A SIXTEEN BIT

DATA ACQUISITION SYSTEM

FOR ELECTROCBXMICAL INSTRUMENTATION

by

Bruce R. Hahn

Submitted in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in the

Chemistry

Program

Aug 19, 1985

Advisor Date

August 29, 1985

Dean of the Graduate School Date

YOUNGSTOWN STATE UNIVERSITY

August, 1985

ABSTRACT

A SIXTEEN BIT DATA ACQUISITION SYSTEM FOR ELECTROCHEMICAL INSTRUMENTATION

Bruce R. Hahn

Master of Science

Youngstown State University, 1985

The preparation of a sixteen bit data acquisition system is described. Included is a brief introduction to computers, computer languages and their uses in analytical chemistry, especially electrochemistry. The design and construction of circuit boards is included. A sample program for data acquisition is also p esented.

ACKNOWLEDGEMENTS

I wish to express my gratitude to Dr. Daryl Mincey for his guidance and encouragement on this project. I also give my thanks to Patty for her patience and support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

In the past few years the growth of mini-or-micro-computer use in the laboratory has progressed rapidly. The development and improvement of the computer industry has also taken place very rapidly and the implementation of laboratory computer systems has been consistently growing as well.

The use of computers in analytical laboratories has evolved from data processing to the actual collection and storage of data from a wide spectrum of instruments. Computers have enabled the analytical chemist to perform many tasks which would have been impossible a few years ago. The widest use of computerization in the laboratory has been fourier transform spectroscopy. Computers are becoming important in electrochemistry by enabling the researcher to both monitor and control experiments such as linear sweep, cyclic, normal pulse, and differential pulse voltammetry. Computers have also freed the analytical chemist from many different analyses which were so time-consuming in the past.

The proliferation of computers in the lab, while making many tasks simpler, has also added a new dimension to the training and functioning of the analytical chemist. Just a look inside the door of a modern analytical lab should convince anyone that the need for computer literacy and an understanding of computer hardware is a must for the complete knowledge of modern analysis techniques.

The availability of computer systems for the laboratory has improved in recent years. Many fine systems are now available for routine analysis. The instruments industry is constantly making the systems easier to use. This improvement throughout the industry is taking place so rapidly that it is difficult for the analytical chemist to make intelligent selections as to the instruments for his or her own needs. Although the industry is rapidly changing and growing, cost and availability of equipment to meet the exact needs of the researcher still remain as prohibitive factors for many laboratories.

In the research lab today analytical chemists must be able to choose a system and be able to adapt it to their needs. To be able to adapt a system to the needs of a research lab requires the understanding of both the experimentation techniques and an understanding of the computer itself.

To completely understand a computer system, a working knowledge of basic electronics, computer hardware and computer logic is necessary. A working knowledge of these will enable the researcher to understand the qualities, both good and bad, of the data collected. Chemists with the ability to change or adapt computer systems have the advantage of being able to improve the quality of their work.

Electrical information

The information obt ined in electrochemical analysis is obtained as a group of particular electrical signals. These signals consist of three different data domains: analog, time, and digital.[1] The raw electrical signals must then be converted into information by analysis of their electrical properties: charge, current, voltage, and power.

Data from analog domains are obtained as the magnitude of charge, current, voltage, and power in the electrical signal. Input and output transducers are usually used to control and measure these analog signals. Input transducers convert nonelectrical energy into electrical energy. An example of an input transducer is a thermocouple device. Output transducers convert electrical energy into another form of energy, usually physical, such as in a dial on a pH or voltmeter.

Analog signals are very useful in electrochemical analysis but they are susceptible to noise and measurements made from analog signals consist of both the signal and the noise. Data which are in digital or time domains are more accurate as they are less affected by noise[1].

Data in the time domain consist of logic-level signals. Electrical signals consist of a high or low logic, the data are made up of the time between successive high and low signals, a period, or the time between low-high and high-low signals, a pulse width, or the number of high-low transitions that occur in a unit of time, frequency.

In the digital domain, data exist also as high-low logic, but data are represented as a combination of high and low signals. which are encoded to represent a certain digit or symbol. Digital domain data are the data which is used in modern electrical logic circuits.

Digital data do not need any further changes or transformations in order to render numerical information, the signal only needs to be decoded. Data from the analog and time domains must be transformed into digital data in order to make use of modern electronic logic circuitry.

Digital signals can be encoded in one channel or in a series of parallel signals. 'Data represente . in a single channel (serial) are encoded as a number of pulses (count waveform). This type of information is less efficient than the data represented as a parallel series of signals.

The most efficient form of parallel digital signals is known as the binary-coded serial signal[1]. In binary-coded data each pulse in a series represents a binary digit (bit). The resolution achieved with this type of data is very high. In a binary coded signal a series of 16 pulse times would have a resolution of one part in 2^{16} or one part in 6.5536×10^4.

BINARY NUMBERS

In a binary number system, a combination of ones and zeroes represent integer numbers. In our decimal system of numbering, each digit is given a value equal to a power of ten. In the binary number system each digit is given a value equal to a power of two. The farthest number to the right in a decimal system is called the least significant digit and the farthest digit to the right in a binary system is called the least significant bit (LSB). An example of the binary number system is listed in table 1.

Negative numbers are usually represented by two's complement notation in which all 1's are changed to zero and all 0's to ones and then the result is incrernented by one. For example: -4 = 1100.

TABLE 1

BINARY NUMBER SYSTEM

Binary				Decimal
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

LOGIC CIRCUITS

In 1854 George Boole wrote "An investigation of the Laws of Thought on Which Are Pounded the Mathmatical Theories of Logic and Probabilities"[2]. In this work he introduced the idea of two-state logic, which is the basis of modern electronic logic circuitry. The logic behind this circuitry is known as BOOLEAN ALGEBRA and an understanding of Boolean Algebra is essiential in understanding modern instruments in the analytical laboratory.

Logic circuitry is composed. of "gates", known as logic gates, which are the basis of modern digital circuitry.

All two level logic functions are combinations of the three basic logic operations: AND, OR, and INVERT. Combinations of these operations make up most digital circuitry. In these logic operations HI AND LO voltages are used to represent true and false respectively. Boolean algebra can express and manipulate the combinations of these logic functions.

Some of the main logic gates are listed in table 2[3].

Table 2

LOGIC GATES

LOGIC GATE	TRUTH TABLE
AND PUNCTIOJT	C=A*B

C	A	B
0	0	0
0	0	1
0	1	0
1	1	1

LOGIC GATE	TRUTH TABLE
OR FUNCTION	C=A+B

C	A	B
0	0	0
1	0	1
1	1	0
1	1	1

LOGIC GATE	TRUTH TABLE
NAND FUNCTION	$C=\overline{A*B}$

C	A	B
1	0	0
1	0	1
1	1	0
0	1	1

LOGIC GATE	TRUTH TABLE
NOR FUNCTION	$C=\overline{A+B}$

C	A	B
1	0	0
0	0	1
0	1	0
0	1	1

MICROCOMPUTERS

In recent years the technology behind mini–and microcomputers has grown so rapidly that the advantages of microcomputers compared to mainframes has eclipsed the disadvantages. "In fact, in technical terms there really are no more practical boundaries between microcomputers, minicomputers, and mainframes."[4] In the past, mainframes had a large advantage in the area of storage and programming capabilities which are important for use in obtaining and processing electrochemical information. Today inexpensive microcomputers are availible with sufficient memory and programming capabilities for use in electrochemical work, which gives microcomputers an advantage over mainframes in terms of size and cost.

Electrochemical data, which are analog data, can be transformed into digital data and processed by a microcomputer. The data are transformed by an analog–to–digital converter and transferred through input–output units (I/O) into the computer. The data then go to a Central Processing Unit (CPU) and are transformed through a program into files, graphical analyses, or other useful forms.[5]

The transfer of data as described above is carried out in response to a set of instructions which are logically interpreted and carried out by the CPU. The CPU is the main part of a. microcomputer. The integrated circuit (I/C) which makes up the CPU is known as the microprocessor.

The transfer of data takes place along a data bus, which determines the length of the microcomputers word. Data buses in the past were 4 bits or 8 bits wide. Recently, advanced circuit technology has provided microprocessors with 16 bit data buses, and a 32 bit data bus is coming in the near future. The increase in the size of the data bus means a large increase in the resolution of the computer. An 8 bit word length has a resolution of 1 part in 2^8, while a 16 bit word length has a resolution of 1 part in 2^{16} and a 32 bit word length has a resolution of 1 part in 2^{32}!

The CPU sends data to the appropiate memory location or device by using an address code. Each memory location in a computer and every interfaced device must have its own specific address. Addressing takes place along the address bus. The size of the address bus determines the number of address codes availble to the microprocessor. A 16-bit address bus would have 2^{16} possible address codes.

The CPU sends data to the appropriate memory location or device by using an address code. Each memory location in a computer, and every interfaced device, must have its own specific address. Addressing takes place along the address bus. The size of the address bus determines the number of address codes availible to the microprocessor. A 16--bit address bus would have 2^{16} or 65536 possible address codes.

The speed. of a microcomputer is determined by its clock. The CPU sequences its operations by the clock's oscillations. The faster the clock, the faster the CPU.

The CPU is made up of registers. A register is a set of latches which can hold one word. For example, an 8-bit register is made up of 8 latches. The primary registers in a CPU are located in the ALU (arithmetic or logic unit). There are also general-purpose and indexing registers within a CPIJ. These registers can hold an address or an instruction.

A microcomputer's memory size also is a factor in determining its utility in the laboratory. Memory devices are made up of storage cells which can hold one bit of information. This memory is divided into two types: Head Only Memory (ROM) and Random Access Memory (RAM, or read-write memory). RAM memory can be changed by the CPU and ROM memory is permanent and cannot be changed by the CPU. One type of ROM is erasable programmable read only memory (EPROM) which can be erased and changed.

RAM memory is made up of Flip-Flop type circuitry. The simplest Flip-Plops have two inputs and one output. One input is a clock input and the other a data input. The data input nay change states (HI-LO), however the output will remember the last data. input until a transition in the clock input occurs. This circuitry allows for memory and also a change of memory.

ROM memory is made up of circuitry that has inputs and outputs with no clock input and permanent memory. ROM's come in various sizes. A common ROM has 10 address inputs and 8 data outputs. This gives the ROM 2^{10} or 1024 memory locations 8 bits in length. Eight bits is known as one byte and 1024 is known as 1 K. This type of ROM would be a 1 KByte memory chip.

In order to utilize a microprocessor in a laboratory, it must have Input/Output (I/O) capabilities. Most microcomputers have an input register and an output register which are connected to the data bus. Since the CPU does have complete control over the utilization of the I/O registers, some sort of compatability is required with outside devices: disk memory, printers, data converters, etc. This compatability is achieved through the use of flags and interrupts. These flags and interrupts set certain status bits as high or low. These status bits tell the CPU through a program what to do.

Computer Languages

A computer's language is merely an extension of the logic circuitry within the computer. Since the circuit logic is a combination of HI and LO, a microprocessor's vocabulary consists entirely of HI's and LO's or 1's and 0's. The language of 1's and 0's is known as machine language. For the laboratory chemist, as with most users, machine language is too difficult to use as a programming language. Over the years programmers as well as scientists have developed human type languages for operating a computer.

Assembly language is usually the first language developed for a computer system. Assembly language is a set of codes and mnemonics, which directly represent machine language instructions.[6] Assembly language ("Assembler") is translated by an assembler into the machine code. Assembler gives the user complete control over the computer, is often time consuming and confusing, but for electrochemical interfacing a knowledge of this language is needed.

Programmers have developed many higher level languages, languages which operate differently from the microprocessor, which are very useful in the laboratory. These languages require an operating system in order to function. Each of these languages has its advantages and disadvantages in regard to research work.

For use in the laboratory, a language requires many different characteristics. Sone of these characteristics are listed below.[7]

TABLE 3

LANGUAGE CHARACTERISTICS

1. Mathematic and logic funcions.
2. Variables and constants.
3. Ability to define new operations and commands.
4. A set of conditionals and repetitives which allow branching and loops.
5. Reentrancy or sharing of routines among users.
6. Pointer or labeled addressing.
7. User interactive.
8. Large storage and array capabilities.
9. Immediate use of the primary functions, such as +,-.
10. The language should also allow multitasking (the ability to carry out more than one operation at a time. i.e, control data and acquisition of an electrochemical experiment by the computer and various instruments.)

The two most common languages in use in the laboratory are BASIC and FORTRAN. These languages are good for number manipulation (crunching), but both are lacking in speed in the solving of complex problems, and in interfaced-instrument control.

A relatively new language, FORTH, is being used. by many instrument manufacturers.[7] FORTH meets the requirements of a good laboratory language and it is relatively simple and extremely fast.

FORTH was started as a language to control large telescopes[7]. Being written in mind as an instrument controlling language gives FORTH an advantage in the electrochemical lab.

ELECTROCHEMICAL DATA ACQUISITION

All electrochemical data exist in the analog domain. In order to be able to process these data they must be either recorded manually and entered into a computer, or they must be converted to digital data with the use of an analog-to-digita converter. An analog-to-digital converter will take an analog voltage input and convert it into a parallel digital data output, which can then be transferred to the computer for storage and processing.

