11-3-1978

The Development of a Low Cost Microcomputer

Jerry G. Farrell
University of Northern Iowa

The Development of a Low Cost Microcomputer

Abstract

It is the purpose of this study to: (1) review all available literature on the microprocessors; (2) study the function and application of the microcomputer; (3) become familiar with the programming techniques used in the microcomputer; (4) design and construct a microcomputer system; (5) develop an instructional unit around the functions and application of the microcomputer; and prepare a written report on the microcomputer system.

DEPARTMENT OF
INDUSTRIAL TECHNOLOGY
University of Northern Iowa
Cedar Falls, Iowa 50614-0178

The Development of a Low Cost Microcomputer

A Research Paper for Presentation
to the Graduate Committee
of the Department of Industrial Arts
and Technology
University of Northern Iowa

In Partial Fulfillment of the Requirements for
the Non-Thesis Master of Arts Degree

by
Jerry G. Farrell
November 3, 1978

Approved by:

Technical Advisor — Date: Dec. 6 1978

Graduate Committee, Chairman — Date: 12/9/1978

TABLE OF CONTENTS

LIST OF FIGURES

Chapter I

INTRODUCTION

The Problem

Recent advances in large scale integration (LSI) technology have made it possible to place a digital computer central processing unit on a single integrated circuit chip. This advancement is destined to have as much impact on all areas of technology as did the development of the integrated circuit itself. With the introduction of the microprocessor, as it is called, it is now possible to utilize digital computer technology in a wide variety of applications ranging from home entertainment equipment to control systems. As in any new technology, it is first necessary to understand the basic microprocessor in order to exploit it adequately and get all the benefits from it. This study is concerned with the design of an microcomputer system and the determination of principles, content, techniques, and application of this system. The researcher hopes to use this information effectively in industrial applications or instructing students in electronic technology. (Oliver, 1977, P.180)

Purpose of the Study

It is the purpose of this study to: (1) review all available literature on the microprocessors; (2) study the function and application of the microcomputer; (3) become familiar with the programming techniques used in the microcomputer; (4) design and construct a microcomputer system; (5) develop an instructional unit around the functions and application of the microcomputer; and prepare a written report on the microcomputer system.

Importance of the Study

Since the transistor was invented, no single electronic innovation has made such an impact as the microprocessor. Powered by tiny simiconductor chips containing computing elements with the same powers and functions previously found only in large scale digital computers. These dedicated microprocessor systems are now being applied to literally thousands of applications. Microprocessors are automating assembly lines, providing the heart of sophisticated electronic games, and making "intelligent" computer peripherals even smarter. This revolution is occuring because microprocessors are very inexpensive - costing as little as $30.00 in production volume - are easy to implement into a system, and they significantly reduce the time and cost of product development. (Noyce, 1977, P.62)

But there has been one serious drawback to this exploding industry. Training materials and study courses in the basic microprocessor technology have been virtually non-existant and the various published manuals have been undecipherable to those not familiar with sophisticated logic design.

Limitations of the Study

This study will be made from library research related to microprocessors and from written material supplied by the manufacturers of these devices. The project activity will be limited to design, construct, and test of a microcomputer system.

Definition of Terms

Accumulator: The register within a computer where the results of all arithmetic and logical operations are placed.

Address bus: A unidirectional bus over which digital information appears to identify either a particular memory location or a particular I/O device.

Central processing unit (microcomputer): A single integrated circuit chip that performs data transfer, control, input-output, arithmetic, and logical operations by executing instructions obtained from memory.

Computer program: A sequence of instructions which, taken as a group, allow the computer to accomplish a desired task.

Control: Those parts of a computer which carry out instructions in proper sequence, interpret instructions, and apply proper signals.

Control bus: A set of signals that regulate the operation of a microcomputer system, including I/O devices and memory. They functions much like "traffic" signals or commands. They may also originate in the I/O devices, generally to transfer to or receive signals from the CPU. According to the Intel Corporation literature, a control bus is a unidirectional set of signals that indicate the type of activity - memory read, memory write, I/O read, I/O write, or interrupt acknowledge - in current process. (Sippl, 1974, Pg. 1-5)

CPU: Abbreviation for central processing unit.

Device code: For an 8080A microcomputer, the specific input or output device, coded in eight-bit binary, with which eight bits of information and a device select pulse are exchanged.

Digital computer: An electronic device that is capable of accepting, storing, and arithmetically manipulating information, which includes both data and the controlling program. The information is handled in the form of coded binary digital (0 and 1) that are represented by dual voltage levels. (Sippl, 1974, Pg. 17)

Disable: To prevent the passage of digital signals by the application of the proper signal to the disable terminal of a digital device.

Dynamic RAM: A random access memory that uses a capacitive element for storing a data bit.

Encode: To use a code, frequently one composed of binary numbers, to represent individual characters or groups of characters in a message. To change from one digital code to another. If the codes are greatly eifferent, the process is called code conversion.

Hardware: The mechanical, magnetic, electronic, and electrical devices from which a computer is fabricated; the assembly of material forming a computer. (Sippl, 1974, Pg. 22)

Inhibit: To prevent an action from taking place or digital data from being accepted or transmitted. This is accomplished through the application of the appropriate signal to the proper input of a digital device.

I/O device: Input/output device. A card reader, magnetic tape unit, printer, or similar device that transmits data to or receives data from a computer or secondary storage device. In a more general sense, any digital device, including a single integrated circuit chip, that transmits data to or receives data or strobe pulses from a computer. (Rony, 1974, Pg. 20)

Instruction: A set of characters that define an operation, alone or together with other information, and which, as a unit, causes a computer to perform the operation.

Instruction cycle: For an 8080A microcomputer, a successive group of machine cycles, as few as one or as many as five, which together form a single microprocessor instruction. (Rony, 1974, Pg. 18-1)

Interrupt: In a digital computer, a break in the normal execution of a computer program such that the program can be resumed from that point at a later time.

Language: The whole body of words and of methods of combination of words used by a nation, people, or race.

Machine code: A binary representation of a computer instruction.

Magnetic core storage: A type of computer storage which employs a core of magnetic materials with wires threaded through it. The core can be magnetized to represent a binary 1 or 0.

Magnetic drum: A storage device consisting of a rapidly rotating cylinder, the surface of which can be easily magnetized and which will retain the data. Information is stored in the form of magnetized spots (or no spots) on the drum surface.

Magnetic disc: A flat circular plate with a magnetic surface on which data can be stored by selective magnetization of portions of the flat surface.

Magnetic tape storage: A storage system based on the use of magnetic spots (bits) on metal or coated-plastic tape. The spots are arranged so that the desired code is read out as the tape travels past the read-write head.

Memory: Any device that can store logic 1 and logic 0 bits in such a manner that a single bit or group of bits can be accessed and retrieved.

Memory address: The storage location of a memory word.

Volatile Memory: A semiconductor memory device in which the stored digital data is lost when the power is removed.

Non-Volatile Memory: A semiconductor memory device in which the stored digital data is not lost when the power is removed.

Microcomputer: A fully operational digital computer that is based upon a microprocessor chip family.

Microprogram: A computer program written in the most basic instructions or subcommands that can be executed by the computer. Frequently, it is stored in a read-only memory. (Microdata, 1977, Pg. 62)

Mnemonic code: Computer instructions written in a form that programmer can easily remember, but which must be converted into machine code later by a computer or by the user.

Mnemonic language: A programming language that is based upon easily remembered symbols and that can be assembled into machine code by a computer.

Octal code: Pertaining to a binary-coded numbering system with the radix 8, in which the natural binary values of 0 through 7 are used to represent octal digits with values from 0 to 7.

Operation: A specific action which a computer will perform whenever an instruction calls for it (e.g., division, addition, subtraction, OR, AND, etc.).

Parallel: This refers to the technique for handling a binary data word that has more than one bit. All bits are acted upon simultaneously. It is like the line of a football team. Upon signal, all line men act. Many wires are required for parallel data transmission.

Programmable ROM: Abbreviated PROM or pROM. A read-only memory that is field programmable by the user.

Random access memory: Abbreviated RAM. A semiconductor memory into which logic 0 and logic 1 states can be written (stored) and then read out again (retrieved).

Read/write memory: A semiconductor memory into which logic 0 and logic 1 states can be written and read out again.

Read-only memory: A semiconductor memory from which digital data can berepeatedly read out, but cannot be written into as in the case for read/write memory.

Register: A short-term digital electronic storage circuit the capacity of which is usually one computer word or byte.

Serial: This refers to the technique for handling a binary data word which has more than one bit. The bits are acted up one at a time. It is like a parade going by a review point. Only a single pair of wires is required for serial data transmission.

Stack pointer: A register which comes into use when the microprocessor must service an interrupt - a high-priority call from an external device for the central processing unit to suspend temporarily its current operations and divert its attention to the interrupting task.

Software: The totality of programs and routines used to extend the capabilities of computers, such as compilers, assemblers, narrators, routines, and subroutines. Contrasted with hardware.

Static RAM: A random access memory that uses a flip-flop for storing a data bit.

Synchronism: Counters can be either asynchronous (or ripple) or synchronous. In an asynchronous counter, a change in the output of one flip-flop initiates a possible change in the next flip-flop, which in turn initiates a change in the following flip-flop, and so on.

Word: The number of bits that a computer can manipulate simutaneously.

Write: In semiconductors: To transmit data into a semi-conductor memory from some other digital electronic device or integrated circuit chip.

Chapter II

HISTORY OF MICROCOMPUTERS

In the early 1960's the commercially available integrated circuits incorporated at most a score of components such as diodes, transistors and resistors. The fraction of circuits that worked were low, and packaging technology did not allow the realization of practical devices with more than a dozen leads, or connections. The basic technology, however, was so emenable to improvment and the rivalry among manufacturers was so keen that every year since then the number of components that could be economically placed on a single chip has doubled. Today chips less than a quarter of an inch on an edge can incorporate well over 20,000 components. As a result the cost per component has in 10 years dropped by a factor of more than 100% from about 2 cents to a small fraction of a cent.

The steady increase in component density, combined with parallel advances in circuit organization and complexity, has predictably led to the microcomputer, a full-fledged general-purpose machine whose logic and memory circuits can be mounted on a single plastic card that would fit comfortably inside a cigar box. Where space is at a special premium the complete microcomputer can be squeezed into a substrate two inches square. (Oliver, 1977, P.180)

The microcomputer is a direct descendant of the minicomputer, whose first embodiments were the PDP-5 and PDP-8, small parallel data processors introduced in 1963 and 1965 by the Digital Equipment Corporation.

The PDP-8, roughly the size of two-drawer legal file cabinet, became known as a minicomputer primarily because of its physical size, not because of limitations in its performance. Nearly as powerful as much larger computers costing several times more, it was soon widely imitated. Within a decade it has given rise to an entire industry concerned not only with "hardware" (the computers themselves) but also with "software" (the programs) and with innumerable peripheral devices and auxiliary services.

Minicomputers rapidly found their way into existing systems. To them they brought, in addition to cost reductions, the flexibility and simplified design of stored-program machines. More important, they made possible a wide range of new applications that called for an inexpensive resident computer.

Much of the success of the minicomputer industry was due to the dramatic advances being made in microelectronics throughout the 1960's. These advances had a twofold effect. The performance of minicomputers improved steadily, and the size and cost of the systems decreased at a rate of averaging 30 percent per year. (Altman, 1975, Pg. 2-10)

It was clearly just a question of time until the further integration of microscopic components would lead to a microcomputer, a machine that would have tens of thousands of components on a single chip, or at most a few chips, and that would require no more than a few hundred milliwatts of power.

By the late 1960's it was still difficult to predict when single-chip computers would become a reality because semiconductor manufacturers had been concentrating on the development of what are termed bipolar devices. In such devices the currents in the transistors are carried by both electrons and "holes" (a hole is a site in a semi-conductor where an electron is absent, so that whereas the electron is a negative current carrier the hole is a positive current carrier). With bipolar devices high component densities were difficult to achieve, and substrate sizes were limited because of low production yields and problems with the dissipation of heat. At about this time however, older fabrication problems that had plagued the metal-oxide-semi-conductor (MOS) technology were solved, and it became possible to economically manufacture large unipolar devices with high component densities and low head production. Initially these devices were exclusively of the P-channel type (PMOS); their transistors operated with positive current carriers (holes). Although PMOS devices were an order of magniture slower than bipolar devices, they have higher component densities, lower power requirements and better production yields that make them ideally suited for applications such as electronic calculators and computer memories.

The PMOS technology contributed to a major conceptual advance in 1971, when the Intel Corporation, which had undertaken to develop a calculator chip, chose to design it as a more versatile, programmable, single-chip microprocessor. A microprocessor is equivalent to the central processing unit of a larger computer. Known as the Intel 4004, the device processes blocks consisting of four bits, or binary digits, at a time. (Intel, 1975, Pg. i)

It has 2,250 transistors on a silicon substrate measuring .117 by .159 inch, and it comes in a package with 16 pins, or leads. All that is needed to make the microprocessor into a minimal general-purpose microcomputer that could sell for less than $50 are two additional devices a memory (a control memory and a temporary storage memory) and a master clock to time the system's operation.

