A DIGITAL PHYSIOLOGICAL SPECTRUM ANALYZER

by

GARY BRENT WOOD, B.S. IN E.E.

A THESIS

IN

ELECTRICAL ENGINEERING

Submitted to the Graduate Faculty
of Texas Tech University in
Partial Fulfillment of
the Requirements for
the Degree of

MASTER OF SCIENCE

IN

ELECTRICAL ENGINEERING

Approved

May, 1975

ACKNOWLEDGMENTS

I would like to thank Dr. William M. Portnoy, often my adversary in methodology but always an inspiration of principles and motivation, for his guidance and encouragement throughout my Master's program and especially in the preparation of this thesis.

I also wish to thank Dr.'s D.L. Gustafson and M.K. Rylander for serving on my committee.

Finally, I wish to thank my wife Rosanne, for her assistance and understanding.

TABLE OF CONTENTS

TABLE OF CONTENTS (CONTINUED)

LIST OF TABLES

Tables Page

LIST OF FIGURES

LIST OF FIGURES (CONTINUED)

CHAPTER I

INTRODUCTION

Clinical application of modern electronics and related fields has progressed through the years with the development and the increased sophistication of practical science and instrumentation. Hundreds of years before the science of electricity, Galen's doctrine of "animal spirits" was universally accepted as the explanation of nerve function. The electrical nature of living tissue was not considered credible until late in the eighteenth century. The work of the anatomist and physician Galvani in 1791 provided the impetus for new developments in the science of electricity. The physicist Volta was interested in Galvani's results, related to animal electricity, and proceeded with research that eventually led to the voltaic pile (early battery), and to Oersteds' discovery of magnetic fields[1]. The techniques of electrophysiology, electrocardiology, electroencephalography, and others, have grown from these early investigations of electric phenomena in living tissue. Detecting, recording, and evaluating the electrical rhythms of living tissue, and specifically the human body, is proving to be one of the most valuable clinical tools.

Medical explanation and interpretation of the more gross human body electrical activity, such as the rhythmic contractions of the cardiac muscle or the more sporadic responses of skeletal muscles, has become fairly routine. That is not to say that these areas are totally understood, but their degree of complexity appears to be less than that of the neurosystem, more specifically, the neuro-electrical phenomena of the human brain. Regardless of the type of electrical activity of the human body that we consider, there are many problems that arise in the gathering, ordering, and useful presentation of the derivable information contained in that activity. The electrical activity of the human body may take the form of evoked responses or may follow spontaneous rhythms; it may vary in frequency content and location, and change with mental attitudes and other physiological conditions. The human body is an almost endless maze of electrical pathways and generators providing a vast amount of interrelated, eletrically analyzable, and medically valuable information.

The most basic of all electrical activity is associated with the individual nerve cell. The potential difference between the inside and outside of the cell is the result of differences in chemical ion concentrations. During stimulation the cell membrane permeability is changed. the ion balance is disturbed so that the respective concentra-

tions are altered, resulting in a new potential difference. This simple changing potential is detectable and is in the order of 90 millivolts (Figure 1). Considering the millions of cells at work in the body, the resulting variable rates possible, and the influences of a propogated group of such reactions, the complexity of the interpretation of total electrical activity is more easily appreciated. It should be noted that while some of the simpler forms of electrical phenomena in the body are well understood there are those areas in which complete explanations are still lacking.

The electrical rhythms of the brain are among those unexplained areas. The existence of various electrical signals from the brain has been known since the late 1800's but their exact origin remains unknown. Medical Science proceeded to apply clinically the information contained in the brain's electrical activity rather than study its origin. The study of the brain's electrical activity is the science of electroencepalography and is a valuable medical analysis tool.

There is a substantial amount of electrical activity originating in the human body. The activity contains medically useful information if presented effectively to the correct people. Among the most useful methods of repre-

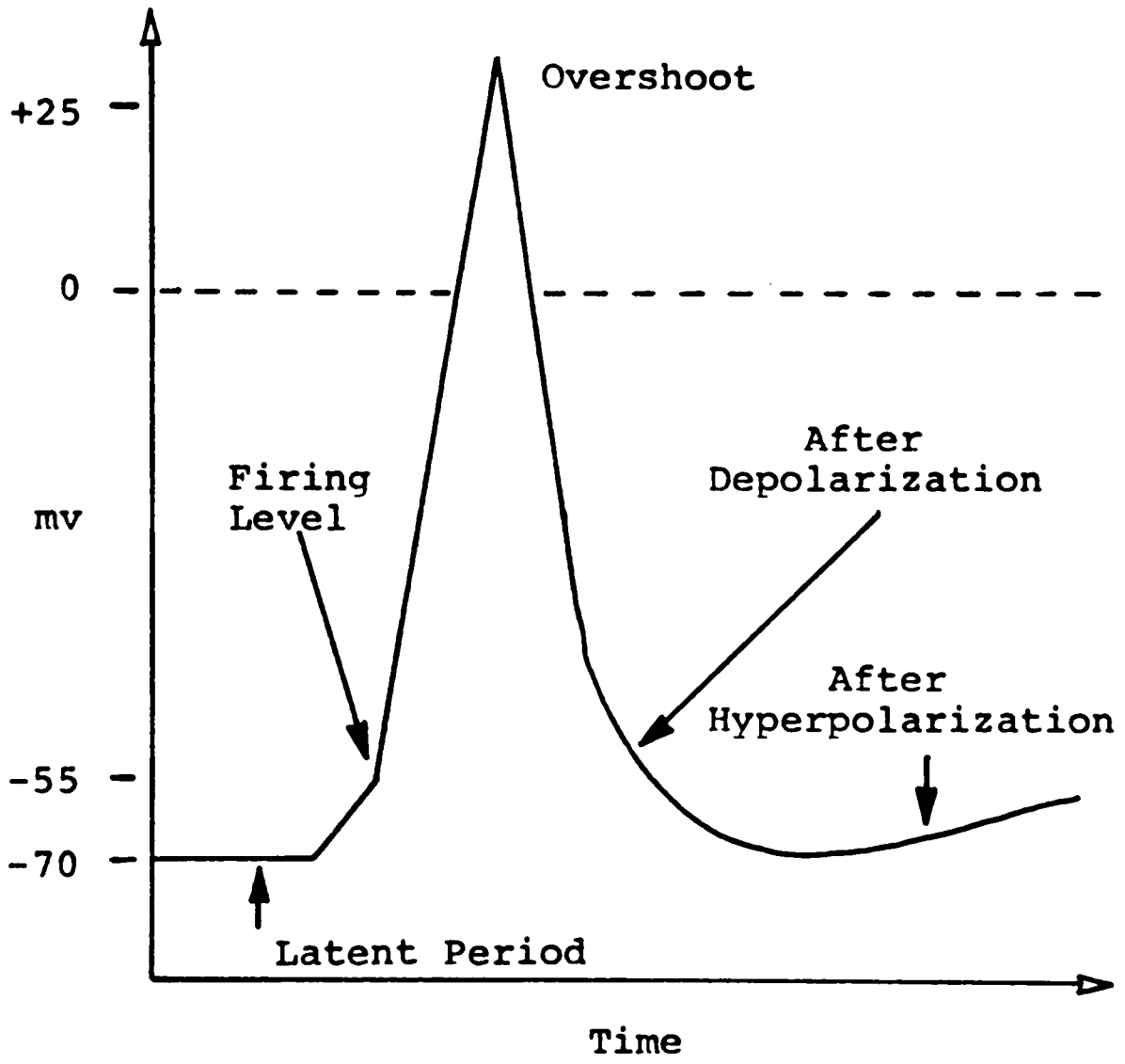


Figure 1. Single Cell Depolarization.

sentation of any type of electrical signal is its spectral content. There are techniques, such as Fourier analysis, by which any electrical waveform, meeting certain fundamental criteria, may be represented by its content of various frequency components. The representation can be in terms of frequency versus voltage amplitudes or frequency versus power to some unit load (Figure 2). Such representations of the electrical activity of the body allow catagorizations and initiate effective interpretation.

The maximum frequency of most physiological signals is less that 1 kiloHertz, excluding only some muscle potentials and intracellular potentials, which extend to 10 kiloHertz. A great majority of the clinically useful information lies in the frequency range between zero and 250 Hertz. Most cardiovascular, respiratory, brain, and muscular potentials are included in this range.