Electrochemical data are in the analog form and so the instrument that controls the experiment, such. as a potentiostat, operates in the analog domain. In order to have control over the experiment, a data acquisition system must have digital-to-analog converters. A digital-to-analog converter accepts parallel digital data and converts them into an analog voltage output.

Many different types of analog-to-digital converters (ADC's) are availible on the market. Most of these are either feedback ADC's, or integrating type ADC's. A brief discussion of both types follows.

Feedback ADC's

Feedback ADC's consist of a comparator, a DAC, and digital logic circuitry.[8] When an analog signal is received, the digital logic signal will increment a digital input to the DAC until the comparator senses that the DAC output and the analog input are equal. Once the two signals are equal, the digital number in the DAC will be the digital output of the ADC.

Three types of feedback ADC's are commonly used: the ramp type, the tracking type, and the successive approximation type. The ramp type simply starts at zero volts ramps and compares it to the input until the signals are equal. The tracking type ADC will start like the ramp type ADC but it also has the ability to decrement the digital signal so a more precise equality between the analog input and the digital output exists. The third type of feedback ADC is the successive approximation type. This ADC is much faster than the ramp or tracking types. The ramp and tracking type ADC's compare by incrementing the least significant bit (LSB) until an equality exists. The successive approximation type compares the most significant bit first and increments or decrements it accordingly. The conversion time (T_c) for this method is

$$T_c=(n+1)T_{cp}=(n+1)/F_{cp}$$

where n is the number of bits and F_{cp} is the clock frequency. The conversion time for a ramp type ADC is

$$T_c=(V_{in}/I_{ref}R_{in})x2^nT_{cp}$$

T_{cp} is the clock period ($T_{cp}=1/F_{cp}$). The successive approximation conversion is easily seen to be much faster.

Integrating ADC's

ADC'S have been developed that give a digital ouput depending on the average or integral of the analog input over a given time period. These types of ADC's will give good results even when high frequency noise interferes with the analog input. Two types of integrating ADC's exist: dual-slope and quantized-feedback.

In the dual-slope type the analog input is integrated for a period of time. If the signal is constant then the integrator will output a linear ramp. Then a reverse reference voltage is applied to the ramp and the time it takes to go back to zero will be proportional to the analog input and the digital output is taken from that time or a corresponding number of clock pulses.

The quantized method is similar to the dual-slope, except that the conversion processes occur simultaneously.

The importance of knowing these different types of converters lies in whether the research needs a fast or an accurate type of data conversion, for most electrochemical data, an accurate type would be the preference. Integrating converters are the most accurate, but also the slowest. Tracking type converters lack the accuracy of the integrating converters and the speed of the succesive approximation converters, therefore they are rarely used. Sucessive approximation converters are the most freyuently used because they are a compromise of both speed and accuracy.

Digital-to-Analog converters exist in many different forms, however the type used is not as critical as with ADC's.

Peripheral Interface Adapter

Once the DAC's and ADC's have been selected, communication with the CPU must be established. This is done with the use of a Peripheral Interface Adapter (PIA). Programmable and nonprograramable PIA's exist. The choice for the researcher should be the programmable type. The programmable type provides much more flexibility and it can be directly add-ressed by the CPU.

PIA's serve as the interface between the various I/O devices (printers, disks, data converters, etc.. .) and the CYU. Since each I/O device has its own address the PIAs must be programmed to respond to that particular address.

A programmable PIA can generate interrupts (usually by setting a flag HI or LO) which tell the GPU that new data are ready. The programmable PIA can also be programmed to provide "handshaking" with the CPU. Handshaking occurs when the interrupt flag is set and the CPU comes back and sets another flag (usually the data request line) which tells the PIA whether or not it is ready for the new data.

PIA's contain their own registers for storing, addressing, and programming. The researcher can program the PIA's for his needs. Common PIA's have two 8 bit parallel ports. Each port also has two control lines which can be set HI or LO to provide a variety of functions. The control lines can be set four different ways to control the 8 bit ports. The ports can be set as input or ouput lines, to generate interrupts, or to serve as handshaking lines to the CPU.[3]

PIA's are made to interface with only one microprocessor type, so each microprocessor manufacturer usual-ly makes it's own PIA chip.

JUSTIFICATION

Electrochemical analysis with the aid of microprocessors and minicomputers has become commonplace in the laboratory. Most of the work done to date has been with 8-bit microcomputers or with minicomputers, using 12-bit analog-to-digital and digital-to-analog converters (ADC's, DAC's). With the electronics available today, it is possible to greatly improve both the resolution (the magnitude of the smallest detectable variation in a quantity) and dynamic range (the range or amount of change in a quantity, i.e., voltage) of these experiments with the use of a 16-bit microcomputer and 16-bit ADC's and DAC's.

The use of these converters should provide a better view and understanding of electrochemical reactions. With the improvement of resolution and dynamic range comes better monitoring and controlling of electrochemical experimentation.

The speed. of the computer systems is another factor to consider when improving upon existing systems. Computer speed for input and output (I/O) routines depends upon the Central Processing Unit's (CPU) clock and the computer language used to run the routines. Combining both a fast CPU clock with a low level language is the best way to optimize the speed of data acquisition. While some commercially availble systems use this combination, the research chemist should he aware of which do and which do not and the limitations imposed.

A computer's memory is also another factor the researcher must take into consideration when designing or adapting a system to his use. While data can be stored on disk or tape systems, data taken during an experiment are much more efficiently stored in a computer's on board random access memory (RAM). A system with a large RAM will allow the researcher to store the data quickly and efficiently during the experiment and data transfer can be made to disk or tape storage after the run of the experiment.

Computer systems must also be flexible. Many commercially designed computer systems for the laboratory are designed. with limited user applications. Many times a commercially supplied computer is only capable of running one instrument or performing one function in the lab. For the researcher this can lead to problems, especially when the total needs and demands of a computer system cannot be determined in advance.

In summary, a good computer system for the researcher should have good interfacing capabilities, high resolution, and a large dynamic range. It should be fast, with a large memory size, and designed with flexibility in mind.

CHAPTER 2

STATEMENT OF PROBLEM

Purpose

The purpose of this work is to design and build 16-bit analog-to-digital and digital-to-analog converter systems and interface them with a state-of-the-art Motorola 68000 central processing unit. The Motorola 68000 CPU is used on a computer system designed by EMS (Educational Micro Systems). The ADC's and DAC's would then allow monitoring of electrochemical reactions, with complete control by the computer. Developing programs to monitor and control an electrochemical experiment (pulse polarography, cyclic voltammetry, etc.) and to process data is also a part of this work.

Rationale

The need for the enhancement of electrochemical data is evident (For example: to better control polarigraphic experiments). With conventional 8-bit ADC's and DAC's boththe use of 16-bit ADC's and DAC's a very great enhancement of resolution and dymnamic range is realized.

The need for a flexible and easily adaptable computer system in the lab, with state-of-the-art functioning, is always in demand. With this system availible in the lab, many future uses may be realized.

DATA ACQUISITION SYSTEM

Computer System

The computer system used for this work is the M68K Single Boasd Computer from Educational Microcomputer Systems. This system comes as a kit which was assembled as part of this work. The M68K is based on the Motorola 66000 microprocessor. The M68K has several expansion capabilities; some of these are also used in this work.

The M68K's timing is based on the AM9513 System Timing Controller. u This is a support device that provides the capability for programmable frequency synthesis, high resolution programmable duty cycle waveforms, retriggerable digital timing functions, time-of-day clocking, coinicidence alarms, complex pulse generation, high resolution baud rate generation, frequency shift keying, stop watch timing, event count accumulation, and waveform analysis.[9] The Am9513 has a variety of programmable operating modes which allow it to be personalized for specific uses and to dynamically change its f nctions during a program run.

The M68K also has a Memory/Device Expansion Interface which allows 128/680K memory expansion and interface to data acquistion systems such as: disk storage, ADC's and DAC's. This expansion interface will be used for the data acquistion systen.

The Microprocessor

The Motorola 68000 is a 16-bit external 32-bit internal microprocessing unit. The data system bus is 16 bits wide and all registers in the M68000 are 32 bits wide. The M68000 is somtimes called a 16/32-bit microprocessor.

The memory organization of the 1468000 is very large and can address 2^{24} (16,777,216) bytes in a linear sequence. A word (16 bits) can be obtained in one access and a long word (32 bits) takes two accesses.

The instruction set of the M68000 is made up of 56 basic commands which have numerous variations. One of the most versatile instructions for data acquistion is the MOVE instruction. This instruction is capable of moving data from any memory or register to any other memory or register and of moving data anywhere else.

The M68000 has eighteen 32-bit registers (8 data, 9 address, and 1 program counter), which allow for a vast amount of data manipulation.

In order to allow for smooth efficient data acquisiton, the M68000 has seven different interrupt levels, all of which can be programmed for various uses.[10]

A SIXTEEN BIT
DATA ACQUISITION SYSTEM
FOR ELECTROCHEMICAL INSTRUMENTATION

by

Bruce R. Hahn

Submitted in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in the

Chemistry

Program

Daryl W. Mincey Aug 19, 1985

Advisor Date

Sally M. Hotchkiss August 29, 1985

Dean of the Graduate School Date

YOUNGSTOWN STATE UNIVERSITY

August, 1985

A SIXTEEN BIT
DATA ACQUISITION SYSTEM
FOR ELECTROCHEMICAL INSTRUMENTATION

by

Bruce R. Rahn

Submitted in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in the

Chemistry

Program

Daryl W. Mincey Aug 19, 1985

Advisor Date

Sally M. Hotchkiss August 29, 1985

Dean of the Graduate School Date

YOUNGSTOWN STATE UNIVERSITY

August, 1985

ABSTRACT

A SIXTEEN BIT
DATA ACQUISITION SYSTEM
FOR ELECTROCHEMICAL INSTRUMENTATION

Bruce R. Hahn
Master of Science
Youngstown State University, 1985

The preparation of a sixteen bit data acquisition system is described. Included is a brief introduction to computers, computer languages and their uses in analytical chemistry, especially electrochemistry. The design and construction of circuit boards is included. A sample program for data acquisition is also presented.

ACKNOWLEDGEMENTS

I wish to express my gratitude to Dr. Daryl Mincey for his guidance and encouragement on this project. I also give my thanks to Patty for her patience and support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

II'TTRODUCTION

In the past few years the growth of mini-or-micro-computer use in the laboratory has progressed rapidly. The development and improvement of the computer industry has also taken place very rapidly and the implementation of laboratory computer systems has been consistently growing as well.

The use of computers in analytical laboratories has evolved from data processing to the actual collection and storage of data from a wide spectrum of instruments. Computers have enabled the analytical chemist to perform many tasks which would have been impossible a few years ago. The widest use of computerization in the laboratory has been fourier transform spectroscopy. Computers are becoming-important in electrochemistry by enabling the researcher to both monitor and control experiments such as linear sweep, cyclic, normal pulse, and differential pulse voltammetry. Computers have also freed the analytical chemist from many different analyses which were so time-consuming in the past.

The proliferation of computers in the lab, while making many tasks simpler, has also added a new dimension to the training and functioning of the analytical chemist. Just a look inside the door of a modern analytical lab should convince anyone that the need for computer literacy and an understanding of computer hardware is a must for the complete knowledge of modern analysis techniques.

The availability of computer systems for the laboratory has improved in recent years. Many fine systems are now available for routine analysis. The instruments industry is constantly making the systems easier to use. This improvement throughout the industry is taking place so rapidly that it is difficult for the analytical chemist to make intelligent selections as to the instruments for his or her own needs. Although the industry is rapidly changing and growing, cost and availability of equipment to meet the exact needs of the researcher still remain as prohibitive factors for many laboratories.

In the research lab today analytical chemists must be able to choose a system and be able to adapt it to their needs. To be able to adapt a system to the needs of a research Lab requires the understanding of both the experimentation techniques and an undertanding of the computer itself.

To completely understand a computer system, a working knowledge of basic electronics, computer hardware and computer logic is necessary. A working knowledge of these will enable the researcher to understand the qualities, both good and bad, of the data collected. Chemists with the ability to change or adapt computer systems have the advantage of being able to improve the quality of their work.

Electrical information

The information obt ined in electrochemical analysis is obtained as a, group of particular electrical signals. These signals consist of three different data domains: analog, time, and digital.[1] The raw electrical signals must then be converted into information by analysis of their electrical properties: charge, current, voltage, and power.

Data from analog domains are obtained as the magnitude of charge, current, voltage, and power in the electrical signal. Input and output transducers are usually used to control and measure these analog signals. Input transducers convert nonelectrical energy into electrical energy. An example of an input transducer is a thermocouple device. Output transducers convert electrical energy into another form of energy, usually physical, such as in a dial on a pH or voltmeter.

Analog signals are very useful in electrochemical analysis but they are susceptible to noise and measurements made from analog signals consist of both the signal and the noise. Data which are in digital or time domains are more accurate as they are less affected by noise[1].

