Heriot-Watt University

St. Andrews University

Optoelectronic Neural Networks for Switching

Abstract

Current software systems suffer from an exponential increase in computational complexity when solving a quadratic assignment problem. Such problems exist in today's telecommunication systems as a network tries to rout calls optimally through its switches to minimise blocking. This project considers the problem and proceeds to propose a solution using the inherent parallelism of a neural network to reduce computation times. In conclusion, a hardware implementation is examined which uses free space optical interconnects to reduce circuit complexity and its performance is closely scrutinised.

MSc Optoelectronics and Laser Devices Project Dissertation

Students Name:	Keith J. Symington
Registration number:	9711710098481
Company/Address:	BT Labs, Martlesham Heath, Ipswich, IP5 7RE.
Company Supervisor:	R. P. Webb
Internal Supervisor:	M. R. Taghizadeh
Date:	8th September 1998

1 Contents

2 Introduction

2.1 The Assignment Problem

As the complexity of modern communications and computational systems increases so does the need to develop new techniques which deal with common assignment problems ([6] and [7]) 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 throughput and minimise blocking.

This report examines specifically the assignment problem in a crossbar switch for packet routing [11]. These switches are present in many telecommunication systems and computer networks, one good example being ATM (Asynchronous Transfer Mode) networks.

2.2 Neural Network Implementation

The problem of packet routing in crossbar switches is known to be analogous to the travelling salesman problem (TSP). The TSP problem is a renowned NP complete problem [22] which means that although it can be solved by linear programming techniques, such as the Murnkes algorithm [23], 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 [8], [9]. The advantage of a neural net lies in the speed obtained through its inherent parallel operation, especially when dealing with large problems. Such an implementation will easily outperform any other method at higher orders of network size ([1], [4], [5], [6], [10], [14] and [16]) providing a very good, but not optimal, solution. It has been shown [6] 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 the optimal solution.

The problem which remains with any neural network solution is its adaptation to act as a controller for the crossbar switch.

2.3 Implementation Overview

Figure 1 shows a high level overview of the system. Each neuron in the Hopfield network controls a single crosspoint switch. Collectively, the neural network examines all incoming packet buffers and, based on the packets' requested output connections, chooses an optimal combination of packets to throughput. The neural network considers any output to be optimal if it maximises the crossbar switch's usage. All appropriate connections are then made by setting their crosspoints on the crossbar switch. This allows the selected packets to be routed through the switch.

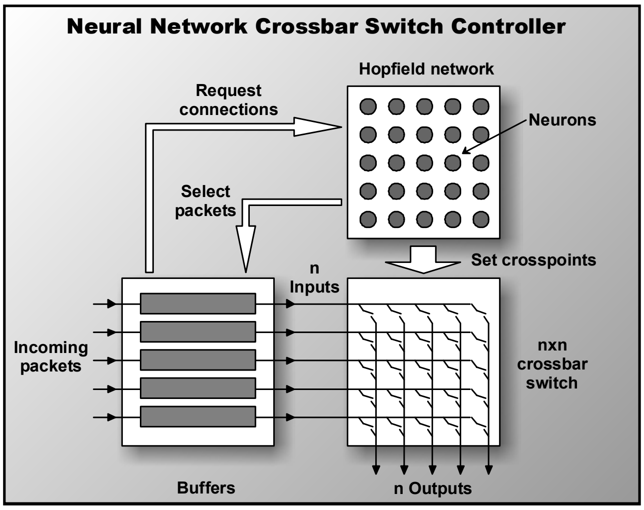


Figure 1

The proposed system uses a Hopfield neural network to examine all incoming packet buffers and rout packets through the crossbar switch in an optimal manner.

Neural networks use simple processing elements where communication between processors is an integral part of their design. This leads to a highly interconnected system and typically a fabrication layout nightmare at higher orders: where neural network control really proves itself.

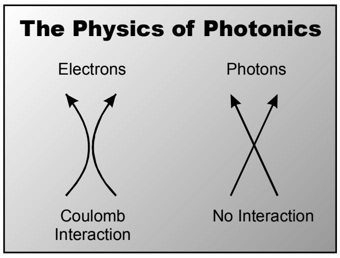


Figure 2

Photons have the advantage of being non-interacting in free space.

Therefore, this project proposes optical interconnection of neurons ([17], [18], [21], [28], [33] and [34]). Light has the property that it is non-interacting in free space and therefore the interconnects can effectively cross each other (figure 2 and [25]). Since the interconnects can then be more direct, not only is the amount of routing reduced but signal skew becomes less of a problem.

2.4 Report Outline

The objective of this report is to present a modified Hopfield neural network as an implementation method for throughput optimisation in crossbar switches.

The report is divided into 2 main chapters. The first chapter is dedicated to theory while the second to procedure and results. This report also includes extensive appendices which will be referred to throughout the text.

3 Theory

3.1 Crossbar Switches and Notation

A crossbar switch can be simply abstracted as a set of N inputs and N outputs where each input can be switched to any output.

An example of this can be seen in figure 3 where, by simply closing the correct crosspoint switch, any input line may be connected to any output line. This system has the limitation that it is mutually exclusive: any input or output lines that are in use cannot be reused. Thus, two incoming requests for the same output line will result in one becoming blocked regardless of the routing algorithm which is used.

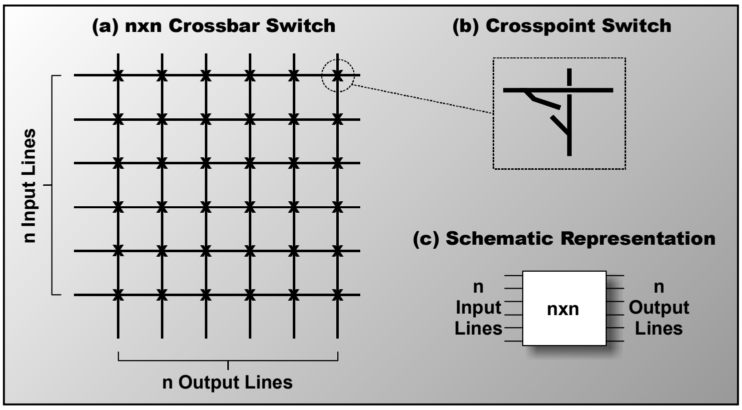


Figure 3

An NxN crossbar switch is shown here at various levels of detail.
(a) Shows an overall connection diagram for a typical crossbar switch.
(b) Details how each of the crosspoint switches work.
(c) Depicts a high level schematic of a crossbar switch.

To clarify the notation used throughout the rest of the report, please examine figure 4. This diagram details how a matrix may be mapped onto the crossbar switch, each crosspoint having a corresponding matrix element. A specific element in any matrix y 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 one of two values: 1 when there is a connection (or connection request) or 0 otherwise. The value and legality of the matrix is dependent on situation. Please examine the matrices shown in equations 1 and 2.

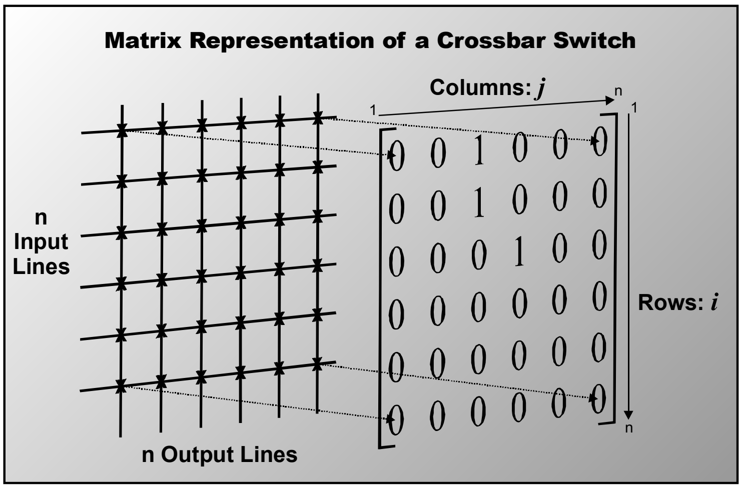
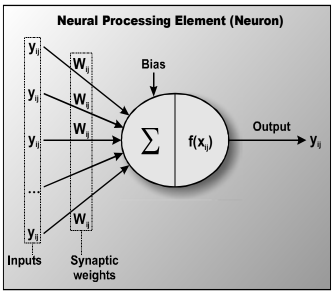
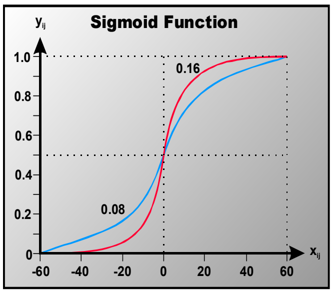


Figure 4

This diagram shows how a matrix can be mapped onto the crossbar switch thus aiding representation.





Equation 1

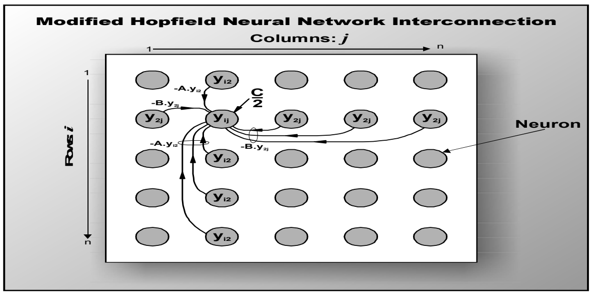
This matrix shows a set of requested connections. Input i=1 has requested a connection with output j=3 and both inputs i=2 and i=3 have requested a connection to output j=4.

Equation 2

This matrix shows a solution or response to the request in equation 1. It is legal because there are no other connections on the input rows and output columns which have been selected.

These matrices represent the crossbar switch in figure 4 but from different points of view. Equation 1 represents a set of desired connections where three input lines have requested connection to two different output lines: one request is obviously going to have to wait. Such a matrix is legal regardless of the combination of zeroes and ones. Equation 2 shows a sample response. One request has been discarded in favour of another since only one input line can be connected to one output line at a time. A response is considered to be legal if there are no other closed switches on the same lines, i.e. all other elements in the same row and column as the active element must be zero.

The real optimisation problem comes in when you start to consider a system which has buffered input (as shown in figure 1). In such systems there can be multiple packets waiting on a single input line for various output lines, as can be seen in equation 3. Requests for multiple connections can be seen in the left matrix and the only optimal solution which maximises throughput on the right. This request matrix proves useful for testing crossbar control systems.



Equation 3

The left matrix shows a request and the right the only optimal response. This matrix is useful for testing a system.

As an enhancement to packet systems, each element could be converted to an integer value representing the number of packets waiting on each connection. This is, however, not within the scope of this report.

Note that although this description limits itself to square switches with the same number of inputs as outputs, it is possible to have different numbers of inputs and outputs. The system built and described in this report has in fact 6 inputs and 8 outputs.

3.2 The Hopfield Neural Network

The key to utilising the parallelism of a neural network is matching the network as closely as possible to the problem. This section explains the theory behind the modified Hopfield neural network used in this project but does not give a generalised description due to space constraints. For more information please refer to references [12], [29], [30], [31], [32] or [35].

3.2.1 The Neuron or Node

A Hopfield neural network consists of a large number of processing elements called neurons (see figure 5 or references [13] and [35]) which are highly interconnected to each other in a specific fashion. Neurons are the basic building blocks of neural networks and are an approximation of the neuron found in nature. A neuron takes inputs from other neurons' outputs y_{ij} (referenced by ij) and multiplies their strengths by a scalar weight W_{ij} known as the synaptic weight.

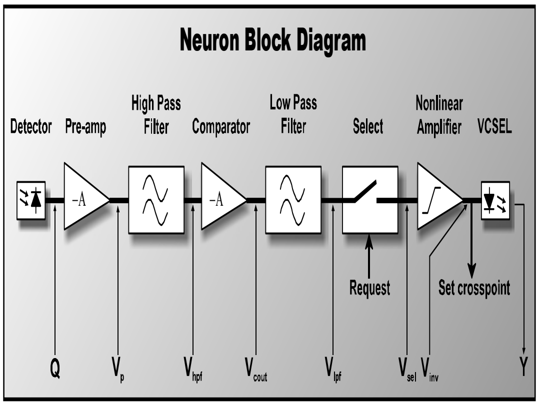
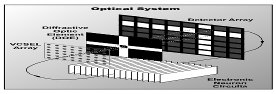


Figure 5

The building block of any neural network: the neuron.



Equation 4

All inputs are summed by the neuron along with a specific bias to find x_{ij}. The neuron's output y_{ij} can then be determined using a monotonic activation function $f(x_{ij})$, as shown in equation 4. Here β is used to control the gain of the sigmoid function, a higher value resulting in a steeper transition (as can be seen in figure 6).

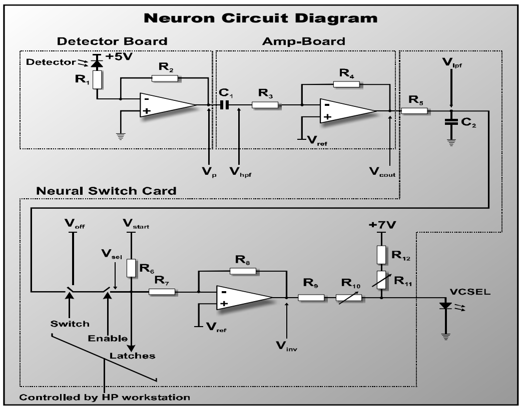


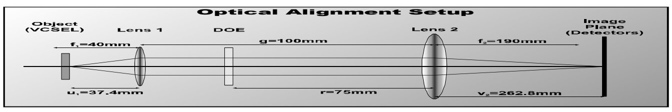
Figure 6

Sigmoid activation function of a neuron as in equation 4.

The exact form of $f(x_{ij})$ is not particularly important and in fact any appropriate non-linear monotonically increasing function could be used. The preferred embodiment is, however, the sigmoid function.

3.2.2 The Updating Rule

Adapting a neural network to any problem requires that an updating rule is defined and thereby the network interconnection structure. The updating rule determines the next value that a neuron will take with respect to time based upon the previous outputs of other neurons, as shown in equation 5:



Equation 5

where:

x_{ij}: is a summation of all inputs to the neuron referenced by ij including the bias.

y_{ij}: is the output of a neuron referenced by ij.

A: Optimisation value weighting the input from any element in the same column.

B: Optimisation value weighting the input from any element in the same row.

C: Optimisation value representing external bias supplied to each neuron.

and x_{ij} is related to y_{ij} using equation 4.

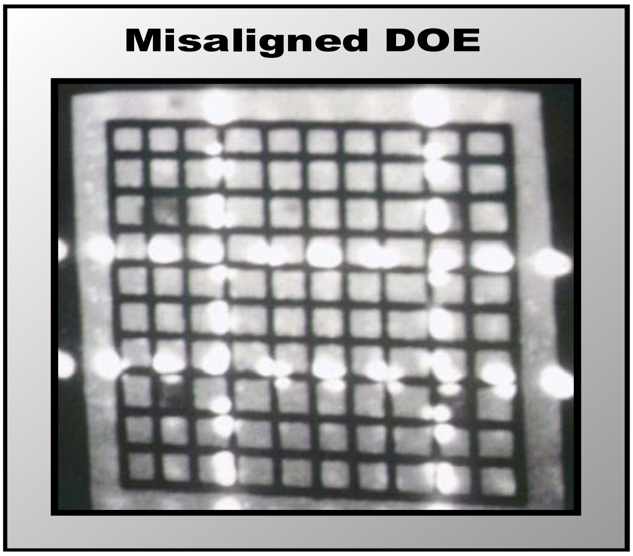
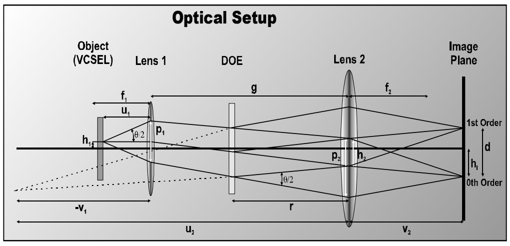


Figure 7

To illustrate this rule further, figure 7 shows an interconnection diagram for the modified system. Here the neuron marked with output y_{ij} has inputs from all the other neurons in the same row $-B.y_{2j}$ and column $-A.y_{i2}$. The important point to note here is that the neural network works in an inhibitory fashion so any active input will inhibit y_{ij}. $C/2$ describes the external bias supplied to each neuron which is not inhibitory.

The idea behind this interconnection strategy is that any active neuron will try and turn all the others off, eventually resulting in only one of the requests remaining active in each row and column. However, to demonstrate its ability to find an optimal solution, the example in figure 7 needs to be extended slightly, as in equation 6. The left matrix here represents a request 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 here since both y_{24} and y_{42} are inhibiting y_{22}, thus resulting in it being switched off before the others and essentially losing. If y_{22} had won in this case then it would have resulted in a poor solution since y_{24} and y_{42} would be off: obviously not maximising potential throughput.



Equation 6

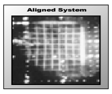
The left matrix is a request and the right its solution.

It has been shown by Hopfield that with symmetric connections and a monotonically increasing activation function $f(x)$, the dynamical system described by the neural network possesses a Lyapunov (energy) function which continually decreases with time. The existence of such a function guarantees that the system converges towards equilibrium which is often referred to as a 'point attractor'.

The 'optimisation parameters' A, B and C [15] have been determined purely by trial and error in previous work [24]. If these parameters are not chosen carefully then equation 5 will converge either very slowly or not at all. A further possibility is that the system might converge to an invalid solution.

3.2.3 Determination of Optimisation Parameters

It is possible to determine the optimisation parameters by a more methodical method than simply trial and error. A solution for equation 5 can be found when the system is under conditions of equilibrium, as shown in equation 7. This results in equation 8.