This work consisted of the design of a power spectrum analyzer, which was intended to provide a real time, clinically useful display of specified physiological electrical activity. Real time analysis is desirable from the standpoint of clinical applicability. Various forms of filtering, both linear (passive and active) and digital were discarded in favor of a more exact real time Fourier analysis. Data are processed digitally through the

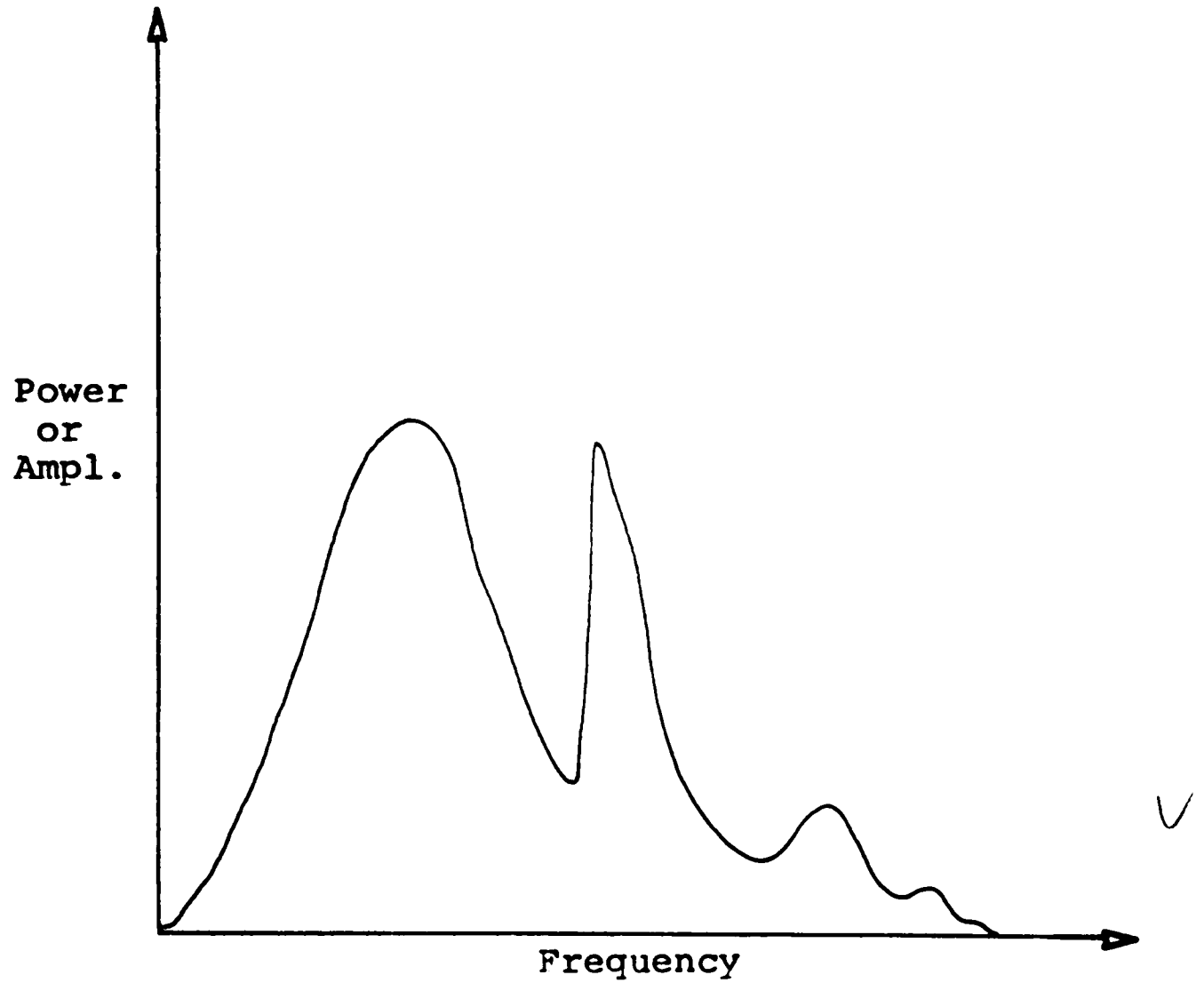


Figure 2. Typical Spectrum Example.

analyzer and then additionally manipulated by a Hewlett-Packard 2100A minicomputer (HP 2100) for display on a C.R.T. graphics terminal. The frequency range of the spectrum analyzer is between 1 Hertz and 256 Hertz with a resolution of 1 Hertz.

CHAPTER II

BASIC THEORY

There are several Fourier based spectrum analysis techniques. Specific criteria such as continuity and finiteness are necessary before a Fourier series or Fourier integral may be evaluated [2]. The machine presented here is a real time numerical calculation of the Z-transform for a given sequence of data, which is called a discrete Fourier transform (DFT) [3]. Adherence to the conditions imposed by the definition of Z-transform theory will be explained in the following paragraphs. Digital integrated circuits which have been announced allow the real time calculation of the DFT without the use of Fast-Fourier techniques. MOS storage and a fast bipolar arithmetic array are utilized to make the scheme feasable.

A. Numerical Calculation of Z-Transform

The basis of operation is the following definition of Z-transform:

> Let X(k) be a digital signal (more rigorously, an exponential ordered sequence) [4] that is zero for k<0. The Z-transform, X^*, of X is defined to be the following function of Z:

$$X^*(Z) = X(0) + X(1)Z^{-1} + X(2)Z^{-2} + \ldots$$
$$= \sum_{n=0}^{\infty} X(n)Z^{-n}$$

Z is defined in this case to be e^s; where $s=j\omega$. When Z is on the unit circle in the complex Z-plane, its angle is interpreted as a frequency variable [3]. Therefore the Z-transform of a sequence or digital signal produced by the sampling of some continuous input waveform may be represented in the complex Z-plane on the unit circle. This representation has a frequency related interpretation (Figure 3).

The numerical evaluation of Z-transform at a specific point on the unit circle in the complex Z-plane produces the transform value at the related frequency. Thus a total evaluation around the unit circle would produce a series of magnitudes at corresponding location-oriented frequences. The magnitude of the Z-transform versus frequency is precisely a frequency-magnitude spectrum.

Placing a finite limit on the sequence of data implies a finite record of data and a finite transform. Only a specified sample interval will be considered and the input signal will be assumed to be equal to zero at every other point in time. This will produce a modified definition of the Z-transform which more closely meets the objectives here.

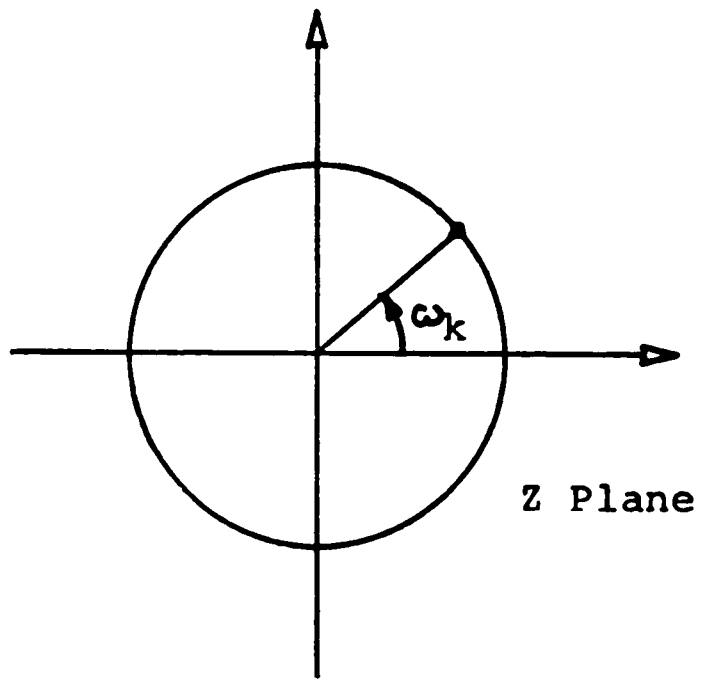


Figure 3. Complex Z Plane

The modified Z-transform of X(k) can then be defined as

$$X^*(k) = A \sum_{n=0}^{N-1} X(n) Z^{-n} \qquad k = 0,1,\ldots,N-1$$

where A is a scaling factor related to the sampling interval, which is itself related to the Nyquist frequency. k represents a specific point along the unit circle and is related to frequency by

$$\omega = \frac{2\pi k}{N} ,$$

where N is the number of samples available. The sampling rate, S, is related to the separation between points, Δk, and N by the relation,

$$\Delta k = \frac{S}{N} .$$

This implies that for 1 Hertz resolution the sampling rate must equal the value of N. Also, since samples can not be used before they appear in real time, 1 Hertz is the smallest frequency increment available for real time analysis.

Substituting for Z and n in the modified Z-transform definition;

$$X^*(k) = A \sum_{n=0}^{N-1} X(n)\, e^{-j \frac{nk2\pi}{N}} .$$

Applying the identity

$$e^{j\theta} = \cos\theta + j \sin\theta,$$

the final working definition for implementation becomes

$$X^*(k) = A \sum_{n=0}^{N-1} X(n) \left(\cos \frac{nk2\pi}{N} + j \sin \frac{nk2\pi}{N}\right.$$

The real and imaginary parts are, respectively,

$$\text{Re} \quad X^*(k) = A \sum_{n=0}^{N-1} X(n) \cos \frac{nk2\pi}{N},$$

$$\text{Im} \quad X^*(k) = A \sum_{n=0}^{N-1} X(n) \sin \frac{nk2\pi}{N}.$$

The sampling rate and number of samples (sample duration) is a function of the device speeds of the components used to implement the calculation, which will be discussed below. Once a sampling rate has been chosen, the frequency range of the available spectrum with 1 Hertz resolution is established by the Nyquist value theorem to be one-half the maximum sampling frequency. Therefore a sampling rate of $\frac{1}{N}$, utilizing N samples in the calculation, will produce a 1 Hertz resolution over a frequency range of zero to $\frac{N}{2}$ Hertz.