Data in the time domain consist of logic-level signals. Electrical signals consist of a high or low logic, the data are made up of the time between successive high and low signals, a period, or the time between low-high and high-low signals, a pulse width, or the number of high-low transitions that occur in a unit of time, frequency.

In the digital domain, data exist also as high-low logic, but data are represented ax a combination of high and low signals which are encoded to represent a certain digit or symbol. Digital domain data are the data which is used in modern electrical logic circuits.

Digital data do not need any further changes or transformations in order to render numerical information, the signal only needs to be decoded. Data from the analog and time domains must be transformed into digital data in order to make use of modern electronic logic circuitry.

Digital signals can be encoded in one channel or in a series of parallel signals. Data represented in a single channel (serial) are encoded as a number of pulses (count waveform). This type of information is less efficient than - the data represented as a parallel series of signals.

The most efficient form of parallel digital signals is known as the binary-coded serial signal[1]. In binary-coded data each pulse in a series represents a binary digit (bit). The resolution achieved with this type of data is very high. In a binary coded signal a series of 16 pulse times would have a resolution of one part in 2^{16} or one part in $6.5536x10^{4}$.

BINARY NUMBERS

In a binary number system, a combination of ones and zeroes represent integer numbers. In our decimal system of numbering, each digit is given a value equal to a power of ten. In the binary number system each digit is given a value equal to a power of two. The farthest number to the right in a decimal system is called the least significant digit and the farthest digit to the right in a binary system is called the least significant bit (LSB). An example of the binary number system is listed in table 1.

Negative numbers are usually represented by two's complement notation in which all 1's are changed to zero and all 0's to ones and then the result is incremented by one. For example: -4 = 1100.

TABLE 1

BINARY NUMBER SYSTEM

Binary				Decimal
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

LOGIC CIRCUITS

In 1854 George Boole wrote "An investigation of the Laws of Thought on Which Are Founded the Mathmatical Theories of Logic and Probabilities"[2]. In this work he introduced the idea of two-state logic, which is the basis of modern electronic logic circuitry. The logic behind this circuitry is known as BOOLEAN ALGEBRA and an understanding of Boolean Algebra is essiential in understanding modern instruments in the analytical laboratory.

Logic circuitry is composed of "gates", known as logic gates, which are the basis of modern digital circuitry.

All two level logic functions are combinations of the three basic logic operations: AND, OR, and INVERT. combinations of these operations make up most digital circuitry. In these logic operations HI AND LO voltages are used to represent true and false respectively. Boolean algebra can express and manipulate the combinations of these logic functions.

Some of the main logic pates are listed in table 2[3].

Table 2

LOGIC GATES

LOGIC GATE	TRUTH TABLE		
AND FUNCTION	$C=A*B$		
	C	A	B
	0	0	0
	0	0	1
	0	1	0
	1	1	1
OR FUNCTION	$C=A+B$		
	C	A	B
	0	0	0
	1	0	1
	1	1	0
	1	1	1
NAND FUNCTION	$C=\overline{A*B}$		
	C	A	B
	1	0	0
	1	0	1
	1	1	0
	0	1	1
NOR FUNCTION	$C=\overline{A+B}$		
	C	A	B
	1	0	0
	0	0	1
	0	1	0
	0	1	1

MICROCOMPUTERS

In recent years the technology behind mini-and microcomputers has grown so rapidly that the advantages of microcomputers compared to mainframes has eclipsed the disadvantages. "In fact, in technical terms there really are no more practical boundaries between microcomputers, minicomputers, and mainframes."[4] In the past, mainframes had a large advantage in the area of storage and programming capabilities which are important for use in obtaining and processing electrochemical information. Today inexpensive microcomputers are availible with sufficient memory and programming capabilities for use in electrochemical work, which gives microcomputers an advantage over mainframes in terms of size and cost.

Electrochemical data, which are analog data, can be transformed into digital data and processed by a microcomputer. The data are transformed by an analog-to-digital converter and transferred through input-output units (I/O) into the computer. The data then go to a Central Processing Unit (CPU) and are transformed through a program into files, graphical analyses, or other useful forms.[5]

The transfer of data as described above is carried out in response to a set of instructions which are logically interpreted and carried out by the CPU. The CPU is the main part of a microcomputer. The integrated circuit (I/C) which makes up the CPU is known as the microprocessor.

The transfer of data takes place along a data bus, which determines the length of the microcomputers word. Data buses in the past were 4 bits or 8 bits wide. Recently, advanced circuit technology has provided microprocessors with 16 bit data buses, and a 32 bit data bus is coming in the near future. The increase in the size of the data bus means a large increase in the resolution of the computer. An 8 bit word length has a resolution of 1 part in 2^8, while a 16 bit word length has a resolution of 1 part in 2^{16} and a 32 bit word length has a resolution of 1 part in 2^{32}!

The CPU sends data to the appropiate memory location or device by using an address code. Each memory location in a computer and every interfaced device must have its own specific address. Addressing takes place along the address bus. The size of the address bus determines the number of address codes availble to the microprocessor. A 16-bit address bus would have 2^{16} possible address codes.

The CPU sends data to the appropriate memory location or device by using an address code. Each memory location in a computer, and every interfaced device, must have its own specific address. Addressing takes place along the address bus. The size of the address bus determines the number of address codes availible to the microprocessor. A 16--bit address bus would have 2^{16} or 65536 possible address codes.

The speed of a microcomputer is determined by its clock. The CPU sequences its operations by the clock's oscillations. The faster the clock, the faster the CPU.

The CPU is made up of registers. A register is a set of latches which can hold one word. For example, an 8-bit register is made up of 8 latches. The primary registers in a CPU are located in the ALU (arithmetic or logic unit). There are also general-purpose and indexing registers within a CPIJ. These registers can hold an address or an instruction.

A microcomputer's memory size also is a factor in determining its utility in the laboratory. Memory devices are made up of storage cells which can hold one bit of information. This memory is divided into two types: Read Only Memory (ROM) and Random Access Memory (RAM, or read-write memory). RAM memory can be changed by the CPU and ROM memory is permanent and cannot be changed by the CPU. One type of ROM is erasable programmable read only memory (EPROM) which can be erased and changed.

RAM memory is made up of Flip-Flop type circuitry. The simplest Flip-Plops have two inputs and one output. One input is a clock input and the other a data input. The data input may change states (HI-LO), however the output will remember the last data, input until a transition in the clock input occurs. This circuitry allows for memory and also a change of memory.

ROM memory is made up of circuitry that has inputs and outputs with no clock input and permanent memory. ROM's come in various sizes. A common ROM has 10 address inputs and 8 data outputs. This gives the ROM 2^{10} or 1024 memory locations 8 bits in length. Eight bits is known as one byte and 1024 is known as 1 K. This type of ROM would be a 1 KByte memory chip.

In order to utilize a microprocessor in a laboratory, it must have Input/Output (I/O) capabilities. Most microcomputers have an input register and an output register which are connected to the data bus. Since the CPU does have complete control over the utilization of the I/O registers, some sort of compatability is required with outside devices: disk memory, printers, data converters, etc. This compatability is achieved through the use of flags and interrupts. These flags and interrupts set certain status bits as high or low. These status bits tell the CPU through a program what to do.

Computer Languages

A computer's language is merely an extension of the logic circuitry within the computer. Since the circuit logic is a combination of HI and LO, a microprocessor's vocabulary consists entirely of HI's and LO's or 1's and 0's. The language of 1's and 0's is known as machine language. For the laboratory chemist, as with most users, machine language is too difficult to use as a programming language. Over the years programmers as well as scientists have developed human type languages for operating a computer.

Assenbly language is usually the first language developed for a computer system. Assembly- language is a set of codes and mnemonics, which directly represent machine language instructions.[6] Assenbly language ("Assembler") is translated by an assembler into the machine code. Assembler gives the user complete control over the computer, is often time consuming and confusing, but for electrochemical interfacing a knowledge of this language is needed.

Programmers have developed many higher level languages, languages which opera-te differently from the nicroprocessor, which are very useful in the laboratory. These languages require an operating system in order to function. Each of these languages has its advantages and disadvantages in regard to research work.

For use in the laboratory, a language requires many different characteristics. Some of these characteristics are listed below.[7]

TABLE 3

LANGUAGE CHARACTERISTICS

1. Mathematic and logic funcions.
2. Variables and constants.
3. Ability to define new operations and commands.
4. A set of conditionals and repetitives which allow branching and loops.
5. Reentrancy or sharing of routines among users.
6. Pointer or labeled addressing.
7. User interactive.
8. Large storage and array capabilities.
9. Immediate use of the primary functions, such as +,-.
10. The language should also allow multitasking (the ability to carry out more than one operation at a time. i.e., control data and acquisition of an electrochemical experiment by the computer and various instruments.)

The two most common languages in use in the laboratory are BASIC and FORTRAN. These languages are good for number manipulation (crunching), but both are lacking in speed in the solving of complex problems, and in interfaced-instrument control.

A relatively new language, FORTH, is being used. by many instrument manufacturers.[7] FORTH meets the requirements of a good laboratory language and it is relatively simple and extremely fast.

FORTH was started as a language to control large telescopes[7]. Being written in mind as an instrument controlling language gives FORTH an advantage in the electrochemical lab.

ELECTROCHEMICAL DATA ACQUISITION

All electrochemical data exist in the analog domain. In order to be able to process these data they must be either recorded manually and entered into a computer, or they must be converted to digital data with the use of an analog-to-digita- converter. An analog-to-digital converter will take an analog voltage input and convert it into a parallel digital data output, which can then be transferred to the computer for storage and processing.

Electrochemical data are in the analog form and so the instrument that controls the experiment, such as a potentiostat, operates in the analog domain. In order to have control over the experiment, a data acquisition system must have digital-to-analog converters. A digital-to-analog converter accepts parallel digital data and converts them into an analog voltage output.

Many different types of analog-to-digital converters (ADC's) are availible on the market. Most of these are either feedback ADC's, or integrating type ADC's. A brief discussion of both types follows.

Feedback ADC's

Feedback ADC's consist of a comparator, a DAC, and digital logic circuitry.[8] When an analog signal is received, the digital logic signal will increment a digital input to the DAC until the comparator senses that the DAC output and the analog input are equal. Once the two signals are equal, the digital number in the DAC will be the digital output of the ADC.

Three types of feedback ADC's are commonly used: the ramp type, the tracking type, and the successive approximation type. The ramp type simply starts at zero volts ramps and compares it to the input until the signals are equal. The tracking type ADC will start like the ramp type ADC but it also has the ability to decrement the digital signal so a more precise equality between the analog input and the digital output exists. The third type of feedback ADC is the successive approximation type. This ADC is much faster than the ramp or tracking types. The ramp and tracking type ADC's compare by incrementing the least significant bit (LSB) until an equality exists. The successive approximation type compares the most significant bit first and increments or decrements it accordingly. The conversion time (T_c) for this method is

$$T_c=(n+1)T_{cp}=(n+1)/F_{cp}$$

where n is the number of bits and F_{cp} is the clock frequency. The conversion time for a ramp type ADC is

$$T_c=(V_{in}/I_{ref}R_{in})x2^nTcp$$

I_{cp} is the clock period ($T_{cp}=1/F_{cp}$). The successive approximation conversion is easily seen to be much faster.

Integrating ADC's

ADC'S have been developed that give a digital ouput depending on the average or integral of the analog input over a given time period. These types of ADC's will give good results even when high frequency noise interferes with the analog input. Two types of integrating ADC's exist: dual-slope and quantized-feedback.

In the dual-slope type the analog input is integrated for a period of time. If the signal is constant then the integrator will output a linear ramp. Then a reverse reference voltage is applied to the ramp and the time it takes to go back to zero will be proportional to the analog input and the digital output is taken from that time or a corresponding number of clock pulses.

The quantized method is similar to the dual-slope, except that the conversion processes occur simultaneously.

The importance of knowing these different types of converters lies in whether the research needs a fast or an accurate type of data conversion, for most electrochemical data, an accurate type would be the preference. Integrating converters are the most accurate, but also the slowest. Tracking type converters lack the accuracy of the integrating converters and the speed of the succesive approximation converters, therefore they are rarely used. Sucessive approximation converters are the most freyuently used because they are a compromise of both speed and accuracy.

Digital-to-Analog converters exist in many different forms, however the type used is not as critical as with ADC's.

Peripheral Interface Adapter

Once the DAC's and ADC's have been selected, communication with the CPU must be established. This is done with the use of a Peripheral Interface Adapter (PIA). Programmable and nonprogrammable PIA's exist. The choice for the researcher should be the programmable type. The programmable type provides much more flexibility and it can be directly addressed by the CPU.

PLA's serve as the interface between the various I/O devices (printers, disks, data converters, etc...) and the CPU. Since each I/O device has its own address the PIAs must be programed to respond to that particular address.

A p ogrammable PIA can generate interrupts (usually by setting a flag HI or LO) which tell the CPU that new data are ready. The programmable PIA can also be programmed to provide "handshaking" with the CPU. Handshaking occurs when the interrupt flag is set and the CPU comes back and sets another flag (usually the data request line) which tells the PIA whether or not it is ready for the new data.

