12-1984

The Application of Multiderivative Infrared Spectroscopy to Iodine Number Determination

Jeffrey C. Lowry

THE APPLICATION OF MULTIDERIVATIVE INFRARED SPECTROSCOPY TO IODINE NUMBER DETERMINATION

by

Jeffrey C. Lowry

A Thesis
Submitted to the
Faculty of The Graduate College
in partial fulfillment of the
requirements for the
Degree of Master of Arts
Department of Chemistry

Western Michigan University
Kalamazoo, Michigan
December 1984

THE APPLICATION OF MULTIDERIVATIVE INFRARED SPECTROSCOPY TO IODINE NUMBER DETERMINATION

Jeffrey C. Lowry, M.A.

Western Michigan University, 1984

The Food and Drug Administration requires the disclosure of total unsaturation of commercial products as an indication of nutritional value. This unsaturation is a guide to the expected stability of food additives, such as edible oils and foods in which they are used. Unsaturation is typically evaluated as an iodine number, calculated for these additives based on their consumption of iodine through addition across the carbon-carbon double bonds. This titration method can typically take 45 minutes.

The investigation presented develops an alternative method for determining iodine numbers of some common oils using a computerized infrared spectrophotometer. The quantitation technique employed involves multiderivative spectroscopy. Results of the analytical data give iodine numbers with relative errors of approximately 1 percent. The higher order derivative quantitation technique is limited by the noise within the spectrophotometer. The analysis time, as compared to the titration method, is reduced by 20 fold.

ACKNOWLEDGEMENTS

I would like to thank Jim Howell for his support, patience and time throughout the years. My appreciation is also extended to my family, without whom this thesis would not be possible. Finally, to my best friend, Dianne, for that continuing supply of moral support.

Jeffrey C. Lowry

1324828

LOWRY, JEFFREY CRAIG

THE APPLICATION OF MULTIDERIVATIVE INFRARED SPECTROSCOPY TO IODINE NUMBER DETERMINATION

WESTERN MICHIGAN UNIVERSITY

M.A. 1984

University Microfilms International 300 N. Zeeb Road, Ann Arbor, MI 48106

PLEASE NOTE:

In all cases this material has been filmed in the best possible way from the available copy. Problems encountered with this document have been identified here with a check mark _√_.

1. Glossy photographs or pages _____
2. Colored illustrations, paper or print _____
3. Photographs with dark background _____
4. Illustrations are poor copy _____
5. Pages with black marks, not original copy _____
6. Print shows through as there is text on both sides of page _____
7. Indistinct, broken or small print on several pages _√_
8. Print exceeds margin requirements _____
9. Tightly bound copy with print lost in spine _____
10. Computer printout pages with indistinct print _____
11. Page(s) __________ lacking when material received, and not available from school or author.
12. Page(s) __________ seem to be missing in numbering only as text follows.
13. Two pages numbered __________. Text follows.
14. Curling and wrinkled pages _____
15. Other____

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Infrared Spectroscopy

Infrared (I.R.) spectroscopy is one of the more widely used instrumental techniques for studying the structure of chemical compounds. Since its discovery by Sir William Herschel in 1800 (1), it has found application in many disciplines throughout the scientific world. Infrared spectroscopy has become very important to the physicist, the biologist, and the chemist as a tool for solving a wide variety of problems. Infrared spectroscopy provides investigators with a powerful insight into molecular structure and chemical identification.

The infrared region of the electromagnetic spectrum lies between the visible and microwave regions. This region starts near the red end of the visible spectrum at 14,000 cm^{-1} or 0.7 μm and extends to approximately 10 cm^{-1} or 1000 μm. Both micrometers (μm) and wavenumbers in reciprocal centimeters (cm^{-1}) are used to characterize infrared wavelengths. As shown in Figure 1, the infrared spectrum can be divided into three subdivisions according to the different kinds of molecular

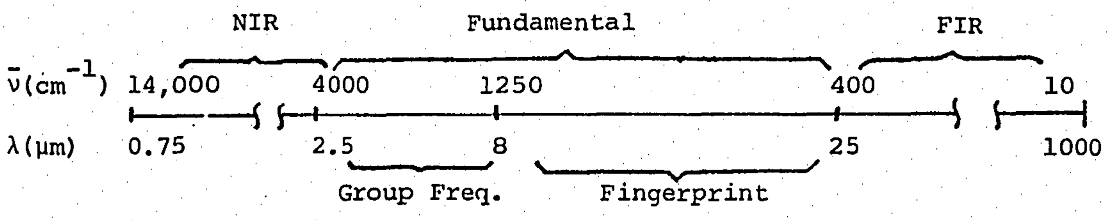


Figure 1. Common Subdivisions of the Infrared Region

information which they can provide. The first region, ranging from 14,000 to 4,000 cm^{-1}, is called the near infrared (NIR) region. Absorption bands in the NIR region result from overtones or combinations of hydrogen stretching vibrations. This region can be easily used for quantitative analysis of certain functional groups containing unique hydrogen atoms within a compound.

The majority of qualitative and quantitative work is done in the middle region or the fundamental infrared region. This region's absorption bands result from vibration and rotation of functional groups. The fundamental region is divided into a group frequency region from 4,000 to 1,250 cm^{-1} and a fingerprint region from 1,250 to 400 cm^{-1}. In the group frequency region the absorption bands tend to be characteristic of specific groups of atoms within the structure of the molecule. The fingerprint region bands, on the other hand, result from effects of the total molecular structure.

The far infrared (FIR) region gives information mainly about rotational transitions, vibrational modes of crystal lattices, and skeletal vibrations of large molecules. This region starts at 400 cm^{-1} and extends to the microwave region at 10 cm^{-1}. It is generally not useful for quantitative analysis due to the relatively poor signal-to-noise ratio implicit in long wavelength spectroscopy and by the relatively low intensity of the absorption bands (1).

Most quantitative infrared spectroscopic measurements are performed in the fundamental region. Several difficulties commonly occur in quantitative I.R. analysis. The major problem arises from an instrumental source and is commonly referred to as stray or scattered

radiation. Due to the small radiation energy available from I.R. sources, relatively wide slit widths are used. This causes a proportional increase in the scattered radiation and a broadening of the absorption bands. This distortion in the maxima causes deviations from Beer's Law (2), the basic quantitative relationship in spectroscopic analysis.

Another difficulty in I.R. quantitative analysis is the uncertainty in determining the 100 percent transmittance level. This uncertainty is caused by the variability of path length of the very thin cells commonly used in I.R. spectroscopy. The problem of determining the 100 percent level causes variations in the baseline over the width of an absorption maximum. Thus, it is necessary to draw the best approximate baseline across the peak for quantitation. This method is referred to as the base-line method and assumes that the background is linear over the width of the band. This assumption is often unjustified as bands overlap, thus producing error in the peak height measurements.

Other common sources of error in quantitative I.R. arise from solvent absorption, atmospheric absorption (humidity and carbon dioxide), and heat of radiation. As absorption cells are exposed to the fairly intense beam of infrared radiation, a warming effect occurs. The result is that the temperature of the solution increases causing the absorption bands to vary.

In spite of all these problems, infrared spectroscopy can be used to effect quantitative analyses with accuracies on the order of 1 percent (1). In order to achieve this, extreme care must be taken to use only the best analytical techniques. These techniques include: choosing

the analytical wavelength for minimum interference, adjusting the sample thickness and concentration such that transmittance of the analytical band is between 30 and 40 percent, selecting the optimum slit width for highest signal-to-noise ratio without losing required resolution, preparing the standards at expected sample concentrations, and minimizing the atmospheric interferences.

Derivative Spectroscopy

Derivative spectroscopy is a relatively new analytical technique. Even though it was introduced more than twenty-five years ago, it has been accepted only hesitantly. This was partly due to the high cost of instrumentation. Reliable derivatives can now be taken electronically and consequently this has resulted in a significant reduction in the cost of instrumentation. The derivative of a spectrum (3) is a graphic representation of the mathematical differentiation of an absorbance spectrum.

$$\text{1st derivative} = \frac{dA}{d\bar{\nu}} \tag{1}$$

where A = absorbance and $\bar{\nu}$ = wavenumber.

Actually, most electronically generated derivative spectra are differentiating the change in the absorbance signal with respect to the time, rather than the wavelength or wavenumber (4). However, since spectra are collected at a constant wavelength scanning speed, these functions become equivalent.

$$\frac{dA}{dt} = \frac{dA}{d\bar{\nu}} \tag{2}$$

where t = time.

There are generally four methods for obtaining derivative spectra. The first and most expensive is to use a specially designed optical instrument. This instrument requires two monochromators to produce two separate scans at the same time. This method, referred to as dual-wavelength (5), scans simultaneously with both monochromators one of which has a small wavelength offset relative to the other. This produces the derivative by providing a difference in the absorbance readings between the two monochromators with respect to the difference in their wavelength.

The second method for producing derivative spectra also employs optical modification of the instrument. In this method, called wavelength modulation (6), the wavelength to which the monochromator is set is repetitively and rapidly scanned back and forth over a small spectral interval. This modulation interval produces an alternating current at the detector which represents the derivative of the spectra with respect to wavelength. Both the dual-wavelength and the modulation wavelength methods have been limited to first derivative spectroscopy.

More recently, derivative spectra have been taken using electronic circuits. Electronic circuits use the original output of the detector to produce the derivative instead of extensive instrumental modification. This method, referred to as electronic differentiation (3), is achieved by using an operational amplifier in the differentiator configuration. The signal from the detector is connected to the amplifier to produce the first derivative of the spectra. Multiderivative spectra can be produced by connecting together a series of differentiating

amplifiers. The major problem involved with the electronic differentiation technique results from the noise introduced into the spectra by the operational amplifiers.

As a result of advancements in computer interfacing the fourth method, numeric analysis (3), has become possible. The analog signal from the detector is converted to a digital signal using an electronic circuit called an analog-to-digital (A/D) converter. These data points, which represent absorbance readings versus wavelength, are stored in memory within a computer from the A/D. The digital spectrum is then numerically differentiated by subtracting consecutive absorbance readings. Therefore, the derivative by this technique is the change in absorbance with respect to the change in wavelength which can be a good approximation of the true derivative (5).

$$\lim_{\Delta\bar{\nu}\to 0} \frac{A}{\Delta\bar{\nu}} = \frac{dA}{d\bar{\nu}} \tag{3}$$

where ΔA = change in absorbance and $\Delta\bar{\nu}$ = change in wavenumber. The validity of this equation is improved as the change in wavelength increments become smaller. The limit of change in wavelength is inherently restricted by the resolution of the spectrophotometer used. Higher order derivatives are easily obtained by repetitive differentiation by the computer.

The derivative technique has found many applications throughout the field of spectroscopy. Derivative spectroscopy has been applied in UV-visible (3), flame emission (7), flame absorption (8), luminescence (7), fluorescence (9), and also infrared (10,11) spectroscopic instruments. Most experimentation has been performed in the UV-visible

region and has been restricted to the use of the first derivative. The principal advantage of derivative measurements is the improvement in the detectability of minor spectral features. This spectral enhancement reduces the potential for measurement errors caused by overlapping and unresolved bands. Minor differences in the spectra can be so enhanced, that quantitation can become possible.

The derivative is a measure of the rate change in the slope of the spectra. Figure 2 shows the absorbance spectra obtained for compounds A, B, and a mixture of A and B. Between wavelengths c and d, the rate change in the slope for spectra B is greater than that for A. Over the same wavelength range on spectra A + B, the rate change in the slope is more sensitive to changes in compound B than it is to changes in compound A. Thus, the derivative emphasizes slope change with respect to wavelength and can sequently provide selectivity of B relative to A.

The application of quantitative derivative spectroscopy is based on the measurement of the maxima and minima of the derivative spectra. Figure 3a depicts two overlapping absorption bands. Figure 3b depicts the corresponding derivative maxima. The derivative maximum with the greatest response relative to concentration are usually used for quantitation. Occasionally a lesser maximum may be used in order to improve selectivity.

Most derivative spectroscopy performed in the infrared region has been done in the group frequency region between 4 and 8 μm. In several investigations (10,11), the half-width peak height ratios for the overlapping bands were in the order of 1:2. The limiting factor in

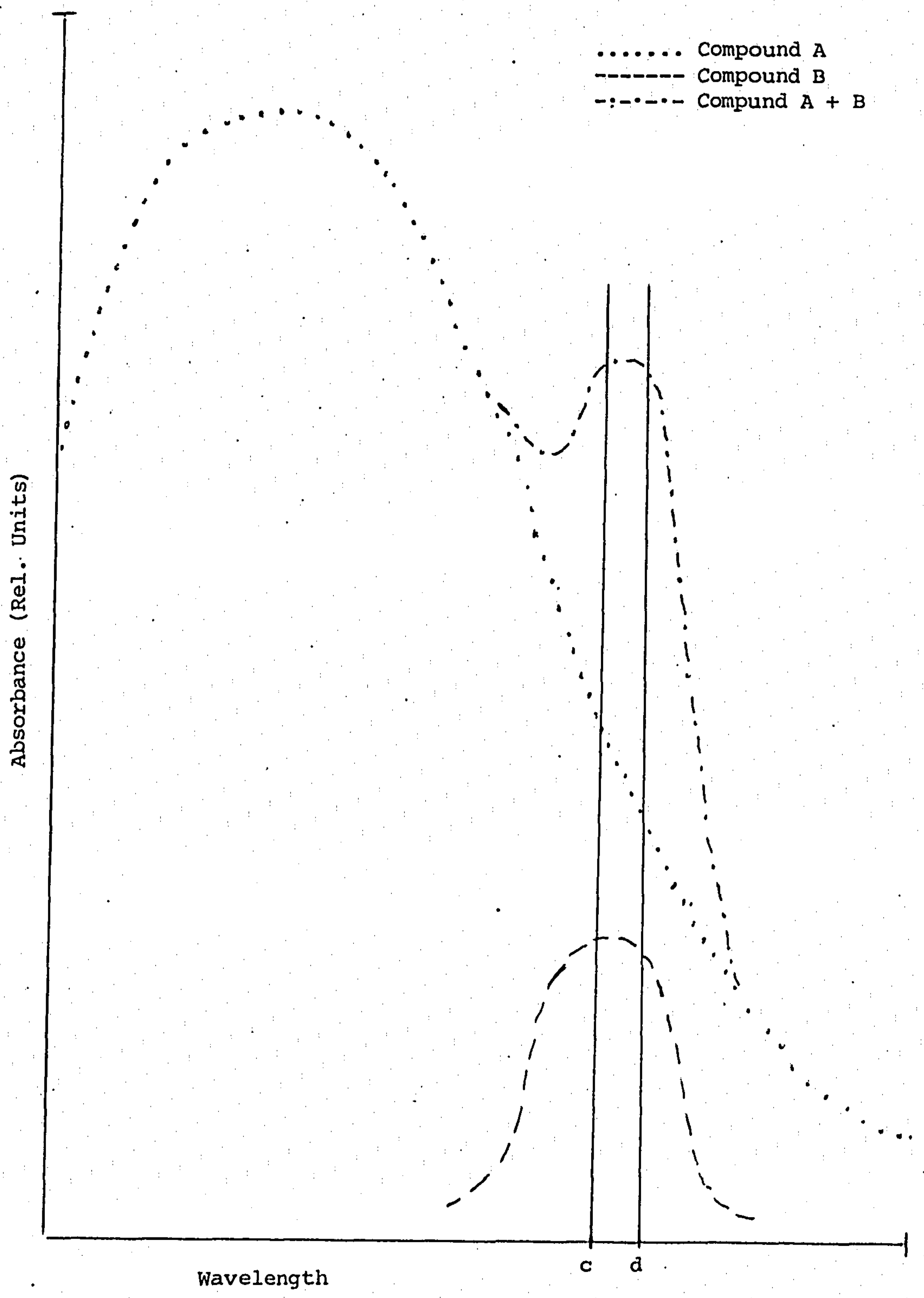
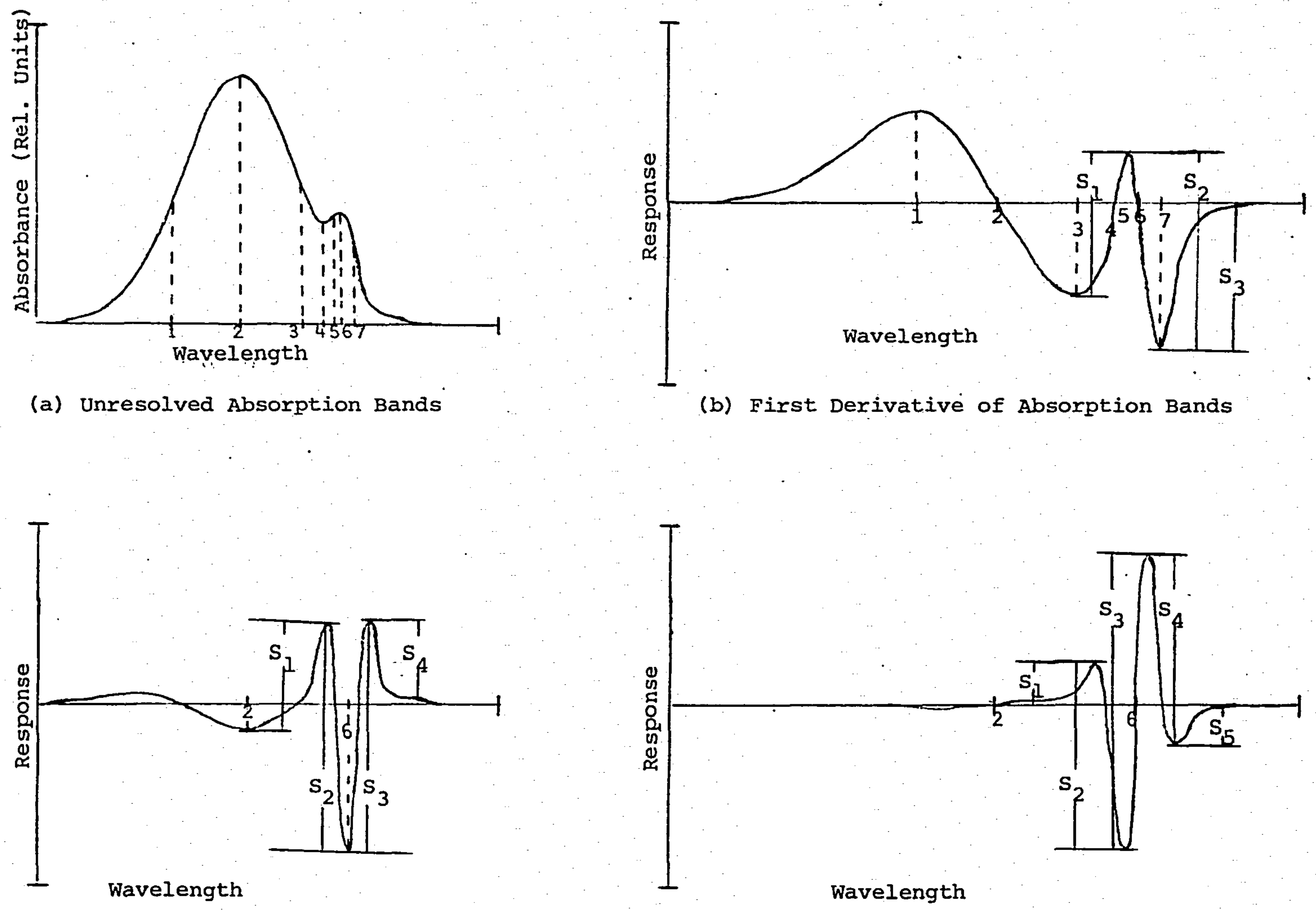


Figure 2. Overlapping Absorbance Curves



(a) Unresolved Absorption Bands

(b) First Derivative of Absorption Bands

(c) Second Derivative of Absorption Bands

(d) Third Derivative of Absorption Bands

Figure 3. Absorption Maxima and Derivative Spectra

derivative infrared spectroscopy is the inherent noise accompanying the signal. As the signal-to-noise ratio decreases within the spectra, the error in the derivative measurements becomes more significant.

Computers

The computer has made a dynamic impact on modern scientific instrumentation. It has made data acquisition, data processing, and instrument control extremely efficient and simple. Many modern spectrophotometic instruments are being designed with computer interfaces. This makes derivative spectroscopy possible in the majority of instruments currently available.

The digital computer (12) performs operations in response to commands called instructions. To perform a task, the computer uses a sequence of these instructions called a program. The device within the computer that executes the program is referred to as the central processing unit (CPU). Most CPUs include a circuit that provides arithmetic and logic operations (ALU) and a variety of registers for temporary data storage. This CPU which contains only a few integrated circuits is referred to as a microprocessor. In most smaller computers, the microprocessor is the CPU. This type of computer is called a microcomputer.

