This is to certify that the

thesis entitled

THE DEVELOPMENT OF AN ARRAY
DETECTOR SPECTROPHOTOMETER

presented by

PETER JOSEPH AIELLO

has been accepted towards fulfillment
of the requirements for

M.S. degree in Chemistry

C. G. Enke
Major professor

Date Apr 8, 1980

THE DEVELOPMENT OF
AN ARRAY DETECTOR SPECTROPHOTOMETER

By

Peter Joseph Aiello

A THESIS

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

MASTER OF SCIENCE

Department of Chemistry

1980

ABSTRACT

THE DEVELOPMENT OF AN ARRAY DETECTOR SPECTROPHOTOMETER

By

Peter Joseph Aiello

A microcomputer controlled linear diode array spectrophotometer has been developed. The linear diode array used has 512 individual light sensitive diodes each of which is sensitive to electromagnetic radiation with wavelengths from 200-1000 nm. A polychromator was designed using a concave holographic grating which helps reduce stray light levels and produces a flat field image (400 nm) across the light sensitive area of the detector. The detector can be moved in the focal plane of the grating with no degradation of resolution. The inlet system optics were designed for maximum light intensity at all wavelengths.

The linear diode array (LDA) is self-scanned to provide real-time, computer-compatible, serial electronic output for all 512 channels in succession. The microprocessor, an Intel 8085A, is used to collect and store the data on a floppy disk. The microcomputer also controls the detector integration time. By increasing the integration time, weak

signals can be enhanced but there is a loss of dynamic range as the dark current can approach saturation at room temperature in a time as short as 2 sec. Charge integration can be used to optimize the output of the detector while avoiding saturation. Resolution remains constant at all integration times as there is no evidence of interchannel cross-talk (blooming).

The instrument has been applied to molecular absorbance spectroscopy.

To Patsy

ACKNOWLEDGMENTS

My sincere appreciation goes to Professor Chris Enke for the help and encouragement that he has given me during the course of my research here at Michigan State University. I would also like to thank Professor Stan Crouch for many invaluable discussions.

I would like to thank Michigan State University, Abbott Laboratories and the Office of Naval Research for the financial support they provided.

I would like to thank the members of Professor Enke's research group for their help and friendship, especially Hugh Gregg and Bruce Newcome.

Finally, I offer my thanks and appreciation to my wife, Pat, for her continuous support and love.

TABLE OF CONTENTS

Chapter Page

LIST OF TABLES

LIST OF FIGURES

Figure | Page

Figure		Page

Figure		Page

CHAPTER I

INTRODUCTION

Traditionally, the technique for the measurement of electromagnetic radiation in the UV-Vis region involves a dispersive element such as a grating or prism and a detector system such as a photomultiplier tube or photographic plate. Spectroscopists have long searched for an electronic measurement system that combines the simultaneous multiple wavelength coverage of the photographic plate with the real time electronic readout, large dynamic range, linear response, and sensitivity of the photomultiplier tube (PMT)(1). The first and most common multichannel detector is the photographic plate. It has several advantages; a relatively easy channel identification, simple operation, minimal investment, integrating capability, and it can be constructed in any size for any number of channels. Unfortunately, it has a limited dynamic range, a non-linear response and a time-consuming data retrieval procedure.

One type of multichannel system is based on the use of a PMT array which is optically arranged across the focal plane of a polychromator (2). Advantages included wide dynamic range, excellent sensitivity and rapid response. However, the number of possible channels is limited, and the PMT's must be arranged at wavelengths appropriate for predetermined specific applications.

New detector systems based on modern electronic image sensors can provide over 1000 independent optical channels. These detectors, when interfaced to a computer, provide excellent flexibility in data handling, signal processing, modes of integration, data accumulation and real time display. Nevertheless, they have limited resolution (due to the number of discrete light sensing elements), lower sensitivity than the PMT, incomplete readout (lag) and interchannel cross-talk (blooming).

The great potential of multichannel array, or TV type detectors such as charge coupled devices (CCD), vidicons (V) and linear diode arrays (LDA) for spectrometric measurements has been discussed in numerous recent reviews (3-10). In atomic spectroscopy, these detectors have been exploited primarily to carry out multielement analysis (9-16). For atomic absorption spectrometry, where light levels are reasonably high, much success has been achieved (15-18). For atomic emission or flourescence spectrometry, where light levels are much lower, Winefordner and co-workers (11,19) have concluded that multichannel detectors will find more limited use because of seriously degraded detection limits. However, in many cases where analyte concentrations are well above conventional detection limits, multi-element quantitative analysis with multichannel detectors by atomic emission has been demonstrated to be quite feasible (13,14,20,21). In atomic work, multichannel detectors have also been employed for profiling (22,23), correlation (24), time-resolved studies (25), spectral stripping (26) and internal standard compensation (27).

In molecular spectrometry, these detectors are not used as much for multicomponent analysis because molecular bands are much broader than atomic lines. However, Pardue, Milano and co-workers (28,29) have shown that multicomponent absorbance analysis is feasible for a few components in certain cases. In molecular work, these detectors are often employed as one means to construct a rapid scan spectrometer to monitor the output from an HPLC or GC (11,30-33).

One type of electronic image sensor is the self-scanning linear array of silicon photodiodes. This dissertation describes the design and characterization of a microcomputer controlled linear diode array detector spectrophotometer. The instrument was designed for multi-component determinations with absorbance values from 0 to 2.0 or more. Many of the properties of the polychromator and the detector are measured as these properties ultimately limit the performance of the instrument.

Chapter II presents a general overview of the instrument. Several important features of the linear diode array (LDA) spectrophotometer including the inlet system optics, polychromator and detector are described. Also the microcomputer system and the software system are presented. Chapter III discusses the design of the system electronics.

Included here is a discussion on the control circuitry and operation of the LDA, the interface between the LDA amplifier circuitry and the microprocessor, and other peripherals also connected to the microprocessor bus. Chapter IV presents the design of the system optics. The design of the inlet system optics was chosen for maximum light intensity over the entire wavelength region from 300-1000 nm. The polychromator was designed for a minimum of stray light and a flat field image across the light sensitive elements of the detector.

A major goal of the research was the characterization of the LDA spectrophotometer. The experiments and results are discussed in Chapter V. Properties of the polychromator examined include spectral region covered, resolution and stray light. Characteristics of the LDA examined are integration time, dark current, and saturation effects. Chapter IV explains the routine operation of the instrument and its application in molecular absorbance studies.

CHAPTER II

A FUNCTIONAL DESCRIPTION OF THE INSTRUMENT

A. Introduction

In this chapter a general overview of the instrument is presented. First, each basic part of the instrument is described in the context of its relationship with the remainder of the instrument. Second, the microcomputer software and its use are discussed, and further software which was used on the PDP 11/40 minicomputer for data analysis and plotting of the acquired data is also described.

B. A Block Diagram of the Instrument

The spectrophotometer discussed here is different from a traditional spectrophotometer in one most important aspect. This is the fact that light intensities over a large spectral region can be monitored in the time a normal spectrophotometer takes to produce results for a single wavelength. This is achieved through use of a linear diode array (LDA) which was purchased from Reticon Corp (34). A block diagram of the instrument can be seen in Figure 2.1.

The linear diode array, which is the distinguishing feature of this spectrophotometer, is a Reticon type RL-512S. This was the first solid state image sensor designed specifically for spectroscopy applications. Some

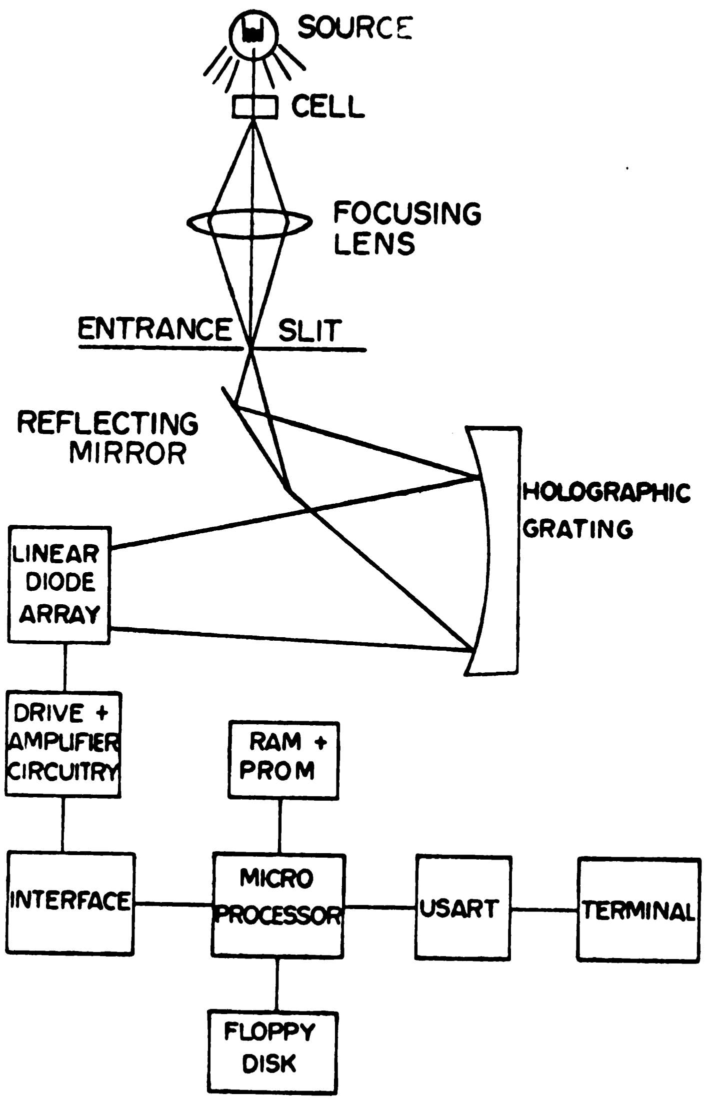


Fig. 2.1 A block diagram of the LDA spectrophotometer

important features of this device are:

1) There are 512 individual sensor elements (each 25 microns wide by 2.5 mm high) which allow the spectrum to be divided into 512 simultaneously exposed channels.

2) The LDA is self-scanned to provide real time computer compatible serial electronic output for all 512 channels in succession.

3) The variable integration time (from 2.3-400 msec at 20°C) can provide a dynamic range greater than 10,000:1. The integration time can be optimized depending on the wavelength and the light level.

4) The detector is responsive over a range of 200-1000 nm.

When using a device such as this linear diode array which can generate data at extremely fast rates, it becomes apparent that the data acquisition is best done by computer. For the operation and control of this instrument, an Intel 8085A microprocessor was chosen. It is an 8-bit microprocessor that is powerful enough to fully control the instrument functions as well as perform the data acquisition process. The microprocessor and a variety of interfacing integrated circuits have been designed to make interfacing relatively simple. Two kilobytes of programmable read only memory (PROM), 4 k-bytes of random access memory (RAM) and a universal synchronous/asynchronous reciever transmitter (USART), all designed by Bruce Newcome (35), have been built and implemented in this system. Operator interaction and control of the instrument is provided through a Hazeltine 1400 terminal which is connected through the USART to the microprocessor bus. Also interfaced to the bus is a Persci dual floppy disk system. This magnetic medium provides storage space mainly for data files but also for software. The floppy disk is currently the only link between the spectrophotometer and our research group's PDP 11/40 minicomputer where further data analysis is performed.

The microprocessor is also directly interfaced to the linear diode array. Data are acquired by first converting the analog video signal to a sequential digital output which can be stored in RAM. The conversion time for the 12 bit DATEL EH12B analog to digital converter is less than 4 microseconds.

The integration time is the time between the readout of a specific diode from one scan to the next. This integration time can also be selected by the operator through a command to the computer.

Every spectrophotometer needs not only a good means of data acquisition and subsequent analysis, but also a very good optical design. The major optical components of the polychromator in this spectrophotometer are a concave holographic grating, a reflecting mirror and an entrance slit assembly. All of these were purchased from Instruments S.A. Inc. The grating provides a flat field image with minimum stray light. The grating was specially made (F 2.0) for the desired dispersion (400 nm) across the 512 elements of the linear diode array.

The source optics were designed for high light intensity so that the absorption of highly absorbing materials could be measured. A General Electric #1974 lamp was chosen because it has a very compact filament and sufficient output in the visible region of the spectrum. The light then passes through the cell (in its constant temperature incubator) and is then focused onto the entrance slit with a double convex lens. The order of these optical components is different from that in spectrometers with single-channel detectors. In any array detector spectrometer the dispersive element must directly precede the detector. This requires the cell to precede the entrance slit in the optical path. The fused silica lens (F 1.5) is closely matched to that of the grating to optimize light throughput.

C. Microcomputer System Software

A microcomputer software system could be either a single large program or at the other extreme, a collection of subroutines. However, all software systems must allow ease of instrument operation and data storage. In cases where many different types of experiments will be conducted, it is advantageous to be able to reprogram quickly. This is possible in a subroutine structured software system.

Listings of all the routines mentioned in this section can be found in Appendix A. The software system is called Structured Library Oriented Programming System (SLOPS). This operating system, as well as all the general use commands listed in Table 2.1, were written by Hugh Gregg (36). This software system allows the user to "create" new entries in the library of commands. Each entry can do a specific task or can call any or all of the other library entries as subroutines. For example, when you type the DISPLY command on the terminal, the microprocessor first looks through its library and if it finds DISPLY it begins execution. Upon completion of the DISPLY routine the computer will prompt SLOPS> and wait for a new input command.

All library entries are assigned a specific address in RAM when the routines are compiled. It is through this "stack" that the cpu searches its library entries. If one subroutine calls another, the compiler gives the address of the called subroutine. It is then executed and returns to the original subroutine.

A number of library entries were created specifically for use with the linear diode array spectrophotometer and these are listed in Table 2.2. These subroutines are easily incorporated as library entries in SLOPS and are easy to use. The SCAN routine enables the interface to collect and

TABLE 2. 1

Current SLOPS Commands

Command	Subroutines called	Action
ADDR		Returns address of library entry
TOASCI		Binary to ASCII
UNASCI		ASCII to Binary
BLANK	TTYOUT	Blanks terminal screen
BRKDWN	DCMP	Breaks input line into words
CHARIN		Inputs single character
CHECK		Compares current word in library
CLRDSK		Disk data sink
CONVERT	NUMBER, CVTEXT PUSH2, PRINT PRINTR, CRLF	Converts any number to any base
CRLF	TTYOUT	Sends CR+LF to terminal
CVTEXT	PUSH1, DIV TOASCI, DCMP	Converts number to ASCII string, any base
CVTINT	DMULT, UNASCI	Converts ASCII string to number
DCMP		Compares (D, E) to (H, L)
DDIV	DSUB	(B, C)=(H, L, B, C)/(D, E)

Command	Subroutines called	Action
DISK	PUSH2, PRINT	Disk monitor routine
	TTYIN, CHARIN	
	DINCHR, DDATA	
	DDATUM, DCMD, DWAIT	
DDATUM	DWAIT	Outputs data character to disk
DDATA	DDATUM	Outputs data string to disk
DCMD	DWAIT	Outputs command to disk
DWAIT		Waits till disk ready
DINCHR		Gets character from disk
DIV	DDIV	(D,E)=(D,E)/(A)
DMULT	MULT	(D,E)=(A)*(D,E)
DSUB		(H,L)=(H,L)-(D,E)
INIT	BLANK, KERNAL	Initializes system
KERNAL	PUSH2, PRINT	Main monitor
	TTYIN, BRKDWN	
	WORD, SEARCH, ADDR	
LINK		Library pointer update
MULT		(D,E)=(D)*(E)
NUMBER		Gets number on stack
PRINT		Prints message on terminal from top of stack
PRINTR		Prints ASCII characters found on top of stack
SEARCH	CHECK, LINK	Searches library for command

Command	Subroutines called	Action
POP1		Pop 1 byte off stack
PUSH1		Push 1 byte on stack
POP2		Pop 2 bytes off stack
PUSH2		Push 2 bytes on stack
TTYIN		Accept and echo input line
TTYOUT		Send character to terminal
WORD		Pops pointer to next word off stack

TABLE 2.2

Current Library Entries for Instrument Operation

Command	Subroutines called	Action
SCAN		Enables interface and collects data
DISPLY	CVTEXT, PRINTR DCMP, TTYOUT, CRLF CHARIN, PUSH2, PRINT	Display data on terminal
STORE	CLRDSK, PUSH2 PRINT, DDATUM TTYIN, DDATA, DCMP DINCHR, TTYOUT	Stores the data on disk
CDATA	PUSH2, PRINT, SCAN DISPLY, PRINT2 TTYIN, STORE	Combines the SCAN, DISPLY, and STORE routines
SIGAVG	DCMP, PUSH2 PRINT, TTYIN BRKDWN, NUMBER DISPLY, STORE	Signal averages user entered number of scans
INT	PUSH2, PRINT, TTYIN BRKDWN, NUMBER DMULT, DIV	Sets the integration time
PLOT	DCMP, DSUB TTYOUT, CRLF	Plots the data on the terminal

store the data from the linear diode array in RAM. The DISPLY routine displays the data in base 10 on the terminal screen in a 51 row by 10 column format. The STORE routine copies the data from RAM to a file on the floppy disk after asking the user for a file name. The routine CDATA basically calls all the above routines for easier operation. SIGAVG, a signal averaging routine, asks the operator for the number of scans to average, then collects the data, averages and displays the averaged data on the terminal. Another routine, INT, asks the user for the desired integration time and sets it accordingly. A final routine, PLOT, plots the data in a 24 by 64 character grid on the terminal screen.

D. Data Analysis Software.

As mentioned before, all data analysis software consists of routines created on our PDP 11/40 minicomputer. The data files stored on a floppy disk are first transferred to the minicomputer. They are currently in an unformatted binary form. After loading a data floppy on the 11/40, the routine ABSORB can be executed. It calculates the absorbance at each wavelength according to the following equation:

$$A_I = -\log[S_I - D_I]/[R_I - D_I]$$

where S_I is the signal intensity at wavelength I, R_I is the reference intensity with a reference solution in the sample cell, and D_I is the signal obtained when no light is falling on the detector (dark current). The absorbance file is created and stored. Before plotting, the file must be converted from an unformatted binary file to an ASCII file. The routine CONVERT is used for this conversion. At this point, readily available plotting routines can be used. One of these is MULPLT, written by Dr. T.V. Atkinson, which can plot intensity or absorbance vs. wavelength on either the terminal or the line printer. Figure 2.2 shows a typical absorbance spectrum of $KMnO_4$.

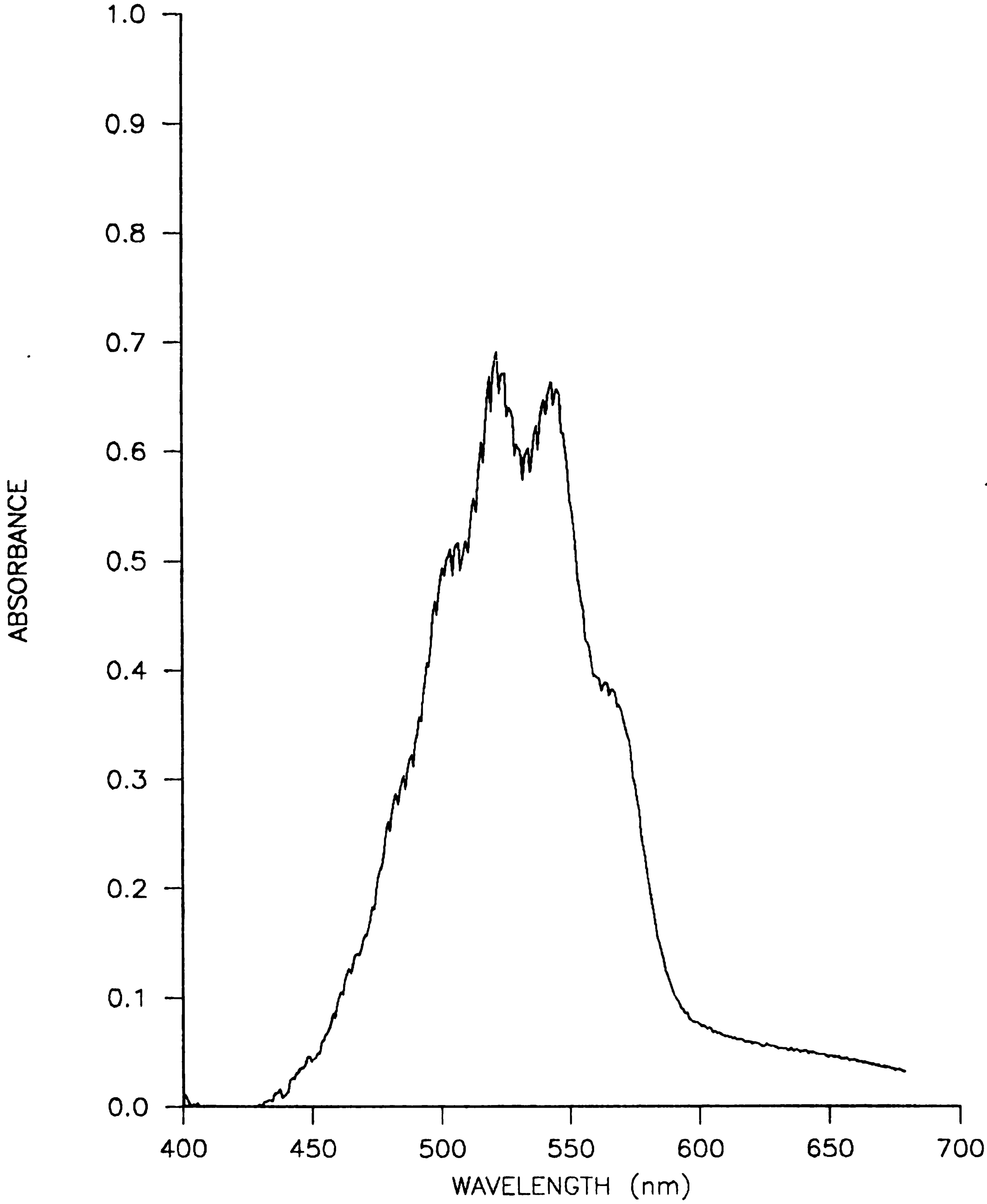


Fig. 2.2 Absorbance spectrum of $KMnO_4$

CHAPTER III

THE DESIGN OF THE SYSTEM ELECTRONICS

A. Introduction

With an instrument such as this linear diode array spectrophotometer, data can be produced at an incredible rate. To avoid loss of information, it is necessary to collect and store these data at the same rate they are produced. This requires the use of a computer. Microcomputers today are very cost-effective. The instrument can be automated with one of these processors very easily. This might allow both the operator and larger computers to spend their time on bigger and better things.

B. Linear Diode Array, Clock Drive and Amplifier Circuits

The RL512S array as well as the clock drive and amplifier circuits were purchased from Reticon Corp (34). The Reticon RL512S is a monolithic self-scanning linear photodiode array. This device consists of a row of silicon photodiodes, each with an associated storage capacitor on which the photocurrent is integrated. The multiplex switches at each photodiode are switched in sequence by an integrated shift register scanning circuit to provide a serial output of the charge accumulated at each diode. The array is packaged in a 22 lead dual in line integrated circuit package with a polished quartz window face plate. The pin configuration is shown in Figure 3.1. The diode elements are on 25 micron centers which correspond to a density of 40 diodes/mm. The overall length of our 512 diode array is thus 12.8 mm. The height of each diode is 2.5 mm, which gives each element a slit-like geometry with a 100:1 aspect ratio. This is desirable for use with a spectrograph. The sensor geometry is shown in Figure 3.2. The diodes consist of diffused p-type bars in an n-type silicon substrate. Light incident on the sensing area generates charge which is collected and stored on the p-type bars during the integration period. The accumulated charges are then sequentially switched into the video output for readout. The n-type as well as the p-type silicon surface is photosensitive. Light incident on one of the p-regions will generate charge which is stored on that diode. Charge generated by light incident on the n-type surface between 2 p-regions will divide between the adjacent diodes to produce the response functions shown in Figure 3.2.

A simplified equivalent circuit of the linear diode array is shown in Figure 3.3. Each cell consists of a photodiode and a dummy diode, both with associated storage capacitance. These diodes are connected through MOS multiplex switches to video and dummy recharge lines. One pair of recharge lines is common to all the odd elements and another pair is common to all the even elements.

