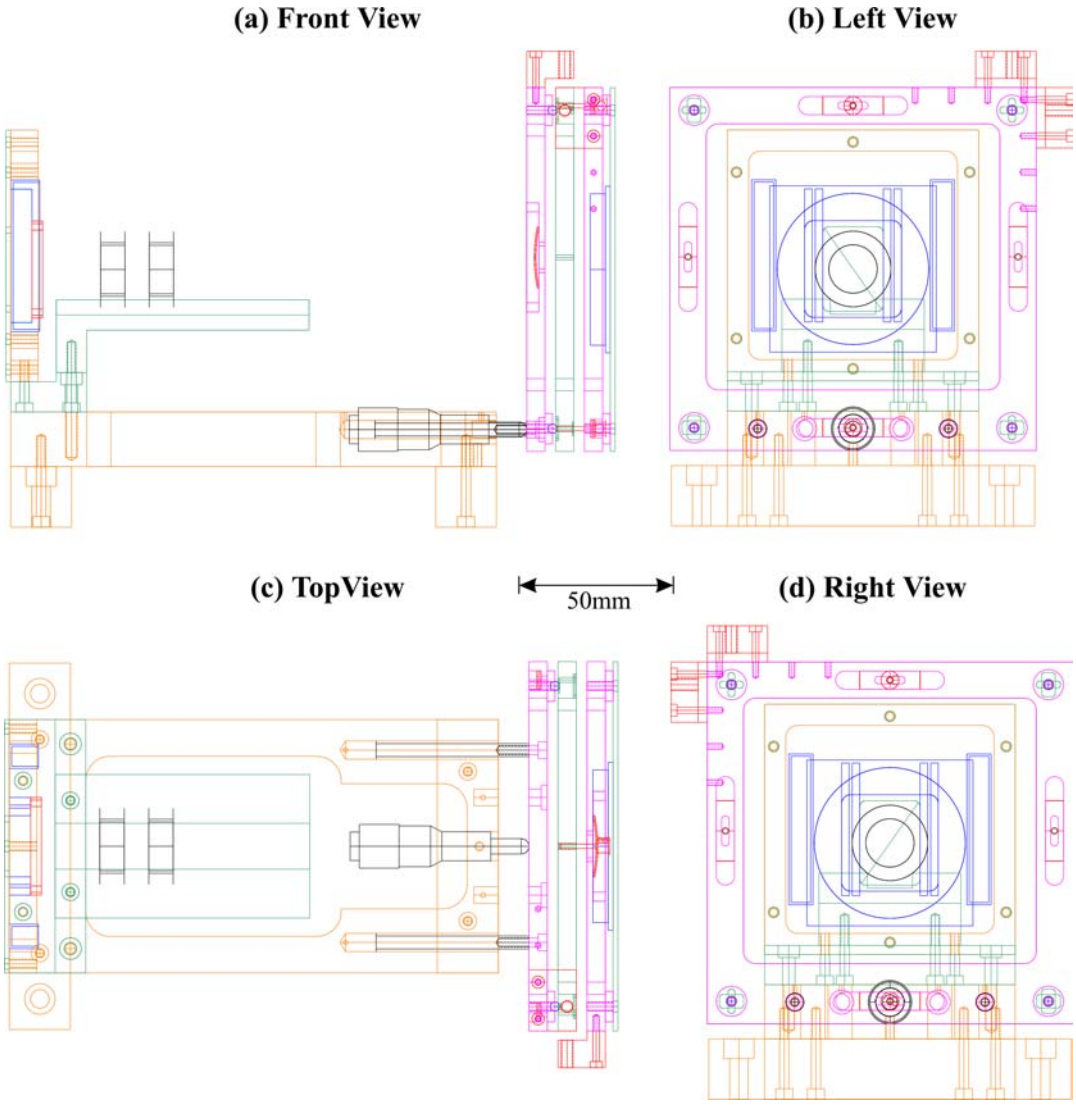

Figure 101: Baseplate Schematics
Compact optomechanical baseplate construction diagrams. Scale is valid for all diagrams.

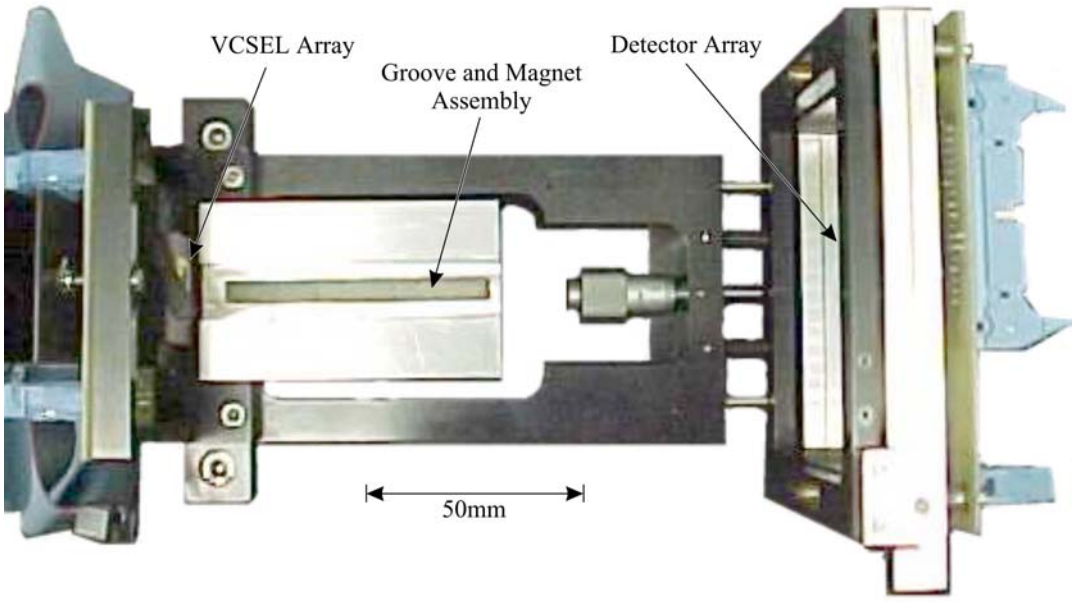

Figure 102: Baseplate Image

Groove and magnet assembly used for placement of passive optical elements. Detector position adjustable in *x*, *y* and *z* directions.

The passive optical components were mounted in metal rings allowing accurate and stable placement in the groove. This baseplate design was highly successful in terms of mechanical stability and size.

Circuit boards had to be constructed to position and interface with both VCSEL and detector arrays. Attention was paid to placement of the optical components as any error could potentially prevent alignment at a later stage. The VCSEL array was mounted on a circuit board as shown in Figure 103.

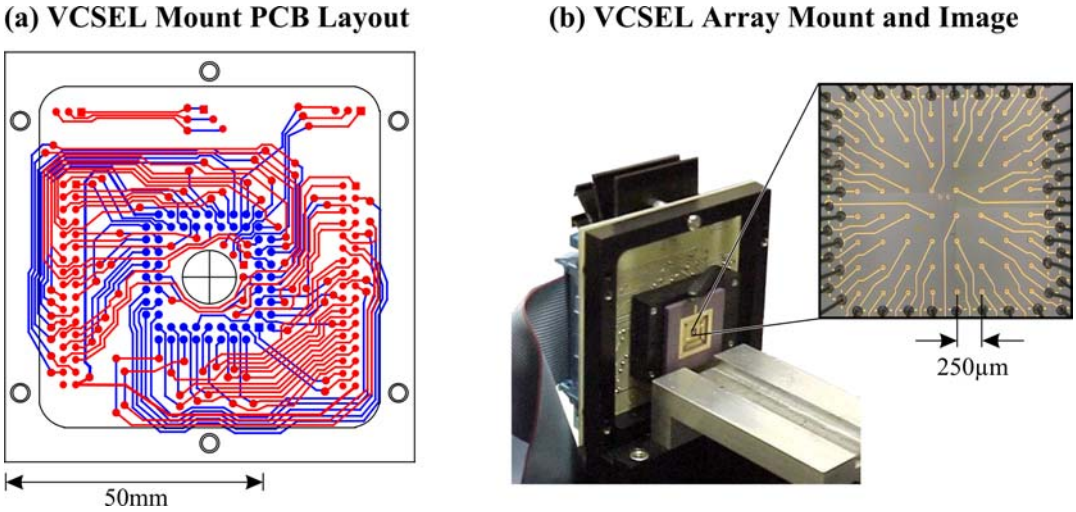

Figure 103: VCSEL PCB Layout and Mount

PCB mask uses red to represent the upper layer and blue the lower. VCSEL array mounted in zero insertion force (ZIF) socket to allow replacement.

The VCSEL mount is electrically connected to the VCSELs as shown in Figure 104. Each VCSEL is numbered as shown in the figure. The array is split into four channels, each handled by an external processor that controls 2 rows or 16 VCSELs and numbered from 1 to 4.

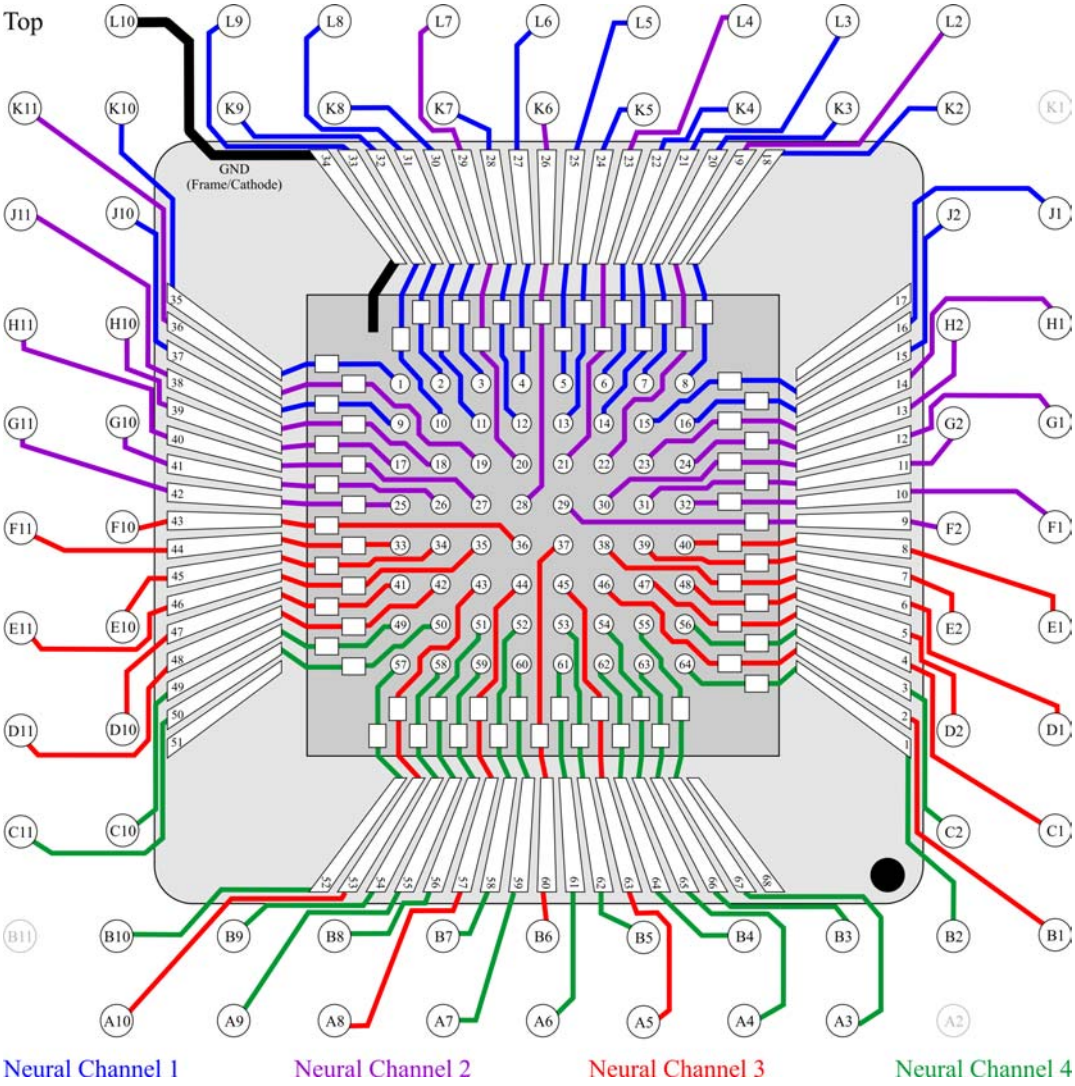

Figure 104: VCSEL Array Connections

The four DSP modules are associated with blocks of VCSELs arranged in 2×8 sets.

Each VCSEL has an associated detector as shown in Figure 105. The optical system inverts the VCSEL image both horizontally and vertically.

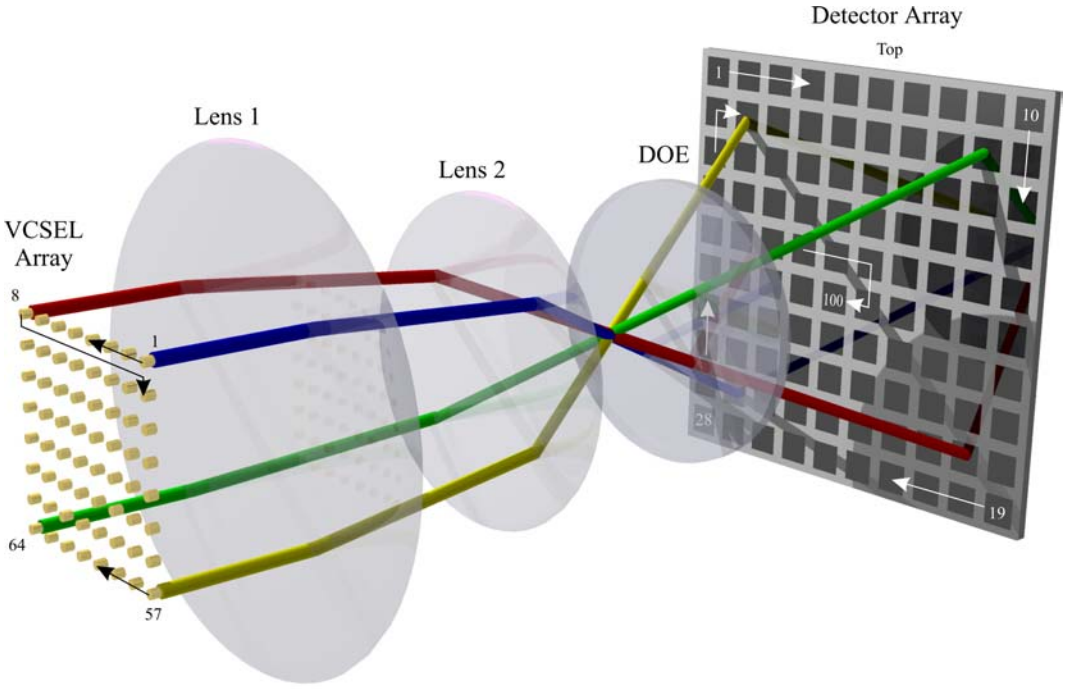

Figure 105: VCSEL to Detector Mapping
Illustrates association of VCSEL and detector pairs to construct a neuron. This figure is not an accurate optical representation.

A detector mount is also required and can be seen in Figure 106. The detector array used is identical to that in the first generation system.

(a) Detector Mount PCB Layout **(b) Detector Mount Image**

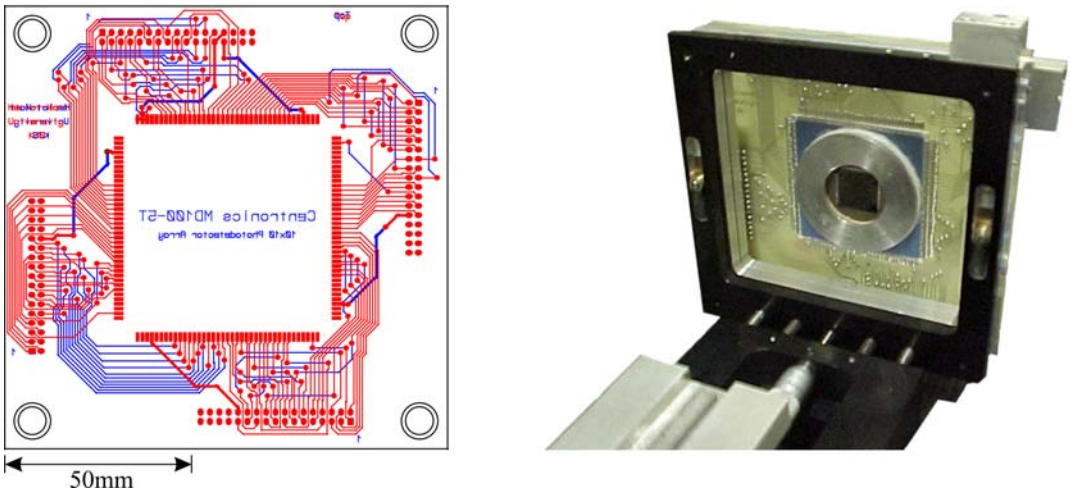

Figure 106: Detector PCB Layout and Mount
PCB mask uses red to represent the upper layer and blue the lower.

The optical system was aligned and performed as predicted.

5.3.5 DOE Characteristics

The first generation DOE design did not prove to be of adequate uniformity for the required task. Therefore, the design was re-optimised using an element with double the diffraction order spacing of the initial element. Instead of orders 1, 2, 3 etc. the new element used orders 2, 4, 6 etc. This change to on-order spacing allowed an element of double the initial period to be fabricated with a commensurate increase in minimum feature size. The characteristics of these new DOEs are shown in Table 15, both proving more than adequate to implement their respective interconnect architectures.

DOE	T_{DOE} (%)	Δr (%)
Crossbar Switch	50.0	0.81
Banyan Switch	50.0	0.83

Table 15: Second Generation DOE Efficiency and Non-Uniformity
Non-uniformity across the array has been decreased by using every second order. This also results in a decrease in transmission efficiency. The elements shown here are well suited to an N=8 optoelectronic neural network.

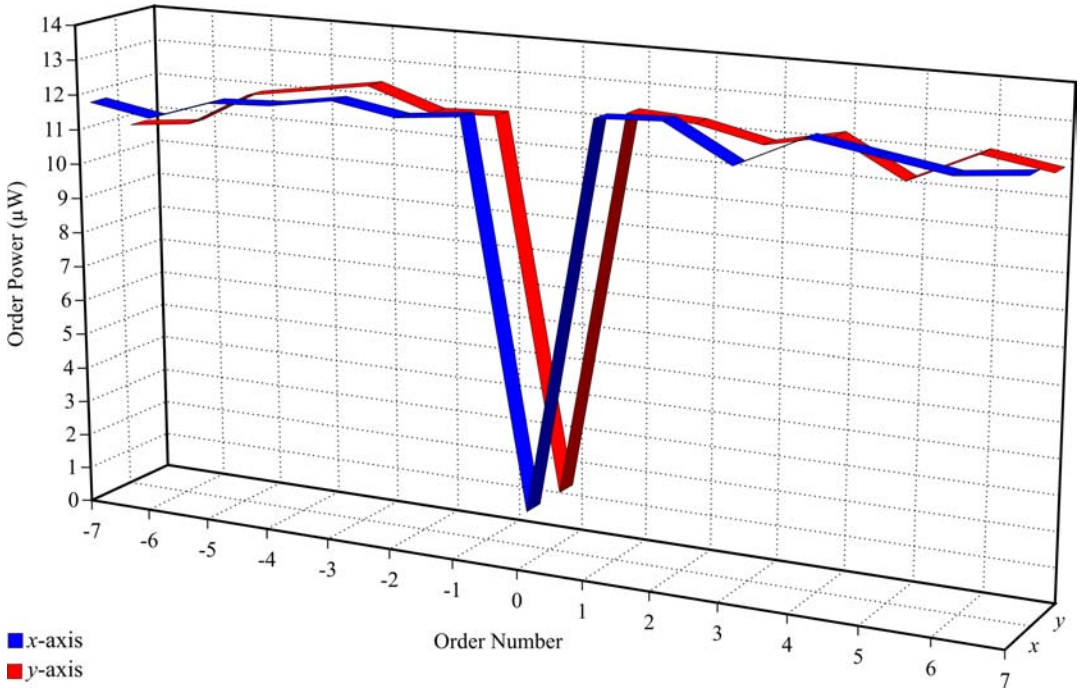

Figure 107: Second Generation Diffractive Optic Element Profile
Measured optical power in each spot given a total input power of 714.4µW ±0.2µW. All readings taken with VCSEL stabilised at 16ºC.