In December 1971, Intel introduced the first general purpose, 8-bit microprocessor, the 8008. It was implemented in P-channel MOS technology and was packaged in a single 18 pin, dual in-line package (DIP). The 8008 used standard semiconductor ROM and RAM and, for the most part, TTL components for I/O and general interface. It immediately found applications in byte-oriented and products such as terminals and computer peripherals where its instruction execution (20 micro-seconds), general purpose organization and instruction set matched the requirements of these products. Recognizing that hardware was but a small part in the overall system picture, Intel developed both hardware and software tools for the design engineer so that the transition from prototype to production would be as simple and fast as possible. The commitment of providing a total systems approach with the 8008 microcomputer system was actually the basis for the sophisticated, comprehensive development tools that Intel has available today. (Intel, 1975, Pg. 8)

With the advent of high-production N-channel RAM memories and 40 pin DIP packaging, Intel designed the 8080 microprocessor. It was designed to be software compatible with the 8008 so that the existing users of the 8008 could preserve their investment in software and at the same time provide dramatically increased performance (2 microsecond instruction execution), while reducing the amount of components necessary to implement a system. Additions were made to the basic instruction set to take advantage of this increased performance and large system-type features were included on-chip such as DMA, 16 bit addressing and external stack memory so that the total spectrum of application could be significantly increased. The 8080 was first sampled in December 1973. Since that time it has become the standard of the industry and is accepted as the primary building block for more microcomputer based applications than all other microcomputer systems combined. (Intel 1975, Pg. i)

The two Intel microprocessors and two other early PMOS models (Made by Rockwell International and National Semiconductor) were quickly incorporated into a wide range of applications, from specialized laboratory instruments to sales terminals and electronic games. As more engineers become aware of the microprocessor technology, other new uses evolved rapidly. At the same time more semiconductor manufacturers sought to establish a foothold in the rapidly expanding microprocessor market.

Although there have been a few failures, more than 20 different microprocessors have appeared within the five years since the introduction of the Intel 4004. Most of the Microprocessors still use the PMOS technology, which has been developed to such a level that one 12-bit microprocessor, made in Japan by Toshiba, has 11,000 transistors on a single chip .22 by .24 inch. Most of the newer microprocessors, however, use either the faster NMOS technology, in which the transistors operate by means of negative current carries (electrons), or the complementary MOS (CMOS) technology. CMOS combines the PMOS and NMOS technologies to achieve a reduction in power requirements and to improve resistance to extraneous noise. The first bipolar microprocessors began to appear late last year.

Coming three decades after the first electronic computers, microprocessors benefited from the experience accumulated in system organization and computer architecture. Many advanced concepts and features, frequently unavailable in machines several order of magnitude larger and more expensive, are standard in almost every microprocessor.

Four trends are emerging, first established families are being enhanced. The system throughout is being increased and instruction sets enlarged as manufacturers turn to new metal-oxide-semiconductor processing and improved central-processor architecture. Input/output power, too, is being increased with new sets of programmable I/O chips.

Second, the new 16-bit single-chip processing units are heading upwards. What they are aiming for is the high-performance end of the microprocessor market, where precision arithmetic and large memories must be accommodiated.

Third, the one-chip controllers, as their name implies, contain enough computing power to handle many stand-along controller fucntions on their own. On the same chip as the central processing unit and its control is read only memory for program storage, random-access memory for data storage, and input/output registers for system manipulation.

Finally, there's a host of single-board microcomputers, beguiling alternatives to the do-it-yourself approach of buying just the chips. (Torrero, 1975, Pg. 1-4)

Chapter III

TECHNICAL INFORMATION

What is a Microprocessor

There is a distinction between a microprocessor and microcomputer. To some people, Microprocessor means microcomputer, to others the words microprocessor and microcomputer are not the same thing. Microprocessor to them is a broader and more generic term which describes an extremely small electronic system capable of performing specific tasks. Thus, microcomputer is an application of microprocessors.

A Microcomputer is fully operational digital computer that is based upon a microprocessor chip. A microprocessor is a single integrated circuit chip that possesses at least 75% of the computing and data manipulation power of a digital computer. In digital electronics, the term chiefly applies to circuits containing semiconductor elements. The microprocessor chip is a product of advanced technology in the semiconductor industry, basically the capability that manufacturers now have to fabricate thousands of transistors on a single silicone chip no longer than 0.05 square inches.

This researcher considers a microprocessor to be a single integrated circuit chip that contains approximately 75% of the power of a very small digital computer. It usually must have the aid of support chips and memory. In contrast, a microcomputer is a full operational computer system based upon a microprocessor chip. Such a system contains memory, latches, counters, input/output devices, buffers, and power supply in addition to the microprocessor chip. (Lewis, 1973, Pg. 106-110)

There may be as much cost involved in the other hardward components as there is in the microprocessor chip itself.

The researcher would also like to quote from the article by Laurence Altman in the April 18, 1974 issue of Electronics:

> What a microprocessor is ... but first, what it isn't. A microprocessor is not a computer but only part of one. To make a computer out of a microprocessor requires the addition of memory for its control program, plus input and output circuits to operate peripheral equipment. Also, the word is not short for microprogrammable central processing unit. For though some microprocessors are controlled by a microprogram, most are not.
>
> What a microprocessor is, then, is the control and processing portion of a small computer or microcomputer. Moreover, it has come to mean the kind of processor that can be built with LSI MOS circuitry, usually on one chip. Like all computer processors, microprocessors can handle both arithmetic and logic data in bit-parallel fashion under control of a program. But they are distinguished both from a minicomputer processor by their use of LSI with its lower power and costs, and from other LSI devices (except calculator chips) by their programmable behavior.
>
> In short, if a minicomputer is a 1-hoursepower unit, the microprocessor plus supporting circuitry is a ¼-hp unit. However, as LSI technology improves, it will become more powerful.

Architecture

The everyday meaning of architecture, referring to both a style of construction and a particular way of assembling structural materials to achieve a functional goal, carries over into the field of computer design. Computer architecture describes the arrangement of the central processing unit (CPU), the memory elements for the storage of programs and data, the input and output devices and the master clock. Thus, one architecture may emphasize facility of arithmetic operations and another may stress convenience of input and output operations. Whereas both have a CPU, a memory and input and output parts, the first is more suited for lengthy numerical analysis and the second is more suited for control applications and the monitoring of external equipment.

The CPU, or microprocessor, is the most expensive component of a microcomputer. The microprocessor provides the circuit designer with a stored program digital computer capability that can be used in his applications.

To understand the basic microprocessor, let's first examine the organization of the digital computer. Figure 1(Pg. 16)shows the main sections of a typical stored program digital computer. Instructions and data are stored in memory. The control unit decodes the instructions, one at a time, and through the arithmetric/logic unit performs the specified operations. The instruction program may be computing a pay-roll in a business application, or guiding a milling machine in a control application. The input and output units in the computer provide a means of entering input data and instructions into the system and receiving output data from the system. The major advantage of a stored program digital computer is that the function it performs is determined by the instructions stored in memory. The same computer can be used to compute wages or control a milling machine -- the system designer merely writes the program instructions to accomplish the desired task. (Intel, 1975, Pg. 2-3)

Present day microprocessors have the equivalent of the digital computer's control unit and arithmetric unit, timing, and control sections. These units serve the same functions as described for the general purpose system of Figure 1.(Pg. 16)

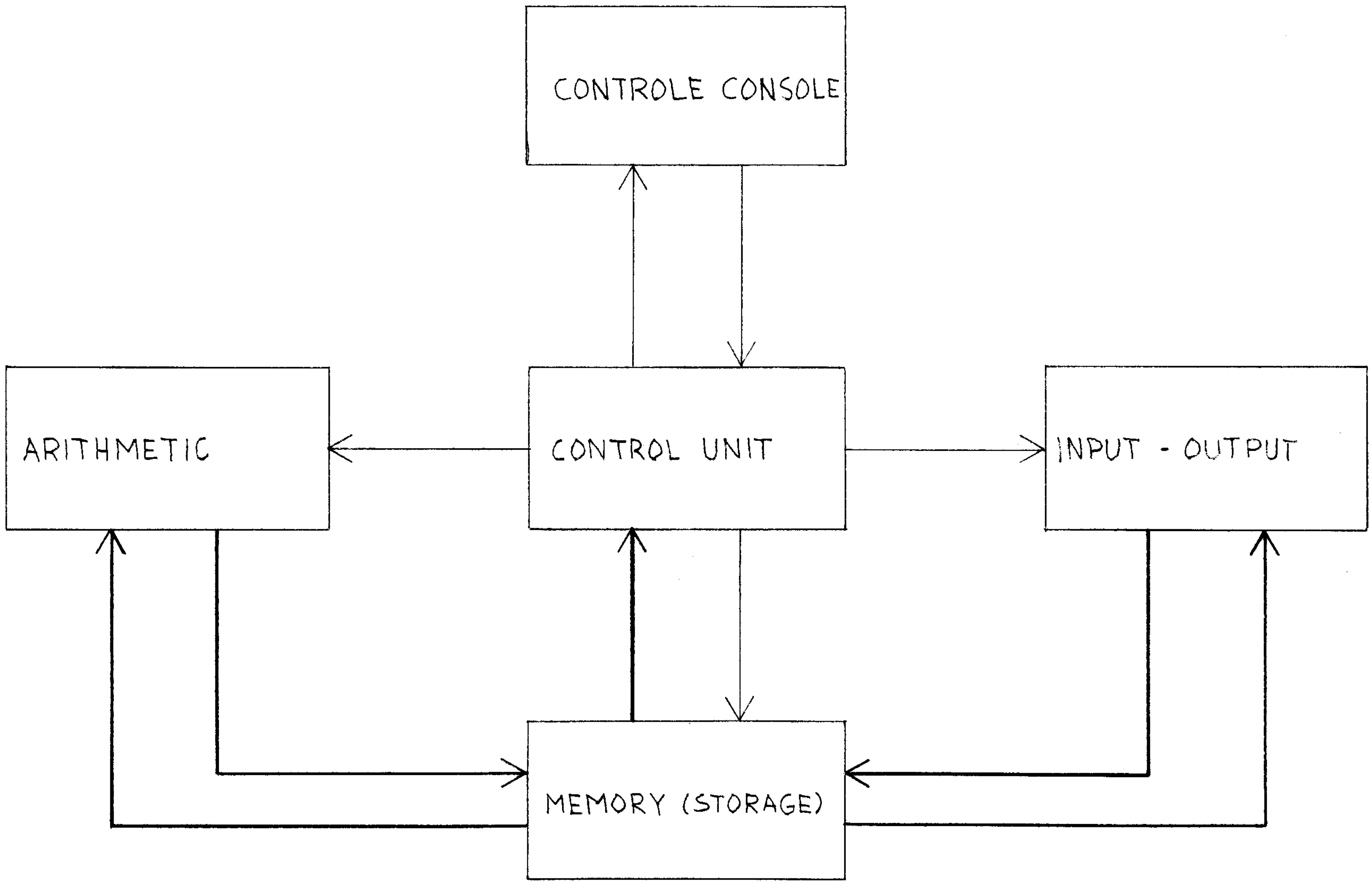


Figure 1. Digital Computer

The simplified computer diagram of Figure 2(Pg.18)indicates separate connections between the memory, input, output, control, and arithmetic/ logic units. There is a limit to the number of pins or terminations that can be used on a integrated circuit. As a result, the concept of a bi-directional data buss is used for data input and output, as well as for internal data flow on the chip. In the microprocessor, instructions and data (both input and output) are communicated over a common data buss. The microprocessor generates a number of signals which control external gating circuits to connect the input, output, and memory units to the processor chip at the correct time in the processing cycle. The use of a bi-directional data buss is economical in terms of the circuitry required on the chip and is one of the techniques that made the microprocessor possible.

Large scale integration has also revolutionized data storage. Chips capable of storing a thousand data bits are readily available. The LSI memory chips can be ganged together to provide the necessary memory capability. Thus, the microprocessor chip with a handful of memory and miscellaneous logic chips becomes a complete stored program digital computer.

Microcomputers are usually classified according to the number of bits that can be handled by their CPU. Their performance is judged by the richness of their instruction set, by the bit efficiency of their program and by the speed with which they execute typical programs. Such distinctions have mainly to do with their capacity for operating the "real time." If speed of operation is not a consideration, almost any Microcomputer can serve in a given application. Some systems,

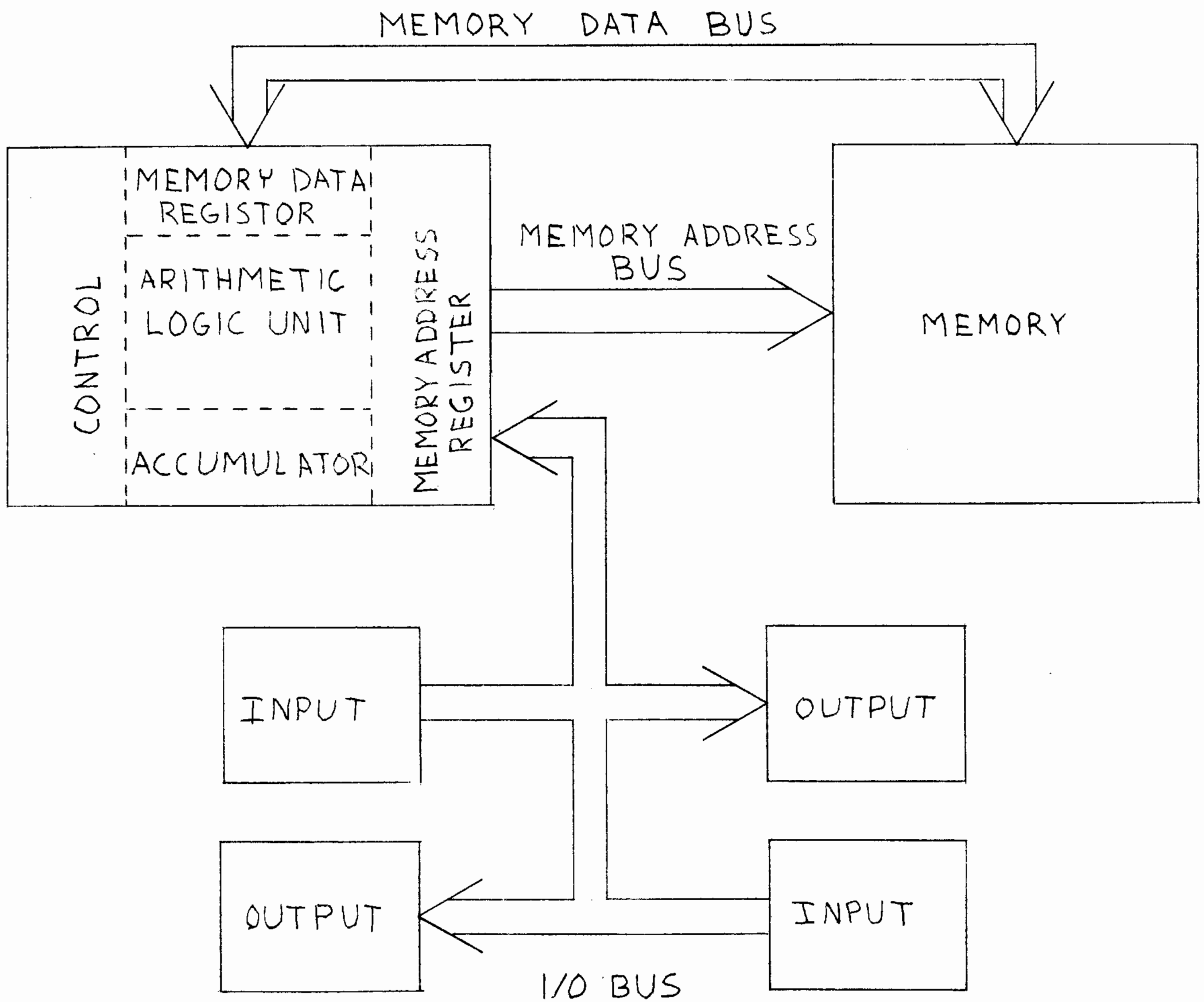


Figure 2. Single Chip Microprocessor

however, may be more economical than others for particular jobs.