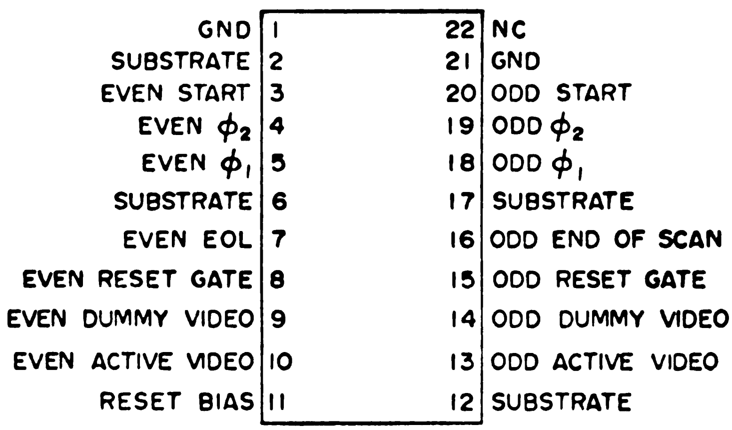


Fig. 3.1 Pin configuration of the LDA integrated circuit

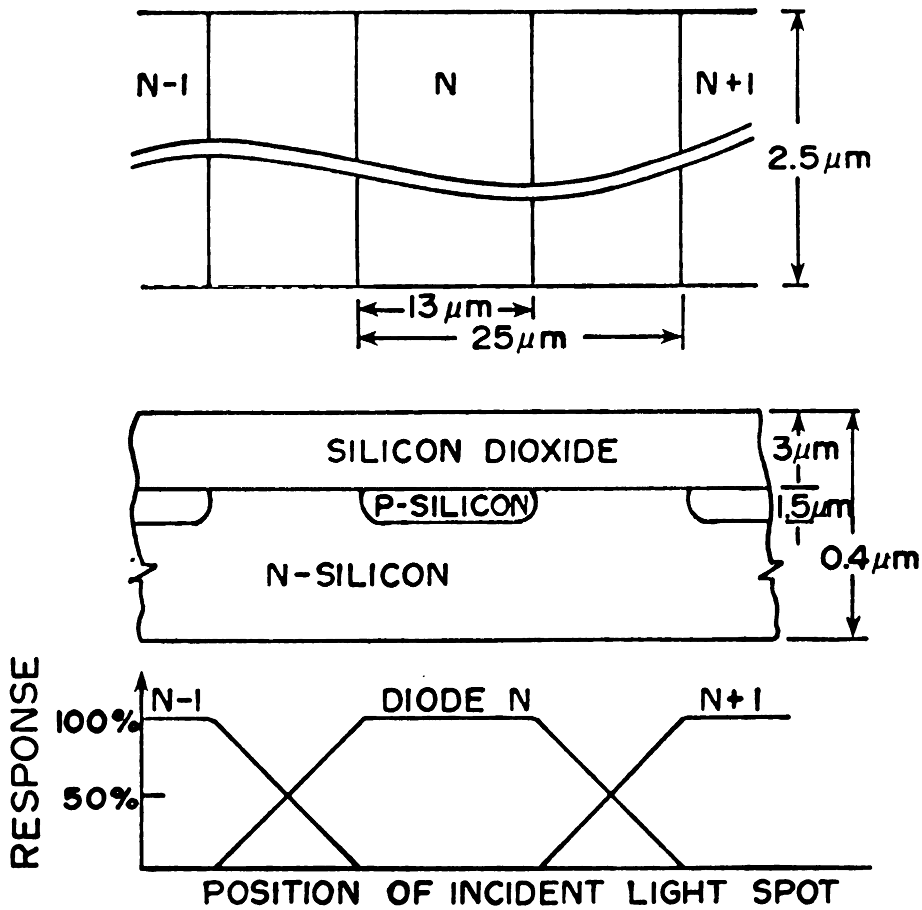


Fig. 3.2 Sensor geometry and aperature response function

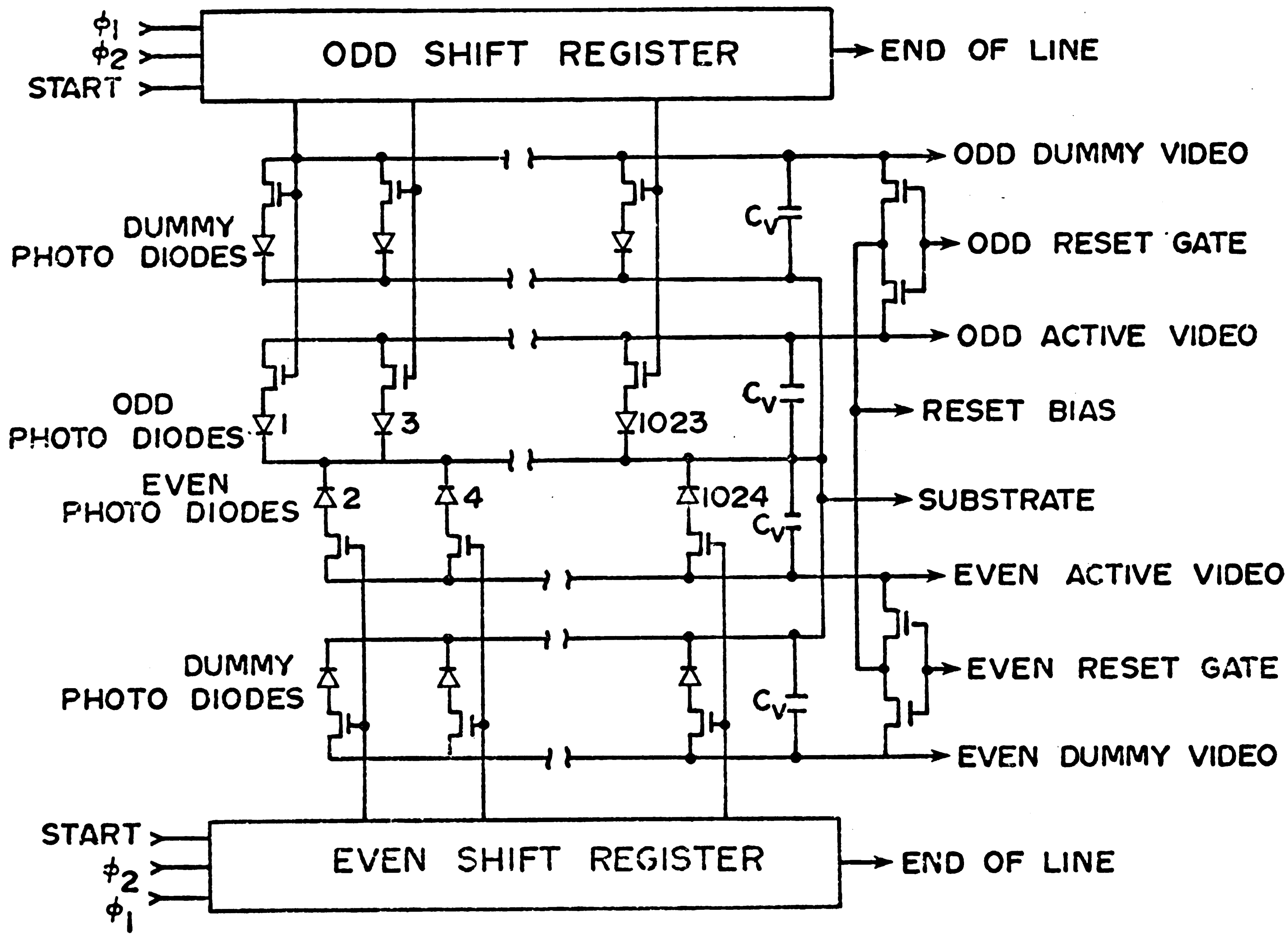


Fig. 3.3 Equivalent circuit

The multiplex switches are turned on and off in sequence by shift register scanning circuits which periodically recharge each cell to 5 volts and store a charge Q(sat) on its capacitor. The shift registers are driven by multiphase clocks with periodic start pulses introduced to begin each scan. The cell-to-cell sampling rate is determined by the clock frequency. The integration time is the interval between start pulses. During this integration time, the charge on each capacitor is gradually removed by the reverse current flowing in the associated diode. The reverse current is made up of two components, the photo current i_p, and the dark leakage current i_d. The photocurrent is the product of the diode responsivity and the light intensity or irradiance.

During a line scan time, the charge Q removed from each cell is the product of the photocurrent and the line time. This charge must be replaced through the video line when the diode is sampled once each scan. Thus, the output signal obtained from each scan is a train of 512 charge pulses each proportional to the light intensity on the corresponding photodiode. By properly phasing the clock drives to the two shift registers all of the diodes can be sampled in proper sequence. The two video lines can then be simply connected together to provide a continuous train of output charge pulses.

In addition to the signal charge, switching transients are capacitively coupled into the video lines by the multiplex switches, These same transients are introduced into the dummy lines and therefore can be eliminated by reading out the video and dummy lines differentially. Output pulses are provided when the last odd and even elements are sampled by the shift register scanning circuit.

The RC-1024SA interface board has 5 outputs:

1) Two end of line outputs for even and odd shift registers

2) Start output to monitor the beginning of the scan

3) An oscillator clock output

4) A combined even-odd video output

C. The Microcomputer Interface to the RC-1024SA Amplifier Board

The interface discussed here was designed and built for use between the SDK-85 microcomputer and the RC-1024SA amplifier board. A block diagram of the circuit is shown in Figure 3.4. All circuit diagrams can be found in Appendix B.

The basic operation of this interface includes a direct memory access (DMA) circuit which loads the output of the analog to digital converter (ADC) directly into RAM after which the microprocessor can access the RAM for any other purpose. To initiate data collection, a chip select ($\overline{CS}$) and high DO signals are sent to the control logic. At this

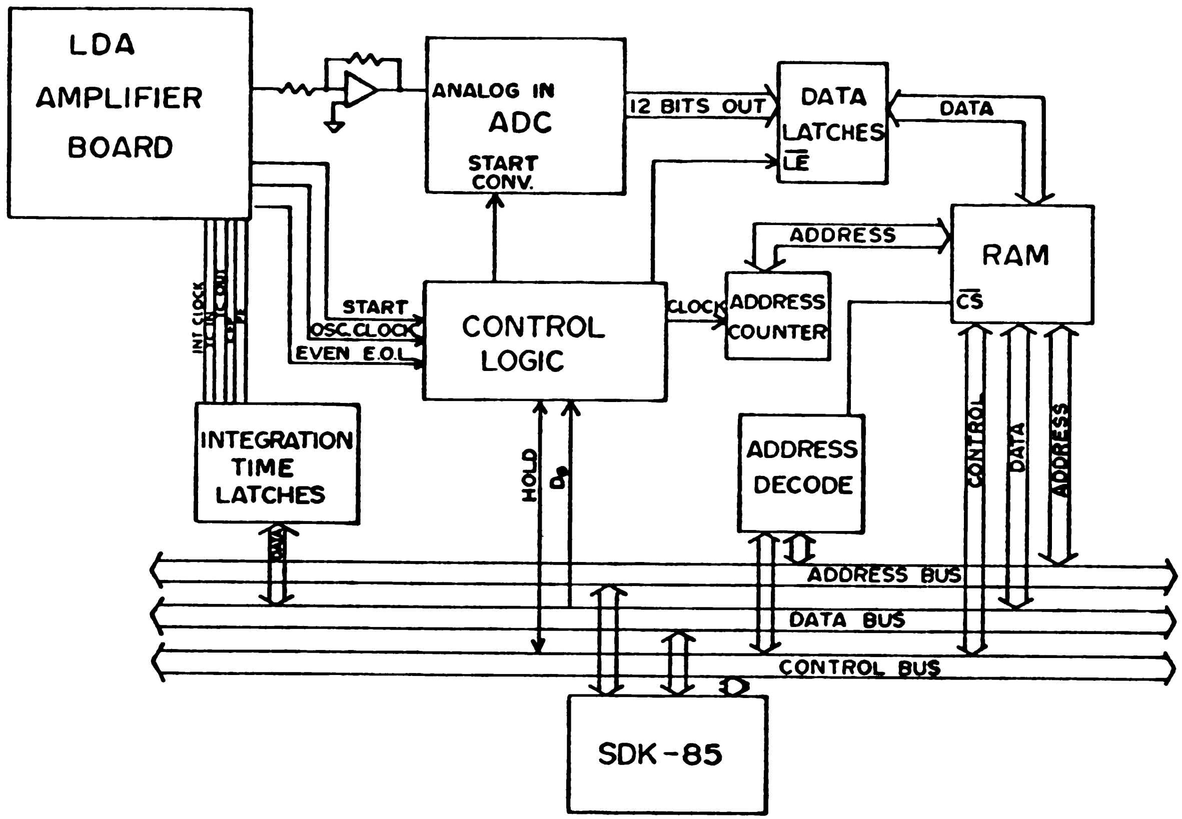


Fig. 3.4 Block diagram of interface between amplifier board and SDK-85

point the microprocessor is put on hold and the control logic waits until a start pulse is found. This pulse comes from the Reticon amplifier board. Once this happens, the control logic sends out start conversion, address increment, write enable, chip select, and latch enable pulses. All these are timed properly to load each converted data point into its sequential place in RAM. See the timing diagram also in Appendix B. Immediately before the ADC however, is a simple amplifier with gain which converts the 0 to 3 volt output (no light to saturation) of the detector board to the 0 to 10 volt input range of the converter. The conversion time of the Datel EH12B2 ADC is 4 μ-sec. This allows a maximum theoretical clock rate of 1 MHz. The actual clock rate is 800 KHz. At the end of a complete scan, the even end of scan signal shuts off the control logic and releases the microprocessor to continue execution of any program.

The integration time can also be controlled by the microprocessor. The array amplifier board has a series of 3 4-bit synchronous counters which delay the start pulse by the amount of time set at the preload switches. These preload switches were removed and a set of latches tied to the data bus were installed in their place. These allow the computer to vary the integration time from 2.7 msec to to 1320 msec in 5.2 msec increments.

D. The Microprocessor and its Peripherals

The 8085A microprocessor was chosen because of its high performance (1.3 μ-sec instruction cycle), popular instruction set, and low cost. This processor is included in the MCS series System Design Kit (SDK-85). The SDK-85 was used because it is a complete single board computer with memory and I/O. Included in the SDK-85, which was purchased from Intel Corporation, is the 8085A CPU, 2 kilobytes of Read Only Memory (ROM), 256 bytes of Random Access Memory (RAM), 38 paralell I/O ports, and 1 serial which utilizes the SID, SOD pins of the 8085A and which has a software generated baud rate of 110. The SKD-85 also has an interactive LED display and keyboard with extensive monitor software in the 2 k of ROM which acts as an excellent front panel (useful in debugging custom interfaces).

Figure 3.5 is a block diagram of the SDK-85 and all the peripherals in the system with the exception of the previosly described interface to the LDA amplifier board. Appendix C contains the schematic diagrams for the peripheral boards which Bruce Newcome designed (35). There are 4 k bytes of RAM, 2 k bytes of Programmable ROM (PROM) and 1 Universal Synchronous/Asynchronous Reciever Transmitter (USART) connected to the bus in the micro system. The PROM is used for the storage of all the SLOPS routines listed in Table 2.1. The RAM is used for all user written routines. The USART is used for serial

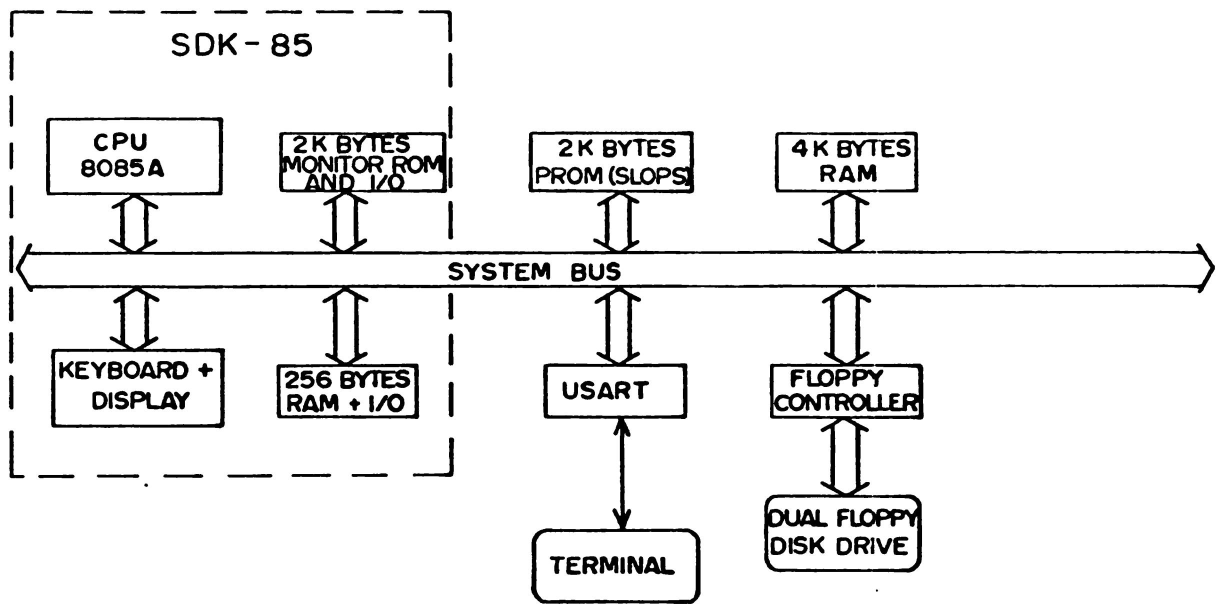


Fig. 3.5 Block diagram of the CPU and its peripherals

communication to the Hazeltine 1400 terminal. The floppy controller is a Persci Model 1070 which has its own 8080A processor and programs which control the connected Persci Model 760 dual floppy disk drive (37).

CHAPTER IV

THE DESIGN OF THE SYSTEM OPTICS

A. Introduction

In this chapter both the inlet system optics as well as the design and construction of the polychromator are discussed. The inlet system optics were designed for high light intensity from a near point source. This facilitates experiments where interest is focused on samples of high absorptivity as the detector sensitivity is not as great as a photomultiplier tube, especially in the ultra-violet region of the spectrum. The polychromator design centers around the use of a holographic grating. The grating is designed to produce a flat field image across the photosensitive elements of the detector with a minimum of stray light. Figure 4.1 is a photograph of the spectrophotometer optics. The polychromator and inlet system optics are mounted on a pair of perpendicular optical rails. This arrangement provided the necessary flexibility during the development of the spectrophotometer.

B. The Inlet System Optics

The design of the inlet system optics was selected so that the light intensity at all wavelengths generated by the source is maximized at the detector surface. The source used is a General Electric #1974 quartz iodide lamp. It is

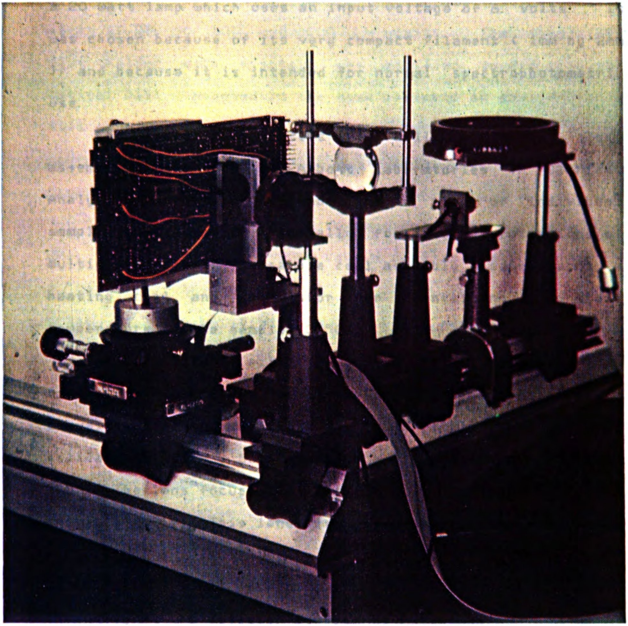


Fig. 4.1 Photograph of the array and optical system

a 20 watt lamp which uses an input voltage of 6 volts. It was chosen because of its very compact filament (1mm by 2mm), and because it is intended for normal spectrophotometric use.

The light then enters the cell assembly. This assembly, taken from an Abbott Laboratories VP Clinical Analyzer, supports a doughnut shaped ring of individual sample compartments which allow for easy determinations of multiple samples. The sample cell assembly also contains a heating coil and thermistor for maintaining a constant temperature of the sample (35°C).

As the light exits from the cell assembly, it disperses. To collect the dispersing light, a simple double convex lens is inserted into the light path. It is a fused silica lens with a 50mm diameter and a 75mm focal length (F 1.5). The lens focuses all the collected light onto the entrance slit. This lens also serves another purpose; the F number of the lens closely matches that of the grating, which allows full use of the grating surface (i.e. light intensity at the detector surface is maximized).

C. The Polychromator

A polychromator is a spectrum dispersing device which produces simultaneous spectral information over a band of wavelengths.

The entrance slit assembly, purchased from Jobin Yvon

Division of Instruments S. A. Inc., has interchangeable slits of 25, 50, 100, and 1000 μm widths. The model H20 slit holder with fishtail was modified to fit onto the optical rail. Mounted in the same assembly as the entrance slit is a front surface mirror which serves only to move the source optics away from the detector. An approximate drawing of the spectrograph is shown in Figure 4.2.

In a linear diode array spectrophotometer such as this one, it is most important to have a grating that can produce a "clean" spectrum on a flat surface. The linear diode array is then positioned in this plane so that the optical quality of the image is identical from one end of the spectrum to the other. One of the newest advances in holographic grating design technology has made possible the production of flat field spectrographic, concave, aberration-corrected gratings which by themselves produce a flat spectrum. This spectrograph, in addition to permitting the use of the full advantages of multiwavelength detection, has two other advantages; its extremely low stray light level, and its extreme simplicity.

The concave, holographic grating, also purchased from I. S. A. Inc., has a focal length of about 100mm and a diameter of 50mm (F 2.0). It is also mounted onto the optical rail. Specifications related to the grating can be found in Appendix D. The array is mounted in a similar manner in the focal plane.

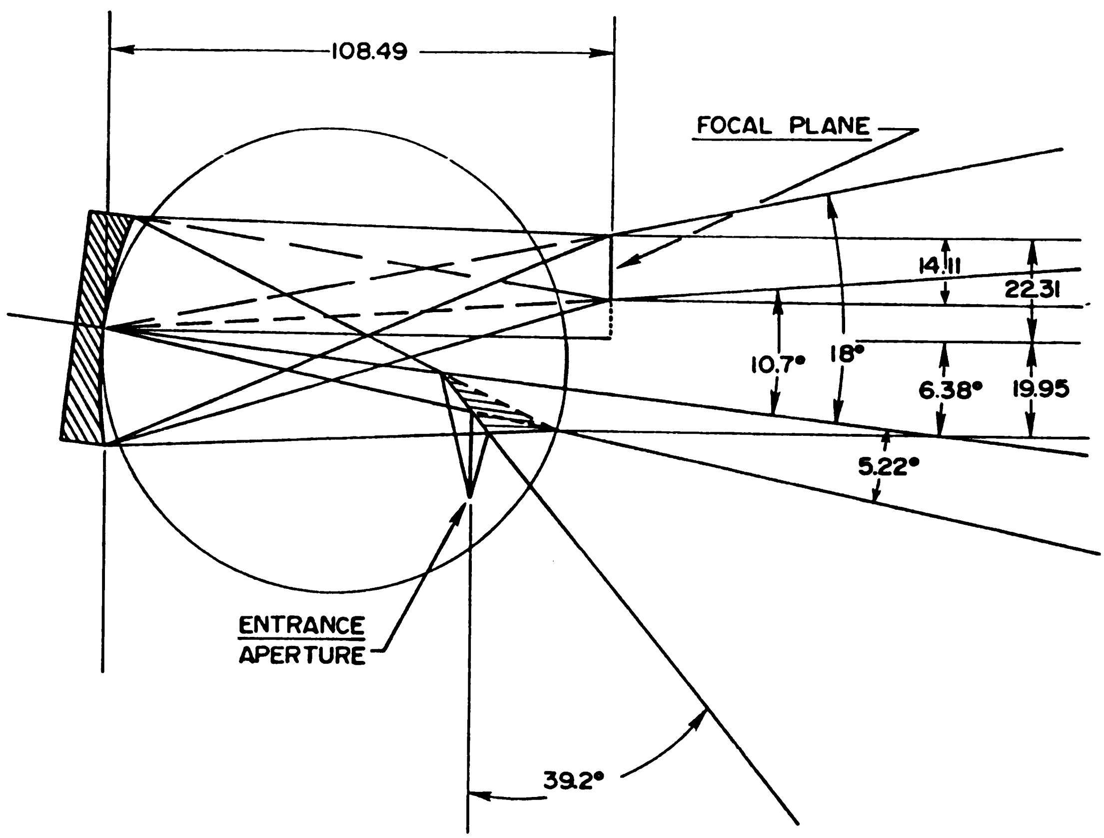


Fig. 4.2 Approximate drawing of the polychromator

CHAPTER V

CHARACTERIZATION OF THE LINEAR DIODE ARRAY SPECTROPHOTOMETER

A. Introduction

In this chapter the characteristics of the linear photodiode array spectrophotometer are discussed. The polychromator properties that are emphasized here include spectral region covered, resolution and stray light. The linear diode array characteristics emphasized are integration time, dark current and saturation effects.