Equation 7

$$x_{0,ij} = -A\sum_{k \neq j}^{n} f\left(x_{0,ik}\right) - B\sum_{k \neq i}^{n} f\left(x_{0,kj}\right) + \frac{C}{2}$$ **Equation 8**

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 per row and column. This information allows us to establish that, if ij is a is a zero position, the equilibrium condition reads as in equation 9, where x_1 denotes the first equilibrium solution.

$$x_1 = -A - B + \frac{C}{2}$$ **Equation 9**

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 (equation 4). Accordingly, we can rewrite equation 9 as the inequality shown in equation 10: 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.

$$-A - B + \frac{C}{2} << 0$$ **Equation 10**

The next consideration must be the ij positions which are tending towards one. In equilibrium, the condition then becomes that shown in equation 11, where x_2 represents the second equilibrium solution.

$$x_2 = \frac{C}{2}$$ **Equation 11**

Again using equation 4, it can be easily seen that y tends to one as x tends to infinity. This allows us to rewrite equation 11 as the inequality in equation 12 or ‘positive attractor’. This condition will have to be satisfied at n positions in the network.

$$\frac{C}{2} >> 0$$ **Equation 12**

The final equilibrium conditions mean that n neurons in the network have converged to one of the two attractors and n^2-n neurons have converged to the other. Combining equations 10 and 12 gives the overall inequality in equation 13.

$$0 < C < 2\left(A + B\right)$$ **Equation 13**

This equation can be refined since a symmetric matrix is desired (i.e. A=B), as shown in equation 14.

$$0 < C < 4A$$ **Equation 14**

3.2.4 Local Minima

In any system with a continually reducing energy function, there is always a risk that the system will become trapped in a local minima. In this system, a local minima can be represented as a solution which satisfies the switching constraints but is not a global optimal solution. The best way round this problem is to introduce noise into the system by varying β, as shown in figure 6, between 0.08 and 0.16. This alteration in the activation curve's gradient is significant enough to provide successful convergence to a global minimum during network simulation.

Note that this strategy is not used in the actual system since there is enough background noise in any real system to make this variation unnecessary.

3.3 System Design

This implementation of a neural network uses optics to interconnect all the neurons in a configuration as described in section 3.2. This method has the advantage that a large and complicated interconnect pattern can be realised with ease.

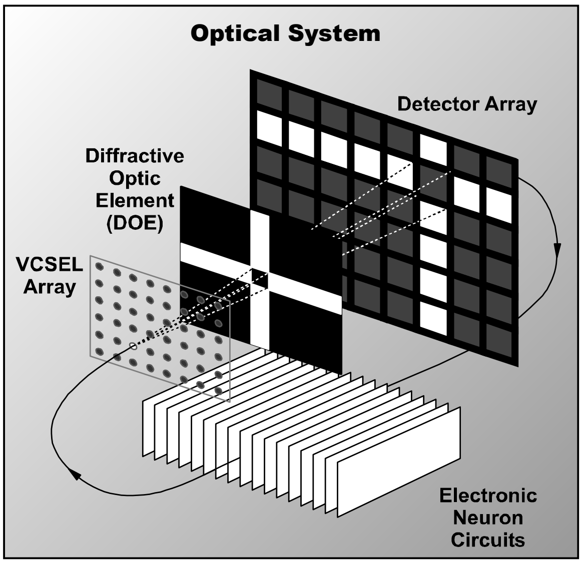


Figure 8

Each neuron has an associated detector and VCSEL which act as input and output respectively. The DOE divides any output light from a neuron's VCSEL to the adjacent neurons' detectors as indicated above.

The optical setup, as illustrated in figure 8, uses a detector as an input to each electrical neuron and a vertical cavity surface emitting laser (VCSEL) as output. As a neuron turns on, so does the appropriate VCSEL. The task of the diffractive optic element (DOE) is to disperse the power from an active VCSEL so that light is directed onto the detectors of neurons in the same row and column [26], [27]. Any light incident on a detector acts in an inhibitory manner causing the associated neuron to turn off: the higher the light intensity, the more likely it is that a neuron will turn off. Note that the VCSEL array is turned through 180° in relation to the detectors. This setup does unfortunately have two major sources of inherent errors.

The first problem is the VCSELs. This system is designed so that the output from each neuron has an equal weight i.e. the output light intensity should be equal for all lasers. This is obviously not the case with a VCSEL array as it is almost impossible to fabricate every VCSEL with the exact same output characteristics. It is, therefore, necessary to calibrate each VCSEL so that the power output for on and off are the same: both of which must lie above threshold.

The second is optical alignment. The system needs to be aligned in such a way that the output from each VCSEL reaches only the correct detector(s). This involves careful alignment of both VCSEL and detector arrays as well as any associated lens system.

Discrete electronics were used to implement each neuron. Figure 9 shows a block diagram representation of the electronic system, whereas figure 10 shows a circuit diagram.

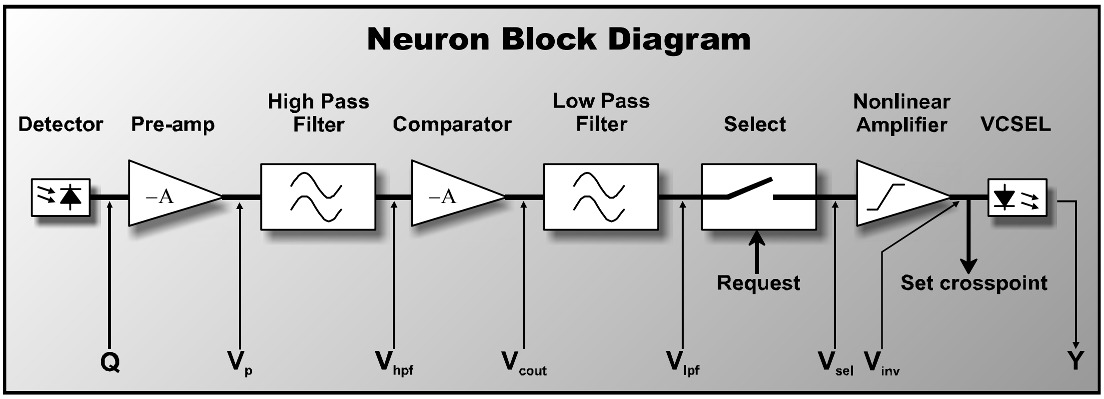
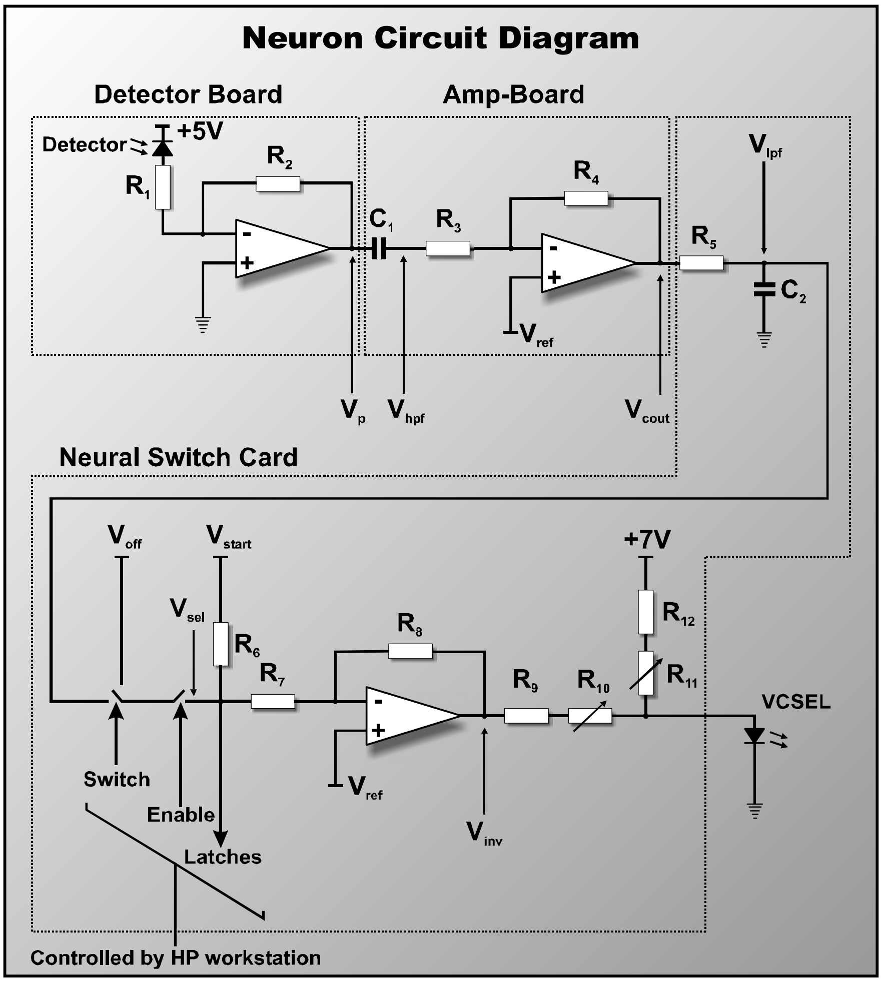


Figure 9

Block diagram schematic of the implementation of each neuron. Various reference points are marked and will be referred to later on in this report.

These electronics were divided up over a series of different circuit board modules, each of which will be described later in this report.

Figure 10



4 Procedure and Results

4.1 Network Simulation

The first aspect of the project examined was simulation of the perfect theoretical case. A pure theoretical model was available as Matlab source code and is included in Appendix A. Theoretical examination was undertaken to determine the significance of each of the optimisation parameters shown in section 3.2 as well as the neuron's activation function. The following points were determined from both papers and examination of the model:

- Noise plays a very significant role in this model. As the noise level increases, the time taken for network stabilisation decreases. However, when the noise value reaches unity the network becomes unstable and does not provide a valid or steady solution.
- Network size plays an important role in convergence to a solution: the larger it is, the longer it takes to converge.
- The value of β should lie within the region 0.08 to 0.16 for optimal performance.
- β is effectively linked to C as in equation 15. $\beta.C \approx 2$ **Equation 15**
- C should remain within the limits 40 to 150 for optimal operation.
- Increasing the value of C encourages the neurons to choose quickly.
- A=B (presuming a symmetric matrix) should be at least ten times greater than C.

The preferred values used during simulation were A=B=1250, C=100 and therefore, from equation 15, β=0.02 (slightly outwith optimum).

Simulation was also performed of a more realistic model based on figure 9 to analyse the system when implemented in the proposed manner. The Matlab code for this can be found in Appendix B.

Both models performed as predicted in the patent application [2] on close examination.

4.2 Optical Alignment

This system relies heavily on the properties of a diffractive optic element (DOE [19]) to split up incoming light and cast it onto the appropriate detectors as shown in figure 8. For optimal results from the DOE, incoming light must be nearly collimated. However, the VCSEL array outputs light with a divergence of approximately 8° thus requiring slight focussing. In addition, a magnification of 6x must be present if 250µm spaced VCSELs are to be focussed onto 1.5mm spaced detectors. It becomes obvious at this point that a lens system is required to perform collimation and subsequent magnification.

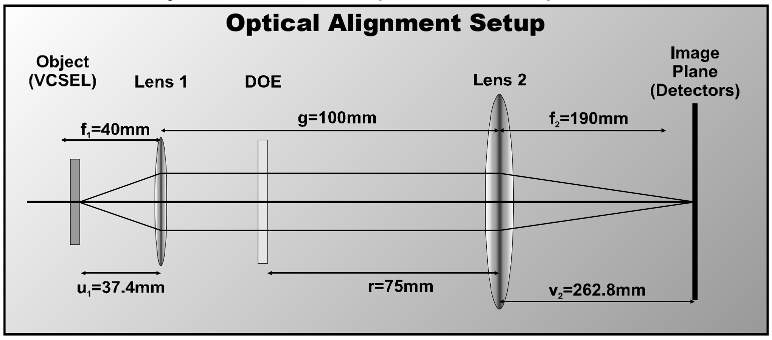


Figure 11

Optical system as proposed by previous calculations in reference [20]. Drawing not to scale.

Figure 11 shows the system setup as previously proposed in reference [20].

A DOE element fitting system specifications was received at the beginning of this project. It was inserted into the system and its output examined by projection onto a grid which was the same size as the detector array. The image projected onto the image plane was not as expected and is shown in figure 12. It can be easily seen here that the projected crosses from the test VCSELs do not fit correctly between the grid lines. Each element in the system was then carefully examined in an effort to find any problematic components and eliminate any errors they are be introducing:

Figure 12

Crosses output from 4 VCSELs should overlap perfectly with spots lying between the grid lines.

- VCSEL: Moving the VCSEL in relation to Lens 1 alters focussing on the image plane. For a sharp image there is only one position for the VCSEL – at Lens 1's focal point.
- Lens 1: Should really only have one position: focussed on the VCSEL array.
- DOE: The position was found to be extremely sensitive to change. Movement away from Lens 1 results in an increase in the number of orders visible between two laser positions: towards and the number of orders decreases.
- Lens 2: Focuses at a specific distance to give the image plane. Movement in relation to Lens 1 allows the size of the image on the output plane to be altered.

The only component sensitive to a change in position was found to be the DOE. The DOE's position was varied between Lens 1 and Lens 2 to try and find a point at which the image was projected correctly but there was none. The DOE was then removed from the system and its characteristics examined more closely. It suddenly became apparent that the DOE's working distance was not the same as that used in [20]. The working distance is the distance at which the DOE correctly projects the desired image and an incorrect value would explain the problems seen in figure 12. The working distance therefore had to be re-measured and turned out to be 187mm rather than one of the two pre-calculated values.

4.3 Revised Lens Model

The calculations made in reference [20] for initial system design were based around a DOE that had a working distance of exactly either 120mm or 230mm. Since the DOE supplied had a different working distance, it was necessary to redesign the optical system. The lens system was remodelled using Matlab V4.2.1c (code in Appendix C) with the previous work as a basis. Figure 13 shows a drawing of the system setup. This section details the formulae used to calculate an optimal system setup, however the origins of each equation are not detailed because of resultant complexity.

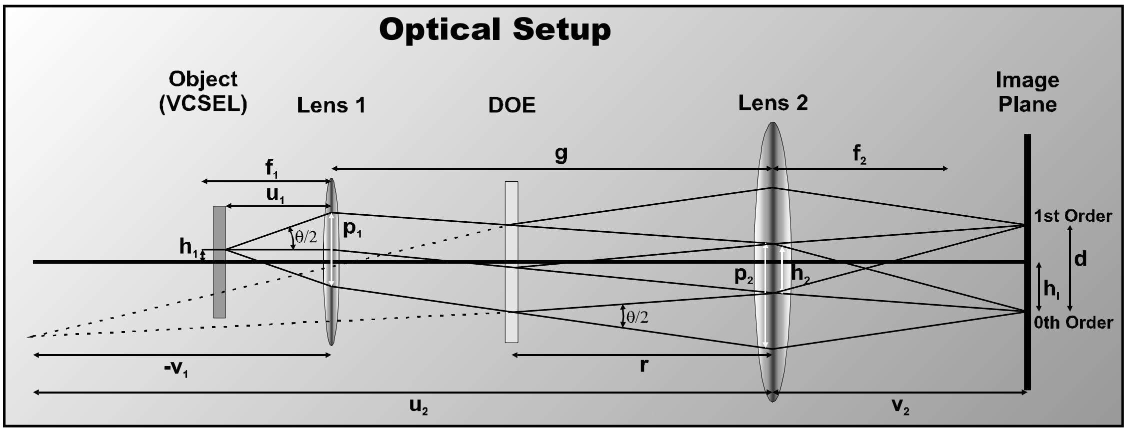


Figure 13

Basic optical design of lens system. All values in this diagram can be calculated given certain known values and simple lens formulae.

4.3.1 Known Parameters

The first task was to determine all known parameters. Using these as a basis, the system can then be further characterised:

f_1: Focal length of Lens 1 (mm).
f_2: Focal length of Lens 2 (mm).
d_1: Diameter of Lens 1 (mm).
d_2: Diameter of Lens 2 (mm).
L: Working distance of DOE (mm).
g: Separation of Lens 1 and Lens 2 (mm).
d: Displacement between 1st and 2nd orders (mm).
M: Magnification desired for entire system (usually negative).
d_{doe}: Diameter of the DOE (mm).
v_{size}: VCSEL size (square) (mm).
v_{nx}: Number of VCSELs in array x direction.
v_{ny}: Number of VCSELs in array y direction.
θ: Divergence in radians.
θ_{div}: Divergence in radians of beam between Lens 1 and Lens 2.
T_{f1}: Lens tolerance of f_1 against u_1 (percent).

4.3.2 u_1: Distance between VCSEL array and LENS 1

u_1 can be determined using the formula in equation 16. This value should essentially be around the same size as the focal length of Lens 1.

$$u_1 = \frac{\frac{f_2 . f_1}{m} + f_1 . (g - f_2)}{g - f_2 - f_1} \qquad \textbf{Equation 16}$$

4.3.3 v_2: Lens 2 to Image Plane

To find the distance between Lens 2 and the image plane, we first need to calculate a few other variables.

$$u_2 = g - \frac{f_1 . u_1}{u_1 - f_1} \qquad \textbf{Equation 17}$$

$$M_2 = \frac{f_2}{u_2 - f_2} \qquad \textbf{Equation 18}$$

Calculation of u_2 (equation 17) allows us to calculate M_2 (equation 18). Hence we can calculate the displacement between DOE and LENS 2, otherwise known as r (equation 19). Note that r cannot be greater than or equal to g since this would invalidate the system.

$$v_2 = \frac{L}{I - \frac{r}{u_2}} \qquad \textbf{Equation 20}$$

$$r = u_2 - \frac{L}{M_2} \qquad \textbf{Equation 19}$$

Finally we can calculate v_2 (equation 20).