The principal component of the microcomputer is the microprocessor or CPU. The CPU controls all main communication to and from the computer through electrical lines called the processor bus. As shown in Figure 4 all other parts of the computer, including input/output devices and memory, are connected to the CPU through the data, address, and control

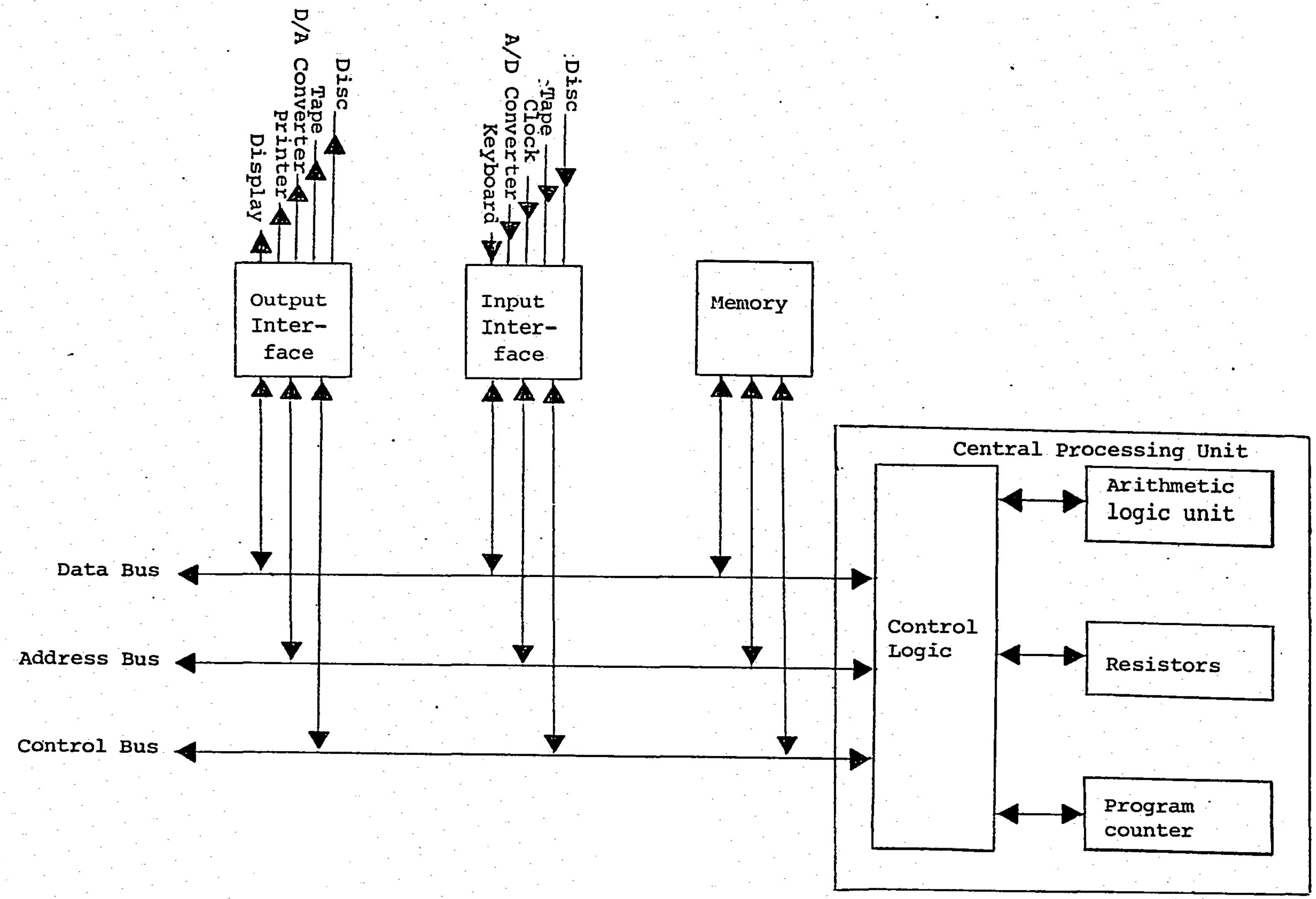


Figure 4. Structure of a Basic Digital Computer

lines of this bus. The control bus uses the data and address busses to input into the CPU (read) and output from the CPU (write).

The standard microprocessor today is an 8-bit processor. Examples are the INTEL 8080A and 8088, the Zilos Z80, the Motorola MC6800, the MOS Technology MCS6500, and the Signetics 2650 (13). These microprocessors are the building blocks for most microcomputers on the market. More recently, 16-bit microprocessors have been introduced. These processors have superior speed over the conventional 8-bit processors. The most widely used 16-bit microprocessors are the Motorola MC68000, the Zilos Z8000, the Intel 8086, the Texas Instrument 9900, the Digital Equipment LSI-11, and the Fairchild 9940 (14).

Within the computer, the CPU has a system of data storage and retrieval called the memory. The memory contains space for both program, needed to perform the current task, and data storage. There are two types of memory. The first is the read-write memory. The binary digits within this type of memory can be changed, as well as read by the CPU. Read-write memory is commonly referred to as random access memory or RAM. The second type of memory is called read only memory or ROM. The binary digits of the ROM can be read, but not changed by the CPU.

Data are transferred to the microcomputer as binary digital signals. There are two ways to input and output digital signals. The first is called parallel communication (12,15). Parallel communication devices latch on to eight bit digital words from a peripheral source and allow the computer to access this information through the data bus. They also hold data from the computer in their "latches" for use by a

peripheral device. The data transferred between a parallel communication device and a peripheral source is in the form of a parallel digital word. Most parallel communication devices are programmable. These devices, called programmable parallel input-output devices (PIO), perform a variety of functions. The PIO provides address decoding, data I/O buffering, and status information gathering from the peripheral device. Through internal registers, the PIO can be programmed by the computer to input or output data from any selected bit of the parallel port. Due to the address decoder, the computer can treat the device as addressable memory. Parallel communication devices supply parallel digital word communication between the computer and peripheral devices.

The other type of digital communication is called serial communication (12,15,16). Instead of latching eight bits of parallel data, a serial device gathers data on a single line, one bit at a time, until it has the complete eight bit word. There are two ways in which the serial transmission may be utilized. The first is synchronous transmission which requires the computer and the peripheral device to concurrently know when data transmission starts and ends. Thus, a consistent time interval must be utilized between the computer and the peripheral to transmit data. The second serial transmission type is asynchronous data transmission. In this type of communication, the peripheral device signals the computer that it has data to be transmitted. Upon receiving the proper responses from the computer, the peripheral device sends the data. Serial communication devices have two functions: to take parallel data and convert it to serial bit streams, and to take a serial stream and convert it to parallel data. The

Standard serial device is called the universal synchronous-asynchronous receiver-transmitter (USART).

Parallel and serial communication devices are the basic building blocks in data acquisition systems from the real world. Within the computer, all data transfer is achieved in the digital world. Outside the computer, in the real world, most measurements or data transfer are analog. Analog signals or voltage fluctuations assume a continuous range of values, whereas digital signals assume only a finite number of values. For an input signal to go into the computer, an analog to digital (A/D) conversion must take place. For a digital signal to be understood in the real world, a digital to analog (D/A) conversion must take place.

The device within an interface system that is the link from the real (analog) world to the computer (digital) world is called an analog to digital converter. There are generally four techniques for A/D conversion (15-17). The first is the successive approximation. This technique generates an initial guess of the input value, converts it to an analog value, and then compares it to the actual input. Depending on the result of the comparison, the initial guess will be increased or decreased. Successive approximation is one of the most frequently used A/D converter with microprocessor interface systems because of the high speed, high resolution, and low cost.

The second technique for A/D conversion is called integration. The dual slope integrating A/D is the basic type of integrating converter. This method measures the time it takes for a capacitor to charge to an unknown voltage and to discharge under an unknown reference

voltage. The ratio of the times measured equals the ratio of the unknown and known voltages. This technique is often too slow to use for many microprocessor interfacing applications. It is commonly used in digital voltmeters because of its excellent precision and noise immunity.

The counter type converter is the third type of A/D conversion. In this technique, the analog input voltage is compared with the output of a D/A. The input of the D/A is supplied by a counter. When the reference voltage from the D/A exceeds the unknown voltage, the output of the comparator turns off the counter. The value in the counter is proportional to the unknown voltage. The speed of this technique is dependent on the unknown voltage.

The fourth technique for A/D conversion is called parallel conversion. This technique of direct comparison converts simultaneously all possible digital values into their analog form, and compares them to the input signal. This method is extremely fast, but its precision is poor.

The device within an interface system which is the link from the digital world to the analog world is the digital to analog converter (12,15-17). The basic D/A converter used today is called the weighted resistor ladder. In the resistor ladder technique, a variety of resistors are connected in a parallel configuration to a reference voltage source through switches. Each resistor represents a bit in the digital value. The most significant bit of the digital number has the least resistance. Thus, as a digital value appears, it closes the respective switches and an analog value is produced through the resistors. Fixed gained

amplifiers are connected at the output of the ladder to increase the analog value. Most D/A converters use this basic concept of laddered resistors. The precision of the ladder D/A converter is dependent on the precision of the resistors used.

D/A and A/D converters are the essential connection between the digital and analog worlds. Within an interface system, they provide data acquisition and control of the real world through the computer world.

Proposed Problem

The U.S. Food and Drug Administration requires the disclosure of total unsaturation in food additives, such as edible oils. The total unsaturation of an oil is an indication of the nutritional value, as well as a partial guide to its expected stability. Food technologists use the term iodine number as an expression of the level of unsaturation in a sample. Iodine number of an oil is defined as the number of grams of iodine absorbed by 100 grans of oil (18).

Oils are esters of glycerol and fatty acids. As shown in Figure 5, the general structure of an oil is a triacylglycerol.

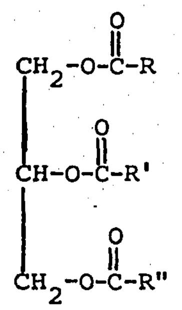


Figure 5. Triacylglycerol

Hydrolysis of common oils yields acyl (R) groups or unsaturated

fatty acids, some of which are shown in Table 1. The carbon-carbon

Table 1

Common Fatty Acids from the Hydrolysis of Oils

Oleic Acid	$CH_3(CH_2)_7CH{=}CH(CH_2)_7COOH$
Linoleic Acid	$CH_3(CH_2)_4CH{=}CH{-}CH_2CH{=}CH{-}(CH_2)_7COOH$
Linolenic Acid	$CH_3CH_2CH{=}CHCH_2CH{=}CHCH_2CH{=}CH{-}(CH_2)_7COOH$
Eleostearic Acid	$CH_3(CH_2)_3{-}CH{=}CH{-}CH{=}CH{-}CH{=}CH{-}(CH_2)_7COOH$
Ricinoleic Acid	$CH_3(CH_2)_5{-}\underset{\displaystyle OH}{\underset{\vert}{CH}}{-}CH_2{-}CH{=}CH{-}(CH_2)_7COOH$

double bonds of the fatty acids are the active reaction sites of the oil utilized for the determination of the iodine number. The iodine number is traditionally obtained from a titration technique called the Wijs or Hanus method (19). In these titrations excess iodine is added to a sample of an oil in order to add across the carbon-carbon double bonds. The absorbed iodine is determined by back titration of the unreacted iodine with sodium thiosulfate. The iodine solutions for each method are some what different. The Wijs titration adds chlorine to increase the reactivity of the iodine solution, whereas the Hanus titration adds bromine (20). The typical reaction time of the oil and the iodine solution is 30 to 45 minutes. Mercuric acetate can be added to the iodine solutions to speed the reaction time to 3 to 5 minutes (19). Total analysis time for an oil using these titration methods is at least 45 minutes.

Another method for determining iodine number of an oil is the use of a refractometer (21). The refractive indices of an oil can be measured before and after iodination with the iodine solution. A ratio

of the refractive indices of a standard versus an unknown oil correspondeds to the iodine number of the unknown oil.

Iodine numbers have been determined using nuclear magnetic resonance, Raman spectroscopy, and electrolytic instruments. Although iodine number determinations have been obtained through a variety of instruments, the sodium thiosulfate titration remains the standard method used by the Association of Official Agriculture Chemists (A.O.A.C.)(20).

More recently infrared spectroscopy has been used to obtain iodine numbers comparable to the titration methods (22). The infrared spectrum of a typical oil (soybean) is shown in Figure 6. Two distinct bands are obtained due to the carbon-carbon double bonds within the oil. The first is at 3018 cm^{-1}. This band is caused by the alkene carbon to hydrogen stretching vibrations. Also in the same proximity is the band caused by alkane carbon to hydrogen stretching vibrations. Because of the high degree of overlap, the band at 3018 cm^{-1} can not reliably be used for quantitation of iodine numbers. A second band due to the carbon-carbon double bond stretching vibrations is isolated at 1654 cm^{-1}. This weaker band can be used for iodine number determinations in neat oils. Due to the large background caused by the carbon-oxygen band at 1750 cm^{-1}, a correction method must be employed to facilitate quantitation. A computer controlled background correction method has been used to obtain peak height measurements of the band at 1654 cm^{-1} (22). Standard oils with known iodine values were used to compare with standard deviations of 1 percent or less were obtained using the computerized infrared background correction method. The time for an analysis

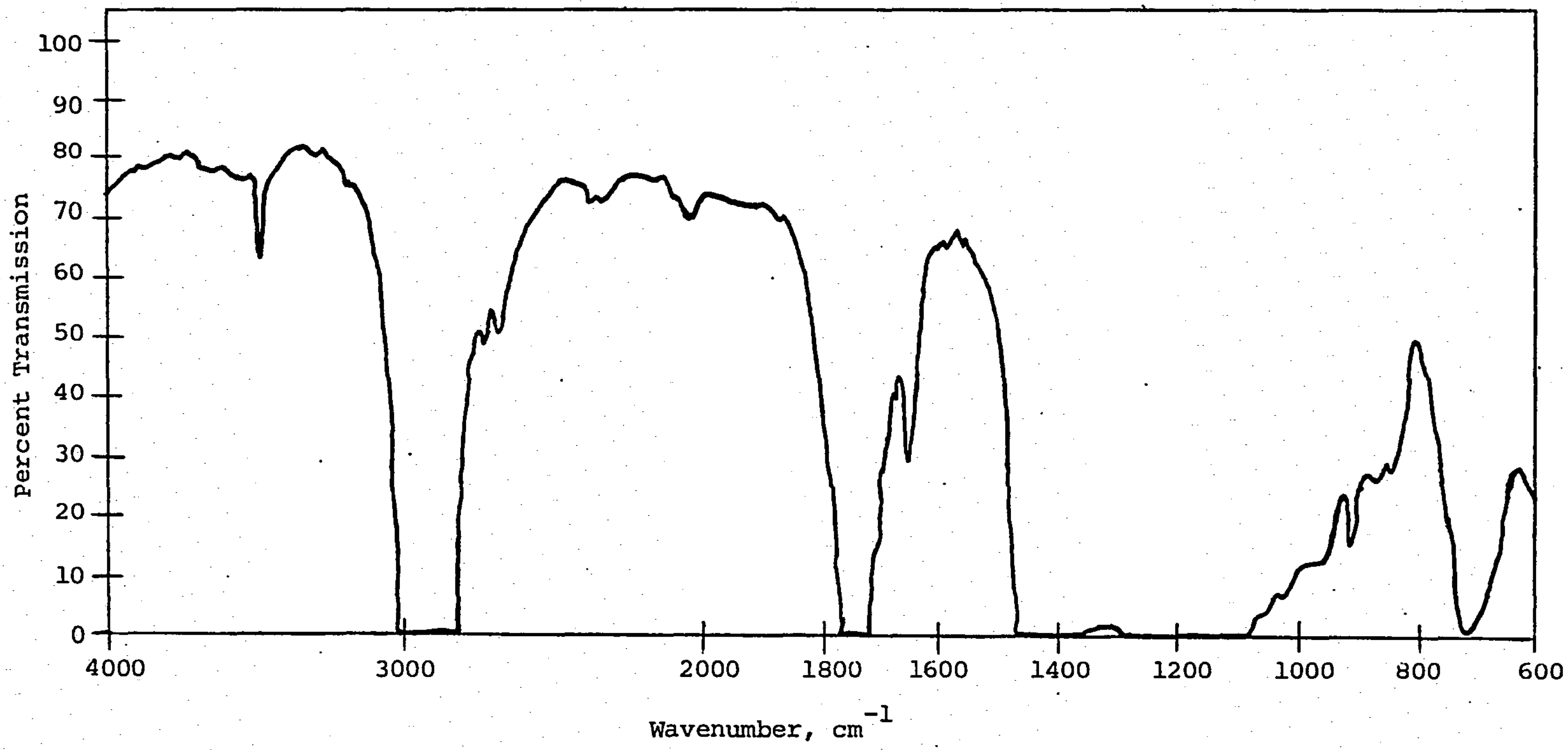


Figure 6. Infrared Spectrum of Soybean Oil

was 2 minutes.

The to maxima of an oil at 1654 and 1750 cm^{-1} overlap each other significantly. The partially resolved carbon-carbon double bond peak provides an excellent opportunity to employ the derivative technique. Derivative infrared spectroscopy should improve the detectability of the partially resolved peak at 1654 cm^{-1}. This should produce an improvement in quantitation of the iodine number of an oil.

CHAPTER II

DESIGN AND CONSTRUCTION

Infrared Spectrophotometer

The infrared spectrophotometer used in the study was a Beckman Acculab 8 (23,24). This instrument employs a rotating wedge filter and a 100-line/mm grating (used in two orders) to provide a spectral range of 4000 to 600 cm^{-1}. The order break of the scan is at 2000 cm^{-1}. The source of radiation for the double beam instrument is a nichrome wire. The resolution produced by the optics at 1000 cm^{-1} is 3 cm^{-1}. The detector which converts the radiant energy into electrical signals is a germaniun lens thermocouple.

The Acculab 8 is equipped with a connector terminal for limited analog/digital output and control. The optical system may be controlled through two lines which start, stop, and reverse the wavelength scan. Four output lines provide both an analog signal from 0 to 1 volts of the transmittance and a wavenumber pulsing signal from the optical system. All logic circuit communication to and from the terminal is in standard transistor-transistor logic (TTL). This terminal supplies enough information and control for data acquisition from the instrument.

Heathkit ET-3400 Trainer Interface

Hardware Description

Input

A Heathkit ET-3400 trainer (25) with an ETA-3400 memory accessory (26) was used in the initial study for data handling and instrument control. The ET-3400 contains a Motorola MC6800 as its central processing unit (CPU). The MC6800 is an eight-bit parallel mircoprocessor. The trainer provides 512 bytes of memory, but with the addition of an ETA-3400 accessory to the trainer the memory can be expanded to 4096 bytes. The ETA 3400 also provides an RS-232C interface for video terminal connection, a cassette recorder interface for data and program storage, and a Tiny Basic interpreter in ROM for higher level programming.

A programmable parallel input-output communication device (PIO) was used for data acquisition and instrument control in the interface system. The PIO used was the Motorola MC6820 peripheral interface adapter (PIA). A description of PIA operation is given in Appendix A. Thirteen lines of the two eight bit ports of the PIA 1 are employed to collect data through an analog-to-digital (A/D) converter from the spectrophotometer. Attached to each of the thirteen lines is an inverter circuit to convert the output of the A/D to TTL logic. The A/D converter provides a four bit binary coded decimal (BCD) number to PIA 1. This BCD number represents the transmittance from the Acculab 8 which is an analog signal between 0.000 and 1.000 volt. This provides transmittance readings as low as 0.1%. The A/D converter employed in the system was a Systron Donner Model 1234 integrating ramp type converter. A description of the A/D operation and the TTL conversion circuit is also provided in Appendix A.

The control of the interface system is achieved through four lines. Three lines from port A of PIA 1 and one line connected directly to the microprocessor. This is shown in Figure 7a, where the wavenumber scan inhibit (pin 3) and the A/D reset (pin 46) are controlled through lines 8 and 9 of PIA 1 respectively. A description of the switching circuit between line 9 and the A/D reset line is provided in Appendix A. The wavenumber pulse and the order break of the spectrophotometer are connected through a logic circuit to the interrupt request line ($\overline{IRQ}$) of the MC6800. This logic circuit provides wavenumber pulsing only when the monochromator is scanning. The wavenumber pulse from the monochromator is also connected to line 7 of PIA 1. This provides the MC6800 microprocessor with the wavenumber position information. Thus, the MC6800 need only take data during the wavenumber range of interest. The chart drive inhibit (pin 4) and the wavenumber reverse (pin 6) lines of the spectrophotometer are connected to common logic ground. This inhibits both lines as they are not used for data acquisition.