B. Dark Current

One of the major operational characteristics of the linear photodiode array is its electronic background noise. This background signal can be seen in the spectrum of a neon filled hollow cathode lamp shown in Figure 5.1. The slit width used is 25 μm and the integration time is 200 msec. The electronic background, or dark current, is the major source of the overall height of the pedestal observed in Figure 5.1. The stray light can also be seen. However, it is a small percentage of the total height. The noise across the array is due to diode to diode dark current variations.

A complete dark current spectrum is obtained by blocking the entrance slit, collecting the data and storing them on a floppy disk. At any given integration time, the dark current is very reproducible. At an integration time

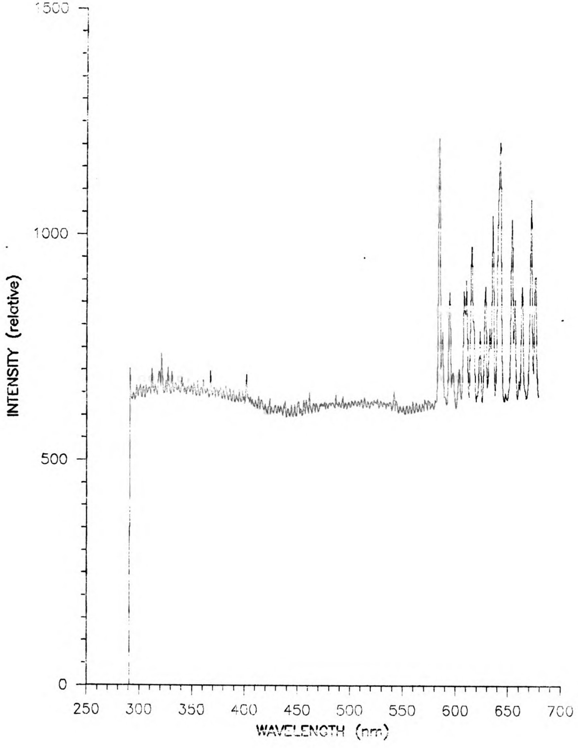


Fig. 5.1 Spectrum of a neon filled hollow cathode with electronic background

of 50 msec, the dark current in a given channel has a maximum relative standard deviation of 1.29%. The %RSD is calculated at all wavelength channels from 5 collected scans and the reported value is the maximum observed at any channel.

As the dark current is very repeatable from scan to scan, it can be removed from any spectrum by subtraction. This operation is easily accomplished by software on the 11/40. The neon spectrum shown in Figure 5.1 is shown in Figure 5.2 minus the dark current. Even if the spectral signal is so small as to be essentially obscured by the background, subtraction can still recover a useful spectrum. Except where noted, all spectra shown in this thesis are background subtracted spectra.

The presence of the dark current can severly limit the effective use of the integrating capability of the array. In an integrating type of detector, such as the linear photodiode array, the charge from the dark current increases as the integration time increases and can, at integration times longer than 2 seconds, completely saturate the array (i.e. completely discharge the reverse bias on the diode). This leaves no dynamic range for signal measurement. The background signal as a function of integration time of one photodiode is shown in Figure 5.3. At an integration time of 10 msec the array dark current is nearly zero. At an integration time of 600 msec, just over one third of the

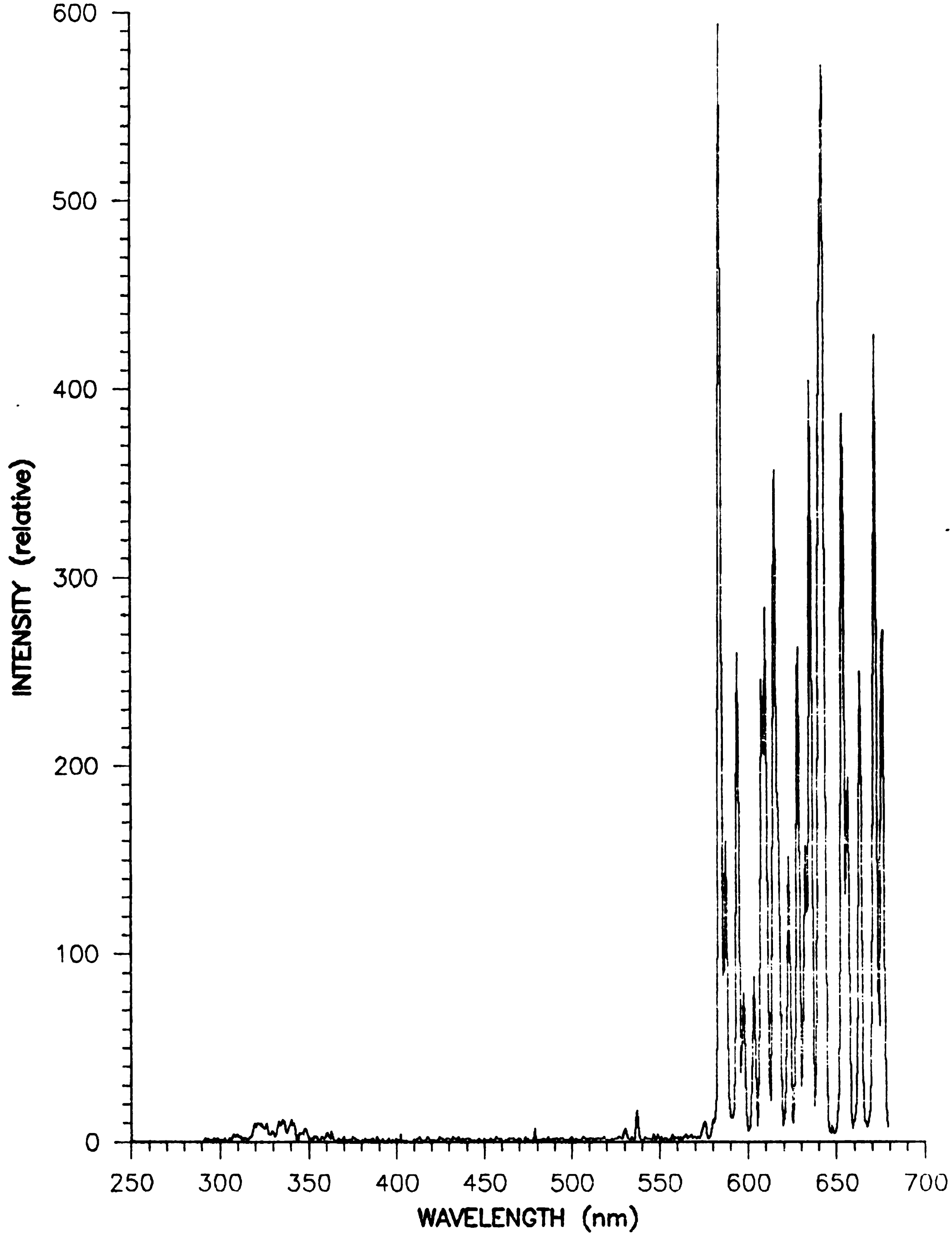


Fig. 5.2 Spectrum of a neon filled hollow cathode with background subtracted out

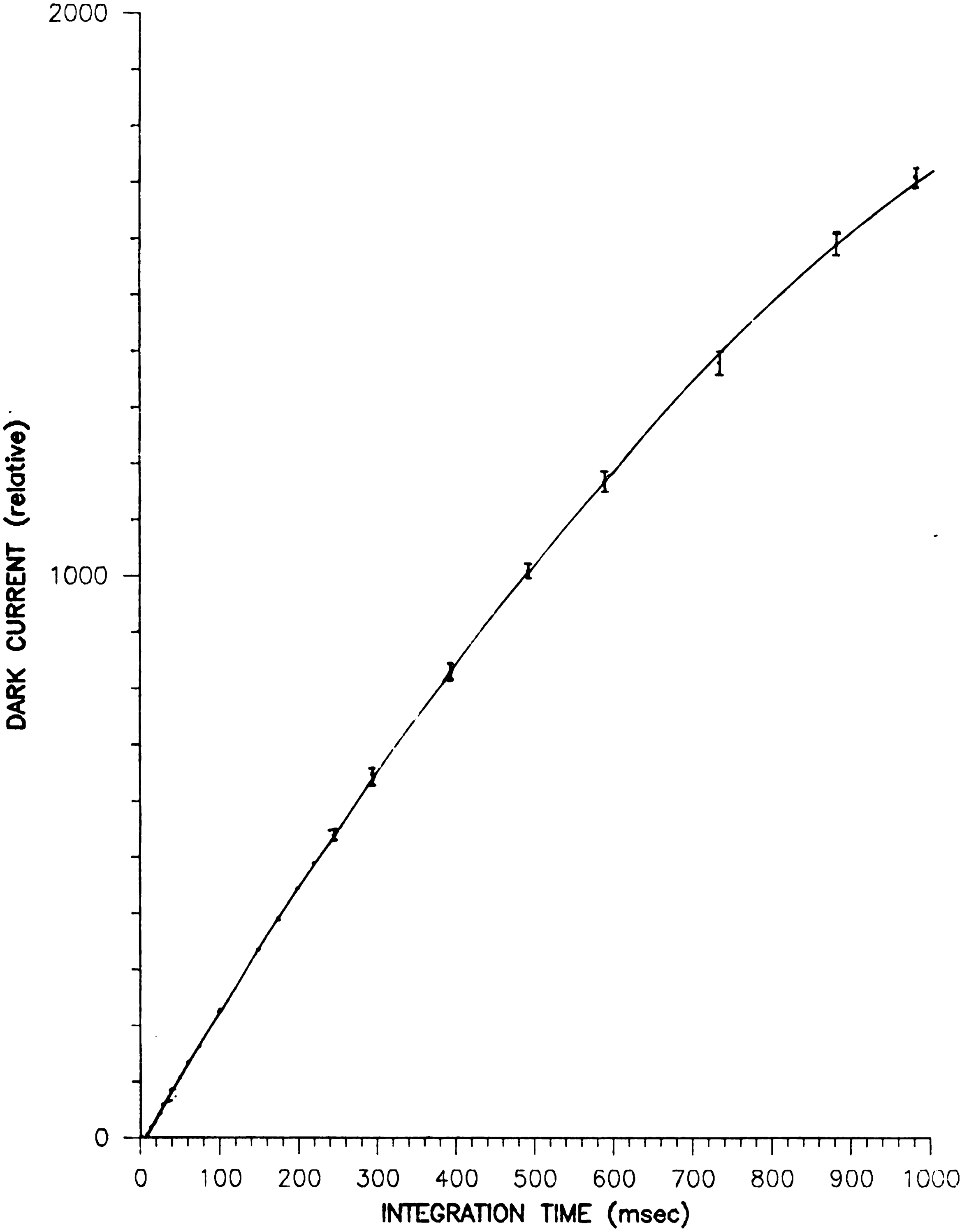


Fig. 5.3 Background signal vs. integration time

useful dynamic range is taken up. At an integration time of 1 sec only one third of the dynamic range is left for the signal and at an integration time of over 2 sec the array is completely saturated; hence no signal can be measured. The non-linear response is most probably due to a charge leakage from the video recharge line on the LDA integrated circuit. It is expected that cooling the array will reduce this charge leakage to a negligible value and the plot will approach linearity. The dark current level of the array can also be significantly reduced by cooling. According to the literature (6), if the array were cooled to about −15 °C, the dark current at an integration time of 15 sec would be the same as that for an uncooled array at an integration time of 45 msec.

C. Integration Time

Charge integration is a signal enhancement technique which takes advantage of the integrating capability of the LDA. The integration time is the time difference between the readout of a specific diode from one scan to the next. Charge integration can be a very useful signal enhancement technique as the S/N (signal to noise ratio) under many conditions increases linearly with the integration time, while the S/N increases only with the square root of the number of scans averaged (38-41). As mentioned before, the integration time is under microcomputer control. Detector response (charge integrated) should be proportional to the integration time.

Figure 5.4 shows a plot of detector output at 650 nm of the quartz halogen lamp versus integration time. The integration time was varied from 7.85 msec to 983.5 msec. The slit width was 25 μm. The intensity of the line was reduced using a neutral density filter which allowed 1.0% of the light to pass. The detector output was recorded for 5 scans and the standard deviation (SD) calculated. The SD never exceeded 11.5. The non-linearity is again explained by the charge leakage in the LDA and is expected to improve with cooling.

D. Stray Light

One of the principal reasons for using the previously described holographic grating is its low stray light levels. To measure stray light, a monochromatic source at a known intensity is used. The intensity of radiation observed at the other wavelength channels is measured and the stray light is reported as a percent. Figure 5.5 shows the percent stray light at all wavelength channels which results from a He-Ne laser as the source (632 nm). An integration time of 1.068 sec was used. The intensity of the laser line was calculated by measuring the intensity at a minimum integration time and extrapolating. This is possible as the detector response is related to the integration time as

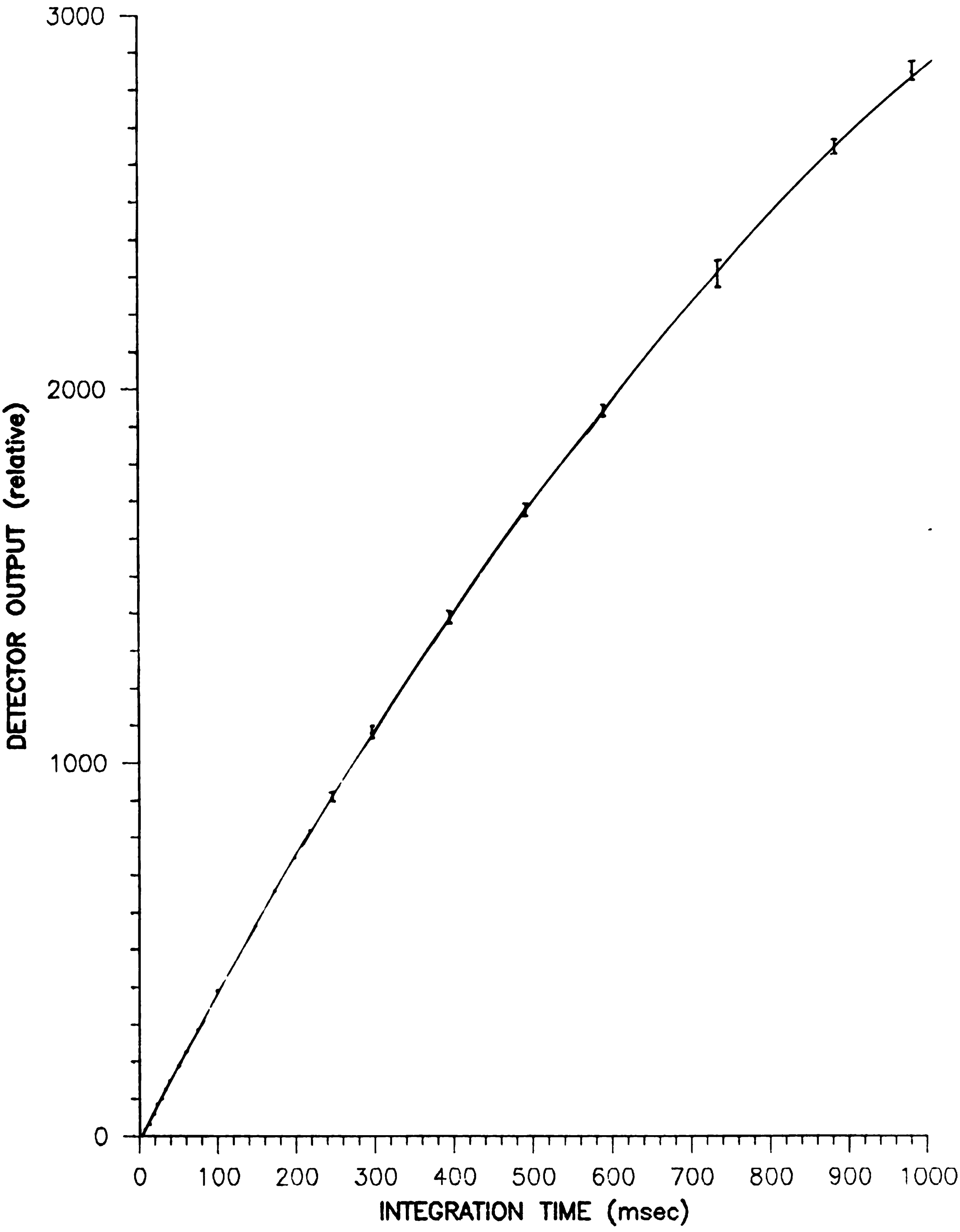


Fig. 5.4 Detector output vs. integration time

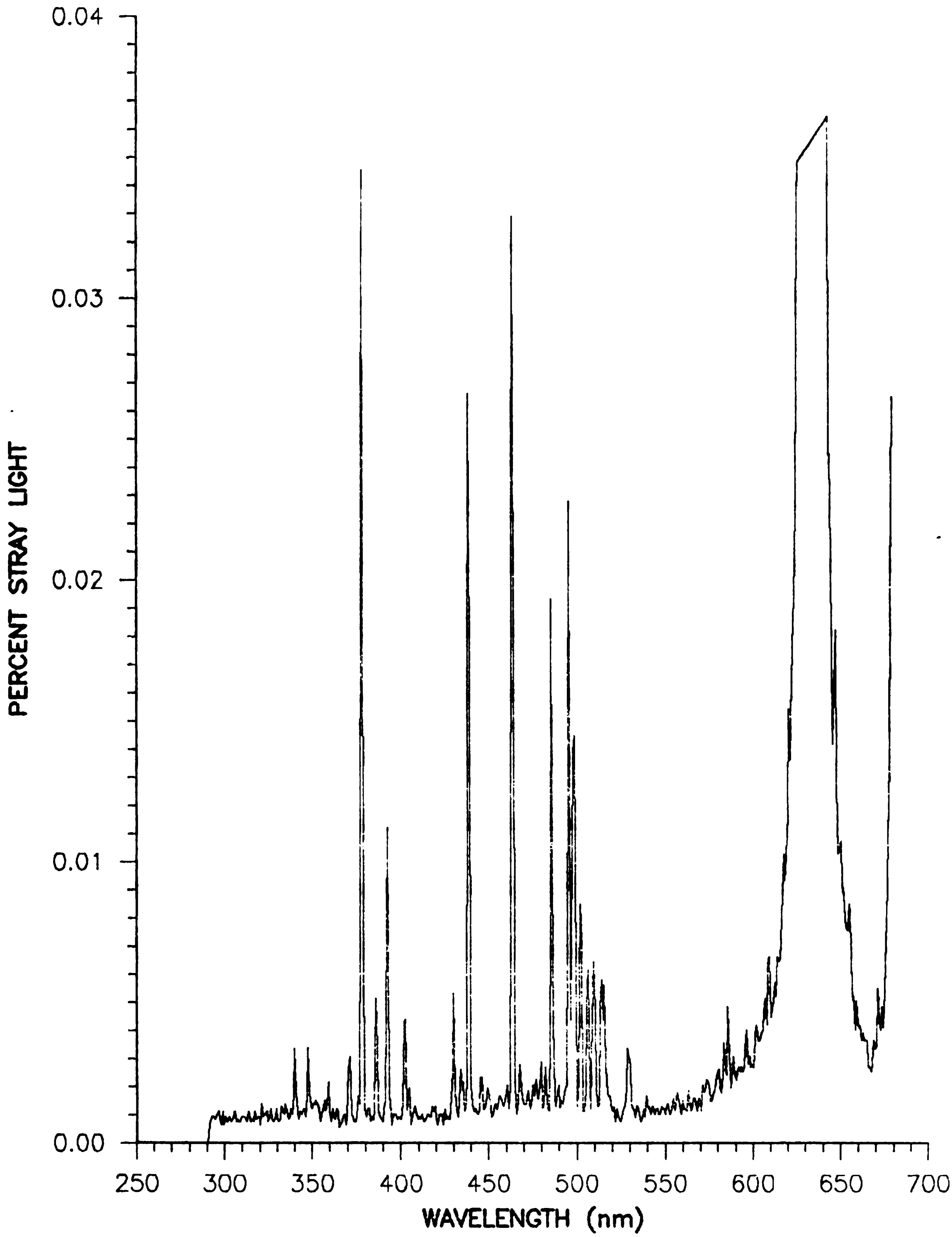


Fig. 5.5 Percent stray light

shown in the last section. The spikes seen in several of the channels of Figure 5.5 are due to reflections inside the polychromator.

Even at stray light levels as low as those shown here, it can be seen that at long integration times, errors in measurements can easily be observed. Because the polychromator is after the sample in the optical path, stray light is produced at all wavelengths that the source emits. For experiments in the ultra-violet region of the spectrum, stray light from the red region of the spectrum where the source intensity is much greater (in the case of the quartz halogen source) becomes a serious problem.