The DOEs constructed were of binary type and optimised for λ_{2g}=960nm. Profiling of spot intensity distribution was performed on the element. The power in each spot on both x and y axes can be seen in Figure 107. The diffracted power profile is well defined with an overall experimental diffraction of 47.1% of input power. 0.2% of this is present in the zero order and 52.7% is scattered outside the diffraction window, effectively being lost.

5.3.6 VCSEL Array Characteristics

The VCSEL array examined here was produced by CSEM Zurich [160] and is an oxide confined device. It has an output wavelength of 960nm with a tight maximum variation in wavelength variation across the array of $\Delta\lambda_{max}$=0.25nm. The array has a mean threshold current of 0.45mA ±0.05mA and a mean threshold voltage of 1.4V ±0.05V. See Figure 108 for current-voltage and Figure 109 for current-power curves. From these graphs, at an operating current of 5mA the device gives 1.15mW ±0.05mW mean optical output power. Thus the power conversion efficiency can be approximated as 10.9% ±0.1%.

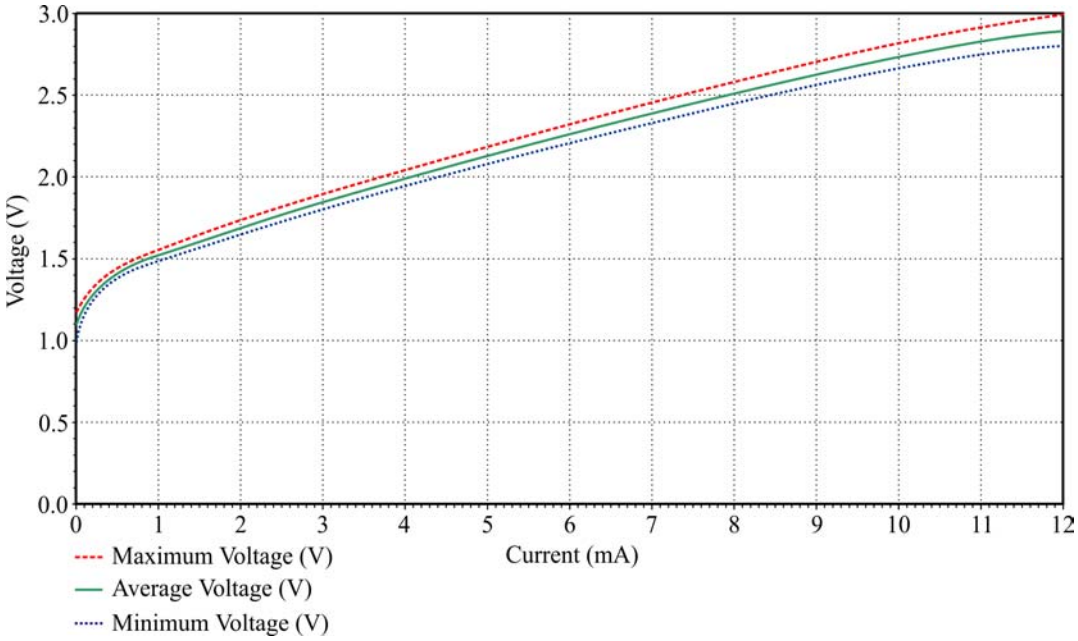

Figure 108: CSEM D6 VCSEL Current to Voltage Characteristics
Minimum, maximum and average values sampled across all 64 array elements.

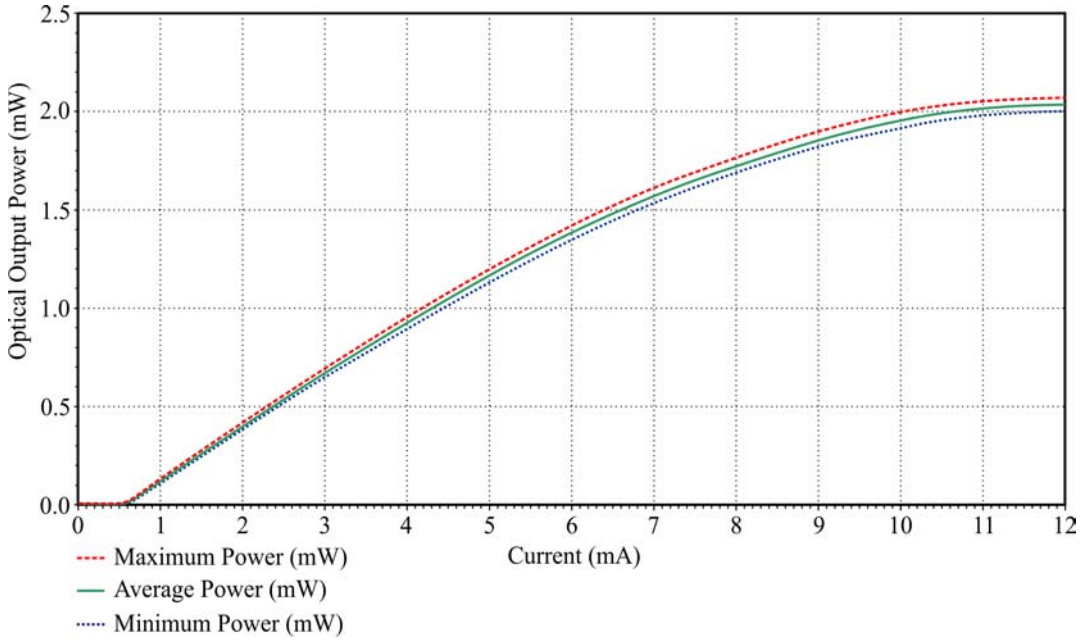

Figure 109: CSEM D6 VCSEL Optical Output Power Characteristics
Minimum, maximum and average values sampled across all 64 array elements.

Profiling of the optical power output from each device in the array was performed. Given that the divergence of each VCSEL is 21º ±0.5º, there was a power loss from each measurement due to the size of the detector input aperture of 8mm. This aperture was set 12mm from the array. Assuming an even cross-sectional power distribution, calculations indicate that only 71.8% ±0.5% of emitted power should be measured.

A current of 6.5mA ±0.3mA was applied to each VCSEL and the optical power output recorded. Theoretically, this should be in the region of 1.4mW ±0.05mW. The entire array profile can be seen in Figure 110.

Even after the divergence has been taken into consideration, there is still an unaccounted loss of 0.335mW on average. One possible reason is that oxide confinement at the aperture leads to localised heating of the injection region and consequent thermal lensing which varies beam direction and divergence with respect to local temperature. Another possible, and simpler, reason could be that the VCSEL array had simply degraded since fabrication in 1999. Regardless, the measured powers were within acceptable demonstrator operational tolerances.

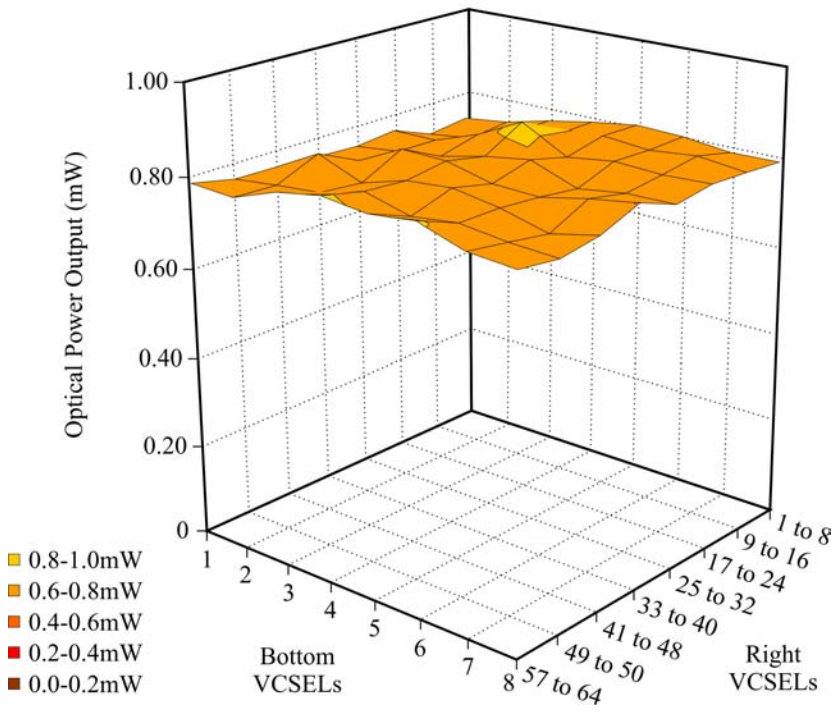

Figure 110: VCSEL Array Power Profile

Each point represents the measured output power of a single VCSEL. All measurements have an error of ±0.05mW.

This system uses a *thermoelectric cooler* (TEC) [161] to regulate the temperature of the VCSEL array and thereby minimise any potential temperature stability issues such as those observed in the fist generation demonstrator.

Profiling was performed on the array to assess temperature stability. The temperature was slowly reduced and optical power output measured. As expected, optical power increased as temperature decreased. The results can be seen in Figure 111. The temperature was not taken lower than 9ºC as the VCSEL output became unstable at this point.

Two interesting points were noted. Firstly, jitter of the VCSEL optical output is noticeable as temperature decreases. This is probably due to changes in direction and divergence from thermal lensing.

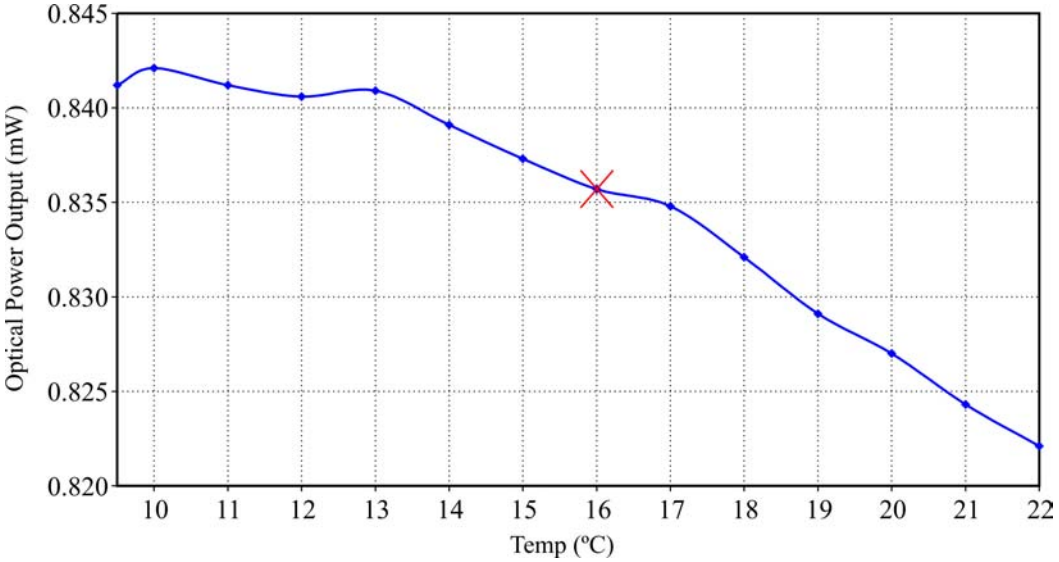

Figure 111: Temperature Stability of CSEM D6 VCSEL Array
The preferred temperature is marked with a cross. Note that temperature stability is excellent with little real variation in optical power output.

Secondly, there is a noticeable temperature gradient effect. For example, a VCSEL is switched on at an initially low temperature. Activation of the VCSEL results in excess heat being produced locally around the VCSEL. The longer that the VCSEL is active, the more heat that is produced. This gives an initially higher level of optical output power due to a cooler VCSEL that slowly decays to an equilibrium point. Activating more VCSELs reduces the equilibrium point further by adding excess heat and consequently decreases the optical output power of each individual VCSEL. This can be compensated for to a certain degree by increasing the thermoelectric cooler drive current.

Care also has to be taken that the VCSEL array temperature does not pass below the *dew point* as condensation of water would obviously prove problematic. The dew point defines the temperature at which condensation will start to form based on environmental *relative humidity* (RH). This has been measured at 53% ±1% with an ambient room temperature of 26ºC ±0.5ºC. The dew point can be calculated as approximately 15.5ºC. The array should preferably not be cooled below this temperature during operation. Examination of temperature versus optical output power curves also allows us to conclude that a temperature of 16ºC is desirable for all subsequent measurements. This is because a short term rise of 1ºC has a reduced, if nevertheless minimal, impact on output power. Furthermore, 16ºC is not too cool to be unachievable on hot summer days and not too hot to be impractical on cold winter days. The TEC used to cool the VCSEL array is a Marlow DuraTEC DT3-4-01LS and can be seen in Figure 112.

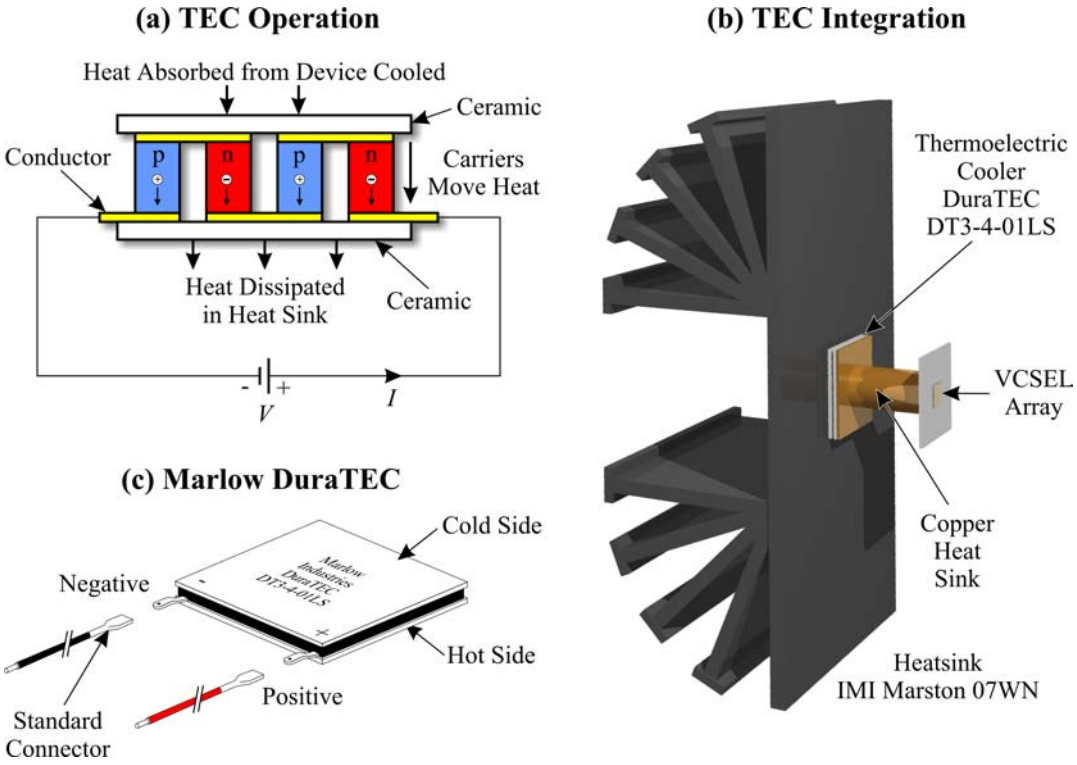

Figure 112: DuraTEC DT3-4-01LS Thermoelectric Cooler
Device parameters can be seen in Table 16.

Parameter	Value	Parameter	Value
ΔT_{max} (Vacuum)	68ºC	ΔT_{max} (Dry N_2)	64ºC
V_{max}	3.6V	I_{max}	3.7A
Q_{max}	9.0W	Stages	1

Table 16: DuraTEC DT3-4-01LS Device Parameters
ΔT_{max} indicates the temperature difference between hot and cold sides of the device at maximum load.

There are two points for temperature measurement at which semiconductor temperature sensors are installed. One is directly behind the VCSEL array on the copper conductor and the other is after the thermoelectric cooler on the main heatsink. The heatsink is made by IMI Marston and can dissipate 2.8ºCW^{-1}.

To ensure that the VCSEL array temperature does not go lower than the dew point, the minimum temperature must be sustained at 16ºC with no elements on. Given an atmospheric temperature of 27ºC ±0.5ºC, the thermoelectric cooler drive current was gradually increased until the temperature stabilised at 16ºC ±0.5ºC. This was found to

be I_{TEC}=0.5A. The TEC's performance [162] can therefore be examined by plotting the line (a) on Figure 113 given that $I/I_{\max} = 0.139$.

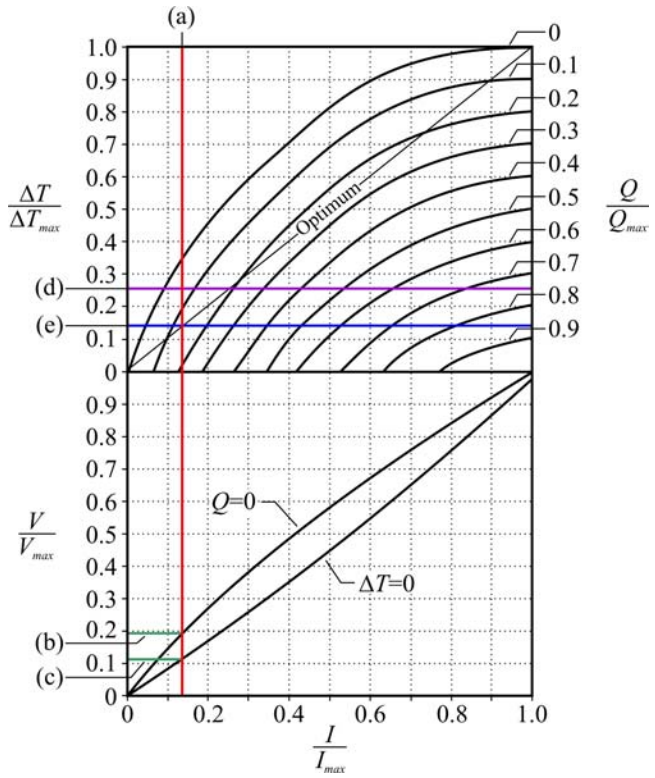

Figure 113: DuraTEC DT3-4-01LS Performance Graph
Graph used to estimate performance of TEC.

By drawing lines (b) and (c) on Figure 113 we can extrapolate high and low TEC voltages respectively. The lower voltage cools smaller differences in temperature and the higher handles larger. Therefore our voltage must lie within the regions of $V_{TEChigh} = 0.7\text{V}$ at the high end and $V_{TEClow} = 0.414\text{V}$ at the low end.

Measurement indicated that the system was operating almost exactly at $V_{TEChigh}$=0.7V. Therefore, we can calculate the TEC power consumption as being $Q_{TEC} = 0.35\text{W}$. This resulted in a measured temperature increase at the heatsink of 5ºC to 32ºC ±0.5ºC. Given this rise, we can determine the temperature difference, and thereby power, using the graph as being $\Delta T/\Delta T_{\max} = 0.25$. This value is drawn on Figure 113 as line (d) and allows us to calculate the convective heat energy dissipated as being $Q_{conv} = 0.54\text{W}$.

This value can be verified theoretically by working out the convected heat absorbed from the atmosphere into each of the three cooling element components. The summation of these gives $Q_{Tconv} = 0.554\text{W}$ and is close to the measured power dissipated Q_{conv}. Indeed, the theory is very close to measurement with an error margin

of less than 2.5%. This is probably due to other thermal factors which were considered negligible in this case.