C. Basic Block Diagram

A basic block diagram of machine organization is shown in Figure 4. The basis of machine feasibility and

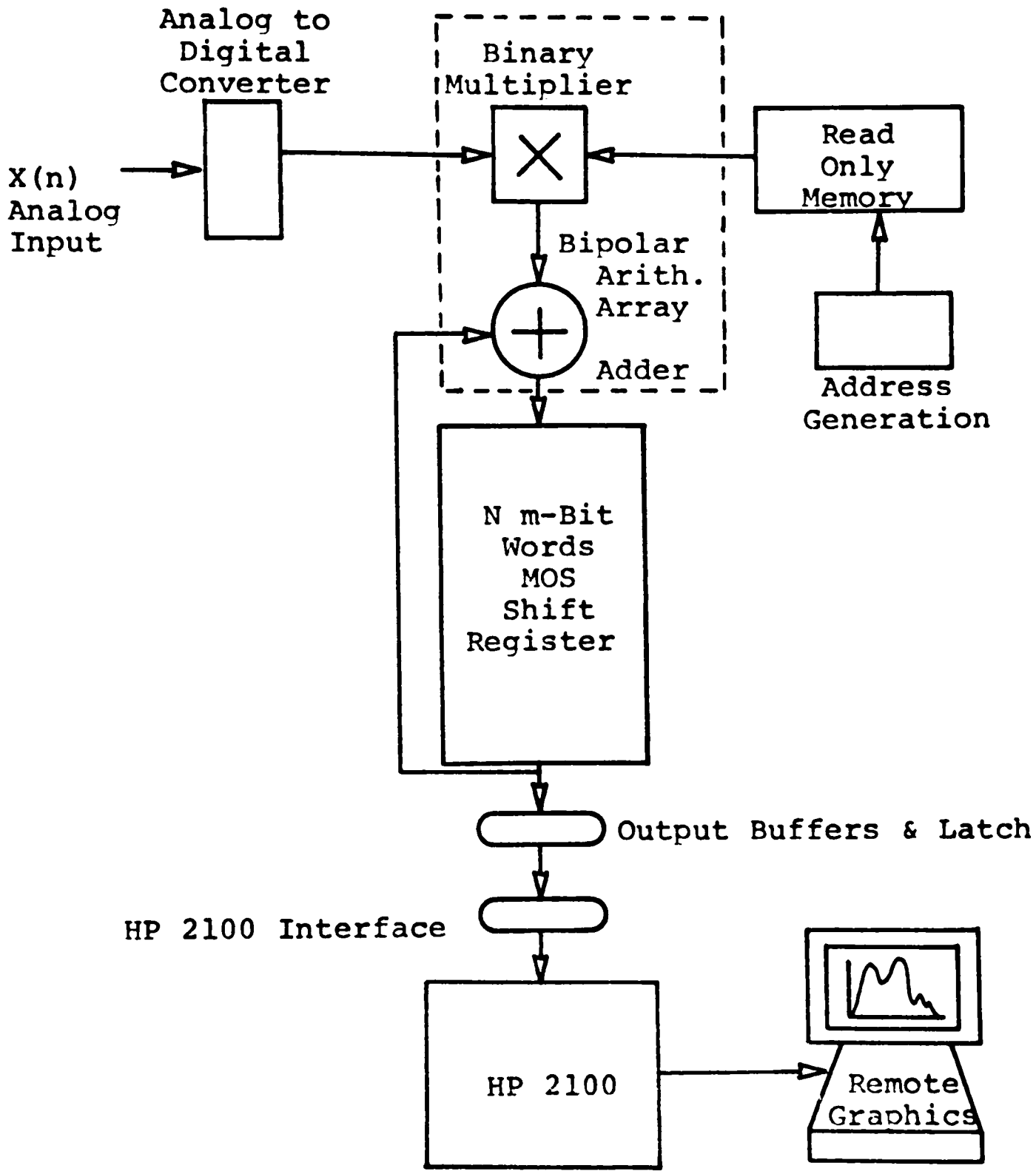


Figure 4. Basic Block Diagram.

implementation are two MOS memories and a fast bipolar arithmetic unit. One of the two memories is a read-only-memory (ROM) in which the values of sin $\frac{2\pi nk}{N}$ and cos $\frac{2\pi nk}{N}$ are stored. The other memory is a circulating one in which N m-bit words are stored. As the memory is circulated the words appear sequentially and in parallel format at the output. The output of the circulating memory is always applied to the adder in the arithmetic unit and a new result is stored back into memory. The structure will perform a DFT calculation if X(n) cos $\frac{nk2\pi}{N}$ is added to the $2n^{th}$ register, and X(n) sin $\frac{nk2\pi}{N}$ is added to the $(2n + 1)^{th}$ register, upon the n^{th} circulation[5]. Sucessive pairs of registers will have the real and imaginary portions of the DFT for the N point sequence f(n) shown in Figure 5. After N circulations of the circulating storage, the output is taken sequentially and in parallel format to an HP 2100 minicomputer where the real and imaginary portions are combined and displayed.

C. Limitation Criteria

The speed at which the DFT can be performed, and, in turn, the frequency range limitations at 1 Hertz resolution, is dependent on two factors. The first is the

10 x 10 Bit MOS Shift Register (Example)

Unit Sample Rate = 10 Hertz

Sampled Value ▽ Cosine Value ▽

Machine Capacitively Coupled No D.C.

$F(0)=f(0)c(0)+f(1)c(0)+f(2)c(0)+f(3)c(0)+f(4)c(0).....$

$F(1)=f(0)c(0)+f(1)c(1)+f(2)c(2)+f(3)c(3)+f(4)c(4).....$

Valid Results

$F(2)=f(0)c(0)+f(1)c(2)+f(2)c(4)+f(3)c(6)+f(4)c(8).....$

$F(3)=f(0)c(0)+f(1)c(3)+f(2)c(6)+f(3)c(9)+f(4)c(2).....$

$F(4)=f(0)c(0)+f(1)c(4)+f(2)c(8)+f(3)c(2)+f(4)c(6).....$

$F(5)=f(0)c(0)+f(1)c(5)+f(2)c(0)+f(3)c(5)+f(4)c(0).....$

Results Above 5 Hertz are Erroneous (Nyquist Theorem)

$F(6)=f(0)c(0)+f(1)c(6)+f(2)c(2)+f(3)c(8)+f(4)c(4).....$

$F(7)=f(0)c(0)+f(1)c(7)+f(2)c(4)+f(3)c(1)+f(4)c(8).....$

$F(8)=f(0)c(0)+f(1)c(8)+f(2)c(6)+f(3)c(4)+f(4)c(2).....$

$F(9)=f(0)c(0)+f(1)c(9)+f(2)c(8)+f(3)c(7)+f(4)c(6).....$

▽ Spectrum Value

Shift Register Contents After One Circulation

Shift Register Contents After Two Circulations

Shift Register Contents After Three Circulations

.
.
.

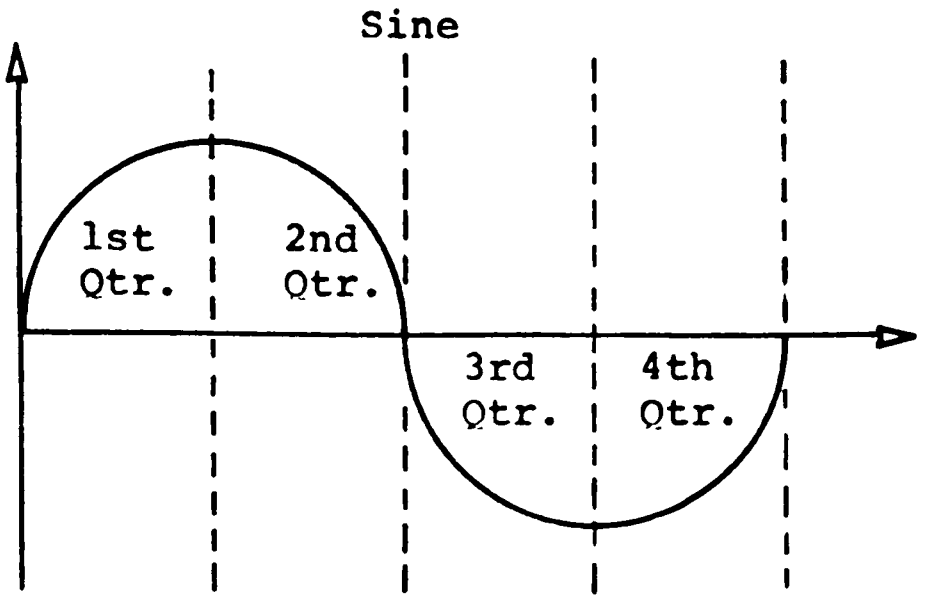
Real (Cosine) And Imaginary (Sine) Would Alternate In Positions

Under Actual Conditions

Fig. 5. Circulating Memory Contents

access speed of the two memories. The second is the speed at which the arithmetic operations (multiplication of X(n) by the appropriate ROM address contents, and addition into the circulating storage) can be completed. Components used in implementation had memory access speed of 1μsec and an arithmetic speed of 0.5μsec.

The number of ROM's required was reduced by taking advantage of the quarter-symmetry of both sine and cosine magnitudes (Figure 6). With correct address generation, only a positive and negative quarter of a sine wave requires storage and the resultant sine and cosine multiplication terms can be accessed as needed.