Four-bit single-chip microprocessors such as the Intel 4004 or the PPS-4 made by Rockwell International are particularly economical. They are well suited to systems designed for decimal arithmetic, or to systems that do not have to deal with "words" consisting of many bits. Four-bit microprocessors are the most natural choice for products such as electronic scales (which are now common in supermarkets) and sales terminals. For such applications a microcomputer control system consisting of three or four chips costing less than $50 can be placed on a printed-circuit board between four and six inches square.

Eight-bit microprocessors are the most popular at the present time. Their word length makes them a natural choice for all applications that involve communications equipment, which commonly works with eight-bit encoded characters. They have more complete instruction sets and more computing power than the four-bit units and have many of the features found in larger machines. Although many of them require more supporting devices, some of the most recent eight-bit CPU's can make a complete minimal microcomputer with as few as five devices. A few 12 and 16 bit machines have been introduced, mostly for process control and other complex tasks. Some of them are highly integrated versions of previously available minicomputers, for which they are economical substitutes when speed is not critical.

Memory

A memory is any device that can store logic 1 and logic 0 bits in such a manner that a single bit or group of bits can be accessed and retrieved.

Large amounts of memory are required to store computer programs, data, special computer languages, operating procedures for the computer, and temporary information generated while making a computation. Some common computer memory devices include, punched paper card, magnetic tape, magnetic disc, magnetic drums, magnetic core, bubble memories, and semiconductor memories.

A semiconductor memory is a digital electronic memory device in which logic 0 and logic 1 bits are stored in the form of electronic signals in a semiconductor matrix. Semiconductor memories are known by various names, including random access memories, ready-only memories, scratch pad memories, shift registers, holding registers, and memory buffers. Most semiconductor memories found in microcomputers are of the "Random-Access Memory" (RAM) and "read-only Memory" (ROM).

A read-only memory, (ROM) is a semiconductor memory from which digital data can be repeatedly read out, but cannot be written into, as is the case from a random access memory. Once the desired data has been programmed into a ROM, there is in general no way that it can be altered short of destroying the chip.

The writer would like to draw a distinction between the basic ROM described in the previous sentences and an "erasable" read-only memory, which in fact can be repeatedly programmed with the aid of special erasing techniques such as the use of ultraviolet light. The important distinction is between the RAM, which loses the contants of its memory when power is removed, and the ROM, pROM, or erasable pROM, which doesn't. The ROM is a fixed memory or non-volatile memory. There are three types of ROM's:

The basic ROM is programmed by the manufacturer while the integrated circuit chip is being fabricated. Once programmed in a basic manufacturing step, there is no way to alter the memory.

The programmable ROM, (pROM), is programmed by the user at his own location using a special type of programming instrument that reads data into the memory of the ROM. Once the memory is programmed, it cannot be altered further.

The erasable ROM, is truly programmable, in that the memory can be re-programmed repeatedly. In one type of erasable pROM, the memory contents are erased through exposure to ultraviolet light. A more recent type of erasable pROM requires a single electrical pulse to erase the entire memory of data.

There are only four types of inputs or outputs, memory call select inputs, power inputs, memory outputs, and a single enable input, which, when at a logic 1, inhibits all of the outputs and sets them each at a logic 1 state.

An example of a ROM is the 74187 1024 bit, 256 words of 4 bits each, as shown in Figure 3(Pg.22) (Rony 1974, pg. 9-10 9-18) (Morris, 1971, pg. 125-149).

A random access memory, (ROM) is a semiconductor memory into which data, logic 0 and logic 1 states can be written and then read out again. Any location in the RAM must be accessible without regard to any other

1024-Bit Read Only Memories

General Description

These circuits are custom-programmed, 1024-bit read only memories organized as 256 words of four bits each. These high-speed TTL memory arrays are addressed in straight eight-bit binary, with full on-chip decoding. Two overriding memory-enable inputs are provided which, when either one or both are taken high, will inhibit the function causing all four outputs to remain high (off). Data, as specified by the customer, are permanently programmed into the 1024-bit locations. This organization is expandable to 41,472 words of n-bits, with no additional output buffering.

The address of a four-bit word is accomplished through the buffered binary select inputs, with low-level voltages at both enable inputs. The most significant binary select inputs, D through H, are decoded internally in the X plane to select one-of-32 lines, and the least significant bits, A, B, and C, are internally decoded in the Y plane to accomplish one-of-eight decoding to drive the four output buffers. Where multiple devices are used in a memory system, the enable input allows easy decoding of additional address bits.

Data are programmed into the memory cell at the emitters of a 32-by-32 matrix of transistors. In the X plane each of the 32 address decoding gate outputs supply common base drive to 32 transistors. In the Y plane the 32 transistors are arranged into four groups of eight. This permits each of the bit lines to be terminated in four one-of-eight decoders, which achieves the four-bit word length.

The open-collector outputs are capable of sinking 16 milliamperes of current and may be wire-AND connected to increase the number of words available. An external pull-up resistor is recommended for definition of the high (off) level output voltage.

The customer can specify the output logic level desired at each of the 1024 bit locations by completing the supplementary ordering data and a set of data cards, punched in accordance with the data format shown under ordering instructions. It is important that the customer specify not only the output levels desired at all 1024 bit locations, but also the other information requested.

WORD SELECTION

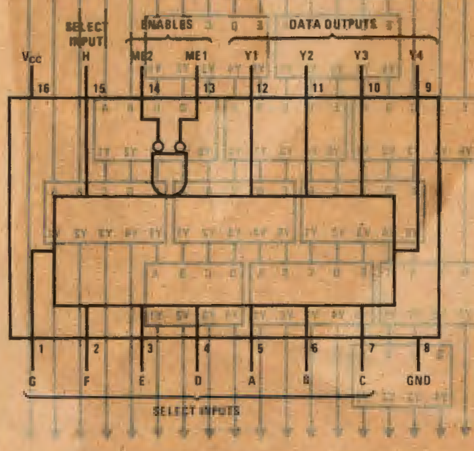
Word selection is accomplished in a conventional 8-bit positive-logic binary code with the A-select input being the least-significant bit progressing alphabetically through the select inputs to H which is the most-significant bit.

Features

- Organized as 256 words by 4 bits
- Ideal for microprogramming, reference tables and code converters
- Easily expandable
- Fully decoded, buffered inputs
- Diode-clamped inputs
- Full fan-out, open-collector outputs

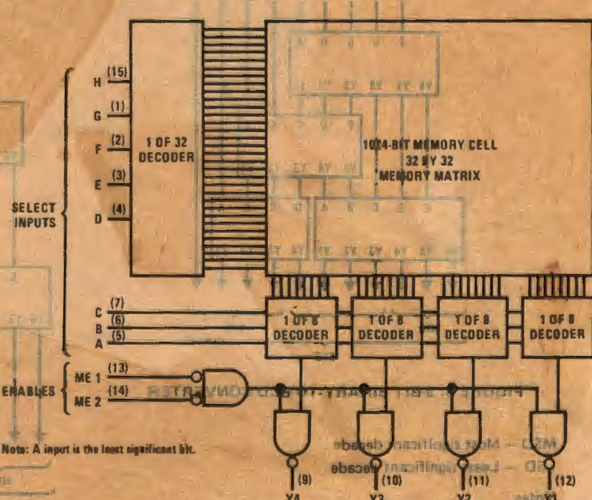
TYPE	TYPICAL ACCESS TIME	TYPICAL POWER DISSIPATION
187	37 ns	0.36 mW/Bit
L187A	90 ns	0.09 mW/Bit

Connection Diagram



54187(J); 74187(J), (N);
54L187A/74L187A(J), (N), (W)

Logic Diagram



Note: A input is the least significant bit.

Figure 3 (74187 1024 Bit, 256 words of 4 bits) location.

A random access memory, (RAM) is a semiconductor memory into which data, logic 0 and logic 1 states can be written and then read out again. Any location in the RAM must be accessible without regard to any other location.

The two classes of RAM's are static and dynamic. A static RAM stores each bit of information in a flip-flop, and this information is retained as long as power is supplied to the circuit. The static RAM is the main memory element for many microcomputers. Dynamic RAM's are devices in which the information is stored in the form of electronic charge on the gate-to-substrate capacitance of an MOS transistor. This charge dissipates in a few milliseconds, and the element must be refreshed periodically. Dynamic RAM's are important because fewer elements are required to store a bit, so that more bits can be packed into an IC of a given physical area. They are also faster than the static RAM and consume less power in the quiescent state. The refreshing cycle requires additional circuitry, however, which is often external. Therefore, there are a certain number of memory elements required before dynamic memory becomes profitable. Smaller memories are generally static elements, whereas larger memories are typically dynamic.(Rony, 1974, Pg. 21-1)

A typical RAM has the following inputs and outputs:

Memory cell select inputs. These inputs select the specific memory cell or group of memory cells desired for the read/write operation.

Memory enable input. This input determines whether one has access to the memory bank or not. When enabled, one can read/write into or from memory. When disabled, storage of data in memory is inhibited.

The chip, when disabled, may also do nothing.

Read/write select input. This input determines whether one writes data into the memory or reads data from memory.

Data input or inputs. This input or this group of inputs are the actual data that are stored in the memory.

Data output. This output or group of outputs provide the data, usually in inverted form, that is stored in a memory location or group of locations selected by the memory cell select inputs.

An example of an inexpensive and popular random access memory in the 7400-series of chips is the 7489 64-bit read/write memory, which is also known as the bipolar scratch pad memory (16 X 4 RAM) as shown in Figure 4 (Pg. 25).

Software - The Real Challenge

It is well known that in any computer system the software is likely to account for the largest fraction of the development cost. This is no less true for microcomputers, and it is indicative of the importance of giving the user comprehensive support. Although few microcomputer software packages compare favorably with those available for larger and more expensive machines, they considerably simplify the task of putting together a microcomputer system.

A microcomputer program consists of a sequence of binary words stored in a control memory. The instructions thus defined are said to be written in machine language Figure 5 (Pg.26) Although a programmer can elect to write his program directly in this form, the process is time-consuming and prone to error. Programming is made considerably

MSI **DM54/DM7489,L89A**

64-Bit Read/Write Memories

General Description

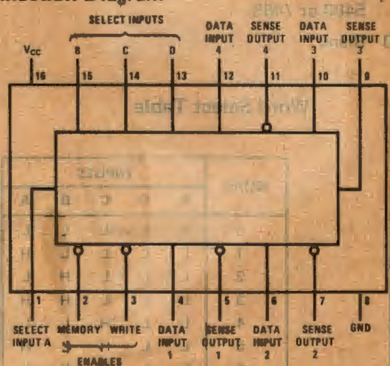
The DM5489/DM7489, DM54L89A/DM74L89A are fully decoded 64-bit RAMs organized as 16, 4-bit words. The memory is addressed by applying a binary number to the four Address inputs. After addressing, information may be either written into or read from the memory. To write, both the Memory Enable and the Write Enable inputs must be in the logical "0" state. Information applied to the four Write inputs will then be written into the addressed location. To read information from the memory the Memory Enable input must be in the logical "0" state and the Write Enable input in the logical "1" state. Information will be read as the complement of what was written into the memory. When the Memory Enable input is in the logical "1" state, the outputs will go to the logical "1" state.

The "A" suffix on the low power versions is used to denote that full "tenth-power" technology has been employed in building this RAM.