Output

After the acquisition of the data from the Acculab 8 spectrophotometer, the derivative was calculated within the MC6800. The derivative was observed using PIA 2 as an output device. Data from PIA 2 was directed to a digital-to-analog (D/A) converter, and subsequently stored in a digital storage oscilliscope. This way spectra were immediately viewed on the oscilliscope screen. A permanent recording of the spectra was later obtained using a X-Y recorder connected to the oscilloscope output. Figure 7b shows the schematic of the hardware connections for

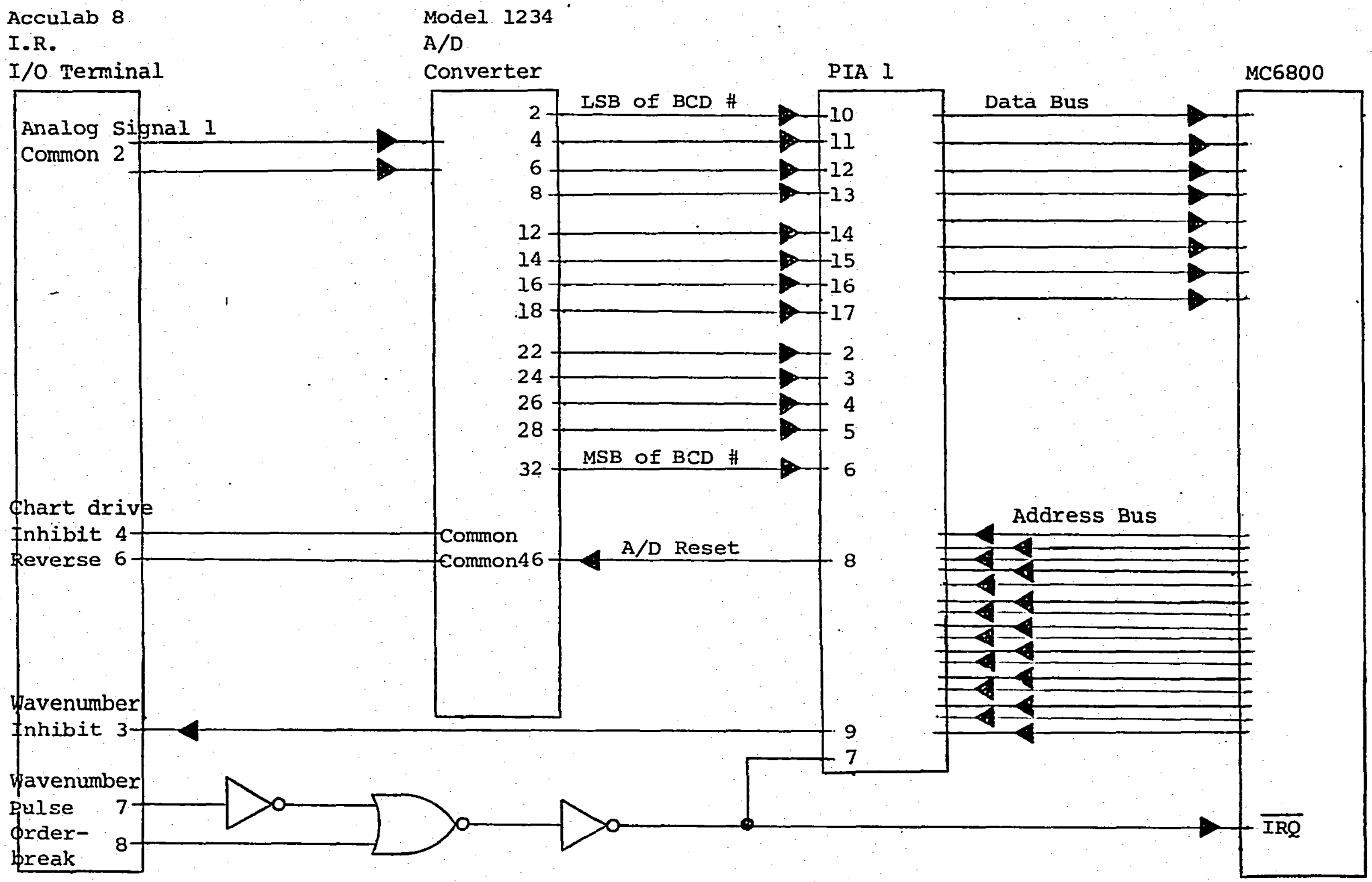


Figure 7a. MC6800 Microprocessor Interface System (Input Hardware)

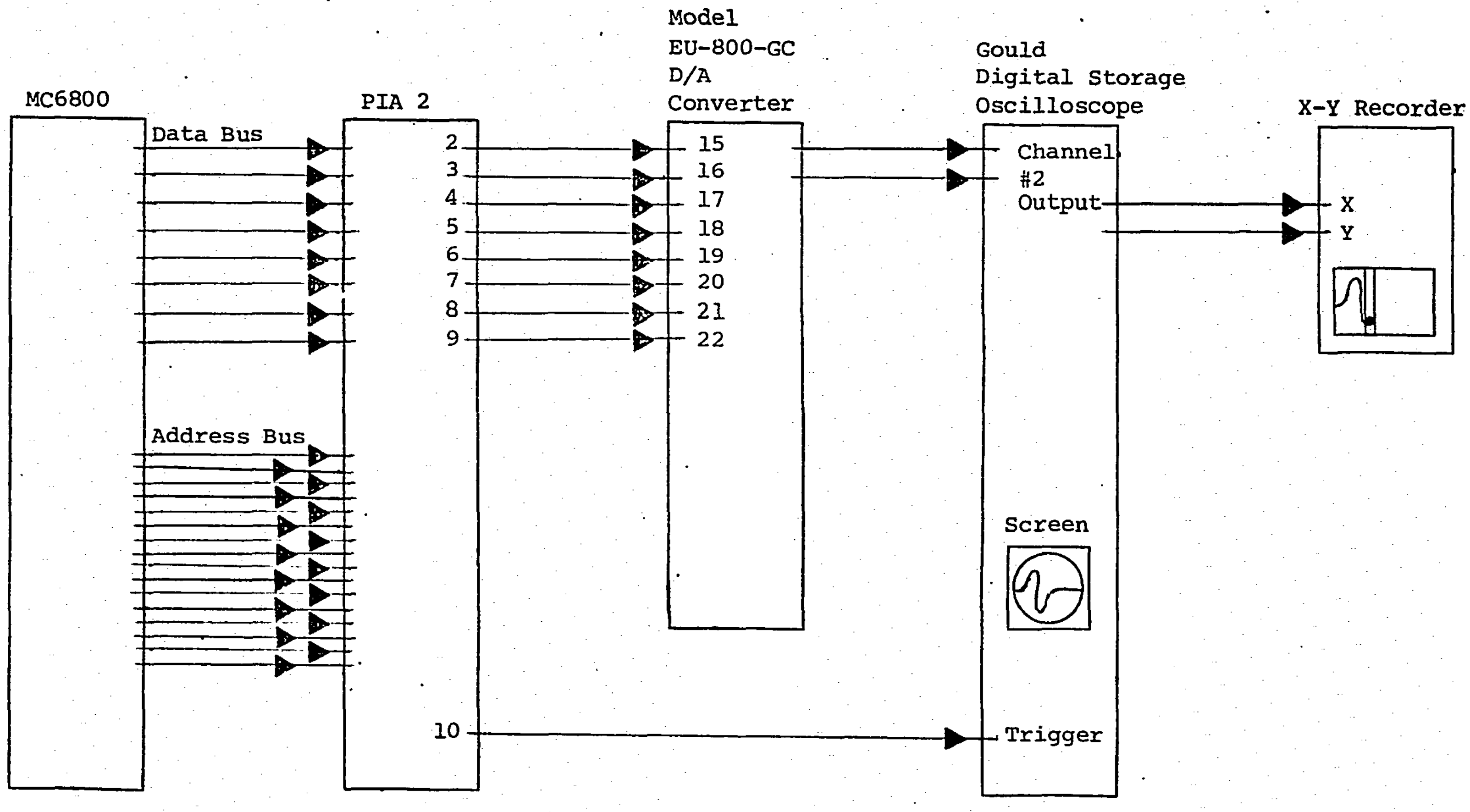


Figure 7b. MC6800 Microprocessor Interface System (Output Hardware)

the output of the derivative.

A Heath Model EU-800-GC D/A converter (27) was connected to the eight-bit port A of PIA 2. Only eight of the most significant bits of the 10-bit binary resistor ladder D/A converter were connected to PIA 2. The oscilloscope used was a Gould Model OS4000 digital storage oscilloscope (28) with the 4096 byte memory accessory. The oscilloscope is controlled by the microprocessor through port B (line 10) of PIA 2. A high signal on line 10 allows the trigger level of the oscilloscope to be adjusted manually. A low signal on line 10 allows the oscilloscope to receive data from the PIA through the D/A converter. This data is latched in the memory of the oscilloscope. By pressing the memory output of the oscilloscope a recording of the derivative could be obtained on a Princeton Applied Research Model RE0074 X-Y recorder (29).

Software Description

Input

The software for data acquisition and instrument control consists of a Tiny Basic program and four assembly language programs. The central basic program, called "Main Input", interfaces between the assembly language programs and the real world. As seen in Appendix B, Flowchart B-1, "3400 Input", the basic program sets the parameters for data acquisition. These parameters consist of the starting wavenumber of the instrument and the wavenumber range of interest. These are input through the keyboard. The basic program then branches to an assembly language program called "Input Control". This program sets the system for input

and starts the monochromator scan. It also counts the wavenumber pulses from the optical system. Once the beginning of the wavenumber range of interest has been reached, the program executes the Wait For Interrupt instruction.

A high status from the wavenumber pulse to the interrupt request ($\overline{\text{IRA}}$) line starts data acquisition. The $\overline{\text{IRA}}$ overrides all operations in the microprocessor and starts an interrupt routine. This interrupt routine searches addresses 00FA and 00FB of the RAM and loads their contents into the program pointer. This results in the starting of a new assembly level program called "Take Data". The "Take Data" program resets the A/D converter and acquires eight data points. It then branches to another assembly language program to convert the BCD data to hexadecimal. This subroutine, "BCD Convert", transforms the eight 4-bit data points to their hexadecimal equivalents. Subroutine "BCD Convert" then returns to the interrupt subroutine, "Take Data". After the data points are acquired, the "Take Data" program returns from the interrupt routine to the assembly level program "Input Control". The "Input Control" program checks for the end of the wavenumber range. If the data acquisition is complete, it returns to the basic program "Main Input". If the end of the wavenumber range is not reached, it returns to Wait For Interrupt and continues to take data.

The "Main Input" program branches to another assembly language program to stop the monochromator scan. This program, called "Stop Scan", resets the system to output and inhibits the instrument from scanning. The data points are located between memory positions 0720 to 0DFF. They can then be stored on a cassette recorder or displayed on the

monitor. Listings of all input programs may be found in Appendix B.

Data Processing

The data obtained from the infrared spectrophotometer represents the transmittance readings from the wavenumber scan. In order to average data points and calculate the derivative, transmittance values must be converted to absorbance values. The software for the conversion is compiled in Appendix B. The conversion is achieved using a transmittance-absorbance look-up table.

Referring to Appendix B, Flowchart B-2, "3400 Data Processing", the software consists of two assembly language programs and a basic program. The only function of the basic program is to branch to the assembly level program called "Transmittance-Absorbance Conversion" (TAC). This program converts a hexadecimal data point into a 4-bit decimal number. This decimal value depicts the percent transmittance ±0.1 percent. The program then branches to a subroutine called "Set Memory". This subroutine uses the decimal value to fix the corresponding absorbance value in a prestored table. It then returns to "TAC" for data point averaging. After the conversion of the data set is complete, the data is stored on a cassette recorder or displayed on a monitor.

Output

The software for outputting the data consists of four programs. As shown in Appendix B, Flowchart B-3, "3400 Output", a basic program accesses three assembly language programs. The first task in the

software is to calculate the first derivative of the absorbance data. This is achieved by branching to an assembly level program called "Derivative". This program calculates the first derivative and stores it in half the memory space needed for the original data. It then branches back to the basic program, "Main Output". This program sets the trigger of the storage oscilloscope for manual level adjustment. Once the trigger level is adjusted manually on the oscilloscope, the basic program branches to a second assembly language program. This program, called "Output Data", outputs the derivative from the PIA through the D/A converter to the memory of the oscilloscope. The derivative can be viewed immediately on the screen of the oscilloscope.

Higher order derivatives are calculated by branching from the basic program to an assembly level program called "Multiderivative". The order of the derivative of interest is entered through the basic program. The program "Multiderivative" then calculates the derivative and returns to the basic program. The basic program then reuses the assembly level program "Output Data" to send the derivative to the oscilloscope. This loop continues until the operator enters a stop code of 10 for the order of derivative. Listings of the output programs are contained in Appendix B.

Apple/Isaac Interface

Hardware Description

Input

A second interface system was developed in order to acquire data

from the infrared spectrophotometer. This system included the Apple II (30) as the microcomputer. The Apple II contains an eight-bit MOS Technology MCS6502 as its central processing unit. The MCS6502 is the most commonly used microprocessor of MOS Technology's 6500 series. The microcomputer system provides 48 kilobytes of RAM with an onboard basic interpreter called Applesoft. Two floppy disk drives were also connected to the Apple II. This provided data and program storage and retrieval. The disk drives expand the storage capabilities by 90 kilobytes. An 80-column Epson MX-80 parallel printer and a high resolution monochrome monitor were also employed for data printout and review.

The hardware for the interface system is that provided in the Cyborg Inc. Isaac 91A unit (31). This universal interface system contains all the electronic devices to achieve a computer interface to most real world devices. Contained within the Isaac are analog I/O, binary I/O, counter, and trigger devices. All these devices are controlled through software stored in the Apple II. This software is an extension of the resident Applesoft basic interpreter and is called Cyborg Inc. Labsoft (32). Labsoft takes up 8 kilobytes in the Apple II RAM.

As shown in Figure 8, to collect the data from the spectrophotometer I/O terminal, one of the sixteen 12-bit A/D converters in the Isaac unit was utilized. The A/D converters are successive approximation types. Three of the four binary resistor ladder D/A converters are connected to the Acculab 8 for instrument control. The "Counterin 2" of the Isaac unit is connected as an input from the order

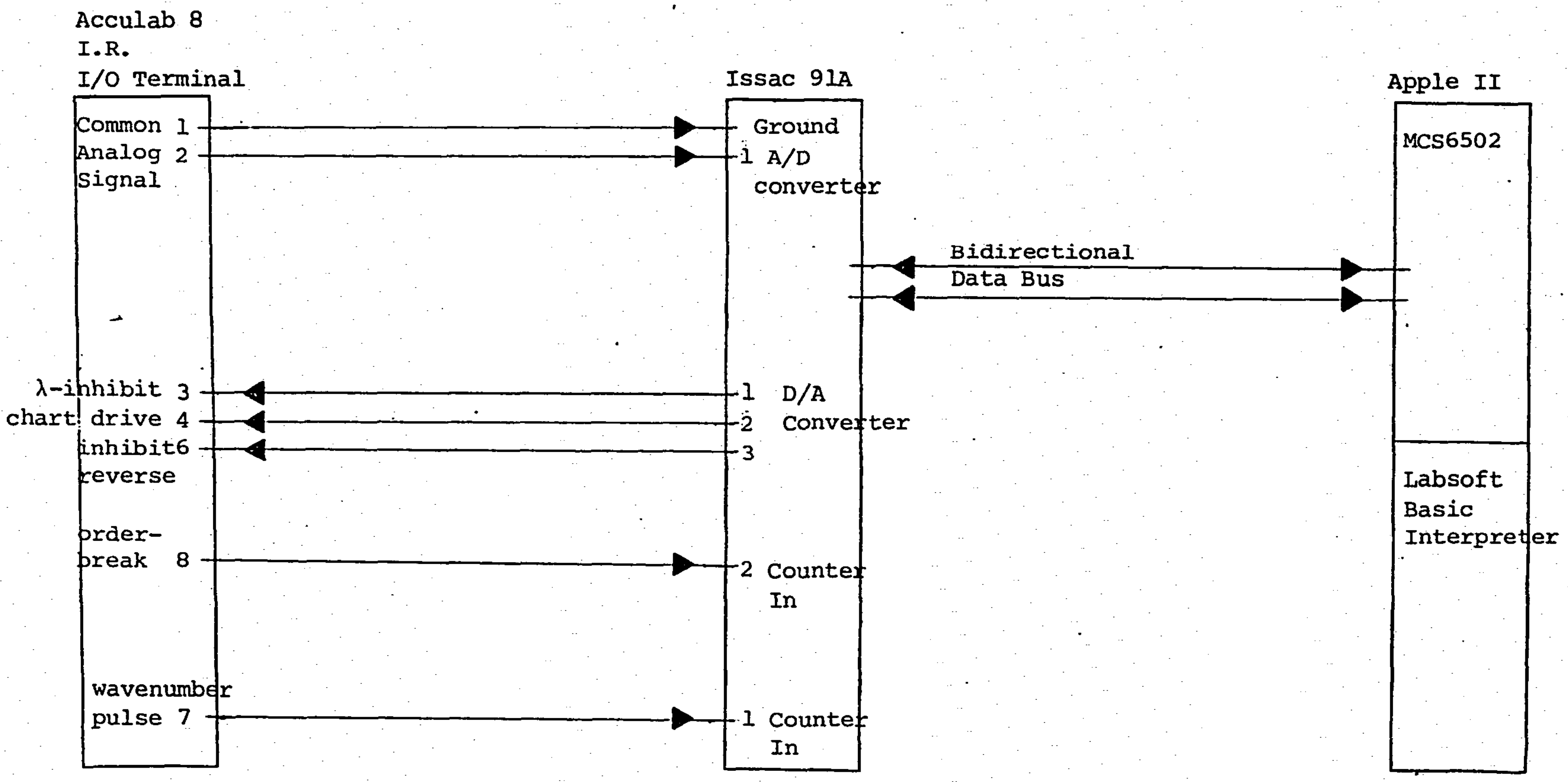


Figure 8. Apple/Isaac Interface System

break, line 8 of the spectrophotometer. Line 7, the wavenumber pulse of the Acculab 8 I/O terminal is connected to the "Counterin 1" of the Isaac unit for determining the optical system status. These counters are an externally clocked TTL up-counters. All these electronic devices are controlled by a basic program which utilizes the Labsoft interpreter.

Output

The output of the data involves the monitor and printer, which are already part of the Apple II system. The software to utilize these peripheral devices is described in the Output section in the Software Description.

Software Description

Input

Data acquisition and instrument control were achieved with one basic program. This program is called "Apple Input". "Apple Input" uses the Labsoft basic commands to control the Isaac unit for I/O operation. The ampersand (&) which preceeds all Labsoft commands is the Applesoft's expansion character. When the Applesoft interpreter encounters the ampersand, it branches to the Labsoft interpreter for proper operation direction. A listing of "Apple Input" program is located in Appendix C.

The "Apple Input" program sets the parameters for data acquisition. Referring to Appendix C, Flowchart C-1, "Apple/Isaac Input", the program then sets the instrument control lines to acquire data through the D/A converter. The optical system status is monitored by

the program through the "Counterin". This provides for data acquisition in the wavenumber range of interest. The data is acquired using a subroutine called "Acquire". In the subroutine, each data point is an average of ten data points. This is achieved by converting the transmittance data to absorbance during data acquisition. An average of the absorbance values is calculated immediately. The absorbance data is then stored by the "Apple Input" program on a disk for further analysis.

Output

The output and processing of the data is achieved through a basic program called "Data Analyzer". This program first loads the data from the disk drive. Data smoothing is provided in the program using the boxcar method. In this technique, a data point value is calculated based on the average of itself with two subsequent data point values. This data smoothing technique significantly decreases the noise in the data. After the data smoothing option, the program displays the wavenumber range of interest on the monitor. A copy of the absorbance values can then be produced on the printer complete with their corresponding wavenumber values.

The "Data Analyzer" program calculates any order derivative desired. An input from the keyboard of zero for the derivative order terminates the program. Once the derivative is calculated, it is displayed on the monitor and subsequently printed. Appendix C contains Flowchart C-2, "Data Analyzer" and a listing of the program.

Systems Comparison

The Apple/Isaac interface system has many advantages over the ET-3400 system. The hardware devices utilized in the connection to the spectrophotometer were essentially equivalent. The Isaac 91A unit's capabilities were not fully utilized in the interface system. A smaller unit, the model 41A, from the manufacture can be configured for the minimal hardware needs that are required.

The major difference in the two interface systems lies in the available memory of the microcomputer units. The Apple/Isaac system provides 20 times the onboard data storage, a faster access to mass storage, and a higher level programming language.

CHAPTER III

EXPERIMENTAL

Procedure

All data acquisition from the Acculab 8 was taken at the scan speed of 120 cm^{-1}/minute. The optics of the spectrophotometer were set in the order break to start all scans at 2000 cm^{-1}. The wavenumber range of interest was entered as 1750 to 1550 cm^{-1}. This range sufficiently covers the carbon-carbon band at 1654 cm^{-1}. A sodium chloride sample cell was used in the sample beam with a one millimeter cell path thickness. This cell was polished before each day of analysis. Each sample was scanned three times.

At the beginning of each day, the instrument's optical system was checked by scanning a polystyrene film versus air within the wavenumber range of interest. The 100 percent transmittance was set and the scan was taken. The data was checked immediately for the band at 1601.8 cm^{-1}. In all cases, the band was found to be within the instrument's wavenumber resolution of 3 cm^{-1}. Before the sample were scanned, the Acculab 8 was set at 2000 cm^{-1}. A blocking attenuator was used in the reference beam, while the sample cell was placed in the sample beam. The 100 percent transmittance was then set using the blocking attenuator. Between sample analysis, a cell cleaning procedure was followed. First, the cell was rinsed three times with chloroform. It was then dried

with a stream of dry nitrogen and rinsed several times with the next sample solution. Finally, the cell was filled with the sample and tightly capped. The outside of the cell was rinsed with chloroform and dried with nitrogen to remove all residual sample. Chloroform was used as a cleaning agent because it exhibits no absorbance bands in the wavenumber range of interest.

Chromophore Study I

The first study to utilize the computerized infrared spectrophotometer involved standard solutions containing the two chromophores of interest. 1-Hexene was used to simulate the carbon-carbon band found in an oil's spectrum at 1654 cm^{-1}. The 1-hexene absorbance maximum is at 1640 cm^{-1}. 2-Butanone was used to produce the carbon-oxygen band at 1750 cm^{-1}. Mixtures of these standards were prepared and analyzed. The concentration range of the 1-hexene was adjusted so that the intensity of the band produced simulated the carbon-carbon band intensities of some common oils. The range was between 0.05 and 0.25 milliliters of 1-hexene per milliliter of 2-butanone.