PIA's contain their own registers for storing, addressing, and programming. The researcher can program the PIA's for his needs. Common PIA's have two 8 bit parallel ports. Each port also has two control lines which can be set HI or LO to provide a variety of functions. The control lines can be set four different ways to control the 8 bit ports. The ports can be set as input or ouput lines, to

r

generate interrupts, or to serve as handshaking, lines to the CPU.[3]

PIA's are made to interface with only one microprocessor type, so each microprocessor manufacturer usual-ly makes it's own PIA chip.

JUSTIFICATION

Electrochemical analysis with the aid of microprocessors and minicomputers has become commonplace in the laboratory. Most of the work done to date has been with 8-bit microcomputers or with minicomputers, using 12-bit analog-to-digital and digital-to-analog converters (ADC's, DAC's). With the electronics available today, it is possible to greatly improve both the resolution (the magnitude of the smallest detectable variation in a quantity) and dynamic range (the range or amount of change in a quantity, i.e., voltage) of these experiments with the use of a 16-bit microcomputer and 16-bit ADC's and DAC's.

The use of these converters should provide a better view and understanding of electrochemical reactions. With the improvement of resolution and dynamic range comes better monitoring and controlling of electrochemical experimentation.

The speed of the computer systems is another factor to consider when improving upon existing systems. Computer speed for input and output (I/O) routines depends upon the Central Processing Unit's (CPU) clock and the computer language used to run the routines. Combining both a fast CPU clock with a low level language is the best way to optimize the speed of data acquisition. While some commercially availble systems use this combination, the research chemist should be aware of which do and which do not and the limitations imposed.

A computer's memory is also another factor the researcher must take into consideration when designing or adapting a system to his use. While data, can be stored on disk or tape systems, data taken during an experiment are much more efficiently stored in a computer's on board random access memory (RAM). A system with a large RAM will allow the researcher to store the data quickly and efficiently during the experiment and data transfer can be made to disk or tape storage after the run of the experiment.

Computer systems must also be flexible. Many commercially designed computer systems for the laboratory are designed. with limited user applications. Many times a commercially supplied computer is only capable of running one instrument or performing one function in the lab. For the researcher this can lead to problems, especially when the total needs and demands of a computer system cannot be determined in advance.

In summary, a good computer system for the researcher should have good interfacing capabilities, high resolution, and a large dynamic range. It should be fast, with a large memory size, and designed with flexibility in mind.

CHAPTER 2

STATEMENT OF PROBLEM

Purpose

The purpose of this work is to design and build 16-bit analog-to-digital and digital-to-analog converter systems and interface them with a state-of-the-art Motorola 68000 central processing unit. The Motorola 68000 CPU is used on a computer system designed by EMS (Educational Micro systems). The ADC's and DAC's would then allow monitoring of electrochemical reactions, with complete control by the computer. Developing programs to monitor and control an electrochemical experiment (pulse polarography, cyclic voltammetry, etc.) and to process data. is also a part of this work.

Rationale

The need for the enhancement of electrochemical data is evident (For example: to better control polarigraphic experiments). With conventional 8-bit ADC's and DAC's boththe use of 16-bit ADC's and DAC's a very great enhancement of resolution and dymnamic range is realized.

The need for a flexible and easily adaptable computer system in the lab, with state-of-the-art functioning, is always in demand. With this system availible in the lab, many future uses may be realized.

DATA ACQUISITION SYSTEM

Computer System

The computer system used for this work is the M68K Single Board Computer from Educational Microcomputer Systems. This system comes as a kit which was assembled as part of this work. The M68K is based on the Motorola 68000 microprocessor. The M68K has several expansion capabilities; some of these are also used in this work.

The M68K's timing is based on the AM9513 System Timing Controller. This is a support device that provides the capability for programmable frequency synthesis, high resolution progr mable duty cycle waveforms, retriggerable digital timing functions, time-of-day clocking, coinicidence alarms, complex pulse generation, high resolution baud rate generation, frequency shift keying, stop watch timing, event count accumulation, and waveform analysis.[9] The Am9513 has a variety of programmable operating nodes which allow it to be personalized for specific uses and to dynamically change its functions during a program run.

The M68K also has a Memory/Device Expansion Interface which allows 128/680K memory expansion and interface to data acquistion systems such as: disk storage, ADC's and DAC's. This expansion interface will be used for the data acquistion system.

The Microprocessor

The Motorola 68000 is a 16-bit external 32-bit internal microprocessing unit. The data system bus is 16 bits wide and all registers in the M68000 are 32 bits wide. The M68000 is somtimes called a 16/32-bit microprocessor.

The memory organization of the M68000 is very large and can address 2^{24} (16,777,216) bytes in a linear sequence. A word (16 bits) can be obtained in one access and a long word (32 bits) takes two accesses.

The instruction set of the M68000 is made up of 56 basic commands which have numerous variations. One of the most versatile instructions for data acquistion is the MOVE instruction. This instruction is capable of moving data from any memory or register to any other memory or register and of moving data anywhere else.

The M68000 has eighteen 32-bit registers (8 data, 9 address, and I program counter), which allow for a vast anount of data manipulation.

In order to allow for smooth efficient data acquisiton, the M68000 has seven different interrupt levels, all of-which can be programmed for various uses.[10]

Digital-to-Analog Converter.

The DAC used in this work is a 16-bit, monolithic type converter manufactured by the Burr-Brown Corpora-tion. This is a very accurate, .003% linearity error and .006% differential linearity error, monotonic converter. Monotonic converters continuously increase their analog output as the digital input is incremented.

Analog-to-Digital Converter.

The ADC used in this work is a 16-bit, dual-slope integrating type ADC manufactured by Intersil. This ADC has accuracy !guaranteed to 1 count, also it has true polarity at zero count for precise null detection.

A dual-slope integrating ADC was chosen to eliminate background noise from the incoming analog signal. The disadvantage with this choice is the speed at which it operates, however in most electrochemical work, speed may be less important than accuracy.

Peripheral Interface Adapter

The PIA used in this work is the Motorola 6821 PIA. This PIA has two 8-bit channels or ports. These ports can be set as either input or output ports. Each port has three 8-bit registers which control and store data for the corresponding port. A block diagram of the Motorola 6821 PIA is shown in figure 1.

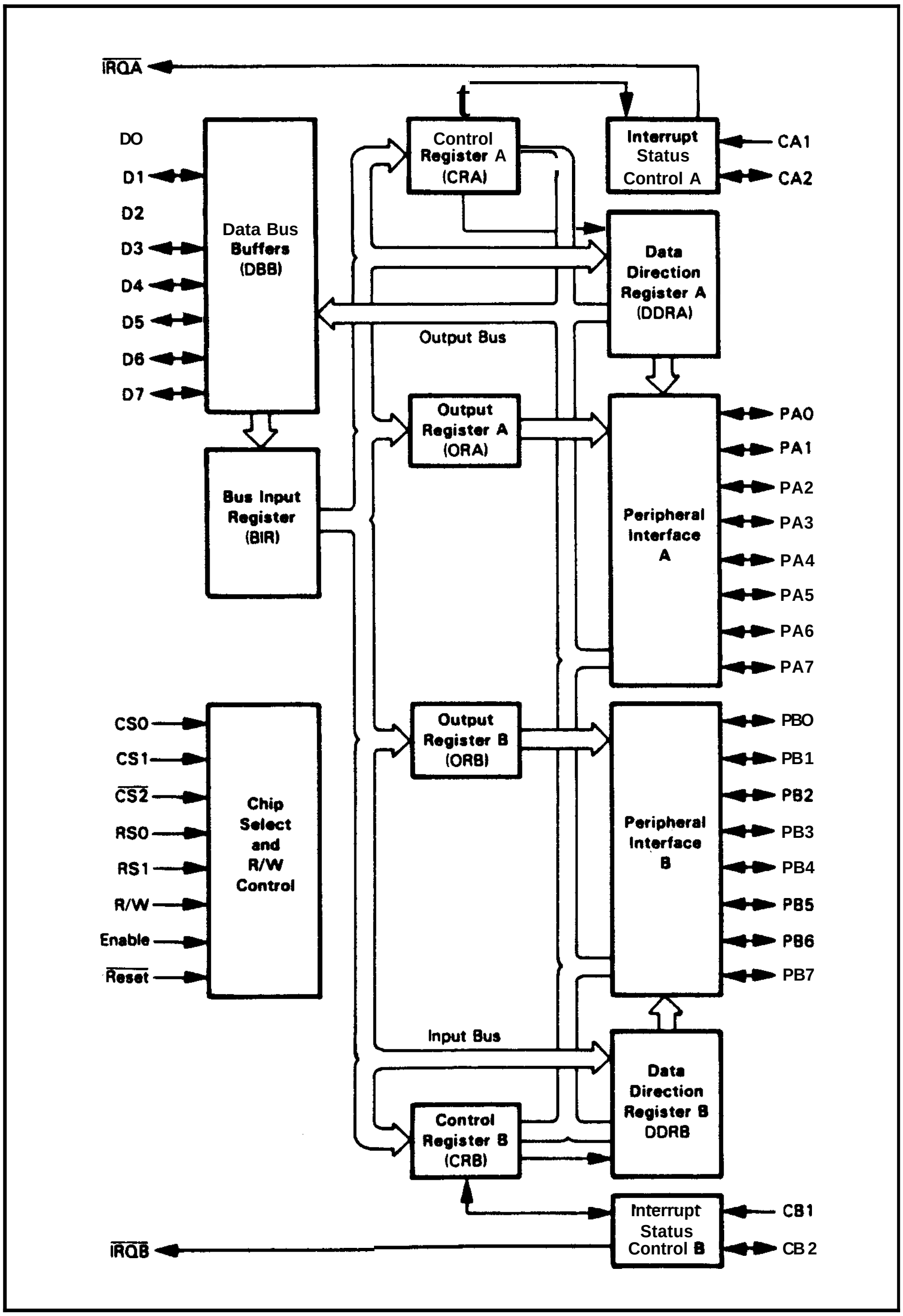


Figure 1. Block diagram of the Motorola 6821 PIA.

PORTH Language

The FORTH computer language has been chosen for this work, partially due to its flexibility and ease of use with instrumentation. The other reason for using FORTH is the increased speed in handling and manipulating of data obtained with FORTH.

As FORTH was first developed for the control of large telescopes,[7] it is a logical choice for use in the laborato y, especially for controlling instruments.

FORTH is an incremental compiler, it compiles text as it is input.[11] All inputs to Forth are interpreted and executed directly, unless the input is enclosed in a colon-definition. Data compiled this way allow for ease of manipulation and processing.

FORTH actually functions in two different modes. One mode is the compile or defining mode, and the other is the execution mode. All sources of input are handled exactly the same way in FORTH. Definitons are converted to a data structure and stored as such.

FORTH's data structure is divided into three parts: dictionary, outer interpreter, and the inner interpreter.

The dictionary is actually a special structure of program entries. These entries are linked together and allow for searching and starting at the last program entered.

The dictionary enables the user to define words or addresses as functions or programs. The orde of entry in the dictionary determines the order of the program. For example a definition or program cannot be used in another definition unless it was previously defined or programmed.

The outer interpreter has two functions: to search the string and to execute the routine, and to compile new definitions or programs. The outer interpreter determines the order of execution and serves to search for definitions, if called for within another definition.

The inner interpreter serves as a "virtual computer emulator"[7] by keeping track of the addresses of all definitions or programs entered by the user. As the outer intrepreter compiles the definitions an address is assigned to each definition or program. This inner interpreter serves to keep track of these addresses so the outer intrepreter can determine the order of execution.

FORTH is totally dependant on the user stack. A stack is an area of memory where data is temporarily stored. Data is stored in the first-in last-out type of stack.

The manipulation of the data or user stack is very important to the data acquisition system. The stack operators are listed below:

TABLE 4

FORTH STACK OPERATORS

1.	SWAP	this switches the order of the last two numbers in the stack.
2.	DROP	this drops the last number in the stack.
3.	OVER	this puts a copy of the second to the last number at the top (last) of the stack.
4.	DUP	this copies the last element and places it at the top (last) of the stack.
5.	ROT	this puts the third number at the the top (last) of the stack and pushes the first two numbers down.
6.	-ROT	this puts the top number at the third position and pushes the second up to the first and second spot.
7.	PICK	this copies the n^{th} element of the-- stack and places it at the top.
8.	ROLL	this places the n^{th} element at the top and pushes the others downward.

FORTH uses its first-in last-out storage when doing mathematical operations. Because of this math functions must be expressed in the Lobachevski (Polish or postfix) notation. For example: 4 * 5 is expressed as 4 5 *.

The reason for this Polish notation is speed and simplicity. In most high level languages a compiler must convert mathematical infix notation to Polish notation in order to utilize the CPU's registers; in FORTH the programmer does this and one stack is used rather than two. This allows for a simpler compiler and greater operating speed.