Features

- For application as a "scratch pad" memory with nondestructive read-out
- Fully decoded memory organized as 16 words of four bits each
- Fast access time DM54/74—35 ns typical; DM54L/74L—110 ns
- Diode-clamped, buffered inputs
- Open-collector outputs provide wire-OR capability
- Typical power dissipation DM54/74—400 mW; DM54L/74L—75 mW
- Pin compatible with 3101, MM5501

Connection Diagram



5489(J); 7489(J), (N); 54L89A/74L89A(J), (N), (W)

Truth Table

MEMORY ENABLE	WRITE ENABLE	OPERATION	OUTPUTS
0	0	Write	Logical "1" State
0	1	Read	Complement of Data Stored in Memory
1	X	Hold	Logical "1" State

Logic Diagram

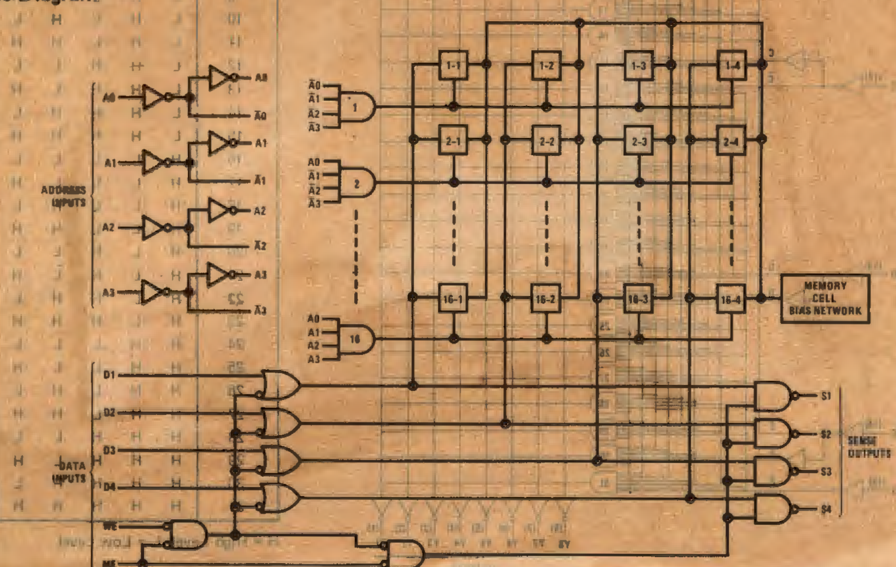


Figure 4 (64-Bit Read/Write memories)

MACHINE LANGUAGE	ASSEMBLY LANGUAGE	EXPLANATION
00 100 001 00 000 010 00 000 111	ABS LXIH 2 7	Load into the memory-address "pointer" (a special register in the CPU) the address, defined by 2 and 7, of the location where X is to be found in memory.
01 111 110	MOVA, M	Move the contents of memory X into the accumulator.
00 100 011	INXH	"Increment" the pointer (that is, add 1 to the address of X to locate the address of Y).
10 010 110	SUSM	Subtract the contents of memory Y from the accumulator (so that it now contains X - Y).
11 110 010 00 001 011 00 000 011	JP LOC	If the accumulator is positive (X > Y), jump to LOC. (The address of LOC is given by the next two eight-bit words, the binary representations of 11 and 3.)
00 101 111	CMA	Since the accumulator represents a negative number (X < Y). The corresponding positive number is obtained, in binary arithmetic, by "complementing" the accumulator (that is, exchanging all 1's and 0's) and ...
00 111 100	INRA	.... by incrementing the accumulator (Adding 1).
11 000 110 00 000 101	LOC ADI 5	Add 5 to the accumulator (Which gives either X - Y + 5 or Y - X + 5).
00 100 011	INXH	Increment the pointer (which means adding another 1 to the address of Y to locate the address of Z).
01 110 111	MOVMA	Move the contents of the accumulator into memory (Z).
11 001 001	RET	The subroutine has been completely executed. Return to the main program.

(Microdata Corporation, 1971 Pg 152)

Figure 5 Sample Program

simpler by assembly languages, which are available for all microcomputers. These languages allow the substitution of mnemonic words such as ADD, SUB, and JUMP for the binary words of the machine language; they also simplify the task of putting program data into a memory by giving the memory "address" arbitrary labels instead of absolute locations. An assembly-language program must be translated into machine language before it is committed to a memory. This conversion is accomplished by an assembler, which checks the assembly-language program for certain types of errors and, if none are found, produces the desired machine language code. The assembler is a program that sometimes can be executed by the microcomputer itself.

A higher level in the hierarchy of programming languages is represented by procedure-oriented languages, such as Fortran or PL/M. To translate statements written in these languages into machine language one uses a compiler. When a compiler is available, it speeds up programming, and the resulting easier to understand programs simplify the problems of documentation and maintenance. Unfortunately compilers, because of their general nature, tend to generate machine-language programs that are not highly efficient in speed of execution or in number in instructions. Typically they require from 10 to 100 percent more control memory than would have been needed if the programmer had worked at the assembly-language level. The decision whether or not to use a compiler is based on trade-offs involving the experience of the programmer, the time available for software development and, probably most important, the expected production volume of the system. Economics in memory achieved by more efficient programs can obviously justify greater software-development costs. (Brana, 1976, 21-26)

The differences between microcomputer software and minicomputer software go beyond the support level. In general, microcomputers have a significantly less extensive instruction set. Their shorter word lengths (four to 16 bits against the minicomputer's eight to 32) restrict the number of binary combinations that can be used in the instructions and the number of ways memories can be addressed, which increases the number of steps needed to execute an operation. This fact combined with the inherently slower performance of MOS devices to make present microcomputers between three and 10 times slower than minicomputers.

Microprocessor Selection

The importance of individual microprocessor characteristics depend heavily on the application. The key features to consider before making a selection are:

Word length is the first feature usually considered. The determining requirements include analog resolution, computational accuracy, character length, and width of parallel inputs or outputs.

Architectural features include the number of CPU registers, type of return stack, interrupt capability, interface structures, and memory types. The number of registers in the processor is obviously an important feature of its architecture.

Stack is used for nesting of subroutines, processing interrupts, and for temporary storage of data. Hardware stacks are implemented as on-chip pushdown stacks. Software stacks are stored in RAM, with an on-chip stack pointer number of registers in the CPU stack. The software stack, on the other hand, is restricted only by the size of external RAM.

The speed of the microprocessor has been gauged using numerous measures. Some include cycle time, state time, minimum instruction time, register-to-register additional time, and interrupt response time. Register-to-register addition time is a popular estimate of the computing speed because nearly every processor has an add instruction.

The degree of programming flexibility can be assessed by an examination of memory, simplify programming, and increase speed by using one-word memory reference instructions.

The number of additional IC packages required for a microcomputer system is an indication of the completeness of the microcomputer set. Support hardware is often required for clock generation and timing, memory and I/O control, data and address buffers, multiplexer inputs, interrupt control, and power supply voltages. The following microprocessor checklist is very useful in the evaluation of a candidate processor as shown in Figure 6 (P.30)(Lee, 1975, Pg. 159-164)

Application of Microcomputers

A good example of an application for which microcomputers are ideally suited is systems designed to control automobile traffic lights. Although these control systems have a constant objective (the regulation of vehicle and pedestrain flow), the requirements and constraints can change significantly, depending on state and city regulations, the intersection configuration, the traffic patterns and densities, the seasons and even the time of day. In earlier designs, which incorporated

MICROPROCESSOR SCORECARD	Classification	Technology	Clock Driver	I/O Interface	UART/USRT	RAM	ROM/PROM	Interface	Interrupts	One-Chip CPU	Microprogrammed	Accessible Stack	DMA Ability	BCD Arithmetic	Word Size (Data/Instruction)	Address Capacity (Program Words)	Clock (kHz/Phases)	Register Add Time (μsec per Data Word)	Number of CPU Registers: ALU	XR	GP	Return Stack Size (Nr x Bits)	Voltages Required	Power Dissipation	Operating Temperature Range (°C)	Package Sizes (DIP Pins)	Price Range (approx., 100 qty. CPU)	Status: First Samples	Status: First Deliveries	Remarks
BURROUGHS MINI-D	8-Bit CPU	PMOS								✓					8/12	256	1000/1	9	3		1	1 x 8	−12, +5		0 ~ 70	16	$ 60	2Q73	3Q73	ROM on CPU
ELECTRONIC ARRAYS	8-Bit CPU	NMOS		✓		✓	✓		✓	✓		✓	✓	✓	8/8	64K	1000/2		1	1	15	7 x 16				40		3Q75		16-Byte String Operations
FAIRCHILD F-8	8-Bit CPU	NMOS	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	8/8	64K	2000/0	2	65			(RAM)	+5, +12	.6	0 ~ 70	40	$ 75	4Q74	1Q75	Clock on Chip
FAIRCHILD PPS-25	4-[illegible] CPU	PMOS		✓		✓	✓							✓	4 x 25/12	6656	400/2	62.5	1			4 x 12	−9, +5	.6	0 ~ 70	16,18,24,40	$ 60		2Q71	
INTEL 4004	4-Bit CPU	PMOS	✓	✓		✓	✓	✓		✓				✓	4/8	4K	740/2	10.8	1		16	3 x 12	15 or (−10,+5)	1.0	0 ~ 70	16	$ 30	2Q71	4Q71	
INTEL 4040	4-Bit CPU	PMOS	✓	✓		✓	✓	✓	✓	✓				✓	4/8	4K	740/2	10.8	1		24	7 x 12	15 or (−10,+5)	1.0	0 ~ 70	16, 24	$ 40	4Q74	4Q74	8K Program Space in Two Banks
INTEL 8008-1	8-Bit CPU	PMOS	✓	✓	✓	✓	✓	✓	✓	✓					8/8	16K	800/2	12.5	1		6	7 x 14	−9, +5	1.0	0 ~ 70	18	$ 60	4Q71	1Q72	Second Source: MIL
INTEL 8080	8-Bit CPU	NMOS	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	8/8	64K	2083/2	2	1		6	(RAM)	−5,+5,+12	1.0	0 ~ 70	40	$ 200	4Q73	2Q74	
INTEL 3001	2-Bit Slice	Bipolar				✓	✓	✓	✓		✓		✓		2N/18+	512	6061/1	.165	2		10	(NONE)	5	4.5	0 ~ 70	28,40	$ 550	3Q74	3Q74	
INTERSIL 6100	12-Bit CPU	CMOS	✓			✓			✓	✓			✓		12/12	4K	4000/0	5	1		1	Modifies Program	5	.01	−55 ~ +125	28, 40	$ 175	4Q74	1Q75	PDP-8 Code; Clock on Chip
MONOLITHIC MEMORIES 6701	4-Bit RALU	Bipolar									✓				4/17		6666/1	.2	3		16	(NONE)	+5	1.	0 ~ 75	40	$ 95	1Q74	2Q74	
MOSTEK 5065	8-Bit CPU	PMOS							✓	✓		✓	✓		8/8	32K	1400/3	10	3			(RAM)	−12,−5,+5	.7	0 ~ 50	40	$ 100	1Q74	3Q74	3-State CPU
MOTOROLA 6800	8-Bit CPU	NMOS		✓	✓	✓	✓		✓	✓		✓		✓	8/8	64K	1000/2	2	2	1		(RAM)	5	.25	0 ~ 70	24, 40	$ 150	2Q74	4Q74	Second Source: AMI
NATIONAL CMP-8	8-Bit CPU	NMOS		✓	✓	✓	✓	✓	✓	✓		✓	✓		8/8	64K	/2	1.6	2	2		(RAM)				40		1Q75	2Q75	
N[illegible]AL GPC/P	4-Bit Slice	PMOS									✓	✓	✓		4N/23	100	715/4	1.4	8			16 x 4N	−12, +5	.7	0 ~ 70	22,24	$ 150	1Q73	3Q73	1 ≤ N ≤ 6
[illegible]AL IMP-4	4-Bit CPU	PMOS						✓	✓		✓			✓	4/4	4096	500/4	12	4			7 x 12	−12, +5	1.0	0 ~ 70	24,40	$ 150	3Q74	4Q74	16 x 4 Data Stack
NATIONAL IMP-8	8-Bit CPU	PMOS							✓		✓	✓	✓		8/8	64K	715/4	4.6	3	1		16 x 8	−12, +5	1.0	0 ~ 70	22, 24	$ 230	4Q73	1Q74	
NATIONAL IMP-16	16-Bit CPU	PMOS							✓		✓	✓	✓		16/16	64K	715/4	4.6	2	2		16 x 16	−12, +5	1.4	0 ~ 70	22, 24	$ 310	1Q73	3Q73	
RAYTHEON RP-16	4-Bit Slice	Bipolar							✓		✓				4N/48	64K	5000/1	1	1	1	2	(RAM)	5		−55 ~ +125	48		4Q74		
RCA COSMAC	8-Bit CPU	CMOS							✓	✓		✓	✓	✓	8/8	64K	2670/1	6	1		p	r x 16	5 ~ 12	.01	−55 ~ +125	28, 40	$ 300	4Q74	2Q75	(p+2r) ≤ 30
ROCKWELL PPS-4	4-Bit CPU	PMOS	✓	✓	✓	✓	✓			✓		✓		✓	4/8	4K	200/2	5	1		1	2 x 12	17	.225	0 ~ 70	42	$ 45	1Q72	3Q72	8K Program Space in Two Banks
ROCKWELL PPS-8	8-Bit CPU	PMOS	✓	✓	✓	✓	✓		✓	✓		✓	✓	✓	8/8	16K	256/2	4	1	1	2	(RAM)	17	.3	0 ~ 70	42	$ 47	4Q74	1Q75	
SIGNETICS 2650	8-Bit CPU	NMOS							✓	✓				✓	8/8	32K	1200/1	4.8	7			8 x 15	5	.5	0 ~ 70	40	$ 120	4Q74	1Q75	
TOSHIBA TLCS-12	12-Bit CPU	NMOS		✓		✓	✓		✓	✓	✓				12/12	4K	1000/3	13	4		6	(RAM)	−5, +5	.8	−20 ~ +80	16,24,26,42	$ 215	2Q74	3Q74	
TRANSITRON TMC/1601	4-Bit Slice	Bipolar							✓		✓	✓	✓		16/16	32K		4	u	v		(RAM)				40		2Q75	3Q75	(u+v) ≤ 8

Figure 6 Microprocessor Scorecard
(Torrero, 1975, Pg. 105)

a great many separate curcuits and relays, sections of each controller had to be custom-wired to meet the requirements of a specific intersection. With the substitution of microcomputers, traffic-light controllers can use common equipment throughout; the only custom-designed features are at the program level and are included in the read-only memories. Thousands of such controllers are currently being installed.