The purpose of this study was to determine which derivative maxima to use for the best quantitative results. This was achieved by performing linear least squares analysis on the plot of concentration of the 1-hexene versus the peak height of each derivative maxima. A list of these values is found in Table 2. Each peak height is the average of three sample scans.

Within each derivative, comparing the slopes of the analysis gives an indication of which maxima to use for quantitation. The maximum with

Table 2

Selected Derivative Spectral Values as a Function of C=C Chromophore Concentration

C=C (mL/mL sol)	Abs	$\frac{dA(S_1)}{d\bar{\nu}}$	$\frac{dA(S_2)}{d\bar{\nu}}$	$\frac{dA(S_3)}{d\bar{\nu}}$	$\frac{d^2A(S_2)}{d\bar{\nu}^2}$	$\frac{d^2A(S_3)}{d\bar{\nu}^2}$	$\frac{d^2A(S_4)}{d\bar{\nu}^2}$	$\frac{d^3A(S_1)}{d\bar{\nu}^3}$	$\frac{d^3A(S_2)}{d\bar{\nu}^3}$	$\frac{d^3A(S_3)}{d\bar{\nu}^3}$
0.050	0.133	0.0262	0.0683	0.0472	0.0302	0.0075	0.0305	0.0344	0.0215	0.0261
0.100	0.301	0.0333	0.104	0.0571	0.0429	0.0173	0.0353	0.0417	0.0381	0.0323
0.150	0.401	0.0328	0.125	0.0669	0.0453	0.0239	0.0416	0.0373	0.0328	0.0357
0.200	0.486	0.0371	0.150	0.0754	0.0485	0.0218	0.0451	0.0375	0.0371	0.0374
0.250	0.539	0.0395	0.168	0.0779	0.0476	0.0176	0.0526	0.0459	0.0414	0.0420
C.C.*	0.978	0.950	0.992	0.982	0.860	0.618	0.995	0.662	0.794	0.982
Y-Int	0.073	0.0266	0.049	0.0410	0.0308	0.0103	0.0248	0.0337	0.0225	0.0236
Slope	1.99	0.06	0.49	0.16	0.08	0.05	0.11	0.04	0.08	0.07

*C.C = Correlation Coefficient

the greatest slope response should give the best quantitation. As shown in Table 2, the first derivative maximum with the greatest slope response is (S_2). The degree of increase in slope response for maxima (S_2) as compared to (S_1) and (S_3) can best be visualized in the plot shown in Figure 9. With reference to Table 2, the second derivative maximum with the larger slope response is (S_4); yet maxima (S_2) slope value is not significantly smaller. Both maxima could be used for quantitation. The two maxima of the third derivative with the greatest slope response are (S_2) and (S_3). Maximum (S_2) is not particularly linear with respect to the ethylene concentration, as was evidenced by the correlation coefficient of the least squares analysis. Thus, the maxima of interest for further quantitative analysis studies are: first derivative (S_2), second derivative (S_2) and (S_4), and third derivative (S_3).

Chromopore Study II

The second study also involved standard solutions for simulation. In order to simulate the fatty acids of the oil more closely, n-butyl acetate was used as the standard for the carbon-oxygen band. This compound contains the ester group common to all oils. A longer chain alkene was also used to produce the carbon-carbon band of the oils at exactly 1654 cm^{-1}. The alkene used in this study was cyclooctene. The purpose of this investigation was to determine the relative independence of the quantitative derivative measurements from the concentration of the carbon-oxygen band. Ratios of the two standards were mixed. Four standard sets were prepared. In each standard set, the

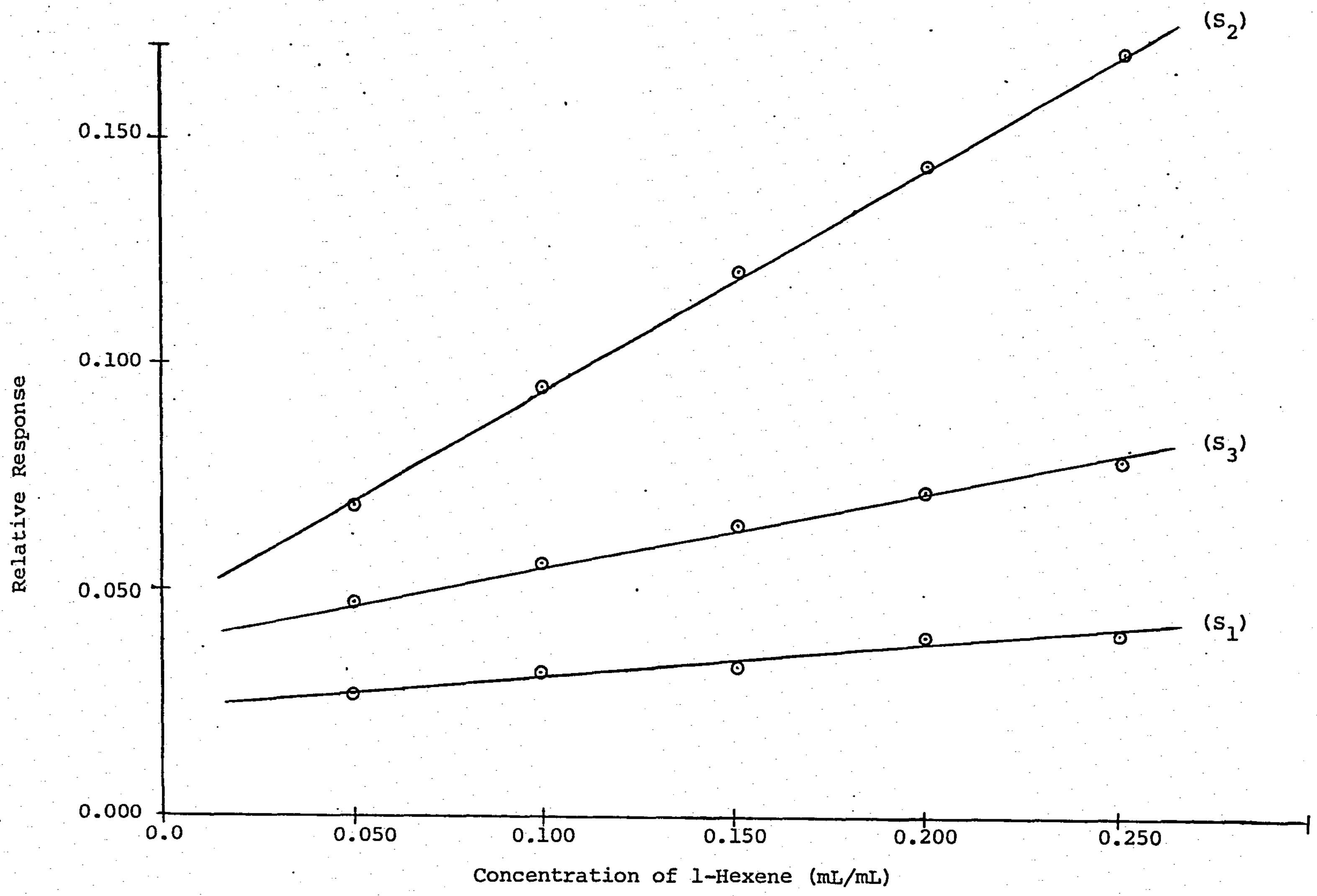


Figure 9. Plot of Concentration of 1-Hexene vs. First Derivative Maxima Values

volume of cyclooctene was varied from 1.0 to 2.5 milliliters while the volume of n-butyl acetate remained constant. The volume of the acetate for the four standard sets (1 through 4) were 8.0, 10.0, 12.0, and 14.0 milliliters, respectively. To assure the two liquids were additive, 20 milliliters of n-butyl acetate were pipetted into a 25 milliliter volumetric flask. Cyclooctene was added from a buret with mixing until the flask was full. The amount of cyclooctene added to the flask was found to be 4.95±0.05 milliliters. This error of 1.0% in the volume was not considered to be significant in these studies. Thus, within each standard set, the concentration of the acetate varies slightly over the cyclooctene range but the relative intensity of the band remains the same.

A linear least squares analysis was performed on the data from the plot of the concentration of cyclooctene versus the derivative maxima values. The resulting slopes from the analysis are compiled in Table 3. The variation in the slopes over the four standard sets represents the dependency of each derivative maxima value on the slope of the ester band. This variation is the percent mean deviation of the slopes contained in Table 3. The first derivative maximum (S_2) is the least affected by the ester concentration variation of butyl acetate. However, the second derivative maximum (S_4) and the third derivative maximum (S_3) seem to have been effected significantly by the ester concentration. This may not be entirely true since a marked loss in precision for higher derivatives may be obscuring the real trends. Nevertheless, this deviation in their slope could cause a large uncertainty in quantitative analysis. The mean deviations of

Table 3

Slope Values from the Least Squares Analysis of Concentration of C=C Chromophore vs. Selective Derivative Values

Standard Sample Set #	$\frac{\text{mL C=C}}{\text{mL COOR}}$	Slope Values and Coefficients of Variation: Abs	$\frac{dA(S_2)}{d\bar{\nu}}$	$\frac{d^2A(S_2)}{d\bar{\nu}^2}$	$\frac{d^2A(S_4)}{d\bar{\nu}^2}$	$\frac{d^3A(S_3)}{d\bar{\nu}^3}$
1	1.0/8.0					
	1.5/8.0	0.330	0.121	0.076	0.016	0.050
	2.0/8.0	(12.4)	(2.3)	(9.4)	(24.0)	(17.5)
	2.5/8.0					
2	1.0/10.0					
	1.5/10.0	0.412	0.106	0.092	0.029	0.070
	2.0/10.0	(19.5)	(4.9)	(22.3)	(17.1)	(26.1)
	2.5/10.0					
3	1.0/12.0					
	1.5/12.0	0.437	0.109	0.106	0.048	0.075
	2.0/12.0	(9.4)	(2.7)	(17.6)	(6.9)	(45.0)
	2.5/12.0					
4	1.0/14.0					
	1.5/14.0	0.359	0.113	0.092	0.045	0.125
	2.0/14.0	(16.5)	(10.8)	(20.5)	(19.8)	(19.8)
	2.5/14.0					
% Mean Deviation of Slopes		10.4	4.2	8.5	34.8	28.1
Average Coefficient of Variation ($\overline{CV}$)		14.4	5.2	17.4	17.0	27.1

the slopes for the absorbance peak height and the second derivative maxima (S_2) are each approximately twice the first derivative maxima (S_2) value. The first derivative would therefore be preferred in order to better accomplish quantitation of the carbon-carbon band.

The coefficients of variation (34) for the slopes in Table 3 is an indication of data scatter. The average coefficient of variation (CV) over the four standard sets is also listed in Table 3. This value shows an increase with increasing order of derivative which suggests that data scatter from the noise within the spectrophotometer is amplified as higher order derivatives are produced. Interestingly, the CV for the first derivative maximum (S_2) is significantly smaller than the CV for the standard base-line method employed on the absorbance peak. This is probably due to the uncertainty in establishing a reproducible base line in the absorbance mode.

Selected Oil Study

Six common unsaturated oils were purchased from local food retailers and subsequently studied. They were soybean, corn, olive, sunflower, safflower, and peanut oil. Each oil was scanned three times as a neat liquid. The determination of the iodine numbers of the oils were calculated based on Beer's Law relationship.

$$\frac{A_{std}}{C_{std}} = \frac{A_{unk}}{C_{unk}} \qquad (4)$$

where std = standard and unk = unknown. Also c = iodine number and A = derivative maxima value. All iodine numbers were based on a soybean oil as a standard sample. The oils were also analyzed by the Hanus

titration method (33). Experimentally determined iodine numbers of the selected oils are listed in Table 4. Each iodine value determined spectroscopically is the average from the three separate scans. Also contained within the tables are the pooled standard deviation (34) and the coefficient of variation for each derivative maximum studied.

As shown in Table 4, most of the spectroscopically determined iodine numbers fall within the accepted range of literature values, with the exception of olive oil. The olive oil iodine values were significantly high. In order to determine if the original olive oil was adulterated, a second olive oil sample produced by Fisher Chemical Company was analyzed. Again, the spectroscopically determined iodine numbers of the olive oil were still significantly higher than the accepted range. This implies that the technique was at fault and the original olive oil was not adulterated.

The probable cause for positively biased iodine numbers for olive oil may be due to the effect of the band shape resulting from overlapping bands of significantly different intensities. Figure 2 depicts two overlapping bands, which are significantly different in intensities. This situation is analogous to the spectrum encountered with oils where a very intense ester band (A in Figure 2) overlaps a very much weaker ethylene band (B in Figure 2). Since the spectral derivatives are dependent on the slope of the original spectra, hopefully some region of the spectrum (c to d in Figure 2) of the mixture may be found which is significantly dependent on the weaker component. Nevertheless, the degree of dependence will necessarily be a function of the ratio of the intensities of the major band (ester) to the minor

Table 4

Iodine Numbers of Selected Oils

Oil Name	Accepted Iodine Number Range(*)	Iodine Number ± Standard Deviation					
		Abs.	$\frac{dA(S_2)}{d\nu^{-2}}$	$\frac{d^2A(S_2)}{d\nu^{-2}}$	$\frac{d^2A(S_4)}{d\nu^{-2}}$	$\frac{d^3A(S_3)}{d\nu^{-3}}$	Hanus Titration
Soybean (std.)	122-134	128	128	128	128	128	128±1
Corn	111-128	120±7	129±4	139±6	129±4	129±16	128±1
Olive	79-88	108±5	132±2	138±23	102±5	114±2	78±1
Sunflower	122-134	126±7	128±7	119±12	131±5	126±7	126±1
Safflower	135-155	139±2	134±2	101±5	151±8	154±13	136±1
Olive (Fisher)	79-88	107±4	124±7	90±5	91±4	125±32	82±1
Standard Deviation Pooled (S_p)		1.58	1.48	2.26	1.61	2.64	0.71
Coefficient of Variation (CV)		1.32	1.14	1.92	1.33	2.04	0.63

*Reference 18

band (ethylene). In the case of olive oil, this ratio, due to the lower degree of unsaturation, is significantly larger than the corresponding ratio for soybean oil which was used as the standard. Thus it follows that the best results will be obtained when a standard is selcted such that its ratio more nearly approximates that of the sample. Another way of stating this is to say that the iodine number of the standard should be close to that of the sample. The validity of this argument is shown to be true in Table 5. The iodine number of peanut oil was determined using olive oil as the standard. Considering the standard deviations, all the iodine values of the peanut oil agree with the accepted literature range.

The iodine numbers determined by the derivative spectroscopic method tend to exhibit a high bias as compared to the titration method. The derivative maxima that best approximates values obtained by the titration method of all the oils is the first derivative maximum (S_2). This maximum also exhibits satisfactory precision as can be seen from the coefficient of variation in Table 4. The CV indicates the precision of the technique employed. The first derivative's CV exhibits the lowest value and thus suggests that the iodine number determined by the maximum (S_2) of the first derivative provides the best precision and consequently should be the maxima of choice.

Table 5

Iodine Numbers of Peanut Oil vs. Olive Oil

Oil Name	Accepted Iodine Number Range (*)	Iodine Number ± Standard Deviation					
		Abs.	$\frac{dA(S_2)}{d\bar{\nu}}$	$\frac{d^2A(S_2)}{d\bar{\nu}^2}$	$\frac{d^2A(S_4)}{d\bar{\nu}^2}$	$\frac{d^3A(S_3)}{d\bar{\nu}^3}$	Hanus Titration
Olive (std.)	79–88	78	78	78	78	78	78±1
Peanut	88–98	79±2	96±3	92±4	89±1	108±19	----

*Reference 18

CHAPTER IV

DISCUSSION

Advantages and Disadvantages

The main advantage of derivative infrared spectroscopy lies in the increased quantitative capabilities it provides relative to the standard quantitative techniques employed with infrared spectroscopy. The infrared region has always been very useful for qualitative analysis. However, its utilization for quantitative analyses enhances its power. In this investigation, it was shown that quantitation of iodine numbers of oils was accomplished quickly and accurately with the derivative technique. Many of the problems encountered in the standard titration method for determining iodine numbers are eliminated in the spectroscopic method. The typical analysis time for the computerized derivative infrared technique is approximately 2 minutes. This decreases the analysis time of the standard titration method used by food technologists by approximately 15 fold.

As a result of this investigation, it was found that the derivative technique is limited by the noise within the spectrophotometer. The inherent noise in the Acculab 8 infrared instrument limits the usefulness of the quantitative derivative technique to the first derivative. If higher signal-to-noise ratios could be achieved, as with Fourier Infrared Spectroscopy, higher order derivatives would

likely eliminate the problem of the choice of appropriate standard oils encountered in this study.

Conclusion

The infrared derivative spectroscopic technique provides a quantitative method that closely approximates the accepted method. The quantitative derivative method provides data with a relative error of about 1 percent. This technique permits quantitation in the infrared region at approximately the same level of precision and accuracy as the more elaborate computerized base-line correction methods.

This technique should be applicable to any compound or compounds containing measurable amounts of unsaturation. A good application would be in the agricultural area to measure total unsaturation of food products. Another industry involving unsaturation measurements would be the petroleum and allied industries. Many in-process checks involving unsaturation of the products can be accomplished quickly and accurately by employing the derivative I.R. technique.

Recommendations

The construction of the interface system used in these studies was built as a matter of convenience. A commercially built unit would likely use the optimum amount of hardware consistent with the quality of spectrophotometer employed. The best system might contain, within the instrument, a microprocessor instead of a full microcomputer. All the programming needed to control the instrument and to acquire and process data would be resident as firmware in the system. The minimum

RAM meeded to acquire data over the full spectral range would be approximately 16 kilobytes (8 bit). The data acquisition hardware would be comparable to that found in the ET-3400 interface system. The cost of such a spectrophotometer interface system would be significantly lower than the Apple/Isaac system employed in this study.

Appendix A

APPENDIX A

Peripheral Interface Adapter (PIA) Operation

The Motorola 6820 peripheral interface adapter (PIA) (35,36) has two input/output (I/O) ports each possessing three independent registers. These registers are shown in Figure A-1. They are referred to as: the output registers (ORA and ORB), the data direction registers (DDRA and DDRB), and the control registers (CRA and CRB). These six registers are each individually addressable by the microprocessor unit (MPU). The A and B indicate the two different ports of the PIA. The PIA provides two parallel eight-bit ports (A and B) which can be programmed as either an input, output, or special function port.