Forth also contains user variables and storage operators. The commands "constant" and "variable" allow the programmer to set constants and variables. The command "allot" allows the user to set up arrays. For example:

```
0 VARIABLE INPUT 499 ALLOT
```

In this operation, input is defined as a variable array consisting of 500 elements. In order to access this array the command "@" is used. In the following command one piece of data is placed in the input array:

```
0 VARIABLE DATA
INPUT @   INPUT 1 + @   +   DATA ;
```

In the command, DATA is set up as a variable, values for DATA could come from ADC circuitry, disk input, etc. The value that DATA has at the time of execution is then placed into the first space in the INPUT array. If the value needed to be placed into the fourth space, the following would be used instead:

```
INPUT @   INPUT 4 + @   +   DATA ;
```

Once the commands are learned, the ease of using FORTH for data manipulation is realized.

FORTH's vocabulary can also be augmented by the programmer through the use of user definitions. These definitions are known as COLON definitions. A COLON definiton co ld be thought of as a miniprogram. For example:

```
: FIT   DUP 100 < IF
                ." no good"
                DROP
             ELSE
             200 < IF
                ." Good "
             ELSE
                ." no good "
           THEN    THEM;
```

In the above FIT is defined as a function which will -- determine if a number fits between the values of 100 and 200.

For data acquisition, FORTH has the distinct advantage of jumping into and out of assembler language. This is done using the CODE type of definition. By using "CODE" at the start of a definition, the program can jump into an assembly language subprogram, a data acquisition program for example, and then the next command will bring about a return to FORTH. While most high level languages require extensive programming to use both assembler and the high level language, FORTH is capable of handing CODE type definitions along with the rest of its system. The ability to do this allows for great flexibility and fast data acquisition in the real-time environment,

In electrochemical analyses FORTH is very useful for multitasking. FORTR uses the STATUS and ADDRESS definitions to accomplish -this. In order to control electrochemical instrumentation, both input and output to the instrument must be controlled. When using Forth, the STATUS definition can be set as either a negative or positive value. A positive value allows for input and a negative for output; -4 would allow for the output of four words and -1-4 would allow for the input of four words. When the STATUS value goes to zero then 'the data transfer is complete. The ADDRESS contains the initial memory address involved.

FORTH uses- a circular or round-robin type scanner-to monitor the STATUS values for all tasks involved in the system. This type of scanning is very fast. A data acquisition system using an A/D converter has been bench-marked at 120 points/sec/channel with a 2KHz throughput.[15]

FOHTH is a very capable language for laboratory use.

Through the use of its definition and tasking capabilities, it is only limited by the capabilities of the researcher or programmer. "FORTH is serving as a greenhouse for the development of new concepts in laboratory and applications software".[16]

Printed Circuit Boards

A major part of this work was the design and construction of the printed circuit boards for the analog-to-digital, digital-to-analog, and periphial interface adaptor circuitry. Circuit board designing and etching proved is a time-consuming effort, which requires a great deal of practical knowledge and expertise.

The decision to make printed circuit boards was based on their many advantages over conventionally wired boards. Some of these advantages are listed below:

1. Location of parts is fixed.
2. Miswiring and short-circuited wiring is not normally possible.
3. The wiring is permanently attached to the dielectric base; this base also provides a mounting surface for the components.
4. The size and bulk of the circuitry is greatly reduced.
5. The circuitry is easier to debug and change.

The great disadvantages are that preparation of printed circuitry requires skill and for small numbers of circuits the expense is relatively high.[12]

Layout and Design

The first step in making the printed circuit boards is preparing the original schematic. These were drawn up based on the manufacturer's specifications for each of the components used. When designing the data acquisiton circuitry, the type of addressing used also had to be taken into account.

Once the schematic is drawn, the artwork for the printed circuit board is made. In drawing the artwork, all the circuitry is drawn at twice the actual size. This allows for ease and clarity of design. The drawings are then taped.

In taping, a clear sheet of plastic is placed over the top of the drawing, and black tape is used to overlay the drawing. Tape for the chip sockets and other components can be bought predesigned at double size.

The taped image of the artwork is then photographed and reduced. A negative of this photograph is then used for the etching process. Here, the negative allows the drawn circuitry to be-exposed to UV light while blocking the-rest of the board from exposure. The UV light initiates the crosslinking reaction described in the next section.

Once the artwork is ready, a piece of copper plated board, usually PVC type, is cut to size and prepared.

Board Preparation

The first step in preparing the copper plated board is cleaning and polishing. Several methods were tried. Cleaning with water and a copper polishing compound was found. to be the best. The compound used was a commercial product containing oxalic acid. Pumus powder was also found to be effective.

After cleaning and drying, the board is then coated with. a negative photoresist polymer. A 3 to 1 dilution of chlorobenzene and the polymer was found to coat with the best results. The polymer used was 2-hydroxyethyl methacrylate(HEMA), purchased from KODAK.

This polymer is then dried and the negative is placed over the board. The board is then exposed to UV light. The UV light crosslinks the polymer, which is then resistant to trichloroethylene and an etching bath. The unexposed polymer is dissolved off the board with trichloroethylene(C_2HCl_3).

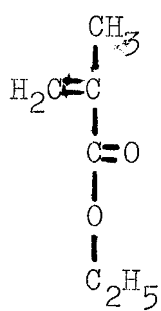
The board is then placed in an etching bath containing iron (III) chloride($FeCl_3$) and hydochloric acid(HCl). The bath is sprayed onto the board and the areas not protected by the crosslinked polymer are oxidized by the iron (III) ion. The resulting copper (II) (CuCL2 dissolves off the board, which now has the circuit design remaining.

Circuit Board Chemistry

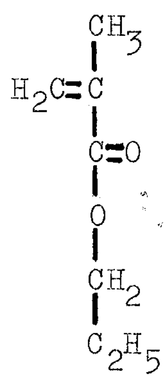
Photoresists

The polymer used for photoresists is actually a co-polymer of ethylacrylate (EA) and 2-hydroxyethylmethacrylate (HEMA). Once the polymer is synthesized, a photosensitive group, cinnamoyl chloride (C_9H_6ClO) is added to the polymer, shown below:[12]

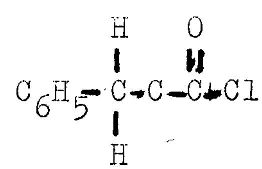
Monomers



(EA)

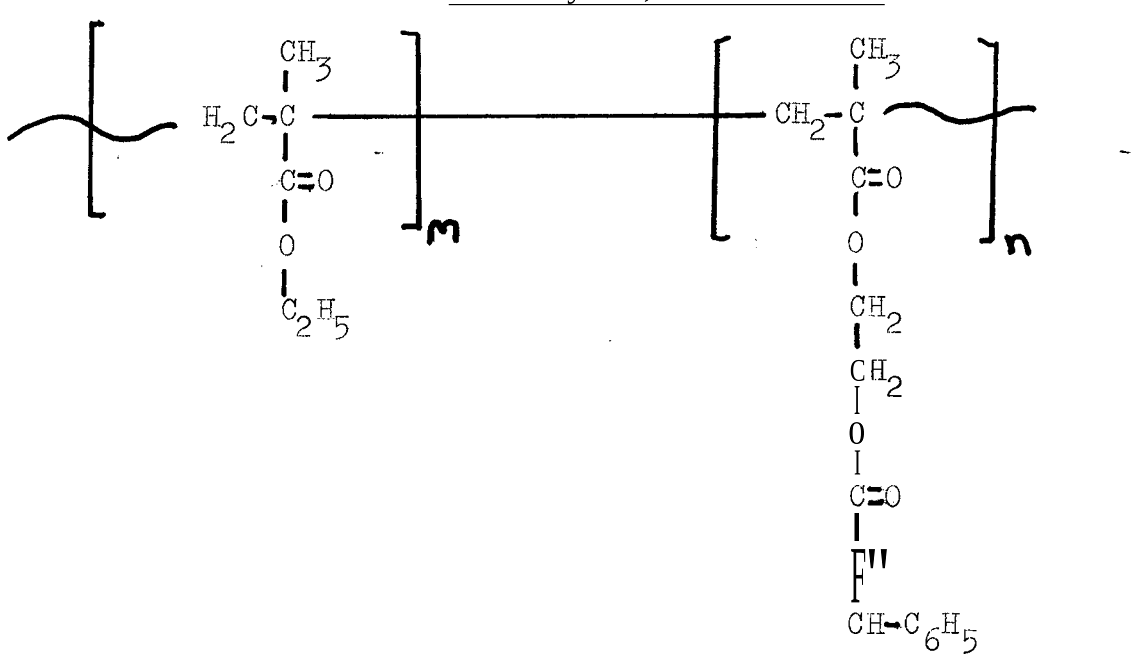


(HEMA)

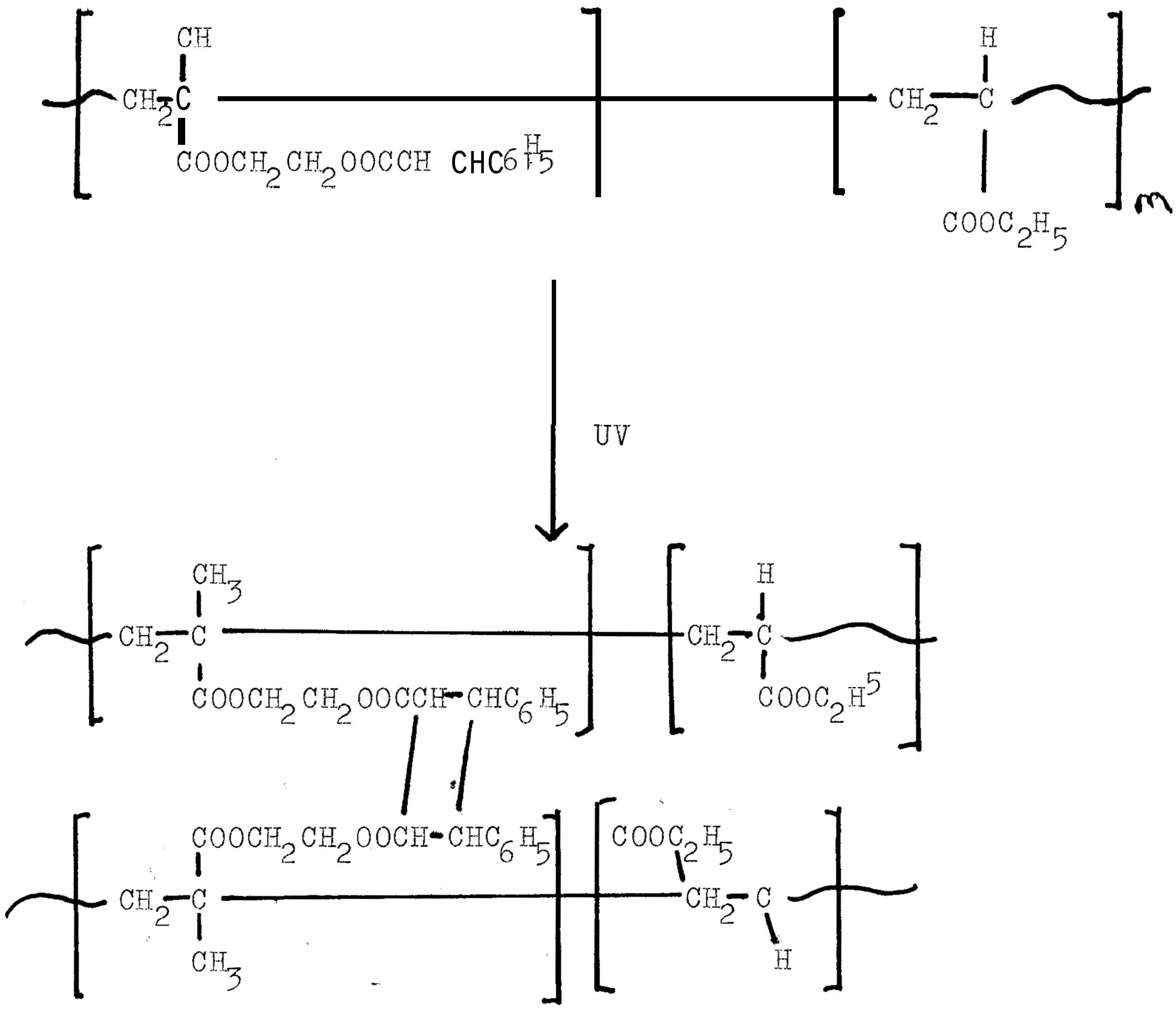


Cinnamoyl Chloride

Co-Polymer, Photoresist



The polymer is then exposed to UV as shown below:



Etching

The etching solution used in this work was a iron (III) chloride solution. The composition of this solution is 28 to 42% $FeCl_3$ in 5% HCL. An additional, small amount of acid is present due to the hydrolysis reaction:

$$FeCl_3 + 3H_2O \rightarrow Fe(OH)_3 + 3HCl$$

When the etchant is sprayed onto the board, the iron (III) ion oxidizes the copper to the copper (I) ion and green iron (11) chloride forms:

$$FeCl_3 + Cu \rightarrow FeCl_2 + CuCl$$

In the etchant, the copper (I) chloride is further oxidized to copper (II) chloride:

$$PeCl_3 + CuCl \rightarrow FeCl_2 + CuCl_2$$

With build up of copper (II) chloride in the solution, an additional reaction takes place:

$$CuCl_2 + Cu \rightarrow 2CuCl$$

It has been found that 84% of the copper is etched by the second reaction.[13]

The rate of etching is controlled by the $PeCl_3$ -- concentration, the temperature, and the type of agitation. For this work a commercial etching sprayer, the Kepro BTE-202 double side spray etcher, was used and the etchant was kept at $38^{\circ}C$. The $FeCl_3$ concentration was approximately 35%.