The next stage was to examine the array temperature when all VCSEL elements are active. This resulted in a rise at the array of 7ºC ±0.5ºC to 23ºC ±0.5ºC. Therefore, we can calculate $\Delta T/\Delta T_{\max} = 0.14$. By plotting line (e) in Figure 113 it can be seen that the thermoelectric cooler is approaching optimum efficiency at full VCSEL array power. This gives a heat load of $Q_{active} = 1.125\text{W}$. Linear extrapolation allows estimation of the convected heat load with a difference of 4ºC ±0.5ºC as being $Q_{aconv} = 0.196\text{W}$ where the additional heat from the VCSEL array is $Q_{vcsel} = 0.929\text{W}$.

The accuracy of this can again be calculated theoretically. Each VCSEL in the array has been measured to consume $P = 14.95\text{mW}$ of energy. Since the theoretical efficiency of these VCSELs is 10.9%, 89.1% of this power would be lost as excess heat. For 64 VCSELs this would be a generated heat of $Q_{Tvcsel} = 0.853\text{W}$. The slightly higher measured temperature Q_{vcsel} is due to the three thermal interfaces through which the active heat load must travel before being thermoelectrically pumped into the heatsink. Initially, these interfaces used a standard thermal compound, Dow Corning 340, which had a thermal transfer coefficient of $0.48\text{Wm}^{-1}\text{ºC}^{-1}$. It was suspected that they were bottlenecking heat transfer, so a silver thermal compound was applied, Arctic Silver II, which had thermal conductivity of $8.6\text{Wm}^{-1}\text{ºC}^{-1}$. This significantly improved the rate at which heat was transferred and reduced the temperature difference at full load between VCSEL and TEC cold side by 6ºC.

The calculations and measurements made in this chapter allow array temperature prediction. As long as I_{TEC}=0.5A and $V_{TEChigh}$=0.7V, the VCSEL array will have the following temperature:

$$T_{vcsel}(\text{ºC}) = T_{air} - 11 + (n_v \times 0.11) \qquad \textbf{Equation 72}$$

where n_v is the number of VCSELs continually active ranging from 0 to 64. Note that n_v will need to be modified depending on VCSEL load. For example, if there are n_v=32 VCSELs active but only for 50% of the time then one should half the value of n_v. These calculations are specific to the thermoelectric cooler and associated heat sinking used here.

The VCSEL system was run at full duty cycle twice, for about five minutes at a time, to assess stability at high temperatures. Appropriate cooling was applied using the thermoelectric module. Unfortunately, the array had not been mounted on a sapphire

plane which allows localised heat to be effectively removed. This resulted in the degradation of power output from one of the VCSELs. It is not known if this is the sole cause, as the VCSELs have a *mean time between failure* (MTBF) of ~7 hours. System alignment and testing must have used a fair percentage of their lifetime by this point.

5.3.7 Transimpedance Amplifier Module

Transimpedance amplifiers, or current to voltage converters, take a current produced by a photodiode and convert it into a voltage. In so doing, they improve both the linearity and bandwidth of photodiode response. This section examines the transimpedance amplifier module used in this system and its associated characteristics.

This section is written in reverse, defining output and then matching input characteristics. Since the analogue to digital converter which takes information from the amplifiers has a specific input voltage range requirement, 1.25V for 0% to 3.75 for 100%, signal amplification and conditioning is necessary to ensure easily discernable readings. Figure 114 shows the designed circuitry.

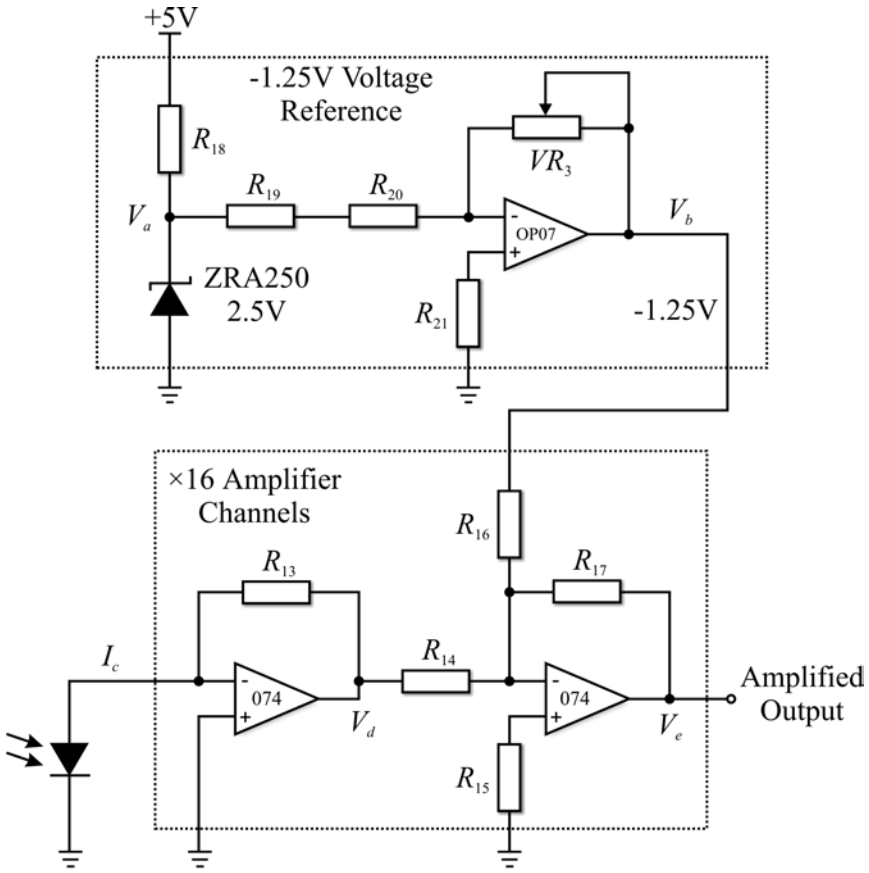

Figure 114: Transimpedance Amplifier Circuit Diagram

In total 64 amplifier channels exist divided up across four modules. Table 17 shows component values.

There are 64 channels in total, each channel requiring its own dedicated circuitry, with all the channels divided up across 4 identical modules. Each module has a voltage reference source as shown in the upper branch and 16 amplifier channels as shown in the lower branch.

Component	Value	Component	Value
Resistor R_{13}	33kΩ	Resistor R_{14}	33kΩ
Resistor R_{15}	12kΩ	Resistor R_{16}	33kΩ
Resistor R_{17}	33kΩ	Resistor R_{18}	33kΩ
Resistor R_{19}	33kΩ	Resistor R_{20}	33kΩ
Resistor R_{21}	22kΩ	Variable Resistor R_3	50kΩ

Table 17: Second Generation System Component Values
Component types and values for Figure 114.

First let us examine the voltage reference circuitry in Figure 114. This circuitry exists because the analogue to digital converter in the next stage requires a signal that is 1.25V at its lowest point. However, this point should be adjustable to compensate for component tolerances. With reference to the points marked on Figure 114, a precision voltage reference source ZRA250 was used in conjunction with R_{18} to give a steady 2.5V at V_a. Since 1.25V was desired, this value then needed to be halved and inverted. This was done using a precision voltage offset amplifier which has high voltage output stability over time. The device was configured in inverting mode and a variable resistor VR_3 used to adjust the voltage reference output. Given that V_a is a 2.5V input, the voltage output at V_a can be described as:

$$V_b = 2.5\frac{-VR_3}{R_{19} + R_{20}} \qquad \textbf{Equation 73}$$

Under normal circumstances, VR_3 should be set to $33\text{k}\Omega$ for $-1.25\,\text{V}$ output. Adjusting VR_3 toward $0\text{k}\Omega$ will cause it to approach 0V. Increasing VR_3 towards $50\text{k}\Omega$ will theoretically give an output of $-1.89\,\text{V}$. This value has been measured at 1.8V which remains within the 5% resistor tolerance.

Input bias current [13] can prevent a zero input voltage from giving a zero output voltage. To minimise this effect, the value of R_{21} has to be chosen carefully. In this case:

$$R_{21} = \frac{VR_3R_{19} + VR_3R_{20}}{VR_3 + R_{19} + R_{20}}$$ **Equation 74**

Thus, if VR_3 is set to $33\text{k}\Omega$:

$$R_{21} = 22 \times 10^3\,\Omega$$ **Equation 75**

Therefore R_{21} was set to $22\text{k}\Omega$.

There are 16 identical amplifier channels on every module, each with two amplifiers in series using a Texas Instruments TL074CD [163]. The first amplifier takes an input current from a photodetector I_c in Figure 114 and converts it into an output voltage. All amplification of incident light must be done in this stage to minimise signal to noise ratio. It is configured in transimpedance mode, otherwise known as a current-to-voltage converter, the voltage at point V_d being described by the equation:

$$V_d = -I_c R_{13}$$ **Equation 76**

where R_{13} is $33\text{k}\Omega$. The system should theoretically generate a 34mV deflection for every 1μA of current generated by the detector. Therefore, there should be a 116mV deflection for a single incident VCSEL channel. Deviations are to be expected between channels due to imperfection in component manufacture.

The input voltages from points V_b and V_d are then put through a second amplifier stage configured in summing mode. This gives the following voltage at point V_e:

$$V_e = -\left(\frac{V_b R_{17}}{R_{14}} + \frac{V_d R_{17}}{R_{16}}\right)$$ **Equation 77**

Again, the effect of input bias current must be minimised by careful selection of R_3:

$$R_{15} = \frac{R_{14}R_{16}R_{17}}{R_{14}R_{16} + R_{14}R_{17} + R_{16}R_{17}}$$ **Equation 78**

$$R_{15} = 11 \times 10^3\,\Omega$$ **Equation 79**

As there was no $11\text{k}\Omega$ component available, a $12\text{k}\Omega$ resistor was used instead.

Figure 115 shows four completed modules and the associated masks. The use of surface mount components means that all circuitry required for two amplifier channels fits neatly into 15mm×15mm using only a single side of the circuit board. *Decoupling* capacitors were used on each amplifier of 100nF and large value electrolytic capacitors of 10μF and 100μF were used for power supply smoothing at point of entry to each module.

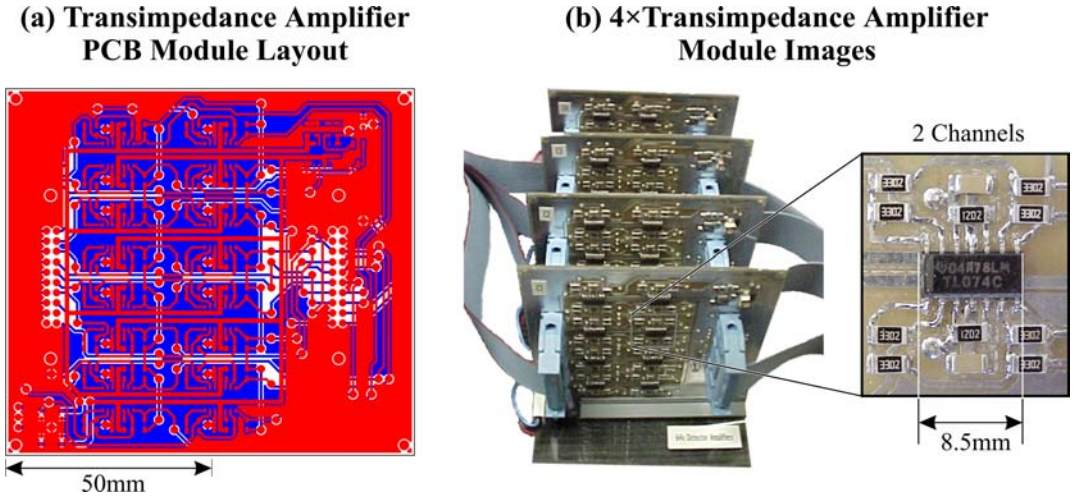

Figure 115: Transimpedance Amplifier Module PCB Layout and Image
PCB mask uses red to represent the upper layer and blue the lower.

The photodiode transimpedance amplifier built was analysed both theoretically and experimentally. However, the theory indicated that the amplifier would resonate. A phase compensation capacitor was added and the system re-examined. The addition of this capacitor reduces resonance but will also decrease bandwidth. Resonance has also been observed experimentally and almost certainly stems from the detector circuitry since laser relaxation oscillations, the only other possible source, have died away 5ns after any state change [160].

The circuit constructed in Figure 114 was examined experimentally to evaluate the magnitude of resonance. Calculations predict that the resonance frequency will be approximately 499kHz. Measurement indicates that it is slightly less at 465kHz ±22kHz. Figure 116 examines the response closely and also shows traces with the addition of a phase compensation capacitor. It is clear here that a 10pF capacitor is nearly, but not quite, enough to remove amplifier resonance whereas 22pF does the job well. However, initial rise times are badly affected as capacitance increases. Returning to the previous theoretical bandwidth analysis, 0pF has a -3 dB point of 699kHz, 10pF of 471kHz and 22pF of 217kHz. Examining the oscilloscope images highlights the difference in rise times with varying feedback capacitance.

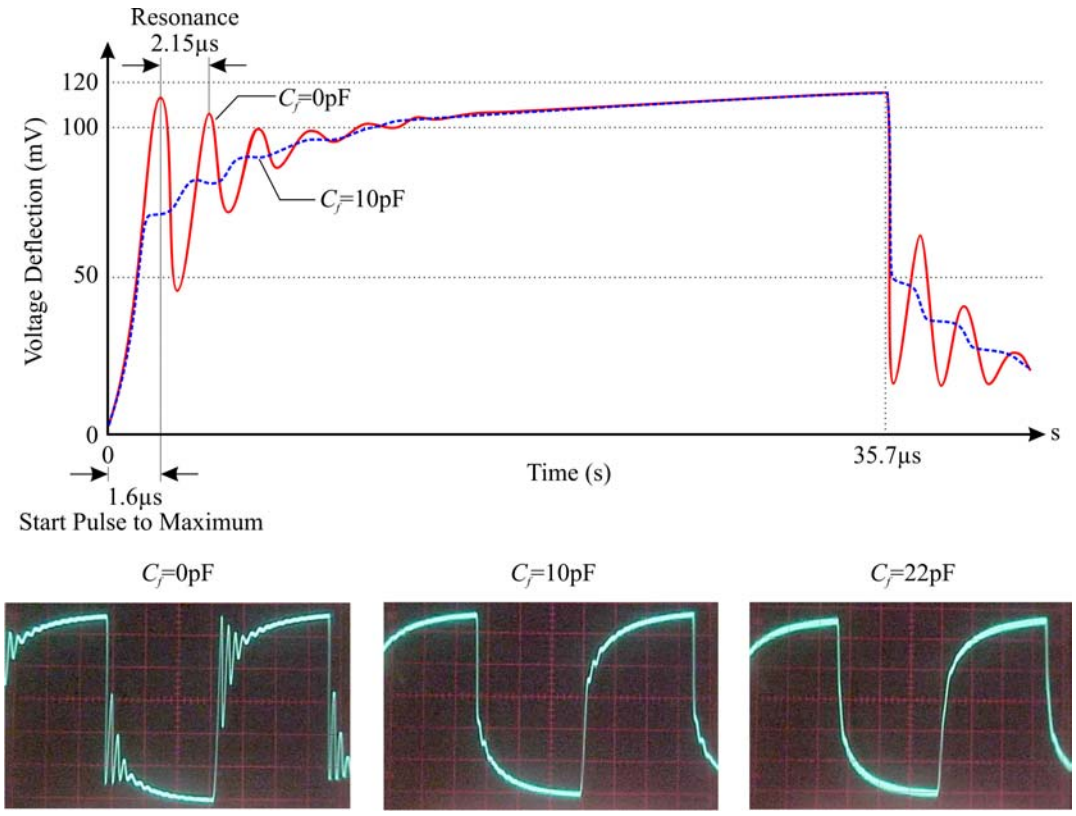

Figure 116: Transimpedance Amplifier Response and Resonance
Results taken at ~14kHz.

Although not obvious at 14kHz, the slower rise time of the phase compensated circuit becomes visible at 55kHz as shown in Figure 117. However, this circuit was constructed without a capacitor for one good reason - information flow is controlled by a digital signal processor. Addition of a capacitor prevents the voltage from reaching full scale until nearly 8µs after the start pulse has been sent. Without phase compensation, this value is reached much more quickly, but not stably, in a constant time of 1.6µs. Given that the system knows when the start pulse was sent, careful timing allows precise sampling of the peak value. As the DSP has a clock frequency of 40MHz, NOP operations can be executed to exactly time sampling given that each one has a delay of 25ns. Timing begins immediately after the start pulse has been issued. To ensure best results, the ADC hold time of 120ns must be timed to straddle the first peak exactly so sampling must begin 60ns beforehand.

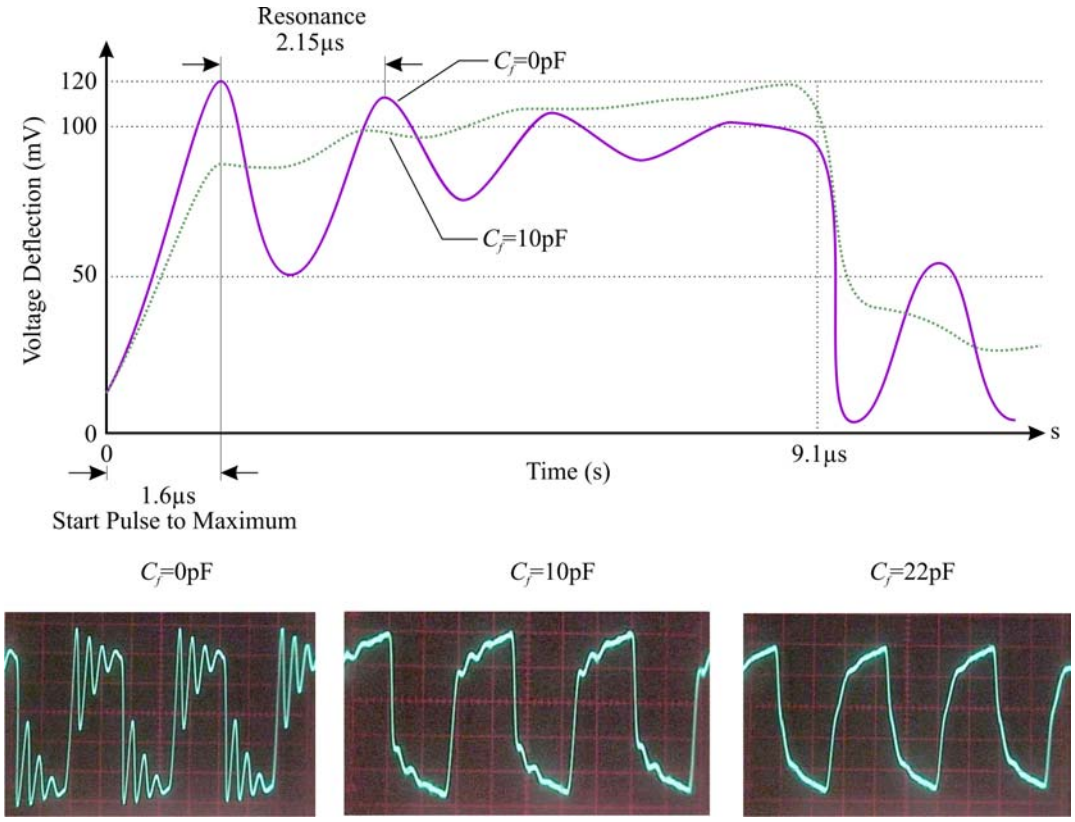

Figure 117: Transimpedance Amplifier Response and Resonance
Results taken at ~55kHz.

To ensure correct timing a total of 58 NOP commands are required. This is a large amount of time, as far as the DSP is concerned, and can be used to perform other calculations if care is taken to ensure accurate sequencing. Subsequent results could be taken by waiting for the next peak. This trick squeezes bandwidth out of the system where it is not normally available.

Sometimes this timing method may not be practical, but there is another option. By taking two samples half the resonance frequency apart (approximately 1.15µs) we can average the values to get a response similar to that of the phase compensated circuit. However, this does have its limitations as approximately 6µs after the start pulse the phase compensated circuitry will begin to give more accurate readings. In the worst case scenario, the results will be 20% below that of a phase compensated system. This figure improves with time.