4.3.4 w_1: Beam waist at Lens 1

The next task is to calculate the diameter of the beam at Lens 1: If it is larger than d_1 then the system will not work since the image is to large to fit through Lens 1. First we must calculate the furthest point from the axis to be imaged (equation 21).

$$h_1 = \frac{\sqrt{(v_{nx} . v_{size})^2 + (v_{ny} . v_{size})^2}}{2} \qquad \textbf{Equation 21}$$

$$P_1 = 2u_1 \tan\left(\frac{\theta}{2}\right) \qquad \textbf{Equation 22}$$

However, the beam from the furthest point still diverges and this additional distance is calculated as in equation 22. Combining these calculations gives a beam waist as shown in equation 23.

$$w_1 = 2\left(\frac{|P_1|}{2} + |h_1|\right) \qquad \textbf{Equation 23}$$

4.3.5 w_2: Beam waist at Lens 2

Analogous to w_1 above, we can calculate the beam waist at Lens 2.

$$h_2 = h_1\left(\frac{g - f_1}{f_1}\right) \qquad \textbf{Equation 24}$$

$$v_1 = \frac{1}{\left(\frac{1}{f_1}\right) - \left(\frac{1}{u_1}\right)} \qquad \textbf{Equation 25}$$

First we must calculate h_2 (equation 24) followed by v_1 (equation 25). This beam waist also incurs additional size due to divergence, as shown in h_2 (equation 26).

$$P_2 = \left|\left(\frac{u_2}{v_1}\right) \cdot P_1\right| \qquad \textbf{Equation 26}$$

To complete the calculation all we need do is calculate equation 27. Once again, if w_2 is greater than or equal to d_2 the system will not be able to function correctly.

$$w_2 = 2\left(\frac{|P_2|}{2} + |h_2|\right) \qquad \textbf{Equation 27}$$

4.3.6 h_I: Image Size

To ensure that all calculations are correct, a quick check can be made by calculating the image size h_I.(equation 28): its value should be M times the magnitude of h_1.

$$h_I = \frac{h_2.M.f_1}{(g - f_1)} \qquad \textbf{Equation 28}$$

4.3.7 w_H: Beam Waist at DOE

One of the preconditions of this system is that there is not a focal point between Lens 1 and Lens 2 i.e. f_1+f_2>g. This is advantageous in that we can calculate the divergence of the beam θ_{div} (equation 29) between Lens 1 and Lens 2 using trigonometry. A diverging beam is represented by a positive value, a converging by a negative value.

$$\theta_{div} = 2\tan^{-1}\left(\frac{w_2 - w_1}{2g}\right) \qquad \textbf{Equation 29}$$

This allows us to calculate the beam waist at the DOE w_H (equation 30). If this value is larger than the DOE's diameter then the system will again be invalid.

$$w_H = w_1 + 2\tan\left(\frac{\theta_{div}}{2}\right).(g - r) \qquad \textbf{Equation 30}$$

4.3.8 A Distance Model

The Matlab program produced hundreds of values on each test pass as g was gradually varied, so a method was needed to grade each result. It was decided that a value which represented the overall optical system size and also beam divergence between Lens 1 and Lens 2 should be used (Note that system size was considered to be twice as important as beam divergence). This allowed the quality of any valid system solution to be estimated while the program exhaustively tried different lens combinations and varied g.

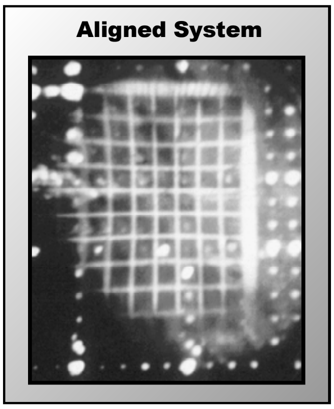


Figure 14

System after re-alignment.

4.3.9 Lens System Solution

Given tolerances of a maximum system size of 1000 mm, 5 mm minimum distance between components and maximum deviation of VCSEL to LENS 1 distance u_1 against focal length f_1 of 50%, the program gave the test results shown in section 10.1. The best distance measure had a solution for g=15mm with f_1=40 mm and f_2=80 mm. This solution was implemented and after careful alignment proved to be a valid solution.

Figure 14 shows a photograph of four VCSELs being projected onto the detector array as before.

Although the image quality is poor what is important here is that each of the projected orders from the VCSELs land exactly on a detector.

4.4 Detector Array

	V_p max (V)	V_p min (V)	I_{cc1} min (μA)	I_{cc1} min (μA)
Minimum	3.80	0.50	0.30	3.60
Average	4.10	0.83	0.46	3.98
Maximum	4.30	1.00	0.90	4.20
St. Dev.	0.12	0.10	0.14	0.14
With Minimum Error (-0.1 from all values)				
Average	4.00	0.73	0.36	3.88
With Maximum Error (+0.1 on all values)				
Average	4.20	0.93	0.56	4.08

Figure 15

The detector array is a 10 by 10 matrix with a spacing of 1.5 mm between the centre of each detector. Obviously, not the entire matrix is needed and only the middle 6 rows and 8 columns are actually used. This section tested the detector array by examining the sensitivity range of each element used. A diagram of the detector electrical circuit can be seen in figure 10 and is marked as 'Detector Board'. The problem associated here was that because the system was on a pre-fabricated board it was only possible to take measurements at specific points. The two values chosen were:

I_{cc1}: The current sunk through the photodiode is directly proportional to the amount of light detected. If the efficiency of the photodiode array was known it would be possible to calculate the exact amount of light in watts, but unfortunately it was not.

V_p: Voltage output from the detector board pre-amplifier.

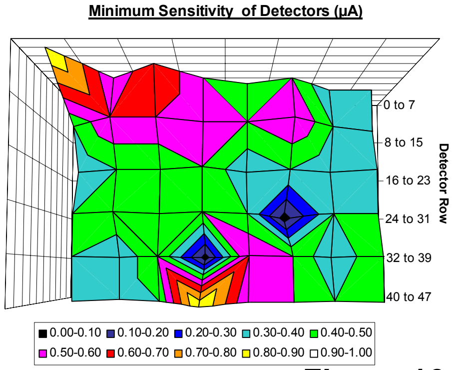


Figure 16

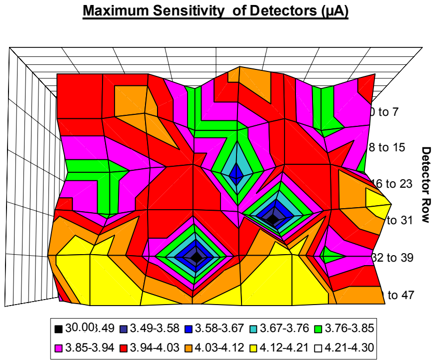


Figure 17

The experimental setup simply consisted of a VCSEL's output being directed through an aperture onto a single detector. By slowly increasing the power, it was possible to determine the minimum amount of current which needed to be sunk to start having an effect on the output voltage. The same method was also used to find the point at which the detector board saturated and any further difference in incident intensity would not be detected. This allowed determination of the working range. Figures 16 and 17 graph the results of minimum and maximum photo-currents with statistical analysis in figure 15. Detailed are results available in Appendix D.

It can be easily seen that, due to a complete lack of sensitivity, detectors 29 and 35 are not working correctly. This result proves significant in that if these detectors are avoided during testing it will prevent erroneous results. In addition, it was also detected that the detectors for channels 9 and 10 were wired round the wrong way.

Note that problems were encountered with the connectors between both detector board and amp-board. Fortunately, an easy method was found to diagnose this problem: the detector's output, when measured at the amp-board, will be seen to float about 2-3V with no light rather than the normal of ~4V.

4.5 The Diffractive Optic Element (DOE)

This project also examined the efficiency of the DOE. Various VCSELs were chosen at random and a driven such that their output power did not saturate the detectors. The photo-current sunk by each detector was then measured thus allowing a comparison of the optical power in each order.

The problem with this examination is that there are many sources of error, ranging from imprecision in VCSEL and driver output to detector non-linearity. However, to help reduce channel specific values, the optical powers were normalised against the 0th order thus making them more comparable to one another.

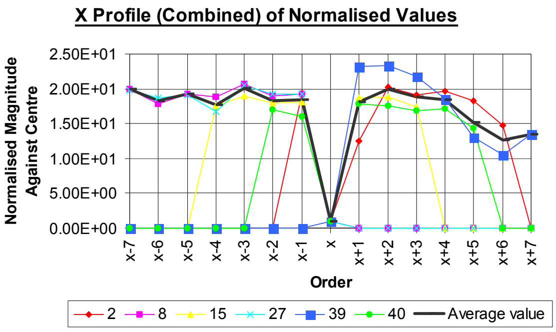


Figure 18

The *x* axis is the horizontal axis when viewed on the detector array. The 0th order is *x* and is found at the centre of the cross.

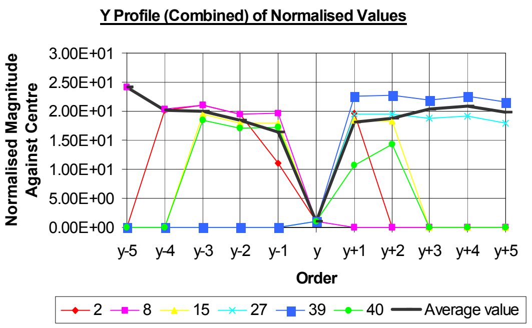


Figure 19

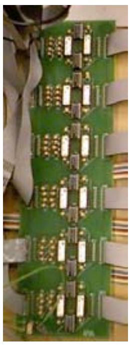
The *x* axis is the horizontal axis when viewed on the detector array. The 0th order is *x* and is found at the centre of the cross.

Figures 18 and 19 show the results taken for a random set of channels (Appendix E shows more detailed results). These graphs consider the x orders to be the horizontal line of the DOE output when looking onto the detector array and y orders the vertical.

The most important line here is the 'average value'. This is the best indication of the response of the DOE. It clearly shows that most of the orders seem fairly stable at 20 times the magnitude of the zero order: except for in the positive x direction where x+5 and x+6 prove to be consistently low.

4.6 Electronic Modules

Figure 20



4.6.1 Amp-Board

This module is designed to amplify the output from the detector board and also includes a high pass filter to remove any DC component from the input signal. Figure 10 shows the layout of the amp-board and figure 20 a picture.

This module was tested by inputting a signal which swept the entire voltage range output by the stage before it. With amplification set at –1, the expected inverted output was received. This test was repeated for each and every channel and the output monitored.

Testing found a damaged amplifier chip where one of the four operational amplifiers was not working as expected. The damaged chip was promptly replaced. This implementation uses the Texas Instruments operational amplifier LM324N as detailed in Appendix F.

4.6.2 Neural Switch Card

Before testing could commence, it was also necessary to test and calibrate the neural switch card. Figure 10 shows the neural switch card's layout and figure 21 a photograph of the implemented system.

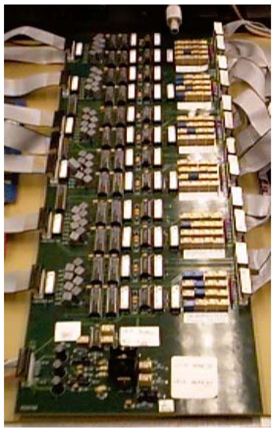


Figure 21

The first task was to set up correct reference voltages, as defined previously by calculation [3]:

V_{start} = VR9 = 5.01V ±0.001V
$V_{rio/ref}$ = VR10 = 2.81V ±0.001V
V_{off} = VR11 = 3.92V ±0.001V
Analog 7V = VR12 = 7.000V ±0.5V
Analog 6V = VR13 = 6.000V ±0.001V

Significant instability was noticed on the analog 7V channel and a dry joint was suspected. Careful soldering in the suspected area lead to its discovery and after re-soldering the reference voltage became stable:

Analog 7V = VR12 = 7.000V ±0.001V

The next step was to calibrate the VCSELs using available optical output power versus drive current data. A solution was devised where an ammeter was inserted into the circuit just before the VCSEL to measure drive current. A square wave was then applied to the channel being measured with a frequency of 0.5Hz so that the full range of neuron input voltages were swept (i.e. input voltage between the amp-board's output limits). By observing the drive current carefully, minimum and maximum values could be determined, allowing variable resistors R_{10} and R_{11} (figure 10) to be adjusted to give the appropriate optical power output. The optical output powers chosen were 0.05mW representing an 'off' state and 0.8mW for 'on'. Previous data is available on the HP Workstation under 'VSL1:DATA6'. This method of testing also had the advantage that the electrical circuit for each neuron would be tested simultaneously.

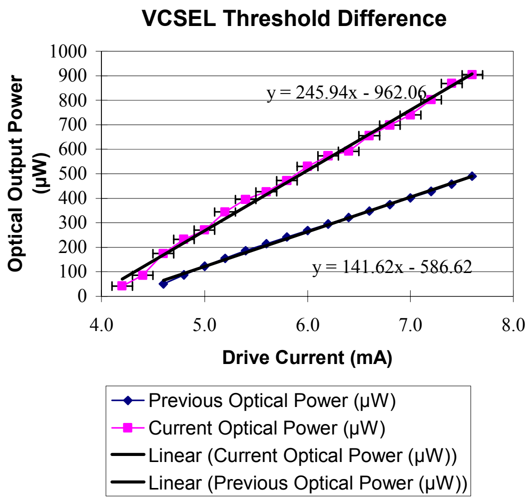


Figure 22

Before adjustment of the system could begin it was necessary to verify the validity of the previous data. The reason for this was that the VCSELs were originally profiled in a colder environment than that during the experiment. Any large variance in threshold would indicate a temperature dependent change in characteristics. One VCSEL was chosen at random and its optical output power versus drive current curve plotted to find its threshold (see figure 22). The data on the graph allowed determination of a change in threshold: all new thresholds are now 94% of the original.

A foreseeable problem was that the resistance of the ammeter would be high in comparison to that of the VCSEL. The manufacturer's data and application of Ohm's law allowed calculation of minimum (235Ω) and maximum (500Ω) VCSEL resistances, dependent on drive current. Measurement of the ammeter's resistance showed that it was 7.6Ω. This is a worst case difference in resistance of 3.2% which was considered unacceptable. The range on the ammeter was then changed and one selected which had a resistance of 1.1Ω (0-200mA). This gave an influence of 0.47% worst case and is well within tolerable limits.

Conversion of all values extracted from the HP workstation was also required since it only displays the optical power for a given current in 0.2mA steps. Presuming that the increase between two points is relatively linear, we can create a formula to calculate the desired optical power output given current and optical powers of the two points next to it. Note that equation 31 also takes into account the temperature change, where:

P_{req} = Power output desired.
P_U = Power output from VCSEL with a drive current of I_U. These are the upper (or higher) values.
P_L = Power output from VCSEL with a drive current of I_L. (not used in this equation). These are the lower values.
I_{req} = Current to be used to drive VCSEL.

$$I_{req} = \left(\frac{3.91}{4.14}\right) \cdot \left(\frac{P_{req} - \left(P_U - \left(\frac{P_U - P_L}{0.2} \right) I_U \right)}{\left(\frac{P_U - P_L}{0.2} \right)} \right)$$

Equation 31

Calculation of each value can be found in Appendix G.

This data now allows calibration of the Neural switch card. Systematic adjustment of R_{10} and R_{11} should swing the VCSEL current between the two desired values for the appropriate channel. Before calibrating any VCSEL, it was ensured that both variable resistors were at absolute minimum power out. Even so, VCSELs began to fail during calibration. Careful examination revealed that when negative was not connected on the ammeter there was an AC coupling of ~±1.1V present. Any negative bias is capable of damaging a VCSEL if it exceeds ~-2V (the tolerance of which is not known): but this should not be enough to cause considerable damage. A very serious problem was noticed later: the outputs from the neural switch card take on a -5V potential when the negative terminal is not connected. Avoidance of this situation was made to prevent possible damage to any more VCSELs.

Once calibration was completed, the following points were noted:

Ch.	**Notes**
0	VCSEL fail. Power outputs calibrated.
3	VCSEL fail. Cannot calibrate: biases set to minimum.
10	Drive current low. 750Ω connected in parallel with 470Ω bias.
16	Detector and VCSEL fail. Cannot calibrate so current set to minimum.
17	VCSEL fail. Calibrated.
18	Drive current low: parallel 1kΩ resistor connected.
19	VCSEL fail. Calibrated.
22	VCSEL fail. Cannot calibrate so current set to minimum.
24	VCSEL works but optical power output low. Current calibrated.
34	VCSEL fail. Calibrated.
35	Detector fail.
36	Drive current low: parallel 750kΩ resistor connected.
37	VCSEL fail. Calibrated.
38	Drive current low: parallel 750kΩ resistor connected. VCSEL fail.
39	VCSEL fail. Cannot calibrate so current set to minimum.

Some channels are marked as 'cannot calibrate'. This is not actually the case as all channels could be calibrated if components were replaced. However, there is little point in doing this as the VCSELs do not work in the first place so the currents were set to a minimum so that as little power as necessary was drawn.

The only task left was to test the system.

4.7 Investigation of System Response

This section examines the complete system where all components and modules were assembled and tested. Figure 23 lists all component values with reference to figure 10 for component integration.