In Figures 5.6 and 5.7, spectra of the quartz halogen lamp are shown without and with a red cutoff filter in place. Each plot shows multiple spectra at integration times of 5, 10, 20, 50, 200 and 500 msec. At 300 nm, the source intensity is near 0. However, it is easily seen that light is being detected (Fig 5.6). This light is stray light due to all wavelengths entering the polychromator. Even with the red cutoff filter in place (Fig. 5.7), the stray light can be seen (300 nm), but it is significantly reduced. This greatly increases the dynamic range which is of great importance for experiments in the ultra-violet part of the spectrum. The saturation level appears to decrease in the higher integration time spectra of Figures 5.6 and 5.7 because the dark current level, which is increasing, is

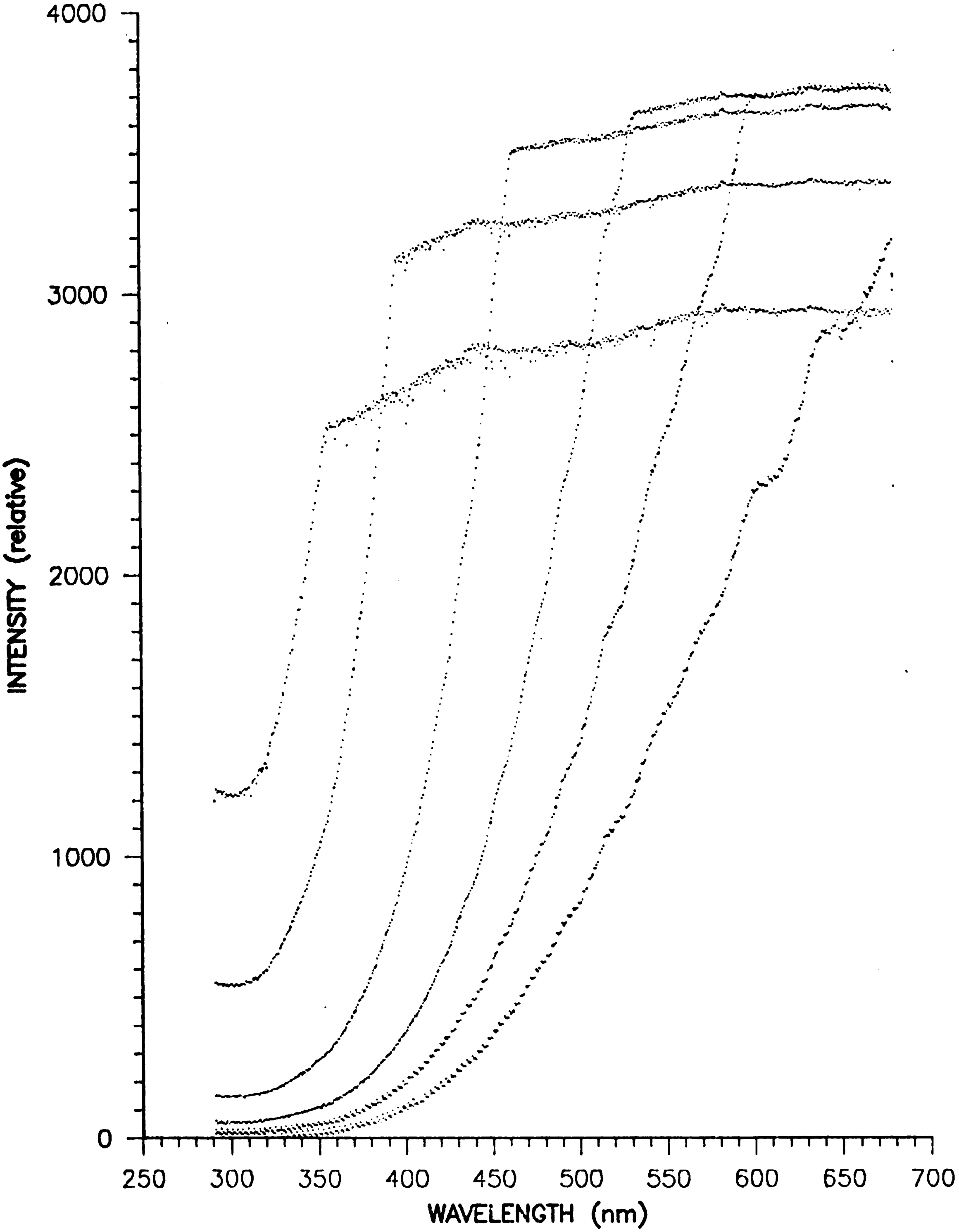


Fig. 5.6 Q. I. lamp spectra at varying integration times

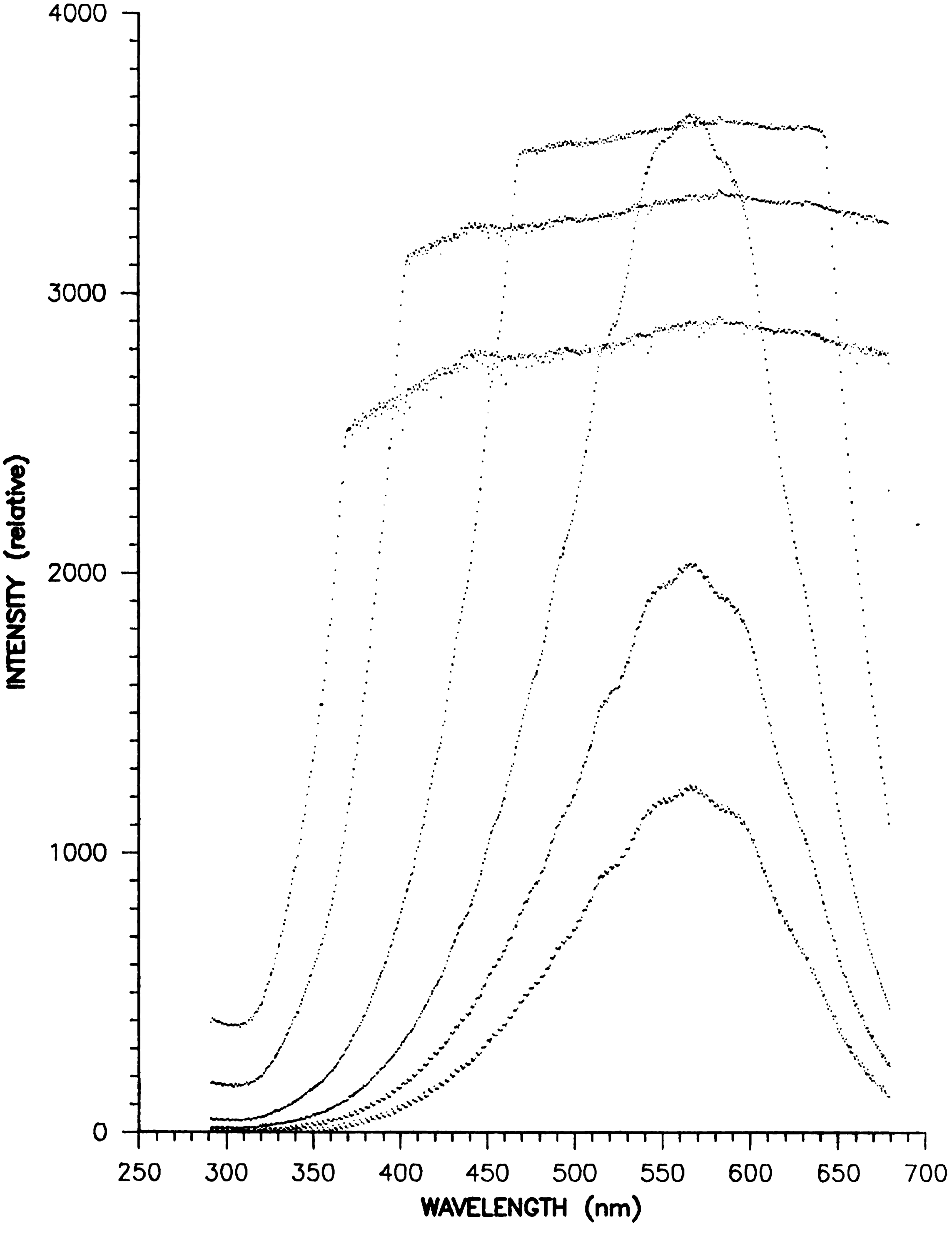


Fig. 5.7 Q. I. lamp spectra with 500 nm cutoff filter

subtracted out.

E. Resolution

In multi-channel spectrophotometers the resolution is dependent on several things. The reciprocal linear disperison of the grating (32 nm/mm) is important as is the physical width of the array itself with respect to the spectral region covered. The spectral window is 400 nm, and the detector can be moved in the focal plane of the grating to obtain spectral information between 200 and 1000 nm.

As mentioned previously, the entrance slit is interchangable. The resolution is ultimately limited by the width of the detector. However, with the smallest available slit width (25 μm) the resolution is limited by the slit. This can be seen in Figures 5.8, 5.9 and 5.10. Figure 5.8 is a spectrum of a neon-filled hollow cathode lamp at a slit width of 100 μm. Figures 5.9 and 5.10 were obtained with slit widths of 50 and 25 μm respectively. Resolution is calculated using the formula;

$$\frac{\bar{\lambda}}{\Delta\lambda}$$

where baseline resolution is achieved. Calculated resolution values are 75, 88 and 106 as the slit width is decreased. These were calculated using the identified lines on each spectrum. These values can be compared to the slit-limited bandpass values which are calculated using the following formula:

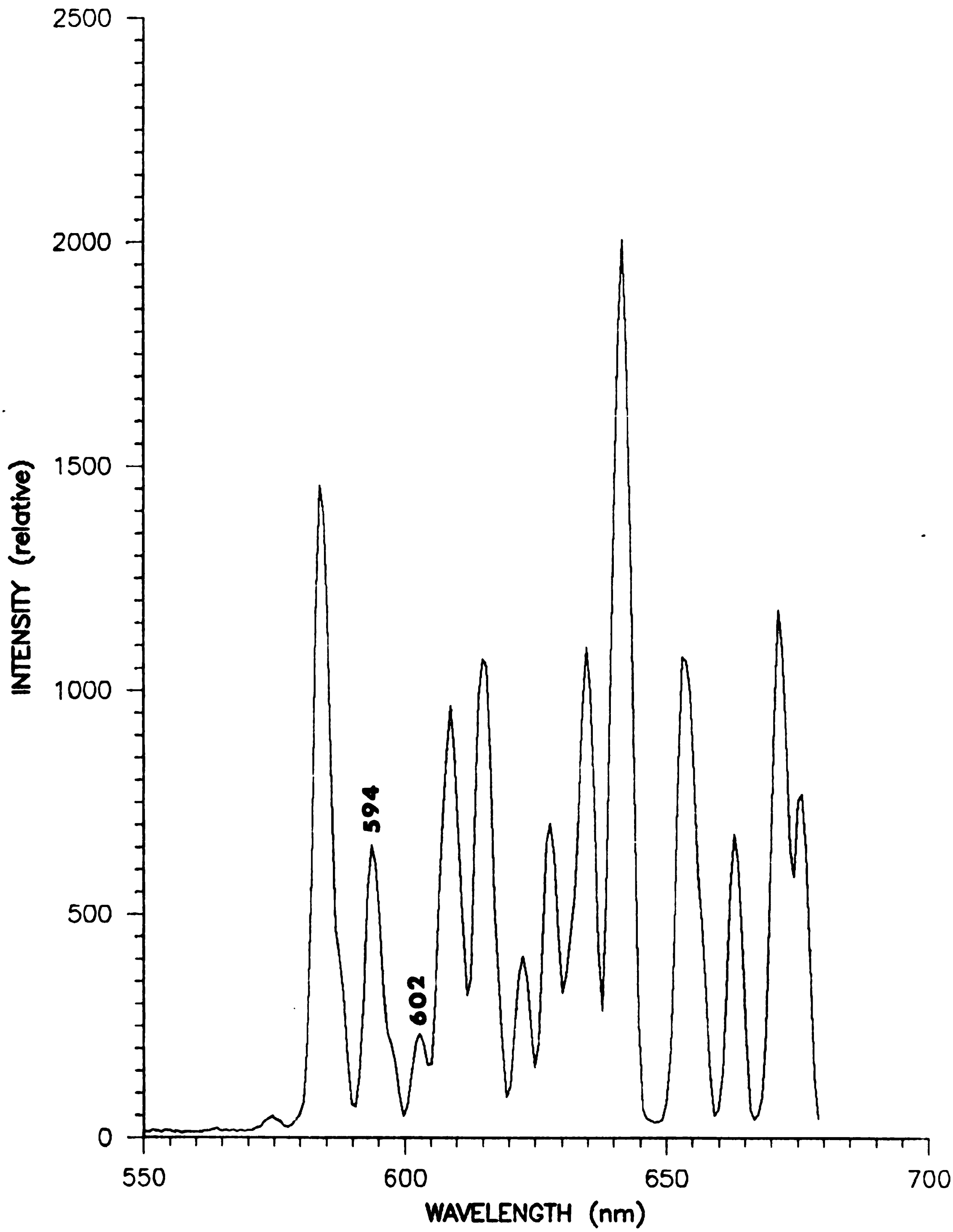


Fig. 5.8 Spectrum of a neon filled hollow cathode with a 100 μm slit width

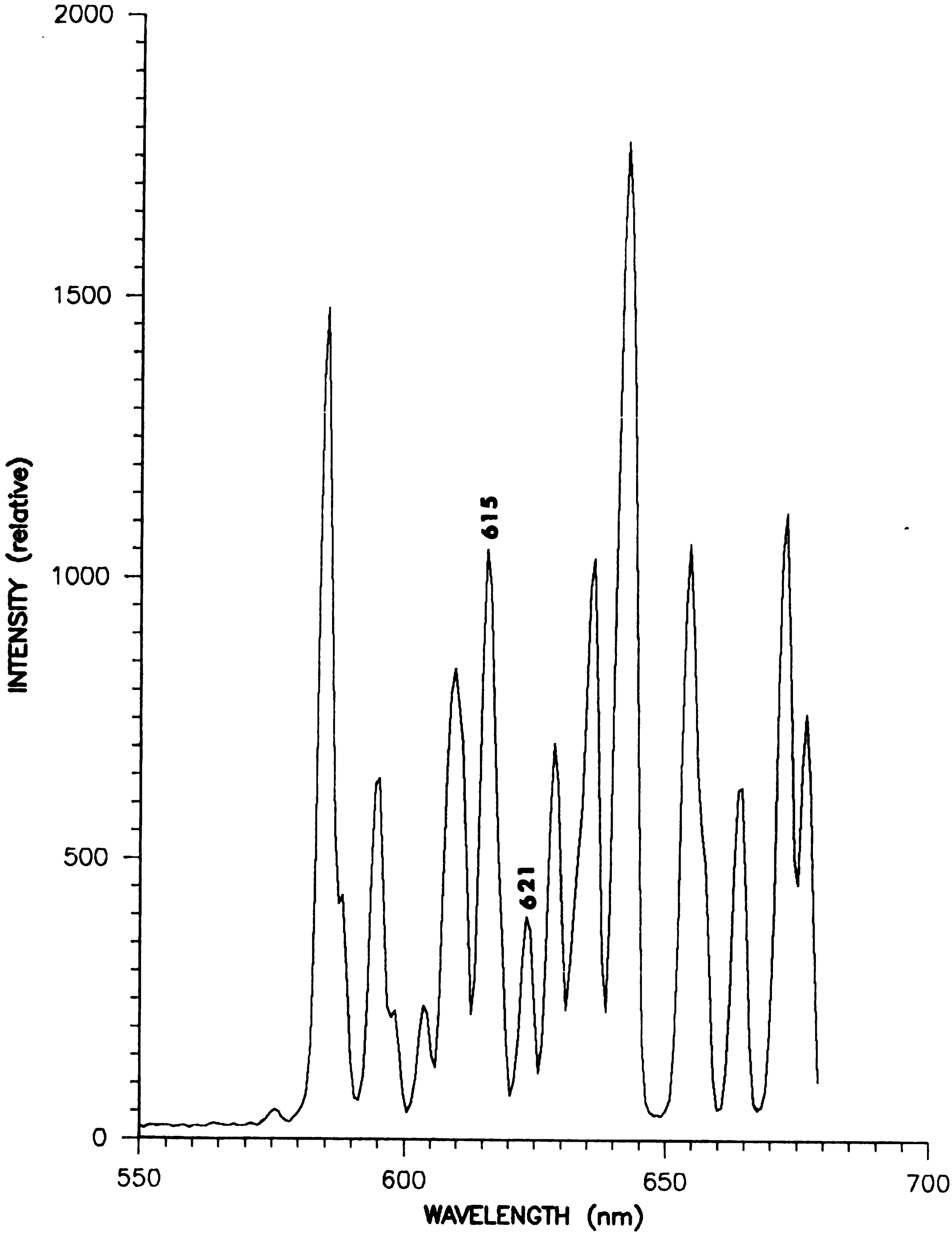


Fig. 5.9 Spectrum of a neon filled hollow cathode with a 50 μm slit width

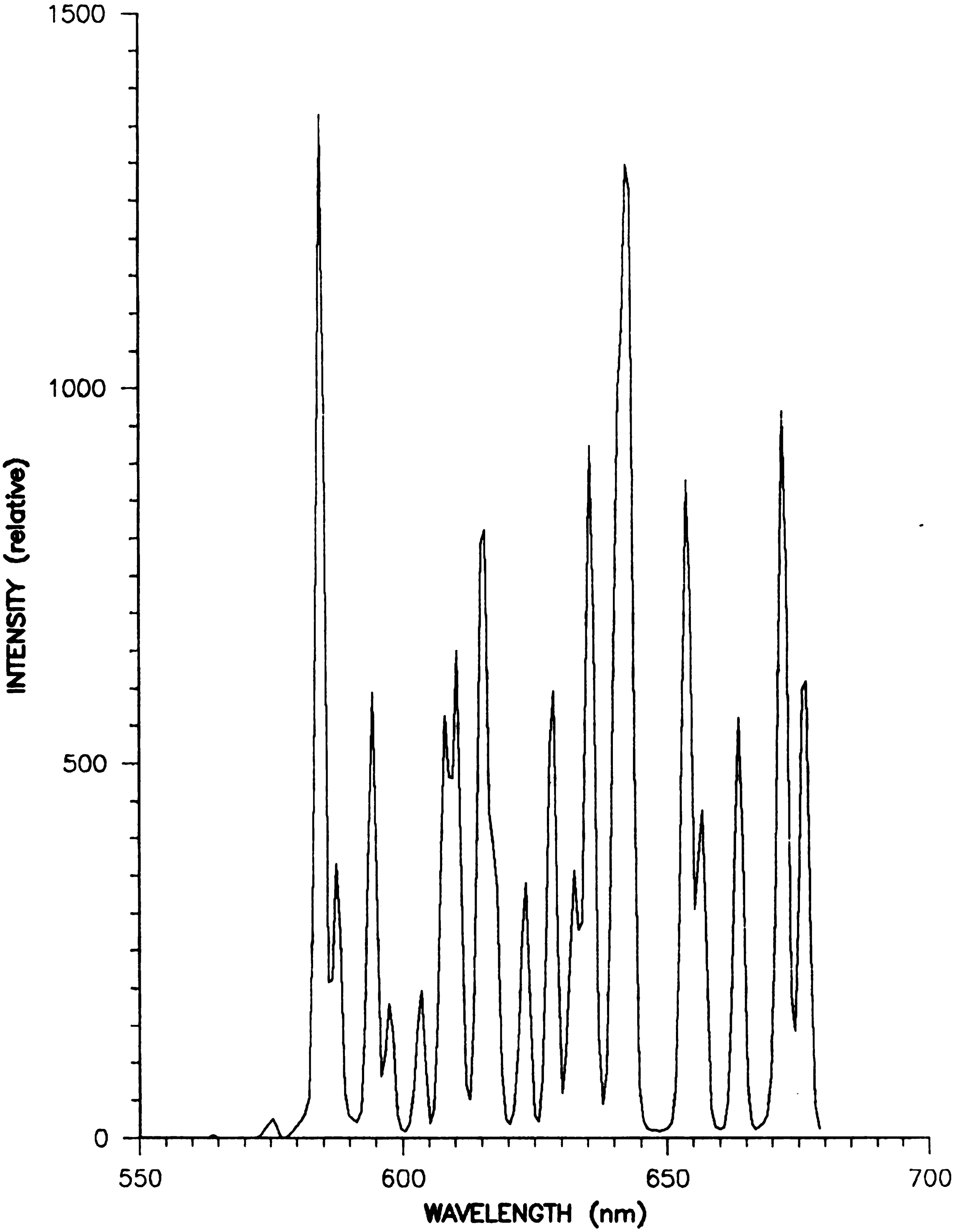


Fig. 5.10 Spectrum of a neon filled hollow cathode with a 25 µm slit width

$$\frac{\bar{\lambda}}{BP}$$

where BP (bandpass) is equal to the slit width multiplied by the reciprocal linear dispersion of the grating. The values calculated are 187, 386 and 795 for the 100, 50 and 25 μm slits respectively.

Further experiments were conducted to see if specific integration times degrade or improve resolution. Figures 5.11, 5.12 and 5.13 are spectra of the same hollow cathode lamp using a 25 μm slit width at 100, 500 and 1000 msec integration times respectively. Peak heights approach saturation as the integration time is lengthened, yet the resolution is not significantly decreased. The only perceptible difference is the loss of baseline resolution in the 1000 msec integration time spectrum. This is due to the broader base of the peaks. Also note the stray light (550 nm) in the longer integration time spectra.

F. Saturation Effects

The spectrum of a neon filled hollow cathode lamp in Figure 5.13 (integration time is 1000 msec and the slit width is 25 μm) illustrates a very important characteristic of linear diode arrays. The array shows essentially no tendency to bloom. Blooming refers to the situation where a strong signal spreads to adjacent sensor elements. Therefore, even minor blooming can seriously degrade resolution and severely limit the use of the integrating

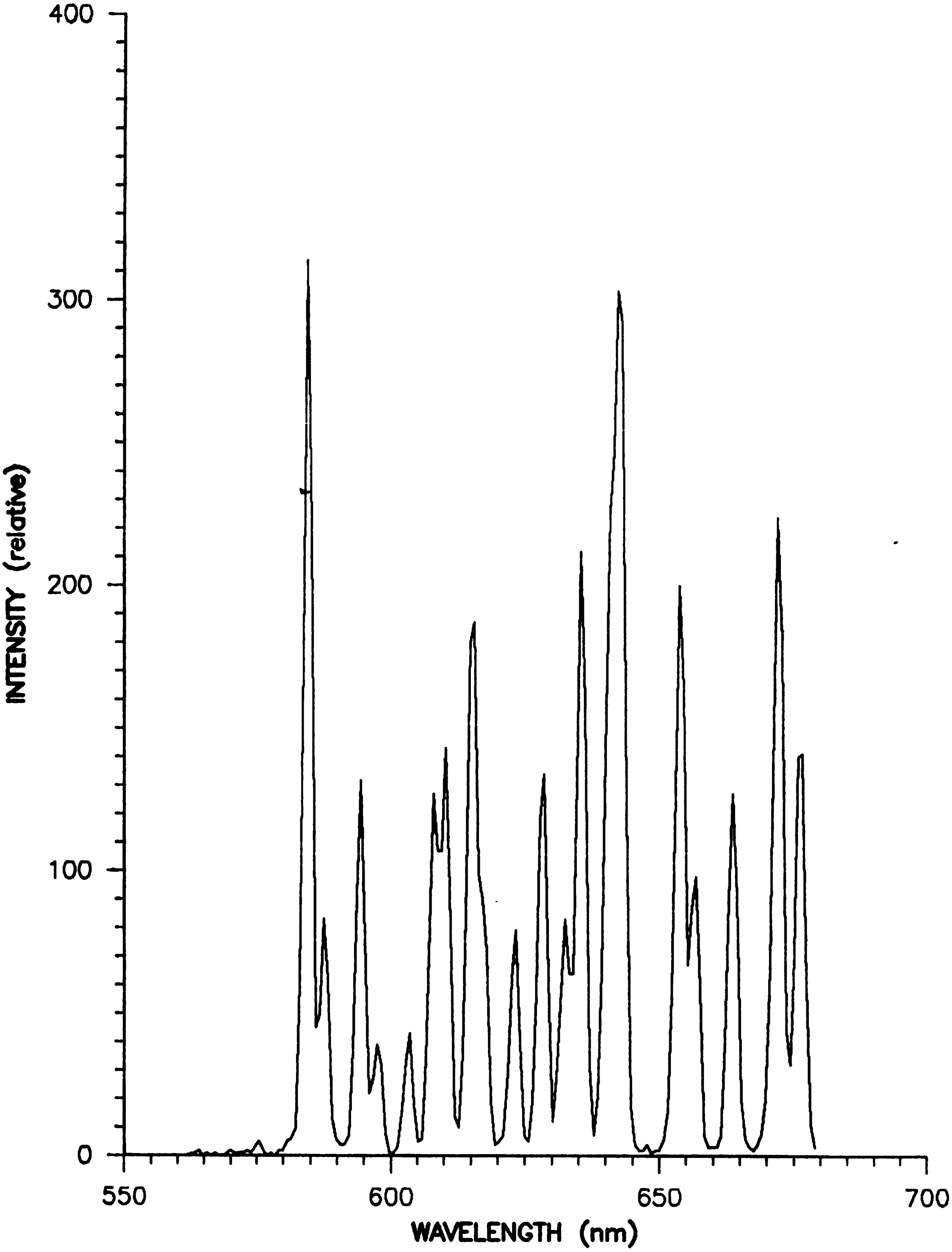


Fig. 5.11 Spectrum of a neon filled hollow cathode at an integration time of 100 msec

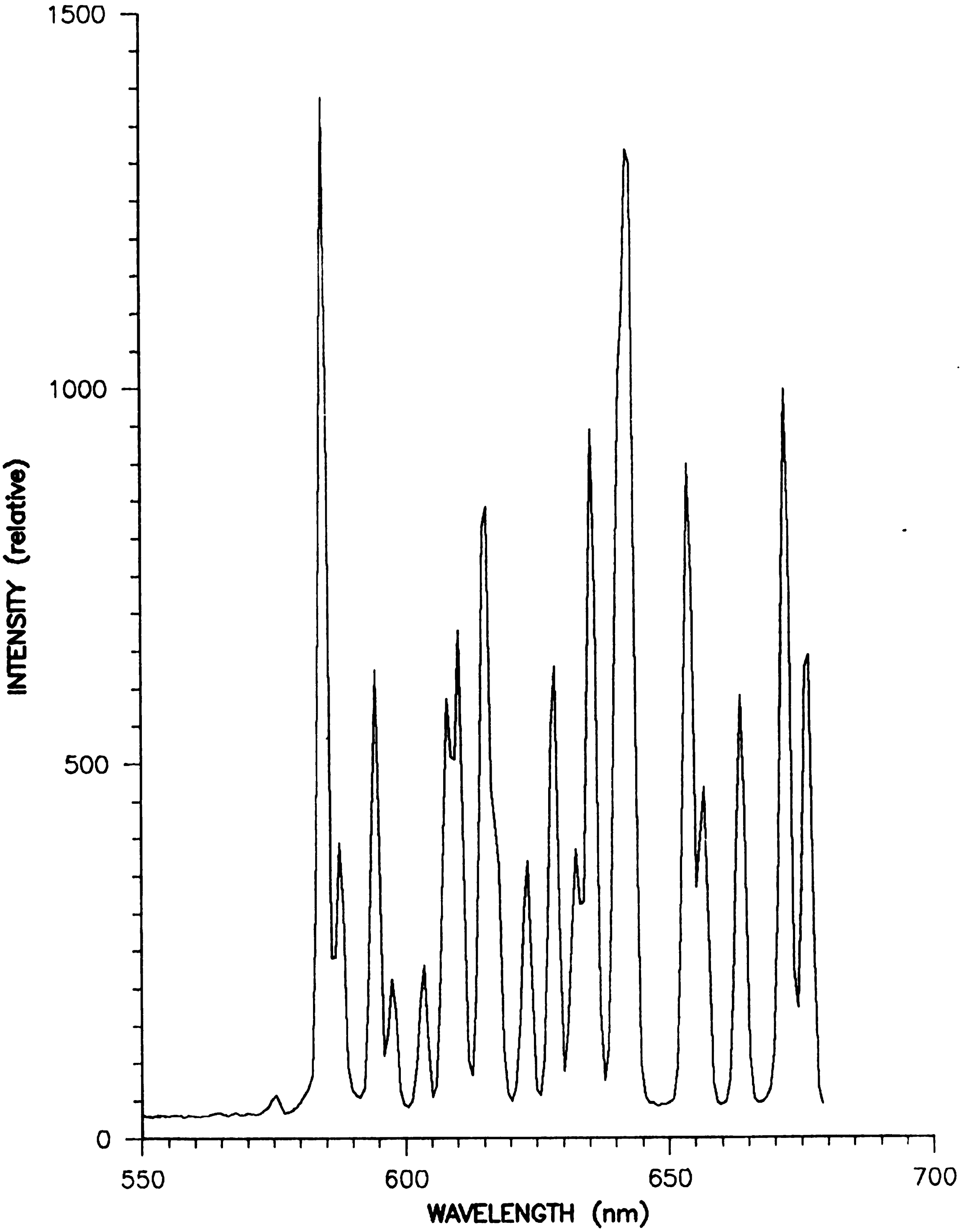


Fig. 5.12 Spectrum of a neon filled hollow cathode at an integration time of 500 msec