The experimental bandwidth of the transimpedance amplifier system was measured with and without phase compensation. Figure 118 shows the results taken. To aid comparison the -3 dB voltage can be calculated using:

$$V_{-3dB} = \frac{V_0}{\sqrt{2}}$$ **Equation 80**

where V_0 is the peak-to-peak voltage at low frequency.

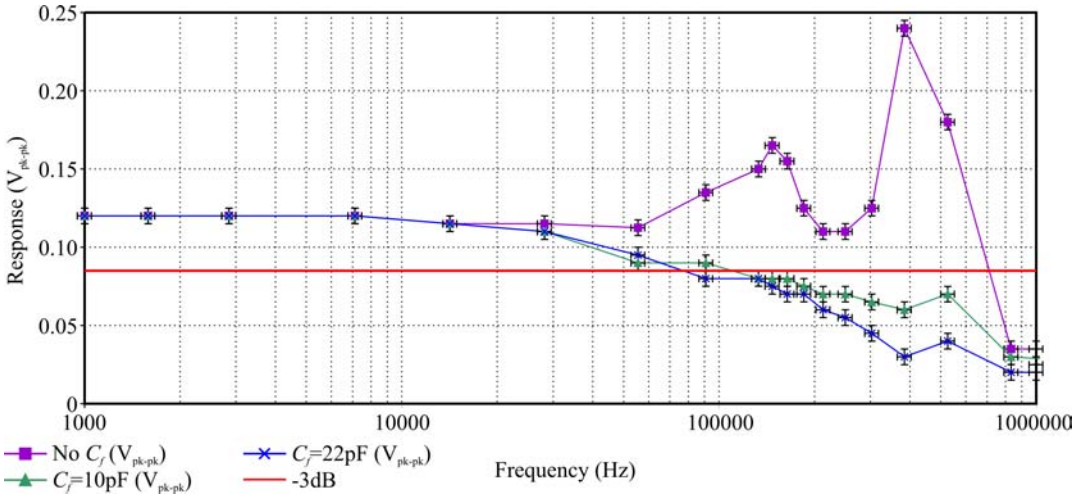

Figure 118: Transimpedance Amplifier Measured Bandwidth
Note that dual pole behaviour is visible, especially in the uncompensated circuit, as characterised by the additional peak.

First, let us examine the case with no feedback capacitor. Theoretically, there will be an amplification peak at 499kHz followed by a -3 dB bandwidth of 699kHz. Measurement indicates that the amplification peak is somewhere between 400 and 500kHz, which concurs with theory and points to the measured resonance frequency of 465kHz. The bandwidth was measured to be 710kHz which, again, is very close to the theoretical 699kHz.

Next we have a system with a feedback capacitor where measurement indicates that phase compensation not only reduces bandwidth but also removes peaking, as theoretically predicted. If the value of the feedback capacitor is increased, the bandwidth is correspondingly decreased. 10pF reduces the -3 dB bandwidth to 110kHz and 22pF to 78kHz. This is significantly lower than expected, indicating that there is another pole in the system which remains unaccounted for.

Although the voltage mode amplification stage remains well within its gain bandwidth product, there is an obvious secondary peak in the frequency response curve with no feedback capacitance. Phase compensation also results in frequencies lower than those theoretically predicted, indicating that this pole is removing available gain from the transimpedance amplifier. This can again be seen in Figure 116 and Figure 117. Even as amplifier resonance dies away the output voltage continues to rise, by up to 15mV if

given sufficient time. This behaviour is suggestive of a large value capacitor with a small influence on the system. The symptoms fit with an effect known as power supply noise coupling [84]. The theory indicates that fitting a larger amplifier decoupling capacitor of 10µF, rather than the 100nF capacitor currently present, can significantly reduce this problem.

There are a few recommendations for any future implementation of such a photodiode transimpedance amplifier. First, ensure that the unity gain bandwidth product of the amplifier is as high as possible. Use a component such as OPA689 where f_c>200MHz to maximise available bandwidth. Note that current feedback amplifiers do not function well in this system design. Choose R_f to maximise first stage bandwidth and use the second stage to provide further amplification. This is a trade-off so it will need to be balanced. Next, add a feedback capacitor to stabilise the current-to-voltage stage and prevent ringing. Finally, remove any potential power supply noise coupling problems by adding a high value op-amp decoupling capacitor.

5.3.8 Results

This section shows test results from the second generation neural network and extrapolates potential performance.

Firstly, a series of test patterns used to test the VCSEL array are shown in Figure 119. Note that all VCSELs are working correctly at this point in testing.

Software was written to aid alignment of the system which reads the analogue inputs and displays the amount of incident light as an OpenGL image on the PC as shown in Figure 120. Each triangle has a magnitude appropriate to the amount of incident light. These readings have been five times oversampled to ensure accuracy and to average out noise. What is obvious here is that there is a different response for each detector channel. This is due to component tolerances altering the final output values. All four detector amplifier modules were adjusted so that the minimum signal present across the entire array was just above zero. However, a large amount of noise was still observed on all input channels, ~70$mV_{pk\text{-}pk}$. This was discovered to be interference from the DSP, so shielding was applied to the DSPs and ferrite cores to interconnects to attenuate RF noise. This reduced interference significantly. Five sample data sets were then taken to calibrate zero level noise variation with no incident light. Note that the ADC has a resolution of 9.8mV for 1 LSB difference. This zero level was then used to

calibrate subsequent readings, giving a more accurate overall output, a good example of the DSP being used to remove inherent system non-linearity.

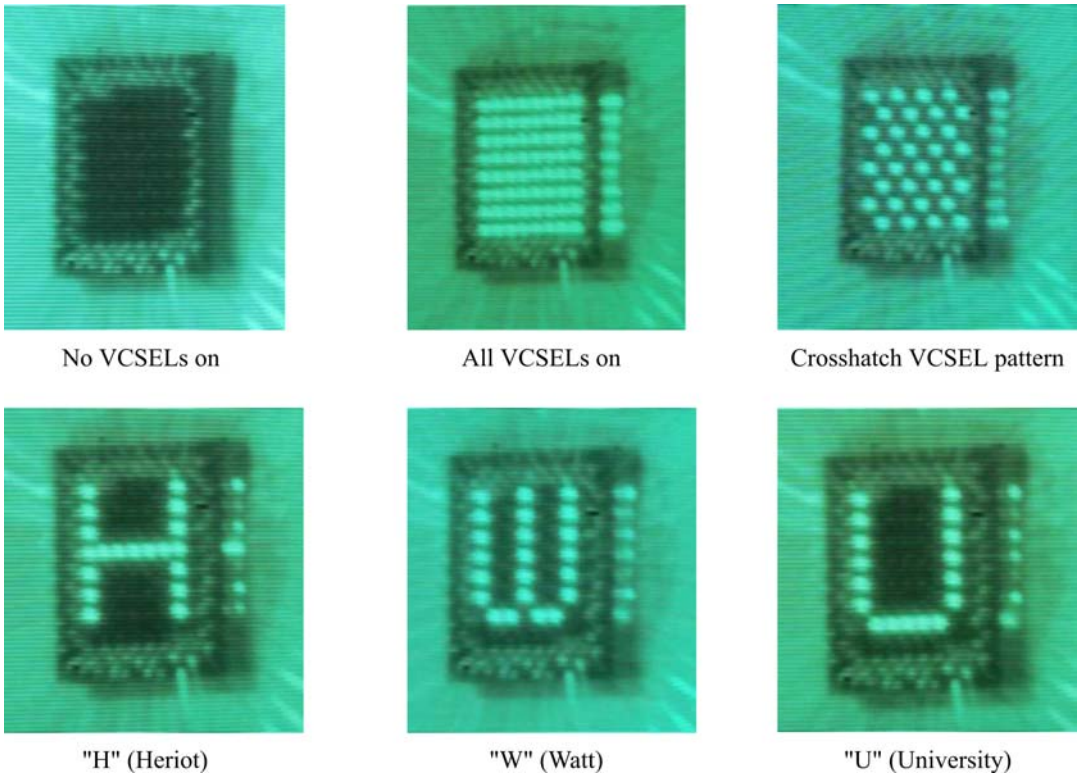

Figure 119: VCSEL Test Pattern Images
Images taken using an NIR CCD camera with high magnification lens.

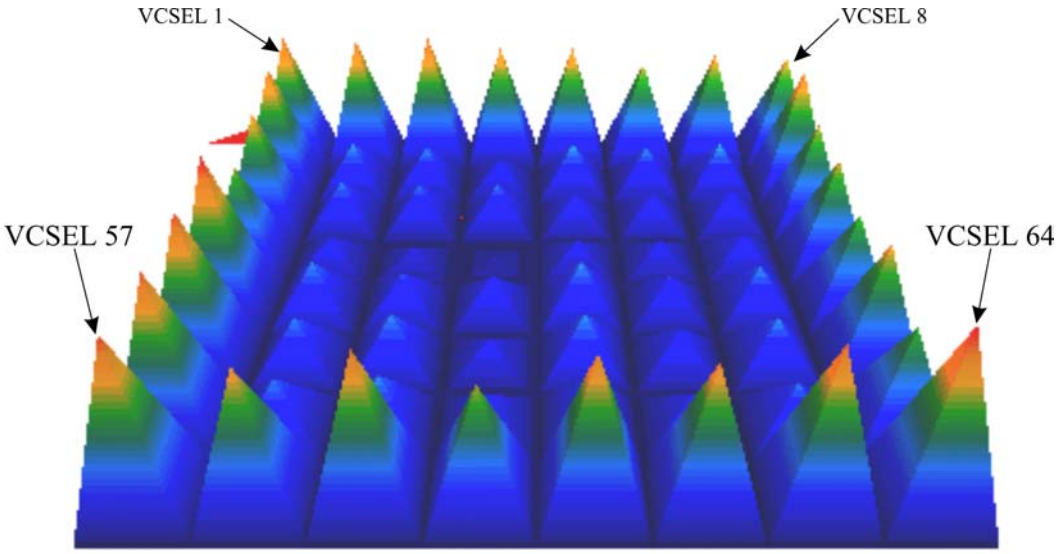

Figure 120: System Optical Input
The height of a triangle represents the magnitude of optical signal incident on a detector. This measurement was taken with the four corner VCSELs active. Snapshot from PC software using OpenGL for visualisation.

Measurements were then taken with VCSEL 1, VCSEL 8, VCSEL 57 and VCSEL 64 simultaneously active. Each clearly shows the cross pattern generated by the installed crossbar DOE. This data was then analysed and adjusted using the previously measured zero level. It showed that an inhibitory signal produced a voltage deflection of 117mV on average, previously theoretically predicted as 116mV. Unfortunately, optical system design results in a zero order, or neuron self inhibition, of 50mV on average. However, this can again be compensated for by using the same technique as the zero order since the DSP has plenty of free cycles. DSP noise is unfortunately still present, however it has been reduced to ± 22 mV in the extreme case. Another important measure is inter-channel cross-talk. This is only present in physically adjacent optical channels and has been measured as 11mV on average, i.e. 9.4% of full scale channel optical power.

The synchronisation of DSPs was assessed and it was concluded that regular resynchronisation is unnecessary. After 1 second the difference between two DSP modules is typically 9.0µs ±2.7µs. This should not be significant where evolution times of 400 iterations are involved as the difference after this period of time is a fraction of a clock cycle in the worst case.

The VCSEL array has been driven at 2MHz experimentally. Unfortunately this is superior to that of the detector array, which is directly limited by the transimpedance amplifier due to a high level of amplification. Careful DSP programming yields an input bandwidth of 250kHz as 2MSPS ADC are multiplexing eight channels. Therefore we can calculate the CPS rating of the second generation neural network demonstrator as 224×10^6 CPS in crossbar switch configuration, 304×10^6 CPS in a Banyan switch configuration and 1×10^9 CPS in a fully interconnected configuration.

5.3.9 Conclusion

The design and construction of both systems have been an evolutionary process. Theory and experiment have given an increasing understanding of the circuitry coupled with improvements which enhance both bandwidth and response in present and future iterations.

The second generation demonstrator enhanced optical system characteristics considerably. Non-uniformity is now so low that systems of N=30 can be constructed using current techniques. Indeed, the entire active free space optical system will fit into a tube 25mm in diameter and 187mm long and it is still possible to reduce this volume.

5.4 Further Work

This section suggests short, medium and long term enhancements for the neural network demonstrators.

First of all, there are three areas that could be quickly and easily improved in the short term. All of these solutions are simple component substitutions that improve performance, with replacement components probably even pin compatible. First is the transimpedance amplifier which requires a higher speed op-amp in place of the existing low speed component. Second are the ADCs which suffer from serialisation of data. Replacement of only this component could instantly treble neural network performance. The third is the DSP array as additional processing power is always welcome.

In the medium term, serialisation should be completely removed from the electronic system with the replacement of the current four DSPs with either a SIMD processor or an array of fine grained processing elements with one dedicated to each channel. Another possible addition may be a current offset for each VCSEL to allow pre-biasing above threshold, decreasing turn-on times and compensating for erroneous VCSEL values. A programmable current source could be used in combination with the DSPs such that the system is calibrated and correctly biased at startup. This was not implemented as not only did it introduce unnecessary complexity to what is essentially a demonstrator but VCSEL rise times and closely matched resistances were such that it was not deemed necessary.

More long term goals involve integration of the entire neural network onto a single chip containing optical input, processing and optical output stages. Electronic interconnection from optical destination to optical source is surmountable in either one of two ways as shown in Figure 121.

The *folded system* uses a single optoelectronic neural network chip and returns data to the displaced location on the chip after performing interconnection. Optical system length is halved but volume remains constant. The *pipelined system* is another construction option which uses two neural network chips that simultaneously perform two independent calculations. The viability of such optical hardware has already been demonstrated in a robust manner where a shock hardened and sealed module withstood $\pm 50 GHz^{-1}$ at 125Hz for several seconds [164].

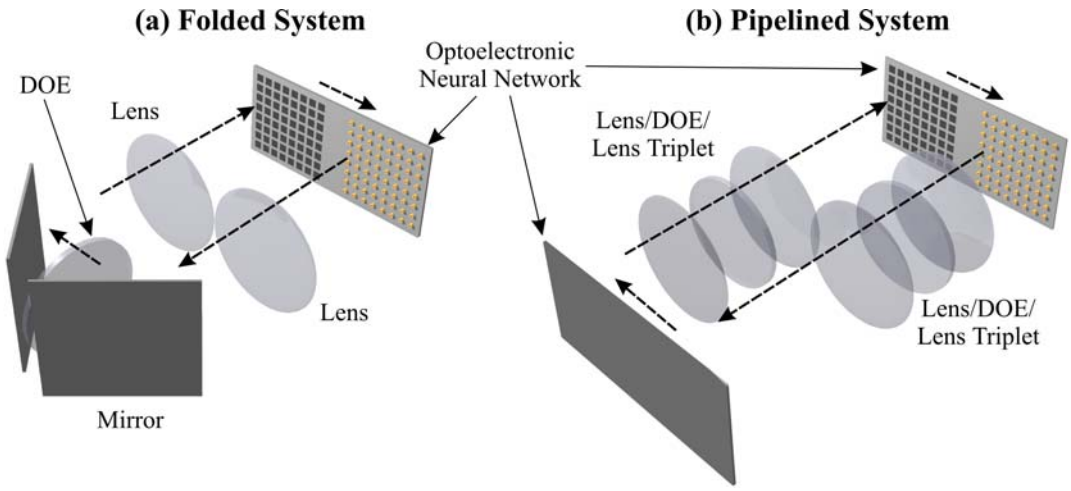

Figure 121: Long Term Neural Network Integration
The integrated optoelectronic neural network contains all necessary components for an operational neural network. Information flow is indicated by the dashed lines. Diagrams not to scale.

An important measure that has not been made is quality of service, mainly due to a lack of accurate network traffic models. Such analysis will conclusively define the future of the optoelectronic neural network scheduler.

5.5 Conclusion

The construction of two fully operational demonstrators has illustrated the potential of optical interconnection to enable architectures precluded by electronics alone. The architecture behind these systems is itself unique yet without optoelectronics its construction is simply not feasible. Scalability of these system has also proven to be unprecedented. If the system was scaled to the next size up where N=16 and run at 100MHz, rather conservative for optical components, the fully interconnected neural network reaches a ground-breaking performance of 6.5 Terra connections per second.

6 Optically Interconnected FPGAs

This chapter will examine the evolution of reconfigurable hardware to its current embodiment, specifically the *field programmable gate array* (FPGA). The FPGA is normally reconfigured by an external controller to implement a desired circuit. Reconfiguration can be performed with speeds at or near real time, depending on both the extent of reconfiguration and the time taken to download configuration data. As with all VLSI systems, the increasing density and speed of silicon circuits frequently transfers the performance bottleneck in any system to its communications, with FPGAs no exception. To make dynamically reconfigurable computing at all viable, new FPGA configurations must be downloaded at a rate which puts the component out of action for the shortest possible time period. Optoelectronic interconnects are widely considered as a potential solution to the interconnection problem and are already being deployed at a crude level in many commercial systems, but conspicuously not for the FPGA. Aside from the potential of optics for raw data throughput simply unattainable in conventional systems, what is perhaps more exciting is the enabling of new architectural concepts by a combination of this throughput with the potential to reconfigure at high speed.

The addition of optical I/O opens up relevant areas in high bandwidth reconfigurable computing previously excluded by the low bandwidth I/O of high density FPGAs. This chapter will consider FPGA architectures along with their application in the field of reconfigurable computing. Two case studies are examined along with their implementation using dynamically reconfigurable optically interconnected FPGAs.

6.1 Reconfigurable Hardware

Reconfigurable hardware can be categorised into one of four classifications, presuming that a device has some inherent level of configurability. This chapter adopts the terminology defined in [165].

- *Configurable hardware* can be configured one or two times. An example of such hardware is a standard circuit board produced by a manufacturer for a specific product line. Although the board supports all product versions from the low to high ends, its exact function is configured by adding the appropriate components and modifying sets of switches or jumpers.

- *Reconfigurable hardware* has the advantage that it can be reconfigured many times to suit the task on hand. Such systems usually contain components such as *programmable logic devices* (PLDs), which act as truth tables, *Boolean* equations or state machines, or *programmable read-only memories* (PROMs) which can be used as non-volatile memories for microprocessors or microcontrollers. Some forms of reconfigurable hardware are *in-system programmable* (ISP) and can be re-programmed while still resident on the circuit board.
- *Dynamically reconfigurable hardware* emerged with the advent of *static random access memory* (SRAM) based FPGAs and opened up another chapter in reconfigurable hardware. The SRAM cells allowed an FPGA's logic to be configured by simply loading a bit pattern into the SRAM. Thus a system could be quickly reprogrammed to perform any appropriate task. For example, on start-up the FPGA could be configured to perform diagnostics on itself and its circuit board before dynamically reconfiguring to perform the main task(s) for which it was designed.
- *Partially reconfigurable hardware* addresses the problem with the majority of dynamically reconfigurable hardware in that in order to reconfigure the device, its operation needs to be halted and the entire SRAM contents reloaded resulting in the irretrievable loss of any data in FPGA registers. This led to the development of a new type of FPGA which supported dynamic reconfiguration of selected portions of internal logic with no disruption to the device's I/O or system level clocking and enabled continued operation of portions of the device not undergoing reconfiguration. One feature of particular benefit is that the contents of internal registers are not lost during reconfiguration. This allows one instantiation of a function to hand over data to a new instantiation of another function.

6.2 Device Evolution

There are many methods at a designer’s disposal for integrating discrete components onto a single VLSI ASIC [166], each method having a specific capacity, performance, flexibility and unit cost in both time and money. This section outlines the functions of various technologies with the aim of highlighting new opportunities presented to designers through the use of FPGAs.

A *full custom* system is where the entire VLSI circuit is carefully tailored to the designer's requirements. This results in a high performance chip with optimal silicon usage and is sometimes the only method of implementation considered for certain applications such as state-of-the-art microprocessors. Fabrication costs and initial set-up times for full custom solutions are also high since the flexibility afforded to the designer results in chip design being complex and time consuming.

A *standard cell* design sacrifices the flexibility and performance of a full custom design to speed up the design process. This is done by using a specific set of design restrictions and standard format cells enabling the use of software tools to automate the design process. A standard cell method predefines all gates as a series of cells with the same height and placement of power and ground lines as shown in Figure 122.