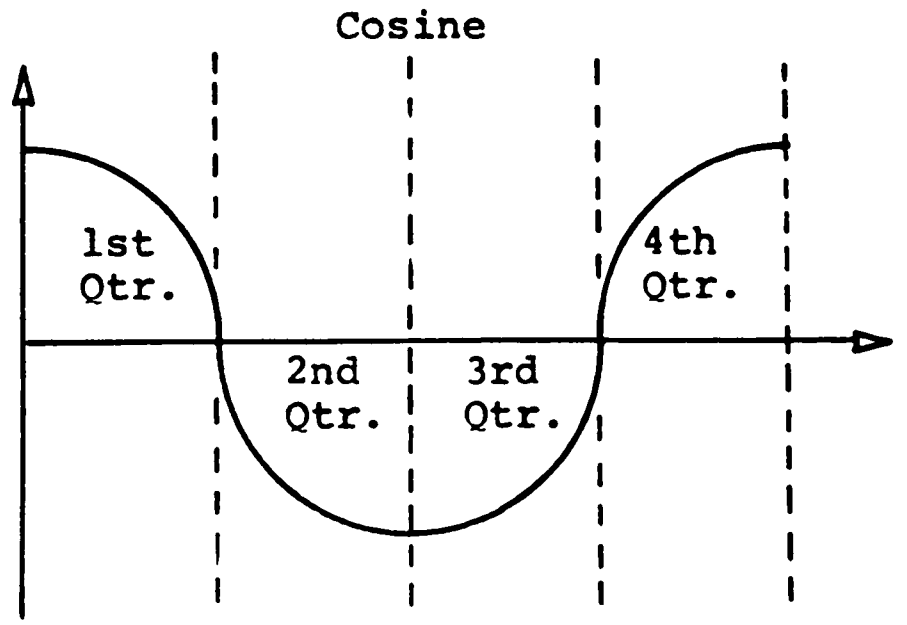


Figure 6. Quarter Symmetry Example.

CHAPTER III

DESIGN CONSIDERATIONS

The basic operation theory as presented in the preceeding chapter is sufficiently simple that theoretical verification was not necessary for the overall digital implementation approach. However, the compromises and considerations necessary in the physical implementation of even this simple theory were very involved. There were obvious choices, such as frequency range relative to cost and pertinent information yield. Other considerations were not so obvious, such as the use of the HP 2100 versus another arithmetic unit and the choice between integer and floating point arithmetic, relative to error acceptability and cost. One very important consideration was the advantages of real time (approximate) spectrum as opposed to a more delayed generation of the same spectrum using a Fast-Fourier algorithm on available digital computers. Also availability and adaptability of state-of-the-art digital integrated circuits, both bipolar and MOS, had to be considered.

A. Frequency Range

The frequency range of the spectrum would have been ideal if it could have extended from d.c. to 10,000 Hertz. This range includes the frequency components of all physiological signals in the human body, as illustrated in Table 1. A resolution of 1 Hertz was the most important constraint and therefore device speeds became a significant limiting factor. As discussed in the preceding chapters, the device speeds limited the number of calculations possible over a given sample interval. Since each calculation corresponded to a 1 Hertz increment over the preceding calculation, the number of calculations equalled the frequency range. With very critical timing tolerances and some intricate timing coordination among device sections, the maximum frequency range of the chosen configuration was approximately 300 Hertz. Various hybird designs utilizing various speed-up organizational schemes can greatly improve this range, at an economic sacrifice, as will be discussed in Chapter VI. This range is based on a 1 μsec accessability time of the ROM and a 0.5 μsec time of the arithmetic unit. Further consideration of the frequency components in physiological signals indicated that most information of a majority of the responses was less than the 300 Hertz maximum.

TABLE I

TABLE OF PHYSIOLOGICAL FREQUENCY SPECTRUM COMPONENTS

Physiological Parameter	Measurement Required	Spectrum Components
<u>Circulatory System</u>		
Heart Potentials	Electrocardiogram	F(ω) Hertz
Blood Pressure	Arterial & Venous Pressure	.05-80
Blood Flow	Peripheral Flow	DC-500
Blood Volume	Blood Volume	DC-50
		DC-50
<u>Respiratory System</u>		
Breathing	Pneumogram	.05-2
Respiratory Flow	Pneumotachogram	DC-2
Respiratory Volume	Spirogram	DC-1
<u>Brain Functions</u>		
Electrical Activity	Electroenceph Alogram	.5-100
Evoked Responses	Intracellular & Extracellular	1-10.000
Eye Responses	Electroretinogram	.05-20
Brain Midline Position	Ultrasonic Echo	(Graph)
<u>Muscular Functions</u>		
Muscle Excitability	S-D Curve	(Graph)
Muscle Strength	Myogram	DC-50
Muscle Potentials	Electromyogram	10-5000
Nerve Conduction	H-Reflex Response	10-5000
Smooth Muscle Activity	Electrogastrogram	.05-2

The absolute maximum of a floating point arithmetic machine is less than 200 Hertz. The complexity of a floating point machine is considerably greater than an integer machine; however, accuracy is also considerably greater. The increased complexity is the result of the inclusion of an additional adder and shift registers to adjust and align decimal points and keep track of exponents. These additional operations also reduce the arithmetic unit speed and as a result reduce overall frequency range, as previously explained. The increased complexity and device count also substantially increase the cost of a floating point machine over the cost of an integer machine.

B. Accuracy

Computer simulations of floating point machines were carried out on an IBM 370 digital computer utilizing Fortran IV language under an OS-370 operating system. Various simulations were run according to the degree of accuracy of the input analog to digital converter and the number of digital bits carried throughout the machine. The input waveform was simulated by a composite addition of 20 cosines of different frequencies, phases, and amplitudes as shown in Tabel 2. The simulation of a machine having 16 bits of input signal and 16 bits of sine and cosine terms produced ex-

TABLE II

TABLE OF WAVEFORM SIMULATION COMPONENTS

Cosine Number	Amplitude	Frequency	Phase
1	4.0	1.0	0.25
2	5.0	2.0	1.25
3	1.0	3.0	1.25
4	2.0	4.0	3.12
5	3.0	21.0	5.11
6	5.0	22.0	0.24
7	5.0	23.0	5.77
8	2.0	24.0	0.88
9	3.0	25.0	0.86
10	3.0	47.0	0.08
11	5.0	55.0	0.95
12	1.0	56.0	250.2
13	1.0	58.0	3.00
14	2.0	60.0	3.14
15	2.0	99.0	1.88
16	5.0	100.00	1.53
17	4.0	147.00	2.74
18	1.0	149.00	0.12
19	3.0	153.00	13.22
20	2.0	205.00	98.14

tremely accurate results with errors of less than one percent. The simulation of a machine having 8 bits of input and sine-cosine produced errors of 5 to 7 percent. The simulations demonstrated the accuracy possibilities of a floating point arithmetic machine. Note that the high degree of accuracy also implies great complexity and higher cost relative to an integer machine. Errors due to truncation in the 5 to 7 percent range are entirely acceptable because of the machine application. In physiological studies absolute accuracy is often less important than relative information. In other words, an electrophysiological spectrum with an error which is not too large will provide as much useful information as a very accurate spectrum. The simulations verify accuracy possibility; however, accuracy may not be the most important criterion.

Computer simulation of an integer arithmetic machine of comparable word length was also done on the IBM 370, and several advantages of integer over floating-point arithmetic were found. Integer arithmetic yields very nicely to binary application when calculations are related to powers of two. Also, there are no decimal points to keep track of, therefore machine complexity is greatly reduced. If all numbers are in two's complement form, the arithmetic operations are even further simplified as a result of numbers

being in the correct form for addition and subtraction. The reduction in machine complexity increases machine speed such that an upper frequency of 300 Hertz can be obtained. The details of two's complement integer arithmetic will be discussed in Chapter IV.

The errors were considerably larger than the original 16 bit-floating-point-arithmetic machine; however, the result was a relative, rather than an absolute, spectrum. Also, the integer machine had an greater frequency range at less than one-half the cost of a floating-point-machine.

C. Economic Considerations

Economic Considerations were of great importance in the final design choices of this machine. There was a weighting of priorities relative to the useful information output possible and costs. The choices essentially were either an expensive, extremely accurate floating-point-arithmetic machine with a small frequency range, or a less expensive, less accurate integer-arithmetic machine with a greater frequency range. Under the criteria set by the application of this particular spectrum analyzer, the integer machine approach was chosen for implementation. The less accurate, relative spectrum, over an extended range, with corresponding reductions in cost and complexity, was the final choice.

D. The Role of the HP 2100

The use of the HP 2100 minicomputer was considered relative to cost and procedure. At the beginning of the work, display of the resultant spectrum was specified to be on a remote graphics terminal. This automatically involved the HP 2100, software, and interface hardware. Since the HP 2100 was involved by specification, there were other advantages to its use. The output of the circulating memory consists of consecutive pairs of real and imaginary portions of spectrum values. To create a frequency versus amplitude spectrum, the real and imaginary values had to be squared, added, and the square root of the sum taken. For a power versus amplitude spectrum the real and imaginary values had to be squared and added. (Power here refers to the square of the voltage divided by a unit load. Absolute power-measurements are not possible because of nature of the source and the nonlinear and variable nature of tissue impedances.) These mathematical operations could be built in hardware inside the device, substantially increasing its complexity and cost. The calculations could be quickly performed with the HP 2100 before display with only slight delay (one to two seconds) introduced. The obvious decision was to include a software program routine in the HP 2100 to perform the final mathematical operations

on the output of the circulating memory contents.

E. Advantages of Real Time Analysis

The only acceptable alternative to the procedure developed in this work is the Fast-Fourier Transform. However, the use of this technique requires a 20 to 40 second delay in the presentation of data to the observer. A delay of this magnitude introduces the possibility that significant effects will not be presented until after a physiological event has occurred, or worse, that important data will have been missed during the processing period. Real-time analysis (at most one second) eliminates these possibilities, and furthermore, permits the use of averaging techniques to eliminate noise and other non-repetitive error signals.