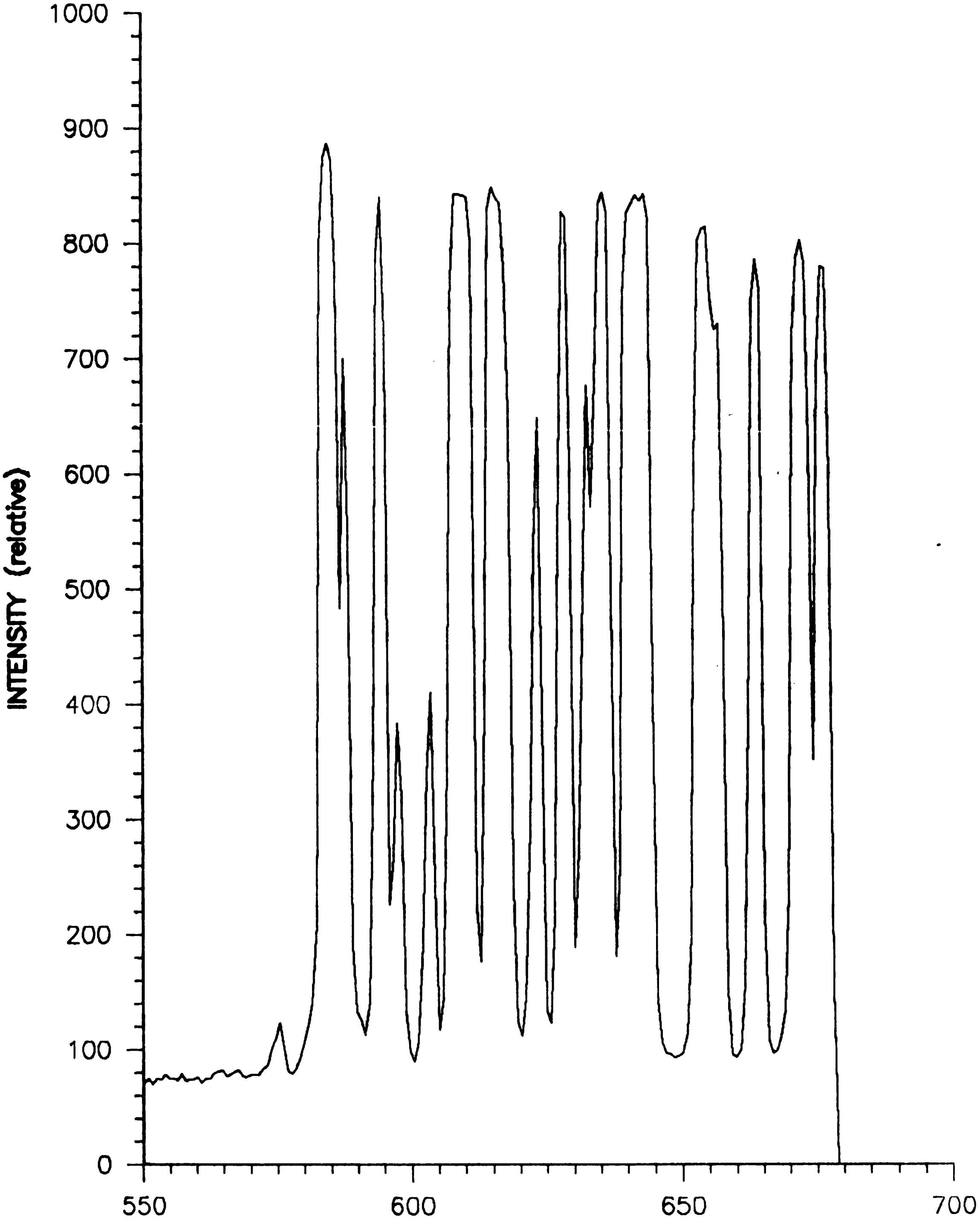


Fig. 5.13 Spectrum of a neon filled hollow cathode at an integration time of 1.0 sec

capability of the array when measuring weak spectral lines in the presence of strong lines. Both silicon vidicons and charge coupled devices have problems with blooming. Thus with photodiode arrays the integration time can be used to increase sensitivity for weak lines, and intense lines will not interfere because of blooming even if they saturate the array. This would be of most benefit if the array were cooled. The only interference will be the broad base of strong lines which are present because of the finite slit width.

Another important characteristic of electronic image sensors is lag. Lag refers to the image carry over from one scan to the next. In most electronic image sensors, the image cannot be completely read out in one scan. With the silicon vidicon for example, only 90% of the image may be erased on a readout cycle which leaves 10% to be carried over to the next scan. In general, lag is undesirable and is especially so if the detector is being used for time resolution studies. Linear diode arrays, however, do not, according to the literature (6), exhibit any lag. We have not tested this electrically since the present interface cannot store consecutive scans.

G. Diode to Diode Sensitivity Variations

This characteristic can be measured using the quartz halogen lamp as described in Chapter IV. A spectrum with dark current subtracted is shown in Figure 5.15 (the integration time is 2.3 msec, the slit width is 100 μm). The variations seen in intensity along the curve (±1%) are sensitivity variations from diode to diode along the array. The shape of the spectrum is not an accurate measure of the overall sensitivity variation across the array as this depends on a number of parameters such as the spectrum of the quartz halogen lamp, the polychromator throughput function and the spectral response of the photodiodes. With a calibrated source an overall spectral sensitivity function could be calculated and used to correct future scans.

H. Dynamic Range

The dynamic range of the array at a fixed integration time is an important characteristic. The dynamic range specified by Reticon is up to 4 orders of magnitude. The source used is the quartz halogen lamp at a 25 msec integration time and a 100 μm slit width. A plot of percent transmittance at 650 nm versus peak amplitude is shown in Figure 5.16. The %T axis was established using a series of neutral density filters of values 0.0132, 0.0525, 0.178, 0.302, 1.02, 2.95, 10.0, 19.5, 29.5, 41.7, 66.1 and 81.3 percent. The peak amplitude was recorded 5 times with each filter in place. The error bars show ±1 standard deviation. A dynamic range of nearly 3 orders of magnitude is observed.

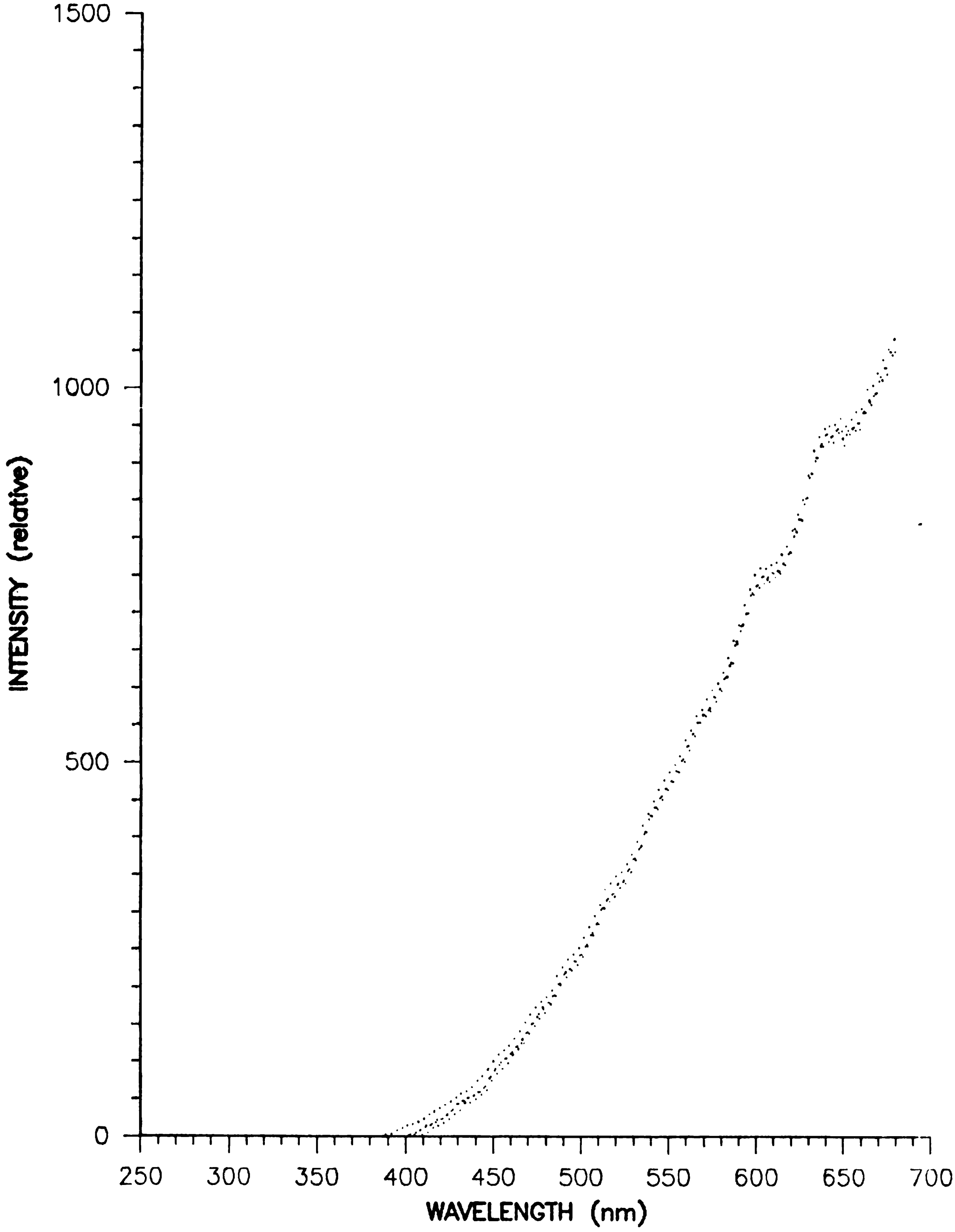


Fig. 5.14 Q. I. lamp spectrum showing sensitivity variations from diode to diode

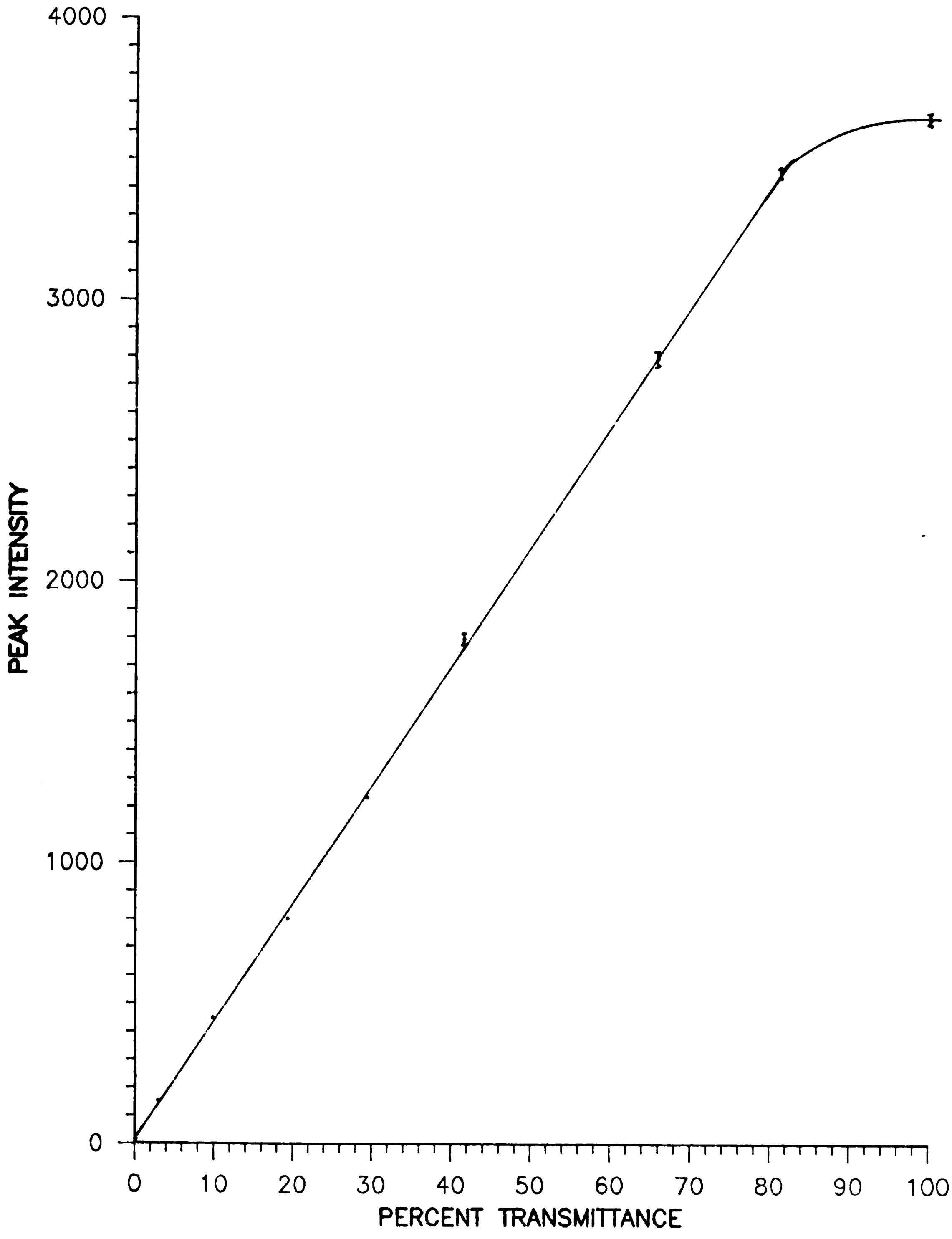


Fig. 5.15 Peak intensity vs. %transmittance

CHAPTER VI

OPERATION AND APPLICATION OF THE INSTRUMENT

A. Routine Operation of the Instrument

The procedure for the operation of the LDA spectrophotometer is described in this section. The process begins by writing the desired subroutines using TECO (Text Editor and Corrector) on the 11/40 minicomputer. The operating system used on the 11/40 is RSX11M V3.2. After creating the subroutine, it must be assembled. This is done by entering the following line:

```
MAC (FILENAME),(FILENAME)=MACROS,GLOBALS,(FILENAME),END
```

The first three letters, MAC, call the program MACRO which assembles the programs listed (after the equals sign) and creates the object and listing files (before the equals sign). A copy of the listing file serves as an invaluable debugging aid. The file MACROS is used to translate 8085 mnemonics into object code and also enter the library entries. GLOBALS contains pointers to all SLOPS routines which can be used by any or all user written routines. After GLOBALS all the user written routines are entered. END marks the end-of-library after the last user written subroutine.

When the MACRO assembly is complete, the object file is stripped of its files-11 format and converted to an unformatted object file. At this point the file is ready to be transferred to a PERSCI formatted floppy disk. This is done using a program called PIPERSCI. The floppy disk is then compatible with the PERSCI dual disk drive on the LDA system. To run the subroutines written on the instrument, the following steps are taken:

1) Reset both the floppy disk and the microprocessor

2) Put the floppy disk in the drive

3) Depress the following keys on the SDK-85: GO, A000, EXEC

The microprocessor will then execute SLOPS which is written in the 2k of PROM which begins at memory address A000 (hexadecimal). The prompt SLOPS> will appear on the terminal screen. To load the user written routines, the disk handler must be called by typing DISK <cr>. The prompt Disk> will appear. Now the user written subroutines can be loaded into RAM by typing L (FILENAME) <cr> and responding Yes to the computer when it asks LOAD PROGRAM?. A control Z character exits from the Disk handler routine. The subroutine is now executed by entering its name followed by a carriage return. After execution is complete, the prompt SLOPS> will again appear.

To obtain an absorbance curve of a sample, three spectra must be recorded; the reference, the dark current and the sample. With the current software this is done by entering the command CDATA, and storing the data when asked to do so.

The data files stored on the floppy disk can be read and transferred to a files-11 formatted file using the program PIPERSCI on the 11/40. The program ABSORB is now executed. This routine asks for the three input filenames and an output filename. The absorbance values at all 512 wavelength channels are sent to the output file in an ASCII data file. The routine MULPLT is the used to plot the data on the terminal or the lineprinter.

B. Application of the Instrument

One application of a linear diode array spectrophotometer is multicomponent analysis. It has been shown that multicomponent absorbance analysis is feasible for a few components in certain cases (28-29). Other applications include those in which the simultaneous recording of all wavelengths or the freedom from mechanical scanning systems are required.

The first experiment conducted was the analysis of a series of potassium dichromate ($K_2Cr_2O_7$) solutions of varying concentration (5 ppm to 250 ppm). At a slit width of 100 μm and an integration time of 50 msec the observed absorbance at 350 nm with the most concentrated solution was 0.18. The expected absorbance is 1.87 at the 350 nm peak. The large error is due to the large amount of stray light present and the low source intensity at 350 nm. The experiment was repeated using a 500 nm cutoff filter (as shown in Fig. 5.7). The results improved somewhat to a maximum absorbance of 0.39 at 350 nm. This is still a very poor result for much the same reasons as before. The experiment was repeated a third time using a 400 nm cutoff filter and the resulting working curve is shown in Figure 6.1. The integration time was increased to 1000 msec because of the very low light levels in this region of the spectrum. The absorbance for the 250 ppm solution is now observed to be near 2.6, well above the expected 1.87. The working curve also has an increasing slope. These results are probably due to the non-linear dark current and light level response of the detector. Upon cooling, it is expected that these results will improve. Also note that the standard deviations are very large which is due to the lower light level at high absorbance values.

Another experiment conducted was the analysis of a series of potassium permanganate ($KMnO_4$) solutions. This experiment was conducted to check the accuracy of the absorbance measurements at a longer wavelength where the source intensity is greater and the stray light is less of a problem. The resulting working curve is shown in Figure 6.2. The integration time is 35 msec and the slit width is 100 μm. The results also show a positive deviation from Beer's law which is due to the non-linear response of the detector. However, at this higher light level, the results are much closer to the expected value of 0.65 for the 42 ppm

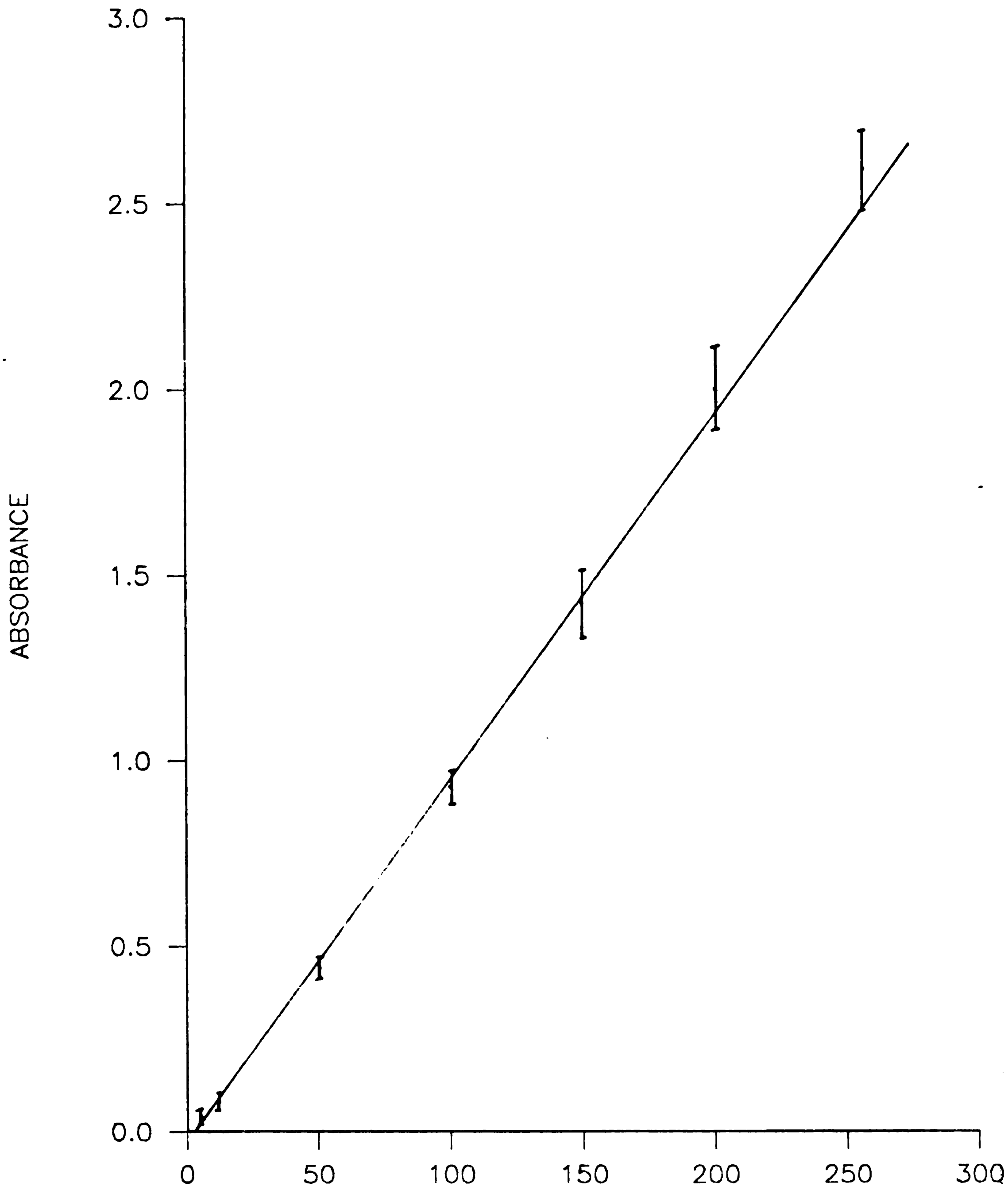


Fig. 6.1 $K_2Cr_2O_7$ working curve

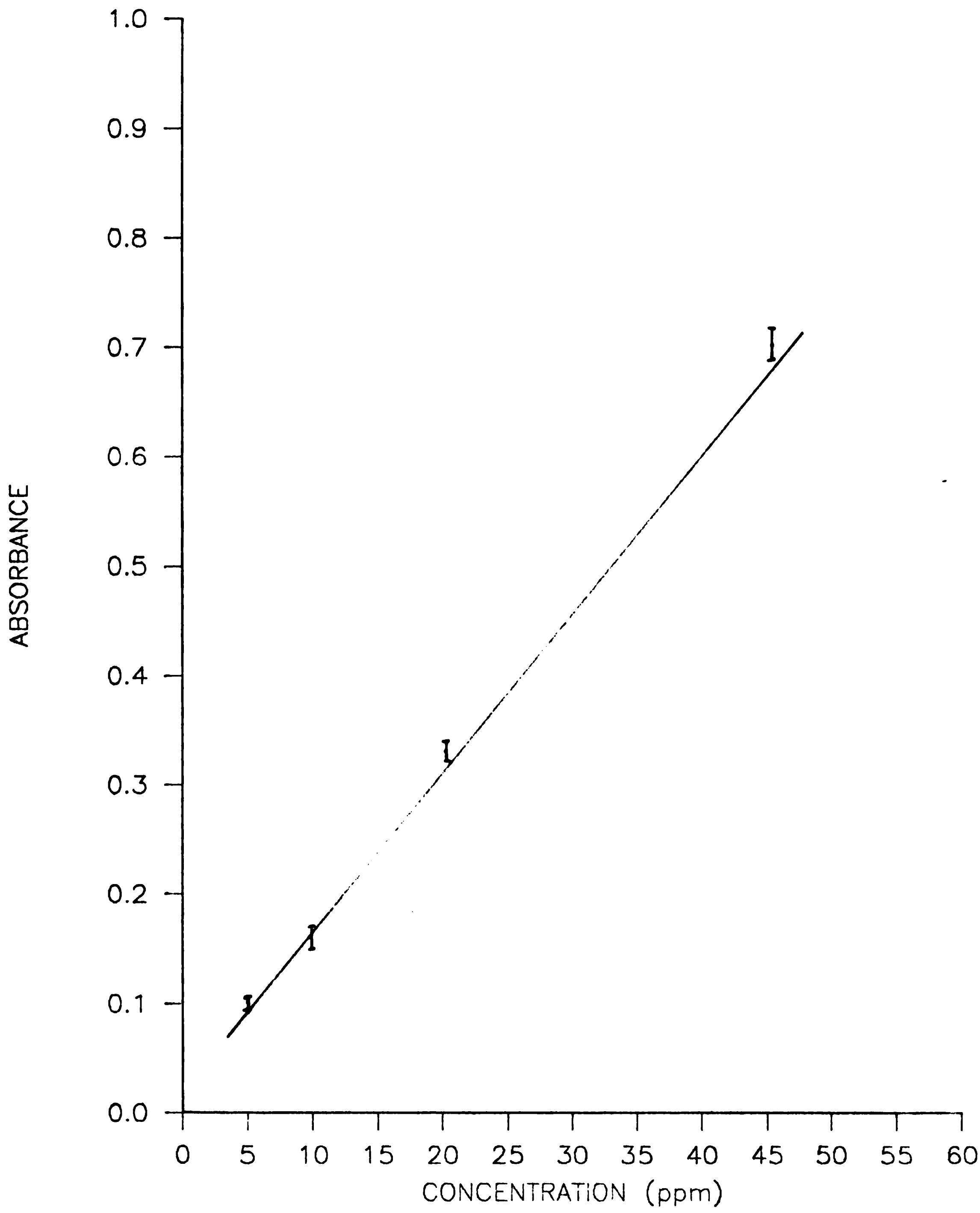


Fig. 6.2 $KMnO_4$ working curve

solution and the standard deviations are much smaller.