Component Values	
R_1=100Ω	R_2=100kΩ
R_3=470Ω	R_4=1kΩ
R_5=3.3kΩ	R_6=100kΩ
R_7=100kΩ	R_8=100kΩ
R_9=470Ω	R_{10}=500Ω
R_{11}=1kΩ	R_{12}=1kΩ
C_1=47pF	C_2=10nF

Figure 23

During testing, all channels with failed VCSELs, low power VCSELs and failed detectors were not used – these channels are listed in section 4.6.2. Optical alignment was again re-checked to ensure accuracy. Three important points were carefully re-checked:

- Total VCSEL power output did not saturate detectors in off state.
- V_{ref} produced a correct response.
- Amplification on amp-board was set correctly.

It was found that the total power output was too high, so instead of laboriously re-calibrating every part of the system a beam splitter was simply inserted.

Three channels were chosen at random from the usable selection and their output examined. It became clear that certain neurons seemed to have priority over others. Examination of the system showed that the VCSELs did not seem to be correctly calibrated and switching on some VCSELs induced a photo-current twice the size of others. The induced photo-currents were therefore carefully examined as can be seen in Appendix H. These measurements proved that the VCSELs were indeed miscalibrated but what was not clear was by how much. Although not considered before, examining figure 22 shows that not only is the threshold different but so is the gradient. This threw into question the accuracy of the calibration data so the optical power output from a few VCSELs was measured: and found to be drastically different. For example, channel 7 produced 1.293mW when turned on while channel 36 produced 2.193mW – both were supposedly calibrated at 0.8mW and such output powers are beyond VCSEL's safe operating limits.

Time constraints at this point in the project prevented re-calibration of the VCSEL array, so a set of channels were selected that had a similar induced photo-current level of 1.6µA per detector (±0.1µA), as shown in equation 32.

$$\text{Selected} = \begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array} \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Equation 32

All channels which contain a 1 were selected for testing due to similar VCSEL characteristics.

Testing was performed by requesting a set of neurons and examining which turned on using the program 'NETRUN' on the HP workstation. If the neurons which turned on indicated a valid and optimal solution then the test was considered successful. The test data is saved in a file on the HP workstation under HOP:TSEQ.

Figures 33 and 34 show some sample results and outputs with more detailed results in Appendix I.

$$\begin{array}{c} \text{Request} \\ \begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array} \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array} \Rightarrow \begin{array}{c} \text{Response} \\ \begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

$V_{ref} = 0.78V$

Equation 33

During testing it became obvious that V_{ref} played an important role in as far as finding a valid solution is concerned, sometimes requiring extremely fine adjustment.

$$\begin{array}{c} \text{Request} \\ \begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array} \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \end{array} \Rightarrow \begin{array}{c} \text{Response} \\ \begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array} \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \end{array}$$

$V_{ref} = 0.78V \to 0.79V$

Equation 34

Invalid solution to request.

Examination of the system indicated that detector saturation could be causing a problem, thus photographic film was inserted into the system which absorbed ~33% of throughput power. This did result in valid solutions for higher power levels, but not for lower ones: V_{ref} had to be adjusted to a specific level before the system would find a solution for request matrices.

There was obviously something more fundamentally wrong with the system than simply a power problem. The next stage was to check the amplifier outputs (V_{inv}, figure 9) and ensure they were as expected. It was decided to monitor two neuron outputs: that of a neuron which was requested but turned off and that of a neuron which was not requested, nor did it turn on.

A major problem immediately became apparent: when a neuron turns off, it should fall from ~4V to the same value as the switched off neuron (V_{off}, ~2V) before switch off time. Unfortunately it does not and will go no lower than ~3V. The first solution was to increase amplification on the amp-board but this only resulted in the neuron choosing quicker and still going no lower than ~3V.

Various attempts were made to bring the minimum value down from ~3V to ~2V including increasing the amp-board drive voltage from 5V to 10V. This solution, although helpful, still did not solve the problem.

Next, an attempt was made to adjust the voltage levels of $V_{rio/ref}$, V_{off} and V_{start}. This started to alter the voltage levels, but because of the circuit design it was not possible to adjust them to a great enough degree. After trying various methods it was concluded that without changing component values or perhaps even re-designing the reference voltage system on the neural switch card it would not be possible to create a fully working system.

There is also one final point that any further work should consider: the system seemed very sensitive to any movement of the inserted beamsplitter, suggesting that the filter is setting up a resonance cavity. Either this possibility should be investigated or the filter replaced by some other method of reducing optical throughput.

5 Conclusion

This report has carefully looked at the theory and implementation of an optoelectronic neural network for switching and provided some promising results. It has been shown that, with further work, the optical neural network can be implemented as proposed. Nevertheless, various problems still need to be eradicated in the hardware system, one of which being size. Even though the system is exceedingly efficient at routing, it still faces the problem of hardware complexity when embedded in large switches.

What makes this system so interesting is its diversity: switching is only one of its many applications. Essentially, this system could 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 such systems could make to the field of computing.

5.1 Future Work

There are a few areas which need refinement in this system, but to bring it into working order the following two recommendations should be carried out:

- Each VCSEL needs to be re-profiled so that the system can be calibrated correctly.
- The neural switch card needs to be modified so that the reference voltages can be varied over a larger range.

A further interesting point is temperature sensitivity: in particular that of the VCSELs. The current VCSEL characteristics differ dramatically from those measured beforehand – the only change being a temperature difference. Although it is unlikely that such a large difference was caused by air temperatures in hot and cold rooms, it is worth eliminating as a possible cause.

5.2 Acknowledgements

I wish to acknowledge and thank both Rod Webb and Mohammed Taghizadeh for their help and advice during my industrial project.

6 Glossary

AC	Alternating Current
ATM	Asynchronous Transfer Mode
DC	Direct Current
DOE	Diffractive Optic Element
TSP	Travelling Salesman Problem
VCSEL	Vertical Cavity Surface Emitting Laser

7 Bibliography

This Bibliography includes some comments on certain references. The idea is to help assess the relevance of any source before it is looked up. Some comments require an understanding of the system described in this report.

[1] Peter W. Protzel, Daniel L. Palumbo and Michael K. Arras, *"Performance and Fault-Tolerance of Neural Networks for Optimisation"*, IEEE Transactions on Neural Networks, volume 4, number 4, July 1993.

This paper discusses neural network applications for solution of both the assignment problem and travelling salesman problem and the inherent advantages/disadvantages of such a solution in any situation.

[2] M. Gell, *"Constrained Optimisation of Neural Networks for Switching: Hopfield Neural Networks"*, Patent Application (draft) to Kilburn and Strode, MNM/BH/P16935GB, IPD Case A24518, 13th January 1993.

Patent application by BT to Kilburn and Strode. Does a nice job of explaining the theoretical system and current application.

[3] Andreas Ludolph, *"Examinations on a Neural Switch Controller"*, Master of Science Dissertation Project in Applied and Modern Optics at University of Reading, September 1998.

Explains the current system in detail. More examination of the electronic side of things. Also includes an examination of a terabit backplane.

[4] C. Bousoño-Calzón and M. R. W. Manning, *"The Hopfield Neural Network Applied to the Quadratic Assignment Problem"*, BT Labs paper, Martlesham Heath, Ipswich, IP5 7RE, publication date unknown.

Examines the Quadratic Assignment Problem and current solutions from a problem complexity point of view. It proposes using a neural algorithm to solve the QAP and compares it to these current solutions concluding that the neural approach has performance and scalability benefits.

[5] Joydeep Ghosh, Ajat Hukkoo and Anjun Varma, *"Neural Networks for Fast Arbitration and Switching Noise Reduction in Large Crossbars"*, IEEE Transactions on Circuits and Systems, volume 38, number 8, August 1991.

Presents two VLSI solutions for large crossbar switching systems controlled by neural networks. It examines their performance and draw the conclusion that hierarchical control is superior to batch control.

[6] M. R. W. Manning and M. Gell, *"Evaluation of the Hopfield Neural Network for Service Assignment"*, BT Labs paper, Martlesham Heath, Ipswich, IP5 7RE, publication date unknown.

Discusses the parameters A, B, C and D and their effect on a Hopfield neural network when configured for switching.

[7] 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 Transactions on Neural Networks, volume 4, number 2, March 1993.

This paper examines a series of networks that are closely related to the Hopfield-Tank model. It concludes that, although these models do not achieve optimal performance, their performance (measured by simulation) is very similar to the HT model.

[8] J. J. Hopfield and D. W. Tank, *"'Neural' Computation of Decisions in Optimisation Problems"*, Biological Cybernetics, volume 52, pages 141-152, 1985.

This is the defining paper for the Hopfield-Tank (HT) model. The results of computer simulations are presented here to illustrate the problem solving power of neural networks.

[9] R. D. Brandt, Y. Wang, A. J. Laub and S. K. Mitra, *"Alternative Networks for Solving the Travelling Salesman Problem"*, IEEE International Conference on Neural Networks, 24th to 28th Feb. 1998, San-Diego.

Examines a HT neural network model with a modified energy function and concludes that it has better performance than their model of the HT network. The paper then proceeds to examine fixed parameter networks which have superior fabrication ease.

[10] T. X. Brown, *"Neural Networks for Switching"*, IEEE Communications Magazine, November 1989.

Examines two possible configurations of Neural Nets for crossbar switch control. Consideration is also given to parallel machines.

[11] T. X. Brown, *"Chapter3: Controlling Circuit Switching Networks"*, Extract from T. X. Brown's Thesis from CalTech.

A good introduction to the switch and how a neural network will be mapped onto it. Reference [10] is a more comprehensive but less detailed version.

[12] J. J. Hopfield, *"Neural Networks and Physical Systems with Emergent Collective Computational Abilities"*, Proc. Natl. Acad. Sci. USA, volume 79, pages 2554-2558, April 1982.

Initial paper examining and defining the neural network and the role of the neuron. It combines these neurons and examines the emergent collective properties exhibited.

[13] J. J. Hopfield, *"Neurons with Graded Response Have Collective Computational Properties Like Those of Two-State Neurons"*, Proc. Natl. Acad. Sci. USA, volume 81, pages 3088-3092, May 1984.

Builds upon the original model in [12] to create a large network of neurons with graded response. Proposes an electrical model of the system.

[14] A. Marrakchi and T. Troudet, *"A Neural Net Arbitrator for Large Crossbar Packet Switches"*, Circuits and Systems Letters, IEEE Transactions on Circuits and Systems, volume 36, number 7, July 1989.

Proposes a Hopfield Neural Network architecture for controlling a crossbar switch and backs up proposition with simulation results and a VLSI system.

[15] S. U. Hegde, J. L. Sweet and W. B. Levy, *"Determination of Parameters in a Hopfield Tank Computational Network"*, Depts. of Computer Science and Neurosurgery, University of Virginia, Charlottesville, Virginia 22903, publication date unknown.

Higher level examination of network size in relation to the parameters in a Hopfield neural network leads to the conclusion that it does not scale well when applied to the TSP or any related problem.

[16] S. B. Aiyer, M. Niranjan and F. Fallside, *"A Theoretical Investigation into the Performance of the Hopfield Model"*, IEEE Transactions on Neural Networks, volume 1, number 2, June 1990.

An in depth theoretical mathematical analysis and proof of the Hopfield Neural Network. Proposes a solution to make the network robust and allow higher order (up to 50) TSP problems to be reliably solved.

[17] R. P. Webb, *"Optoelectronic Implementation of Neural Networks"*, International Journal of Neural Systems, volume 4, number 4, pages 435-444, December 1993.

Proposes three solutions for implementing Optoelectronic Neural Networks all of which have been demonstrated operational up to 50Mhz.

[18] P. S. Guilfoyle and F. F. Zeise, *"Global Optical Free Space 'Smart' Interconnects"*, Proceedings of the SPIE, code 1849-20.

Proposes a new architecture for implementing an N bit Boolean multiplier which uses N4 optical interconnects.

[19] P. S. Guilfoyle, F. F. Zeise and R. V. Stone, "Diffractive Optical Interconnect Element Design for Digital Switching", OptiComp Corporation, Zephyr Cove, NV 89448-2889, 1994.

Examines the advantages of using a diffractive optical interconnect element over conventional bulk elements for digital switching.

[20] R. P. Webb, Internal documentation, BT Labs, Martlesham Heath, Ipswich, IP5 7RE, Various dates.

[21] R. P. Webb and A. W. O'Neill, *"Optoelectronic Neural Networks"*, BT Technology Journal, Volume 10, No. 3, pages 144-154, July 1992.

[22] M. R. Garey and D. S. Johnson, *"Computers and Intractability"*, New York, W. H. Freeman, 1979.

[23] J. Munkres, *"Algorithms for Assignment and Transportation Problems"*, J. Soc. Ind. Appl. Math., 5, 32-8, 1957.

[24] M. M. Ali and H. T. Nguyen, *"A Neural Network Controller for a High Speed Packet Switch"*, 1990.

[25] B. S. Wherret, *"Optics in Computing"*, Lecture Notes from Heriot-Watt University, 1998.

[26] M. Taghizadeh and J. Turunen, *"Synthetic Diffractive Elements for Optical Interconnection"*, Optical Computing and Processing, 2 (4), pages 221-242, (1992).

[27] F. A. P. Tooley, *"Optoelectronic Devices"*, Lecture Notes from Heriot-Watt University, 1998.

[28] A. J. Waddie, *"Optoelectronic Neural Networks: The Design and Analysis of a Network Based Around Optoelectronic Smart Pixels"*, PhD Thesis, Heriot-Watt University, September 1995.

[29] Robert L. Harvey, *"Neural Network Principles"*, Prentice Hall International Editions, 1994.

[30] James A. Anderson, *"An Introduction to Neural Nets"*, IT Press, 1995.

[31] Simon Haykin, *"Neural Networks"*, Macmillan Publishing Company, 1994.

[32] Clifford Lau, *"Neural Networks: Theoretical Foundations and Analysis"*, IEEE Press, 1991.

[33] A. J. Waddie and J. F. Snowdon, *"A Smart Pixel Optical Neural Network Design Using Customised Error Propagation"*, Department of Physics, Heriot-Watt University.

[34] D. Huang, J. F. Snowdon and A. J. Waddie, *"Free Space Optical Technology and Neural Network Applications"*, Paper presented at Appl. Opt. Div. Conf. Reading, 16-19 September 1996.

[35] J. Hertz, A. Krough and R. G. Palmer, *"Introduction to the Theory of Neural Computation"*, Addison-Wesley, 1991.

8 Appendix A

This section contains Matlab V4.2.1c for Mac code for theoretical simulation of the neural network used in this project.

8.1 Theory_model.m

```
% Theoretical Switch Controller
% Step by step

function Theory_model()

% Clear all variables, functions and MEX links.
clear all
% Set up neuron type.
program = 'Theory';
% Indicate startup and tell user neuron type.
fprintf('____________________________________________________\n')
fprintf(['Running network with ', program, '\n'])

% Set up local variables.
order=10;                       % Order x order crosspoints.
A0=1250;                        % A=B. Weights to elements in same row or column.
A=A0*ones(order);               % Create a matrix of size 'order' where all values are A0.
C=100;                          % Set optimisation value.
dt=0.1;                         % Time increment.
Tlpf=3;                         % LPF time constant.
Tph=10*Tlpf;                    % Length of run: 10 times Tlpf.
noise=1e-3;                     % rms noise amplitude.
slope=0.02;                     % For linear neuron (max sigmoid slope for beta = 0.08)
randn('seed', cputime);         % Choose new seed for gaussian noise based on CPU time.

request=ones(order);            % Intial requested crosspoints.
% request=tril(ones(order));    % Intial requested crosspoints.
trecord=[0: dt: Tph];
X=zeros(order);                 % Initial states.
Y=zeros(order);

% Initialise memory for record of successive states
Xrecord=zeros(length(trecord), order^2);
Yrecord=zeros(length(trecord), order^2);
% Start with initial states: Fill row 1 of Xrecord with contents of X (same for Y).
Xrecord(1,:) = X(:)';
Yrecord(1,:) = Y(:)';

% Start timing.
tic

% Repeat for every element in trecord.
for i=2: length(trecord)

        % Amplify and truncate each neuron output, then multiply by request.
        Y=lin_neuron(X, slope).*request;

        % Update input voltage to each neuron
        X=X+dt/Tlpf.*(-X-A.*Xbar_wts(Y)+C/2)+noise.*randn(order);

        % Let the user know it's alive.
        if (rem((i-1), 50)==0)
                fprintf('.\n');
        else
                fprintf('.');
        end;

        % Record successive states.
        Xrecord(i,:)=X(:)';
        Yrecord(i,:)=Y(:)';
end

% Tell the user that your finished.
fprintf(['Finished. Time taken = %5.1f sec.\n'], toc)
fprintf('____________________________________________________\n')
figure ('Name', 'Final output')
Start_end_image(trecord, Yrecord, request)
plot_x
Xmax=max(max(Xrecord));
```

```
% Axis([trecord(1), trecord(length(trecord)), -Xmax, Xmax])
plot_y
```

8.2 Lin_neuron.m

```
% Y=lin_neuron(X, slope)
%
% Each element in X is multiplied by slope and has 0.5 added.
% Values then truncated to within [0, 1].
%
% Amplifier with gain = slope acting on elements of X.
% Output limits at 0, 1.
% lin_neuron(0)=0.5.
function Y=lin_neuron(X, slope)

Y=max(0, min(1, 0.5+slope.*X));
```

8.3 Xbar_wts.m

```
% Ysum(i,j) is the sum of row i + the sum of column j in Y excluding element Y(i,j).
function Ysum=Xbar_wts(Y)

Ysum=sum(Y')'*ones(1, size(Y, 2))+ones(size(Y, 1), 1)*sum(Y)-2*Y;
```

8.4 Plot_y.m

```
% Plot evolution of outputs
figure ('Name', ': outputs')
plot(trecord, Yrecord); % plot Y/time for each neuron
grid
xlabel ('time');
ylabel ('Y');
```

8.5 Plot_x.m

```
% Plot evolution of inputs
figure ('Name', [program, ': inputs'])
plot(trecord, Xrecord); % plot X/time for each neuron
grid
xlabel ('time');
ylabel ('X');
```

8.6 Start_end_image.m

```
% Show outputs as image
function Start_end_image(trecord, Yrecord, request)

% Create colormap
maplength = 16;
shade1=[0, 0, 0.5];                          % 'bottom' shade for colourmap (R, G, B).
shade2=[1, 0, 0];                   % 'top' shade for colourmap.
map=[linspace(shade1(1),  shade2(1),  maplength)',  linspace(shade1(2),  shade2(2),  maplength)',
linspace(shade1(3), shade2(3), maplength)'];
colormap(map)

t=trecord(length(trecord));
order=sqrt(size(Yrecord, 2));
Y=reshape(Yrecord(size(Yrecord, 1), :), order, order);

image(max(Y, request/2)*maplength)
axis square
title(['Request & final state. t = ', num2str(t)]);
drawnow
```

9 Appendix B

This section contains Matlab V4.2.1c for Mac code for theoretical simulation of the electronic and optical system used to implement a neural network.