F. Availability and Adaptability of Digital Integrated Circuits

The availability and adaptability of digital integrated circuits was also a point of consideration and compromise. Access time of the MOS ROM was one of the key limiting factors in the operation speed of the device. There are memories available which may be accessed at a 2 to 3 megaHertz rate: however, they involve mask programming procedures and therefore great expense. The compromise was a product line made by the Intel Corperation which provides electrically

programable ROMS. The cost was two orders of magnitude less. The sacrifice in speed reduced access time to 1 μsec and limited the maximum frequency range to those values previously discussed. Speed and complexity of the arithmetic unit was another range limiting factor. Discrete integrated circuit implementation of binary multiplication is too slow and extremely complex. A new device has recently been announced by Texas Instruments Incorporated: a 4 bit by 4 bit binary multiplier, which can be combined with fast-look-ahead-carry adders to produce mxn multipliers.

G. Final Block Diagram

The final design compromises resulted in the block diagram shown in Figure 7. The frequency range was between 1 Hertz to 256 Hertz. The range was adequate for the application relative to the constraints.

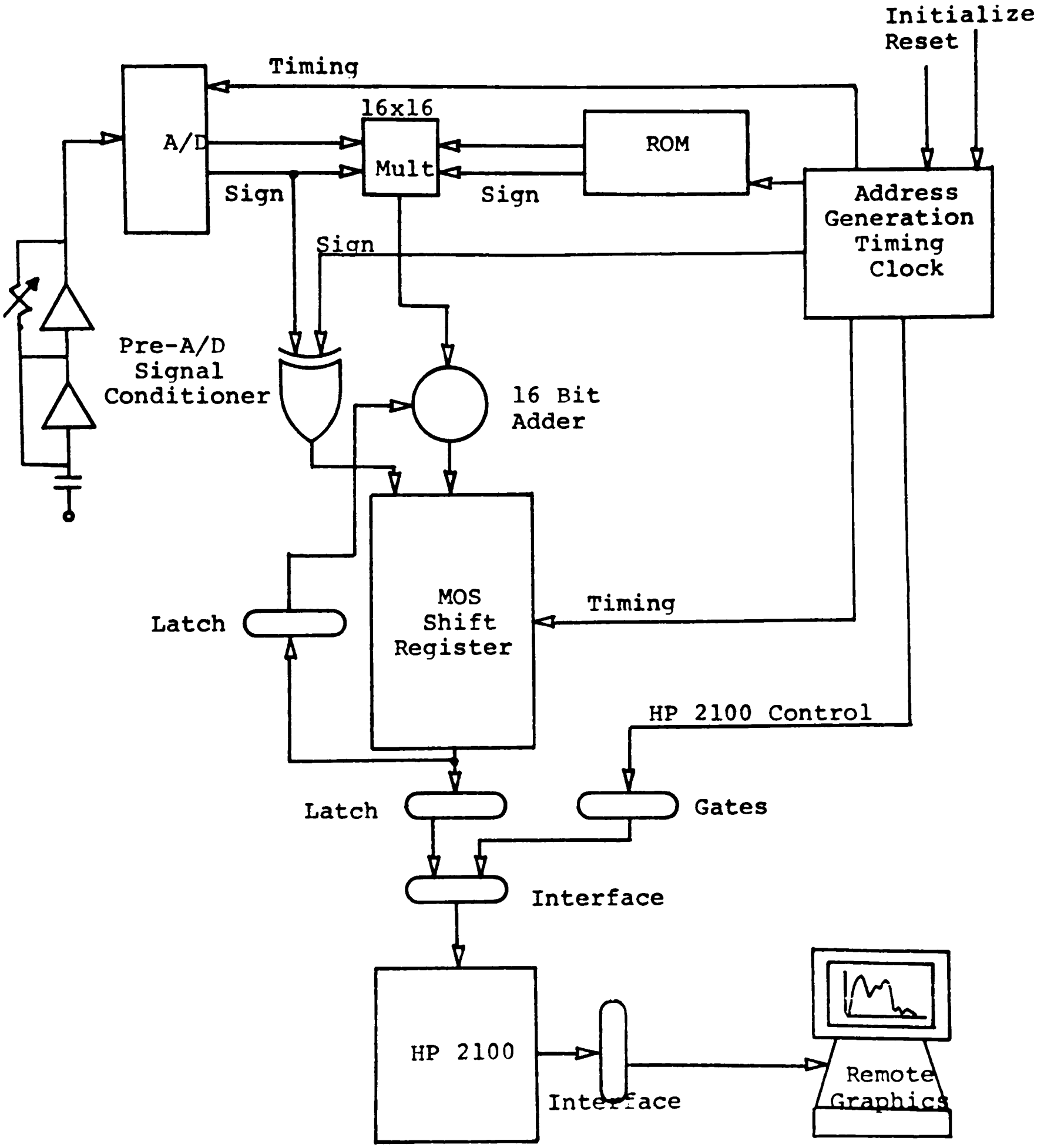


Figure 7. Final Block Diagram

CHAPTER IV

DETAILED SYSTEM DESIGN

The basic operation theory, constrained by the design considerations of Chapter III, resulted in a final system design[5] (Figure 7). The device may be divided into six sections for purposes of discussion; these six sections are shown in Figure 8. Each of the first five individual sections was mounted on a separate circuit board and appropriate interconnections were made between them. Section six was the HP 2100 minicomputer, which performs the final arithmetic functions prior to graphic display. In this chapter, detailed logic diagrams and operation will be discussed for each of the five hardware sections. The role of the HP 2100 relative to general software and remote graphics display will also be discussed. Layout information, power supply requirements, final parts list, and device orientation will be discussed in the latter part of the chapter.

A. Timing Generation

The detailed logic diagram of the timing generation section is shown in Figure 9. (For medium-scale and large-

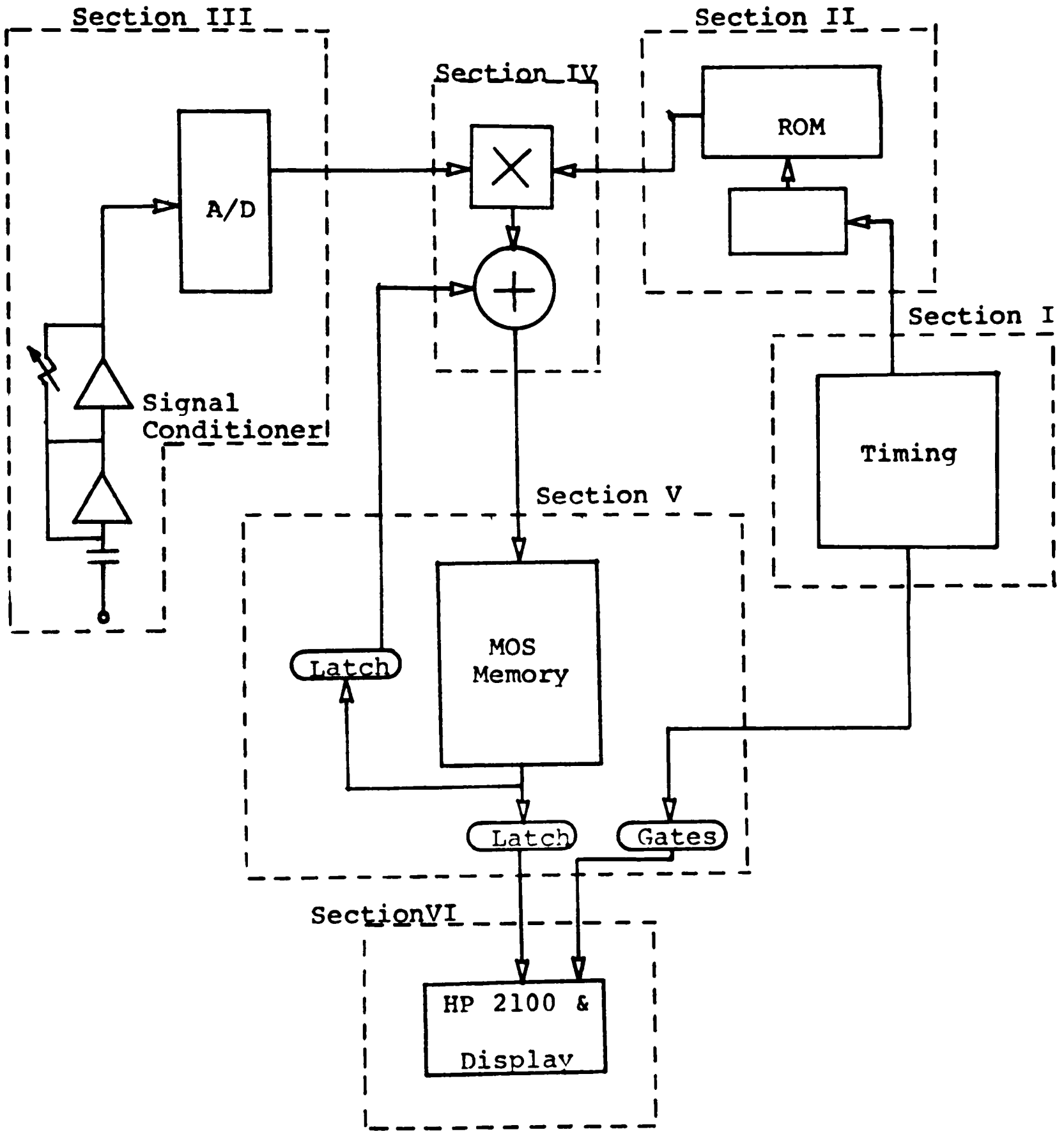


Figure 8. Sectioned Block Diagram.