Other typical microcomputer applications are desk-top computers, compact business machines, bank terminals, check processors, payroll systems, process controllers and chemical analyzers. Over the next few years microcomputers should penetrate strongly into such areas as communications (with "smart" terminals), biological and medical research (with better monitoring and diagnostic instruments) and education (with more practical and economical teaching machines). In the 1980's microcomputers will be commonplace in the home in consumer electronic products, appliances, security devices and innumerable gadgets and toys. (Lewis, 1973, Pg. 106-110)

The automative industry will probably be the largest single user of microcomputers. They will replace the custom-designed LSI devices already installed by many manufacturers for monitoring brakes, lights and battery, for seat-belt interlocks and for skid control. Together with appropriate transducers and sensors, they will take up many new functions such as the control of ignition, the optimization of the fuel-air mixture, the reduction of exhaust emmissions and even the continuous indication of fuel consumption in miles per gallon. (McGlynn, 1976, Pg. 21-25)

The telephone system also offers a large market for microcomputers. Already used for some time at Bell Laboratories in instruments especially for testing equipment, microcomputers are now being included in some of the newer Bell System central-office equipment for automatic maintenance and protection. They are also at the heart of the new family of "Transaction" telephones that enable merchants and banks to check a customer's credit within a few seconds. Under microcomputer control the new telephones automatically read the customer's magnetically encoded credit card, consult a computer data base over the ordinary lines of the direct-dialing network and authorize the transaction. (Lee, Vol 22, Pg. 150)

The demand for microcomputers and the continuing evolution of solid-state technology will reduce the cost of microprocessors and the memories, will improve production yields, and will lead to higher levels of integration. Particularly promising at the present time is one of the newest bipolar technologies: integrated injection logic (IIL). This technology is characterized by a greater density of components on the substrate, higher speed of operation and lower power requirements. Microcomputers will be among the first devices to exploit these advantages. (Terrero, 1975, Pg. 102-103)

Chapter IV

PROJECT ACTIVITY INFORMATION

It is decided by the writer to design, construct and test a 8080A microcomputer system as shown in Figure 7 (P.35)The cost to build this protype was about $40. The total assembly time for the 8080A microcomputer system was about twenty hours. The 8080A microcomputer system was interfaced with a octual keyboard.

The 8080A system consists of a microprocessor, and a number of complementary chips, manufactured using N Channel Silicon MOS technology. Components of the 8080A system include the following devices:

The 8080A Microprocessor

The 8224 Clock Generator

The 8228 System Controller

Bus Structure

Address Decoding

Memory

Keyboard and LED Display

Software Control

This 8080A system consists of four print circuit boards:

Central Processor Board, RAM Board, PROM Board and Interface Board. This 8080A system has a unique buss concept which presents all necessary signals to all the cards in the system. The boards can be plugged into any position on the buss to build a complete system.

The Central Processor Board forms the heart of the microcomputer and contains the 8080A microprocessor, 8224 clock generator, 8228 system controller, 19 MHZ crystal, and 8212 Tri-state buffers. The RAM Board consists of 24, 2102 (1K X 1 Bit), random access memory, capable of 3K on read-write memory and a 74154 address decoding chip. The PROM Board consists of 8, 1702A, untraviolet erasable programmable read-only-memory (PROM) for permanent storage of programs and data. The Interface Board consists of keyboard, LED display and decoding.

The 8080A Microprocessor

The 8080A is a complete 8-bit parallel central processing unit (CPU) for use in general-purpose digital computer systems (Figure
It is fabricated on a single LSI chip using Intel's n-channel silicon gate MOS process, thus offering much higher performance than conventional microprocessors. A complete microcomputer is formed when the 8080 (ACPU) is interfaced with I/O parts and any type or speed of semiconductor memory.

The 8080A contains six 8-bit data registers, an 8-bit accumulator, four 8-bit temporary registers, four testable flag bits, and an 8-bit parallel binary arithmetic unit. The 8080 also provides decimal arithmetic capability, and it includes 16-bit arithmetic and immediate operators which greatly simplify memory address calculations and high-speed arithmetic operations.

The 8080 also contains a 16-bit stack pointer to control the addressing of this external stack. One of the major advantages of the stack is that multiple-level interrupts can readily be handled, since complete system status can easily be saved when an interrupt occurs and then be restored after the interrupt. Another major advantage is that almost unlimited subroutine nesting is possible.

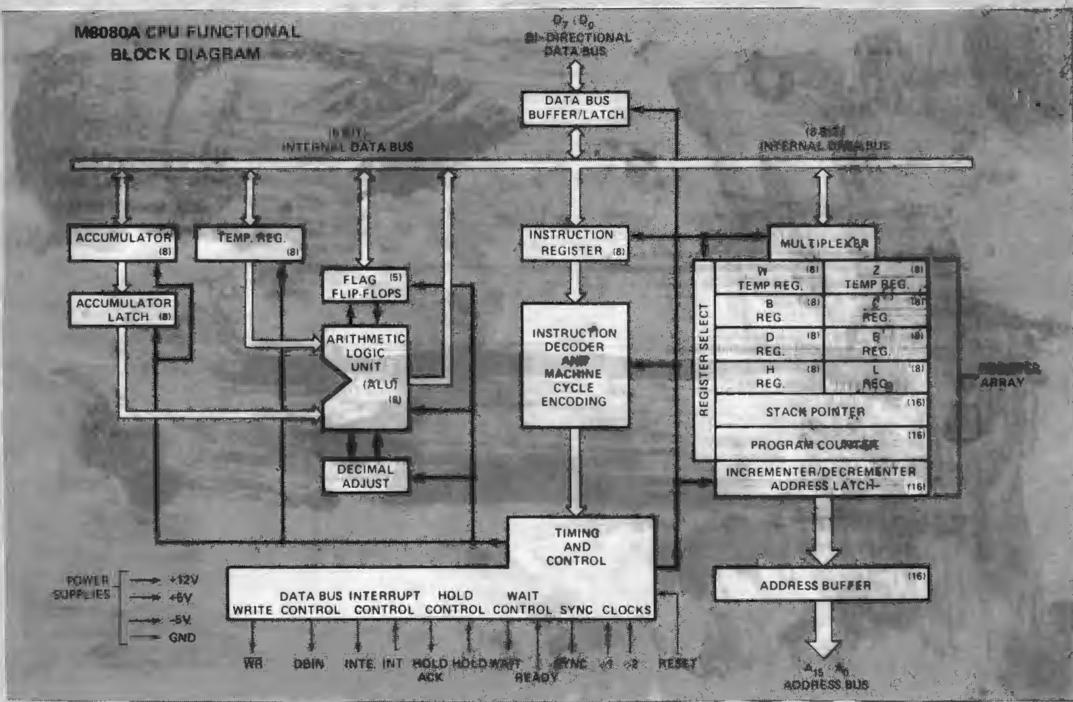


Figure 7 8080A CPU Functional Block Diagram

(Intel, 1975, Pg. 5-29)

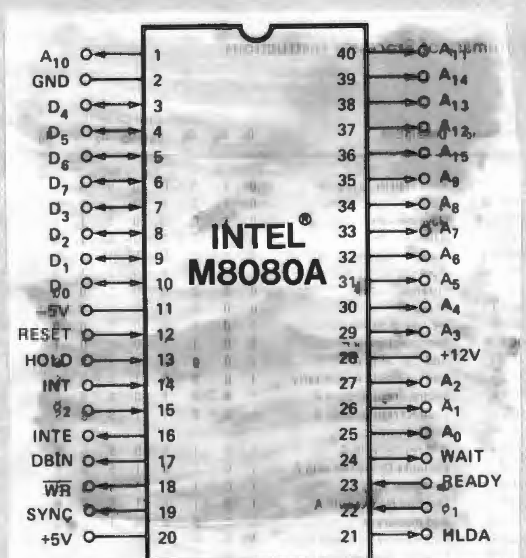


Figure 8 8080A PIN Configuration

(Intel, 1975, Pg. 5-32)

The 8080 has a stack architecture, wherein any portion of the external memory can be used as a last-in/first-out stack to store/retrieve the contents of the accumulator, the flags, or any of the data registers.

This processor has been designed to greatly simplify system design. Separate 16-line address and 8-line bidirectional data busses allow direct interface to memories and I/O ports. Control signals, which require no decoding, are provided directly by the processor. All busses, including control, are TTL comparible. (Intel, 1975, Pg 5-13)

The following describes the functions of all the 8080A pins: The pin connections is shown in Figure 8, (Pg. 38)

OUTPUTS

$A_o - A_{15}$	ADDRESS LINES - Sixteen Bits of address data of memory and 256 output devices (Tri-State)
$D_o - D_7$	BUFFERED DATA OUTPUTS - To memory and external devices. (Tri-State)
ø1	PHASE ONE CLOCK SIGNAL - 2 MH_z; used to synchronize computer operations
HLTA	HALT ACKNOWLEDGE - indicates that the computer has entered a halt state by executing a halt type instruction.
HLDA	HOLD ACKNOWLEDGE - Signals external devices that the computer has suspended operations and is in a hold state. Address lines $A_o - D_7$ are forced into their high-impedance or third state, allowing other devices to use the buss.
INTE	INTERRUPT ENABLED - Output to external devices to signal that the interrupt circuitry in the 8080 chip is now able to sense external interrupts.
INTA	INTERRUPT ACKNOWLEDGE - Used to signal external devices that the computer has been interrupted.
WAIT	WAIT - Inditaces that the computer is waiting for a synchronizing READY input before going further in executing the program. This is used in the Single Step mode.

$\overline{IN}$, $\overline{OUT}$	$\overline{IN}$ and $\overline{OUT}$ - Used to synchronize the flow of data to and from the computer to external devices. Signals are issued with respect to the computer. These signals are generated when either an INPUT or OUTPUT instruction is executed by the computer. Active at Logic 0.

INPUTS

MD_0 - MD_7	MEMORY INPUT LINES - Data from a selected memory location will always be present on these lines, ready to be inputted into the computer through circuits on the CPIC-80/B board.
D_0 - D_7	DATA INPUT BUSS - Used to input data to the computer from external devices or for access to the bi-directional eight bit data buss.
HOLD	HOLD SIGNAL - Used to put the 8080 in the HOLD state, suspending operation and forcing the address and data output busses into their high-impedance or third state.
$\overline{INT}$, $\overline{EXT}$.	$\overline{INTERNAL}$ and $\overline{EXTERNAL}$ INTERRUPT - Interrupts the computer and allows external hardward to jam an instruction into the computer to cause it to perform some other function. There are two of these inputs. One os for external devices and the other is for the MICRO Control Panel (MCP-80/B.
READY	READY - Used to indicate that the computer is to wait for an external pulse before going on to the next instruction. When at logic 0, the computer enters the WAIT state. At a logic 1, the the computer operates normally. (Intel, 1975, Pg. 5-14)

The 8224 Clock Generator

The 8224 is a crystal-controlled oscillator that divides the crystal frequency by 9 Figure 9, (Pg.39). Therefore, this microcomputer is operating at 1.111 mHz, this is slower than the average operating speed of the 8080, which normally operates at 2 mHz.

The 8224 generates the two phase clock needed for the 8080 and phase 2 for any TTL circuitry need, along witb auxiliary timing signals.

The status strobe is the most important auxiliary signal generated in the 8224. It is used to latch the status information in the 8228 at the start of each machine cycle. This event is triggered by the SYNC signal from the CPU; or the power-on REset, which generates the status strobe for a longer period of time, causing the 8080 to be automatically reset. The status strobe is fed directly to the 8228. (Intel, 1975, Pg. 5-10)

The 8228 System Controller

This chip generates all signals required to directly interface the EPROM and RAM used in this system Figure 10, (Pg. 39). It also provides a bi-directional bus driver for high system TTL fan-out, and isolation for the 8080 data bus from memory. This allows for the optimization of control signals, enabling the system designer to use slower memory and enhances system noice immunity, the 8228 also controls the signals for a multiple byte instruction. These functions simplify the necessary circuit for decoding compared to that needed by the 8080.

As mentioned earlier the 8228 latches status information at the beginning of a machine cycle. A low is needed at STSTB to accomplish this function.

The "read" control signals (MR, INTA, and I/O IN), are derived from the logical combination of the appropriate status bits and the DBIN input from the 8080 CPU.

The "Write" control signals (I/O OUT and MW are derived from the logical combination of the appropriate status bits and $\overline{\text{WD}}$ from the CPU.