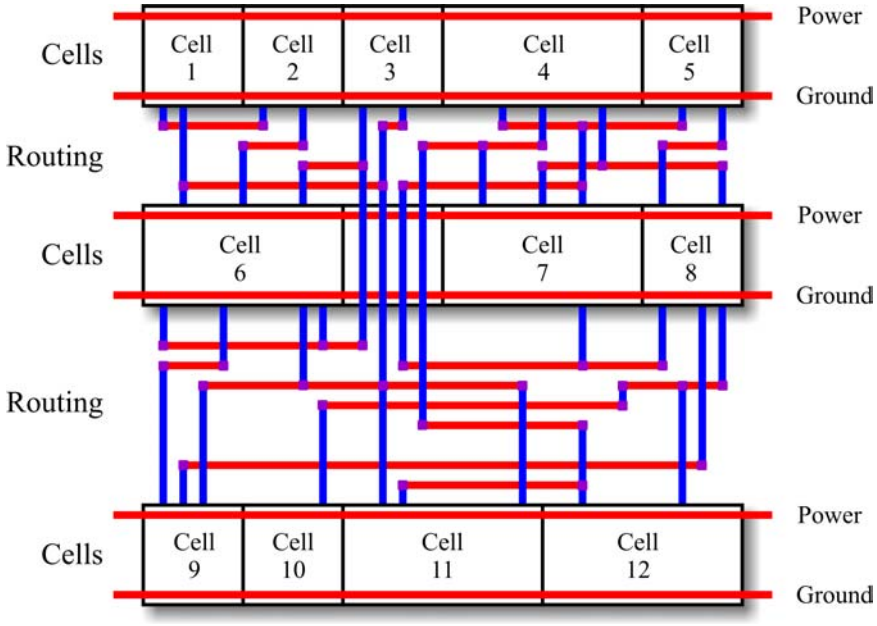

Figure 122: Standard Cell Layout

Sample routing of a series of standard cells. Numbered cells have a programmed functionality whereas routing cells, such as the one between Cell 6 and Cell 7, serve to transfer information between rows.

The chip is laid out in interleaved logic and routing rows, the latter being as large as is necessary to accommodate all routing. Routing between routing rows is accomplished using higher metal layers or by route through cells. Essentially, the designer need only specify functionality and software will create the layout. Fabrication of a standard cell is done by a silicon foundry, resulting in approximately the same fabrication times as full custom designs.

The problem with both full custom and standard cell designs is that they have to be fabricated from scratch. In an *mask programmable gate array* (MPGA) most of the fabrication is carried out beforehand. For example, an MPGA may consist of sets of transistors in specific locations which are partially interconnected to form the basis for mapping certain types of logic gate. Thus all any designer need do is interconnect the required elements to give the desired functionality. This method reduces both cost and time of fabrication as a foundry can store large amounts of standard design, partially fabricated chips. The designer still has a great deal of flexibility and is only limited by the amount of routing available on the chip. Unfortunately it is inevitable that there will be inefficiencies as some designs require more or less of a specific resource or routing bandwidth than are available on chip.

One of the first general purpose pre-fabricated chips to achieve widespread use was the *programmable read only memory* (PROM). It consists of an array of one time write and thereafter read only cells pre-programmed with a truth table function. Information on a PROM is normally stored using one of three methods:

- *Fuses/antifuses* can be used to store bits, the condition of which determining the bit value read. They are blown by applying a high voltage.
- *Erasable* PROMs (EPROMs) are programmed by application of a high voltage but differ in the fact that it can be erased by exposure to *ultraviolet* (UV) light. This allows the EPROM to be reprogrammed.
- *Electrically erasable* PROMs (EEPROMs) are again programmed using a high voltage but this time the chip can be erased by purely electrical means.

One other technology well worth a mention here, but not strictly classified as a PROM, is *random access memory* (RAM). Its characteristics are essentially identical to a PROM except that it is can be both read and written and therefore reprogrammed in system on the fly.

Programmable logic devices (PLDs) were specifically designed to implement logic circuits and typically consist of an array of AND gates followed by an array of OR gates in a sum-of-products.

- *Programmable array logic* (PAL) is the most common type of PLD with a programmable AND plane followed by a fixed OR plane.

- *Programmable logic arrays* (PLA) are a more flexible version of the PAL which allows both AND and OR planes to be programmed.
- *Generic array logic* (GAL) is a further enhancement of the PLD which can be configured to implement several different types of PAL with optional output inversion.
- *Complex programmable logic devices* (CPLDs) break up complex systems by effectively connecting several PLD elements together using a switching matrix.

PLDs can again be implemented using fuse/antifuse, EPROM or EEPROM programming techniques.

Field programmable gate arrays (FPGAs) are fully prefabricated devices designed to implement multi-layer circuits instead of simple PLD sum-of-product terms with the only limit on chip circuit complexity being available resources. Unfortunately, this results in a less predictable propagation delay. It should be noted that FPGAs are sometimes referred to as CPLD devices, however to avoid ambiguity they will always be referred to here as FPGAs. FPGAs are not only available in fuse/antifuse, EPROM and EEPROM technologies, but also in SRAM and *dynamic* RAM (DRAM) versions [167]. SRAM and DRAM versions allow fast in-circuit reconfiguration of the chip but are disadvantaged by the amount of silicon required by RAM technology. Reconfiguration can either be partial or complete, which is device dependent, and need not even result in the disruption of logic blocks which are not being reconfigured.

6.3 The FPGA

The FPGA [168]-[169] was first introduced in 1985 by Xilinx and since then several other companies have released similar products. This section examines standard FPGA architectures and characteristics.

6.3.1 FPGA Classes

There are four main classifications of FPGA layout as shown in Figure 123. All of these four architectures are commercially available, each describing the positioning of both routing and logic blocks. Although interesting on a chip layout level, when designing a logic circuit sophisticated CAD tools are used which translate the design therefore rendering the underlying architecture transparent. Architecture choice is dependent on application, some being more suited to a particular task than others.

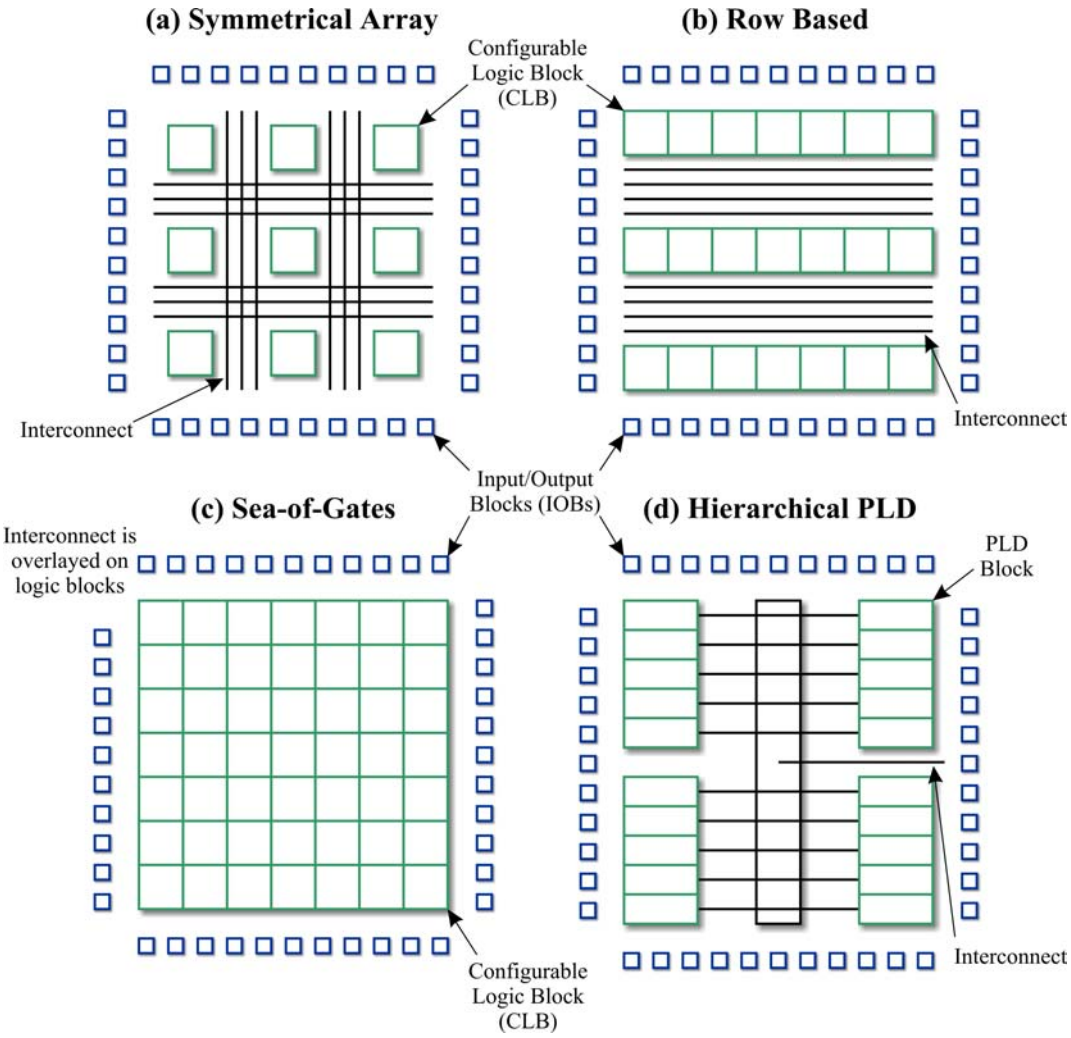

Figure 123: Main FPGA Classes
This diagram shows the four main commercially available FPGA classes. Input/output blocks (IOBs) provide the interface between package pins and internal signal lines.

6.3.2 The Configurable Logic Block (CLB)

Configurable logic blocks (CLBs) provide the functional elements for construction of a user's logic design in an FPGA. The CLB's most important figure of merit is its functionality. Increased functionality allows more complex logic functions to be implemented in a single CLB, however as the functionality of a single CLB increases so does its size, resulting in fewer CLBs on each FPGA. A single CLB contains a number of standard building blocks, arranged in a manufacturer specific manner, to allow implementation of various logic functions. The building blocks are:

- *Look-up tables* (LUTs) take *M* Boolean inputs and give *N* outputs, where *N* is usually 1. The outputs given are chosen to represent an appropriate function. This thesis will adopt *M*-LUT-*N* to identify a LUT's properties: e.g. a 3-LUT-1 is a three input

LUT with one output. Such a look-up table could be used to implement a 3 input AND gate as demonstrated in Table 18. The number of output bits which need to be stored for any lookup table is a direct function of the inputs: 2^M. Thus a 4-LUT-1 requires 16 stored bits in contrast to the 3-LUT-1 which requires 8. The problem with lookup tables is that larger units tend to remain under-utilised. It has been shown that, for most applications, the optimum LUT size is a 4-LUT-1 [168].

- *Programmable logic arrays* (PLA) were introduced for the very reason that LUTs are inefficient with dimensions of $M>5$. The area efficiency of PLAs was examined with M inputs, N outputs and K product terms and the optimal device shown to have values of approximately $K=10$, $M=3$ and $N=12$ [170]. This gives a 4% space to functionality advantage over a 4-LUT-1 implementation.
- *D-type flip-flops* are extremely useful, although not necessary, when any kind of sequential logic operation needs to be performed. It has been shown experimentally [171] that without a flip-flop in a CLB the number of CLBs required to implement a function approximately doubles.

Input *M*			**Output *N***
A	***B***	***C***	***x***
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Table 18: 3-LUT-1
Sample output for a 3 input 1 output look-up table configured to act as a three input and gate.

CLBs can contain other components such as logic gates, buffers and multiplexers, however there is no standard implementation. Figure 124 illustrates a sample 4-LUT-1 based CLB.

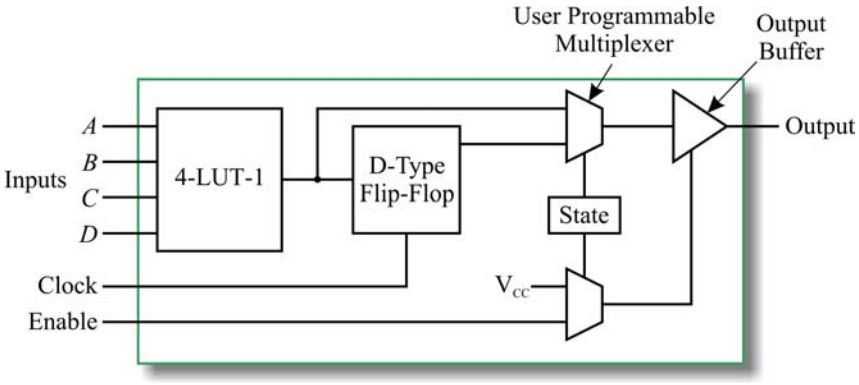

Figure 124: Configurable Logic Block (CLB)
A sample CLB design taken from [171].

The problem that any manufacturer has when designing an FPGA is trading off CLB granularity against functionality. One possible way round this problem is to create a non-homogeneous array of logic blocks, however such an array will almost always have elements that are not suited to a given application. Regardless of CLB configuration, the final efficiency of an FPGA is highly reliant on its CAD tools. If the CAD tools do not map a logic circuit optimally onto the CLBs then any architectural advantage will be squandered.

6.3.3 FPGA Routing

FPGA routing deals essentially with the connection of CLBs together with other CLBs and *input/output blocks* (IOBs) thus providing the necessary functionality. The balance between area given to routing and to CLBs is critical: too much to CLBs and it will not be possible to wire complex systems together, too little and routing will remain unused. On a standard FPGA, routing will normally occupy 70%-90% of all the available area. Routing is expensive in terms of area and delay since the programmable switches take up a significant area and have appreciable resistance and capacitance.

There are two types of block used in routing on an FPGA which are defined as *connection blocks* (CBs) and *switch blocks* (SBs). Figure 125 shows how a symmetrical CLB array architecture is constructed using these two additional and essential blocks. The internal connections possible in both CBs and SBs are again manufacturer dependent. A manufacturer must consider the amount of routing its architecture

requires and optimise CB and SB connectivity to allow maximum interconnection flexibility whilst keeping redundancy to a minimum.

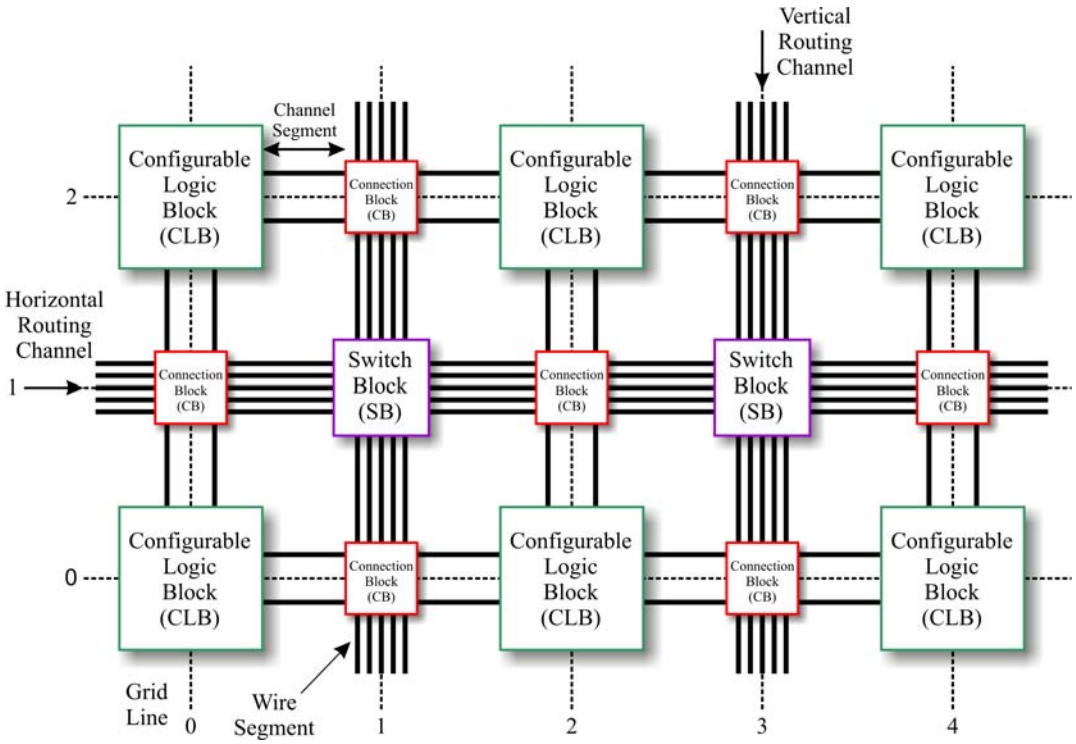

Figure 125: Symmetrical Array Routing

Connection blocks (CBs) and switch blocks (SBs) can be used to route I/O from CLB elements into routing channels.

A sample connection block is shown in Figure 126. It indicates a configurable routing switch point using a cross, the CB itself having a predefined interconnection topology.

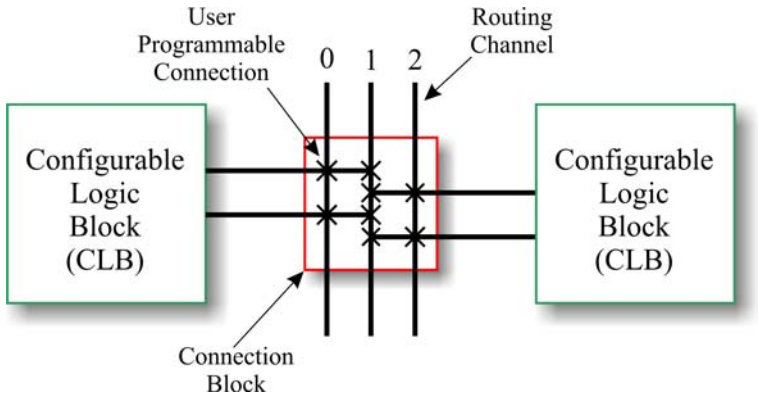

Figure 126: Connection Block

A sample connection block where the crosses represent possible connections.

Flexibility is an important issue here as too few routing switches or programmable connections could result in it being impossible to interconnect two CLB pins.

Figure 127 shows a sample switch block.

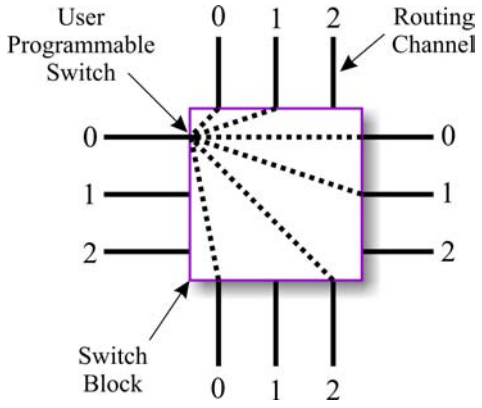

Figure 127: Switch Block
Sample switch block with all possible switch configurations from one channel shown.

The flexibility *F* of a switch block is defined as the number of wire segments any incoming wire segment can be connected to, in the example shown this is *F*=6. When designing a switch block topology the manufacturer must again choose a topology that does not prohibit two CLB pins from being connected together.

Routing resources in an FPGA are normally classified into one of three categories thus defining their primary use. These classifications are:

- *General purpose interconnect* which is used for connections that span one or more local CLBs. It is implemented using routing channels, connection blocks and switching blocks as described above. Unfortunately, each switch or connection block through which a signal must pass has an associated RC delay resulting in signal propagation issues at higher device frequencies.
- *Direct interconnect* provides a direct connection with one or all of a CLB's neighbours to either the right, left, top or bottom.
- *Long lines* are used to route connections that need to span several CLBs with low skew providing a partial solution for signals that would otherwise traverse several routing switches incurring cumulative RC delays.

One final but important consideration in FPGA routing is IOBs, without which an FPGA would be useless. IOBs must be chosen such that they support multiple output modes such as *transistor-transistor logic* (TTL) or *complementary metal-oxide semiconductor* (CMOS) and provide enough, but not excessive, drive current to allow any attached device to function correctly. Unfortunately I/O is the weak link in FPGA design and has scaled disproportionately to chip density.