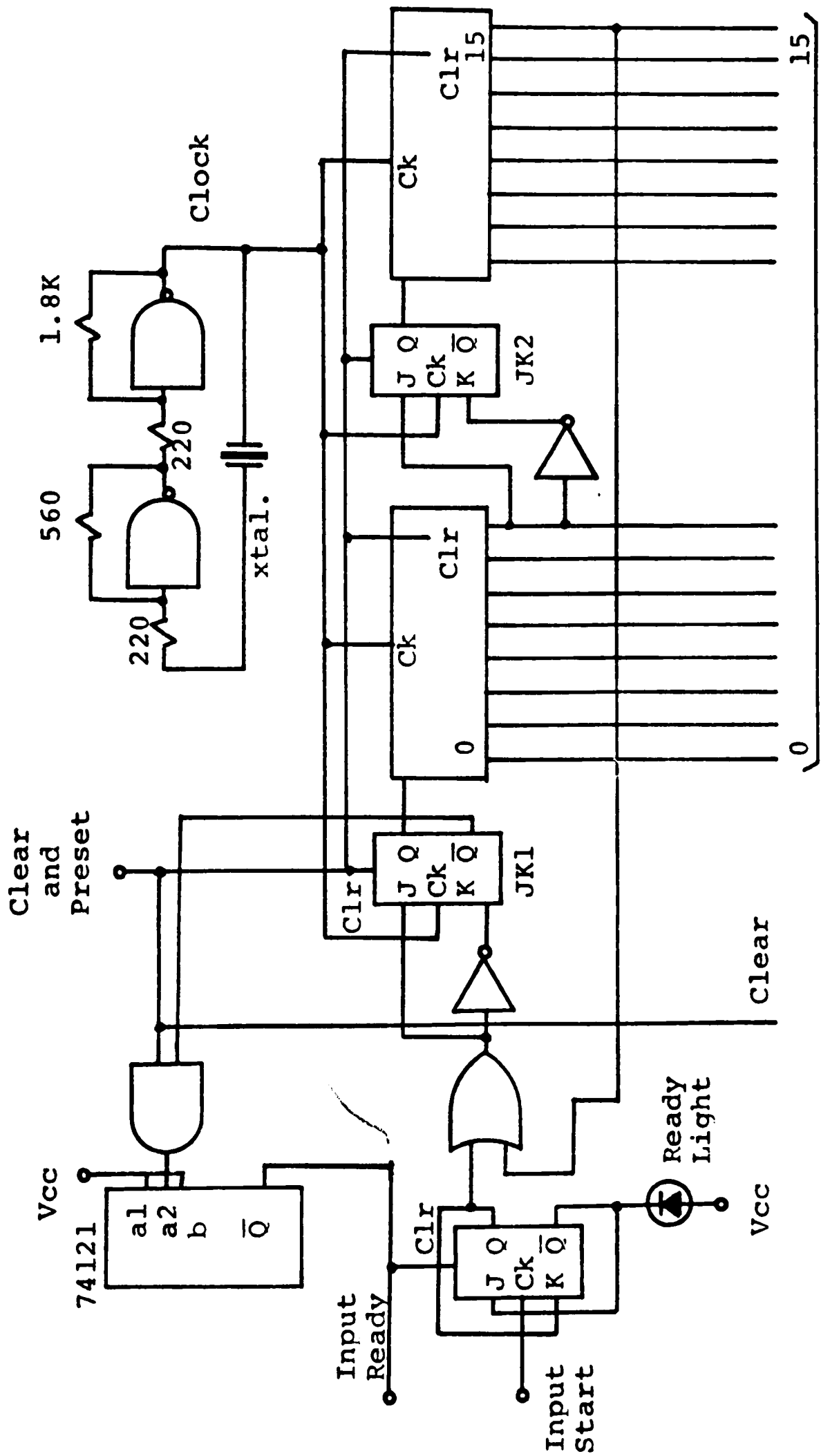


Figure 9(a). Clock Division Of Timing Section.

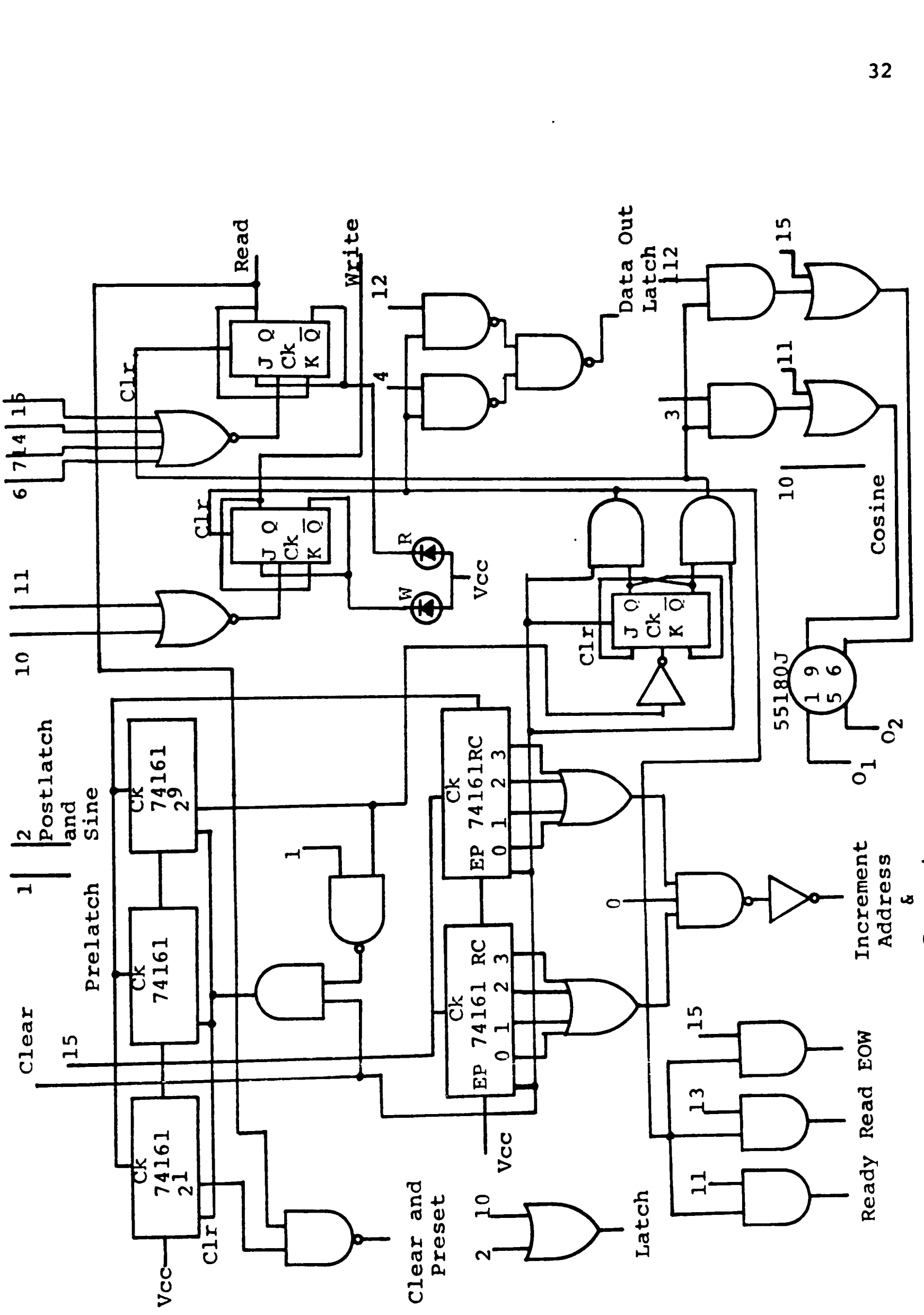


Figure 9(b). Timing Logic O

scale integrated circuits, complete gate-by-gate detail is omitted in favor of an appropriately labeled functional block.) The device is basically sequential in operation. There is a specific order in which events must occur, as in most large digital devices. The overall system timing diagram presented in Figure 10 was divided into two sections:

(1) calculation cycle;

(2) read-out cycle.

The elaboration of the timing particulars follows from the discussion of the individual sections.

The timing was controlled by a clock made up of two nand gates, four resistors and a crystal oscillator. The basic clock frequency was 2.097152 megaHertz, which was followed be a divide by 16 circuit, consisting of a 16 bit, serial in-parallel out, shift register. The clock frequency value was obtained by multiplying the sampling rate (512 Hertz) by 256 real and imaginary pairs per storage shift register circulation, and by multiplying the product by the 16 positions of the divide-by-circuit. During each shift register circulation a control signal was obtained from most of the 16 parallel outputs. Since the 16 always occur in the same sequential order, the signals controlled the device circuity required to perform the calculation. The period of each output pulse of the 16