The utility of this spectrophotometer for precision absorbance measurements is yet to be determined as these preliminary results show that there are system improvements yet to be made. The first step to be taken is cooling the linear diode array. This is expected to nearly eliminate the non-linearities in the dark current and light level responses. In the ultra-violet region of the spectrum, it is necessary to increase the intensity for any reasonable results to be demonstrated. This can be easily done by a change in the source to a deuteriun lamp or similar type UV source. Further studies using various bandpass and cutoff filters are expected to reduce the stray light so that absorbances near 2.5 can be measured accurately.

APPENDIX A

SELECTED PROGRAM LISTINGS

A. Microcomputer Software

```
;*************************************************************
;                  SCAN.MAC
;
;                  PETER AIELLO
;                  DEPT. OF CHEMISTRY
;                  MICHIGAN STATE UNIVERSITY
;                  EAST LANSING, MI 48824
;*************************************************************
        SUBR    SCAN
SCAN:   PUSH    PSW             ;SAVE REGISTERS
        PUSH    D               ;
        PUSH    H               ;
        MVI     A,1             ;BEGIN SCAN
        STA     CSSCAN          ;AND COLLECT DATA
        LXI     D,DATA+2000     ;INITIALIZE COUNTER
        LXI     H,DATA          ;GET BEGINNING ADDRESS
SCAN1:  INX     H               ;GO TO TOP DATA BITS
        MOV     A,M             ;PUT IT ACCUMULATOR
        ANI     17              ;DELETE TOP FOUR BITS
        MOV     M,A             ;STORE TOP FOUR BITS ONLY
        INX     H               ;INCREMENT TO
                                ;SKIP LOW BITS
        CALL    DCMP            ;CHECK IF END OF DATA
        JNC     SCAN1           ;CONTINUE
        POP     H               ;RESTORE REGISTERS
        POP     D               ;
        POP     PSW             ;
        RET


;*************************************************************
;                  DISPLY.MAC
;
;                  PETER AIELLO
;                  DEPT. OF CHEMISTRY
;                  MICHIGAN STATE UNIVERSITY
;                  EAST LANSING, MI 48824
;*************************************************************
        SUBR    DISPLY
DISPLY: PUSH    PSW             ;SAVE REGISTERS
        PUSH    B
        PUSH    D
        PUSH    H
        LXI     H,DATA          ;GO TO START OF DATA
        MVI     C,12            ;SET BASE TO 12
DISPL1: MOV     B,C             ;INITIALIZE COLUMN COUNTER
DISPL2: MOV     E,M             ;LOW 8 DATA BITS TO E
        INX     H               ;GET NEXT DATA
        MOV     A,M             ;TO ACCUMULATOR
        ANI     017             ;DELETE TOP 4 BITS
```

```
        MOV     D,A             ;FINISH D,E PAIR
                                ;OF ALL 12 BITS
        INX     H               ;TO NEXT DATA POINT
        CALL    CVTEXT          ;CONVERT DATA TO BASE 10
        CALL    PRINTR          ;SEND POINT TO TERMINAL
        LXI     D,DATA+2000     ;CHECK IF END OF DATA
        CALL    DCMP            ;COMPARE TO H,L REGISTERS
        JZ      DISPL4          ;IF END, PRINT END OF DATA
        MVI     A,40            ;READY TO PRINT TAB
        CALL    TTYOUT          ;PRINT IT
        DCR     B               ;COUNT COLUMN AS PRINTED
        JNZ     DISPL2          ;IF <12 COLUMNS, CONTINUE
        CALL    CRLF            ;IF 12, THEN CRLF
        CALL    CHARIN          ;CHECK IF CHARACTER
                                ;IS TYPED
        JZ      DISPL1          ;IF NO CHARACTER,TYPE
                                ;NEXT LINE
        LDA     KBDATA          ;GET CHARACTER
                                ;AND THROW AWAY
DISPL3: CALL    CHARIN          ;CHECK FOR ANOTHER
                                ;CHARACTER
        JZ      DISPL3          ;IF NO CHARACTER,
                                ;KEEP SEARCHING
        LDA     KBDATA          ;GET CHARACTER
                                ;AND TYPE NEXT LINE
        JMP     DISPL1          ;PRINT NEXT DATA
DISPL4: LXI     H,DISPL5        ;PRINT
        CALL    PUSH2           ;END
        CALL    PRINT           ;OF DATA
        POP     H               ;RESTORE REGISTERS
        POP     D
        POP     B
        POP     PSW
        RET                     ;AND GO HOME
DISPL5: .ASCIZ  <CR>/END OF DATA/<CR>


;***********************************************************
;               STORE.MAC
;
;               PETER AIELLO
;               DEPT. OF CHEMISTRY
;               MICHIGAN STATE UNIVERSITY
;               EAST LANSING, MI 48824
;***********************************************************
        SUBR    STORE
STORE:  PUSH    PSW             ;SAVE REGISTERS
        PUSH    B               ;
        PUSH    D               ;
        PUSH    H               ;
        CALL    CLRDSK          ;CLEAR OUT ANYTHING
                                ;DISK IS DOING
        LXI     H,STORE3        ;ASK FOR FILENAME
```

```
        CALL    PUSH2           ;PUSH ONTO USER STACK
        CALL    PRINT           ;AND SEND TO TERMINAL
        MVI     A,123           ;LOAD SAVE COMMAND
        CALL    DDATUM          ;AND SEND TO DISK
        CALL    TTYIN           ;PUT FILENAME ON STACK
        LDA     CHRNUM          ;GET NUMBER OF
                                ;CHARACTERS TO SEND
        LXI     H,CHRBUF        ;AND BUFFER POINTER
        CALL    DDATA           ;AND SEND FILENAME TO DISK
        MVI     A,4             ;LOAD AN EOT
        CALL    DCMD            ;AND SEND TO DISK
        CALL    DINCHR          ;GET A CHARACTER
                                ;FROM THE DISK
        CPI     5               ;IS IT AN ENQ?
        JNZ     STORE5          ;NO, START OVER
        CALL    DINCHR          ;GET NEXT CHARACTER
        CPI     4               ;IS IT AN EOT?
        JNZ     STORE5          ;NO, START OVER
        LXI     D,DATA+2000     ;SET UP END OF
                                ;DATA LOCATOR
        LXI     H,DATA          ;GO TO BEGINNING OF DATA
STORE1: MOV     A,M             ;GET BOTTOM DATA BITS
        INX     H               ;INCREMENT TO HIGH 4 BITS
        CALL    DDATUM          ;SEND TO DISK
        MOV     A,M             ;AND GET THEM
        INX     H               ;GO TO NEXT DATA POINT
        ANI     17              ;DELETE TOP 4 BITS
        CALL    DDATUM          ;SEND TO DISK
        CALL    DCMP            ;CHECK IF END OF DATA
        JNC     STORE1          ;IF NOT LAST PT. CONTINUE
        MVI     A,4             ;LOAD AN EOT TO
                                ;END DATA SET
        CALL    DCMD            ;AND SEND IT
        CALL    DINCHR          ;GET RETURN CHARACTER
        CPI     6               ;IS IT AN ACK?
        JNZ     STORE5          ;NO, START OVER
        CALL    DINCHR          ;GET NEXT CHARACTER
        CPI     4               ;IS IT AN EOT?
        JNZ     STORE5          ;NO, START OVER
        LXI     H,STORE4        ;SEND DATA STORE COMPLETE
        CALL    PUSH2           ;PUSH ON STACK
        CALL    PRINT           ;AND SEND TO TERMINAL
STORE2: POP     H               ;RESTORE REGISTERS
        POP     D               ;
        POP     B               ;
        POP     PSW             ;
        RET                     ;AND GO HOME
STORE3: .ASCIZ  <CR>/ENTER FILENAME: /
STORE4: .ASCIZ  <CR>/DATA STORAGE COMPLETED/<CR>
STORE5: LXI     H,STORE7        ;PRINT ERROR MESSAGE
        CALL    PUSH2           ;PUSH ON STACK
        CALL    PRINT           ;AND SEND TO TERMINAL
STORE6: CALL    DINCHR          ;GET ANY CHARACTER
```

```
                                        ;DISK SENDS
        JC      STORE2                  ;IN CONTROL CHARACTER, END
        CALL    TTYOUT                  ;PRINT OUT CHARACTER
                                        ;MESSAGE
        JMP     STORE6                  ;AND LOOP AROUND
STORE7: .ASCIZ  <CR>/THERE HAS BEEN AN ERROR!/<CR>


;************************************************************
;               CDATA.MAC
;
;               PETER AIELLO
;               DEPT. OF CHEMISTRY
;               MICHIGAN STATE UNIVERSITY
;               EAST LANSING, MI 48824
;************************************************************
        SUBR    CDATA
CDATA:  PUSH    H                       ;SAVE REGISTERS
        PUSH    PSW                     ;
        LXI     H,CDATA2                ;SEND INSTRUCTIONS
        CALL    PUSH2                   ;PUSH ON STACK
        CALL    PRINT                   ;SEND TO TERMINAL
        CALL    SCAN                    ;COLLECT DATA
        CALL    DISPLY                  ;DISPLY DATA ON TERMINAL
        LXI     H,CDATA3                ;ASK TO STORE?
        CALL    PUSH2                   ;PUSH ON STACK
        CALL    PRINT                   ;SEND TO TERMINAL
        CALL    TTYIN                   ;GET USER ANSWER
        LDA     CHRBUF                  ;AND LOAD ACCUMULATOR
        CPI     131                     ;YES?
        JZ      CDATA4                  ;IF SO,CALL STORE
CDATA1: POP     PSW                     ;RESTORE REGISTERS
        POP     H                       ;
        RET                             ;AND GO HOME
CDATA2: .ASCII  <CR>/THIS PROGRAM IS NOW COLLECTING DATA/<CR>
        .ASCII  /AND WILL DISPLAY DATA ON THE TERMINAL/<CR>
        .ASCIZ  /ANY KEY HALTS AND RESTARTS THE DISPLAY/<CR>
CDATA3: .ASCIZ  <CR>/DO YOU WANT TO STORE THIS DATA? [Y,N]: /
CDATA4: CALL    STORE                   ;STORE THE DATA
        JMP     CDATA1                  ;AND END ROUTINE


;************************************************************
;               SIGAVG.MAC
;
;               PETER AIELLO
;               DEPT. OF CHEMISTRY
;               MICHIGAN STATE UNIVERSITY
;               EAST LANSING, MI 48824
;************************************************************
        SUBR    SIGAVG
SIGAVG: PUSH    PSW                     ;SAVE REGISTERS
        PUSH    B                       ;
```

```
         PUSH    D                 ;
         PUSH    H                 ;
         LXI     D,LDATA+2000      ;PREPARE TO ZERO MEMORY
         MVI     A,0               ;PUT A 0 INTO ACCUMULATOR
         LXI     H,LDATA           ;GET ADDRESS WHERE
                                   ;LOW 16 BITS
SIGAV1:  MOV     M,A               ;AND PUT 0 IN LOWER 8 BITS
         INX     H                 ;NEXT ADDRESS
         CALL    DCMP              ;(D,E)-(H,L)CHECK
                                   ;IF END OF DATA
         JNC     SIGAV1            ;AND CONTINUE
         LXI     H,HDATA           ;GET ADDRESS WHERE
                                   ;HIGH BITS STORED
         LXI     D,HDATA+2000      ;PREPARE CHECK
SIGAV2:  MOV     M,A               ;STORE 0 IN MEMORY
         INX     H                 ;NEXT ADDRESS
         CALL    DCMP              ;CHECK IF END
                                   ;OF DATA BLOCK
         JNC     SIGAV2            ;AND CONTINUE
SIGAV3:  LXI     H,SIGAV8          ;ASK FOR NUMBER OF SCANS
         CALL    PUSH2             ;PUSH ON STACK
         CALL    PRINT             ;AND SEND TO TERMINAL
         MVI     A,12              ;SET BASE TO 10
         STA     RADIX             ;AND STORE
         CALL    TTYIN             ;GET RESPONSE FROM USER
         CALL    BRKDWN            ;
         CALL    NUMBER            ;PUT IN D,E REGISTER PAIR
         JC      SIGAV9            ;IF ERROR, SEND MESSAGE
         XCHG                      ;PUT IT H,L REGISTER PAIR
         SHLD    NSCAN             ;STORE IN MEMORY
                                   ;FOR DIVISION
         SHLD    COUNT             ;STORE IN MEMORY FOR COUNT
SIGAV4:  CALL    SCAN              ;RUN A SCAN AND
                                   ;STORE IN DATA
         LXI     B,0               ;INDEX AND COUNTER
                                   ;INITIALIZE
SIGAV5:  LXI     H,DATA            ;GO TO STARTING
                                   ;ADDRESS OF DATA
         DAD     B                 ;INCREMENT TO
                                   ;ADDR. OF DATA TO AVG.
         MOV     E,M               ;LOW 8 BITS TO E
         INX     H                 ;GET NEXT BITS
         MOV     D,M               ;AND PUT IN D
         PUSH    D                 ;AND SAVE LOW
                                   ;16 BITS ON STACK
         LXI     H,LDATA           ;GET ADDRESS FOR
                                   ;STORING LOW 16 BITS
         DAD     B                 ;ADDRESS OF AVERAGED DATA
         MOV     E,M               ;LOW 8 BITS TO E
         INX     H                 ;GET NEXT 8 BITS
         MOV     D,M               ;AVERAGED DATA NOW IN D,E
         XTHL                      ;H,L=DATA  STACK=ADDRESS
         DAD     D                 ;AVERAGE IN H,L
```

```
        XCHG                    ;AVERAGE IN D,E
        POP     H               ;ADDRESS IN H,L
        MOV     M,D             ;STORE ANSWER
                                ;(8 HIGH BITS)
        DCX     H               ;NEXT ADDRESS OF LOW BITS
        MOV     M,E             ;AND STORE
        JNC     SIGAV6          ;IF CARRY,FILL
                                ;NEXT 16 BITS
        LXI     H,HDATA         ;GET HIGH 16 BITS ADDRESS
        DAD     B               ;AND POINT TO CORRECT DATA
        MOV     E,M             ;GET LOW 8 BITS TO E
        INX     H               ;INCREMENT ADDRESS
        MOV     D,M             ;HIGH 8 BITS TO D
        INX     D               ;INCREMENT REGISTER
                                ;PAIR D,E
        MOV     M,D             ;STORE ADDED DATA BITS
        DCX     H               ;
        MOV     M,E             ;AND NOW THE LOW 8 BITS
SIGAV6: INX     B               ;INCREMENT COUNTER
        INX     B               ;TWICE(WORDS)
        MOV     D,B             ;MOVE B AND C
        MOV     E,C             ;TO D AND E
        LXI     H,2000          ;CHECK IF LAST
                                ;DATA PT. AVERAGED
        CALL    DCMP            ;(D,E)-(H,L)
        JC      SIGAV5          ;GO TO AVERAGE NEXT POINT
        LHLD    COUNT           ;GET NUMBER OF
                                ;DESIRED SCANS
        DCX     H               ;DECREMENT BY 1 SCAN
        SHLD    COUNT           ;STORE DECREMENTED COUNTER
        LXI     D,0             ;CHECK IF LAST SCAN
        CALL    DCMP            ;AND SET ZERO FLAG
        JC      SIGAV4          ;IF NOT LAST SCAN,
                                ;CONTINUE
        LXI     B,0             ;INITIALZE COUNTER
                                ;TO STORE
SIGAV7: PUSH    B               ;AND SAVE ON STACK
        LXI     H,LDATA         ;GET ADDRESS OF
                                ;LOW 16 AVG'D BITS
        DAD     B               ;ADD POINTER
        MOV     E,M             ;LOW 8 BITS TO E
        INX     H               ;GET HIGH BITS
        MOV     D,M             ;INTO THE D REGISTER
        PUSH    D               ;STORE LOW 16 BITS
        LXI     H,HDATA         ;GET HIGH 16 BITS
        DAD     B               ;ADD POINTER
        MOV     E,M             ;NEXT 8 BITS TO E
        INX     H               ;GET HIGH 8 BITS
        MOV     D,M             ;IN TO THE D REGISTER
        LHLD    NSCAN           ;GET NUMBER OF SCANS
        XCHG                    ;H,L=16 HIGHEST
                                ;BITS,D,E=DIVISOR
        POP     B               ;B,C=LOW 16 BITS
```

```
         CALL     DDIV              ;DIVIDE AND PUT
                                    ;ANSWER IN B,C
         POP      D                 ;POINTER INTO
                                    ;D,E REGISTER PAIR
         LXI      H,DATA            ;GET ADDRESS OF
                                    ;STORAGE SPACE
         DAD      D                 ;ADD POINTER
         MOV      M,C               ;STORE LOW 8 BITS
         INX      H                 ;INCREMENT ADDRESS
         MOV      M,B               ;STORE HIGH 8 BITS
         INX      D                 ;INCREMENT POINTER
         INX      D                 ;TWICE (WORDS)
         LXI      H,2000            ;INITIALIZE CHECK
         CALL     DCMP              ;CHECK
         MOV      B,D               ;RESTORE POINTER
         MOV      C,E               ;TO B,C REGISTER PAIR
         JC       SIGAV7            ;IF NOT LAST POINT
                                    ;CONTINUE
         CALL     DISPLY            ;DISPLAY DATA ON TERMINAL
         LXI      H,SIGA11          ;ASK TO STORE?
         CALL     PUSH2             ;PUSH ON STACK
         CALL     PRINT             ;SEND TO TERMINAL
         CALL     TTYIN             ;GET USER ANSWER
         LDA      CHRBUF            ;AND LOAD TO ACCUMULATOR
         CPI      131               ;YES?
         JZ       SIGA12            ;IF SO, CALL STORE
SIGA13:  POP      H                 ;RESTORE REGISTERS
         POP      D                 ;
         POP      B                 ;
         POP      PSW               ;
         RET                        ;AND GO HOME
SIGAV8:  .ASCII   <CR>/THIS PROGRAM WILL TAKE THE NUMBER OF
                    SCANS ENTERED,/
         .ASCII   <CR>/AVERAGE, STORE, AND DISPLAY THEM./
         .ASCII   <CR>/   NOTE:  ANY KEY HALTS AND RESTARTS
                    THE DISPLAY/
         .ASCIZ   <CR>/PLEASE ENTER THE NUMBER OF SCANS TO
                    BE AVERAGED: /
SIGAV9:  LXI      H,SIGA10          ;PRINT AN ERROR MESSAGE
         CALL     PUSH2             ;PUSH ON STACK
         CALL     PRINT             ;AND SEND TO TERMINAL
         JMP      SIGAV3            ;RETURN TO ASK AGAIN
SIGA10:  .ASCII   <CR>/YOU HAVE MADE AN ERROR!/
                  .ASCIZ  <CR>/PLEASE ENTER A DECIMAL NUMBER
                    LESS THAT 64K/<CR>
SIGA11:  .ASCIZ   <CR>/DO YOU WANT TO STORE THIS DATA? [Y,N]: /
SIGA12:  CALL     STORE             ;STORE THE DATA ON FLOPPY
         JMP      SIGA13            ;AND END ROUTINE
```

```
;*************************************************************
;               INT.MAC
;
;               PETER AIELLO
;               DEPT. OF CHEMISTRY
;               MICHIGAN STATE UNIVERSITY
;               EAST LANSING, MI 48824
;*************************************************************
        SUBR    INT
INT:    PUSH    PSW                 ;SAVE REGISTERS
        PUSH    B                   ;
        PUSH    D                   ;
        PUSH    H                   ;
        LXI     H,INT3              ;ASK FOR DESIRED
                                    ;INTEGRATION TIME
INT1:   CALL    PUSH2               ;PUSH ON STACK
        CALL    PRINT               ;AND SEND TO TERMINAL
        CALL    TTYIN               ;GET RESPONSE
                                    ;FROM OPERATOR
        CALL    BRKDWN              ;AND BREAK INTO WORDS
        MVI     A,12                ;SET BASE TO DECIMAL
        STA     RADIX               ;FOR USER INPUT
        CALL    NUMBER              ;PUT RESPONSE
                                    ;IN D,E REG. PAIR
        JC      INT4                ;IF ERROR, SEND MESSAGE
        MVI     A,INTIME            ;MULTIPLY INT.
                                    ;TIME BY LINEAR SLOPE
        CALL    DMULT               ; TO INTERPOLATE
                                    ;FINAL ANSWER
        MVI     A,100.              ; SLOPE=INTIME/100
        CALL    DIV                 ; COMPLETE
        MOV     A,D                 ;CHECK IF ERROR
        CPI     0                   ;IF SO, SEND ERROR MESSAGE
        JNZ     INT4                ; COMPLETE
        MOV     A,E                 ;LOAD ACCUMULATOR
                                    ;WITH PROPER BITS
        CMA                         ;COMPLEMENT ACCUMULATOR
        STA     INTSET              ;SEND 8 BITS TO
                                    ;INTEGRATE LATCH
        POP     H                   ;RESTORE REGISTERS
        POP     D                   ;
        POP     B                   ;
        POP     PSW                 ;
        RET                         ;AND GO HOME
INT2:   .ASCII  <CR>/YOU HAVE ENTERED AN ILLEGAL NUMBER,
                 PLEASE TRY AGAIN! /
INT3:   .ASCIZ  <CR>/PLEASE ENTER DESIRED INTEGRATION TIME
                 IN MSEC: /
INT4:   LXI     H,INT2              ;AND SEND ERROR MESSAGE
        JMP     INT1                ;AND RETURN
```

```
;*************************************************************
;               PLOT.MAC
;
;               PETER AIELLO
;               DEPT. OF CHEMISTRY
;               MICHIGAN STATE UNIVERSITY
;               EAST LANSING, MI 48824
;*************************************************************
        SUBR    PLOT
PLOT:   PUSH    PSW             ;SAVE REGISTERS
        PUSH    B               ;
        PUSH    D               ;
        PUSH    H               ;
        LXI     D,4080.         ;GET THRESHOLD
                                ;FOR VERTICAL AXIS
PLOT1:  LXI     H,0             ;CHECK THRESHOLD
        CALL    DCMP            ;TO SEE IF BELOW 0
        JM      PLOT5           ;IF SO END EXECUTION
        LXI     H,170.          ;PREPARE TO DECREMENT
                                ;THRESHOLD
        XCHG                    ;EXCHANGE D,E AND H,L
        CALL    DSUB            ;SUBTRACT 170
                                ;FROM THRESHOLD
        XCHG                    ;EXCHANGE D,E AND H,L
        MVI     A,40            ;INITIALIZE AND
                                ;STORE A SPACE
        STA     BSTOR           ;IN RAM
        LXI     H,DATA          ;GET DATA ADDRESS
        SHLD    ASTOR           ;AND STORE IN RAM
PLOT2:  MVI     B,8.            ;INITIALIZE X-AXIS COUNTER
PLOT3:  LHLD    ASTOR           ;GET CURRENT ADDRESS
        PUSH    D               ;SAVE THRESHOLD
        MOV     E,M             ;GET DATA INTO D,E
        INX     H               ;INCREMENT ADDRESS
                                ;TO HIGH 4 BITS
        MOV     D,M             ;GET HIGH BITS INTO D
        INX     H               ;SET ADDRESS TO
                                ;NEXT DATA POINT
        SHLD    ASTOR           ;STORE ADDRESS IN RAM
        POP     H               ;THRESHOLD TO
                                ;H,L REG. PAIR
        CALL    DCMP            ;COMPARE DATA TO THRESHOLD
        JM      PLOT4           ;IF NOT GREATER, NO POINT
        MVI     A,52            ;IF GREATER, SET
                                ;FLAG TO PLOT POINT
        STA     BSTOR           ;AND STORE IN RAM
PLOT4:  XCHG                    ;EXCHANGE D,E AND H,L
        DCR     B               ;DECREMENT X-AXIS COUNTER
        JNZ     PLOT3           ;IF NOT 0, CONTINUE
        LDA     BSTOR           ;CHECK FLAG TO PLOT POINT
        CALL    TTYOUT          ;AND THEN TO TERMINAL
        MVI     A,40            ;SEND SPACE
        STA     BSTOR           ;STORE THE CHARACTER
```

```
         PUSH    D              ;SAVE THE THRESHOLD
         LXI     D,DATA+2000    ;CHECK IF ALL DATA SCANNED
         LHLD    ASTOR          ;GET CURRENT ADDRESS
         CALL    DSUB           ;CHECK
         POP     D              ;RESTORE THRESHOLD
         JM      PLOT2          ;IF NOT COMPLETE, CONTINUE
         CALL    CRLF           ;IF COMPLETE,
                                ;SEND CRLF AND
         JMP     PLOT1          ;LOWER THRESHOLD
                                ;AND CONTINUE
PLOT5:   POP     H              ;RESTORE REGISTERS
         POP     D
         POP     B
         POP     PSW
         RET                    ;AND GO HOME
```