6.3.4 Commercially Available FPGAs

Device	Architecture	SRAM (bits)	IOBs	Ref.
Xilinx XC6264	Sea-of-Gates, FPGA.	16,384	512	[172]
QuickLogic QL4090	PLD.	25,344	316	[173]
Dynachip 6055	Symmetrical Array, FPGA.	51,200	320	[174]
Lucent Orca OR3T165	Sea-of-Gates, FPGA.	134,144	512	[175]
Altera Apex EP20K1000E	Symmetrical Array, PLD.	540,672	780	[176]

Table 19: Commercial FPGAs
Examples of FPGA technology as of June 1999.

To illustrate commercially available technologies, Table 19 contains some sample FPGAs as of June 1999. Note that the trend of increasing SRAM capacity in the detailed devices is not matched by a corresponding increase in the number of IOBs. Although the architectures are obviously different, SRAM will still need to be configured and thus an equivalent amount of data must be downloaded using proportionally fewer IOBs. In addition, larger chips have a larger number of components on the chip fabric, therefore the probability increases that a signal will have traverse more SBs and CBs on route to an IOB introducing even more delays.

6.4 Dynamically Reconfigurable Computing

Dynamically reconfigurable field programmable gate arrays (DRFPGAs) combine, in principle, the speed of a dedicated hardware solution with the flexibility of software [177]-[179]. Such FPGAs can be reconfigured by an external controller to implement any desired set of Boolean operations [180]. This is the concept behind dynamically reconfigurable computing: to combine the flexibility of software with the speed of hardware [181]. At present there is growing interest in the technology with two companies, Triscend [182] and Starbridge Systems [183], both releasing computer systems that take advantage of dynamically reconfigurable computing. Although the field is still emergent [184], this section examines the criteria which are used to classify a reconfigurable computer system [185]-[186].

There are normally two reasons cited for the adoption of dynamic reconfiguration:

- *Speed improvements* through application of a custom architecture to a specific problem resulting in increased performance.
- *Fault tolerance* which improves the manufacturing process by allowing design alterations or problems to be easily overcome.

The size and complexity of the smallest block in any reconfigurable device is classified by its *granularity*:

- *Fine grained* architectures consist of small and simple logic blocks that are configured to perform more complex operations.
- *Medium grained* architectures consist of complex logic blocks that each perform a significant part of any calculation. Non-standard calculations are enabled through reconfiguration of the interconnection network.
- *Coarse grained* architectures consist of a number of execution units each with their own set of instructions. Every unit is simpler than a microprocessor but integrated tightly enough to give high speed communication [187]-[188].

Device integration specifies how closely any reconfigurable system is coupled to its host.

- *Dynamic systems* are bio-inspired systems which are not controlled by an external device. The idea is that such systems evolve themselves.
- *Static, closely coupled* systems bind reconfigurable elements closely as execution units on a host processor's datapath.
- *Static, loosely coupled* reconfigurable units are situated on a separate board from the host. This is generally detrimental to any speedup as data must be transferred to and from any daughterboard.

The reconfigurability of an *external interconnection network* between reconfigurable units can also be classified into one of two categories:

- A *reconfigurable external network* extends the concept of reconfiguration over several reconfigurable circuits effectively providing a large reconfigurable unit. However, the penalties for going off-chip are high.
- *Fixed external networks* use static connections between reconfigurable circuits effectively limiting flexibility in order to maximise speed and minimise cost.

6.5 Optical Interconnection of FPGAs

Previous chapters have demonstrated the advantages of optics, but these are general advantages and not specific to dynamically reconfigurable systems. There are two compelling reasons why optical interconnects are particularly of interest to DRFPGAs:

- *Bandwidth*: FPGAs are routed dynamically and each I/O channel must therefore traverse both switching and routing blocks, each with an associated RC delay, to reach an I/O block. This can result in serious bandwidth limitations
- *Reconfiguration*: To make dynamically reconfigurable computing at all viable, new FPGA configurations must be downloaded at a rate which puts the component out of action for the shortest possible time period. If highly parallel optical I/O was used to download new configurations, reconfiguration times could be minimised thus making best use of all available processing time.

The advantages of optical interconnection in such systems lies in locality of data. Large FPGA arrays must always communicate with edge points if conventional techniques are used, however optics inserts or extracts data streams at a variety of points normal to the surface of the chip. Since routing occupies between 70%-90% of chip area, optical interconnects could ease this problem by freeing up more area for CLBs while preserving sufficient interconnectivity to maintain functionality. Not only does this reduce the amount of routing required but it increases the potential bandwidth available to the chip as a whole since data need no longer traverse multiple blocks to reach its destination. Indeed, any optical solution would probably enhance rather than replace existing electrical I/O. For instance, as the SRAM capacity of larger FPGAs increases, optical I/O could be used to rapidly reconfigure the entire device leaving electrical I/O available to implement device functionality.

The systems proposed here consist of three basic stages. The first stage we will consider as the input stage which is a detector capable of receiving digital optical input be it from free space or waveguide. The second stage is the processing stage and consists of a dynamically reconfigurable FPGA system which could be anything from a single CLB, as shown in Figure 128:

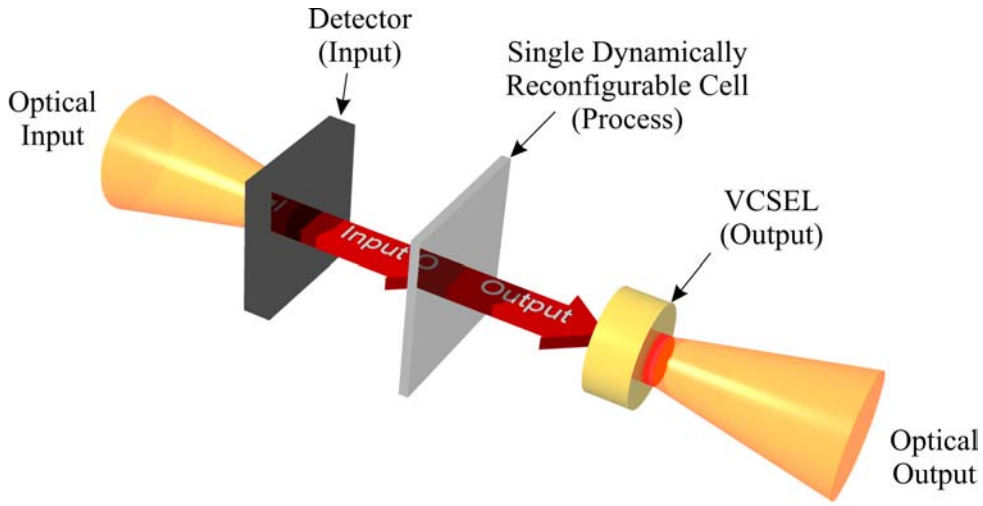

Figure 128: Single CLB Element

Element with one dynamically reconfigurable CLB or gate array. Model valid in coarse grained architectures.

To an array of CLBs, as shown in Figure 129:

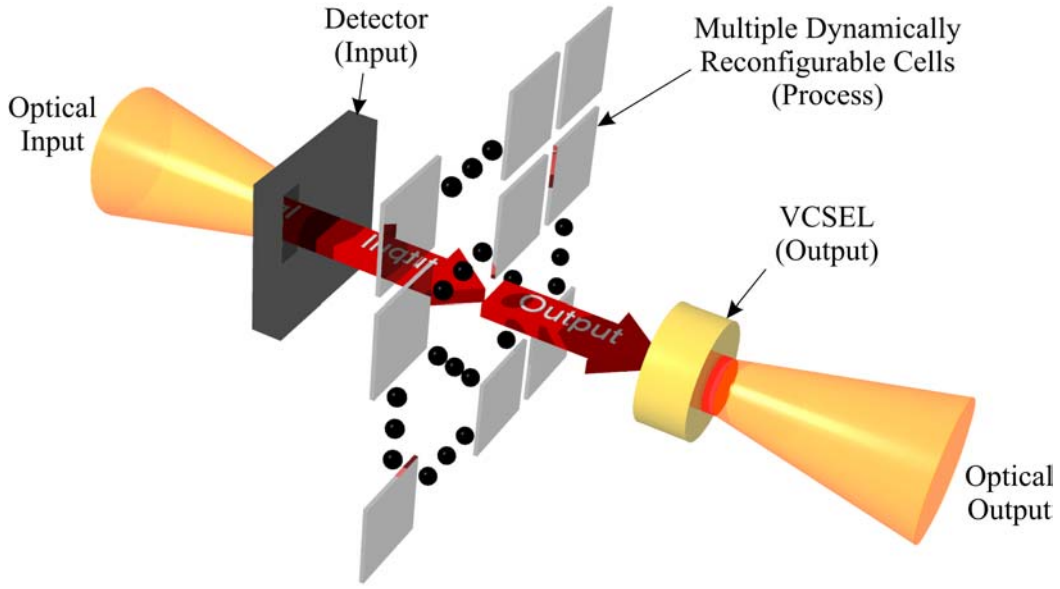

Figure 129: Multiple CLB Elements

Element with multiple dynamically reconfigurable CLBs or gate array. Model valid in fine grained architectures.

The number of CLBs mapped to optical input and output channels is dependent on device granularity. A single CLB element would normally have its own optical communications channels if the architecture was coarse grained, whereas multiple CLBs would share optical channels in fine grained configurations.

The final stage is the output stage and consists of an optical emitter such as a VCSEL or MQW modulator with digital output to either free space or a waveguide. This combination of stages will be referred to as an *element*.

To make full use of parallelism, elements are arrayed in two dimensions with one detector array, one FPGA and one VCSEL array. The FPGA processing stage of each element is capable of communicating with another electronically, or with any other local electronics for that matter, and can be considered as a standard dynamically reconfigurable FPGA with an extra optical input and output available to a specific CLB or set of CLBs. Any CLB has the potential to be reprogrammed by its optical input stream if configuration information is interlaced using a predefined protocol, or to reprogram another CLB in another system by interlacing the same configuration information onto its optical output stream.

It is presumed that any optical information is converted by peripheral interfacing hardware into an electronic data stream. Previous chapters have examined the nuances of such hardware. Interfacing of optical I/O channels can be considered as an additional connection to any component on an FPGA. Connecting the channel directly to a CLB is not considered an optimal solution as the full functionality of the CLB will probably not be used under such circumstances. Thus the CLB would become an additional interface unless system granularity has been carefully adjusted such that there is no wastage of functionality. However, such optimisation defeats the generality that defines reconfigurable computing. Mapping an optical I/O interface as an additional crosspoint on an SB or CB appears to be the most efficient method for implementation. This potentially enables multiple CLBs to connect to a single optical channel using the normal routing network, though not simultaneously, giving generalised rather than specific access. However, if optical I/O is used to exclusively download configuration information the channel interface point becomes irrelevant with configurations downloaded to all configurable components.

6.6 Sample Applications

This section presents two examples of systems that benefit from optically interconnected FPGAs. Note that the combination of optical interface integrated with an FPGA will be referred to from now on as an *optical* FPGA (OFPGA).

The first example addressed here is the neural network packet switch controller detailed in this thesis. An experimental version has shown a performance commensurate with state-of-the-art all electronic switches, albeit using discrete components. Optically interconnected FPGAs would replace the electronic neural network with the optical system directly interfacing directly to the FPGA. Assuming that the FPGA is large and fast enough, a switch fabric could be directly integrated onto the device. This application is novel in the first instance in that the packet switch architecture becomes adaptable, allowing architectural alteration under high link load or in the event of link failure. In the second instance, programmable weights would allow the network to be configured for a wider variety of tasks such as the travelling salesman problem or other optimisation problems. This adds the potential to reconfigure weights in near or at real time so that fully adaptive, supervised and unsupervised learning schemes may be implemented. This combination gives us an adaptable and fault tolerant neural network with real time adaptive weights and an interconnection density that could never be achieved exclusively in electronics.

A *generic multiprocessor harness* is another potential application of OFPGAs. Their role in such a system would be to optimise communications and processing in real time during the execution of a range of algorithms. Note that this section extends the optical highway architecture described in section 3.7, examining a node in further detail. The bandwidth of communications in this system is sufficiently high to implement a flat memory model, but superior performance on particular algorithms may be obtained by changing topology in the interconnect harness. In essence, the destination of any data channel output into the optical domain is determined by the spatial location of the emitter output on the OFPGA. Thus by re-routing signals within the OFPGAs, a particular global topology may be established. The machine could of course be configured arbitrarily into several differently connected regions presuming this was desirable. In addition, the functionality of the interface may be changed. OFPGAs allow us to configure the interface as a router, necessary if we wished to utilise a hypercube in the optical domain, or as a switch of high throughput, necessary if we wished to use a large crossbar in the optical domain. For nodes of sufficient complexity, the maximum throughput of the system may often be attained by changing the width of data words as well as the topology so as to keep all communication channels busy. OFPGAs could support variable width multiplexing as well as routing at the interfaces.

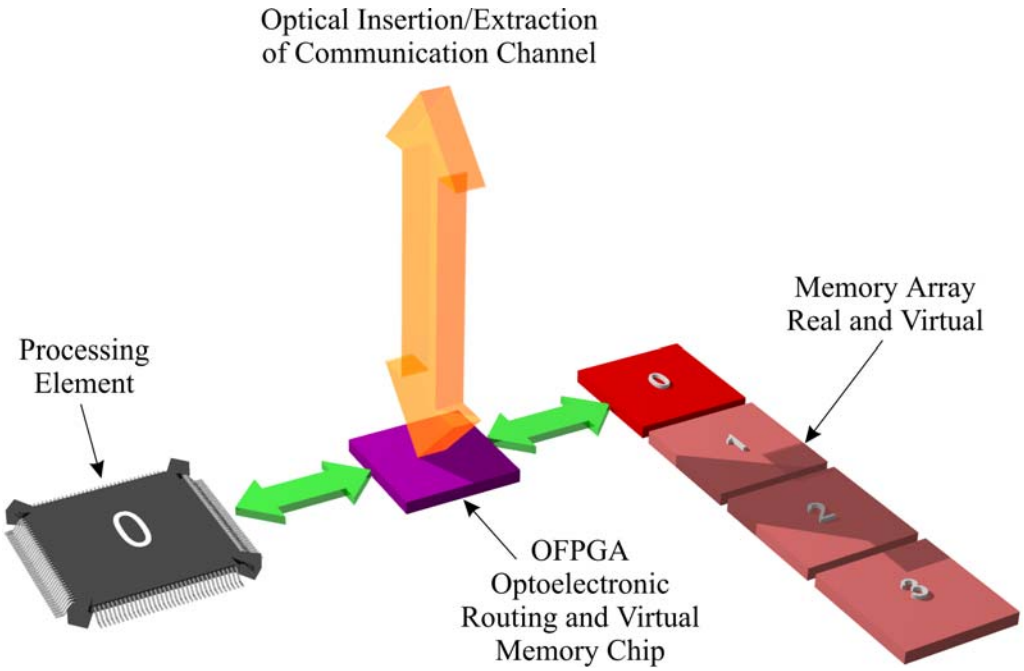

Figure 130: Processing Element or Node

The processing element can be considered as a node in an optical highway. Multiple nodes create a massively parallel machine where the OFPGA dynamically controls topology. Connection of a number of OFPGAs in this manner can be considered as creating a large virtual FPGA.

In a generic interconnect, the combination of FPGAs with high bandwidth optoelectronics enables an intelligent communications interface to be constructed and allows maximisation of the ultra high throughput available. In turn, this facilitates real time optimisation and load balancing of the entire machine over a range of computational models.

An important issue in parallel systems is that of memory organisation with regard to both physical location and address space. The physical location is exactly where the memory is placed. This can be either distributed in chunks next to each processor thus giving fast access times for that processor to its memory or centralised in a particular location resulting in similar access times for all processors. Centralising memory has the advantage that if not all processors are being used the entire memory space is still available to other processors with no inter-processor communication required. However, this reduces overall memory access times due to complicated memory access schemes and the fact that memory can be considered as remote to all processors. The next consideration is address space which is directly tied to the issue of communications. There are two types of address space. *Shared address space* which implies implicit communication and *multiple private address space* which implies

explicit communication. Shared address space systems perform well when communications are fine grain [189] since large blocks of data must be separated into a number of smaller transactions. However, multiple private address space systems are better suited to coarse grain communications as the message overheads need only occur once regardless of the amount of data transferred.

6.7 Conclusion

Dynamic reconfiguration is becoming ever more popular as a means of high speed and flexible processing. As an increasing number of companies begin to release such systems, the issues of bandwidth will continue to become more prominent. The author believes that the only way to overcome the associated bandwidth limitations is to utilise optical interconnection, be it through free-space or waveguides. Optical interconnects are not intended as a replacement for existing electrical interconnect but rather as an enhancement. However, without optical interconnection, dynamically reconfigurable computing will hit a premature performance ceiling.

7 Conclusion

"Accuracy is impossible for all but the most trivial question, but blurred vision is better than none at all."

Ian Pearson, BT Technology Futurologist [190]

Classical computer systems appear to embody obsolete assumptions. Is there any particular reason why processor and memory must remain physically separate? Such architectures are referred to as von Neumann where the processor is differentiated from memory. The bottleneck between both components is becoming ever more pronounced as technology pushes physical limits. Caching schemes attempt to alleviate the problem by adding to the processor a small amount of memory which runs at the same speed. This is essentially an interim solution to a growing problem. Optical technologies are ripe to replace existing electronic interconnection schemes. Perhaps chip densities will soon no longer be relevant since chip-to-chip optical interconnection has already been shown to have a lower latency than long lines across a large substrate.

The connectionist approach seen in neural networks has been proposed as an evolution in computer architecture to the massively parallel domain. This thesis has clearly demonstrated its advantages. Nevertheless, any electronic implementation is still limited by today's technology due to poor interconnection density, a limit that is surmountable if optical interconnection schemes are used.

The author does not see short or medium term adoption of completely optical computing systems where switching is performed in the optical domain. This is mainly due to two reasons. Firstly, there is simply not any interaction between photons. All interaction must take place through a medium which responds to photons in a light-guiding-light manner. However the switching energies are such that the question is raised, why not convert the photons to electrons and utilise existing and proven technologies for switching. This leads on to the second point which is based on commercialisation as investors want to see a return on their investment. Most large companies like to enhance what they have rather than dispose of decades of research and development, in this case the maturing technology of semiconductors, and replace it with something new and unproven. Therefore they will not rapidly replace semiconductors with completely optical technologies. What is likely to happen is that

they will integrate optoelectronics to provide high bandwidth interconnects between their existing semiconductor integrated circuits. It will not be long before PCBs use waveguides rather than metal tracks.

There are various alternative experimental computational architectures on the horizon such as molecular computing, DNA computing and quantum computing [191]-[193]. However, none of these architectures address data locality but they simply provide a faster and sometimes novel way of manipulating data. Indeed, architectures such as quantum computers do not even provide general purpose computing. As there is no challenger with the performance of optical interconnects such future systems will undoubtedly have to use optics just to be able to transfer data at a comparable rate.

Optoelectronics are sneaking in by the back door. They have already superseded electronic interconnection over long distance and are continuing to replace electronics over shorter and shorter distances. It will not be long before electronic chips are directly interfaced to optical systems, closely followed by across chip optical interconnection. Finally, there will only be electronic switching elements between optical transmission lines. The next question then is why do we need that electronic switch at all.

8 Glossary

AC	Alternating current.
ADC	Analogue-to-digital converter.
ANN	Artificial neural network.
AO	Adaptive optics.
APD	Avalanche photodiode.
AR	Anti-reflective.
ASIC	Application specific integrated circuit.
ATM	Asynchronous transfer mode.
BT	British Telecom.
CAD	Computer aided design.
CB	Connection block.
CCD	Charge coupled device.
CCITT	Comité consultatif international téléphonique et télégraphique.
CD	Compact disk.
CLB	Configurable logic block.
CMOS	Complementary metal-oxide semiconductor.
CPLD	Complex programmable logic devices.
CPS	Connections per second.
CPU	Central processing unit.
CSEM	Centre Suisse d'electronique et de microtechnique.
CW	Continuous wave.
DAC	Digital-to-analogue converter.
DBR	Distributed Bragg reflector.
DC	Direct current.
DH	Double heterojunction.