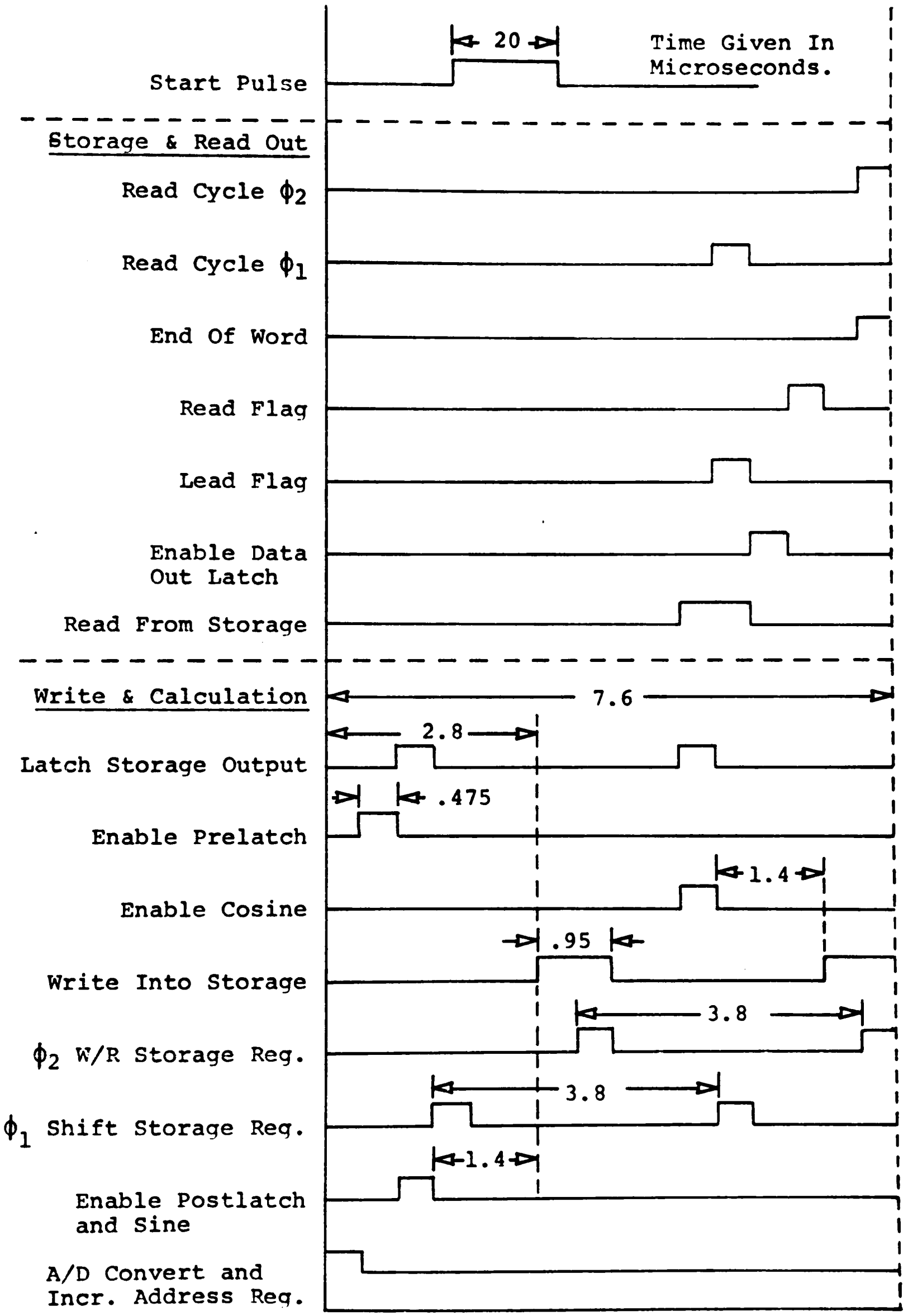


Figure 10. Basic Timing Diagram.

bit shift register was .475 microsecond and well above the minimum clock duration required by inner device integrated circuits. The total duration of one shift register circulation was 7.6 microseconds. For each sampled value, the circulating memory was shifted and had information entered into it 512 times (256 real and imaginary pairs). A real and imaginary pair was generated each 7.6 microseconds.

Two counters were in the timing section. One counter counted from 0 to 256 and incremented the address counter, controlling the introduction of a new sampled value. The other counter counted from 0 to 512 and counted the number of sampled values, signalling the end of the calculation cycle and the beginning of the read-out to the HP 2100.

The timing section generated read and write commands as well as ϕ_1 and ϕ_2 (non-overlapping clock pulses) for the MOS circulating memory. As shown in the timing diagram, the frequency of these control pulses was divided by two during the read-out cycle to the HP 2100, in order to allow the minicomputer sufficient time (7.6 microseconds) to read and store the output of the storage register. The ϕ_1 and ϕ_2 non-overlapping clock frequencies were applied to a TTL-MOS level converter in the timing generation section. The level conversion was a necessary interface between TTL and MOS devices. For a low logic TTL value of approximately

zero volts, a value of -9 volts was applied to the MOS devices. For a high logic TTL value of approximately +3.5 volts, a value of +5 volts was applied to the MOS devices.

Various other control signals were taken from the output of the 16 bit shift register. Three control signals were intended to aid in the control of the HP 2100. These control signals appeared at the device output only during the read-out cycle. The "ready" control line went high just prior to the "enable" control for the data output latch. This was followed by the "read" control and "end of word" control, both of which were high levels. Sine and cosine controls were both high levels and were taken from the shift register output. The address generation section had two latches which required sequential control in order to perform the correct addressing of the ROM. The two control signals for pre-address adder latch and post-address adder latch were outputs from the shift register of the timing section. A complete system clear signal, "clear and preset" was produced by the timing section after the contents of the storage register were read out to the HP 2100. This signal was applied to the device just prior to another sampling interval. (The total sampling interval was one second.)

During "clear and preset" the "increment address and A/D control" counter was preset to 255. At the outset of operation the first command was to increment this counter and therefore it began at zero. All other counter, J-K flip-flops, and shift registers of this section had their Q outputs preset to zero.

To initiate sampling, a TTL high level was applied to the clock input of the input-control J-K flip-flop. The device was enabled on the leading edge of this control signal. The input control J-K flip-flop was disabled once sampling had begun and would not respond to another input command until 10 microseconds after all data had been read out to the HP 2100. This procedure was implemented by the retriggerable monostable multivibrator with a 10 microsecond delay, receiving its pulses every 7.6 microseconds from the circulating 16 bit shift register. The "ready-input" control line went high when the device was ready for another input command. The "ready-input" and "input" lines could be joined for continuous operation, provided that the HP 2100 did not fall too far behind. Otherwise these two control lines were available for external control, providing exact coordination flexibility for the device relative to the specific interval for which examination was desired.

A complete timing sequence is listed in Table 3. The

TABLE III

TABLE OF SEQUENCE OF TIMING EVENTS

1. Manual Reset Or Power Up Or Startpulse Initiates Clear And Preset
2. Convert Command To A/D Converter And Increment Address Counter
3. Enable Pre-Address Adder Latch
4. Enable Post-Address Adder Latch And Sine Line And Latch Storage Output
5. Shift Storage Register (ϕ_1)
6. Write Command To Storage Shift Register
7. W/R (ϕ_2) To Storage Shift Register
8. Enable Cosine Line And Latch Storage Output
9. Shift Storage Register (ϕ_1)
10. Write Command To Storage Shift Register
11. W/R (ϕ_2) To Storage Shift Register
12. Loop 3-11 Is Repeated 512 Times Per Sample (512 Samples)
13. Completion Of Calculation Cycle Enables Read-Out Cycle Control Logic And Disables Write Cycle Control (ϕ_1&ϕ_2 Are Decreased To ½ Original Frequency)
14. Read Command To Storage Shift Register
15. "Ready" Flag To HP 2100
16. Output Data Latch Enabled
17. "Read" Flag To 2100 (Output Data Is Valid)
18. "End Of Word" Flag To HP 2100 (Output Data Subject To Change)
19. Loop 14-18 Is Repeated 512 Times
20. DFT Completion By 2100

events of one DFT calculation performed by the device are listed and numbered in order. The timing diagram for the 16 bit shift register is shown in Figure 11. Other particulars of the timing section include various control gates, inverters, and control J-K flip-flops whose functions are obvious in the context of the total detailed diagram. Light emitting diodes were joined to the $\bar{Q}$ outputs of three of the control flip-flops, to provide an external visual indication of the cycle of operation. The ready light remained on as long as the device was not in a actual calculation: the calculation cycle light remained on for one second, and the read-out cycle light merely blinked, because of the short duration involved. The timing section was the heart of the device and its precise operation was the key to correct calculations. The output of the circulating memory is an example of critical timing importance. The output must be latched and held while the correct entry location moves into position to accept the new value. This latch command was also provided by the timing section.

B. Read Only Memory Address Generation Logic

The MOS ROM contained the values of sin $(2\pi nk/N)$ and cosine $(2\pi nk/N)$ to be addressed and retrieved from memory. The value of nk/N was always less than or equal to one and