The time to etch the copper was determined by trial and error. With the etching solution kept as described above it was found that a double sided board about one square foot in size required six to seven minutes in tne etching solution.

After etching, the board is rinsed with water and washed with acetone to remove the photoresist. This is necessary to assure good soldering.

Circuit Board Schematics and Artwork

On the following pages the schematics and artwork for the circuit boards are presented.

TABLE 5

816821 PIA pin assignments[14]

Pin #	Assignment
1	Ground
2–9	Port A Data Lines
10–17	Port B Data Lines
18	Interrupt Input, Port B
19	Peripheral Control, Port B
20	+5 volts
21	Read/Write
22–24	Chip Selects
25	Enable
26–33	Data Lines
34	Reset
35,36	Register Selects
37,38	Interrupt Requests
39	Peripheral Control, Port A
40	Interrupt Input, Port A

TABLE 6

Computer connector assignments

Pin#	Assignment
1,2	Ground
3-16,37,38,47	Bus Address Bits
17-19	Select Signals
21-36	Bus Data Bits
39	Upper Data Strobe
40	Lower Data. Strobe
41	Write Enable
42	Data Acknowledge Input
43-46	Interrupts
48	Immediate Data Acknowledge Input
49,50	Ground

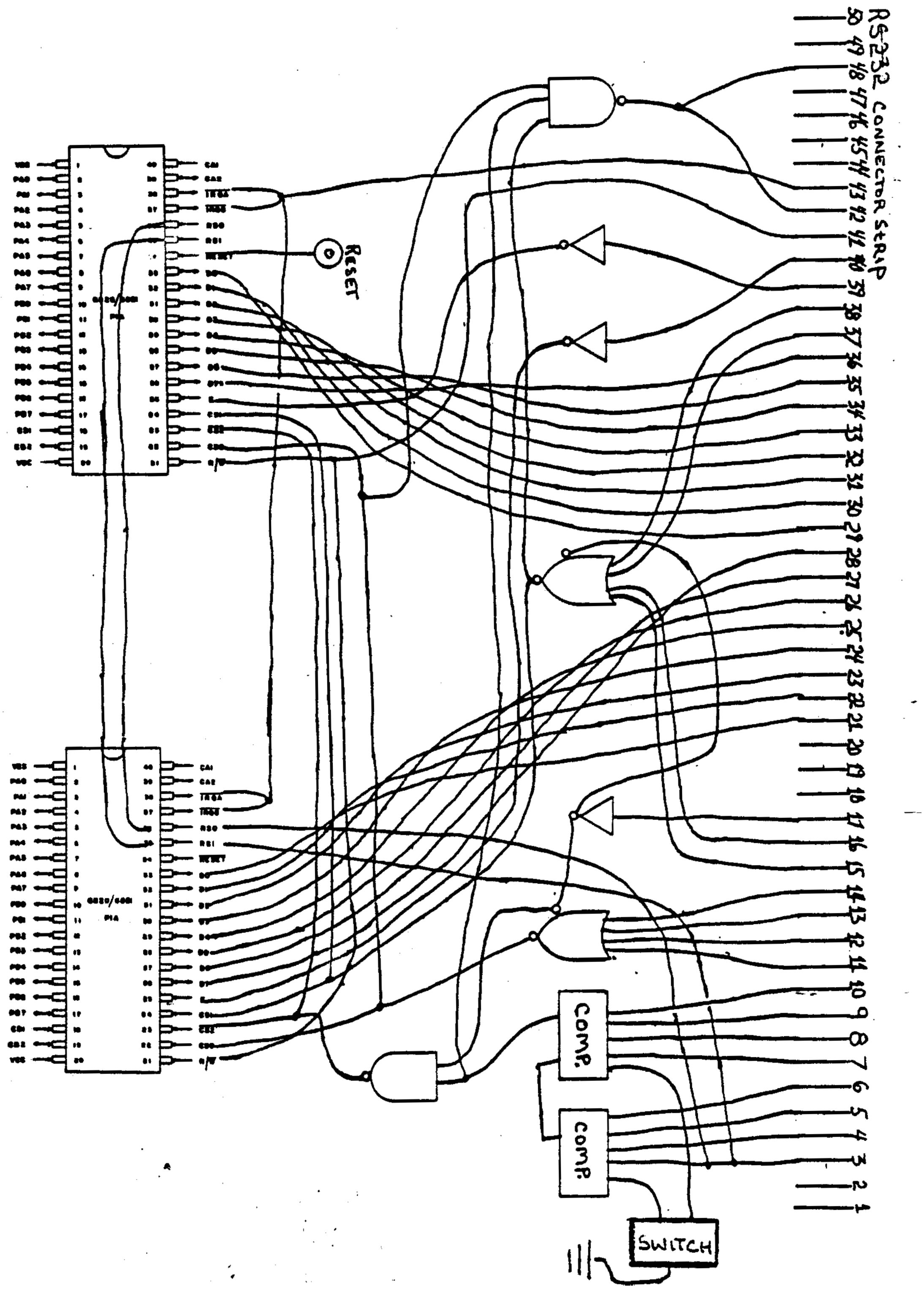


Figure 2. Schematic for the M6821 PIA interface

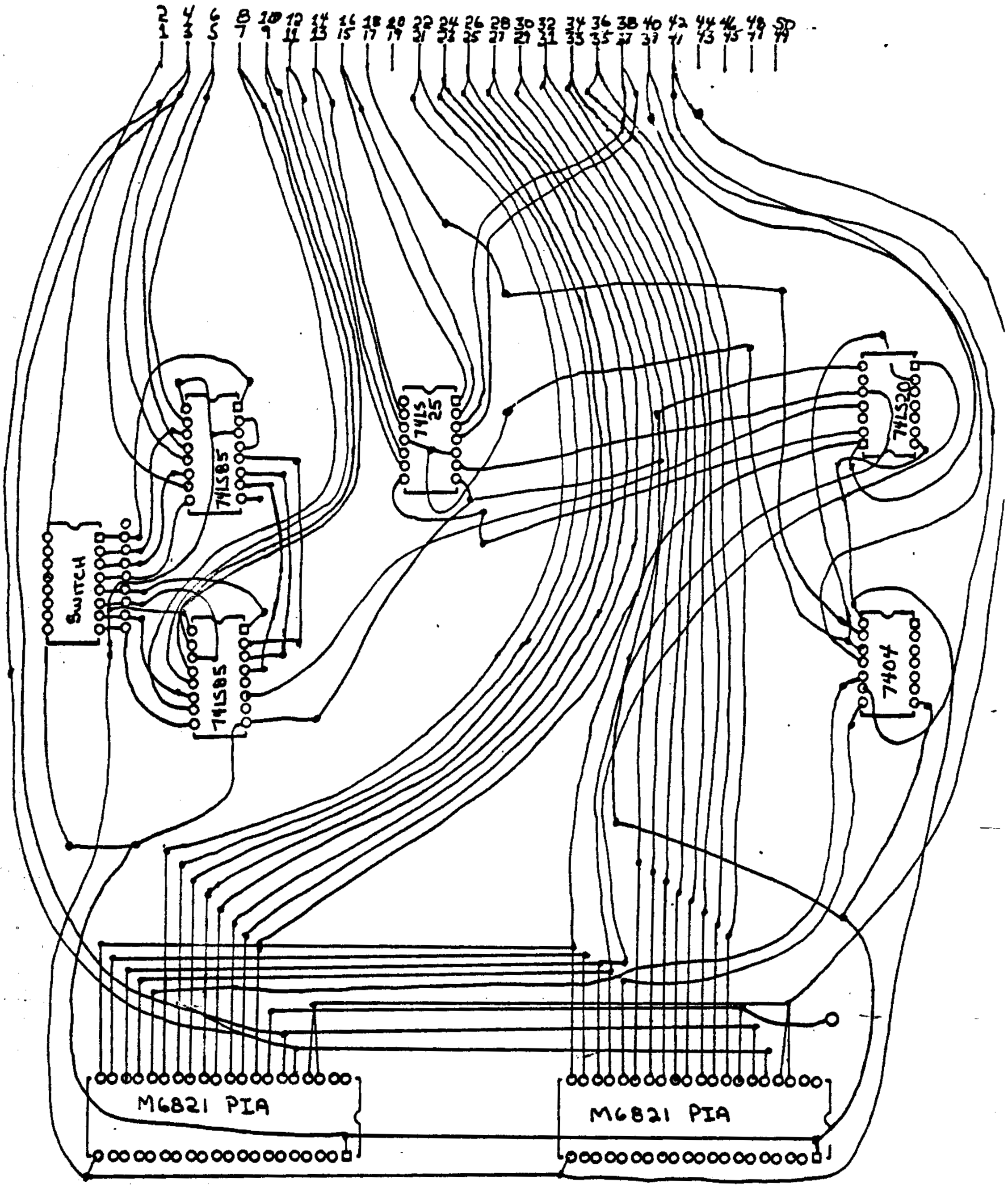


Figure 3. Artwork for the M6821 interface board.

TABLE 7

ICL7104 Analog-to-Digital Converter pin assignments

Pin#	Assignment
1	+15 Volts
2	Ground
3	Status Output
4	Polarity
5	Overrange
6-21	Data Bits
22	Low byte enable
23	High byte enable
24-26	Clocks
27	Mode
28	Run/Hold
29	Send-Enable
30	Chip enable/Load
31	+Logic supply voltage
32	Analog, input
33	Buffer input
34,38	Reference capacitors
35	Analog ground
36	Auto zero
37	Voltage reference
39	Comparator input
40	-Supply voltage

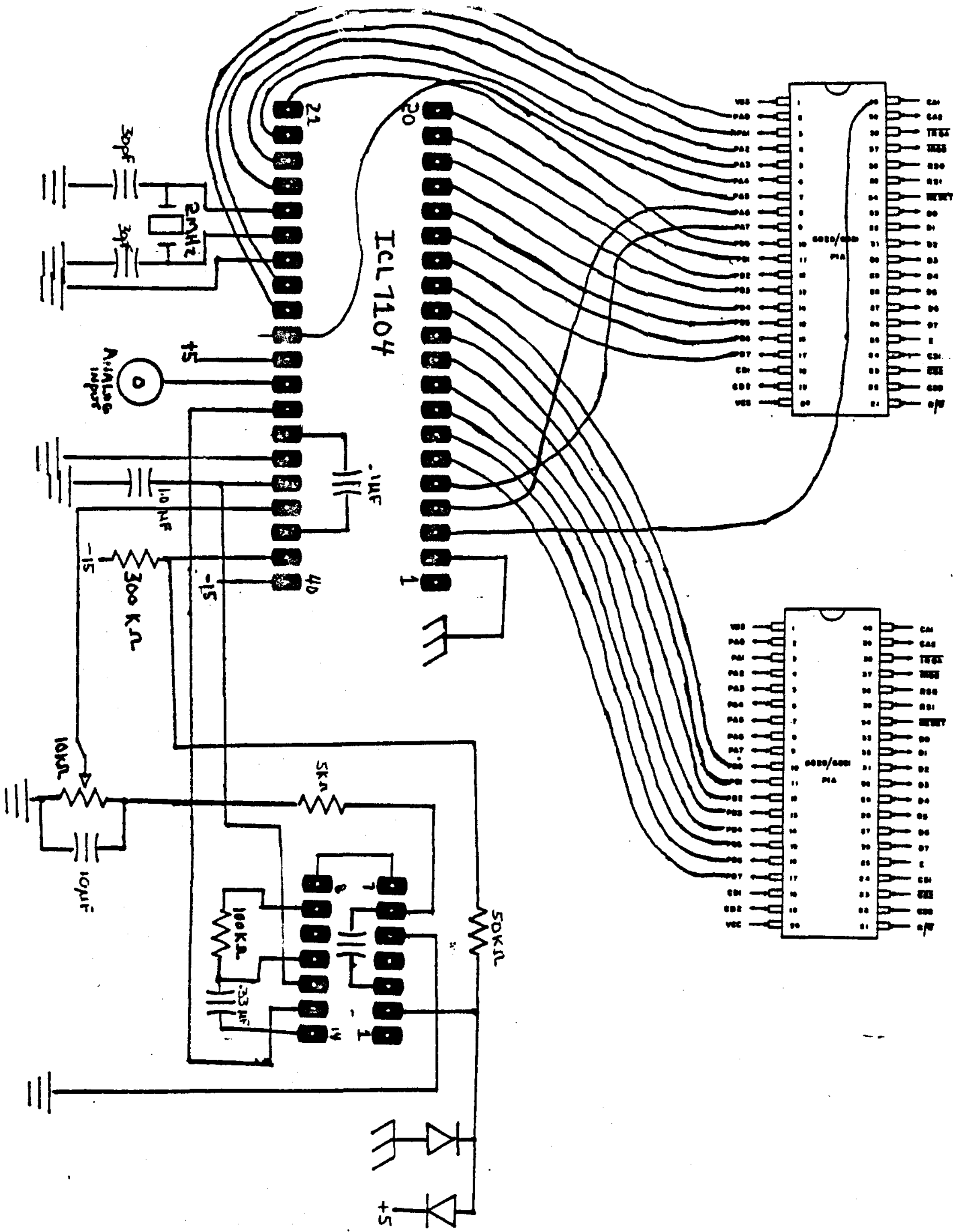


Figure 4. Schematic of the ICL7104 A/D converter interface board.