Figure A-2 shows a diagram of the MPU system interfaced with the PIA. The MPU side of the total system includes three chip select lines CS0, CS1, and $\overline{\text{CS2}}$ for selecting a particular PIA. These lines, along with two register select lines, RS0 and RSI, are connected to the address bus of the MPU. RS0 and RS1 are used in conjunction with four interrupt control bits, CA1, CA2, .CB1, and CB2 to address the six registers within the PIA. The four interrupt lines are held at a high state in the interface by connecting each to the five volt supply through a one thousand ohm resister. This is to prohibit any outside interruption of these lines which would alter the PIA function. The read/write (R/W) and reset ($\overline{\text{RES}}$) lines of the PIA are connected to the same corresponding lines of the MPU. The enable line of the PIA is pulsed by the 02 clock signal of the MPU. Lines $\overline{\text{IRQA}}$ and $\overline{\text{IRQB}}$

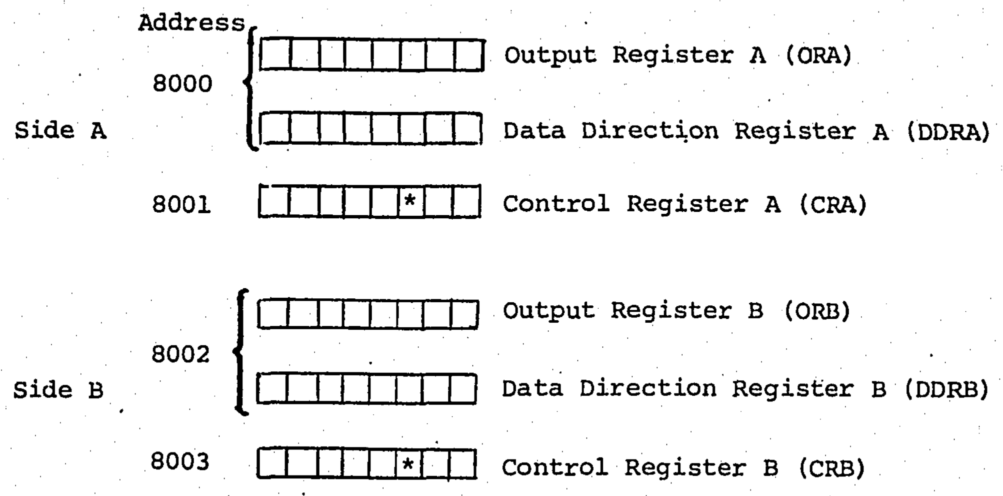


Figure A-1. Address Assignments of PIA Registers

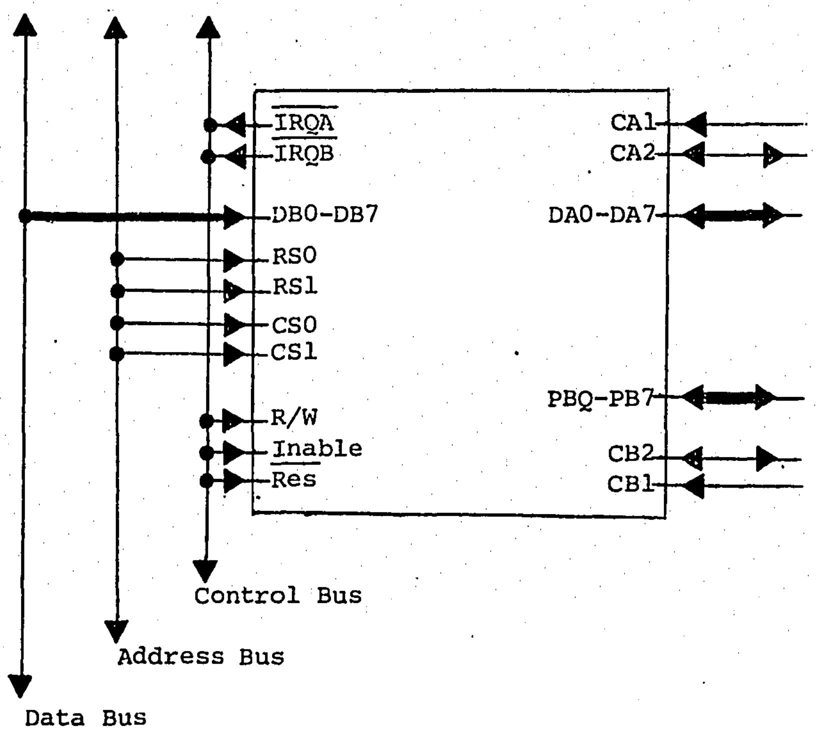


Figure A-2. PIA Interface

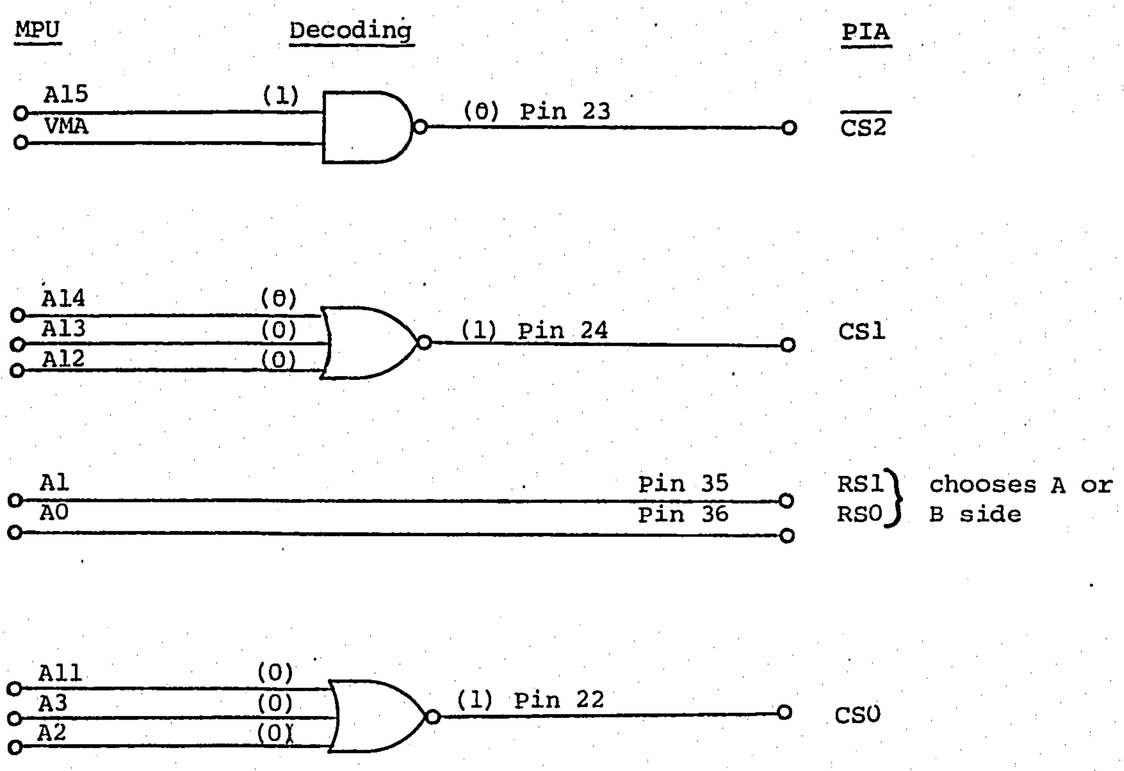
are interrupt request lines for each side of the PIA. These lines are connected to the respective $\overline{\text{IRQ}}$ line of the MPU.

The logic scheme for addressing the PIA is shown in Figure A-3(a). Address memory locations 8000 to 8003 represent the two sets of independent register ports. These addresses on the MPU's address bus initialize the chip select lines CS0, CS1, and $\overline{\text{CS2}}$. The two register select lines RS0 and RS1 are connected to the last two significant bits of the address bus. Thus, addresses 8000 and 8001 control port A of the PIA; while addresses 8002 and 8003 control port B of the PIA.

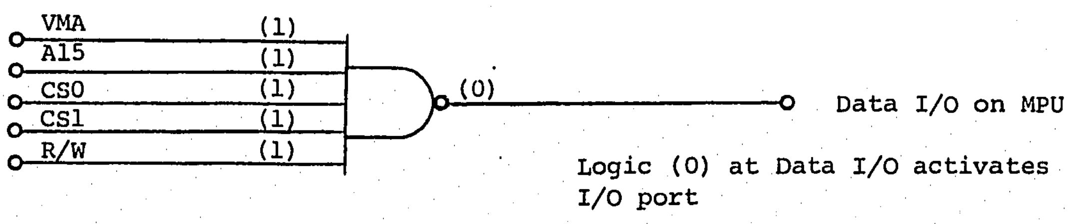
Figure A-3(b) shows the logic circuit that enables the data input/output bus. When the valid memory address (VMA), the address line 15 (A15), the control select lines, CS0 and CS1, and the R/W lines are all high status, the NAND gate enables the read enable line ($\overline{\text{RE}}$). Once the $\overline{\text{RE}}$ line is enabled, the tristate data bus from the PIA to the MPU is opened which makes data transfer possible.

Referring to Figure A-3(a), the two register select lines, RS0 and RS1, control the operation of the PIA. Line RS1 selects which port, A or B, of the PIA will be addressed. A high status on RS1 selects port B, while a low status on RS1 selects port A. Line RS0 selects which register is affected by the MPU. When RS0 is at high status, the CRA or CRB is selected. A low status on line RS0 selects the DDRA or DDRB.

To initialize the PIA, the MPU must first set the control register to either output or input. An address of S001 on the address bus sets the RS1 line at low status. This selects A port of the PIA and sets RS0 at high status, thus initializing the CRA. Bit number 2 of the



(a) Addressing of PIA Decoding



(b) Data Input/Output Decoding

Figure A-3. Addressing to PIA

CRA is the control bit for the selection of input or output for port A. Bit number 2 is at high status by loading the hexidecimal number 04 into memory location 8001. This selects the DDRA as an input register. The data in the DDRA is located at memory location 8000. The MPU only needs to read memory location 8000 to receive the data. For the MPU to output data, the ORA has to be initialized. This is done by setting bit number 2 of the CRA to low status by loading 00 into memory location 8001. The MPU can now output data by writing data into memory location 8000. The PIA essentially becomes four memory locations for data transfer in and out of the MPU.

Analog-to-Digital Converter Operation (Systron Donnor Model 1234)

Model 1234 analog-to-digital (A/D) converter (37) is an intergrating ramp type converter. The converter contains five major electronic components. These are a voltage ramp generator, two voltage comparators, a crystal oscillator, and a counter. The voltage generator consists of a current source and an operational amplifier in the integrator configuration. The generator provides a linearily decreasing voltage ramp from +12 to -12 volts. Referring to the schematic of the converter in Figure A-4, the operation of the converter starts with a high status at the reset line. This signals the voltage ramp generator to start and also resets the counter to zero. The ramp voltage is connected to the inputs of the two comparators. The ramp voltage is compared to zero volts by comparator (1) and to the input voltage through comparator (2). At the time the ramp voltage is equal to zero a pulse is generated at the output of comparator (1). This output

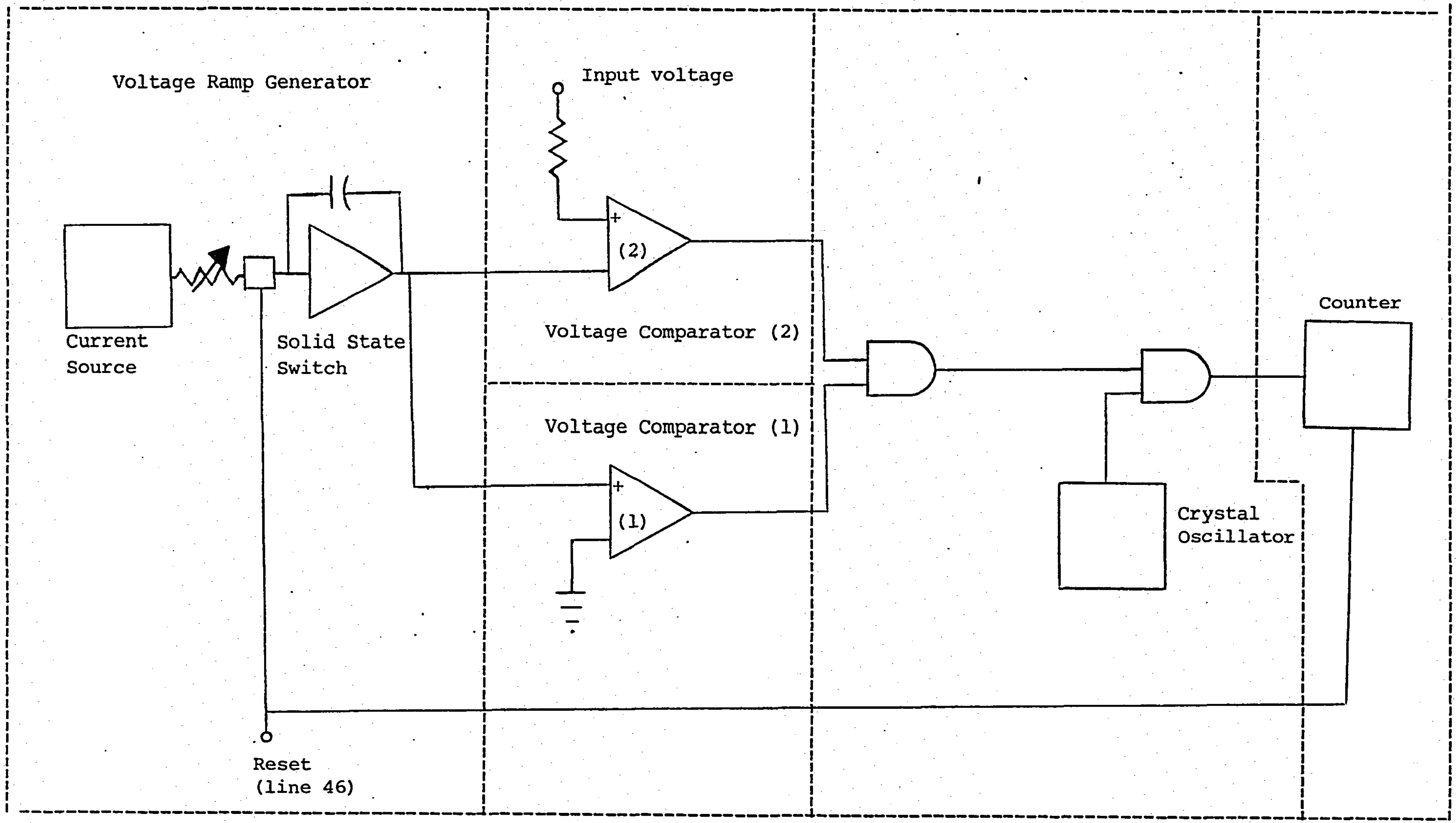


Figure A-4. Model 1234 Analog-to-Digital Converter

gates the crystal oscillator to the counter. The counter continues to count the oscillations until a negative signal occurs at the input of comparator (2). This is caused when the voltage ramp reaches the negative value of the input voltage. The digital value, a four bit BCD number in the counter, is proportional to the analog input voltage.

TTL Conversion Circuit Operation

The output of the Model 1234 A/D converter is not TTL compatible. To accomodate this, the electrical circuit shown in Figure A-5 was constructed. One of these conversion circuits was connected to each

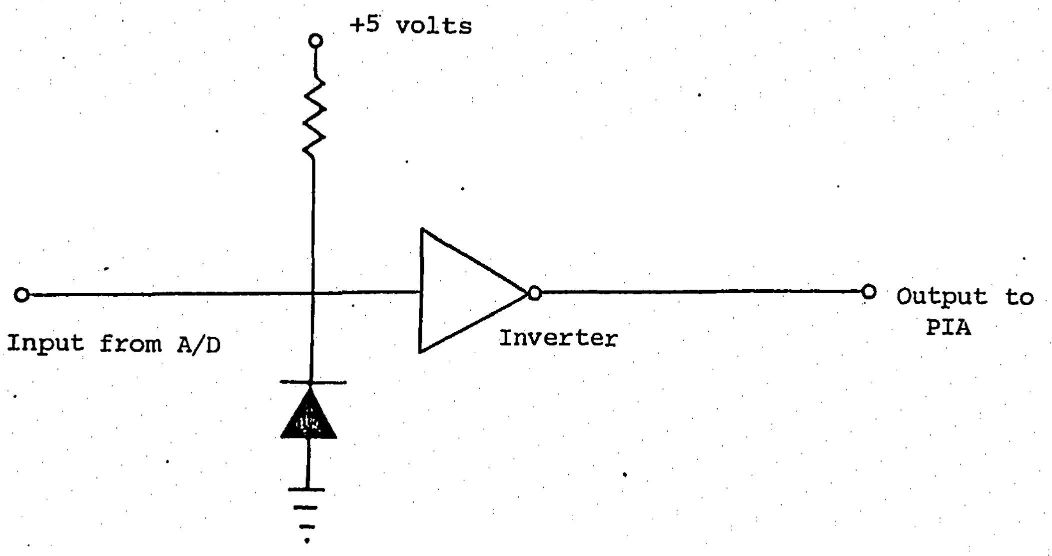


Figure A-5. TTL Conversion Circuit

of the thirteen lines between the A/D converter and the PIA. The device consisted of an inverter, a diode, and a 1 megohm resistor. When a "logic 1" from the A/D converter (-15 volts) is input, the diode conducts to ground. This connects the input of the inverter to ground also. This causes the output of the inverter to have a high TTL status of

approximately 3.4 volts. A "logic 0" (0 volts) from the A/D converter prevents the diode from conducting. This allows the +5 volts source to float the input of the inverter to +5 volts causing a low TTL status, zero volts, at the input of the inverter.

Reset Switch Operation

The switch (38) used between the PIA and the reset line of the A/D converter is illustrated in Figure A-6. It consists of a PNP type

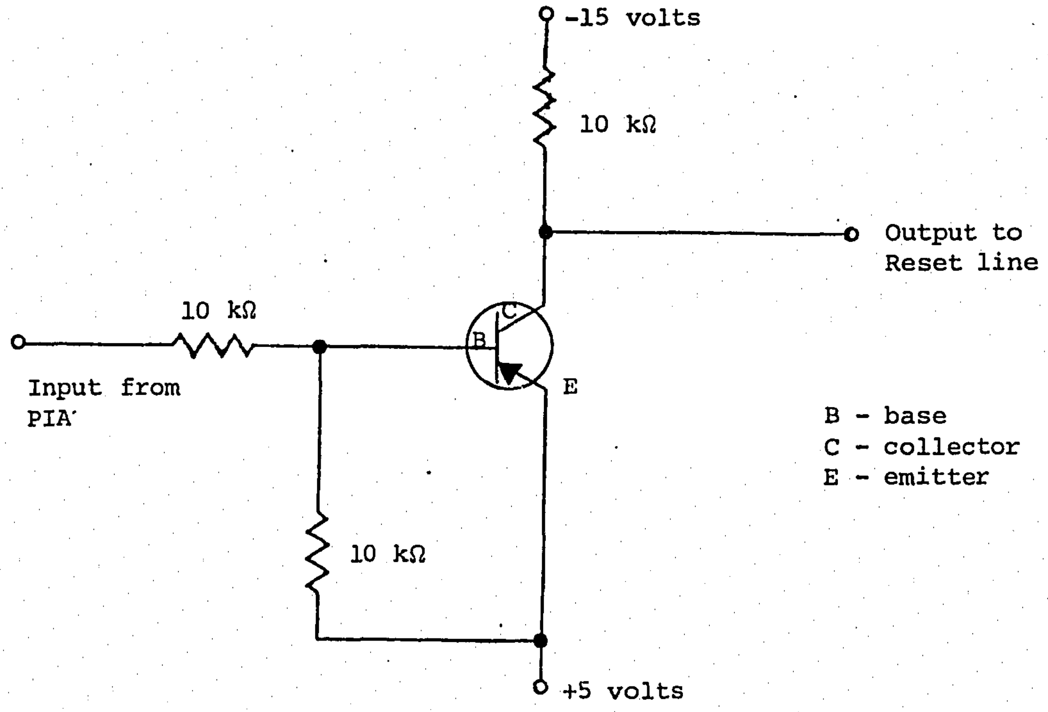
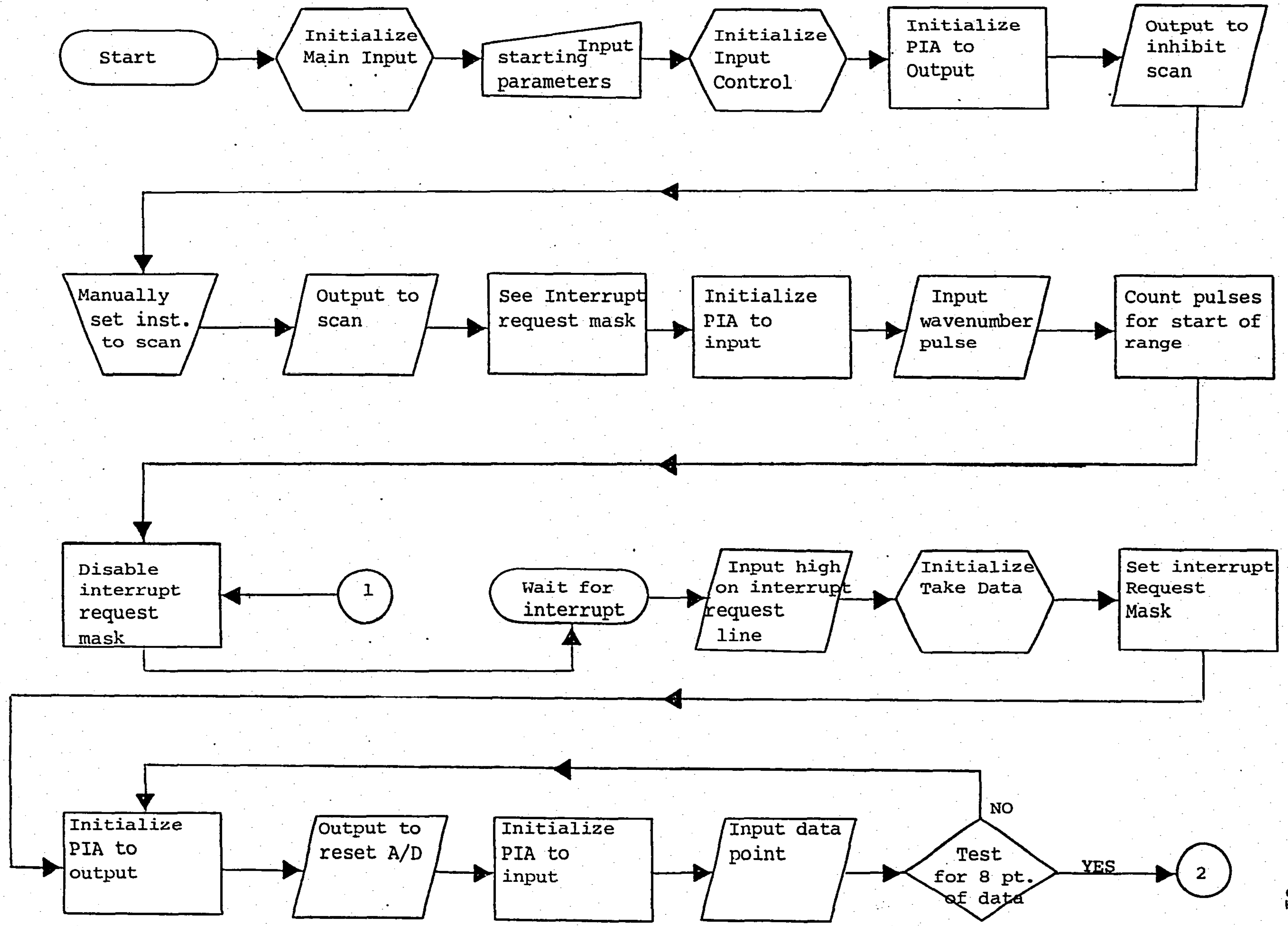