The BUSEN (Bus Enable) input to the gating array is an asynchronous input that forces the data bus output buffers and control signal buffers into their high-impedance state if at a "one". A "one" on the HLDA

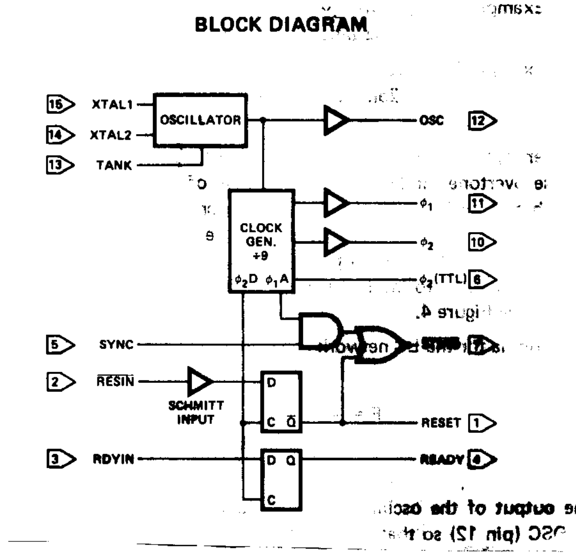


Figure 9 8224 Diagram

(Intel, 1975, Pg. 5-1)

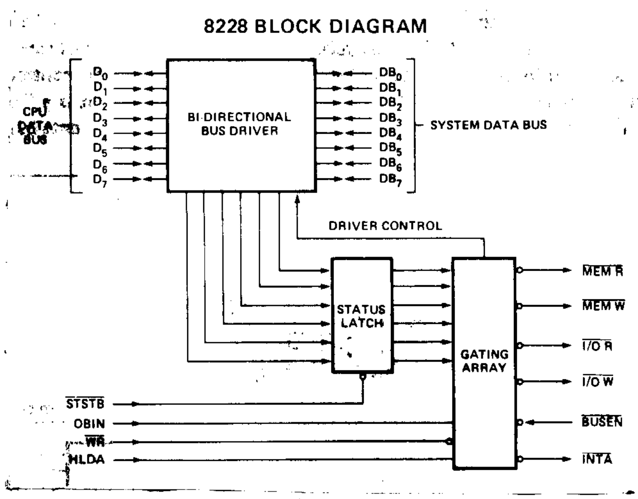


Figure 10 8228 Diagram

(Intel, 1975, Pg. 5-7)

(hold data) will latch data into the 8228. A "zero" on the HLDA and BUSEN will float the data buses. This muct not occur within set-up time of 140nsec. prior to the rising edge of ∅2 clock, with HLDA as the input and Q output floating address and data input busses. $\overline{Q}$ is used to place data output bus in its third state. The HOLD REQUEST must be delayed in the same manner; 7474 TDA is for this purpose. (Intel, 1975, Pg. 5-7)

Bus Structure

The system consists of three busses: Address, Data, and Control. The Address Bus (AO-A15) of the 8080A is sufficient to support a small system that has a moderate Memory and I/O structure. To expand the size of the system that Address Bus can support, a simple buffer can be added, as shown in Figure 11, (Pg.41). The 8212 is an excellent device for this function. Like the address Bus, the data needs the support of buffers for system expandion. The control bus gets its' buffering from the 8228.

Address Decoding

The address decoding is accomplished by a 7442, an AND gate and two 74154's 1A and OB. Figure 11, (Pg. 41). The decoding for the EPOM employees a 74154 (1A) to decode A10 - A13 and an AND gate to encode the output of the 7474 then decodes these inputs and properly selects the EPROM. After the EPOM has been selected, the lower eight address bits select the location in (256X8 EPROM.

The decoding for RAM memory is achieved through 74154. Ths system has the capability to select 13 banks of memory, but only one memory bank of 2102's (1024X1) is shown decoded in Figure 13, (Pg. 44). The A10 - A13

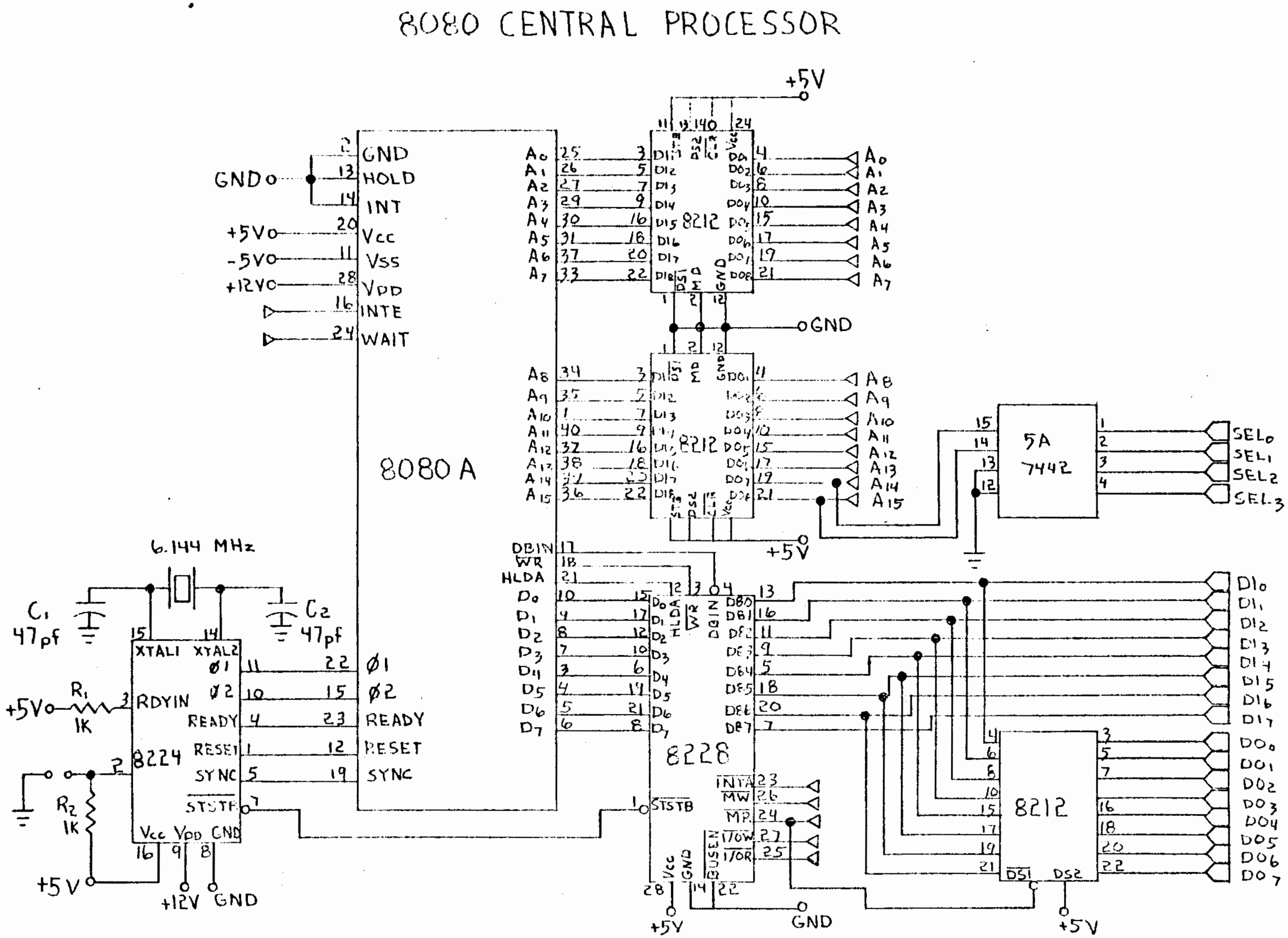


Figure 11 Central Processor Board

are decoded to select the memory bank and A0 - A7 selects the locations within that bank. The $\overline{MW}$ from pin 26 of 8228 selects the Read/Write mode of the 2102's.

Memory

There are two types of memory used in this system: the 1702 (256X8) Erasable Programmable Read-Only Memory (PROM and 2102 (1K) and Random Access Memory (RAM). The 1702A is a 256 word by 8-bit electrically programmable ROM ideally suited for uses where fast turn-around and pattern experimention are important. Initially all 2048 bits of the 1702A are in the zero state. Information is introduced by selectively programming ones in the proper location. The 1702A is packaged in a 24 pin dip dual inline package with a transparent lid. The transparent lid allows the user to expose the 1702A to ultraviolet light to erase the bit pattern. A new pattern can then be written into the device. The circuitry of the 1702A is completely static. No clocks are required. Access times from 650ns to 1.5ns are available. The 1702A is fabricated with silicone gate technology. This low threshold technology allows the design and production of higher performance MOS circuits and provides a higher functional density on a monolithic chip than convential MOS technologies. (Intel Data Catelog, 1977, Pg. 305)

The 2102 is a 1K word by 8-bit static random access memory element using normally off N channel MOS devices integrated on a monalithic array. It uses fully DC stable (static) circuitry and therefore requires no clocks or refreshing to operate. The 2102 is designed for memory applications where high performance, low cost, large bit storage, and simple interfacing