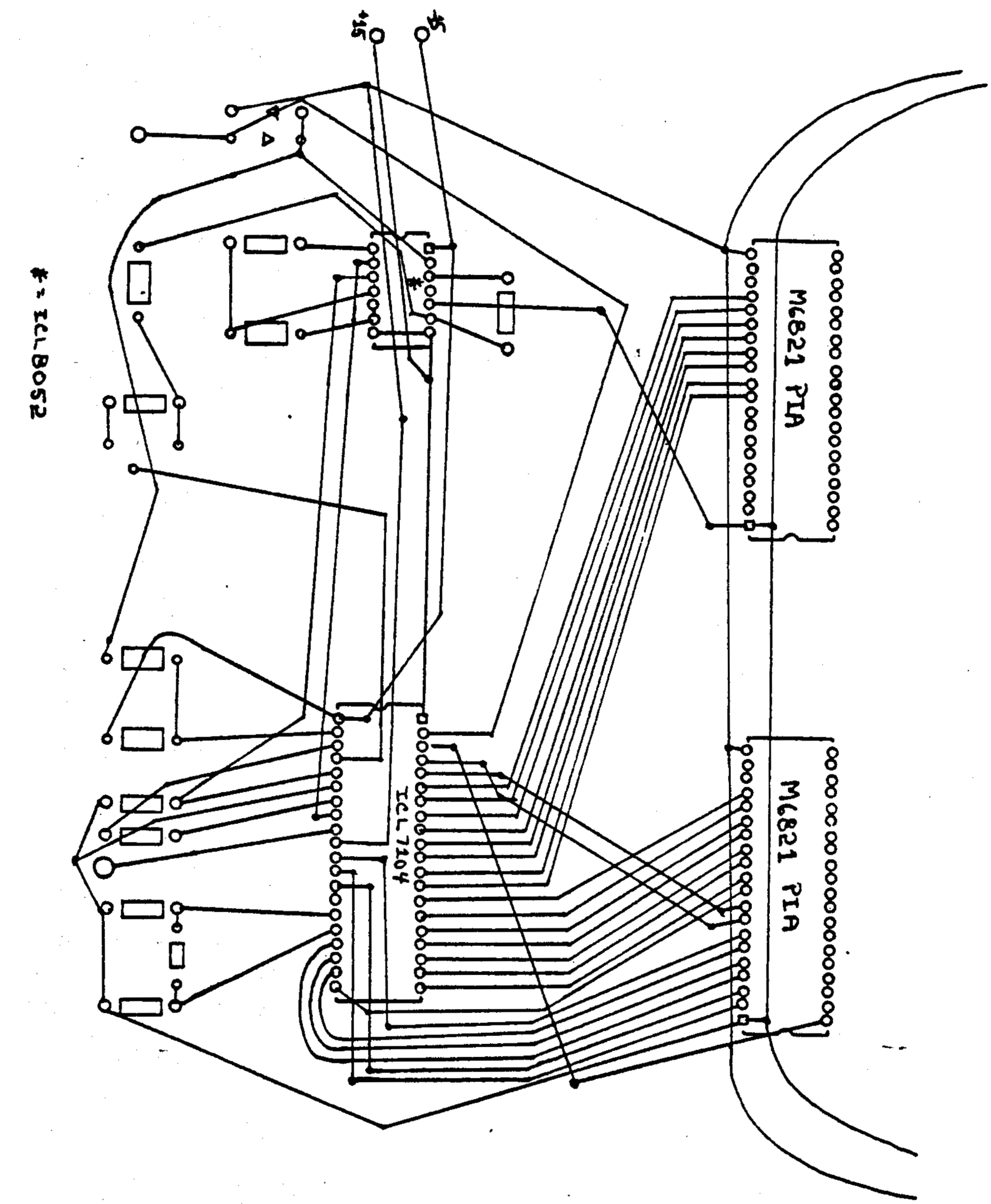


Figure 5. Artwork for the ICL7104 A/D converter interface board.

TABLE 8

DAC703 Digital-to-Analog converter pin assignments

Pin#	Assignment
1-16	Data bits
17	Voltage output
18	+Logic supply voltage
19	-Voltage supply
20	Ground
21	Zero adjust
22	Gain adjust
23	+Voltage suppply
24	Reference output

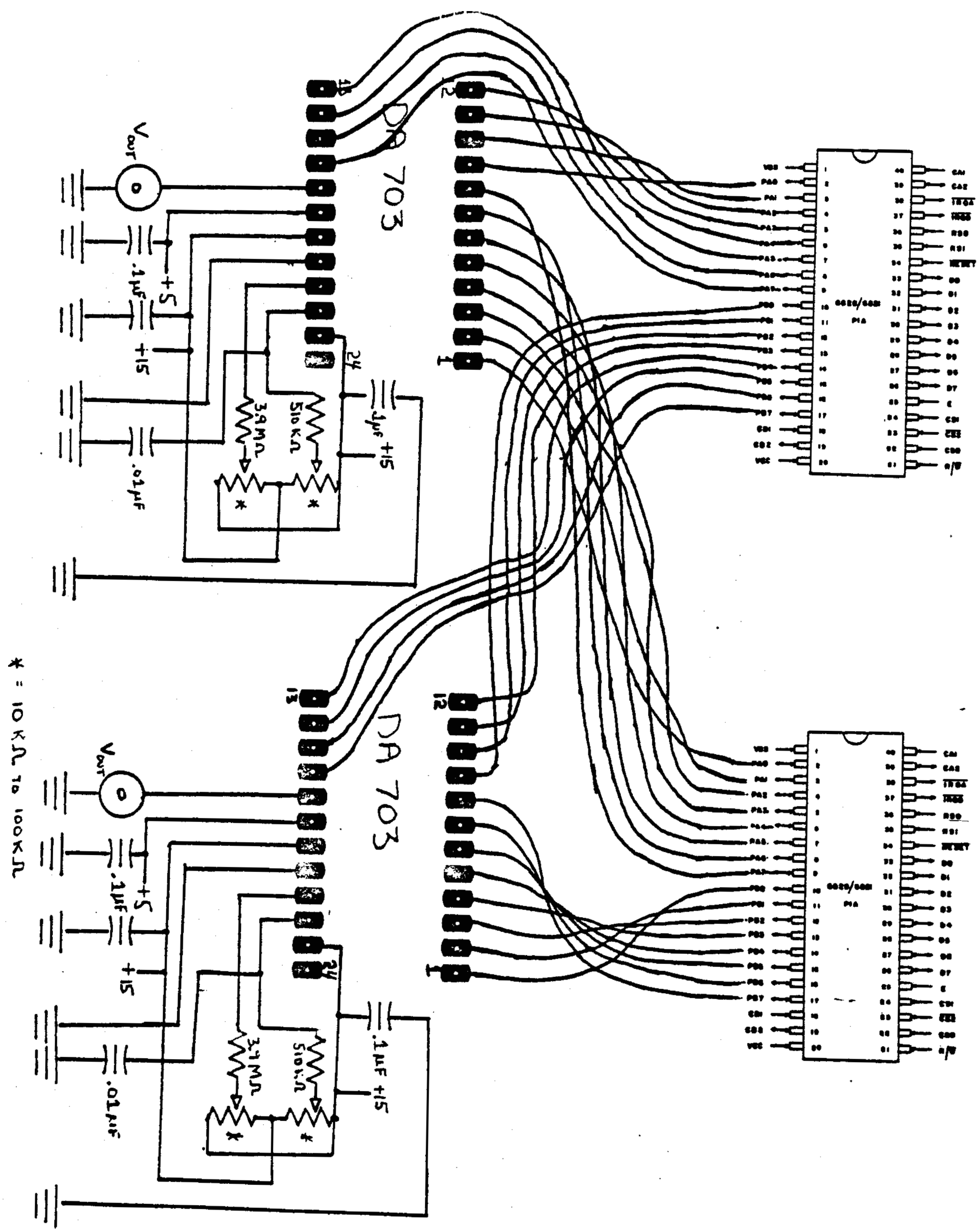


Figure 6. Schematic for the DAC703 D/A Interface board..

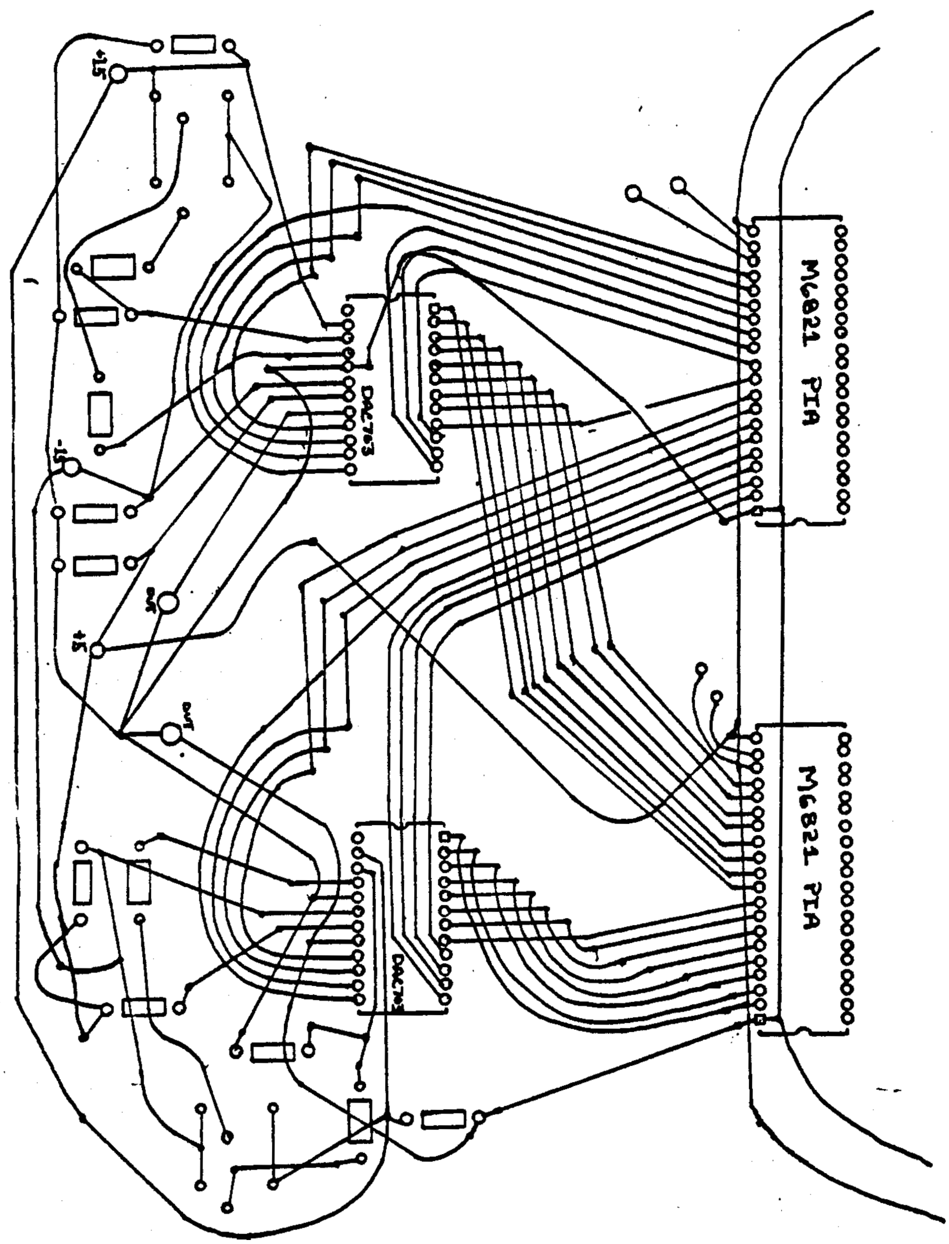


Figure 7. Artwork for the DAC703 D/A interface board.

EXPERIMENTAL

The data acquisition system has been put together and debugged. The testing of this system was performed using a Wavetek Model 164 signal generator. The digital-to-analog circuitry was tested using a Tektronics oscilloscope, model 564 B.

The Wavetek generator was used to generate a sawtooth wave which was recorded by the data acquisition system, processed and plotted using a Houston Instruments Model 200 x-y recorder.

A ramp wave was then generated using the AM 9513 realtime clock via program. and was monitored using the Tektronics scope through the use of the digital-to-analog circuitry.

The following pages show a sample program used for the data acquisition and plots which were recorded on the x-y recorder.