Figure A-6. Reset Switch Circuit

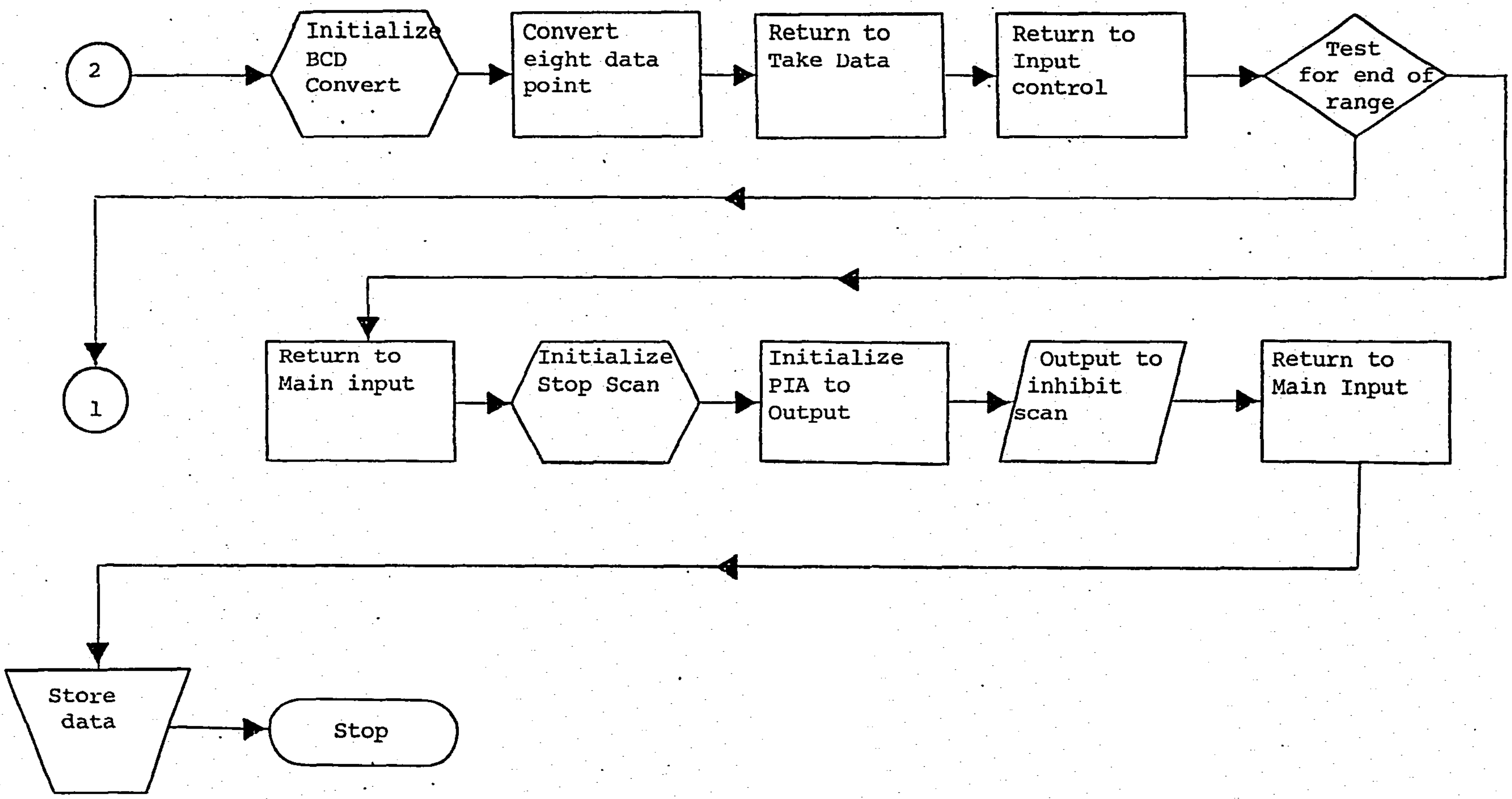
transistor, which functions as a switch; ans three 10 kilohm resistors. An input from the PIA of zero volts opens the base of the transistor. This drains the -15 volt source across the transistor and provides an output to the reset line of zero volts. To reset the A/D converter the PIA outputs a high status of +5 volts. This closes the base of

the transistor and opens the -15 volt source to the reset line.

Appendix B



Flowchart B-1. 3400 Input



Flowchart B-1 cont.

"Main Input" (Tiny Basic)

```
10 PRINT "INPUT STARTING WAVENUMBER OF SCAN IN CM^-1"
15 PRINT "INPUT WILL BE 4000 CM^-1 OR 2000 CM^-1"
20 INPUT A
30 PRINT "INPUT WAVENUMBER RANGE TO BE STUDIED IN CM^-1"
40 PRINT "WAVENUMBER (MAX.)"
50 INPUT B
60 PRINT "WAVENUMBER (MIN.)"
70 INPUT C
80 PRINT "MAKE SURE THE I.R. IS SET AT",A,"CM^-1"
90 PRINT "THE AREA TO BE RECORDED IS BETWEEN"
100 PRINT B,"CM^-1 AND",C,"CM^-1"
110 IF A=4000 GOTO 140
120 D=A-B
130 GOTO 150
140 D=(A-B)/2
150 LET G=0
160 IF D<256 GOTO 230
170 D=D-256
180 G=G+1
190 GOTO 160
230 D=USR(7192,0001,D)
240 G=USR(7192,0000,G)
250 IF A=4000 GOTO 280
260 E=B-C
270 GOTO 290
280 E=(B-C)/2
290 LET G=0
300 IF E<256 GOTO 370
310 E=E-256
320 G=G+1
360 GOTO 300
370 E=USR(7192,0003,E)
380 G=USR(7192,0002,G)
385 PRINT "NOW, START THE I.R."
386 PRINT "BY PRESSING THE SCAN BUTTON"
387 J=USR(1680)
388 PRINT "YOU MAY STOP THE SCAN NOW"
400 K=USR(1456)
410 PRINT
420 PRINT "NOW, ENTER NEXT PROGRAM FROM RECORDER"
430 END
```

"Input Control" (Assembly)

0690	LDAA	86 07	Load memory locations with initialization data.
0692	STAA	B7 00 04	
0695	LDAA	86 20	
0697	STAA	B7 00 05	
069A	SEI	0F	Set interrupt mask so wavenumber pulse does not interrupt.
069B	LDAA	B6 80 00	Load from PIA wavenumber inhibit status.
069E	LSRA	44 44 44	Check wavenumber inhibit status.
06A1	LSRA	44 44 44	
06A4	LSRA	44	
06A5	TSTA	4D	
06A6	BEQ	27 16	Branch if no wavenumber inhibit.
06A8	NOP	01	
06A9	LDAA	86 04	Set PIA port A to input.
06AB	STAA	B7 80 01	
06AE	LDAA	B6 80 00	Load PIA port A into Accum. A.
06B1	LDAB	C6 40	
06B3	SBA	10	Check wavenumber pulse status.
06B4	LSRA	44 44 44	
06B7	LSRA	44 44	
06B9	TSTA	4D	
06BA	BEQ	27 17	Branch to start scan.
06BC	BRA	20 EB	
06BE	LDAA	86 04	Set PIA port A to input.
06C0	STAA	B7 80 01	
06C3	LDAA	B6 80 00	Load PIA port A into Accum. A.
06C6	LDAB	C6 C0	
06C8	SBA	10	Check wavenumber pulse status.
06C9	LSRA	44 44 44	
06CC	LSRA	44 44	
06CE	TSTA	4D	
06CF	BEQ	27 02	Branch to start scan.
06D1	BRA	20 EB	
06D3	LDAA	86 00	Set PIA port A to output.
06D5	STAA	B7 80 01	
06D8	CLI	0E	Clear interrupt mask.
06D9	WAI	3E	Wait for wavenumber pulse to interrupt, branch to interrupt subroutine Take Data.
06DA	SEI	0F	Check wavenumber range covered.
06DB	LDX	CE 00 00	
06DE	CPX	BC 00 02	
06E1	BEQ	27 02	If spectra range is covered return to Tiny Basic.
06E3	BRA	20 F3	If spectra range is not covered return to wait for interrupt or wavenumber pulse.
06E5	RTS	39	Return to Main Input program.

"Take Data" (Assembly)

```
0600  LDX   CE 00 00  Check wavenumber status.
0603  CPX   BC 00 00
0606  BEQ   27 02     Branch to acquire data.
0608  BRA   20 73     Branch to input control.
060A  LDX   CE 00 00  Check wavenumber status.
060D  CPX   BC 00 02
0610  BEQ   27 63     Branch to input control.
0612  LDX   FE 00 02  Set current memory location
0615  DEX   09        for data entry.
0616  STX   FF 00 02
0619  CLR   7F 00 0C  Initialize memory locations.
061C  CLR   7F 00 0D
061F  LDAB  C6 08     Set number of data point
0621  STAB  F7 00 0E  to be taken.
0624  LDAA  86 00     Set PIA port A to output.
0626  STAA  B7 80 01
0629  LDAA  86 40     Reset A/D converter.
062B  STAA  B7 80 00
062E  LDX   CE 01 F8
0631  DEX   09
0632  BNE   26 FD
0634  LDX   FE 00 04
0637  LDAA  86 00     Reset output status.
0639  STAA  B7 80 00
063C  LDAA  86 04     Set PIA ports A and B
063E  STAA  B7 80 01  to input.
0641  STAA  B7 80 03
0644  LDAA  B6 80 00  Store A/D converter BCD
0647  STAA  A7 00 08  number.
064A  LDAA  B6 80 02
064D  STAA  A7 00 08
0650  JSR   BD 0E 00  Branch to subroutine BCD
                      Convert.
0653  LDAA  B6 00 0E
0656  DECA  4A
0657  BEQ   27 05     Check number of data points.
0659  STAA  B7 00 0E
065C  BRA   20 C6
065E  LDX   FE 00 04  Store converted data into
0661  LDAA  B6 00 0C  proper memory location.
0664  STAA  A7 00 08
0667  LDAA  B6 00 0D
066A  STAA  A7 00 08
066D  STX   FF 00 04
0670  LDAA  86 00     Set PIA port A to output.
0672  STAA  B7 80 01
0675  LDX   CE 78 00  Timer loop for data acquire.
0678  DEX   09
0679  BNE   26 FD
067B  BRA   20 0D
```

```
067D    LDX    CE 80 00    Timer loop for no data.
0680    DEX    09          acquire.
0681    BNE    26 FD
0683    LDX    FE 00 00    Decrement wavenumber status.
0686    DEX    09
0687    STX    FF 00 00
068A    RTI    3B          Return  from interrupt
                           to Input Control program.
```

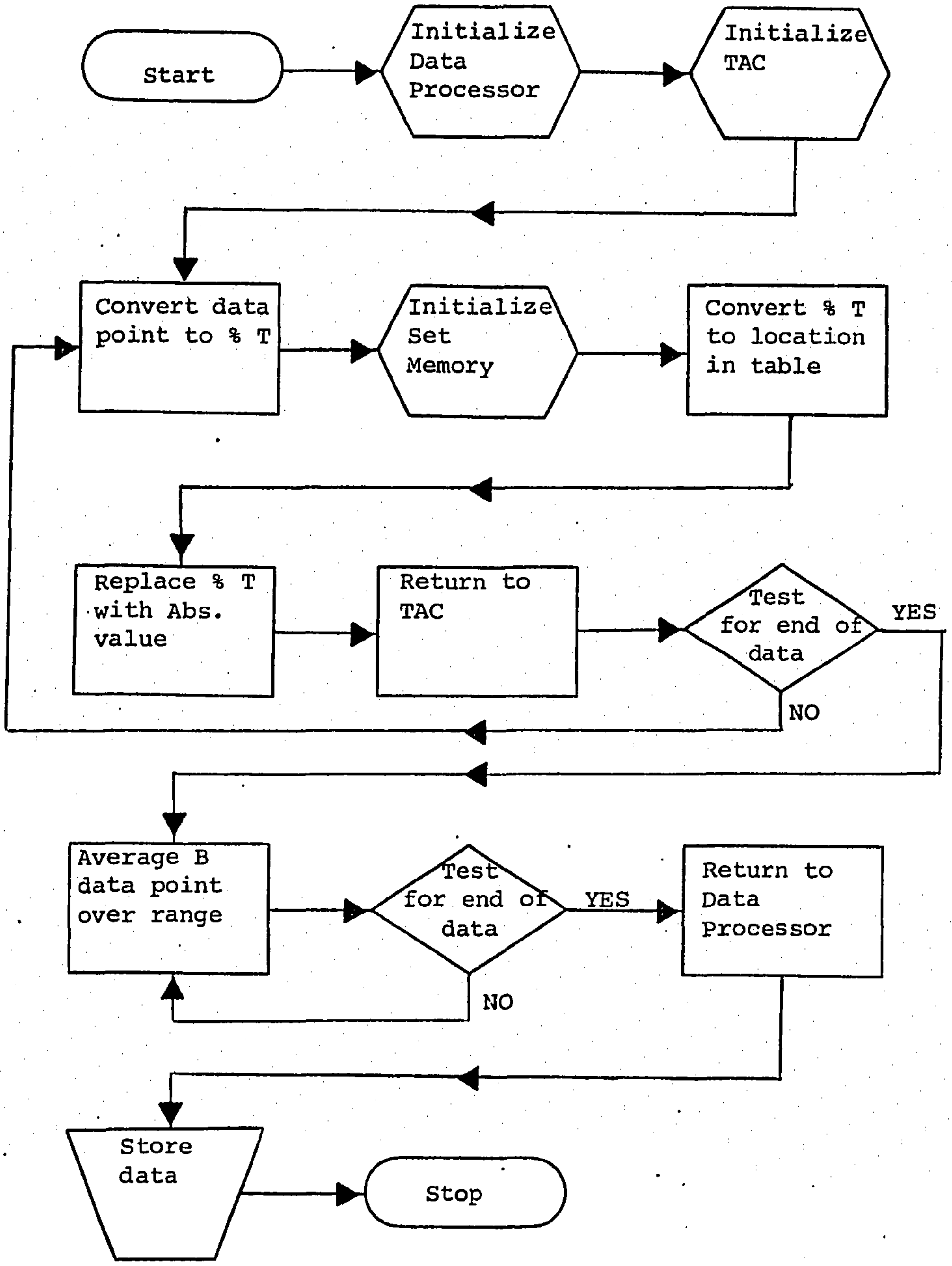
"BCD Convert" (Assembly)

```
0E00   LDX    FE 00 04   Move data to new memory
0E03   STX    FF 00 06   location.
0E06   INT    08
0E07   STX    FF 00 08
0E0A   NOP    01
0E0B   CLR    7F 00 0A   Initialize memory locations.
0E0E   CLR    7F 00 0B
0E11   LDX    FE 00 06   Check status of BCD number
0E14   LDAA   A6 00      bit 4.
0E16   ASLA   48 48 48
0E19   LSRA   44 44 44
0E1C   LSRA   44 44 44
0E1F   LSRA   44
0E20   TSTA   4D
0E21   CLC    0C
0E22   BEQ    27 07
0E24   NOP    01
0E25   LDX    CE 03 E8   Add hexadecimal equivalent
0E28   STX    FF 00 0A   of BCD bit 4 to memory.
0E2B   LDX    FE 00 06   Check status of BCD munber
0E2E   LDAA   A6 00      bit 3.
0E30   ASLA   48 48 48
0E33   ASLA   48
0E34   LSRA   44 44 44
0E37   LSRA   44
0E38   TSTA   4D
0E39   CLC    0C
0E3A   BEQ    27 17
0E3C   DECA   4A
0E3D   LDAB   C6 64      Add hexadecimal equivalent
0E3F   ADDB   FB 00 0B   of BCD bit 3 to memory.
0E42   STAB   F7 00 0B
0E45   BCS    25 02
0E47   BRA    20 EF
0E49   LDAB   C6 01
0E4B   ADDB   FB 00 0A
0E4E   STAB   F7 00 0A
0E51   BRA    20 E5
0E53   BRA    20 02
0E55   BRA    20 B7
0E57   LDX    FE 00 08   Check status of BCD number
0E5A   LDAA   A6 00      bit 2.
0E5C   LSRA   44 44 44
0E5F   LSRA   44
0E60   TSTA   4D
0E61   CLC    0C
0E62   BEQ    27 17
0E64   DECA   4A
0E65   LDAB   C6 0A      Add hexadecimal equivalent
0E67   ADDB   FB 00 0B   of BCD bit 2 to memory.
```

```
0E6A     STAB    F7 00 0B
0E6D     BCS     25 02
0E6F     BRA     20 EF
0E71     LDAB    C6 01
0E73     ADDB    FB 00 0A
0E76     STAB    F7 00 0A
0E79     BRA     20 E5
0E7B     LDX     FE 00 08    Check status of BCD number
0E7E     LDAA    A6 00       bit 1.
0E80     ASLA    48 48 48
0E83     ASLA    48
0E84     LSRA    44 44 44
0E87     LSRA    44
0E88     NOP     01
0E89     CLC     0C
0E8A     ADDA    BB 00 0B
0E8D     STAA    B7 00 0B
0E90     BCS     25 02
0E92     BRA     20 09
0E94     LDAB    C6 01       Add hexadecimal equivalent
0E96     ADDB    FB 00 0A    of BCD bit 1 to memory.
0E99     STAB    F7 00 0A
0E9C     CLC     0C
0E9D     LDAA    B6 00 0B
0EA0     ADDA    BB 00 0D
0EA3     STAA    B7 00 0D
0EA6     BCS     25 02
0EA8     BRA     20 09
0EAA     LDAB    C6 01
0EAC     ADDA    FB 00 0C
0EAF     STAB    F7 00 0C
0EB2     CLC     0C
0EB3     LDAA    B6 00 0A
0EB6     ADDA    BB 00 0C
0EB9     STAA    B7 00 0C
0EBC     RTS     39          Return to Take Data interrupt
                             subroutine.
```

"Stop Scan" (Assembly)

```
05B0    LDAA   86 00       Set PIA port A to output.
05B2    STAA   B7 80 01
05B5    LDAA   86 80       Inhibit instrument scan.
05B7    STAA   B7 80 00
05BA    RTS    39          Return from subroutine to
                           Tiny Basic program Main
                           Input
```



Flowchart B-2. 3400 Data Processing

"Data Processor" (Tiny Basic)

```
10 PRINT "THIS PROGRAM CONVERTS TRANSMITTANCE TO ABSORBANCE"
20 H=USR(1536)
30 PRINT "ENTER NEXT PROGRAM FROM RECORDER"
40 END
```

"TAC" (Assembly)

```
0600   LDX    CE 07 20   Initialize temporary memory
0603   STX    FF 00 06   for beginning of the data.
0606   LDX    CE 07 21
0609   STX    FF 00 08
060C   LDX    CE 0F 90
060F   STX    FF 0F 80
0612   LDAB   C6 04
0614   STAB   F7 0F 98
0617   LDX    FE 00 08   Convert binary LSB's of data
061A   LDAA   A6 00      to decimal percent
061C   LDAB   C6 44      transmittance equivalent.
061E   SBA    10
061F   STAA   A7 00
0621   BCS    25 02
0623   BRA    20 06
0625   CLC    0C
0626   LDX    FE 00 06
0629   DEC    6A 00
062B   LDAB   C6 09
062D   STAB   F7 00 00
0630   LDX    FE 00 08
0633   LDAB   E6 00
0635   CLR    7F 00 01
0638   LDX    FE 00 08
063B   LDAA   A6 00
063D   ABA    1B
063E   STAA   A7 00
0640   NOP    01
0641   BCS    25 02
0643   BRA    20 04
0645   CLC    0C
0646   INC    7C 00 01   Increment temporary memory
                         for LSB larger than 99.
0649   LDAA   B6 00 00
064C   DECA   4A
064D   BEQ    27 05      Check if conversion is com-
                         plete for LSB's.
064F   STAA   B7 00 00
0652   BRA    20 E4
0654   LDAB   C6 09
0656   STAB   F7 00 00
0659   LDX    FE 00 06   Convert binary MSB'S of data
065C   LDAB   E6 00      to decimal percent
065E   STX    EF 00 06   transmittance equivalent.
0661   LDAA   A6 00
0663   ABA    1B
0664   STAA   A7 00
0666   LDAA   B6 00 00
0669   DECA   4A
066A   BEQ    27 05
```

```
066C    STAA    B7 00 00
066F    BRA     20 ED
0671    LDX     FE 00 06
0674    LDAA    A6 00
0676    LDAB    F6 00 01    Add carry to MSB's from LSB'S.
0679    ABA     1B
067A    STAA    A7 00
067C    BRA     20 0D
067E    BRA     20 02
0680    BRA     20 EF
0682    LDAB    C6 01
0684    ADDB    FB 00 0A
0687    BRA     20 83
0689    BRA     20 A0
068B    LDAA    86 05
068D    STAA    B7 00 0A
0690    LDAA    86 3A
0692    STAA    B7 00 0B
0695    CLR     7F 00 0E
0698    LDX     CE 00 0B
069B    LDAB    E6 00
069D    STX     EF 00 08
06A0    LDAA    A6 00
06A2    STAA    B7 00 0D
06A5    SBA     10
06A6    STAA    A7 00
06A8    LDX     FE 00 06
06AB    LDAA    A6 00
06AD    STAA    B7 00 0C
06B0    LDX     CE 00 0A
06B1    SBCA    A2 00
06B3    BMI     2B 0B
06B5    LDX     FE 00 06
06B8    STAA    A7 00
06BA    INC     7C 00 0E
06BF    CLC     0C
06C0    BRA     20 DC
06C2    CLC     0C
06C3    LDAA    B6 00 0E
06C6    LDX     FE 0F 80
06C9    STAA    A7 00
06CB    INX     08
06CC    STX     FF 0F 80
06CF    LDAA    B6 00 0C
06D2    LDX     FE 00 06
06D5    STAA    A7 00
06D7    LDAA    B6 00 0D
06DA    LDX     FE 00 08
06DD    STAA    A7 00
06DF    LDAA    B6 0F 98    Check for complete conversion.
06E2    DECA    4A
06E3    BEQ     27 05
06E5    STAA    B7 0F 98    Store decimal percent
                            transmittance in memory
```