9.1 Run_Circuit.m

```
% Run circuit runs the Optical Network simulation
% Code cleaned up by Keith Symington
% Author unknown.

% Clear all variables and pack memory.
fprintf('Initialisation: Memory cleanup...');
clear all;
pack;
fprintf('done.\n');

% Count flops.
tic
% Define the neuron being used.
program='Neuron8 (linear comparator)';
fprintf('____________________________________________________\n');
fprintf(['Running circuit with ', program, '\n']);

% Set up all global variables.
global Kd Vpb Vpmin Rf Thpf Vcb Gc Vcmax Vcmin Vinrange Tlpf Voff Vstart Vref Rl Kl comp_noise
% Initialise parameter settings.
Init_circuit;

% Set up initial states.
Y=Ninverter(0,Vstart,Vref,0).*ones(order).*Kl./Rl;

% Initialise memory for record of successive states.
fprintf('Initialisation: Memory allocation for recording progress...');
Xrecord=zeros(length(trange), order^2);
Yrecord=zeros(length(trange), order^2);
Vprecord=zeros(length(trange), order^2);
Vhpfrecord=zeros(length(trange), order^2);
Vcoutrecord=zeros(length(trange), order^2);
Vlpfrecord=zeros(length(trange), order^2);
Vinvrecord=zeros(length(trange), order^2);
fprintf('done.\n');

% Calculate for every value in the time sequence.
for i=1:length(trange)
        % Select time value for appropriate iteration.
        t=trange(i);
        % Is this the first cycle?
        if i>1 dt=t-trange(i-1);
        else dt=trange(2)-t;
        end

        % Set enable dependent on iteration number.
        enable=enablerange(i);

        % Optical Input.
        X=H.*Xbar_wts(Y)+noise.*randn(order);
        % Optical Output.
        [Vp,Vhpf,Vcout,Vlpf,Vinv,Y]=neuron8(X,enable,request,dt);

        % Record successive states.
        Xrecord(i,:)=X(:)';
        Yrecord(i,:)=Y(:)';
        Vprecord(i,:)=Vp(:)';
        Vhpfrecord(i,:)=Vhpf(:)';
        Vcoutrecord(i,:)=Vcout(:)';
        Vlpfrecord(i,:)=Vlpf(:)';
        Vinvrecord(i,:)=Vinv(:)';
end

% Tell the user that the system is finished.
fprintf(['Finished. Time taken = %5.1f sec.\n'], toc)

% Reset tic.
tic;
```

```
% Draw the solution that the system proposed.
figure ('Name', 'Final output')
Out_image

% Plot the voltage characteristics at various stages.
plot_volts
fprintf(['Time taken for plotting = %5.1f sec.\n'], toc)
fprintf('____________________________________________________\n')
```

9.2 Init_circuit.m

```
% Intialise circuit

% Random seeds.
randn('seed', cputime);                   % Choose new seed for gaussian noise.
rand('seed', cputime);                    % Choose new seed for uniform noise.

order=6;
dt1=0.02e-6;                              % Time increment 1.
dt2=1e-6;                                 % Time increment 2.
Thpf=2.2e-3;                              % HPF time constant.
Ctol=0;                                   % capacitor tolerance.
Tlpfmean=33e-6;                           % mean value of LPF time constant.
Tlpf=Tlpfmean*(1+Ctol*rands(order));      % LPF time constant:
                                          % (Varies because of capacitor tolerance.)
Tph1=10e-6;                               % Length of settling period.
Tph2=0.3e-3;                              % Length of run.
Tph3=0;                                   % Length of run.

noise=1e-9;                               % Noise equivalent power (rms).
comp_noise=0.002;                         % Comparator noise.
Kd=0.5;                                   % Detector sensitivity.
Vpb=2.1;                                  % Preamp quiescent.
Vpmin=0.1;                                % Preamp lower limit.
Rf=1e6;                                   % Transimpedance.
Vcb=2.11;                                 % Comparator quiescent.
Gctol=0;                                  % Comparator gain tolerance.
Gcmean=213;                               % Mean comparator gain.
Gc=Gcmean*(1+Gctol*rands(order));         % Comparator gain.
Vcmax=3.92;                               % Comparator max.
Vcmin=0.31;                               % Comparator min.
Voff=Vcmax;                               % Off voltage.
Vstart=Voff-((Vcmax-Vcmin)/50)/(order-1); % Start voltage.
Vref=Vcb;                                 % Inverter reference voltage.
Vthresh=Ninverter(Vcb,Vstart,Vref,1);     % Threshold for classifying output.
Rl=1.2e3;                                 % Laser drive resistor.
Kltol=0;                                  % Laser A/W tolerance.
Klmean=0.24;                              % Mean Laser A/W.
Kl=Klmean*(1+Kltol*rands(order));         % Laser A/W.
Htol=0;                                   % Optical loss tolerance.
Hmean=2.0e-3;                             % Mean optical loss.
H=Hmean*(1+Htol*rands(order));            % Optical loss

% Requested crosspoints.
request=ones(order);
% Alternative requested crosspoints.
% request=tril(ones(order));

% Set up trange: Minimum time, step size and maximum time.
% Creates an array with an element for each step.
trange=[-Tph1: dt2:  Tph2+Tph3];
% Alternative trange.
% trange=[-Tph1: dt2: 0, dt1: dt1: Tph2, Tph2+dt2: dt2: Tph2+Tph3];

% enablerange is derived from trange: element contains a 1 when time
% has gone past zero.
enablerange=trange>0;
```

9.3 Ninverter.m

```
% Inverter with gated input.
% Used by neuron7 & neuron8.
function Vinv=Ninverter(Vin, Vstart, Vref, enable)

% Inverter output
Vinv=(2*Vref-Vin).*enable+(2*Vref-Vstart).*(~enable);

% Alternative inverter output
% Vinv=(2*Vref-Vin).*enable+(1.5*Vref-0.5*Vstart).*(~enable);
```

9.4 Rands.m

```
% rands, rands(m) or rands(m, n)
% Generate (arrays of) random numbers uniformly distributed between -1 and +1.
% Based on rand.
function R=rands(m, n)

if nargin==0
        R = 2.*rand-1;
elseif nargin==1
        R = 2.*rand(m)-1;
elseif nargin == 2
        R = 2.*rand(m, n)-1;
end
```

9.5 Xbar_wts.m

```
% Ysum(i,j) is the sum of row i + the sum of column j in Y excluding element Y(i,j).
function Ysum=Xbar_wts(Y)

Ysum=sum(Y')'*ones(1, size(Y, 2))+ones(size(Y, 1), 1)*sum(Y)-2*Y;
```

9.6 Out_image.m

```
% Output image
% Show outputs as image

% Create colormap
maplength = 16;
shade1 = [0, 0, 0.5];                          % 'bottom' shade for colourmap (R, G, B).
shade2 = [1, 0, 0];                                    % 'top' shade for colourmap.
map  =  [linspace(shade1(1),  shade2(1),  maplength)',  linspace(shade1(2),  shade2(2),  maplength)',
linspace(shade1(3), shade2(3), maplength)'];
colormap(map)

Ymin = min(min(Y));
Yrange = max(max(Y)) - Ymin;
image((Y-Ymin)*maplength/Yrange)
axis square
if exist('t')
        title(['Neuron ouputs. t = ', num2str(t)]);
else
        title('Neuron ouputs');
end
drawnow
```

9.7 Neuron8.m

```
% neuron8 approximates to real circuit.
% The comparator output is gated by the enable matrix.
% Q is the optical input.
% Y is the optical output.
function [Vp,Vhpf,Vcout,Vlpf,Vinv,Y]=neuron8(Q,enable,request,dt)

global Kd Vpb Vpmin Rf Thpf Tlpf Voff Vstart Vref Rl Kl comp_noise

Vp=max(Vpb-Q.*Kd.*Rf, Vpmin);                          % Preamp voltage.
Vhpf=HPF(Vp, dt./Thpf);                                % High-pass filter output.
Vcout=lin_comparator(Vhpf);                            % Comparator output.
Vlpf=LPF(Vcout, dt./Tlpf)+comp_noise.*rands(size(Vcout));   % Low-pass filter output.
Vsel=Vlpf.*request + Voff.*(~request);                 % Select outputs.
Vinv=Ninverter(Vsel,Vstart,Vref,enable);               % Inverter output.
Y=Vinv.*Kl./Rl;                                        % Light output.
```

9.8 HPF.m

```
% High-pass CR filter response.
% Vin = input voltage
% dt = time increment normalised to filter time constant (ie. dt/tau),
% Vinit = previous steady-state input voltage (optional). Defaults to Vin.
% If Vin and Vinit are vectors or arrays, they are treated as voltages on
% parallel filters, not as time series.
% Warning: must initialise Vinit or "clear all" if dimensions of Vin change!
function Vout = HPF(Vin, dt, Vinit)
```

```
% ..so that it can be remembered.
global Vcap_hpf

% Set up array for previous states if necessary.
if nargin==3
        % Set previous steady-state input voltage if given as a parameter.
        Vcap_hpf=Vinit;
elseif ~exist('Vcap_hpf')
        % Initialise Vcap_hpf if it doesn't exist already
        Vcap_hpf=Vin;
end

% Perform filtering.
Vcap_hpf=Vin.*dt+Vcap_hpf.*(1-dt);
Vout=Vin-Vcap_hpf;                % dt is assumed to be small
```

9.9 LPF.m

```
% Low-pass RC filter.
% Vin = input voltage,
% dt = time increment normalised to filter time constant (ie. dt/tau),
% Vinit = previous output voltage (optional).
% If Vin and Vinit are vectors or arrays, they are treated as voltages on
% parallel filters, not as time series.
% Warning: must initialise Vinit or clear all if dimensions of Vin change!
function Vout=LPF(Vin, dt, Vinit)

% ..so that it can be remembered.
global Vlast_lpf
if nargin == 3
        % Set previous output voltage if given.
        Vlast_lpf = Vinit;
elseif ~exist('Vlast_lpf')
        % Initialise Vlast_lpf if it doesn't exist already
        Vlast_lpf = Vin;
end

Vout = Vin.*dt + Vlast_lpf.*(1-dt);        % dt is assumed to be small
Vlast_lpf = Vout;
```

9.10 Lin_comparator.m

```
% inverting comparator with linear range
function [Vout]=lin_comparator(Vin)

global Vcmax Vcmin Vcb Gc

% Calculate output.
Vout=max(Vcmin, min(Vcmax, Vcb - Vin.*Gc));
```

9.11 Plot_volts.m

```
% Plot evolution of circuit voltages and optical input and output.

figure ('Name', 'Input Power (inv. wrt OP)')
plot(trange, Xrecord); % plot X/time for each neuron
grid
xlabel ('time');
ylabel ('input power, W');

figure ('Name', 'Preamp output')
plot(trange, Vprecord); % plot Vp/time for each neuron
grid
xlabel ('time');
ylabel ('preamp, V');

figure ('Name', 'High-pass output')
plot(trange, Vhpfrecord); % plot Vhpf/time for each neuron
grid
xlabel ('time');
ylabel ('high-pass, V');

figure ('Name', 'Comparator output (inv. wrt OP)')
plot(trange, Vcoutrecord); % plot Vcout/time for each neuron
grid
xlabel ('time');
ylabel ('comparator, V');

figure ('Name', 'Low-pass output (inv. wrt OP)')
plot(trange, Vlpfrecord); % plot Vlpf/time for each neuron
```

```
grid
xlabel ('time');
ylabel ('low-pass, V');

figure ('Name', 'Inverter output')
plot(trange, Vinvrecord); % plot Vinv/time for each neuron
grid
xlabel ('time');
ylabel ('inverter, V');

figure ('Name', 'Output Power')
plot(trange, Yrecord); % plot Y/time for each neuron
grid
xlabel ('time');
ylabel ('output power, W');
```

10 Appendix C

This appendix contains test results and code in Matlab 4.2.1c (Mac) for the lens system redesign.

10.1 Test Results

```
Lens Modelling Program V1.00
Written 1998 by Keith Symington

Using first 1 and second 1:
Warnings...
ERROR: System unsolvable.

Using first 1 and second 2:
Warnings...
ERROR: System unsolvable.

Using first 1 and second 3:
Warnings...
Optimal solution for f1=25, f2=80 is g=56.
Distance from VCSEL to LENS1: 19.047619 mm
Distance from LENS1 to DOE: 50.900000 mm
Distance from DOE to LENS2: 5.100000 mm
Distance from LENS2 to Image plane: 194.285714 mm
Total size of system: 269.333333 mm
Beam waist at LENS1: 5.163879 mm
Beam waist at LENS2: 7.628594 mm
Beam divergence of: 2.521339 degrees.
Beam waist at DOE: 7.404128 mm
Image size on image plane: -7.500000 mm
Calculations complete.
No errors encountered.
No warnings issued.

Using first 1 and second 4:
ERROR: System unsolvable.

Using first 1 and second 5:
Warnings...
ERROR: System unsolvable.

Using first 2 and second 1:
Warnings...
ERROR: System unsolvable.

Using first 2 and second 2:
Warnings...
ERROR: System unsolvable.

Using first 2 and second 3:
Warnings...
Optimal solution for f1=40, f2=80 is g=15.
Distance from VCSEL to LENS1: 29.841270 mm
Distance from LENS1 to DOE: 5.218750 mm
Distance from DOE to LENS2: 9.781250 mm
Distance from LENS2 to Image plane: 201.904762 mm
Total size of system: 246.746032 mm
Beam waist at LENS1: 6.673410 mm
Beam waist at LENS2: 6.268685 mm
Beam convergence of: 1.545839 degrees.
Beam waist at DOE: 6.532599 mm
Image size on image plane: -7.500000 mm
Calculations complete.
No errors encountered.
No warnings issued.

Using first 2 and second 4:
Warnings...
Optimal solution for f1=40, f2=150 is g=131.
Distance from VCSEL to LENS1: 29.830508 mm
Distance from LENS1 to DOE: 5.255556 mm
Distance from DOE to LENS2: 125.744444 mm
Distance from LENS2 to Image plane: 378.813559 mm
Total size of system: 539.644068 mm
Beam waist at LENS1: 6.671905 mm
```

```
Beam waist at LENS2: 14.517242 mm
Beam divergence of: 3.430308 degrees.
Beam waist at DOE: 6.986650 mm
Image size on image plane: -7.500000 mm
Calculations complete.
No errors encountered.
No warnings issued.

Using first 2 and second 5:
Warnings...
Optimal solution for f1=40, f2=190 is g=197.
Distance from VCSEL to LENS1: 29.898990 mm
Distance from LENS1 to DOE: 5.020000 mm
Distance from DOE to LENS2: 191.980000 mm
Distance from LENS2 to Image plane: 477.878788 mm
Total size of system: 704.777778 mm
Beam waist at LENS1: 6.681482 mm
Beam waist at LENS2: 20.951347 mm
Beam divergence of: 4.148456 degrees.
Beam waist at DOE: 7.045110 mm
Image size on image plane: -7.500000 mm
Calculations complete.
No errors encountered.
No warnings issued.

Using first 3 and second 1:
Warnings...
ERROR: System unsolvable.

Using first 3 and second 2:
Warnings...
ERROR: System unsolvable.

Using first 3 and second 3:
Warnings...
ERROR: System unsolvable.

Using first 3 and second 4:
Warnings...
ERROR: System unsolvable.

Using first 3 and second 5:
Warnings...
ERROR: System unsolvable.

Using first 4 and second 1:
Warnings...
ERROR: System unsolvable.

Using first 4 and second 2:
Warnings...
ERROR: System unsolvable.

Using first 4 and second 3:
Warnings...
ERROR: System unsolvable.

Using first 4 and second 4:
Warnings...
ERROR: System unsolvable.

Using first 4 and second 5:
Warnings...
ERROR: System unsolvable.

Using first 5 and second 1:
Warnings...
ERROR: System unsolvable.

Using first 5 and second 2:
Warnings...
ERROR: System unsolvable.

Using first 5 and second 3:
Warnings...
ERROR: System unsolvable.

Using first 5 and second 4:
Warnings...
ERROR: System unsolvable.

Using first 5 and second 5:
Warnings...
ERROR: System unsolvable.

The best combination is lens 2 first and lens 3 second with g at 15 mm.
Lens Model: Program terminated successfully.
»
```