B. Minicomputer Software

```
C**********
C
C         CONVERT.FTN
C         THIS ROUTINE CONVERTS THE UNFORMATTED BINARY FILES
C         TO C.G.ENKE AND CO.  STANDARD DATA FILES
C
C         P. AIELLO  9/5/79
C
C**********
C
C       DECLARE VARIABLES
          BYTE INFILE(32),OTFILE(32)
          INTEGER DATA(512),IERR,ITABLE(96)
          EQUIVALENCE(INFILE(1),TEST)
          DATA EX/'EXIT'/
C       ASK FOR INPUT FILE NAME
5         WRITE(5,900)
900       FORMAT('$INPUT FILE:  ')
C       READ THE INPUT FILE NAME
          READ(5,905)INFILE
905       FORMAT(32A1)
C       CHECK IF USER HAS TYPED EXIT TO LEAVE ROUTINE
          IF (TEST .EQ. EX)GO TO 50
C       CALL UNFORM TO OPEN FILE TO READ ENABLE
          CALL UNFORM(0,IERR,ITABLE,INFILE,0,1,3)
C       CHECK FOR ERRORS
          IF(IERR .NE. 0) GO TO 10
C       ASK FOR OUTPUT FILENAME
          WRITE(5,910)
910       FORMAT('$OUTPUT FILE:  ')
C       READ THE OUTPUT FILE NAME
          READ(5,905)OTFILE
C       READ IN THE DATA USING UNFORM
          CALL UNFORM(1,IERR,ITABLE,DATA,1024)
C       CHECK FOR ERRORS
          IF(IERR .NE. 0) GO TO 20
C       OPEN OUTPUT FILE USING ASSIGN
          CALL ASSIGN(4,OTFILE,32)
C       SEND HEADER TO OUTPUT FILE
          WRITE(4,915)OTFILE
915       FORMAT('HD ',32A1)
C       SEND NUMBER OF DATA POINTS TO OUTPUT FILE
          WRITE(4,920)
920       FORMAT('NU      512 ')
C       READ IN THE DATA FROM INPUT FILE TO OUTPUT FILE
          DO 100 I=1,512
          A=DATA(I)
          B=FLOAT(513-I)
          WRITE(4,925)B,A
925       FORMAT('RD ',2G15.7)
```

```
100     CONTINUE
C     NOW CLOSE THE STILL OPEN INPUT FILE USING UNFORM
        CALL UNFORM(2,IERR,ITABLE)
C     CHECK FOR ERRORS
        IF (IERR .NE. 0) GO TO 30
C     SEND END OF DATA MARKER TO OUTPUT FILE
        WRITE(4,930)
930     FORMAT('ED ')
        WRITE(5,935)
935     FORMAT(' DATA CONVERSION COMPLETE ')
        GO TO 40
10      WRITE(5,940)
940     FORMAT(' INPUT FILE NOT FOUND ')
        GO TO 50
20      WRITE(5,945)
945     FORMAT(' DATA CANNOT BE READ OUT OF INPUT FILE ')
        GO TO 50
30      WRITE(5,950)
950     FORMAT(' INPUT FILE CANNOT BE CLOSED ')
C     AND CLOSE THE OUTPUT FILE
40      CALL CLOSE(4)
C     RETURN TO BEGINNING TO CONVERT ANOTHER FILE
        GO TO 5
50      CALL EXIT
        END


C**********
C
C       SUBDAR.FTN
C       THIS ROUTINE SUBTRACTS A SCAN FROM ANOTHER
C       AND OUTPUTS IN A UNFORMATTED BINARY DATA FILE
C
C       P. AIELLO 12/19/79
C
C**********
C
C     DECLARE VARIABLES
        BYTE INFILE(32),OTFILE(32)
        INTEGER DATA(512),SUM(512),IERR,ITABLE(96)
        EQUIVALENCE(INFILE(1),TEST)
        DATA EX/'EXIT'/
C     GET LIGHT SCAN FILE NAME
10      WRITE(5,900)
900     FORMAT('$LIGHT SCAN FILENAME: ')
        READ(5,905)INFILE
905     FORMAT(32A1)
C     CHECK IF THE USER HAS TYPED EXIT TO LEAVE ROUTINE
        IF(TEST .EQ. EX)GO TO 80
C     OPEN UP THE INPUT FILE WITH UNFORM
        CALL UNFORM(0,IERR,ITABLE,INFILE,0,1,3)
C     CHECK FOR ERRORS
```

```
          IF(IERR .NE. 0)GO TO 15
          IF(IERR .EQ. 0)GO TO 45
15        WRITE(5,910)
910       FORMAT( 'INPUT FILE NOT FOUND ')
          GO TO 10
C       READ IN THE DATA AND PUT IN ARRAY SUM
45        CALL UNFORM(1,IERR,ITABLE,DATA,1024)
C       CHECK FOR ERRORS
          IF(IERR .NE. 0)GO TO 50
          DO 20 K=1,512
          SUM(K)=DATA(K)
20        CONTINUE
C       CLOSE INPUT FILE USING UNFORM
          CALL UNFORM(2,IERR,ITABLE)
C       GET DARK SCAN FILE NAME
25        WRITE(5,915)
915       FORMAT('$DARK SCAN FILENAME: ')
          READ(5,905)INFILE
C       OPEN UP THE DARK FILE WITH UNFORM
          CALL UNFORM(0,IERR,ITABLE,INFILE,0,1,3)
C       CHECK FOR ERRORS
          IF(IERR .NE. 0)GO TO 30
          IF(IERR .EQ. 0)GO TO 55
30        WRITE(5,920)
920       FORMAT( 'DARK SCAN FILE NOT FOUND ')
          GO TO 25
C       READ IN THE DATA AND SUBTRACT FROM THE ARRAY SUM
55        CALL UNFORM(1,IERR,ITABLE,DATA,1024)
C       CHECK FOR ERRORS
          IF(IERR .NE. 0)GO TO 60
          DO 35 K=1,512
          SUM(K)=SUM(K)-DATA(K)
35        CONTINUE
C       CLOSE INPUT FILE
          CALL UNFORM(2,IERR,ITABLE)
C       GET OUTPUT FILENAME
40        WRITE(5,925)
925       FORMAT('$OUTPUT FILENAME: ')
          READ(5,930)OTFILE
930       FORMAT(32A1)
C       OPEN THE OUTPUT FILE TO WRITE USING UNFORM
          CALL UNFORM(0,IERR,ITABLE,OTFILE,1,1,3)
C       CHECK FOR ERRORS
          IF(IERR .NE. 0)GO TO 65
C       WRITE DATA USING UNFORM
          CALL UNFORM(1,IERR,ITABLE,SUM,1024)
C       CHECK FOR ERRORS
          IF(IERR .NE. 0)GO TO 70
C       CLOSE THE OUTPUT FILE
          CALL UNFORM(2,IERR,ITABLE)
C       CHECK FOR ERRORS
          IF(IERR .NE. 0)GO TO 75
C       IF THERE ARE NO ERRORS SEND ENDING MESSAGE AND GO HOME
```

```
          WRITE(5,935)
935       FORMAT(' SUBTRACTION COMPLETE ')
C       RETURN TO BEGIN AGAIN
          GO TO 10
50        WRITE(5,940)
940       FORMAT(' DATA CANNOT BE READ OUT OF INPUT FILE ')
          GO TO 80
60        WRITE(5,945)
945       FORMAT(' DATA CANNOT BE READ OUT OF DARK FILE ')
          GO TO 80
65        WRITE(5,950)
950       FORMAT(' OUTPUT FILE ALREADY EXISTS ')
          GO TO 55
70        WRITE(5,955)
955       FORMAT(' DATA CANNOT BE TRANSFERRED TO OUTPUT FILE ')
          GO TO 80
75        WRITE(5,960)
960       FORMAT(' OUTPUT FILE CANNOT BE CLOSED ')
C       AND END THE PROGRAM
80        CALL EXIT
          END


C**********
C
C         ABSORB.FTN
C         THIS ROUTINE CALCULATES THE %TRANSMISSION OF TOTAL
C         LIGHT AT EACH WAVELENGTH AND THEN GETS ABSORBANCE
C         TO OUTPUT  AS AN UNFORMATTED BINARY FILE
C
C         P. AIELLO  11/1/79
C
C**********
C
C       DECLARE VARIABLES
          BYTE REFILE(32),DAFILE(32),ABFILE(32),OTFILE(32)
          INTEGER IERR,ITABLE(96),REF(512),DARK(512),LIGHT(512)
          REAL DIF(512),ASRB(512)
C       ASK FOR REFERENCE FILE NAME
5         WRITE(5,900)
900       FORMAT('$REFERENCE FILE:  ')
C       READ THE REFERENCE FILE NAME
          READ(5,905)REFILE
905       FORMAT(32A1)
C       OPEN UP THE REFERENCE FILE USING UNFORM
          CALL UNFORM(0,IERR,ITABLE,REFILE,0,1,3)
C       CHECK FOR ERRORS
          IF(IERR .NE. 0)GO TO 10
          IF(IERR .EQ. 0)GO TO 15
C       GIVE ERROR MESSAGE
10        WRITE(5,910)
910       FORMAT(' INPUT REFERENCE FILE NOT FOUND ')
```

```
      GO TO 5
C     READ IN THE REFERENCE FILE DATA
15    CALL UNFORM(1, IERR, ITABLE, REF, 1024)
C     CHECK FOR ERRORS
      IF(IERR .NE. 0)GO TO 75
C     MOVE REFERENCE ARRAY INTO A DIFFERENCE ARRAY
      DO 20 I=1,512
      DIF(I)=FLOAT(REF(I))
20    CONTINUE
C     NOW CLOSE THE REFERENCE FILE USING UNFORM
      CALL UNFORM(2, IERR, ITABLE)
C     CHECK FOR ERRORS
      IF(IERR .NE. 0)GO TO 80
C     ASK FOR DARK CURRENT FILE NAME
25    WRITE(5,915)
915   FORMAT('$DARK CURRENT FILE:  ')
C     READ IN THE DARK FILE NAME
      READ(5,920)DAFILE
920   FORMAT(32A1)
C     OPEN THE DARK FILE USING UNFORM
      CALL UNFORM(0, IERR, ITABLE, DAFILE, 0, 1, 3)
C     CHECK FOR ERRORS, IF SO LOOP BACK
      IF(IERR .NE. 0)GO TO 30
      IF(IERR .EQ. 0)GO TO 35
C     WRITE OUT ERROR MESSAGE AND LOOP BACK
30    WRITE(5,925)
925   FORMAT(' DARK CURRENT FILE NOT FOUND ')
      GO TO 25
C     READ IN THE DARK CURRENT FILE USING UNFORM
35    CALL UNFORM(1, IERR, ITABLE, DARK, 1024)
C     CHECK FOR ERRORS
      IF(IERR .NE. 0)GO TO 85
C     CALCULATE THE TOTAL LIGHT MINUS THE DARK CURRENT
      DO 40 I=1,512
      DIF(I)=DIF(I)-FLOAT(DARK(I))
      IF(DIF(I) .LE. 0.)DIF(I)=1.
40    CONTINUE
C     NOW CLOSE THE DARK CURRENT FILE
      CALL UNFORM(2, IERR, ITABLE)
C     AND CHECK FOR ERRORS
      IF(IERR .NE. 0)GO TO 90
C     GET FILE NAME OF WHICH TO DETERMINE THE ABSORBANCE OF
45    WRITE(5,930)
930   FORMAT('$FILE TO DET. ABSORBANCE VS. WAVELENGTH:  ')
C     NOW READ THE FILE NAME
      READ(5,935)ABFILE
935   FORMAT(32A1)
C     OPEN THE ABSORBANCE FILE USING UNFORM
      CALL UNFORM(0, IERR, ITABLE, ABFILE, 0, 1, 3)
C     CHECK FOR ERRORS
      IF(IERR .NE. 0)GO TO 50
      IF(IERR .EQ. 0)GO TO 55
C     WRITE OUT ERROR MESSAGE AND LOOP BACK
```

```
50        WRITE(5,940)
940       FORMAT(' INPUT ABSORBANCE FILE NOT FOUND ')
          GO TO 45
C       READ IN THE ABSORBANCE DATA USING UNFORM
55        CALL UNFORM(1,IERR,ITABLE,LIGHT,1024)
C       CHECK FOR ERRORS
          IF(IERR .NE. 0)GO TO 95
C       CALC. LIGHT INTENSITY OF SAMPLE AND STORE IN ABS ARRAY
          DO 60 I=1,512
          ASRB(I)=FLOAT(LIGHT(I)-DARK(I))
60        CONTINUE
C       NOW CLOSE THE ABSORBANCE FILE
          CALL UNFORM(2,IERR,ITABLE)
C       CHECK FOR ERRORS
          IF(IERR .NE. 0)GO TO 100
C       NOW CALC. AND STORE THE ABSORBANCES AT EACH WAVELENGTH
          DO 65 I=1,512
          IF(ASRB(I) .LE. 0.)ASRB(I)=1.
          ASRB(I)=-1.0*ALOG10(ASRB(I)/DIF(I))
65        CONTINUE
C       ASK FOR AN OUTPUT FILE NAME
          WRITE(5,945)
945       FORMAT('$OUTPUT FILE NAME:  ')
C       READ FILE NAME FROM USER
          READ(5,950)OTFILE
950       FORMAT(32A1)
C       OPEN OUTPUT FILE USING ASSIGN
          CALL ASSIGN(4,OTFILE,32)
C       SEND HEADER TO OUTPUT FILE
          WRITE(4,955)OTFILE
955       FORMAT('HD ',32A1)
C       SEND NUMBER OF DATA POINTS TO OUTPUT FILE
          WRITE(4,960)
960       FORMAT('NU      512 ')
C       READ DATA FROM INPUT FILE TO THE OUTPUT FILE
          DO 70 I=1,512
          A=ASRB(I)
          B=FLOAT(513-I)
          WRITE(4,965)B,A
965       FORMAT('RD',2G15.7)
70        CONTINUE
C       SEND END OF DATA MARKER TO OUTPUT FILE
          WRITE(4,970)
970       FORMAT('ED ')
C       AND CLOSE THE OUTPUT FILE
          CALL CLOSE(4)
C       WRITE END OF PROGRAM STATEMENT
          WRITE(5,975)
975       FORMAT(' ABSORBANCE CALCULATIONS COMPLETE ')
          GO TO 105
C       THE FOLLOWING STATEMENTS ARE ALL ERROR MESSAGES
75        WRITE(5,980)
980       FORMAT(' DATA CANNOT BE READ OUT OF REF. FILE ')
```

```
      GO TO 105
80    WRITE(5,985)
985   FORMAT(' REFERENCE FILE CANNOT BE CLOSED ')
      GO TO 105
85    WRITE(5,990)
990   FORMAT(' DATA CANNOT BE READ OUT OF D.C. FILE ')
      GO TO 105
90    WRITE(5,995)
995   FORMAT(' DARK CURRENT FILE CANNOT BE CLOSED ')
      GO TO 105
95    WRITE(5,997)
997   FORMAT(' DATA CANNOT BE READ OUT OF ABS. FILE ')
      GO TO 105
100   WRITE(5,998)
998   FORMAT(' ABSORBANCE FILE CANNOT BE CLOSED ')
105   CALL EXIT
      END
```