DNA	Deoxyribonucleic acid.
DOE	Diffractive optic element.
DRAM	Dynamic random access memory.
DRFPGA	Dynamically reconfigurable field programmable gate array.
DSK	Digital starter kit.
DSP	Digital signal processor.
Ed(s).	Editor(s).
EEPROM	Electrically erasable programmable read-only memory.
EPAC	Electrically programmable analogue circuit.
EPROM	Erasable programmable read-only memory.
FF	Feed forward.
FIFO	First in first out.
FIR	Far infrared.
FKE	Franz-Keldysh effect.
FP	Floating point.
FPGA	Field programmable gate array.
GAL	Generic array logic.
HOL	Head of line.
HWP	Half wave plate.
I/O	Input/output.
IOB	Input/output block.
ISP	In-system programmable.
ITU	International telecommunication union.
LAN	Local area network.
LASER	Light amplification by stimulated emission of radiation.
LCD	Liquid crystal display.
LD	Laser diode.

LED	Light emitting diode.
LSB	Least significant bit.
LUT	Look-up table.
MEL-ARI	Microelectronics advanced research initiative.
MIMD	Multiple instruction streams, multiple data streams.
MIN	Multistage interconnection network.
MIR	Mid infrared.
MISD	Multiple instruction streams, single data stream.
ML	Maximum likelihood.
MLP	Multi-layer perceptron.
MMIO	Memory mapped input/output.
MPGA	Mask programmable gate array.
MQW	Multiple quantum well.
MSB	Most significant bit.
MTBF	Mean time between failure.
NEP	Noise equivalent power.
NIR	Near infrared.
NN	Neural network.
NOP	No operation.
NOR	Not or.
OFPGA	Optical field programmable gate arrays.
PAL	Programmable array logic.
PBS	Polarising beam splitter.
PC	Personal computer.
PCB	Printed circuit board.
PCI	Peripheral component interconnect.
PD	Photodiode.

PE	Processing element.
PLA	Programmable logic array.
PLD	Programmable logic device.
PROM	Programmable read-only memory.
PS	Packet switch.
PSTN	Public switched telephone network.
QCSE	Quantum confined Stark effect.
QoS	Quality of service.
RAM	Random access memory.
RH	Relative humidity.
SB	Switch block.
SCIOS	Scottish collaborative initiative on optoelectronic sciences.
SIA	Semiconductor industry association.
SIMD	Single instruction stream, multiple data streams.
SISD	Single instruction stream, single data stream.
SLM	Spatial light modulator.
SNR	Signal-to-noise ratio.
SPA	Smart pixel array.
SRAM	Static random access memory.
TEC	Thermoelectric cooler.
TN	Twisted nematic.
TSP	Travelling salesman problem.
TTL	Transistor-transistor logic.
UV	Ultraviolet.
VCI	Virtual channel identifier.
VCS	Virtual circuit switching.
VCSEL	Vertical cavity surface emitting laser.

VLSI	Very large scale integration.
WDM	Wavelength division multiplexing.
WTA	Winner take all.
XOR	Exclusive or.
ZIF	Zero insertion force.

9 References

[1] A. L. Schawlow and C. H. Townes, "Infrared and Optical Masers", *Phys. Rev.*, vol. 112, no. 6, pp. 1940-1949, December 1958.

[2] R. N. Hall, G. E. Fenner, J. D. Kingsley, T. J. Soltys and R. O. Carlson, "Coherent Light Emission From GaAs Junctions", *Phys. Rev. Lett.*, vol. 9, no. 9, pp. 366-368, November 1962.

[3] N. Holonyak Jr. and S. F. Bevacqua, "Coherent (Visible) Light Emission From $Ga(As_{1-x}P_x)$ Junctions", *Appl. Phys. Lett.*, vol. 1, no. 4, pp. 82-83, December 1962.

[4] M. I. Nathan, W. P. Dumke, G. Burns, F. H. Dill Jr. and G. Lasher, "Stimulated Emission of Radiation From GaAs p-n Junctions", *Appl. Phys. Lett.*, vol. 1, no. 3, pp. 62-64, November 1962.

[5] T. M. Quist, R. H. Rediker, R. J. Keyes, W. E. Krag, B. Lax, A. L. McWhorter and H. J. Zeiger, "Semiconductor Maser of GaAs", *Appl. Phys. Lett.*, vol. 1, no. 4, pp. 91-92, December 1962.

[6] R. L. Tricker, *Optoelectronic Line Transmission: An Introduction to Fibre Optics*, Heinemann Newnes, ISBN 0434919780, 1989.

[7] National Wire and Cable Corp., *Basic Introductory Information on Capacitance of Shielded Wires and Cables*, July 1999, http://www.nationalwire.com/cdg/basicintro3.html.

[8] R. S. Mroczkowski, "Interassembly and Intersystem Interconnect", Eds. R. J. Hannemann, A. D. Kraus and M. Pecht, *Physical Architecture of VLSI Syst.*, John Wiley & Sons Inc., pp. 334, 1994.

[9] Alcatel, *Design of Optical Fibers*, section 2, August 2001, http://www.alcatel.com/opticalfiber/catalog/fiches/tech/fiber/design.pdf.

[10] J. Hardy, *High Frequency Circuit Design*, Reston Publishing Company Inc, Virginia, ISBN 0835928241, pp. 186-190, 1979.

[11] H. W. Johnson and M. Graham, *High-Speed Digital Design*, Prentice Hall, ISBN 0133957241, pp. 166, 1993.

[12] PCI SIG Industry Standards Organisation for PCI Architecture, *Conventional PCI V2.2*, http://www.pcisig.com/specifications/conventional_pci, September 2001.

[13] P. Horowitz and W. Hill, *The Art of Electronics*, Second Edition, Cambridge University Press, Cambridge, ISBN 0521370957, 1980.

[14] E. Hecht, *Optics*, Second Edition, Addison-Wesley, ISBN 0201116111, pp. 375-376, 1987.

[15] D. A. B. Miller, "Optics for Low-Energy Communication Inside Digital Processors – Quantum Detectors, Sources and Modulators as Efficient Impedance Converters", *Opt. Lett.*, vol. 14, pp. 146–148, 1989.

[16] G. I. Yayla, P. J. Marchand and S. C. Esener, "Speed and Energy Analysis of Digital Interconnections: Comparison of On-Chip, Off-Chip and Free-Space Technologies", *Appl. Opt.*, vol. 37, no. 2, pp. 205-227, January 1998.

[17] Eds. P. Malinverni and A. Frochel, *Technology Road Map: Optical Interconnects for Integrated Circuits*, European Commission, ESPRIT programme – MEL-ARI OPTO, http://www.cordis.lu/esprit/src/melop-rm.htm, 1998.

[18] M. T. Bohr, “Directions and Challenges in Integrated Circuit Scaling", Intel Corporation, PREP 2000 Conf., Nottingham, UK, April 2000.

[19] C. K. Cheng, J. Lillis, S. Lin and N. Chang, *Interconnect Analysis and Synthesis*, Wiley & Sons Inc., ISBN 0471293660, pp. 99-160, 2000.

[20] D. A. B. Miller, H. M. Ozaktas, "Limit to the Bit-Rate Capacity of Electrical Interconnects from the Aspect Ratio of the System Architecture", *J. of Parallel and Distributed Computing*, vol. 41, no. 1, pp. 42–52, 1997.

[21] H. S. Stone, "Parallel Processing with the Perfect Shuffle", *IEEE Trans. on Computers*, vol. C-20, no. 2, February 1971.

[22] M. J. Quinn, *Parallel Computing Theory and Practice*, McGraw-Hill, ISBN 0071138005, 1994.

[23] E. Griese, "Optical Interconnections on Printed Circuit Boards", OSA Technical Digest of *Optics in Computing 2000*, Quebec City, Canada, pp. 60-71, June 2000.

[24] Essex Corp., http://www.essexcorp.com, August 2001.

[25] G. E. Moore, "Cramming More Components Onto Integrated Circuits", *Electronics*, vol. 38, no. 8, April 1965.

[26] Semiconductor Industry Association, *International Technology Roadmap for Semiconductors*, http://www.semichips.org, 1999.

[27] J. Gourlay, T. Y. Yang, M. Ishikawa and A. C. Walker, "Low-Order Adaptive Optics for Free-Space Optoelectronic Interconnects", *Appl. Opt. IP*, vol. 39, no. 5, pp. 714-720, February 2000.

[28] R. K. Tyson, *Principles of Adaptive Optics*, Academic Press Inc., ISBN 0127059008, 1991.

[29] E. W. Williams, *The CD-ROM and Optical Disk Recording Systems*, Oxford University Press, 1994.

[30] J. E. Flood, *Telecommunications Switching, Traffic and Networks*, Prentice Hall, ISBN 0130333093, pp. 1-15, 1995.

[31] International Telecommunication Union, Telecommunication Standardization Sector, http://www.itu.int/ITU-T/, August 2001.

[32] B. Brockmeyer, H. L. Halstron and A. Jensen, *The Life and Works of A. K. Erlang*, Copenhagen Telephone Company, 1943.

[33] J. Dunlop and D. G. Smith, *Telecommunications Engineering*, Chapman and Hall, ISBN 0412562707, pp. 313-396, 1994.

[34] F. Halsall, *Data Communications, Computer Networks and Open Systems*, Addison-Wesley, Fourth Edition, ISBN 020142293X, pp. 423-482, 1995.

[35] H. Ahmadi and W. E. Denzel, "A Survey of Modern High-Performance Switching Techniques", *IEEE J. on Sel. Areas in Communications*, vol. 7, no. 7, pp. 1091-1103, September 1989.

[36] V. E. Benes, "Optimal Rearrangeable Multistage Connecting Networks", *Bell Syst. Tech. J.*, vol. 43, pp. 1641-1656, July 1964.

[37] C. Clos, "A Study of Non-Blocking Switching Network", *Bell Syst. Tech. J.*, vol. 32, pp. 406-424, March 1953.

[38] L. R. Goke and G. J. Lipovski, "Banyan Networks for Partitioning Multiprocessor Systems", Proc. 1st Annual Int. Symp. *Computer Architecture*, pp. 21-28, December 1973.

[39] R. J. McMillan, "A Survey of Interconnection Networks", Proc. of *GLOBECOM '84*, Atlanta, GA, pp. 105-113, December 1984.

[40] J. Y. Hui, *Switching and Traffic Theory for Integrated Broadband Networks*, Kluwer Academic Publishers, ISBN 079239061X, pp. 113-174, 1991.

[41] M. Piekarski, "Reinventing the Switch Fabric Architecture", *Communication Syst. Design*, http://www.csdmag.com/story/OEG20010521S0113, vol. 7, no. 1, June 2001.

[42] J. H. Patel, "Performance of Processor-Memory Interconnections for Multiprocessors", *IEEE Trans. Computers*, vol. C-30, no. 10, pp. 781-781, October 1981.

[43] R. Y. Awdeh and H. T. Mouftah, "Approach for Comparing Delta Based Networks with the Crossbar Switch", *Electronics Lett.*, vol. 30, no. 3, pp. 201-202, February 1994.

[44] J. Walrand and P. Varaiya, *High-Performance Communication Networks*, Morgan Kaufmann Publishers, ISBN 1558603417, pp. 450-478, 1996.

[45] N. Giroux and S. Ganti, *Quality of Service in ATM Networks*, Prentice Hall, ISBN 0130953873, 1999.

[46] J. Singh, *Semiconductor Devices: Basic Principles*, John Wiley & Sons, ISBN 047136245X, http://www.eecs.umich.edu/~singh/bk7intro.pdf, Introductory Chapter, 2001.

[47] D. Halliday, R. Resnick and J. Walker, *Fundamentals of Physics Extended*, John Wiley & Sons, ISBN 0471105597, pp. 662, 1997.

[48] S. M. Sze, *Semiconductor Devices: Physics and Technology*, John Wiley & Sons, ISBN 0471837040, pp. 9-15, 1985.

[49] W. Shockley and J. Bardeen, "Energy Bands and Mobilities in Monatomic Semiconductors", *Phys. Rev.*, vol. 77, pp. 407-408, February 1950.

[50] J. Bardeen and W. Shockley, "Deformation Potentials and Mobilities in Non-Polar Crystals", *Phys. Rev.*, vol. 80, pp. 72-80, October 1950.

[51] K. Gillessen and E. Schairer, *Light Emitting Diodes: An Introduction*, Prentice-Hall, ISBN 0135365333, 1987.

[52] Eds. S. Nakamura and S. F. Chichibu, *Introduction to Nitride Semiconductor Blue Lasers and Light Emitting Diodes*, Taylor & Francis, ISBN 0748408363, 2000.

[53] Ed. S. M. Sze, *Modern Semiconductor Device Physics*, John Wiley & Sons, ISBN 0471152374, 1998.

[54] H. J. Round, "A Note on Carborundum", *Electrical World*, vol. 49, pp. 309, 1907.

[55] S. D. Smith, *Optoelectronic Devices*, Prentice-Hall, ISBN 0131437690, 1995.

[56] S. A. Campbell, *The Science and Engineering of Microelectronic Fabrication*, Oxford University Press, ISBN 0195105087, 1996.

[57] A. Yariv, *Quantum Electronics*, Third Edition, John Wiley & Sons, ISBN 0471617717, 1987.

[58] H. Kroemer, "A Proposed Class of Heterojunction Injection Lasers", *Proc. IEEE*, vol. 51, pp. 1782-1783, 1963.

[59] I. Hayashi, M. B. Panish, P. W. Foy and S. Sumski, "Junction Lasers Which Operate Continuously at Room Temperature", *Appl. Phys. Lett.*, vol. 17, no. 3, pp. 109-111, 1970.

[60] Z. I. Alferov, V. M. Andreev, D. Z. Garbuzov, Y. V. Zhilyaev, E. P. Morozov, E. I. Portnoi and V. G. Trofim, "Effect of the Heterostructure Parameters on the Laser Threshold Current and the Realization of Continuous Generation at Room Temperature", *Fiz. Tekh. Poluprovodn.*, vol. 4, pp. 1826, 1970 and *Sov. Phys. Semicond.*, vol. 4, pp. 1573, 1971.

[61] J. T. Verdeyen, *Laser Electronics*, Third Edition, Prentice Hall, ISBN 013706666X, 1995.

[62] O. Svelto, *Principles of Lasers*, Third Edition, Plenum Press, ISBN 0306429675, 1989.

[63] A. E. Siegman, *Lasers*, University Science Books, ISBN 0935702113, 1986.

[64] H. Soda, K. Iga, C. Kitahara and Y. Suematsu, "GaInAsP/InP Surface Emitting Injection Lasers", *Jpn. J. Appl. Phys.*, vol. 18, pp. 2329-2330, 1979.

[65] J. L. Jewell, A. Scherer, S. L. McCall, Y. H. Lee, S. Walker, J. P. Harbison and L. T. Florez, "Low-Threshold Electrically Pumped Vertical-Cavity Surface-Emitting Microlasers", *Electronics Lett.*, vol. 25, pp. 1123-1124, 1989.

[66] Ed. G. P. Agrawal, *Semiconductor Lasers: Past, Present and Future*, AIP Press, ISBN 156396211X, 1995.

[67] L. M. F. Chirovsky, W. S. Hobson, J. Lopata, S. P. Hui, G. Giaretta, A. V. Krishnamoorthy, K. W Goossen, G. I. Zydzik and L. A. D'Asaro, "Novel Ion-Implanted VCSEL Structures", Technical Digest of *Spatial Light Modulators and Integrated Optoelectronic Arrays*, OSA, Aspen, CO, pp. 38-39, April 1999.

[68] I. Aeby, "Oxide VCSELs Rise to the Challenge", *Compound Semiconductor Mag.*, http://www.compoundsemiconductor.net/7-5Final/CSJunEmcore.htm, vol. 7, no 5, June 2001.

[69] M. Fuller, "New Research Bolsters Development of 1,300nm VCSELs", *Lightwave*, July 2001.

[70] C. R. Stanley, M. McElhinney, F. Pottier, Y. P. Song, C. D. W. Wilkinson and D. J. Goodwill, "The MBE Growth and MCP/RIE Processing of InGaAs-GaAs MQW Structures for 1047-1064nm S-SEED Arrays", IEEE/LEOS Topical Meeting on *Optoelectronic Material Growth and Processing*, Lake Tahoe, NV, July 11-13, 1994.

[71] W. Franz, *Z. Naturforsch.*, vol. A13, pp. 484, 1958.

[72] L. V. Keldysh, "The Effect of a Strong Electric Field on the Optical Properties of Insulating Crystals", *Zh. Eksp. Teor. Fiz.*, vol. 34, pp. 1138-1141, 1958 and *Sov. Phys.*, vol. 7, pp. 788-790, 1958.

[73] D. A. B. Miller, D. S. Chemla, T. C. Damen, A. C. Gossard, W. Wiegmann, T. H. Wood and C. A. Burrus, "Electric Field Dependence of Optical Absorption Near the Bandgap of Quantum Well Structures", *Phys. Rev. B*, vol. 32, pp. 1043-1060, 1985.

[74] J. D. Dow and D. Redfield, "Electroabsorption in Semiconductors: The Excitonic Absorption Edge", *Phys. Rev. B*, vol. 1, pp. 3358-3370, 1970.

[75] D. A. B. Miller, D. S. Chemla and S. Schmitt-Rink, "Relation Between Electroabsorption in Bulk Semiconductors and in Quantum Wells: The Quantum-Confined Franz-Keldysh Effect", *Phys. Rev. B*, vol. 33, pp. 6976-6982, 1986.

[76] A. L. Lentine, F. B. McCormick, R. A. Novotny, L. M. F. Chirovsky, L. A. D'Asaro, R. F. Kopf, J. M. Kuo and G. D. Boyd, "A Two kbit Array of Symmetric Self-Electro-Optic Effect Devices", *IEEE Photon. Technol. Lett.*, vol. 2, pp. 51, 1990.

[77] L. Geppert, "Opto-Chips Shatter Records for Bandwidth and Low Voltage", *IEEE Spectrum*, vol. 37, no. 6, pp28-29, June 2000.

[78] G. E. Stillman and C. M. Wolfe, "Avalanche Photodiodes", *Semiconductors and Semimetals*, vol. 12, Infrared Detectors II, Academic Press, pp. 291-393, 1977.

[79] J. B. Johnson, "Thermal Agitation of Electricity in Conductors", *Phys. Rev.*, vol. 32, no. 1, pp. 97-109, July 1928.

[80] H. Nyquist, "Thermal Agitation of Electrical Charge in Conductors", *Phys. Rev.*, vol. 32, no. 1, pp. 110-113, July 1928.

[81] S. O. Rice, "Mathematical Analysis of Random Noise", *Bell Syst. Tech. J.*, vol. 23, pp. 282-332, July 1944.

[82] S. O. Rice, "Mathematical Analysis of Random Noise", *Bell Syst. Tech. J.*, vol. 24, pp. 46-156, 1945.

[83] UDT Sensors Inc., *Photodiode Characteristics*, http://www.udt.com/pdf/pd_char.pdf, June 2000.

[84] J. Graeme, *Photodiode Amplifiers*, McGraw-Hill Publishing Company, ISBN 007024247X, February 1996.

[85] S. B. Alexander, *Optical Communication Receiver Design*, SPIE and IEE Publishing, ISBN 0852969007, 1997.

[86] J. Graeme, "Divide and Conquer Noise in Photodiode Amplifiers", *Electronic Design*, pp. 10, June 1994.

[87] A. W. Lohmann and D. P. Paris, "Binary Fraunhofer Holograms, Generated by Computer", *Appl. Opt.*, vol. 6, no. 10, pp. 1739-1748, October 1967.

[88] L. B. Lesem, P. M. Hirsch and J. A. Jordon Jr., "The Kinoform: a New Wavefront Reconstruction Device," *IBM J. Res. Develop.*, vol. 13, pp. 150-155, March 1969.

[89] P. Blair, *Diffractive Optical Elements: Design and Fabrication Issues*, Ph.D. Thesis, Heriot-Watt University, September 1995.

[90] J. W. Goodman, *Introduction to Fourier Optics*, Second Edition, McGraw-Hill, ISBN 0071142576, 1996.