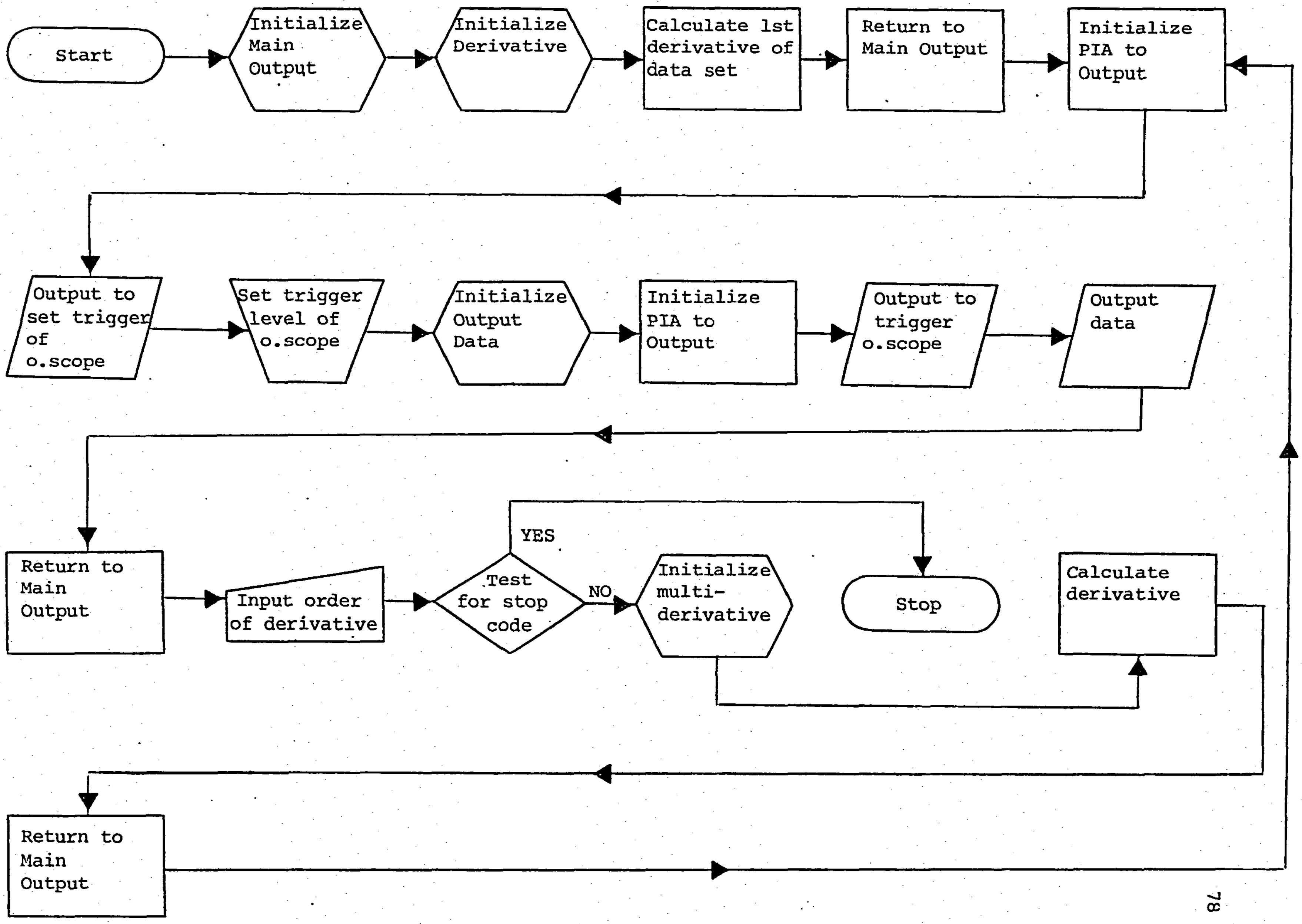
```
                                    positions 0F90 through 0F93.
06E8      BRA      20 9F
06EA      JSR      BD 0E 00         Jump to subroutine Set Memory.
06ED      LDX      FE 00 06         Check for end of data.
06F0      CPX      BC 00 04
06F3      BEQ      27 02
06F5      BRA      20 90
06F7      RTS      39               Return from subroutine to
                                    Tiny Basic program Data
                                    Processor.
```

"Set Memory" (Assembly)

```
0E00   LDX    CE 0F 93   Set bit 1 from tenths position
0E03   LDAA   A6 00      of percent transmittance.
0E05   LDAB   C6 05
0E07   SBA    10
0E08   BMI    2B 39
0E0A   LDX    CE 0F 92   Set bit 2 from ones position
0E0D   LDAA   A6 00      of percent transmittance.
0E0F   LDAB   C6 01
0E11   ABA    1B
0E12   STAA   B7 0F 92
0E15   LDAB   C6 0A
0E17   SBA    10
0E18   BEQ    27 02
0E1A   BRA    20 27
0E1C   CLR    7F 0F 92
0E1F   NOP    01 01
0E21   LDX    CE 0F 91   Set bit 3 from tens position
0E24   LDAA   A6 00      of percent transmittance.
0E26   LDAB   C6 01
0E28   ABA    1B
0E29   STAA   B7 0F 91
0E2C   LDAB   C6 0A
0E2E   SBA    10
0E2F   BEQ    27 02
0E31   BRA    20 10
0E33   CLR    7F 0F 91
0E36   NOP    01 01
0E38   LDX    CE 0F 90   Set bit 4 from hundreds position
0E3B   LDAA   A6 00      of percent transmittance.
0E3D   LDAB   C6 01
0E3F   ABA    1B
0E40   STAA   B7 0F 90
0E43   CLR    7F 00 00
0E46   CLR    7F 00 01
0E49   LDAA   B6 0F 90   Count memory positions for
0E4C   TSTA   4D         bit 1.
0E4D   BEQ    27 14
0E4F   LDAB   C6 64
0E51   ADDB   FB 00 01
0E54   STAB   F7 00 01
0E57   BCS    25 05
0E59   DECA   4A
0E5A   BEQ    27 07
0E5C   BRA    20 F1
0E5E   INC    7C 00 00
0E61   BRA    20 F6
0E63   LDAA   B6 0F 91   Count memory positions for
0E66   TSTA   4D         bit 2.
0E67   BEQ    27 14
0E69   LDAB   C6 0A
```

```
0E6B     ADDB    FB 00 01
0E6E     STAB    F7 00 01
0E71     BCS     25 05
0E73     DECA    4A
0E74     BEQ     27 07
0E76     BRA     20 F1
0E78     INC     7C 00 00
0E7B     BRA     20 F6
0E7D     LDAA    B6 0F 92    Count memory positions for
0E80     ADDA    BB 00 01    bit 3.
0E83     STAA    B7 00 01
0E86     BCS     25 02
0E88     BRA     20 03
0E8A     INC     7C 00 00
0E8D     LDX     CE 0F 92
0E90     LDAB    E6 00
0E92     LDX     CE 0F 91
0E95     LDAA    A6 00
0E97     ASLA    48 48 48
0E9A     ASLA    48
0E9B     ABA     1B
0E9C     STAA    A7 00
0E9E     LDAB    C6 05
0EA0     LDAA    B6 0F 90    Count memory positions for
0EA3     SBA     10          bit 4.
0EA4     BGT     2E 74
0EA6     BLT     2D 08
0EA8     LDAB    C6 56
0EAA     LDAA    B6 0F 91    Logic for determining LSB
0EAD     SBA     10          of memory position of look-up
0EAE     BGE     2C 6A       table.
0EB0     LDAB    C6 03
0EB2     LDAA    B6 0F 90
0EB5     SBA     10
0EB6     BGT     2E 5E
0EB8     BLT     2D 08
0EBA     LDAB    C6 08
0EBC     LDAA    B6 0F 91
0EBF     SBA     10
0EC0     BGE     2C 54
0EC2     LDAB    C6 01
0EC4     LDAA    B6 0F 90
0EC7     SBA     10
0EC8     BGT     2E 48
0ECA     BLT     2D 08
0ECC     LDX     CE 70 00
0ECF     CPX     BC 0F 91
0ED2     BMI     2B 3E
0ED4     LDAB    C6 01
0ED6     LDAA    B6 0F 90
0ED9     SBA     10
0EDA     BEQ     27 32
0EDC     NOP     01 01
0EDE     LDX     CE 94 00
```

```
0EE1   CPX    BC 0F 91
0EE4   BMI    2B 28
0EE6   LDX    CE 52 00
0EE9   CPX    BC 0F 91
0EEC   BMI    2B 1C
0EEE   LDX    CE 90 00
0EF1   CPX    BC 0F 91
0EF4   BMI    2B 10
0EF6   LDX    CE 16 00
0EF9   CPX    BC 0F 91
0EFC   BMI    2B 04
0EFE   LDAA   86 07       Logic for determining MSB of
0F00   BRA    20 1A       memory position of look-up
0F02   LDAA   86 06       table.
0F04   BRA    20 16
0F06   LDAA   86 05
0F08   BRA    20 12
0F0A   LDAA   86 04
0F0C   BRA    20 0E
0F0E   LDAA   86 03
0F10   BRA    20 0A
0F12   LDAA   86 02
0F14   BRA    20 06
0F16   LDAA   86 01
0F18   BRA    20 02
0F1A   LDAA   86 00
0F1C   LDX    FE 00 06    Replace MSB of transmittance
0F1F   STAA   A7 00       with absorbance value from
                          look-up table.
0F21   LDAB   C6 02
0F23   ADDB   FB 00 00
0F26   STAB   F7 00 00
0F29   LDX    FE 00 00
0F2C   LDAA   A6 00
0F2E   LDX    FE 00 08    Replace LSB of transmittance
0F31   STAA   A7 00       with absorbance value from
                          look-up table.
0F33   INT    08 08
0F35   STX    FF 00 08
0F38   LDX    FE 00 06
0F3B   INT    08 08
0F3D   STX    FF 00 06
0F40   RTS    39          Return from subroutine to
                          TAC program.
```



Flowchart B-3. 3400 Output

"Main Output" (Tiny Basic)

```
10 PRINT "MAKE SURE YOU HAVE STORED YOUR ORIGINAL DATA"
20 PRINT "BECAUSE THE CALCULATED DATA WRITES OVER THE ORGINAL
   DATA"
25 PRINT "IF YOU HAVEN'T THEN ENTER A 0 (ZERO) AFTER THE ?
   MARK"
26 PRINT "AND RECORD DATA USING THE MONITOR"
30 PRINT "TO CALCULATE FIRST DERIVATIVE OF SPECTRA"
40 PRINT ENTER A 1 (ONE) FOR INPUT AFTER THE ? MARK"
50 INPUT A
55 IF A=0 GOTO 60
58 GOTO 71
60 C=1824
61 D=USR(7188,0005)
62 E=USR(7188,0004)
63 E=E*256
64 E=E+D
65 PRINT "THE DATA IS BETWEEN",C,"AND",E
66 PRINT "REMEMBER THIS IS IN DECIMAL AND THE MONITOR"
67 PRINT "IS IN HEXADECIMAL SO CONVERT"
68 PRINT "AFTER STORING DATA RESTART THIS PROGRAM BY"
69 PRINT "PRESSING THE B ON THE KEYBOARD AND TYPING RUN"
70 B=USR(5120)
71 B=USR(3696)
80 L=(E-C)/2
90 PRINT "THE OUTPUT PROGRAM TAKES UP 55 USEC"
100 PRINT "YOUR DATA TAKES UP",L,"LOCATIONS IN MEMORY"
110 PRINT "MULTIPLY THESE TWO NUMBERS AND ADJUST THE TIME"
120 PRINT "ON THE O-SCOPE TO COORDINATE"
130 PRINT "NOW ADJUST THE TRIGGER LEVEL ON THE O-SCOPE"
140 PRINT "SO THAT THE L.E.D. LIGHTS UP"
150 F=0
160 F=USR(7192,32769,F)
170 F=USR(7192,32771,F)
180 G=1
190 G=USR(7192,32770,G)
200 PRINT
210 PRINT "NOW TO DISPLAY THE DERIVATIVE IN THE
    NUMBER",A,"ORDER"
220 PRINT "INPUT A 1 (ONE) AFTER THE ? MARK"
230 INPUT Y
235 J=USR(3584)
240 PRINT "IF THE DISPLAY DOESN'T LOOK RIGHT ENTER A 1 AFTER
    THE ? MARK"
250 PRINT "IF THE DISPLAY IS RIGHT ENTER A 2 AFTER THE ?
    MARK"
260 INPUT Z
270 IF Z=1 GOTO 90
280 PRINT "ENTER NEXT ORDER, IF DESIRED, OF DERIVATIVE"
290 PRINT "ENTER A 10 IF YOU WISH TO STOP"
295 INPUT A
```

```
296 L=L-1
300 IF A=10 GOTO 400
310 K=USR(3632)
320 GOTO 90
400 END
```

"Output Data" (Assembly)

```
0E00   LDAA   86 00      Set PIA port B to output.
0E02   STAA   B7 80 03
0E05   NOP    01
0E06   LDAA   86 00      Set PIA port A to output.
0E08   STAA   B7 80 01
0E0B   LDX    CE 07 20   Load first memory position
0E0E   STX    FF 00 00   of data.
0E11   LDAA   86 00      Trigger oscilloscope to
0E13   STAA   B7 80 02   receive data.
0E16   LDX    FE 00 00   Send data to oscilloscope.
0E19   LDAA   A6 00
0E1B   STAA   B7 80 00
0E1E   INX    08
0E1F   STX    FF 00 00
0E22   DEX    09
0E23   CPX    BC 00 04
0E26   BEQ    27 04      Check for end of data.
0E28   NOP    01 01
0E2A   BRA    20 EA
0E2C   RTS    39         Return from subroutine to
                         Tiny Basic program Main
                         Output.
```

"Derivative" (Assembly)

```
0E70    LDX     CE 07 21    Initialize temporary memory
0E73    STX     FF 00 02    for the beginning of the
0E76    LDX     CE 07 20    data.
0E79    STX     FF 00 00
0E7C    LDX     CE 07 23
0E7F    STX     FF 00 08
0E82    LDX     CE 07 22
0E85    STX     FF 00 06
0E88    LDX     FE 00 02    Load LSB of Nth. data point.
0E8B    LDAA    A6 00
0E8D    LDX     FE 00 08    Load LSB of Nth.+1 data point.
0E90    SUBA    A0 00
0E92    STAA    B7 00 0F    Subtract LSB's of data points.
0E95    LDX     FE 00 00    Load MSB of Nth. data point.
0E98    LDAA    A6 00
0E9A    LDX     FE 00 06    Load MSB of Nth.+1 data point.
0E9D    SBCA    A2 00       Subtract MSB's of data points.
0E9F    STAA    B7 00 0E
0EA2    CLC     0C
0EA3    LDAA    B6 00 0E
0EA6    LDX     FE 00 00
0EA9    STAA    A7 00
0EAB    LDAA    B6 00 0F
0EAE    LDX     FE 00 02
0EB1    STAA    A7 00
0EB3    LDX     FE 00 00    Advance temporary memory
0EB6    INX     08 08       positions.
0EB8    STX     FF 00 00
0EBB    LDX     FE 00 02
0EBE    INX     08 08
0EC0    STX     FF 00 02
0EC3    LDX     FE 00 06
0EC6    INX     08 08
0EC8    STX     FF 00 06
0ECB    LDX     FE 00 08
0ECE    INX     08 08
0ED0    STX     FF 00 08
0ED3    DEX     09
0ED4    CPX     BC 00 04    Check for end of data.
0ED7    NOP     01
0ED8    BEQ     27 05
0EDA    NOP     01 01 01
0EDD    BRA     20 A9
0EDF    LDX     CE 07 20    Reset data positions.
0EE2    STX     FF 00 02
0EE5    LDX     CE 07 21
0EE8    STX     FF 00 06
0EEB    LDX     FE 00 06    Crunch data into minimum
0EEE    LDAA    A6 00       storage.
0EF0    LDX     FE 00 02
```

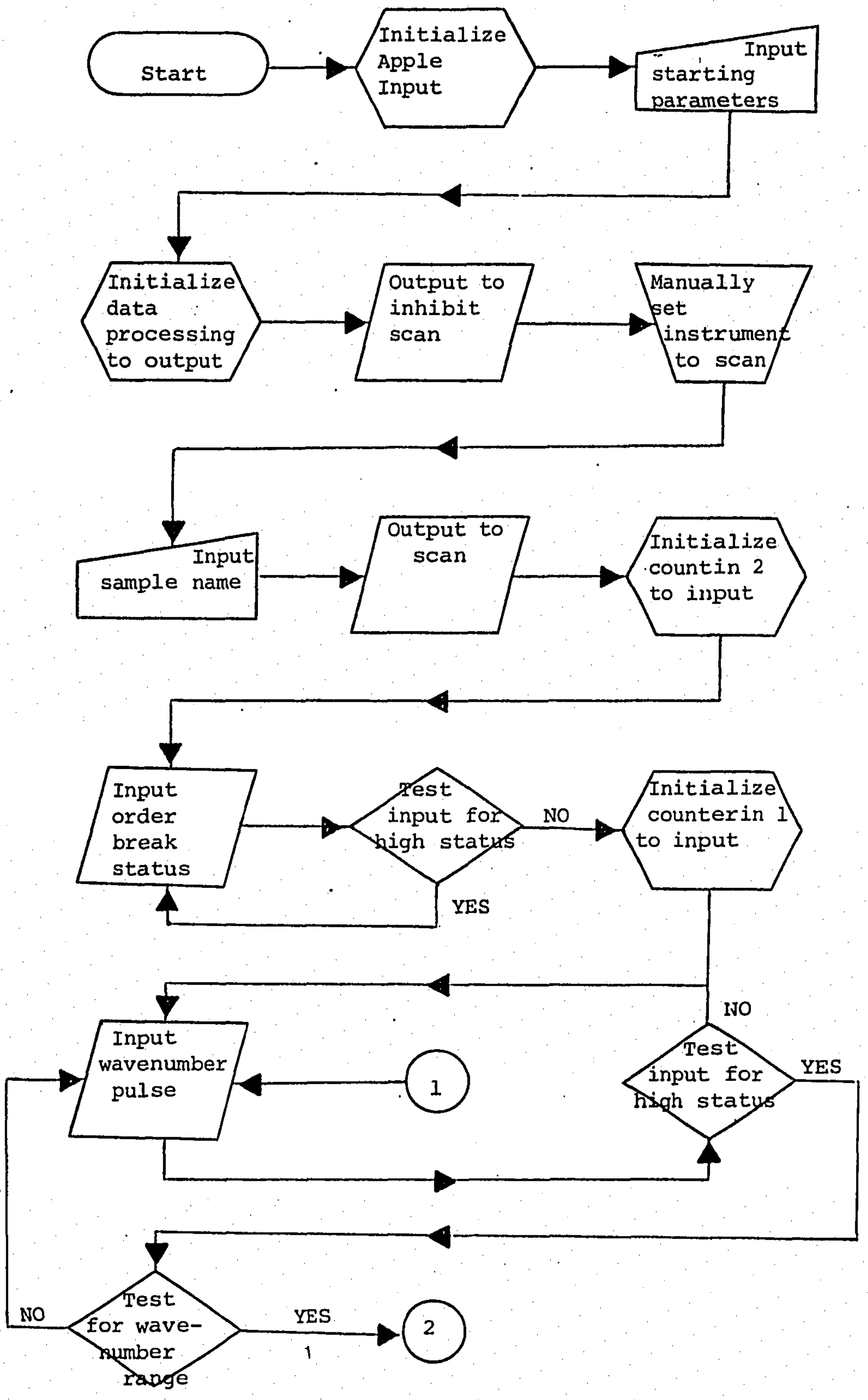
```
0EF3    STAA   A7 00
0EF5    INX    08
0EF6    STX    FF 00 02
0EF8    LDX    FE 00 06
0EFC    INX    08 08
0EFE    STX    FF 00 06
0F01    INX    08
0F02    CPX    BC 00 04
0F05    BEQ    27 02
0F07    BRA    20 E2
0F09    CLC    0C
0F0A    LDX    CE 00 04
0F0D    LDAA   A6 00
0F0F    LSRA   44
0F10    STAA   A7 00
0F12    BCS    25 02
0F14    BRA    20 06
0F16    CLC    0C
0F17    LDAB   C6 90       Reset temporary memory
0F19    BRA    20 03       positions of start and end
0F1B    CLC    0C          data.
0F1C    LDAB   C6 00
0F1E    LDX    CE 00 05
0F21    LDAA   A6 00
0F23    LSRA   44
0F24    ABA    1B
0F25    STAA   A7 00
0F27    LDAA   A6 00
0F29    LDAB   C6 80
0F2B    ABA    1B
0F2C    STAA   A7 00
0F2E    BCS    25 02
0F30    BRA    20 0B
0F32    CLC    0C
0F33    LDAB   C6 01
0F35    LDX    CE 00 04
0F38    LDAA   A6 00
0F3A    ABA    1B
0F3B    STAA   A7 00
0F3D    LDX    CE 00 04
0F40    LDAA   A6 00
0F42    LDAB   C6 03
0F44    ABA    1B
0F45    STAA   A7 00
0F47    LDX    CE 07 20    Load memory position of
0F4A    STX    FF 00 10    beginning of data.
0F4D    LDX    FE 00 10    Convert data to positive
0F50    LDAA   86 80       numbers.
0F52    ADDA   AB 00
0F54    STAA   A7 00
0F56    INX    08
0F57    STX    FF 00 10
0F5A    CPX    BC 00 04
0F5D    BEQ    27 02
```

```
0F5F    BRA    20 EC
0F61    LDX    FE 00 04    Decrement end of data memory
0F64    DEX    09          position by one.
0F65    STX    FF 00 04
0F68    RTS    39          Return from subroutine to
                           Tiny Basic program Main
                           Output.
```