Sample Program for Data Acquistion

```
CODE   INITIALIZEPIA  TD CLEAR
                      SM 1FE14 OOCC
                      SM 1FE15 OOCC
                      SM 1FE16 OOCC
                      SM 1FE10 0011
                      SM 1FE11 0011
                      SM 1FE12 0011
                      SM IFE13 0013

O VARIABLE DATA 1000 ALLOT
O VARIABLE INPUT
CODE ACQUISITION 1FE14
: GRTDATA  1000 O DO
           EXPECT ACQUISITION
           DATA @   DATA 1 + @ + INPUT LOOP ;
: AVERAGE
           0
          1001 I DO
                   I DATA @ +
                 LOOP
          1000 / 2 *
           CONSTANT HIGHVALUE ;
: DATA@  1- 2 * DATA + @ ;
: CHECKNOISE
           CREATE DUP ,
```

```
                2 * ALLOT
          DOES>
                1000 1 DO
                I DATA@
                HIGHVALUE> IF
                           ELSE
                               LOOP
                           THEN
                               SWAP 1- SWAP
                               OVER OVER
                               LOOP;


HEX FOOO CONSTANT VIDEO
DECIMAL 64 CONSTANT WIDTH
        16 CONSTANT HXIGHT
: COORD   WIDTH * + VIDEO + ;
: PLOT    COORD C! ;
WIDTH 1- CONSTANT XMAX
HEIGHT 1- CONSTANT YMAX
VARIABLE X VARIABLE Y
: XYCHECK    X@ O MAX X!
             X@ XMAX MIN X!
             Y@ O MAX Y!
             Y@ YMAX MIN Y! ;
: XYPLOT   1000 1 DO
                  I DATA@
                  XYCHECK X @ Y I PLOT ;
```

The preceding program allows for the acyuisition of 1000 data points, processing, and graphics representation of those points.

The first section of the program allows the computer to jump from FORTH into its built-in programming by using the FORTH code type definition. "INTIALIZEPIA" is defined by the following programming lines which initialize the PIA's.

The next section of the program sets up an array of 1000 words. This is followed by the data acquisition programming. In the data acyuisition programming, the code type definition is again used to allow for proper addressing of the expected type of data input.

The data are averaged in the next program. The average is then used in the following section of programming to check for noise caused by the instrumentation. For example: if the data input ranges from +0 to +2 volts, then on a wave type function the average input would be +one or some value which corresponds to +one. If the data recieved are more than twice +one, or twice some corresponding value for +one, then it will be thrown out by the programm. In this manner only data in a reasonable range or scale will be recorded.

The final part of the program deals with the graphics and plotting. This is a simple graphics program which sets values within coordinate parameters and plots them on the screen or display using the pixels built into the computer.

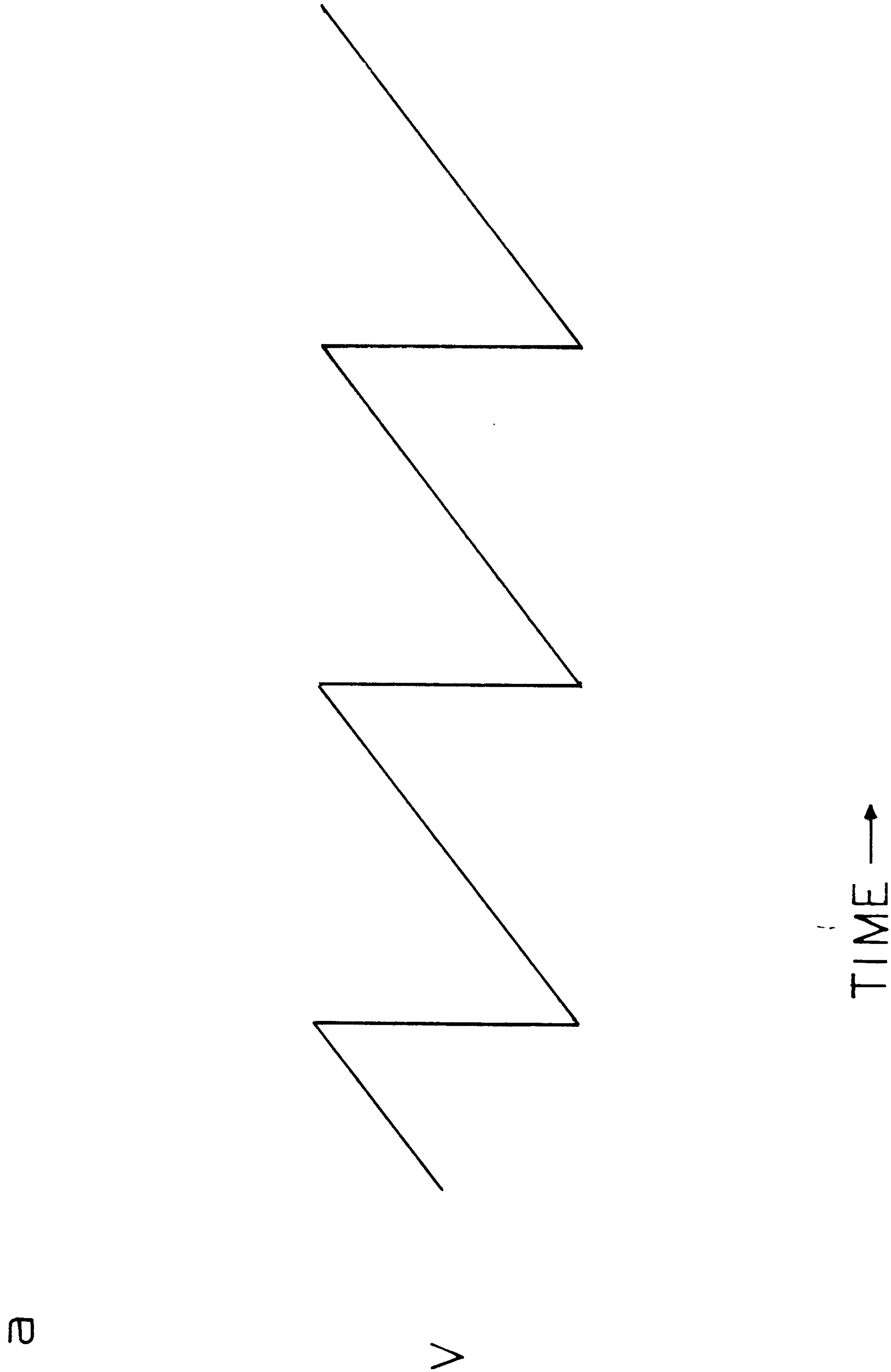
typing in the emit command.

The "unconventionality" of FORTH programming is demonstrated by this prgram, but the ease is also demons-trated. A program to carry out the same process in FORTRAN or BASIC would become much more involved and lengthy.

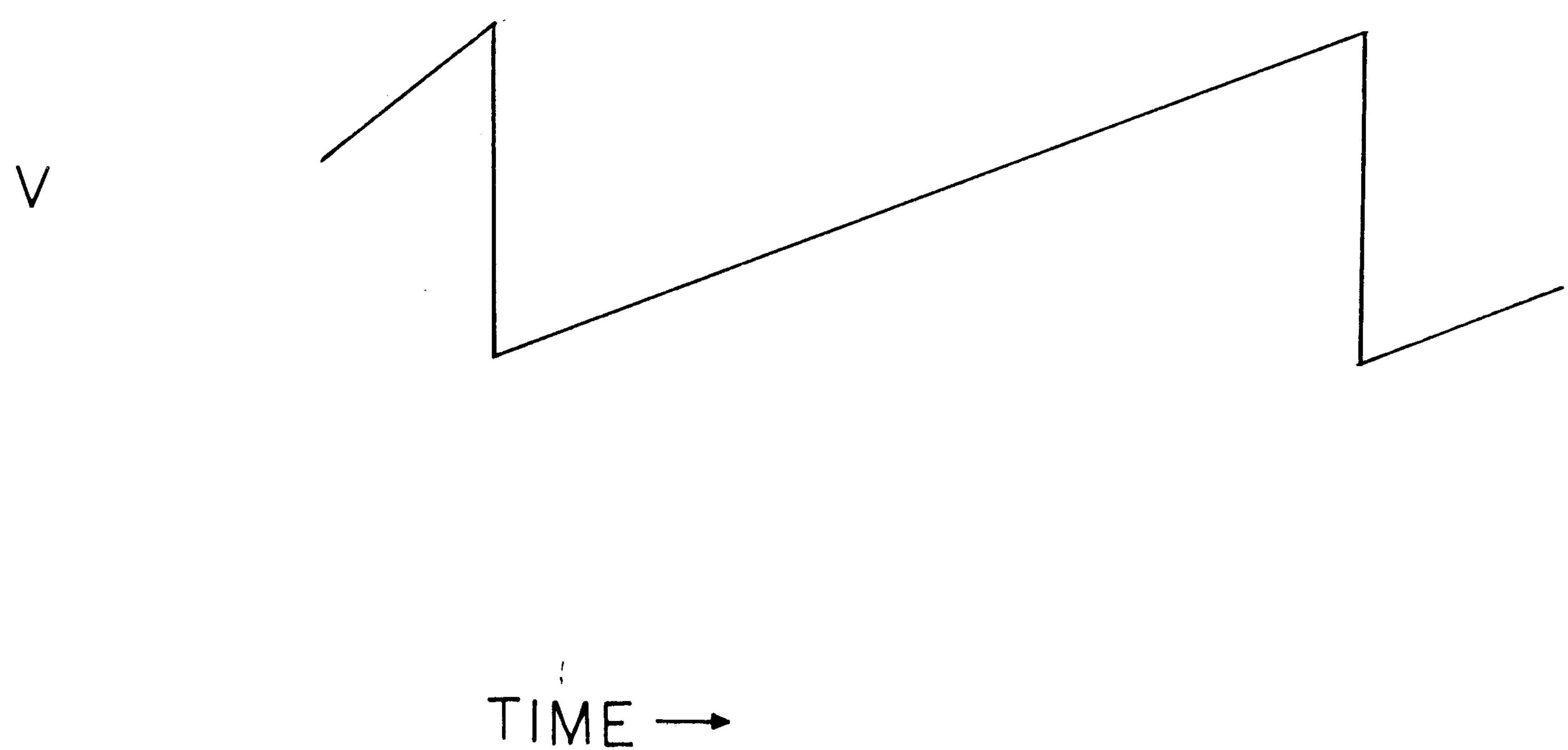
The following two plots are of wave functions which were plotted via the same graphics program using the Houston Instruments Model 200 x-y recorder.

Figure 8 is a plot of the data acquisition of a sawtooth wave generated at 4KHz by the Wavetek Model 164 signal generator.

Figure 9 is a plot of the same wave function after a ramp of 50-mV steps was applied to the signal and output through the digital-to-analog circuitry monitored by the Tektronics scope.


TIME
V
a

B



Conclusion

This data acquisition system has shown potential to be employed as a control and acquisition device for electrochemical experimentation. The versatility of this system is in a sense only limited by the creative and programming abilities of the researcher.

The memory, speed, and programming abilities of this system allow for high quality performance, superior to the limited capabilites of microcomputer systems built for one specific function.

The large memory capacity allows for data storage and manipulation not possible in most systems.

The major work in design, building and programming of this system has been carried out. It is hoped that work on development and use of this system will be continued in the future.

REFERENCES

1. Malmstadt, H.V., Enke, C.G., and Crouch, S.R. 1981. Electronics and Instrumentation for Scientists. Menlo Park: Benjamin/Cummings Publishing Co.

2. Boole, G. 1854. An Investigation of the Laws of Thought on Which Are Pounded the Mathmatical Theories of Logic and Probabilities.

3. Wiatrowski, C.A., and House, C.H. 1980. Logic Circuits and Microcomputer Systems, New York: Mcgraw Hill, Inc.

4. Mcdonell, R.G. 1984. Moving computer-aided chemistry into 1984. American Laboratory 16, no.2:71-77.

5. Plambeck, J.A. 1982. Electroanalytical Chemistry: Basic Principles and Applications. New York: John Wiley & Sons.

6. Kudick, M.D. 1983. Assembly Language Programming. Wm. C. Brown Co.

7. Dessey, R. 1983. Languages for the Laboratory: Part I. Analytical Chemistry. 55, no.6:650A-662A.

8. Clayton, G.B. 1982. Data Converters. New York: John Wiley & Sons.

9. The Am9513. 1983. M68K Users Manual. Irvine, Ca.: Educational Microcomputer systems.

10. Scale, H. 1982. The 68000 CPU. Computers & Electronics November 86-93.

11. Salin,E. 1982. Laboratory computer acquisition considerations: Part two: Language Selection. American Laboratory. March: 35-38.

12. Coombs,C.F. 1979. Printed Circuits Handbook. New York: McGraw-Hill Book Co.

13. Kane,G., Hawkins,D. and Leventhal,L. 1981. 68000 Assembly Language Programming. Berkeley,Ca.: Osborne/Mcgraw-Hill Book Co.

15. Dessey,R. and Hooley,D. "FORTH, A Fourth Generation Language Implements A Network," in Computer Networks in the Chemical Laboratory. 1981. Ed. Levy,G.C. and D. Terpstra. New York: John Wiley & Sons, Inc.

16. A. Taylor. 1979. The Taylor Report: FORTH Becoming a Hothouse for Developing Languages. Computerworld.