10.2 Lens_Model.m

```
% Lens Model V1.00
% 1998 Keith Symington
%
% This script processes and executes analysis of optical distances
% based on lens focal lengths and the diffractive optic element
% working distance.

% Initialise variables.
Startup;

% Create a main output window.
figure(...
      'Name',        'Lens Modelling Program V1.0', ...
      'Color',       [0 0 0], ...
      'NumberTitle', 'off');
hold on;

% Local record of best system.
bestLens1=0;
bestLens2=0;
OptimalG=0;
bestDist=0;

% Iterate all systems.
for LENS_1=1:length(lensSet)
  for LENS_2=1:length(lensSet)
    % Set lens 1.
    f1=lensSet(1, LENS_1);
    d1=lensSet(2, LENS_1);
    f2=lensSet(1, LENS_2);
    d2=lensSet(2, LENS_2);
         disp(sprintf('Using first %d and second %d:', LENS_1, LENS_2));
    % Consider all posibilities in current lens system.
         [currentD, TempG]=Search(wsize, wbeam, verbose, f1, d1, f2, d2, L, d, M, Tsize, Tbeam);
         if (currentD>bestDist)
           bestDist=currentD;
           OptimalG=TempG;
           bestLens1=LENS_1;
           bestLens2=LENS_2;
         end;
  end;
end;

% Print the best.
disp(sprintf('The  best  combination  is  lens  %d  first  and  lens  %d  second  with  g  at  %d
mm.',bestLens1, bestLens2, OptimalG));

% Say bye.
disp('Lens Model: Program terminated successfully.');
```

10.3 Startup.m

```
% Startup module cleans up and sets some fixed startup parameters.
clear all;
pack;
format compact;
format short;
% Clear screen and print program name.
clc;
disp('Lens Modelling Program V1.00')
disp('Written 1998 by Keith Symington');
disp(' ');

% Global variables: (not normally available in functions).
lensSet=[25, 40, 80, 150, 190;10, 15, 25, 30, 50];
L=187;              % Working distance of DOE in mm.
d=1.5;              % Displacement of apparent object in mm.
M=-6;               % Magnification for entire system (image is inverted:
                    % image height over object height is negative)
Tsize=1000;         % Maximum system size in mm.
Tbeam=(pi/180)*10;  % Beam divergence/convergence tolerance in radians.
wsize=2;              % Weight multiplier which weights the input when calculating the optimal for
size.
wbeam=1;              % Weight multiplier which weights the input when calculating the optimal for
beam.
verbose=0;          % A value other than zero outputs information at every stage.
```

10.4 Search.m

```
% Compute performs all the calculations for the opticals model.
% It can be run either silently or with output.  The advantage of
% silent mode is that slower computers do not continually give output
% thus slowing things down.

function [bestdistance, OptimalG]=Search(wsize, wbeam, verbose, f1, d1, f2, d2, L, d, M, Tsize,
Tbeam)

% Setup parameters for return from compute.
searchdist=f1+f2;
results=zeros(1, searchdist);
bestdistance=0;
OptimalG=0;

% Search through all values of g.
for counter=1:searchdist
  % Compute value for current counter size.
  [thetadiv, totalSize, warnings, errors]=Compute(verbose, f1, d1, f2, d2, L, counter, d, M,
Tsize, Tbeam);
  % Calculate distance value for current variable.
  if ((errors+warnings)==0)
    results(1,                        counter)=((((Tsize-totalSize)/Tsize)*100)*wsize)^2+((((Tbeam-
thetadiv)/Tbeam)*100)*wbeam)^2;
  end;
  % If this is the best value so far then store it.
  if (results(1,counter)>bestdistance)
    bestdistance=results(1,counter);
    OptimalG=counter;
    end;
end;

% Output the lens combination statistics.
if (bestdistance>0)
  fprintf('\nOptimal solution for f1=%d, f2=%d is g=%d.\n', f1, f2, OptimalG);
  [thetadiv, totalSize, warnings, errors]=Compute(1, f1, d1, f2, d2, L, OptimalG, d, M, Tsize,
Tbeam);

  % Best solution measure graph.
  title(sprintf('Best solution with f1=%d and f2=%d at g=%d',f1, f2, OptimalG));
  xlabel('LENS1 to LENS2 separation in mm (g)');
  ylabel('Distance value');
  grid on;
  plot(results);
  pause(1);
else
  disp(' ');
  disp('ERROR: System unsolvable.');
  disp(' ');
end;
```

10.5 Compute.m

```
% Compute performs all the calculations for the opticals model.
% It can be run either silently or with output.  The advantage of
% silent mode is that slower computers do not continually give output
% thus slowing things down.

function [thetadiv, totalSize, warningFlag, errorFlag]=Compute(verbose, f1, d1, f2, d2, L, g, d,
M, Tsize, Tbeam)

% Parameter Check
% Check all set values for an error.
if (g>(f1+f2))
  disp('ERROR: Bounds check fail - g cannot exceed focal lengths of lenses 1 and 2: f1+f2 >= g');
  errorFlag=errorFlag+1;
  end;
% Check input values.
if (M>0)
  disp('WARNING: Magnification normally takes a negative value.');
  warningFlag=warningFlag+1;
  end;

% Fixed System variables: these are not normally altered.
Vsize=0.25;         % VCSEL size (square) in mm.
Vnx=8;              % Number of VCSELs in x direction.
Vny=6;              % Number of VCSELs in y direction.
theta=(pi/180)*8;   % Beam divergence from VCSELs in radians.
Tcomp=5;            % Minimum distance between components in mm.
Tf1=50;             % Maximum percentage by which the VCSEL->LENS1 distance can differ.
ddoe=22;            % Diameter of DOE in mm.
```

```
% Error logging.
errorFlag=0;
warningFlag=0;

% VCSEL array to lens 1.
u1=((((f1*f2)/M)+(f1*(g-f2)))/(g-f2-f1));
if (verbose) fprintf('Distance from VCSEL to LENS1: %f mm\n', u1); end;
if (0>=u1)
  if (verbose) disp('ERROR: Bounds check fail - u1 cannot have a negative focal length: u1 > 0');
end;
  errorFlag=errorFlag+1;
  end;
if (Tcomp>u1)
  if (verbose) disp('WARNING: VCSEL too close to LENS1: Tcomp > u1'); end;
  warningFlag=warningFlag+1;
  end;
if (u1>(f1*(1+(Tf1/100)))) | ((f1*(1-(Tf1/100)))>u1)
  if (verbose) disp('WARNING: VCSEL to LENS1 distance is not within tolerance to f1.'); end;
  warningFlag=warningFlag+1;
  end;

% Lens 2 to image plane.
u2=g-((f1*u1)/(u1-f1));
M2=f2/(u2-f2);
r=u2-(L/M2); % This is position of DOE.
if (verbose) fprintf('Distance from LENS1 to DOE: %f mm\n', (g-r)); end;
if (0>r)
  if (verbose) disp('ERROR: Bounds check fail - r cannot take a negative value: r >= 0'); end;
  errorFlag=errorFlag+1;
  end;
if (r>=g)
  if (verbose) disp('ERROR: Bounds check fail - DOE must lie between LENS1 and LENS2: g > r');
end;
  errorFlag=errorFlag+1;
  end;
if (Tcomp>(g-r))
  if (verbose) disp('WARNING: LENS1 too close to DOE: Tcomp > (g-r)'); end;
  warningFlag=warningFlag+1;
  end;
if (verbose) fprintf('Distance from DOE to LENS2: %f mm\n', r); end;
if (Tcomp>r)
  if (verbose) disp('WARNING: DOE too close to LENS2: Tcomp > r'); end;
  warningFlag=warningFlag+1;
  end;
v2=L/(1-(r/u2));
if (verbose) fprintf('Distance from LENS2 to Image plane: %f mm\n', v2); end;
if (0>v2)
  if (verbose) disp('ERROR: Bounds check fail - distance from image plane to LENS2 cannot be
negative: v2 >= 0'); end;
  errorFlag=errorFlag+1;
  end;
if (Tcomp>v2)
  if (verbose) disp('WARNING: LENS2 too close to Image plane: Tcomp > v2'); end;
  warningFlag=warningFlag+1;
  end;
totalSize=u1+g+v2;
if (verbose) fprintf('Total size of system: %f mm\n', totalSize); end;
if (totalSize>Tsize)
  if (verbose) disp('ERROR: Bounds check fail - system too large: u1+g+v2 > Tsize'); end;
  errorFlag=errorFlag+1;
  end;

% Beam waist at lens 1.
h1=sqrt((((Vnx*Vsize)^2)+((Vny*Vsize)^2)))/2;
p1=2*u1*tan((theta/2));
w1=2*((abs(p1)/2)+abs(h1));
if (verbose) fprintf('Beam waist at LENS1: %f mm\n', w1); end;
if (w1>d1)
  if (verbose) disp('ERROR: Bounds check fail - beam waist too large for LENS1: w1 > d1'); end;
  errorFlag=errorFlag+1;
  end;
if (d1>=w1) & (w1>=(d1*(0.9)))
  if (verbose) disp('WARNING: Beam waist w1 is within 10% of LENS1 diameter.'); end;
  warningFlag=warningFlag+1;
  end;

% Beam waist at lens 2.
h2=h1*((g-f1)/f1);
v1=1/((1/f1)-(1/u1));
p2=abs(((u2/v1)*p1));
w2=2*((abs(p2)/2)+(abs(h2)));
if (verbose) fprintf('Beam waist at LENS2: %f mm\n', w2); end;
if (w2>d2)
  if (verbose) disp('ERROR: Bounds check fail - beam waist too large for LENS2: w2 > d2'); end;
  errorFlag=errorFlag+1;
  end;
if (d2>=w2) & (w2>=(d2*(0.9)))
  if (verbose) disp('WARNING: Beam waist w2 is within 10% of LENS2 diameter.'); end;
```

```
  warningFlag=warningFlag+1;
  end;

% Beam convergence/divergence.
thetadiv=2*atan((w2-w1)/(2*g));
% Determine convergence/divergence of beam.
if (0>thetadiv)
  if (verbose) fprintf('Beam convergence of: %f degrees.\n', (abs((thetadiv*180)/pi))); end;
end;
if (thetadiv>0)
  if (verbose) fprintf('Beam divergence of: %f degrees.\n', (abs((thetadiv*180)/pi))); end;
end;
if (0==thetadiv)
  if (verbose) fprintf('Beam is collimated.\n'); end;
end;
if (abs(thetadiv)>Tbeam)
  if (verbose) disp('ERROR: Bounds check fail - divergence/convergence too great: |thetadiv| >
Tbeam'); end;
  errorFlag=errorFlag+1;
  end;

% Beam waist at DOE: Note that the conditions here exclude their being a focal point between
% LENS1 and LENS2 so we can therefore use simple trig to calculate the beam width.
wH=w1+2*tan(thetadiv/2)*(g-r);
if (verbose) fprintf('Beam waist at DOE: %f mm\n', wH); end;
if (wH>ddoe)
  if (verbose) disp('ERROR: Bounds check fail - beam waist too large for DOE: wH > ddoe'); end;
  errorFlag=errorFlag+1;
  end;
if (ddoe>=wH) & (wH>=(ddoe*(0.9)))
  if (verbose) disp('WARNING: Beam waist wH is within 10% of DOE diameter.'); end;
  warningFlag=warningFlag+1;
  end;

% Image size.
hI=(h2*M*f1)/(g-f1);
if (verbose) fprintf('Image size on image plane: %f mm\n', hI); end;

% Final checks to ensure system OK.
if (verbose)
  disp('Calculations complete.');
  % If there are no errors then say so.
  if (errorFlag==0) disp('No errors encountered.');
  else fprintf('%d error(s) present in system.\n', errorFlag);
  end;
  % If there are no warnings then say so.
  if (warningFlag==0) disp('No warnings issued.');
  else fprintf('%d warning(s) present in system.\n', warningFlag);
  end;
  disp(' ');
end;
```

11 Appendix D

Detailed information on minimum and maximum detector sensitivities.

Detector	V_p max (V)	V_p min (V)	I_{cc1} min (μA)	I_{cc1} min (μA)
0	3.80	0.80	0.90	4.00
1	4.00	0.70	0.50	4.00
2	3.80	1.00	0.70	4.00
3	4.00	0.80	0.50	3.80
4	4.00	0.80	0.40	4.10
5	3.90	0.50	0.50	4.00
6	3.90	0.70	0.30	3.80
7	3.80	0.50	0.30	4.00
8	4.10	0.80	0.40	4.00
9	4.10	0.80	0.60	4.00
10	4.00	0.80	0.60	4.10
11	4.10	0.80	0.50	3.80
12	4.10	0.90	0.50	3.80
13	4.10	0.80	0.60	4.00
14	4.10	0.80	0.40	3.80
15	4.10	0.80	0.30	3.90
16	4.10	0.90	0.30	3.80
17	4.10	0.80	0.40	3.80
18	4.20	0.80	0.40	3.90
19	4.10	0.80	0.50	4.00
20	4.10	0.90	0.30	3.60
21	4.10	0.80	0.40	4.00
22	4.10	0.80	0.30	4.00
23	4.10	0.90	0.40	3.80
24	4.10	0.80	0.40	4.00
25	4.10	0.80	0.50	3.80
26	4.10	0.80	0.40	4.00
27	4.10	0.80	0.50	4.00
28	4.10	0.80	0.40	3.90
29	4.10	4.10	0.05	0.48
30	4.10	0.80	0.40	4.00
31	4.10	0.80	0.40	4.20
32	4.10	0.80	0.40	4.10
33	4.20	0.90	0.30	4.20
34	4.10	0.80	0.50	3.90
35	4.20	4.20	0.05	0.48
36	4.10	0.90	0.60	4.00
37	4.20	0.90	0.50	4.20
38	4.20	0.90	0.30	3.80
39	4.20	0.90	0.50	3.90

40	4.30	0.90	0.40	3.90
41	4.30	0.90	0.30	4.20
42	4.20	0.90	0.40	4.20
43	4.20	0.90	0.90	4.10
44	4.30	1.00	0.50	4.20
45	4.20	0.90	0.50	4.20
46	4.20	0.90	0.40	4.10
47	4.20	1.00	0.50	4.10

Error ±	0.10	0.10	0.10	0.10

Minimum	3.80	0.50	0.30	3.60
Average	4.10	0.83	0.46	3.98
Maximum	4.30	1.00	0.90	4.20
St. Dev.	0.12	0.10	0.14	0.14

With Minimum Error (-0.1 from all values)				
Average	4.00	0.73	0.36	3.88
With Maximum Error (+0.1 on all values)				
Average	4.20	0.93	0.56	4.08

12 Appendix E

Detailed results for examination of diffractive optic element (DOE).

Channel No.	2
Laser No.	23
I_{CC1} Error (±A)	2.00E-09

Optical Power (W)	3.00E-04	Error (±W)	1.90E-05
Drive Current (A)	5.30E-03	Error (±A)	1.00E-04

Detector Current I_{CC1} (A)

8.600E-08	3.115E-06	1.260E-07	1.110E-07	9.600E-08	7.900E-08	1.040E-07	8.700E-08
3.060E-06	1.580E-07	1.985E-06	3.195E-06	3.020E-06	3.105E-06	2.875E-06	2.333E-06
9.500E-08	1.736E-06	1.070E-07	8.800E-08	5.300E-08	7.200E-08	6.300E-08	5.700E-08
5.300E-08	3.070E-06	4.800E-08	-	-	-	-	-
6.000E-08	3.332E-06	8.500E-08	-	-	-	-	-
4.900E-08	3.180E-06	8.400E-08	-	-	-	-	-

Normalised Against Centre I_{CC1}

0.54	19.72	0.80	0.70	0.61	0.50	0.66	0.55
19.37	1.00	12.56	20.22	19.11	19.65	18.20	14.77
0.60	10.99	0.68	0.56	0.34	0.46	0.40	0.36
0.34	19.43	0.30	-	-	-	-	-
0.38	21.09	0.54	-	-	-	-	-
0.31	20.13	0.53	-	-	-	-	-

Channel No.	8
Laser No.	9
I_{CC1} Error (±A)	2.00E-09

Optical Power (W)	3.00E-04	Error (±W)	2.10E-05
Drive Current (A)	5.50E-03	Error (±A)	1.00E-04

Detector Current I_{CC1} (A)

3.100E-06	2.763E-06	2.992E-06	2.910E-06	3.207E-06	2.942E-06	2.978E-06	1.550E-07
3.100E-08	6.400E-08	9.500E-08	6.300E-08	3.900E-08	3.900E-08	1.070E-07	3.057E-06
-	-	-	-	-	-	7.900E-08	3.016E-06
-	-	-	-	-	-	6.800E-08	3.258E-06
-	-	-	-	-	-	6.200E-08	3.154E-06
-	-	-	-	-	-	3.400E-08	3.729E-06

Normalised Against Centre I_{CC1}

20.00	17.83	19.30	18.77	20.69	18.98	19.21	1.00
0.20	0.41	0.61	0.41	0.25	0.25	0.69	19.72
-	-	-	-	-	-	0.51	19.46
-	-	-	-	-	-	0.44	21.02
-	-	-	-	-	-	0.40	20.35
-	-	-	-	-	-	0.22	24.06

Channel No.	15
Laser No.	20
I_{CC1} Error (±A)	2.00E-09

Optical Power (W)	3.00E-04	Error (±W)	2.80E-05
Drive Current (A)	5.10E-03	Error (±A)	1.00E-04

Detector Current I_{CC1} (A)

-	-	-	3.100E-08	3.047E-06	1.060E-07	-	-
1.160E-07	1.310E-07	8.100E-08	1.140E-07	3.058E-06	1.820E-07	1.120E-07	7.900E-08
2.906E-06	3.136E-06	2.982E-06	3.006E-06	1.660E-07	3.107E-06	3.120E-06	2.882E-06
6.000E-08	4.800E-08	4.200E-08	1.020E-07	2.973E-06	3.200E-08	6.800E-08	4.200E-08
-	-	-	3.000E-08	2.985E-06	4.600E-08	-	-
-	-	-	7.300E-08	3.227E-06	7.700E-08	-	-