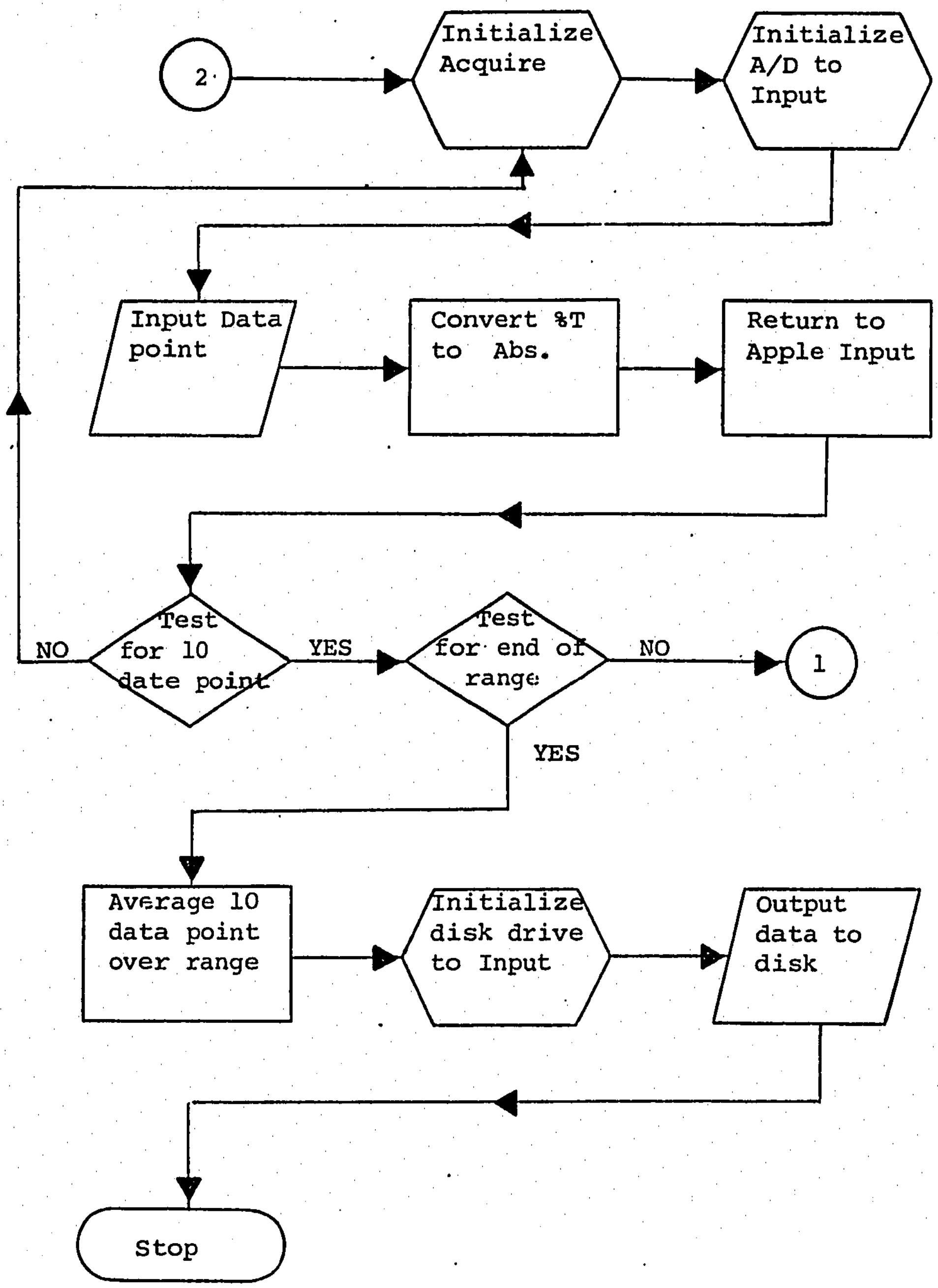
"Multiderivative" (Assembly)

0E30	LDX	CE 07 21	Initialize temporary memory positions for beginning of data.
0E33	STX	FF 00 02	
0E36	LDX	CE 07 20	
0E39	STX	FF 00 06	
0E3C	LDX	FE 00 02	Load Nth. data point.
0E3F	LDAB	E6 00	
0E41	LDX	FE 00 06	Load Nth.+1 data point.
0E44	LDAA	A6 00	
0E46	SBA	10	Subtract data points.
0E47	LDAB	C6 80	Convert difference to positive number.
0E49	ABA	1B	
0E4A	LDAA	A7 00	
0E4C	INX	08	
0E4D	STX	FF 00 06	Store derivative data point in current memory position.
0E50	LDX	FE 00 02	
0E53	INX	08	
0E54	STX	FF 00 02	
0E57	CPX	BC 00 04	Check for end of data.
0E5A	BEQ	27 02	
0E5C	BRA	20 DE	
0E5E	CLC	0C	
0E5F	LDX	FE 00 04	Decrement end of data memory position by one.
0E62	DEX	09	
0E63	STX	FF 00 04	
0E66	RTS	39	Return from subroutine to Tiny Basic program Main Output.

Appendix C



Flowchart C-1. Apple/Isaac Input



Flowchart C-1 (Cont)

1

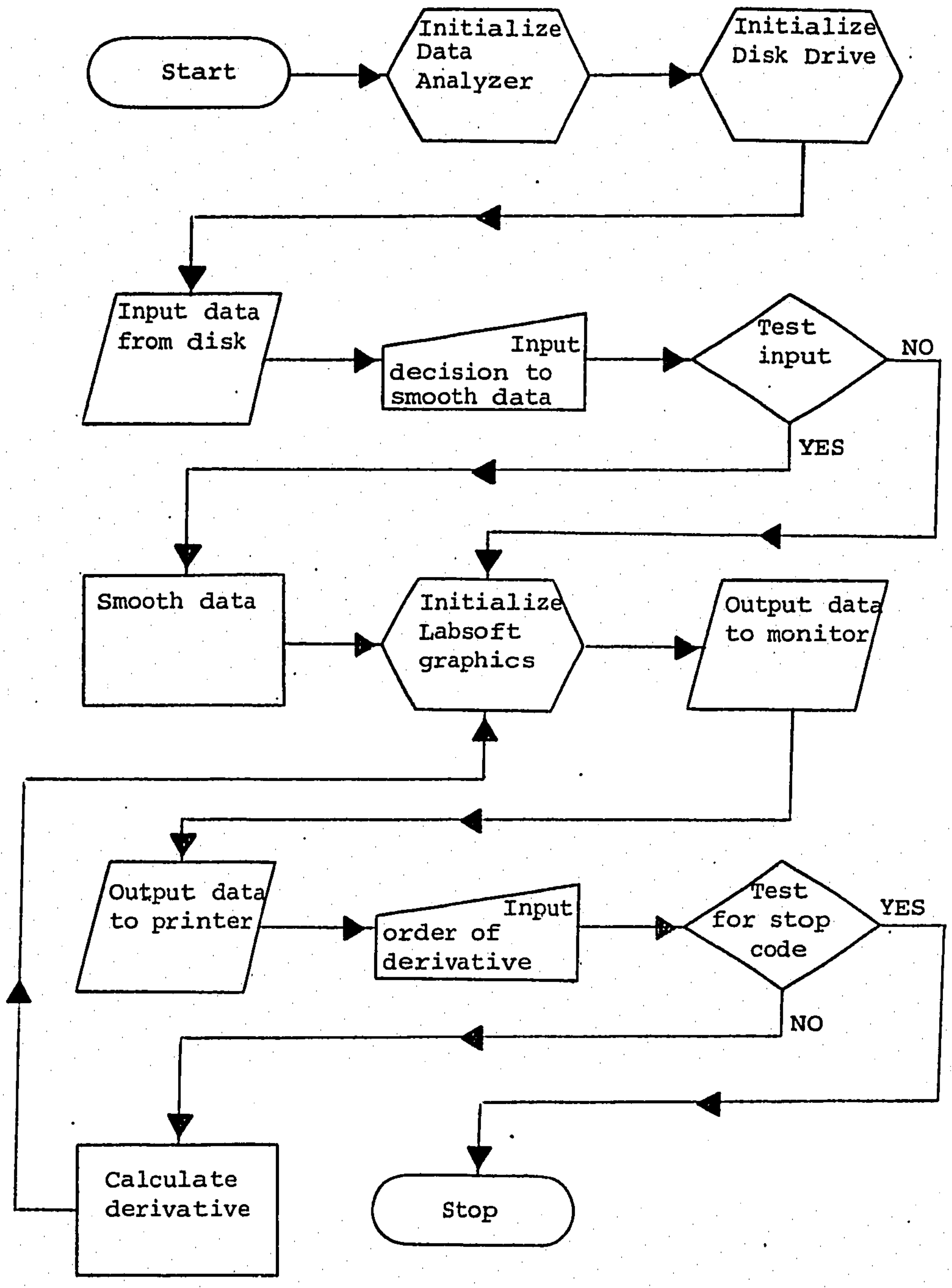
"APPLE INPUT" (BASIC)

```
3 DIM B(20)
4 DIM A(2400)
5 M=1
6 Q=0
7 P=0
8 B(0)=0
10 PRINT "ENTER STARTING WAVENUMBER POSITION OF INFRARED
   MONOCHROMATOR"
20 PRINT
30 PRINT "FOR THE ACCULAB 8 I.R. THIS WILL BE EITHER 4000
   CM^-1 OR 2000 CM^-1"
40 PRINT
50 INPUT S
51 IF S=4000 THEN GOTO 70
52 IF S=2000 THEN GOTO 70
53 GOTO 30
60 PRINT
70 PRINT
80 PRINT "ENTER WAVENUMBER RANGE TO BE STUDIED"
90 PRINT
100 PRINT "WAVENUMBER MAXIMUM (CM^-1)"
110 PRINT
120 INPUT L
130 PRINT
140 PRINT "WAVENUMBER MINIMUM (CM^-1)
150 PRINT
160 INPUT D
163 PRINT
164 PRINT
165 IF L>D THEN GOTO 185
166 PRINT "RE-ENTER WAVENUMBERS CORRECTLY"
167 PRINT
168 GOTO 80
180 PRINT
185 IF S>L GOTO 190
186 GOTO 166
190 PRINT "PRESS ONE OF THE SCAN SPEED BUTTONS ON THE
    ACCULAB 8"
200 PRINT
210 PRINT "ENTER THE SCAN SPEED"
215 PRINT "SLOW=1 ; MEDIUM=2 ; FAST=3"
220 PRINT
230 INPUT G
231 IF G=1 THEN C=700
233 IF G>3 GOTO 200
234 IF G<1 GOTO 200
235 IF G=1 THEN C=700
237 & AOUT,(DV)=4095,(C#)=3
238 & AOUT,(DV)=0,(C#)=1
239 & AOUT,(DV)=0,(C#)=2
```

```
240 PRINT
241 PRINT "ENTER SAMPLE NAME"
242 PRINT
243 INPUT N$
245 PRINT
250 & AOUT,(DV)=0,(C#)=1
260 & AOUT,(DV)=0,(C#)=2
261 PRINT
262 PRINT "PRESS THE SCAN BUTTON"
263 PRINT
264 PRINT "ARE YOU READY (Y OR N)"
265 INPUT E$
266 IF E$="Y" THEN GOTO 270
267 GOTO 261
270 F=S-L
275 Z=L-D
282 IF S=4000 THEN Z=Z/2
283 IF S=4000 THEN F=F/2
286 & AOUT,(DV)=4095,(C#)=1
287 & AOUT,(DV)=4095,(C#)=1
305 & CLRCOUNTER,(C#)=2
306 & COUNTERIN,(TV)=E
307 IF E=1 GOTO 305
309 & CLRCOUNTER,(C#)=1
310 & COUNTERIN,(TV)=W
312 IF W=0 THEN O=0
315 IF W=0 GOTO 310
320 IF O=1 GOTO 310
325 Q=Q+W
330 IF Q)=F GOTO 350
335 O=1
340 GOTO 310
350 GOSUB 2000
360 P=P+1
370 IF P)=Z GOTO 1000
380 GOTO 310
1000 & AOUT,(DV)=0,(C#)=1
1010 PRINT
1011 PRINT "PLACE DATA DISK IN DRIVE #1"
1012 PRINT "ARE YOU READY"
1013 INPUT M$
1014 PRINT
1015 PRINT "DATA BEING STORED UNDER"
1020 PRINT "SAMPLE NAME ",N$
1025 LET D$=CHR$(4)
1030 PRINT D$;"OPEN";N$
1035 PRINT D$;"WRITE";N$
1040 FOR J=1 TO Z
1045 PRINT A(J)
1050 NEXT J
1055 PRINT D$;"CLOSE";N$
1060 END
```

"ACQUIRE" (BASIC)

```
2000 FOR I=1 TO 10
2020 & AIN,(TV)=X,(C#)=0
2030 B(I)=LOG(X)*-.434294483
2040 A(M)=A(M)+B(I)
2045 X=0
2050 NEXT I
2060 M=M+1
2065 O=1
2070 RETURN
```



Flowchart C-2. Data Analyzer

"DATA ANALYZER" (BASIC)

```
10 L=0
20 K=1750
30 P=0
500 DIM D(250),B(250),A(250)
505 INPUT N$
510 FOR I=1 TO 10
520 PRINT D(I)
530 NEXT I
535 LET D$=CHR$(4)
540 PRINT D$;"OPEN";N$
550 PRINT D$;"READ";N$
560 FOR J=1 TO 200
570 INPUT D(J)
580 NEXT J
585 PRINT D$;"CLOSE";N$
586 FOR I=1 TO 200
587 IF L<ABS(D(I)) THEN L=ABS(D(I))
588 NEXT I
595 FOR I=1 TO 200
598 B(I)=D(I)+L
599 NEXT I
600 GOTO 620
610 PR#1
611 PRINT "SMOOTHED DATA"
612 PRINT N$
613 PRINT "WAVENUMBER                    ABS."
614 FOR I=60 TO 150
616 PRINT K-I,B(I)
618 NEXT I
619 PR#0
620 & HIRES1
630 & SCROLLSET
640 FOR I=60 TO 190
645 FOR J=1 TO 2
650 & NXTPLT=B(I)*120
655 NEXT J
660 NEXT I
665 PRINT "DATA SMOOTH AND PLOT?"
670 INPUT M$
675 IF M$="N" GOTO 700
680 IF M$="Y" GOTO 1000
690 GOTO 665
700 PRINT "DERIVATIVE ?"
710 INPUT M$
720 IF M$="Y" GOTO 2000
730 IF M$="N" GOTO 6000
740 GOTO 700
1000 FOR I=1 TO 197
1010 B(I)=B(I)+B(I+1)+B(I+2)
1020 B(I)=B(I)/3
```

```
1030 NEXT I
1040 GOTO 610
2000 P=0
2005 INPUT Z
2006 TEXT
2010 PRINT "ENTER NUMBER OF DERIVATIVE"
2020 INPUT R
2022 FOR I=1 TO 200
2023 A(I)=B(I)
2024 NEXT I
2025 IF R=0 GOTO 6000
2030 FOR I=1 TO 10
2040 FOR J=1 TO 198
2050 A(J)=A(J+1)-A(J)
2060 NEXT J
2070 IF I=R GOTO 2082
2080 NEXT I
2082 PR#1
2084 PRINT N$,"DERIVATIVE =",R
2086 PRINT "WAVENUMBER              DER. VALUE"
2088 FOR I=60 TO 150
2090 PRINT K-I,A(I)
2092 NEXT I
2094 PR#0
3000 FOR I=1 TO 190
3006 E=P
3010 IF P<ABS(A(I)) THEN P=ABS(A(I))
3060 NEXT I
3070 & HIRES1
3080 & SCROLLSET
3090 FOR I=70 TO 190
4000 FOR I=1 TO 2
4010 & NXTPLT=(A(I)+P)*(127/P)*.5
4020 NEXT J
4030 NEXT I
4035 & RETRCE
4040 PRINT
4050 GOTO 2000
6000 END
```

REFERENCES

1. Smith, A.L. Applied infrared spectroscopy. New York: John Wiley and Sons, Inc. 1979.

2. Olsen, E.D. Modern optical methods of analysis. New York: McGraw-Hill Book Co., 1975.

3. Schmitt, A. Application data bulletin: Derivative spectroscopy. Perkin-Elmer Instrument Co., 1979.

4. Botten, D., Honkawa, T. & Tohyama, S. Application data Bulletin: Second derivative spectroscopy. Perkin-Elmer Instrument Co., 1979.

5. O'Haver, T.C. & Green, G.L. Numerical error analysis of derivative spectrometry for the quantitative analysis of mixtures. Anal. Chem., 1976, 48, 312.

6. O'Haver, T.C. Derivative and wavelength modulation spectrometry. Anal. Chem., 1979, 51, 91A.

7. Snellman, W. Flame emission spectrometry with repetitive optical scanning in the derivative mode. Anal. Chem. 1970, 42, 394.

8. Fowler, W.K. Double modulation atomic fluorescence flame spectrometry. Anal. Chem., 1974, 46, 601.

9. Kolb, D.A. & Shearin, K.K. Fingerprinting petroleum oils with low temperature derivative fluorometry. Pittsburgh Conference on Analytical Chemistry and Applied Spectroscopy, Cleveland, Ohio, 1977.

10. O'Haver, T.C. & Green, G.L. Derivative luminescences spectrometry. Anal. Chem., 1974, 46, 2191.

11. Overland, J., Gilby, A.C., Russell, J.W., Brown, C.W., Beutel, J., Bjork, C.W. & Paulet, H.G. A Littrow-McCubbin high resolution infrared spectrometer. Appl. Opt., 1967, 6, 457.

12. Malmstadt, H., Enke, C. & Crouch, S. Electronics and instrumentation for scientists. New York: The Benjamin-Cummings Publishing Co., Inc., 1981.

13. Osborne, A. An introduction to microcomputers, vol. 2. New York: Adam Osborne and Assoc., Inc., 1978.

14. Titus, C.A., Baldwin, A., Hubin, W.N. & Seaulon, L. 16-bit microprocessors. New York: Howard W. Sams and Co., 1981.

15. Zaks, R. & Lesea, A. Microprocessor interfacing techniques (3rd ed.). New York: Sybex, Inc., 1979.

16. Diefender, A.J. Principles of electronic instrumentation (2nd ed.). New York: W.B. Saunders Co., 1979.

17. Hnatek, E.R. A user's handbook for D/A and A/D converters. New York: John Wiley and Sons, Inc., 1967.

18. Astle, M.J. & Shelton, J.R. Organic chemistry. New York: Harper Brothers publishing, 1953.

19. Hiscox, D.J. Determination of iodine numbers. Anal. Chem., 1948, 20, 679-680.

20. Official methods of analysis of the association of official agriculture chemists (10th ed.). 1965.

21. Murthy, R. & Bkat, G.S. Iodine number determinations of milk fats and vegatable fats by refractometry. Amer. Oil Chem. Soc., 1976, 53, 577-581.

22. Bernard, J.L. & Sims, L.G. Infrared determination of total unsaturation. Ind. Res. and Dev., 1980, 22, 81-83.

23. Beckman Instruments, Inc. Acculab T.M. infrared spectrophotometers operating instructions., 1979.

24. Beckman Instruments, Inc. Acculab 7, 8, 9, and 10 service manual. 1978.

25. Heath Co. Microprocessor trainer model ET-3400 manual. 1977.

26. Heath Co. Memory I/O accessory model ETA-3400 manual. 1979.

27. Heath Co. 10 bit D/A converter model EC-800-GC manual. #595-1371.

28. Gould Inc., Instrumental Division. Operating and service manual for digital storage oscilloscope OS4000 and output unit OS4002. 1978.

29. Houston Instrument. Series 2000 Omnigraphic X-Y recorder instruction manual, Princeton applied research model RE0074.

30. Apple Computer, Inc. Apple II reference manual. 1981.

31. Curtis, R.O., Mottola, R.M. & Reilly, J. Isaac 91A hardware reference manual. Cyborg, Corp., 1982.

32. Curtus, R.O., Mottola, R.M. & Reilly, J. Isaac Labsoft reference manual. 1982.

33. Harrow, B., Borek, E., Mazur, A., Stone, G.C.H. & Wasreich, H. Laboratory manual of biochemistry (5th ed.). New York: W.B. Saunders Co., 1967.

34. Bhattacharyya, G.K. & Johnson, R.A. Statistical concepts and methods. New York: John Wiley and Sons, 1977.

35. Moore, A.W. Microprocessor application manual. New York: McGraw-Hill Book Company, 1975.

36. Osborne, A. 6800 programming for logic design. New York: Adam Osborne and Assoc., 1977.

37. Systron-Donnor Co. Instruction manual model 1234 analog-to-digital converter. 1965.

38. Malmstadt, H.V. & Enke, C.G. Electronics for scientists. New York: W.A. Benjamin Inc., 1963.