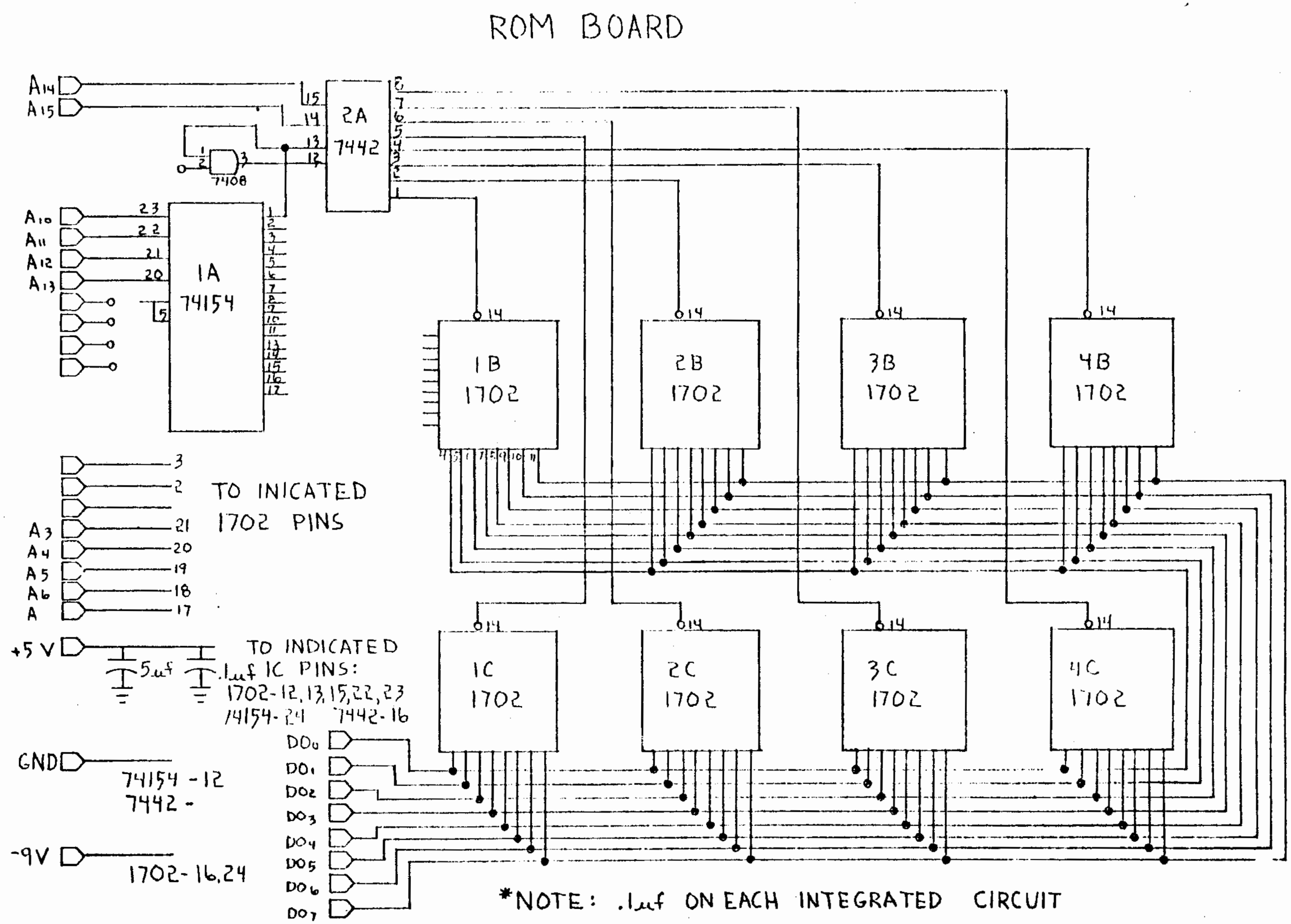


Figure 12 Prom Board

RAM BOARD

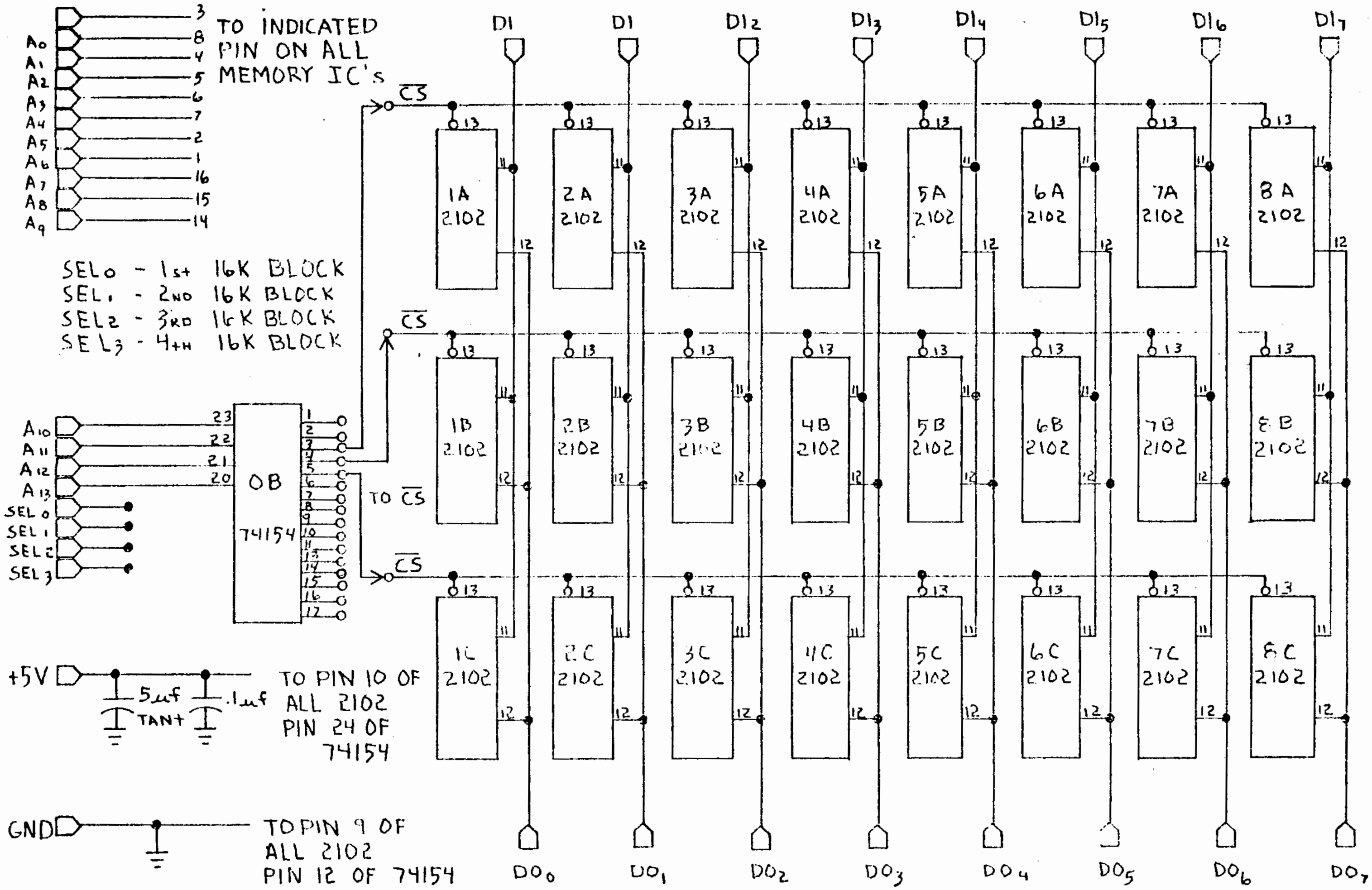


Figure 13 RAM Board

are important design objectives. It is directly TTL compatable in all respects, input, outputs, and a single +5 volt power supply. Two chip enables allow easy selection of an individual package when outputs are OR tied. An output disable is provided so that data inputs and outputs can be tied to a common I/O system. The output disable function eliminates the need for bi-directional logic in a common I/O system. The 2102 is fabricated with N-channel silicon gate technology. This technology allows design and production of high performance, easy to use MOS circuits and provides a higher functional density on a monolithic chip than either conventional MOS technology or P channel silicon gate technology. (Intel Data Catalog, 1977, Pg. 2-30)

Keyboard and Led Display

The debounding and decoding of input from the keyboard is done with a software subroutine that is stored in EPROM Figure 14, (Pg. 46).

There are three groups of LED's; the high location, the low location and memory data. Lighting of groups is controlled by the decoding of AO-A7 by a five input OR gate, 7442 decodes and three two input OR gates. The five input OR gate is constructed from six open collector inverters. A3-A7 are ORed together, a one from the OR is used by the 7442, to disable the keyboard for any value greater then seven.

Address bits AO-A2 are used in the 7442 decoder chip to provide a one out-of-eight code for devices o to 7. The three, two input NORs are used in input data to the appropriate 7475 latch by the $\overline{\text{OUT}}$ pulse from the 8228. The data from the keyboard is placed on the bus via a 74365 by the $\overline{\text{IN}}$ signal from the 8228, and signal from pin 1 of the 7442

KEYBOARD AND OUTPUT PORTS

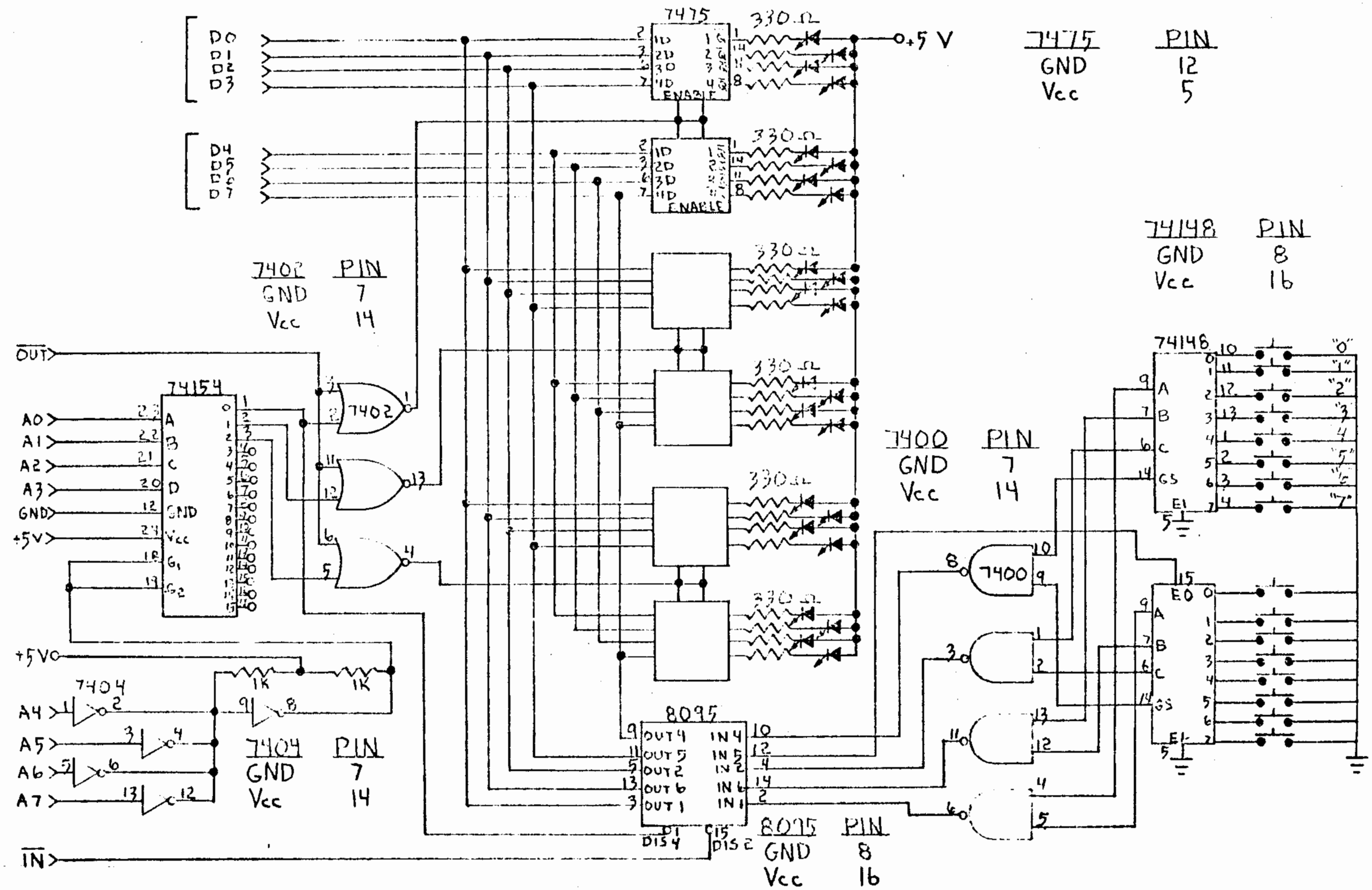


Figure 14 Interface Board

keyboard decoder. A low on both pins 1 and 15 enables the 74365 hex tri-state buffer. This data is latched up in the 8228 at the same time as it is latched in to port 0 LED display.

Key encoding is accomplished with a 8 to 3 line priority encoder and four NAND gates. Each pair of outputs A, B, and C are NANDed together and the GS pair are NANDed to form the key value. EO is fed directly into the data bus to indicate whether the key being depressed is a command or a value. The code is then decoded by software subroutine.

Software Control

The program to control the CPU and keyboard starts by initiating the stack pointer and the high and low address Figure 15, (Pg. 50,51).

Then it stores the memory data found at the high and low address and stores it at TEMP. The following step is to output high and low address to the LED's displayer, followed by the data stored at TEMP. Now the accepting of a value from the keyboard and the debouncing of the keys, takes place in a subroutine called KBRD. This subroutine will be explained later. Upon returning from the subroutine the value obtained in KBRD is compared to 010. If the code is a value it is converted to its binary equivalent and output to the data LED's via TEMP. Then the sequence is routed again.

If however, the key code is a control code. The program jumps to compare the keyboard value to, first 011, which would tell it to move TEMP to the low register. If not this value then the program compares keyboard value to 010, which would tell it to move TEMP to the high register. If not this value then the program compares the keyboard value to 013. This value tells the 8080 to store TEMP at the location pointed to to by the high and low register. Then increment the low register and display what is stored there, via TEMP to data LED's. If the value isn't 013 then the program compares it to 012. which would allow the CPU to go to the address indicated by H and L and execute the program found there. At the end of each step the program returns to specified point in the program and begins again.

The keyboard routine (KBRD) is used to accept a code from the keyboard. However, before accepting the code the subroutine makes sure the key is stable.

The program starts by accepting an input from the keyboard. Then ORing the accumulator with itself. If the result is negative, there is no key depressed and the program jumps back to accept a value. It remains in this loop until a value is accepted. At that time the delay subroutine is called, which delays the program for 10msec. Upon returning the program it checks to see is the same key is still depressed. If it is not, the program jumps back and looks at the keyboard again. If it is the same value the program calls delay then runs through the check process again. When the checking process is satisfied, the program AND's the keyboard value with 017. This masks out all but the needed code. Next H and L are pushed onto the stack H is set to 000. The new address to the key code is added and the accumulator is pushed to the low register. The value now found in memory is now moved to the accumulator, and the high and low address are popped from the stack. Subroutine is ready to return control the main program.

The 10msec delay subroutine sets up a loop that will last for 10msec. The program starts by pushing the accumulator and flags onto the stack.

Then the registers D & E onto the stack. Then it loads them with a value, this value being determined by the speed of the clock used in the system. Registers D & E are then decremented by one and D is moved to the accumulator. Register E is then ORed to the accumulator and the flags are checked to see if the accumulator is zero. If it is not zero, the program jumps back and the routine of decrementing and checking D & E is continued until they are zero. At that time, D & E are popped from the stack, the accumulator and flags are popped from the stack, and control is returned to the main program.