APPENDIX B

INTERFACE SCHEMATIC DIAGRAMS

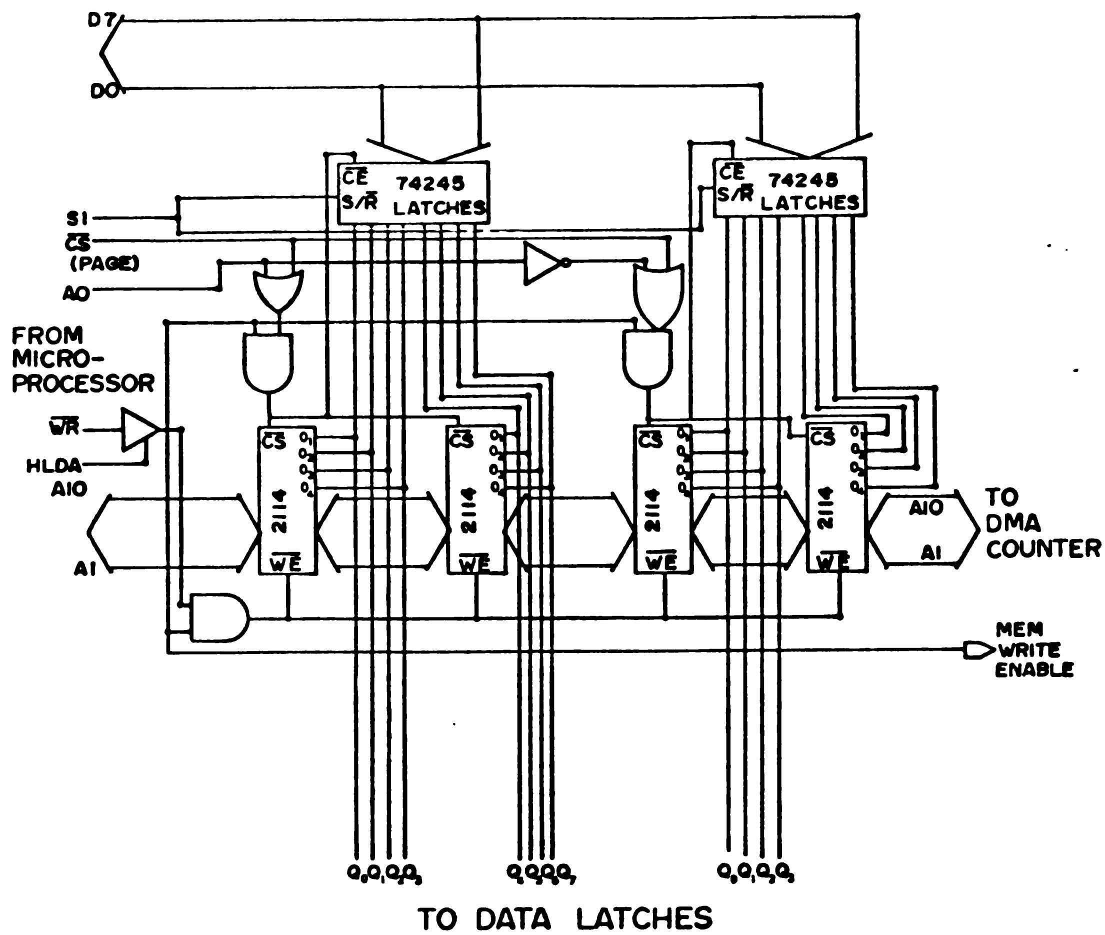


Fig. B.1 Interface RAM schematic diagram

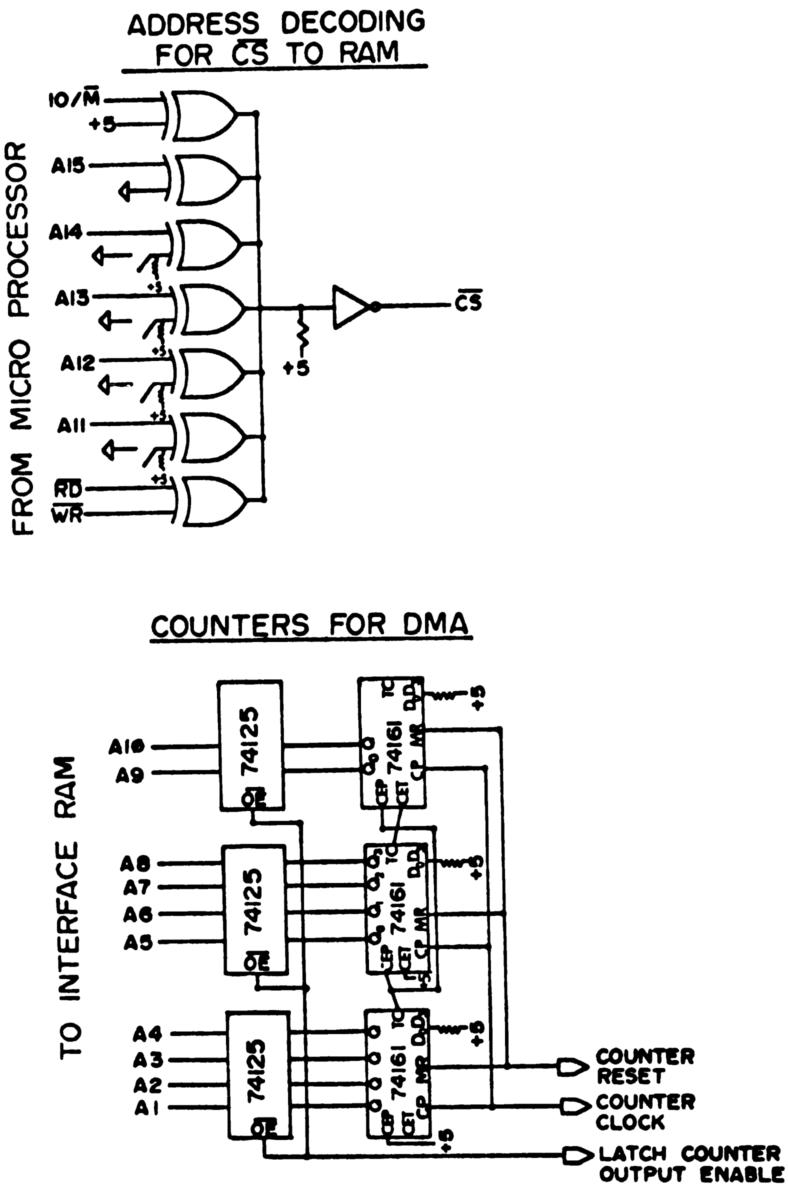


Fig. B.2 Counters and decoding for interface RAM

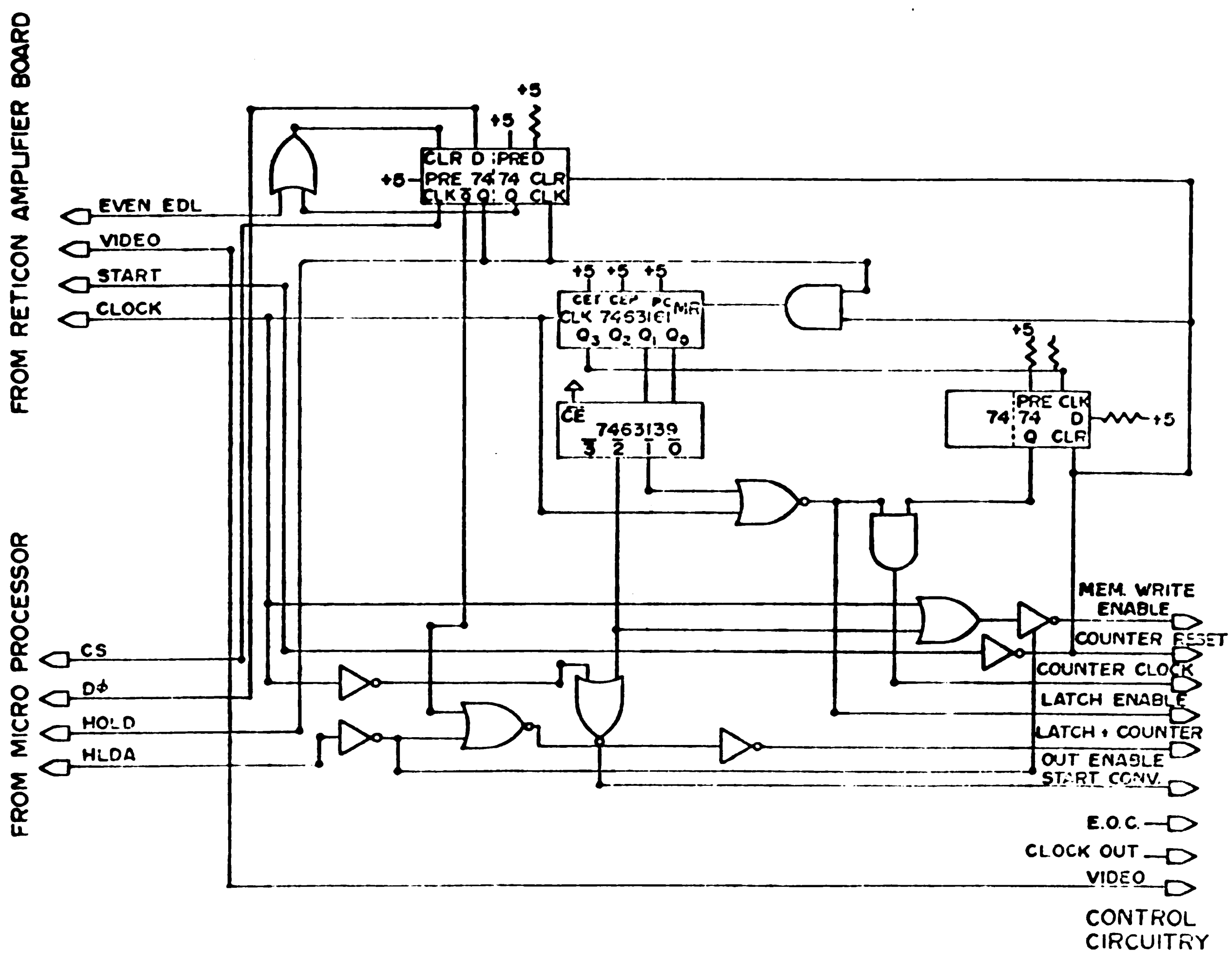


Fig. B.3 Interface control schematic diagram

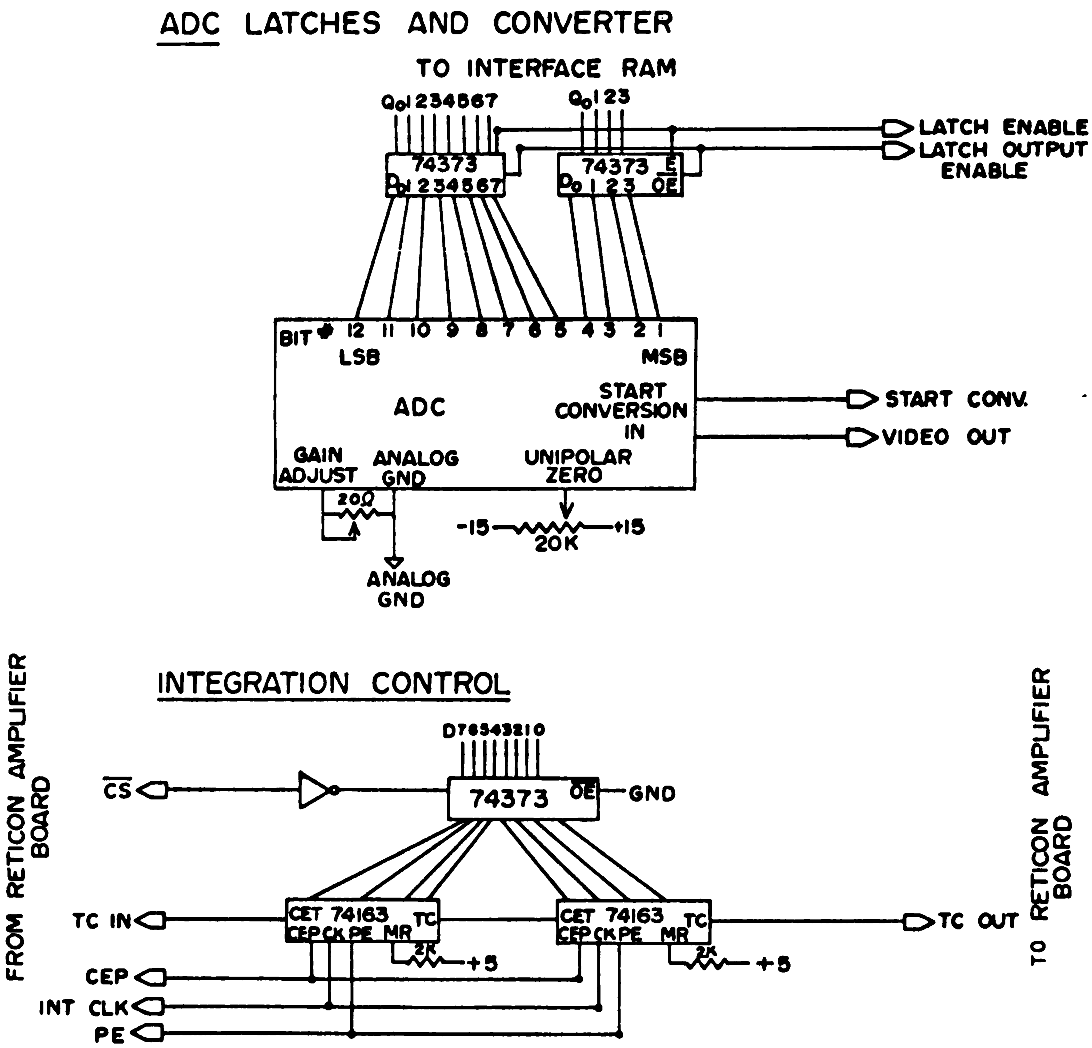


Fig. B.4 ADC and Integration control schematic diagrams

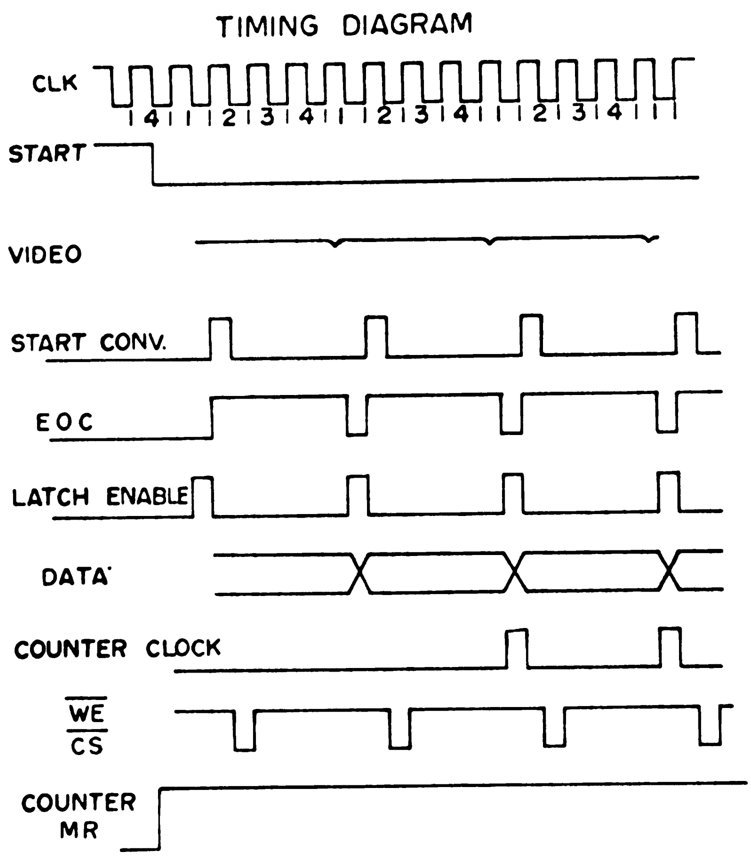
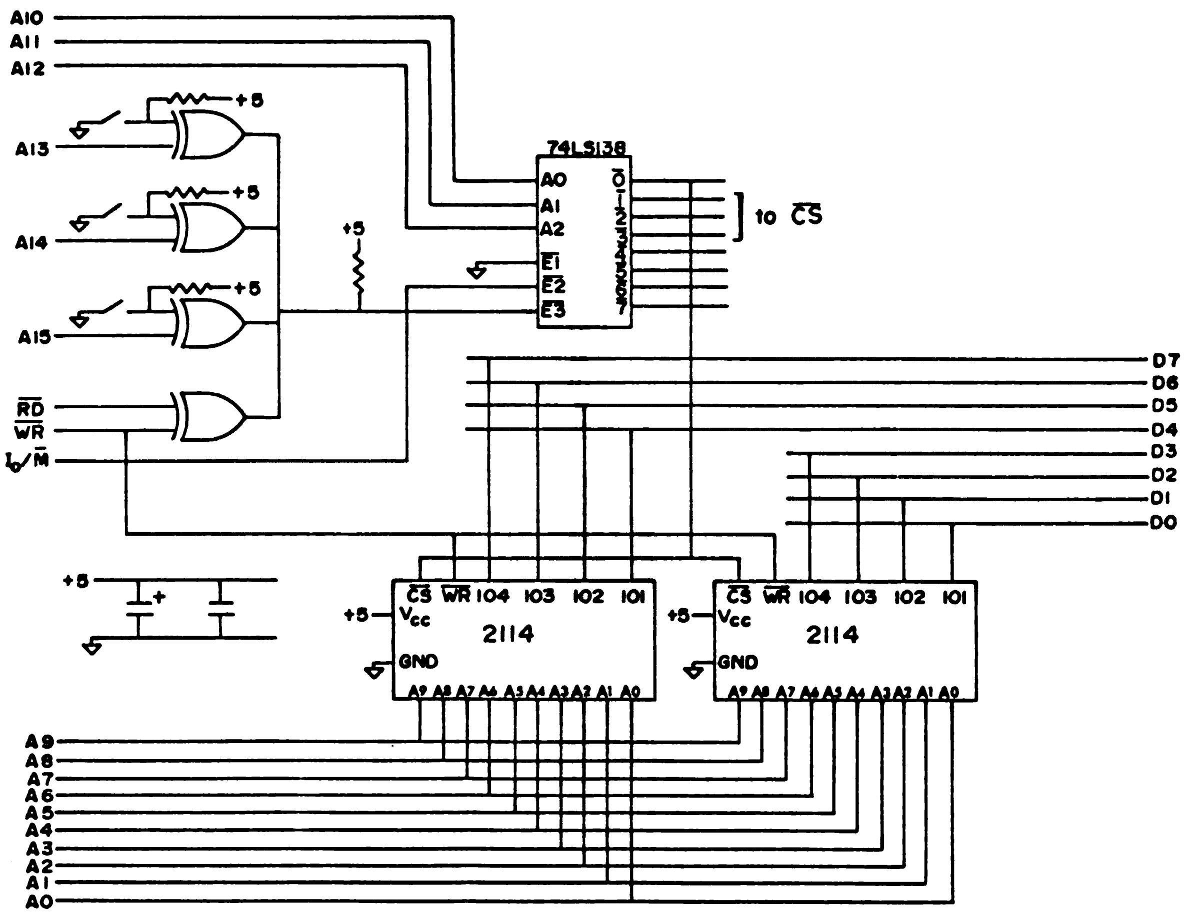


Fig. B.5 Control circuitry timing diagram



4 K RAM BOARD

Fig. C.1 4 k RAM board schematic diagram

APPENDIX C

PERIPHERAL SCHEMATIC DIAGRAMS

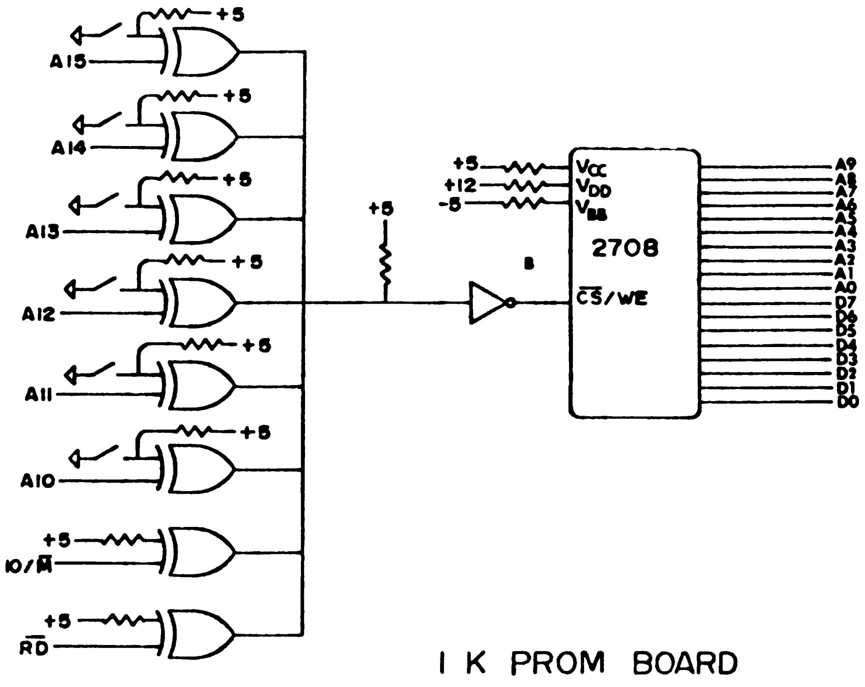


Fig. C.2 1 k PROM board schematic diagram

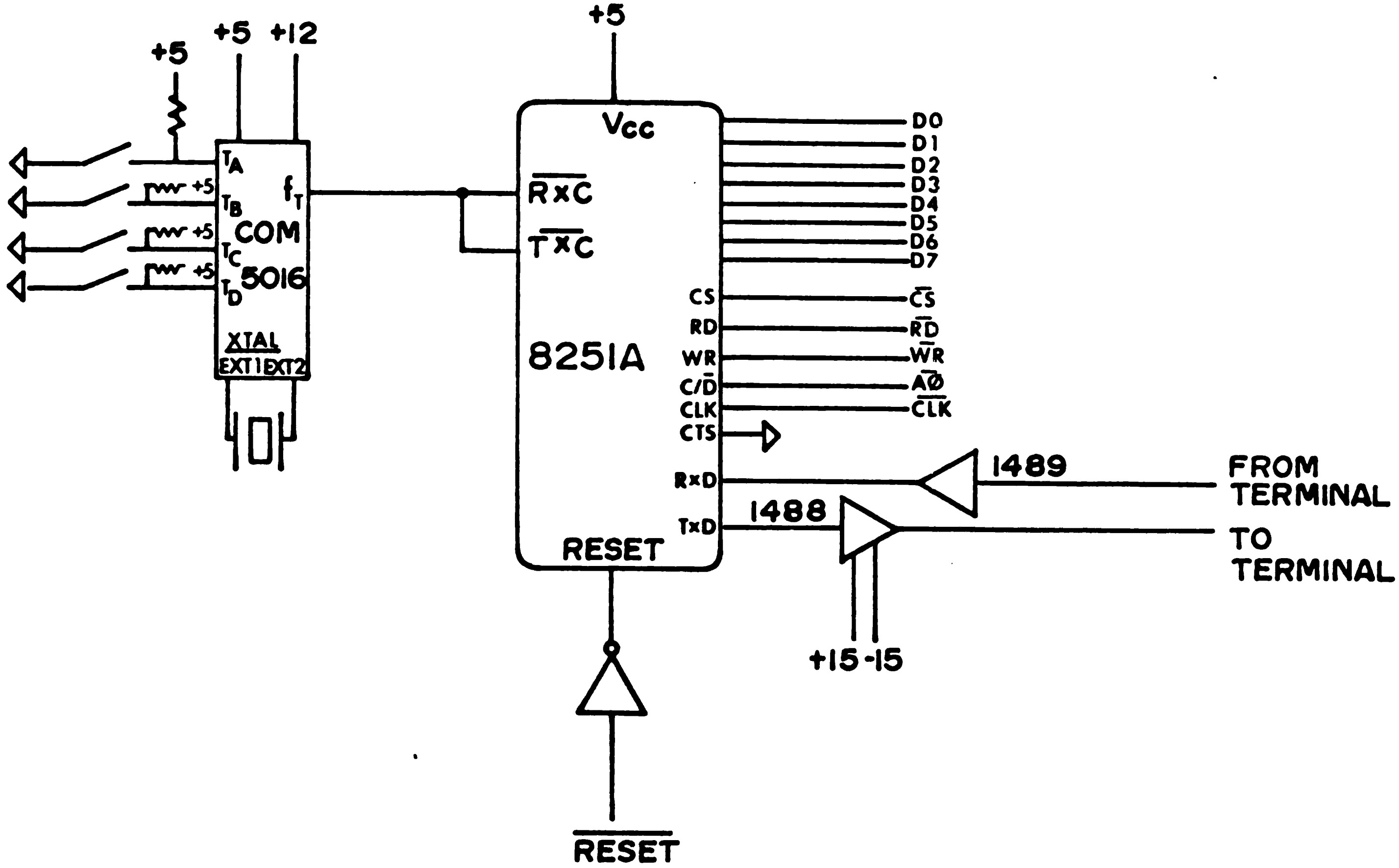
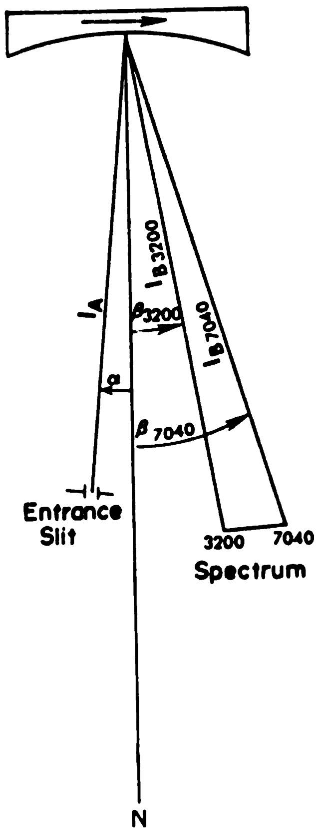


Fig. C.3 USART board schematic diagram

APPENDIX D

GRATING SPECIFICATIONS

GRATING SPECIFICATIONS



l_A = 96.9nm α = -5.2217°

l_B = 105.44nm β_{3200} = +10.778°

l_B = 105.49nm β_{4000} = +12.182°

l_B = 106.92nm β_{6000} = +15.724°

l_B = 107.48nm β_{7040} = +17.590°

Fig. D.1 Grating specifications

REFERENCES

REFERENCES

1. R. M. Barnes, Anal. Chem., 44, 122R (1972)

2. Willard, Merritt and Dean, "Instrumental Methods of Analysis", Fifth Ed., D. Van Nostrand Company, New York, 413 (1974).

3. R. E. Santini, M. H. Milano and H. L. Pardue, Anal. Chem., 45, 915A (1973).

4. G. Horlick and E. G. Codding, Anal. Chem., 45, 1490, (1973).

5. Y. Talmi, Am. Lab., March, 79 (1978).

6. G. Horlick, Appl. Spectrosc., 30, 113 (1976).

7. Y. Talmi, Anal. Chem., 47, 658A (1975).

8. Y. Talmi, Anal. Chem., 47, 697A (1975).

9. K. W. Busch and G. H. Morrison, Anal. Chem., 45, 712A (1973).

10. J. D. Winefordner, J. J. Fitzgerald and N. Omenetto, Appl. Spectrosc., 29, 369 (1975).

11. R. P. Coomey, G. D. Boutilier and J. D. Winefordner, Anal. Chem., 49, 1048 (1977).

12. D. F. Brost, B. Malloy and K. W. Busch, Anal. Chem., 49, 2280 (1977).

13. H. L. Felkel and H. L. Pardue, Anal. Chem., 50, 602 (1978).

14. N. G. Howell and G. H. Morrison, Anal. Chem., 49, 106 (1977).

15. H. L. Felkel and H. L. Pardue, Anal. Chem., 49, 1112 (1977).

16. F. S. Chaung, D. F. S. Natusch and K. R. O'Keefe, Anal. Chem., 50, 525 (1978).

17. G. Horlick and E. G. Codding, Appl. Spectrosc., 29, 167 (1975).

18. K. M. Aldous, D. G. Mitchell and K. W. Jackson, Anal. Chem., 47, 1034 (1975).

19. T. L. Chester, H. Haraguchi, D. O. Knapp, J. D. Messman and J. D. Winefordner, Appl. Spectrosc., 30, 410 (1976).

20. T. E. Cook, M. J. Milano and H. L. Pardue, Clin. Chem. 20, 1422 (1974).

21. E. G. Codding and G. Horlick, Spectrosc. Lett., 7, 33 (1974).

22. T. E. Edmonds and G. Horlick, Appl. Spectrosc., 31, 536 (1977).

23. M. Franklin, C. Baber and S. R. Koirtyohann, Spectrochim. Acta., Part B, 31, 589 (1976).

24. K. R. Betty and G. Horlick, Appl. Spectrosc., 32, 31 (1978).

25. G. Horlick, E. G. Codding and S. T. Leung, Appl. Spectrosc., 29, 48 (1975).

26. K. W. Busch, N. G. Howell and G. H. Morrison, Anal. Chem., 46, 2074 (1974).

27. N. G Howell, J. D. Ganjei and G. H. Morrison, Anal. Chem., 48, 319 (1976).

28. A. E. McDowell, R. S. Hanner and H. L. Pardue, Clin. Chem., 22, 1862 (1976).

29. M. J. Milano and K. Y. Kim, Anal. Chem., 49, 555 (1977).

30. A. E. McDowell and H. L. Pardue, Anal. Chem., 49, 1171 (1977).

31. R. E. Dessey, W. D. Reynolds, W. G. Nunn, C. A. Titus, and G. F. Moler, Clin. Chem., 22, 1472 (1976).

32. J. R. Jadamec, W. A. Sauer and Y.. Talmi, Anal. Chem., 49, 1316 (1972).

33. R. P. Cooney, T. Vo-Dinh and J. D. Winefordner, Anal. Chim. Acta., 89, 9 (1977).

34. E.G.G. RETICON, S-Series Solid State LDA, Brochure (1978)

35. Bruce Newcome, Dept. of Chemistry, Michigan State University (1980).

36. Hugh Gregg, Dept. of Chemistry, Michigan State University (1980).

37. Product Specifications on Model 1070 Intelligent Diskette Controller, Persci, Marina Del Rey, CA. (1976).

38. T. A. Neiman, Ph.D. Thesis, Michigan State University, (1975).

39. J. E. Hornshuh, Ph.D. Thesis, Michigan State University, (1978).

40. T. B. McCord and M. J. Frankston, Appl. Optics, **14**, 1437 (1978).

41. E. M. Carlson, Ph.D. Thesis, Michigan State University, (1978).