Normalised Against Centre I_{CC1}

-	-	-	0.19	18.36	0.64	-	-
0.70	0.79	0.49	0.69	18.42	1.10	0.67	0.48
17.51	18.89	17.96	18.11	1.00	18.72	18.80	17.36
0.36	0.29	0.25	0.61	17.91	0.19	0.41	0.25
-	-	-	0.18	17.98	0.28	-	-
-	-	-	0.44	19.44	0.46	-	-

Channel No.	40
Laser No.	30
I_{CC1} Error (±A)	2.00E-09

Optical Power (W)	3.00E-04	Error (±W)	2.10E-05
Drive Current (A)	5.20E-03	Error (±A)	1.00E-04

Detector Current I_{CC1} (A)

-	3.500E-08	2.436E-06	9.900E-08	-	-	-	-
8.500E-08	1.010E-07	1.827E-06	1.790E-07	1.100E-07	8.500E-08	7.500E-08	1.010E-07
2.902E-06	2.735E-06	1.710E-07	3.048E-06	3.009E-06	2.893E-06	2.942E-06	2.453E-06
3.600E-08	9.400E-08	2.947E-06	9.500E-08	7.000E-08	1.900E-08	6.100E-08	5.300E-08
-	7.500E-08	2.930E-06	1.800E-08	-	-	-	-
-	6.300E-08	3.158E-06	7.300E-08	-	-	-	-

Normalised Against Centre I_{CC1}

-	0.20	14.25	0.58	-	-	-	-
0.50	0.59	10.68	1.05	0.64	0.50	0.44	0.59
16.97	15.99	1.00	17.82	17.60	16.92	17.20	14.35
0.21	0.55	17.23	0.56	0.41	0.11	0.36	0.31
-	0.44	17.13	0.11	-	-	-	-
-	0.37	18.47	0.43	-	-	-	-

Channel No.	39
Laser No.	56
I_{CC1} Error (±A)	2.00E-09

Optical Power (W)	3.00E-04	Error (±W)	3.40E-05
Drive Current (A)	5.10E-03	Error (±A)	1.00E-04

Detector Current I_{CC1} (A)

2.960E-06	3.200E-08	-	-	-	-	-	-
3.105E-06	8.600E-08	-	-	-	-	-	-
2.996E-06	7.400E-08	-	-	-	-	-	-
3.115E-06	1.080E-07	-	-	-	-	-	-
3.083E-06	1.950E-07	1.270E-07	4.500E-08	1.020E-07	1.090E-07	8.700E-08	5.300E-08
1.370E-07	3.183E-06	3.190E-06	2.986E-06	2.538E-06	1.787E-06	1.442E-06	1.852E-06

Normalised Against Centre I_{CC1}

21.61	0.23	-	-	-	-	-	-
22.66	0.63	-	-	-	-	-	-
21.87	0.54	-	-	-	-	-	-
22.74	0.79	-	-	-	-	-	-
22.50	1.42	0.93	0.33	0.74	0.80	0.64	0.39
1.00	23.23	23.28	21.80	18.53	13.04	10.53	13.52

Channel No.	27
Laser No.	49
I_{CC1} Error (±A)	2.00E-09

Optical Power (W)	3.00E-04	Error (±W)	2.50E-05
Drive Current (A)	5.10E-03	Error (±A)	1.00E-04

Detector Current I_{CC1} (A)

-	-	-	-	-	-	3.700E-08	2.369E-06
-	-	-	-	-	-	6.000E-08	2.516E-06
-	-	-	-	-	-	5.000E-08	2.478E-06
-	-	-	-	-	-	3.200E-08	2.574E-06
1.210E-07	9.600E-08	7.900E-08	5.200E-08	1.190E-07	8.700E-08	9.400E-08	2.576E-06
2.617E-06	2.474E-06	2.527E-06	2.213E-06	2.701E-06	2.535E-06	2.535E-06	1.320E-07

Normalised Against Centre I_{CC1}

-	-	-	-	-	-	0.28	17.95
-	-	-	-	-	-	0.45	19.06
-	-	-	-	-	-	0.38	18.77
-	-	-	-	-	-	0.24	19.50
0.92	0.73	0.60	0.39	0.90	0.66	0.71	19.52
19.83	18.74	19.14	16.77	20.46	19.20	19.20	1.00

Cross Analysis of Normalised Values I_{CC1} in X

x-7	x-6	x-5	x-4	x-3	x-2	x-1	x	x+1	x+2	x+3	x+4	x+5	x+6	x+7	Channel
-	-	-	-	-	-	19.37	1.00	12.56	20.22	19.11	19.65	18.20	14.77	-	2
20.00	17.83	19.30	18.77	20.69	18.98	19.21	1.00	-	-	-	-	-	-	-	8
-	-	-	17.51	18.89	17.96	18.11	1.00	18.72	18.80	17.36	-	-	-	-	15
19.83	18.74	19.14	16.77	20.46	19.20	19.20	1.00	-	-	-	-	-	-	-	27
-	-	-	-	-	-	-	1.00	23.23	23.28	21.80	18.53	13.04	10.53	13.52	39
-	-	-	-	-	16.97	15.99	1.00	17.82	17.60	16.92	17.20	14.35	-	-	40
Average value															
19.913	18.284	19.224	17.682	20.015	18.280	18.377	1.000	18.085	19.974	18.797	18.461	15.195	12.646	13.518	All

Cross Analysis of Normalised Values I_{CC1} in Y

y-5	y-4	y-3	y-2	y-1	y	y+1	y+2	y+3	y+4	y+5	Channel
-	20.13	21.09	19.43	10.99	1.00	19.72	-	-	-	-	2
24.06	20.35	21.02	19.46	19.72	1.00	-	-	-	-	-	8
-	-	19.44	17.98	17.91	1.00	18.42	18.36	-	-	-	15
-	-	-	-	-	1.00	19.52	19.50	18.77	19.06	17.95	27
-	-	-	-	-	1.00	22.50	22.74	21.87	22.66	21.61	39
-	-	18.47	17.13	17.23	1.00	10.68	14.25	-	-	-	40
Average value											
24.058	20.237	20.004	18.501	16.463	1.000	18.168	18.710	20.321	20.862	19.776	All

13 Appendix F

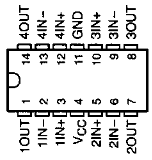
Data sheets for amplifier used in amp-board and neural switch card.

LM124, LM124A, LM224, LM224A
LM324, LM324A, LM324Y, LM2902, LM2902Q
QUADRUPLE OPERATIONAL AMPLIFIERS

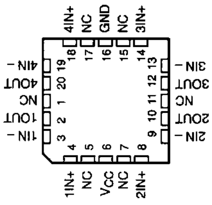
SLOS066D – SEPTEMBER 1975 – REVISED SEPTEMBER 1996

- **Wide Range of Supply Voltages:** Single Supply . . . 3 V to 30 V (LM2902 and LM2902Q 3 V to 26 V), or Dual Supplies
- **Low Supply Current Drain Independent of Supply Voltage . . . 0.8 mA Typ**
- **Common-Mode Input Voltage Range Includes Ground Allowing Direct Sensing Near Ground**
- **Low Input Bias and Offset Parameters:** Input Offset Voltage . . . 3 mV Typ; A Versions . . . 2 mV Typ; Input Offset Current . . . 2 nA Typ; Input Bias Current . . . 20 nA Typ; A Versions . . . 15 nA Typ
- **Differential Input Voltage Range Equal to Maximum-Rated Supply Voltage . . . 32 V (26 V for LM2902 and LM2902Q)**
- **Open-Loop Differential Voltage Amplification . . . 100 V/mV Typ**
- **Internal Frequency Compensation**

LM124, LM224A . . . J OR W PACKAGE
ALL OTHERS . . . D, DB, J, N OR PW PACKAGE
(TOP VIEW)

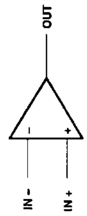


LM124, LM124A . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

symbol (each amplifier)



description

These devices consist of four independent high-gain frequency-compensated operational amplifiers that are designed specifically to operate from a single supply over a wide range of voltages. Operation from split supplies is also possible when the difference between the two supplies is 3 V to 30 V (for the LM2902 and LM2902Q, 3 V to 26 V) and V_{CC} is at least 1.5 V more positive than the input common-mode voltage. The low supply current drain is independent of the magnitude of the supply voltage.

Applications include transducer amplifiers, dc amplification blocks, and all the conventional operational amplifier circuits that now can be more easily implemented in single-supply-voltage systems. For example, the LM124 can be operated directly from the standard 5-V supply that is used in digital systems and easily provides the required interface electronics without requiring additional ±15-V supplies.

The LM2902Q is manufactured to demanding automotive requirements.

The LM124 and LM124A are characterized for operation over the full military temperature range of −55°C to 125°C. The LM224 and LM224A are characterized for operation from −25°C to 85°C. The LM324 and LM324A are characterized for operation from 0°C to 70°C. The LM2902 and LM2902Q are characterized for operation from −40°C to 125°C.

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

1

LM124, LM124A, LM224, LM224A
LM324, LM324A, LM324Y, LM2902, LM2902Q
QUADRUPLE OPERATIONAL AMPLIFIERS

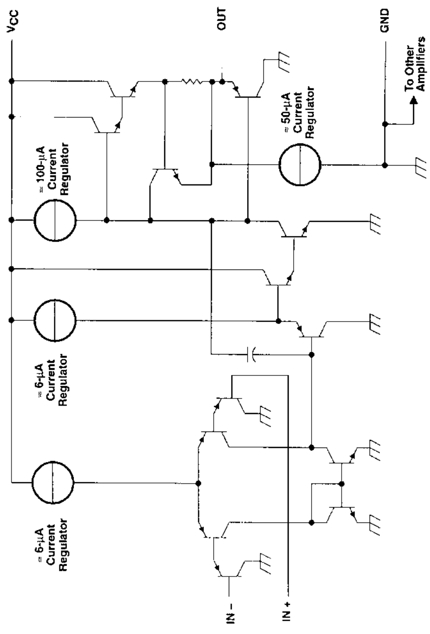
SLOS066D – SEPTEMBER 1975 – REVISED SEPTEMBER 1996

AVAILABLE OPTIONS

T_A	V_{IO}max AT 25°C	PACKAGED DEVICES: SMALL OUTLINE (D)†	VERY SMALL OUTLINE (DB)‡	CHIP CARRIER (FK)	CERAMIC DIP (J)	PLASTIC DIP (N)	TSSOP (PW)‡	FLAT PACK (W)	CHIP FORM (Y)
0°C to 70°C	7 mV	LM324D	LM324DBLE	—	—	LM324N	LM324PWLE	—	LM324Y
	3 mV	LM324AD	—	—	—	LM324AN	LM324APWLE	—	
−25°C to 85°C	5 mV	LM224D	—	—	—	LM224N	—	—	—
	3 mV	LM224AD	—	—	—	LM224AN	—	—	
−40°C to 125°C	7 mV	LM2902D	LM2902DBLE	—	—	LM2902N	LM2902PWLE	—	—
		LM2902QD		—	—	LM2902QN		—	
−55°C to 125°C	5 mV	—	—	LM124FK	LM124J	—	—	LM124W	—
	2 mV	—	—	LM124AFK	LM124AJ	—	—		

† The D package is available taped and reeled. Add the suffix R to the device type (e.g., LM324DR).
‡ The DB and PW packages are only available left-end taped and reeled.

schematic (each amplifier)



COMPONENT COUNT (total device)	
Epi-FET	1
Transistors	95
Diodes	4
Resistors	11
Capacitors	4

2

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

		LM124, LM124A LM224, LM224A LM324, LM324A	LM2902, LM2902Q	UNIT
Supply voltage, V_{CC} (see Note 1)		32	26	V
Differential input voltage, V_{ID} (see Note 2)		±32	±26	V
Input voltage, V_I (either input)		−0.3 to 32	−0.3 to 26	V
Duration of output short circuit (one amplifier) to ground at (or below) T_A = 25°C, $V_{CC} \leq$ 15 V (see Note 3)		unlimited	unlimited	
Continuous total dissipation		See Dissipation Rating Table		
Operating free-air temperature range, T_A	LM124, LM124A	−55 to 125		°C
	LM224, LM224A	−25 to 85		
	LM324, LM324A	0 to 70		
	LM2902, LM2902Q		−40 to 125	
Storage temperature range		−65 to 150	−65 to 150	°C
Case temperature for 60 seconds	FK package	260		°C
Lead temperature 1,6 mm (1/16 inch) from case for 60 seconds	J or W package	300	300	°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds	D, DB, N, or PW package	260	260	°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 1. All voltage values (except differential voltages and V_{CC} specified for the measurement of I_{OS}) are with respect to the network GND.
2. Differential voltages are at IN + with respect to IN −.
3. Short circuits from outputs to V_{CC} can cause excessive heating and eventual destruction.

DISSIPATION RATING TABLE

PACKAGE	$T_A \leq$ 25°C POWER RATING	DERATING FACTOR	DERATE ABOVE T_A	T_A = 70°C POWER RATING	T_A = 85°C POWER RATING	T_A = 125°C POWER RATING
D	900 mW	7.6 mW/°C	32°C	611 mW	497 mW	N/A
DB	775 mW	6.2 mW/°C	25°C	496 mW	403 mW	N/A
FK	900 mW	11.0 mW/°C	68°C	878 mW	713 mW	273 mW
J (LM124_)	900 mW	11.0 mW/°C	68°C	878 mW	713 mW	273 mW
J (all others)	900 mW	8.2 mW/°C	40°C	654 mW	531 mW	N/A
N	900 mW	9.2 mW/°C	52°C	734 mW	596 mW	N/A
PW	700 mW	5.6 mW/°C	25°C	448 mW	364 mW	N/A
W	900 mW	8.0 mW/°C	37°C	636 mW	516 mW	196 mW

LM124, LM124A, LM224, LM224A
LM324, LM324A, LM324Y, LM2902, LM2902Q
QUADRUPLE OPERATIONAL AMPLIFIER
SLOS066D – SEPTEMBER 1975 – REVISED SEPTEMBER 1996

electrical characteristics at specified free-air temperature, V_{CC} = 5 V (unless otherwise noted)

PARAMETER		TEST CONDITIONS†	T_A‡	LM124, LM224 MIN	TYP§	MAX	LM324 MIN	TYP§	MAX	LM2902, LM2902Q MIN	TYP§	MAX	UNIT
V_{IO}	Input offset voltage	V_{CC} = 5 V to MAX, $V_{IC} = V_{ICR}$min, V_O = 1.4 V	25°C		3	5		3	7		3	7	mV
			Full range			7			9			10	
I_{IO}	Input offset current	V_O = 1.4 V	25°C		2	30		2	50		2	50	nA
			Full range			100			150			300	
I_{IB}	Input bias current	V_O = 1.4 V	25°C		−20	−150		−20	−250		−20	−250	nA
			Full range			−300			−500			−500	
V_{ICR}	Common-mode input voltage range	V_{CC} = 5 V to MAX	25°C	0 to V_{CC}−1.5			0 to V_{CC}−1.5			0 to V_{CC}−1.5			V
			Full range	0 to V_{CC}−2			0 to V_{CC}−2			0 to V_{CC}−2			
V_{OH}	High-level output voltage	R_L = 2 kΩ	25°C	V_{CC}−1.5			V_{CC}−1.5						V
		R_L = 10 kΩ	25°C							V_{CC}−1.5			
		V_{CC} = MAX, R_L = 2 kΩ	Full range	26			26			22			
		V_{CC} = MAX, $R_L \geq$ 10 kΩ	Full range	27	28		27	28		23	24		
V_{OL}	Low-level output voltage	$R_L \leq$ 10 kΩ	Full range		5	20		5	20		5	100	mV
A_{VD}	Large-signal differential voltage amplification	V_{CC} = 15 V, V_O = 1 V to 11 V, $R_L \geq$ 2 kΩ	25°C	50	100		25	100			100		V/mV
			Full range	25			15			15			
CMRR	Common-mode rejection ratio	$V_{IC} = V_{ICR}$min	25°C	70	80		65	80		50	80		dB
k_{SVR}	Supply-voltage rejection ratio ($\Delta V_{CC}/\Delta V_{IO}$)		25°C	65	100		65	100		50	100		dB
V_{O1}/V_{O2}	Crosstalk attenuation	f = 1 kHz to 20 kHz	25°C		120			120			120		dB
I_O	Output current	V_{CC} = 15 V, V_{ID} = 1 V, V_O = 0	25°C	−20	−30	−60	−20	−30	−60	−20	−30	−60	mA
			Full range	−10			−10			−10			
		V_{CC} = 15 V, V_{ID} = −1 V, V_O = 15 V	25°C	10	20		10	20		10	20		
			Full range	5			5			5			
		V_{ID} = −1 V, V_O = 200 mV	25°C	12	30		12	30			30		μA
I_{OS}	Short-circuit output current	V_{CC} at 5 V, GND at −5 V, V_O = 0	25°C		±40	±60		±40	±60		±40	±60	mA
I_{CC}	Supply current (four amplifiers)	V_O = 2.5 V, No load	Full range		0.7	1.2		0.7	1.2		0.7	1.2	mA
		V_{CC} = MAX, V_O = 0.5 V_{CC}, No load	Full range		1.4	3		1.4	3		1.4	3	

† All characteristics are measured under open-loop conditions with zero common-mode input voltage unless otherwise specified. MAX V_{CC} for testing purposes is 26 V for LM2902 and LM2902Q, 30 V for the others.
‡ Full range is −55°C to 125°C for LM124, −25°C to 85°C for LM224, 0°C to 70°C for LM324, and −40°C to 125°C for LM2902 and LM2902Q.
§ All typical values are at T_A = 25°C.