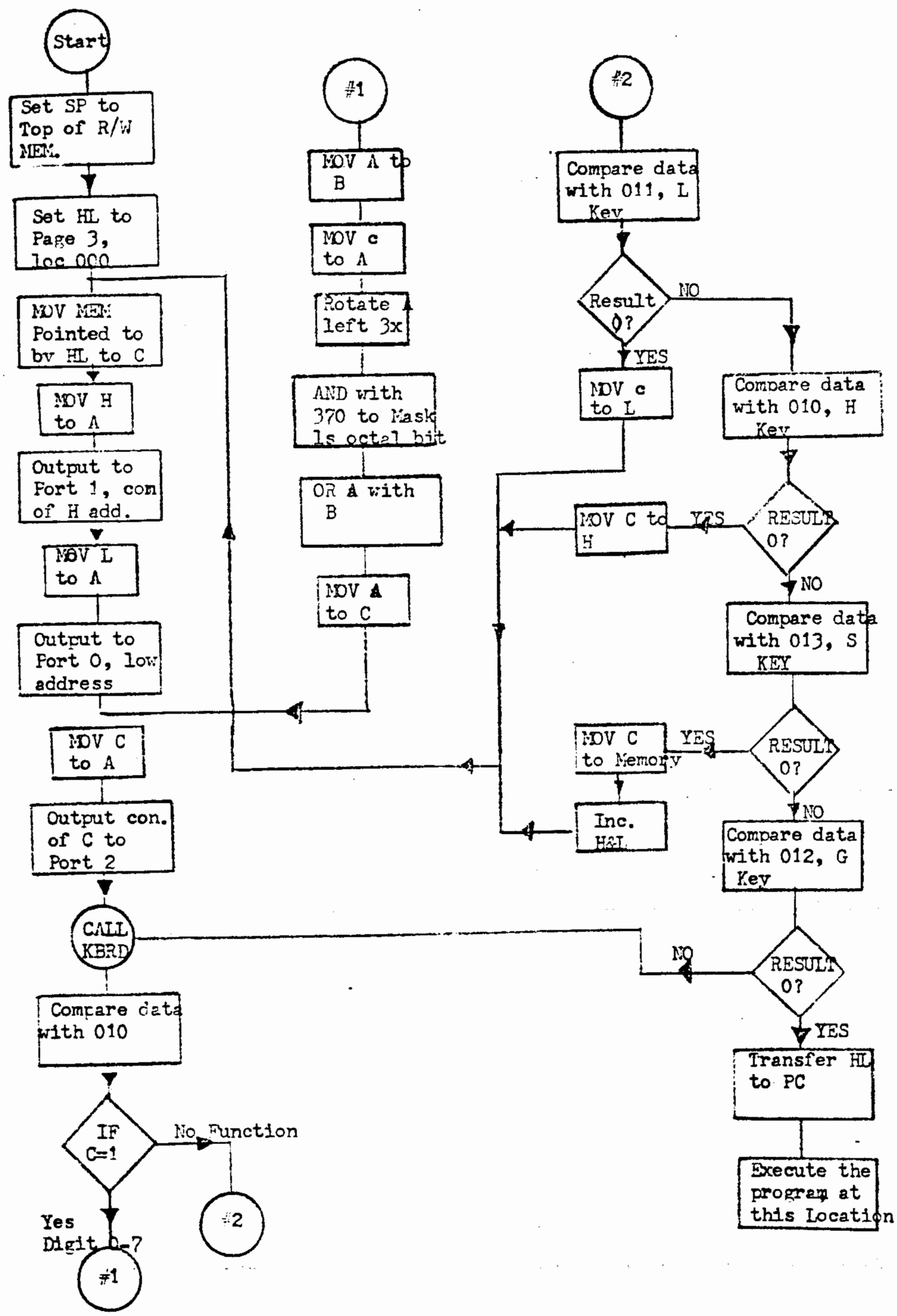


Figure 15 Software Control Flow Chart

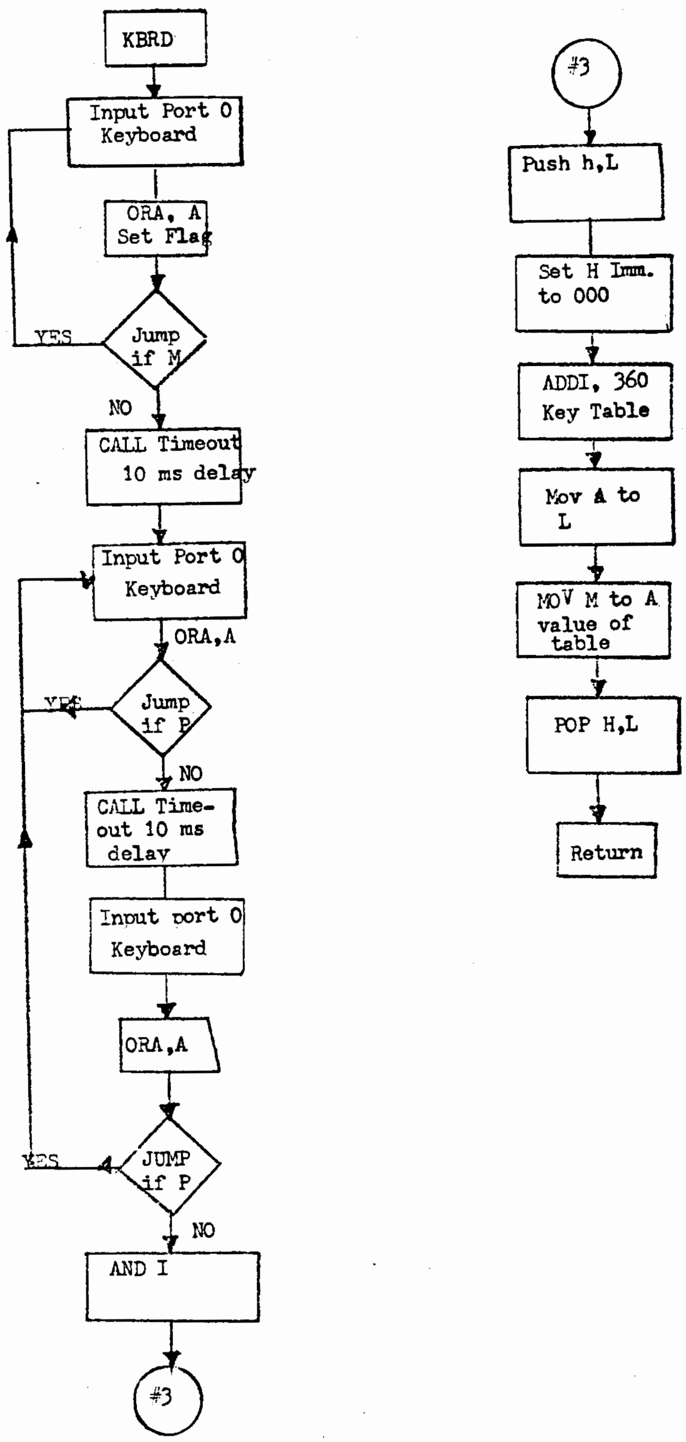


Figure 15 Software Control Flow Chart

Chapter V

RELATED INFORMATION

Protecting an Idea

The first step to patent an idea is to contact a capable patent attorney. The attorney will make a search of the registered patents to determine if the idea is original. If the research proves to be favorable then he will file for a patent at the United States Patent Office in Washington, D.C. The inventor should not under any circumstances attempt to act for himself.

The application is a petition to the commissioner of patents to grant patent letters to the inventor on the basis of the facts disclosed in the application; especially in that section of it, known as specifications. The procurement of the patent will depend not only on the state of the art, but on the skill of patent counsel in preparing the application. (Dock, 1973, Pg. 17-18)

When the application is accepted as complete by the patent office, it is allotted to an examiner in the examination division that has charge of the class of inventions to which the application relates. Generally the examiner, in conducting his own search, finds art that he believes anticipates the invention in one respect or another. He then summarizes his reasoning in a "patent office action", and rejects that part of the claims that in his judgement conflicts with prior art. It is then up to the patent attorney to file an "answer" to the "action", and if this is skillfully done the examiner may concede that the invention was not anticipated by the patents he has cited and withdraws his rejection, with a "notice of allowance" of the claims that he finds acceptable in the light of all the prior art and the attorney's further elaboration on the nature of the invention. If all goes well and the invention is thus determined to determined to be patentable, the notice of allowance will be sent to the inventor or his attorney. Ordinarily the granting of a patent requires some 5 or 6 weeks after notice of allowance, or issuance may be the option of the inventor be deferred for for as long as 6 months, in order to gain that additional period of protection of 17 years. When the patent goes to issue, the red-sealed, blue-ribboned document, signed by the commissioner of patents, is mailed either to the inventor or to his attorney for forwarding to him. In the unhappy event that the examiner cannot be convinced of the novelty of the claimed invention, his patent office action will explain his reasons for denying the claims and will bear the legend "final rejection". (Dock, 1973, Pg. 19-20)

Mass Production

The first step in the mass production of 8080A microcomputer system is the design. Then the design goes to the draftsman. It is the draftsman's job to lay out the schematic diagrams in a logical and orderly way with all the component's value in easy-to-reay prints. To do this he must know the physical size of the component and the area it will fit into so it can be tested by an electronic technician. From the results of his testing he will recommend the component to be scraped or remodeled to work more efficiently. After he gives his approval the component goes into production. If the component is being designed for another company it will be sent to that company for approval.

Production schedules are set for the building process. The etching of the printed circuit board is the next step in production. The resistant to the acid is put on the board by a photographic process, off-set printed or painting. Then the circuit boards are dipped in an acid and the resistant to the acid is removed. If the printed circuit is a multilayer printed board, it is important that each layer be exactly on top of each other. The layers are laminated on top of each other. Then holes for the components are drilled with a gang drill that is usually numerically controlled. The components are then mounted and soldered in place. In some modern factories they use a dip soldering method to save time and cost. The component is assembled in its case, ready for martet.

Study Unit

The goal of this study unit is to demonstrate the principles, concepts, and applications of an eight-bit microcomputer that is based upon the 8080A microprocessor integrated circuit chip. The student will be able to participate in a remarkable electronics revolution in which the computer will be transformed from a large, expensive, and rather esoteric machine into a compact, inexpensive, and common device that will be used by millions of individuals.

I. Unit objectives

A. To develop an understanding of the operation of a typical 8080A microcomputer system.

B. To develop an understanding to the general principles of computer interfacing that apply to most computers.

C. To develop an understanding of the many application of a microcomputer.

D. To develop an understanding of a microcomputer system design.

E. To develop a understanding of the following characteristics: word length, memory, program, input/output, controller, hardware, software, bus structure, decoding, etc.

II. Introduction

A. Historical information

B. Microprocessor information

C. Software information

D. Application information

E. Memory information

III. System design

A. The pin configuration and pin function of the 8080 chip.

B. The internal operation of the 8080A microprocessor.

C. The organization of a typical microcomputer that employs the 8080A chip.

D. Software control.

IV. Materials Used

A. Component Family
1. 8080A Microprocessor
2. 8224 Clock Generator
3. 8224 System Controller
4. 8212 Tri-State Device

B. Memory
1. 2102 Random Access Memory
2. 1702 Erasible Programmable Read-Only Memory

C. TTL Devices
1. 7402 Nor Gate
2. 7400 Nand Gate
3. 7405 Inverters
4. 7410 Nand Gate
5. 7442 Decoder
6. 7474 D Flip-Flop
7. 7475 Latch
8. 74148 Decoder
9. 74154 Decoder

D. Other material
   1. SK-10 Sockets
   2. LED
   3. Keyboard
   4. Power Supply

V. Laboratory Procedure

The purpose of this lab is to demonstrate the principles, concepts and applications of the microcomputer that is based upon the 8080A microprocessor.

A. Obtain all materials and equipment needed.

B. Use the 8080A system design by the instructor and breadboard the system.

C. When the system is operational, notify your instructor.

VI. Pupil Evaluation

To evaluate a pupil means to judge the worth of a pupil correctly in the ways that he has performed in a classroom. Evaluation may be subjective or objective. It would be of great significance to to establish on paper what factors will be utilized in any pupil evaluation. Following is a list of criteria which the writer might include in a standard to determine evaluation of pupils. The list is not intended to be complete since individual instructors may wish to add to or delete from this list.

1. Tests over textural materials, lectures, class discussions, demonstrations, and field trips.
2. The quality of a written circuit description of the 8080A Microcomputer System.
3. The value of student projects.
4. Involvement in class activities.
5. Student progress.
6. The growth of student capabilities.
7. Safe work habits.
8. Attitude.
9. Cooperation.
10. Work habits.

Chapter VI

SUMMARY

The writer has accomplished one main goal of this project. That was to obtain an understanding of the design of a 8080A microcomputer-based system. The motivation for this project was that the writer wanted to design a low-cost microcomputer system.

The lab work was interesting. In the building of the 8080A microcomputer system the writer discovered exactly how the system worked and found it to be fascinating.

An effective microcomputer designer requires at least three distinct capabilities. First, the designer must have a fundamental understanding of available components. This begins with the microprocessor chips, memory chips, and I/O chips which make up a microcomputer. It extends to the keyboards for communicating with the computer and displays for informing the user of results. Second, the designer must thoroughly understand the alogorithmetic processes required by each aspect of the design and be able to translate these into the language of the microcomputer. Third, the designer must understand how the extensive requirements of an instrument or device can be broken down into manageable parts.

An attempt to speculate on the potential applications of a new device as powerful and versatile as the microcomputer is bound to appear naive within a very few years. It is the writer's belief that the microcomputer will soon be as much a part of our lives as transistors or even moderately complex integrated circuits.

Extensive research effort has gone into the preparation of this paper, and it is the sincere hope of the writer that the material covered will provide the reader with an adequate description of the subject for his or her purposes. Yet there is always more material that could be covered. The writer regrets not being able to spend more time on the labatory project because of his teaching duties at Hawkeye Institute of Technology.

BIBLIOGRAPHY

References

1. Altman, Laurence, Microprocessors, New York, McGraw-Hill Book Company, 1975.

2. Brana, Arput, and Porat, Dan I., Introduction to Microcomputers and Microprocessors, New York, John Wiley and Sons, Inc., 1976.

3. Dock, Russell, Ideas, Inventions and Patents, New York, John Wiley and Sons, Inc., 1973.

4. Hilburn, Julich, Microcomputer/Microprocessors: Hardware, Software, and Applications, New Jersey Prentice-Hall, 1976.

5. Hoo-Min D. Toong, Microprocessors, Scientific American (September, 1977 Pg. 230-246).

6. Intel Corporation, 8080 Microcomputer Systems User's Manual, Santa Clara, California, 1975.

7. Intel Corporation, Data Catalog, Santa Clara, California, 1977.

8. Katzan, Harry, Microprogramming Primer, New York, McGraw-Hill Book Company, 1977.

9. Lee, Richard, Microprocessor ICS Improve Instruments, Electronic Design, Vol 22, No. 9,(Pg. 150-154).

10. Lewis, Donald R. and W. Ralph Siena, Microprocessor or Random Logic, Electronic Design (September 1, 1973, Pg. 106-110).

11. McGlynn, Daniel R., Microprocessors Technology, Architecture, and Applications, New York, John Wiley & Sons, Inc., 1976.

12. Microdata Corporation, Microprogramming Handbook, Santa Ana, California, 1971.

13. Noyce, Robert N., Microelectronics, Scientic American (September, 1977, Pg. 62-70).

14. Oliver, Bernard M, The Role of Microelectronics in Instrumentation and Control, (September, 1977, Pg. 180-192).

15. Rony, P.R. and Larsen, D.G., Logic and Memory Experiments Using TTL Integrated Circuits, Derby, Connecticut, E and L Instruments, 1974.

16. Sippl, Charles J. Microcomputer Dictionary and Guide, Champagne, Illinois, Matrix Publishers, Inc., 1976.

17. Terrero, Edward A., Microprocessors New Directions for Designers, New Jersey Hayden Book Company, Inc., 1975.

APPENDIX

Sample letter to Industry

Cedar Falls, Iowa 50613
December 6, 1977

Intel Corporation
3065 Bowers Avenue
Santa Clara, CA. 95051

Dear Sir:

I am an Industrial Technology graduate student doing research in the area of Microcomputer/Microprocessor's at the University of Northern Iowa, Cedar Falls, Iowa 50613. During the next 5 weeks I will be preparing a research report on the design of a low-cost microcomputer system.

I would appreciate information on your 8080A microprocessor and all information which you deem helpful in my research.

Thank you for your time and your assistance in this matter.

Sincerely,

Jerry G. Farrell