[91] J. L. Sanford, P. F. Greier, K. H. Yang, M. Lu, R. S. Olyha Jr., C. Narayan, J. A. Hoffnagle, P. M. Alt and R. L. Melcher, "A One-Megapixel Reflective Spatial Light Modulator System for Holographic Storage", *IBM J. of Res. and Develop.*, High-Resolution Displays, vol. 42, no. 3/4, pp. 411-427, 1998.

[92] Eds. R. T. Chen and P. S. Guilfoyle, *Optoelectronic Interconnects and Packaging*, SPIE Publishing, ISBN 0819420174, vol. CR62, 1996.

[93] N. Suyal, F. A. P. Tooley, A. Fritze, J. Gourlay, J. A. B. Dines, A. McCarthy, A. C. Walker and F. Bresson, "Multimode Planar Lightwave Circuits Using Direct Write of Polymers", OSA Technical Digest of *Optics in Computing 2000*, Quebec City, Canada, pp. 54-59, June 2000.

[94] T. Soga, S. Hattori, S. Sakai, M. Takeyasu and M. Umeno, "Characterization of Epitaxially Grown GaAs on Si Substrates with III-V Compounds Intermediate Layers by Metalorganic Chemical Vapor Deposition", *J. Appl. Phys.*, vol. 57, no. 10, pp. 4578-4582, May 1985.

[95] R. People and J. C. Bean, "Calculation of Critical Layer Thickness Versus Lattice Mismatch for $Ge_xSi_{1-x}Si$ Strained Layer Heterostructures", *Appl. Phys. Lett.*, vol. 47, no. 3, pp. 322-324, August 1985.

[96] W. L. Ng, M. A. Lourenccedilo, R. M. Gwilliam, S. Ledain, G. Shao and K. P. Homewood, "An Efficient Room-Temperature Silicon-Based Light Emitting Diode", *Nature*, vol. 410, pp. 192-194, March 2001.

[97] R. Droopad and J. Ramdani, "Magic Layer Solves Semiconductor Snag", *IoP PhysicsWeb*, http://physicsweb.org/article/news/5/9/3, September 2001.

[98] J. Hecht, "Chip Maker Celebrates Marriage Made in Heaven", *New Scientist*, vol. 171, no. 2308, pp. 20, September 2001.

[99] J. A. B. Dines, J. F. Snowdon, M. P. Y. Desmulliez, D. B. Barsky, A. V. Shafarenko and C. R. Jesshope, "Optical Interconnectivity in a Scalable Data-Parallel System", *J. of Parallel and Distributed Computing*, vol. 41, pp. 120-130, 1997.

[100] G. A. Russell, J. F. Snowdon, T. Lim, J. Casswell, P. Dew and I. Gourlay, "The Analysis of Multiple Buses in a Highly Connected Optical Interconnect", Technical Digest of *Quantum Electronics and Photonics 15*, IoP Publishing, Glasgow, pp. 75, September 2001.

[101] A. C. Walker, M. P. Y. Desmulliez, M. G. Forbes, S. J. Fancey, G. S. Buller, M. R. Taghizadeh, J. A. B. Dines, C. R. Stanley, G. Pennelli, P. Horan, D. Byrne, J. Hegarty, S. Eitel, K.-H. Gulden, A. Gauthier, P. Benabes, J. L. Gutzwiller and M. Goetz, "Design and Construction of an Optoelectronic Crossbar Containing a Terabit Per Second Free-Space Optical Interconnect", *IEEE J. of Selected Topics in Quantum Electronics*, vol. 5, no. 2, pp. 236-249, March/April 1999.

[102] P. Riley, *TeraConnect Demonstrates World's Densest Optical Interconnect*, http://www.teraconnect.com/pr041601.html, October 2001.

[103] M. J. Flynn, "Very-High Speed Computing Systems", *Proc. of the IEEE*, vol. 54, no. 12, pp. 1901-1909, December 1966.

[104] G. Amdahl, "Validity of the Single Processor Approach to Achieving Large Scale Computing Capabilities", *AFIPS Conf. Proc.*, vol. 30, pp. 483-485, April 1967.

[105] J. L. Hennessey and D. A. Patterson, *Computer Architecture a Quantitative Approach*, Morgan Kaufmann, 1996.

[106] P. Walker, "The Transputer: A Building Block for Parallel Processing", *Byte*, pp. 219, May 1985.

[107] *17th Top 500 List*, http://www.top500.org, June 2001.

[108] W. D. Hillis, *The Connection Machine*, MIT Press, 1985.

[109] IBM, *ASCI White*, http://www.rs6000.ibm.com/hardware/largescale/supercomputers/asciwhite/, October 2001.

[110] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Second Edition, Prentice Hall, ISBN 0780334949, 1999.

[111] R. L. Harvey, *Neural Network Principles*, Prentice Hall, ISBN 0130633305, 1994.

[112] J. A. Anderson, *An Introduction to Neural Nets*, IT Press, ISBN 0262011441, 1995.

[113] R. Beale and T. Jackson, *Neural Computing an Introduction*, IoP Publishing, ISBN 0852742622, 1990.

[114] F. Rosenblatt, *Principles of Neurodynamics*, Spartan Books, pp. 23-26, 1959.

[115] J. Hertz, A. Krough and R. G. Palmer, *Introduction to the Theory of Neural Computation*, Addison-Wesley, ISBN 0201515601, 1991.

[116] M. R. W. Manning and M. Gell, "A Neural Network Service Assignment Model", *BT Tech. J.*, vol. 12, no. 2, pp. 50-566, April 1994.

[117] W. J. Wolfe, J. M. MacMillan, G. Brady, R. Mathews, J. A. Rothman, D. Mathis, M. D. Orosz, C. Anderson and G. Alaghband, "Inhibitory Grids and the Assignment Problem", *IEEE Trans. on Neural Networks*, vol. 4, no. 2, March 1993.

[118] T. X. Brown, *Neural Networks Design for Switching Network Control*, Ph.D. Thesis, Caltech, Chapter 3, 1991.

[119] M. R. Garey and D. S. Johnson, *Computers and Intractability*, W. H. Freeman, 1979.

[120] J. Munkres, "Algorithms for Assignment and Transportation Problems", *J. Soc. Ind. Appl. Math.*, vol. 5, pp. 32-38, 1957.

[121] J. J. Hopfield and D. W. Tank, "'Neural' Computation of Decisions in Optimisation Problems", *Biological Cybernetics*, vol. 52, pp. 141-152, 1985.

[122] R. D. Brandt, Y. Wang, A. J. Laub and S. K. Mitra, "Alternative Networks for Solving the Travelling Salesman Problem", IEEE Int. Conf. on *Neural Networks*, San-Diego, USA, vol. 2, pp. 333-340, February 1998.

[123] P. W. Protzel, D. L. Palumbo and M. K. Arras, "Performance and Fault-Tolerance of Neural Networks for Optimisation", *IEEE Trans. on Neural Networks*, vol. 4, no. 4, pp. 600-614, July 1993.

[124] C. Bousoño-Calzón and M. R. W. Manning, "The Hopfield Neural Network Applied to the Quadratic Assignment Problem", BT Labs, unpublished, 1994.

[125] J. Ghosh, A. Hukkoo and A. Varma, "Neural Networks for Fast Arbitration and Switching Noise Reduction in Large Crossbars", *IEEE Trans. on Circuits and Syst.*, vol. 38, no. 8, pp. 895-904, August 1991.

[126] T. X. Brown, "Neural Networks for Switching", *IEEE Communications Mag.*, vol. 5, no. 11, pp. 72-81, November 1989.

[127] A. Marrakchi and T. Troudet, "A Neural Net Arbitrator for Large Crossbar Packet Switches", *IEEE Trans. on Circuits and Syst.*, vol. 36, no. 7, pp. 1039-1041, July 1989.

[128] S. B. Aiyer, M. Niranjan and F. Fallside, "A Theoretical Investigation into the Performance of the Hopfield Model", *IEEE Trans. on Neural Networks*, vol. 1, no. 2, pp. 204-215, June 1990.

[129] R. P. Webb, A. J. Waddie, K. J. Symington, M. R. Taghizadeh and J. F. Snowdon, "A Neural Network Scheduler for Packet Switches", OSA Technical Digest of *Optics in Computing 1999*, Aspen, CO, USA, pp. 193-195, April 1999.

[130] J. J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities", *Proc. Natl. Acad. Sci.*, vol. 79, pp. 2554-2558, April 1982.

[131] J. J. Hopfield, "Neurons with Graded Response Have Collective Computational Properties Like Those of Two-State Neurons", *Proc. Natl. Acad. Sci.*, vol. 81, pp. 3088-3092, May 1984.

[132] S. U. Hegde, J. L. Sweet and W. B. Levy, "Determination of Parameters in a Hopfield Tank Computational Network", IEEE Int. Conf. on *Neural Networks*, San-Diego, vol. 2, pp. 291-298, February 1998.

[133] M. Ali and H. Nguyen, "A Neural Network Implementation of an Input Access Scheme in a Highspeed Packet Switch," Proc. of *GLOBECOM*, pp.1192-1196, 1989.

[134] K. Ballüder and M. R. Taghizadeh, "Optimised Phase Quantisation for Diffractive Elements Using a Bias Phase", *Opt. Lett.*, vol. 24, no. 23, pp. 1576-1578, 1999.

[135] A. Bateman and I. Paterson-Stephens, *The DSP Handbook*, Prentice Hall, ISBN 0201398516, 2002.

[136] N. McKeown, M. Izzard, A. Mekkittikul, W. Ellersick and M. Horowitz, "The Tiny Terra: A Packet Switch Core", *IEEE Micro*, vol. 17, pp. 26-33, 1997.

[137] R. P. Webb, A. J. Waddie, K. J. Symington, M. R. Taghizadeh and J. F. Snowdon, "An Optoelectronic Neural Network Scheduler for Packet Switches", *Appl. Opt.*, vol. 39, no. 5, pp. 788-795, February 2000.

[138] A. Ludolph, *Examinations on a Neural Switch Controller*, M.Sc. Dissertation, University of Reading, September 1997.

[139] K. J. Symington, *Optoelectronic Neural Networks for Switching*, M.Sc. Dissertation, Heriot-Watt University, September 1998.

[140] S. E. Swirhun, R. P. Bryan, W. S. Fu, W. E. Quinn, J. L. Jewell and G. R. Olbright, "Commercial Manufacturing of Vertical-Cavity Surface-Emitting Laser Arrays", *SPIE Proc.*, ISBN 0819414425, vol. 2147, pp.74-84, January 1994.

[141] M. A. Holler, "VLSI Implementations of Learning and Memory Systems: A Review", *Advances in Neural Information Processing Syst. 3*, San Mateo, CA, Morgan Kaufmann, pp. 993-1000, 1991.

[142] *NLX420 Data Sheet*, Neurologix Inc., 800 Charcot Av., Suite 112, San Jose, CA, June 1992.

[143] M. Glesner and W. Pöchmüller, *Neurocomputers: An Overview of Neural Networks in VLSI*, Chapman & Hall, London, 1994.

[144] D. Hammerstrom, "A VLSI Architecture for High-Performance, Low-Cost, On-chip Learning", Proc. *Int. Joint Conf. on Neural Networks IJCNN*, San Diego, CA, vol. 2, pp. 537-544, 1990.

[145] Adaptive Solutions Inc., 1400 N. W. Compton Dr., Suite 340, Beaverton, OR 97006.

[146] *MT19003 Data Sheet*, MCE, Alexander Way, Tewkesbury, Gloucestershire GL20 GTB, 1994.

[147] *MD1220 Data Sheet*, Micro Devices, 30 Skyline Dr., Lake Mary, FL 32746, March 1990.

[148] N. Mauduit, M. Duranton and J. Gobert, "Lneuro 1.0: A Piece of Hardware LEGO for Building Neural Network Systems", *IEEE Trans. on Neural Networks*, vol. 3, pp. 414-422, May 1992.

[149] J. Beichter, N. Bruels, E. Meister, U. Ramacher and H. Klar, "Design of General-purpose Neural Signal Processor", 2nd Int. Conf. on *Microelectronics for Neural Networks*, Münich, Germany, pp. 311-315, 1991.

[150] U. Ramacher, "SYNAPSE - A Neurocomputer that Synthesises Neural Algorithms on a Parallel Systolic Engine", *J. of Parallel and Distributed Computing*, vol. 14, pp. 306-318, 1992.

[151] *80170NX ETANN Data Sheet*, Intel Corp., Santa Clara, CA, 1991.

[152] B. E. Boser, E. Sackinger, J. Bromley, Y. LeCun and L.D. Jackel, "Hardware Requirements for Neural Network Pattern Classifiers", *IEEE Micro*, pp. 32-40, February 1992.

[153] J. Alspector, T. Jayakumar and S. Luna, "Experimental Evaluation of Learning in a Neural Microsystem", *Advances in Neural Information Processing Syst. 4*, Morgan Kaufmann, San Mateo, CA, pp. 871-878, 1992.

[154] P. Masa, K. Hoen, H. Wallinga, "A 20 Input, 20 Nanosecond Pattern Classifier", Proc. *Int. Joint Conf. on Neural Networks IJCNN '91*, Orlando, FL, USA, July 1994.

[155] H. Eguci, T. Furata, H. Horiguchi, S. Oteki and T. Kitaguchi, "Neural Network LSI Chip with On-chip Learning", Proc. of *Int. Joint Conf. on Neural Networks IJCNN '90*, San Diego, CA, June 1990.

[156] *TMS320C5x DSP Starter Kit*, SPRU101A, Texas Instruments Inc., http://www.ti.com, June 1996.

[157] *AD7822/AD7825/AD7829 Data Sheet*, Analog Devices, Rev. 0, http://www.analog.com, 1997.

[158] *MM54HC02/MM74HC02 Quad 2-Input NOR Gate*, National Semiconductor, TL/F/5294, January 1998.

[159] *SN54HC573A, SN74HC573A Octal Transparent D-Type Latches With 3-State Outputs*, Texas Instruments, SCLS147B, http://www.ti.com, December 1982.

[160] Heriot-Watt University, Supelec, University of Glasgow, Trinity College and CSEM, *Supply of 8×8 VCSEL Array as Input Device for Demonstrator*, Deliverable 4, Project No. 22668 SPOEC, April 1998.

[161] Ed. D. M. Rowe, *CRC Handbook of Thermoelectrics*, CRC Press, 1995.

[162] *TEC Design Guide*, http://www.marlow.com/design.htm, June 2001.

[163] *TL074CD Low Noise JFET Input Operational Amplifiers*, Texas Instruments, http://www.ti.com, August 1996.

[164] R. Stone, J. Kim and P. Guilfoyle, "High Performance Shock Hardened Optoelectronic Communications Module", OSA Technical Digest of *Optics in Computing 2001*, Lake Tahoe, NV, pp. 105-107, January 2001.

[165] C. Maxfield, "Logic That Mutates While-U-Wait", *EDN*, vol. 41, no. 23, pp. 137-140 & 142, 7th November 1996.

[166] S. Hauck, *Multi-FPGA Systems*, PhD Thesis, University of Washington, 1995.

[167] M. Motomura, Y. Aimoto, A. Shibayama, Y. Yabe and M. Yamashina, "An Embedded DRAM-FPGA Chip with Instantaneous Logic Reconfiguration", Proc. of IEEE Symp. on *FPGAs for Custom Computing Machines*, ISBN 0818689005, IEEE Computer Society, pp. 264-266, 1998.

[168] S. D. Brown, R. J. Francis, J. Rose and Z. G. Vranesic, *Field-Programmable Gate Arrays*, Kluwer Academic Publishers, USA, 1993.

[169] J. V. Oldfield and R. C. Dorf, *Field Programmable Arrays: Reconfigurable Logic for Rapid Prototyping and Implementation of Digital Systems*, John Wiley and Sons, 1995.

[170] J. Kouloheris and A. El Gamma, "FPGA Area vs. Cell Granularity - Lookup Tables and PLA Cells", *First ACM Workshop on Field Programmable Gate Arrays*, FPGA '92, Berkeley, CA, pp. 9-14, February 1992.

[171] J. S. Rose, R. J. Francis, D. Lewis and P. Chow, "Architecture of Programmable Gate Arrays: The Effect of Logic Block Functionality of Area Efficiency", *IEEE J. of Solid State Circuits*, vol. 25, no. 5, pp. 1217-1225, October 1990.

[172] *XC6200 Field Programmable Gate Array*, Xilinx Corp., http://www.xilinx.com, April 1997.

[173] *QL4090 9,000 Useable PLD Gate QuickRAM ESP*, QuickLogic, http://www.quicklogic.com, August 1998.

[174] *DY6000 Family FAST Field Programmable Gate Array*, Dynachip, http://www.dyna.com, rev. 1.1, April 1999.

[175] *ORCA Series 3 Field Programmable Gate Array*, Lucent Technologies, http://www.lucent.com, rev. 1, September 1998.

[176] *Apex 20K Programmable Logic Device Family*, Altera, http://www.altera.com, ver. 2, May 1999.

[177] J. M. P. Cardoso and M. P. Véstias, "Architectures and Compilers to Support Reconfigurable Computing", *ACM Crossroads Student Mag.*, http://www.acm.org/crossroads/xrds5-3/rcconcept.html, April 1999.

[178] I. Page, "Reconfigurable Processor Architectures", *Microprocessors & Microsystems*, vol. 20, no. 3, pp. 185-196, May 1996.

[179] *Field Programmable Gate Arrays - An Enabling Technology*, Virtual Computer Corporation, http://www.vcc.com/fpga.html.

[180] J. E. Vuillemin, P. Bertin, D. Roncin, M. Shand, H. H. Touati, P. Boucard, "Programmable Active Memories: Reconfigurable Systems Come of Age", *IEEE Trans. on Very Large Scale Integration (VLSI) Syst.*, vol. 4, no. 1, pp. 56-69, March 1996.

[181] D. Conner, "Reconfigurable Logic: Hardware Speed with Software Flexibility", *Australian Electronics Engineering*, vol. 29, no. 8, pp. 40, 42 & 44, August 1996.

[182] *Triscend E5 Configurable Processor Family*, Triscend, http://www.triscend.com, November 1998.

[183] Starbridge Systems Inc., http://www.starbridgesystems.com/Pages/about.html, June 1999.

[184] M. Butts, "Future Directions of Dynamically Reprogrammable Systems", Proc. of IEEE *Custom Integrated Circuits Conf. 1995*, pp. 487-494, 1995.

[185] S. Kumar, L. Pires, D. Pandalai, M. Vojta, J. Golusky, S. Wadi and H. Spaanenburg, "Benchmarking Technology for Configurable Computing System", Proc. of IEEE Symp. on *FPGAs for Custom Computing Machines*, IEEE Computer Society, pp. 273-274, Los Alamitos, CA, 1998.

[186] B. Radunovic and V. Milutinovic, "A Survey of Reconfigurable Computing Architectures", Proc. of *Field Programmable Logic 1998*, ISBN 3540649484, Springer-Verlag, pp. 376-385, 1998.

[187] B. Fawcett, "FPGAs as Reconfigurable Processing Elements", *IEEE Circuits and Devices Mag.*, vol. 12, no. 2, pp. 8-10, March 1996.

[188] K. Nagami, K. Oguri, T. Shiozawa, H. Ito and R. Konishi, "Plastic Cell Architecture: Towards Reconfigurable Computing for General Purpose", Proc. of IEEE Symp. on *FPGAs for Custom Computing Machines*, ISBN 0818689005, Los Alamitos, CA, IEEE Computer Society, pp. 68-77, 1998.

[189] S. C. Woo, *The Performance Advantages of Integrating Block Data Transfer in Cache-Coherent Multiprocessors*, Ph.D. Thesis, Stanford University, 1996.

[190] I. Pearson, http://www.bt.com/sphere/insights/pearson.

[191] N. Donofrio, "Technology Innovation for a New Era", IEE *Computing and Control Engineering J.*, vol. 12, no. 3, pp. 115-124, June 2001.

[192] C. Lazou, "Beyond the Limits", BCS *Computer Bulletin*, November 2001.

[193] I. Gourlay, *Quantum Computing*, Ph.D. Thesis, Heriot-Watt University, December 2000.