Template Release Date: 7–11–94

LM124, LM124A, LM224, LM224A
LM324, LM324A, LM324Y, LM2902, LM2902Q
QUADRUPLE OPERATIONAL AMPLIFIERS
SLOS066D – SEPTEMBER 1975 – REVISED SEPTEMBER 1996

electrical characteristics at specified free-air temperature, V_{CC} = 5 V (unless otherwise noted)

PARAMETER		TEST CONDITIONS†	T_A‡	LM124A			LM224A			LM324A			UNIT
				MIN	TYP§	MAX	MIN	TYP§	MAX	MIN	TYP§	MAX	
V_{IO}	Input offset voltage	V_{CC} = 5 V to 30 V, $V_{IC} = V_{ICR}$min, V_O = 1.4 V	25°C			2		2	3		2	3	mV
			Full range			4			4			5	
I_{IO}	Input offset current	V_O = 1.4 V	25°C			10			2	15	2	30	nA
			Full range			30			30			75	
I_{IB}	Input bias current	V_O = 1.4 V	25°C			−50		−15	−80		−15	−100	nA
			Full range			−100			−100			−200	
V_{ICR}	Common-mode input voltage range	V_{CC} = 30 V	25°C	0 to $V_{CC}-1.5$			0 to $V_{CC}-1.5$			0 to $V_{CC}-1.5$			V
			Full range	0 to $V_{CC}-2$			0 to $V_{CC}-2$			0 to $V_{CC}-2$			
V_{OH}	High-level output voltage	R_L = 2 kΩ	25°C	$V_{CC}-1.5$			$V_{CC}-1.5$			$V_{CC}-1.5$			V
		V_{CC} = 30 V, R_L = 2 kΩ	Full range	26			26			26			
		V_{CC} = 30 V, R_L ≥ 10 kΩ	Full range	27			27	28		27	28		
V_{OL}	Low-level output voltage	R_L ≤ 10 kΩ	Full range			20		5	20		5	20	mV
A_{VD}	Large-signal differential voltage amplification	V_{CC} = 15 V, V_O = 1 V to 11 V, R_L ≥ 2 kΩ	Full range	25			25			15			V/mV
CMRR	Common-mode rejection ratio	$V_{IC} = V_{ICR}$min	25°C	70			70	80		65	80		dB
k_{SVR}	Supply-voltage rejection ratio ($\Delta V_{CC}/\Delta V_{IO}$)		25°C	65			65	100		65	100		dB
V_{O1}/V_{O2}	Crosstalk attenuation	f = 1 kHz to 20 kHz	25°C		120			120			120		dB
I_O	Output current	V_{CC} = 15 V, V_{ID} = 1 V, V_O = 0	25°C	−20			−20	−30	−60	−20	−30	−60	mA
			Full range	−10			−10			−10			
		V_{CC} = 15 V, V_{ID} = −1 V, V_O = 15 V	25°C	10			10	20		10	20		
			Full range	5			5			5			
		V_{ID} = −1 V, V_O = 200 mV	25°C	12			12	30		12	30		μA
I_{OS}	Short-circuit output current	V_{CC} at 5 V, GND at −5 V, V_O = 0	25°C		±40	±60		±40	±60		±40	±60	mA
I_{CC}	Supply current (four amplifiers)	V_O = 2.5 V, No load	Full range		0.7	1.2		0.7	1.2		0.7	1.2	mA
		V_{CC} = 30 V, V_O = 15 V, No load	Full range		1.4	3		1.4	3		1.4	3	

† All characteristics are measured under open-loop conditions with zero common-mode input voltage unless otherwise specified.
‡ Full range is −55°C to 125°C for LM124A, −25°C to 85°C for LM224A, and 0°C to 70°C for LM324A.
§ All typical values are at T_A = 25°C.

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

6

LM124, LM124A, LM224, LM224A
LM324, LM324A, LM324Y, LM2902, LM2902Q
QUADRUPLE OPERATIONAL AMPLIFIERS
SLOS066D – SEPTEMBER 1975 – REVISED SEPTEMBER 1996

electrical characteristics, V_{CC} = 5 V, T_A = 25°C (unless otherwise noted)

PARAMETER		TEST CONDITIONS†	LM324Y			UNIT
			MIN	TYP	MAX	
V_{IO}	Input offset voltage	V_{CC} = 5 V to MAX, $V_{IC} = V_{ICR}$min, V_O = 1.4 V		3	7	mV
I_{IO}	Input offset current			2	50	nA
I_{IB}	Input bias current			−20	−250	nA
V_{ICR}	Common-mode input voltage range	V_{CC} = 5 V to MAX	0 to $V_{CC}-1.5$			V
V_{OH}	High-level output voltage	R_L = 10 kΩ	$V_{CC}-1.5$			V
V_{OL}	Low-level output voltage	R_L ≤ 10 kΩ		5	20	mV
A_{VD}	Large-signal differential voltage amplification	V_{CC} = 15 V, V_O = 1 V to 11 V, R_L ≥ 2 kΩ	15	100		V/mV
CMRR	Common-mode rejection ratio	$V_{IC} = V_{ICR}$min	65	80		dB
k_{SVR}	Supply-voltage rejection ratio ($\Delta V_{CC\pm}/\Delta V_{IO}$)		65	100		dB
I_O	Output current	V_{CC} = 15 V, V_{ID} = 1 V, V_O = 0	−20	−30	−60	mA
		V_{CC} = 15 V, V_{ID} = −1 V, V_O = 15 V	10	20		
		V_{ID} = 1 V, V_O = 200 mV	12	30		
I_{OS}	Short-circuit output current	V_{CC} at 5 V, GND at −5 V, V_O = 0		±40	±60	mA
I_{CC}	Supply current (four amplifiers)	V_O = 2.5 V_{CC}, No load		0.7	1.2	mA
		V_{CC} = MAX, V_O = 0.5 V_{CC}, No load		1.1	3	

† All characteristics are measured under open-loop conditions with zero common-mode input voltage unless otherwise specified. MAX V_{CC} for testing purposes is 30 V.

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

7

14 Appendix G

		P_{req} at 50µW				P_{req} at 800µW			
Channel	VCSEL	P_L (µW)	P_U (µW)	I_U (mA)	I_{req} (mA)	P_L (µW)	P_U (µW)	I_U (mA)	I_{req} (mA)
0	21	43	81	4.8	4.379	755	807	9.0	8.475
1	24	28	65	4.6	4.268	753	806	9.2	8.668
2	23	20	60	4.2	3.919	772	823	8.0	7.470
3	16	15	54	4.0	3.758	770	822	7.2	6.720
4	29	25	68	4.2	3.888	777	827	8.0	7.454
5	15	26	74	4.4	4.061	797	849	9.2	8.511
6	22	25	75	4.2	3.872	779	828	6.8	6.314
7	14	35	82	4.6	4.216	766	826	8.2	7.663
8	9	38	87	4.6	4.202	775	817	8.6	8.046
9	12	17	61	3.8	3.542	753	810	6.6	6.200
10	13	41	78	5.0	4.579	789	832	10.2	9.493
11	20	34	91	4.4	4.020	754	801	7.2	6.796
12	11	31	70	4.2	3.870	760	807	8.4	7.905
13	19	19	66	4.0	3.713	788	835	7.4	6.848
14	10	18	65	4.2	3.906	780	813	7.0	6.537
15	28	20	63	4.2	3.910	794	837	7.8	7.204
16	35	16	57	4.4	4.123	792	835	8.6	7.968
17	18	21	76	4.0	3.688	772	835	6.2	5.751
18	17	26	73	4.4	4.063	791	842	9.4	8.722
19	27	25	74	4.2	3.874	791	833	7.8	7.218
20	26	33	82	4.2	3.843	769	815	7.8	7.305
21	25	37	81	4.6	4.211	764	812	9.0	8.453
22	33	45	107	4.4	3.982	763	819	6.8	6.358
23	34	16	51	4.2	3.961	787	836	7.8	7.228
24	44	-	50	3.8	3.589	772	824	6.8	6.335
25	41	25	74	4.4	4.063	773	816	8.2	7.674
26	42	13	51	4.0	3.773	754	805	6.4	6.026
27	49	20	60	4.2	3.919	788	829	8.2	7.611
28	36	15	55	4.2	3.943	-	800	8.0	7.556
29	50	19	62	4.4	4.103	779	821	7.4	6.894
30	43	26	79	4.2	3.863	754	803	6.4	6.033
31	51	18	51	4.2	3.961	755	811	8.4	7.896
32	46	18	59	4.0	3.736	753	804	7.6	7.163
33	52	35	94	4.0	3.637	744	802	6.2	5.849
34	53	18	53	4.2	3.950	763	814	7.6	7.126
35	45	45	72	5.4	4.946	793	832	11.0	10.234
36	54	37	80	4.6	4.213	776	825	9.4	8.781
37	55	23	54	4.6	4.320	786	837	9.0	8.363
38	37	-	50	4.6	4.344	795	846	9.2	8.519
39	56	24	73	4.4	4.067	791	829	7.2	6.656
40	30	17	55	4.2	3.942	776	826	7.8	7.268
41	47	-	50	3.8	3.589	799	844	7.2	6.615
42	48	27	56	4.8	4.494	773	818	9.4	8.802
43	38	30	87	4.0	3.655	761	811	6.8	6.381
44	39	16	55	4.2	3.942	745	812	6.4	6.011
45	40	31	80	4.0	3.662	756	801	7.8	7.362
46	32	46	103	4.4	3.980	795	843	7.4	6.820
47	31	45	101	4.2	3.795	788	830	7.2	6.665

15 Appendix H

If a channel was not working properly, no measurements were made.

Detector	Total Photocurrent (μA)	Detectors Used	μA/Detector
0	1.93	-	-
1	24.33	12	1.865
2	18.81	12	1.405
3	19.43	11	1.589
4	18.69	12	1.395
5	17.60	11	1.423
6	22.43	12	1.707
7	25.04	12	1.924
8	17.79	12	1.320
9	23.47	12	1.793
10	19.00	12	1.421
11	1.92	-	-
12	20.78	12	1.569
13	17.58	11	1.421
14	1.92	-	-
15	21.12	12	1.598
16	19.48	12	1.461
17	1.92	-	-
18	21.98	12	1.669
19	24.48	11	2.048
20	24.07	12	1.843
21	1.91	-	-
22	20.90	12	1.579
23	19.80	12	1.488
24	17.46	11	1.410
25	14.81	11	1.169
26	4.93	11	0.271
27	1.91	-	-
28	21.27	11	1.756
29	1.92	-	-
30	17.81	11	1.442
31	1.92	-	-
32	17.03	11	1.371
33	19.57	11	1.602
34	20.19	11	1.658
35	17.41	-	-
36	1.93	-	-
37	13.07	10	1.112
38	19.89	11	1.631
39	20.44	11	1.681
40	1.91	-	-
41	1.91	-	-
42	33.26	12	2.609
43	1.91	-	-
44	17.52	12	1.298
45	20.63	11	1.698
46	19.68	12	1.478
47	20.62	12	1.556

16 Appendix I

This section contains further results from system testing.

The next test sequence gave a valid result without optimisation of V_{ref} from equation 33.

$$\begin{array}{c}\text{Request}\\ \begin{array}{r}0\\8\\16\\24\\32\\40\end{array}\begin{bmatrix}0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&0\end{bmatrix}\end{array}\Rightarrow\begin{array}{c}\text{Response}\\ \begin{array}{r}0\\8\\16\\24\\32\\40\end{array}\begin{bmatrix}0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\end{bmatrix}\end{array}$$

$$V_{ref} = 0.78V$$

Equation 35

Unfortunately, V_{ref} had to be adjusted to 0.62V before a solution was found for the next request matrix.

$$\begin{array}{c}\text{Request}\\ \begin{array}{r}0\\8\\16\\24\\32\\40\end{array}\begin{bmatrix}0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&1\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&1\\0&0&0&0&0&0&0&1\end{bmatrix}\end{array}\Rightarrow\begin{array}{c}\text{Response}\\ \begin{array}{r}0\\8\\16\\24\\32\\40\end{array}\begin{bmatrix}0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&1\end{bmatrix}\end{array}$$

$$V_{ref} = 0.62V$$

Equation 36

V_{ref} unadjusted at 0.62V.

$$\begin{array}{c}\text{Request}\\ \begin{array}{r}0\\8\\16\\24\\32\\40\end{array}\begin{bmatrix}0&0&0&1&0&0&0&0\\0&0&0&0&1&0&0&0\\0&0&1&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&1&0&0&0&0&0\\0&0&0&0&0&0&0&0\end{bmatrix}\end{array}\Rightarrow\begin{array}{c}\text{Response}\\ \begin{array}{r}0\\8\\16\\24\\32\\40\end{array}\begin{bmatrix}0&0&0&1&0&0&0&0\\0&0&0&0&1&0&0&0\\0&0&1&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\end{bmatrix}\end{array}$$

$$V_{ref} = 0.62V$$

Equation 37

V_{ref} again worked at 0.62V.

$$\begin{array}{c} \\ 0\\8\\16\\24\\32\\40 \end{array}\overset{\text{Request}}{\begin{bmatrix} 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&1&1\\ 0&0&0&0&0&1&0&1 \end{bmatrix}} \Rightarrow \begin{array}{c} \\ 0\\8\\16\\24\\32\\40 \end{array}\overset{\text{Response}}{\begin{bmatrix} 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&1&0\\ 0&0&0&0&0&1&0&0 \end{bmatrix}}$$

$$V_{ref} = 0.62V$$

Equation 38

Still with V_{ref} at 0.62V...

$$\begin{array}{c} \\ 0\\8\\16\\24\\32\\40 \end{array}\overset{\text{Request}}{\begin{bmatrix} 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&1&1\\ 0&0&0&0&0&1&0&1 \end{bmatrix}} \Rightarrow \begin{array}{c} \\ 0\\8\\16\\24\\32\\40 \end{array}\overset{\text{Response}}{\begin{bmatrix} 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0\\ 0&0&0&0&0&1&0&0 \end{bmatrix}}$$

$$V_{ref} = 0.62V$$

Equation 39

V_{ref} had to be increased to 0.66V before the network's output became valid.

$$\begin{array}{c} \\ 0\\8\\16\\24\\32\\40 \end{array}\overset{\text{Request}}{\begin{bmatrix} 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&1&1\\ 0&0&0&0&0&1&0&1 \end{bmatrix}} \Rightarrow \begin{array}{c} \\ 0\\8\\16\\24\\32\\40 \end{array}\overset{\text{Response}}{\begin{bmatrix} 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&1&0\\ 0&0&0&0&0&1&0&0 \end{bmatrix}}$$

$$V_{ref} = 0.66V$$

Equation 40

Careful adjustment of V_{ref} to 0.79V gave a good solution. The induced photocurrent was 1.89 to 1.90µA.

$$\begin{array}{c} \\ 0\\8\\16\\24\\32\\40 \end{array}\overset{\text{Request}}{\begin{bmatrix} 0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&1&1\\ 0&0&0&0&0&1&0&1 \end{bmatrix}} \Rightarrow \begin{array}{c} \\ 0\\8\\16\\24\\32\\40 \end{array}\overset{\text{Response}}{\begin{bmatrix} 0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&1\\ 0&0&0&0&0&1&0&0 \end{bmatrix}}$$

$$V_{ref} = 0.79V$$

Equation 41

Leaving V_{ref} at 0.79V, a valid result was still received.

$$\begin{array}{cc} \text{Request} & \text{Response} \\ \begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \Rightarrow & \begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \end{array}$$

$V_{ref} = 0.79V$

Equation 42

Fine adjustment of V_{ref} to 0.79V gave a valid result.

$$\begin{array}{cc} \text{Request} & \text{Response} \\ \begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array}\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \Rightarrow & \begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array}\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \end{array}$$

$V_{ref} = 0.79V$

Equation 43

Adjustment of the above by the addition of one other request (detector 38) resulted in an invalid solution. It was possible to adjust V_{ref} to give a valid solution; however, the selection of neurons which remained on proved very unstable. Examination of the system indicated that detector saturation could be causing a problem, thus photographic film was inserted which absorbed ~33% of the optical power throughput. This resulted in a stable solution after slight adjustment of V_{ref}:

$$\begin{array}{cc} \text{Request} & \text{Response} \\ \begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array}\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \Rightarrow & \begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array}\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \end{array}$$

$V_{ref} = 0.76V$

Equation 44

Without adjustment, another valid solution is shown overleaf.

$$
\begin{array}{c}
\text{Request} \\
\begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array}
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1
\end{bmatrix}
\end{array}
\Rightarrow
\begin{array}{c}
\text{Response} \\
\begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array}
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0
\end{bmatrix}
\end{array}
$$

$$V_{ref} = 0.76V$$

Equation 45

All went well until fewer neurons were requested.

$$
\begin{array}{c}
\text{Request} \\
\begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array}
\begin{bmatrix}
0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\end{array}
\Rightarrow
\begin{array}{c}
\text{Response} \\
\begin{array}{r} 0 \\ 8 \\ 16 \\ 24 \\ 32 \\ 40 \end{array}
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\end{array}
$$

$$V_{ref} = 0.76V$$

Equation 46

This solution proved wrong at V_{ref}=0.76V, the reasons for which are explained in section 4.7. Adjustment did reveal that this system could be solved at this power level but V_{ref} needed to be 0.91V before it gave a valid solution: a value at which all previous tests did not work.