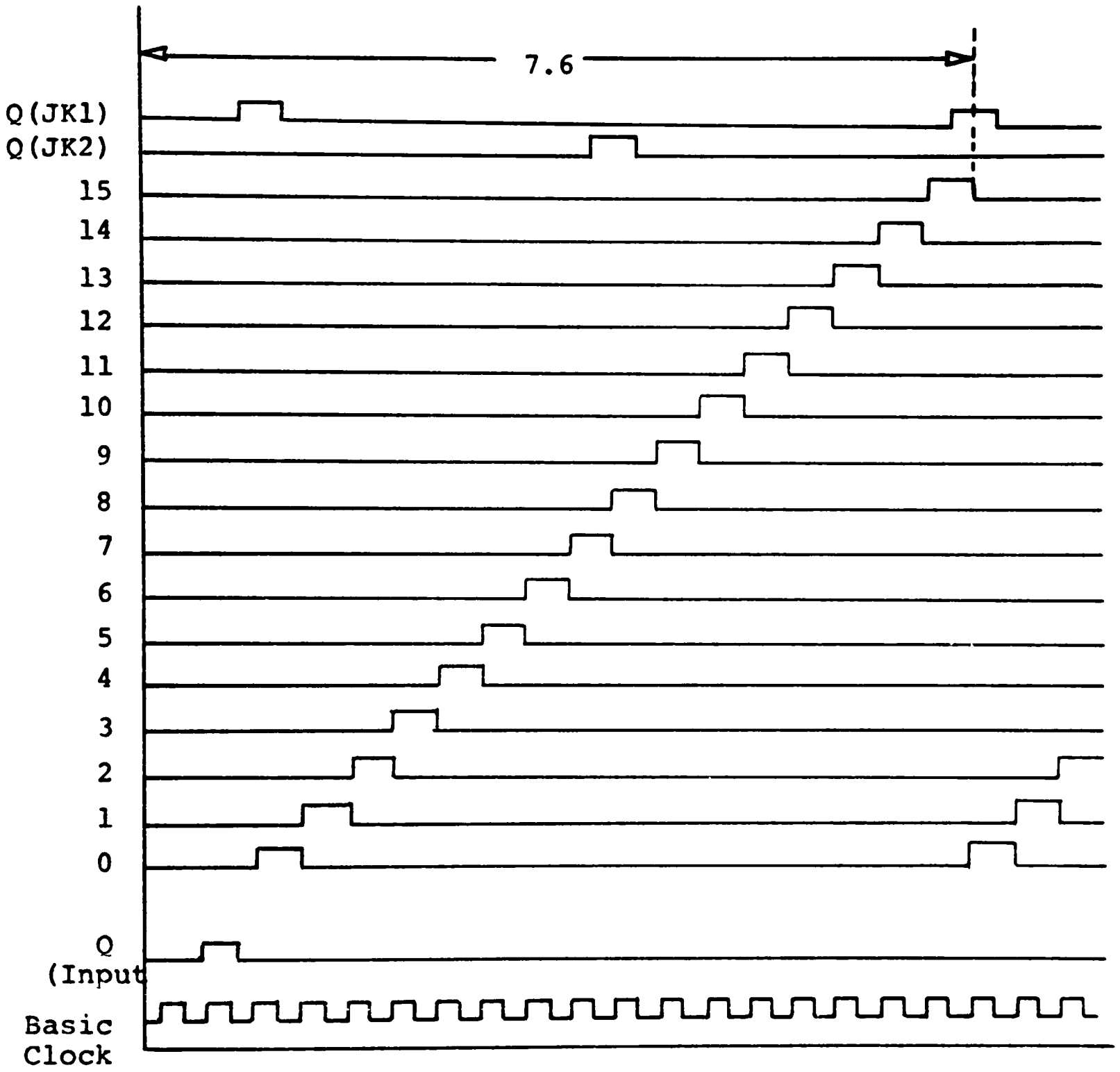
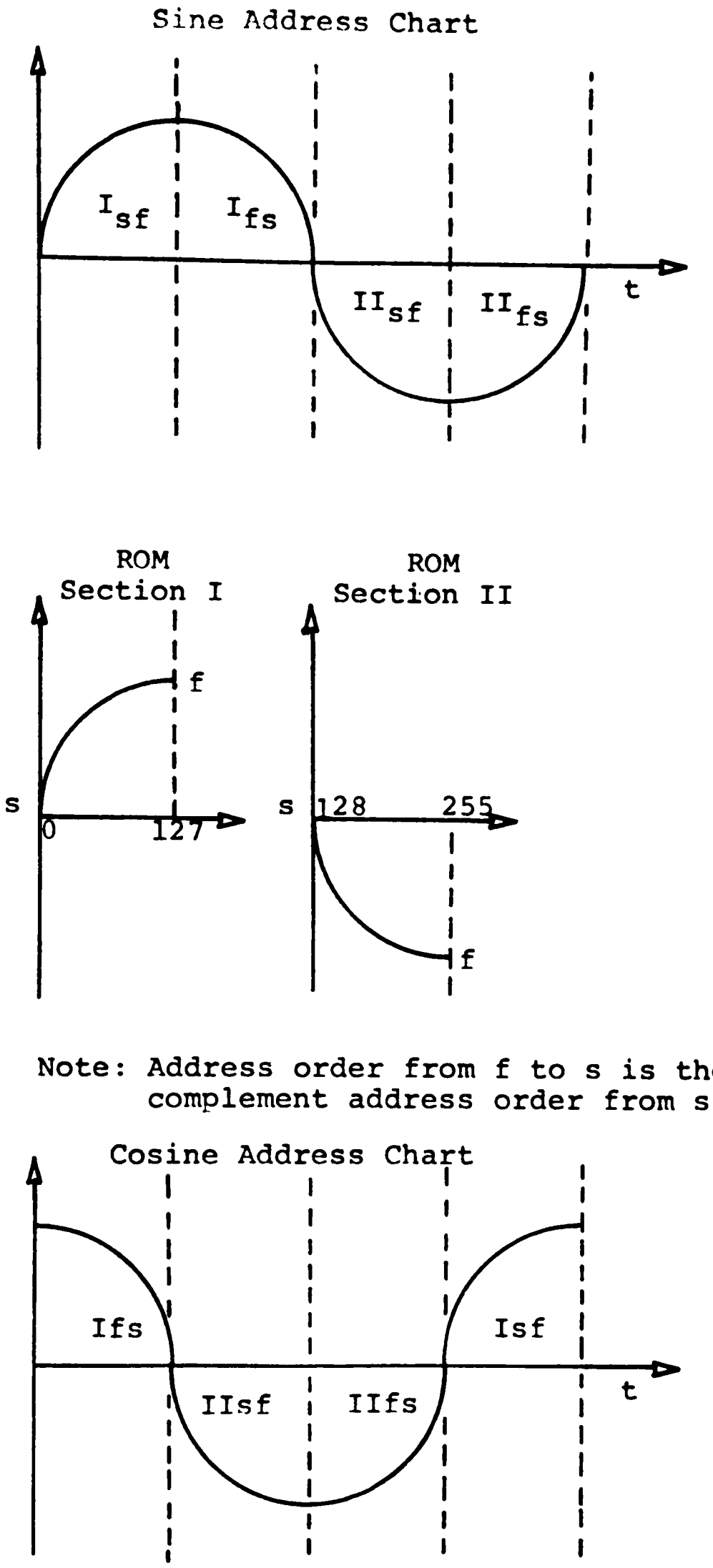


Figure 11. Sixteen Bit Shift Reg. Timing Diagram.

the absolute values of sine and cosine were always less than or equal to one. Since integer arithmetic was to be used, each sine and cosine was scaled by multiplication by 128 (2^8). This value was chosen as the scaling factor because, after multiplication and preceding addition, the least significant seven-bits were truncated, thereby dividing by 128. This scaling procedure permitted the use of integer arithmetic and was easy to manipulate.

The quarter symmetry of both the sine and cosine functions, as shown in Figure 6, allowed a great reduction in required storage space. Using a memory containing 256 eight-bit words, (Intel 1602-A) the address locations 0 to 127 contained sin ($2\pi n/512$), and the address locations 128 to 255 contained -sin ($2\pi n/512$), where n vaired between 0 and 127. The negative numbers were in 2's complement form for arithmetic operation convienience. These corresponded to the positive and negative values of one quarter of a sine wave in increments of 1/512. (512 Hertz was the sample rate of the analog-to-digital converter) As shown in Figure 12, the total sine and cosine terms necessary for the DFT calculation were constructed from the quarter wave values. The question was therefore one of addressing the MOS ROM in the correct sequence in order to retrieve the correct value of sine and cosine as required. It should be noted



Note: Address order from f to s is the complement address order from s to f.

Figure 12. Quarter Wave Construction Of Sine And Cosine.

that all eight bits of the ROM output were required to represent the magnitudes ranging from +128 to -128: therefore, the sign bit for follow-up operations was generated by the address generator and not derived from the ROM output. (Negative values are in 2's complement.)

Figure 12 indicates the construction of the sine and cosine values. The first half of sine wave consisted of the positive sine values addressed from 127 to 0. The second half of the sine wave consisted of the negative sine values addressed from 128 to 255, followed by the negative values addressed from 255 to 128. The most significant bit of the address is the sign bit (0 implying positive sine values; 1 implying negative sine values). Excluding the sign bit, the addresses applied to the ROM for addressing from 255 to 128 or 127 to 0 are simply the complements of the addresses applied when addressing from 128 to 255 or 0 to 127, respectively, therefore, the addressing logic produced the correct address for a given increment number by applying either a positive or a negative sign, and either the address or the complement of the address. The same reasoning applied to the construction of the cosine wave, the difference between the waves being the order of construction. The address applied relative to the address counter increment number depended on whether a sine

value or a cosine value was required.

Figure 13 shows the basic block diagram of the address generating section. The basic operation can be explained referring to the circulating memory contents and Figure 5. For the first sampled value of the input signal, addresses in increments of one were applied to the ROM according to the expression

$$X^{*}(k) = \sum_{n=0}^{N-1} X(n) \cos\left(\frac{nk2\pi}{N}\right) + j \sin\left(\frac{nk2\pi}{N}\right),$$

where k=0,...N-1. According to the previous algorithm, a sample number of two required that addresses were applied in increments of two, and so on. The counter counted from 0 to 512, the adder determined the increment rate, and the register was the buffer between the adder and ROM. The logic block controlled the section of the ROM addressed and also whether the true or complement address was applied; as the application of the true or complement address was determined by two inputs: sine or cosine from the timing generation block, and the most significant bits of the counter.

A complete detailed logic diagram of the address generation section is shown in Figure 14, and a pin diagram and other information respecting the ROM is shown in Figure

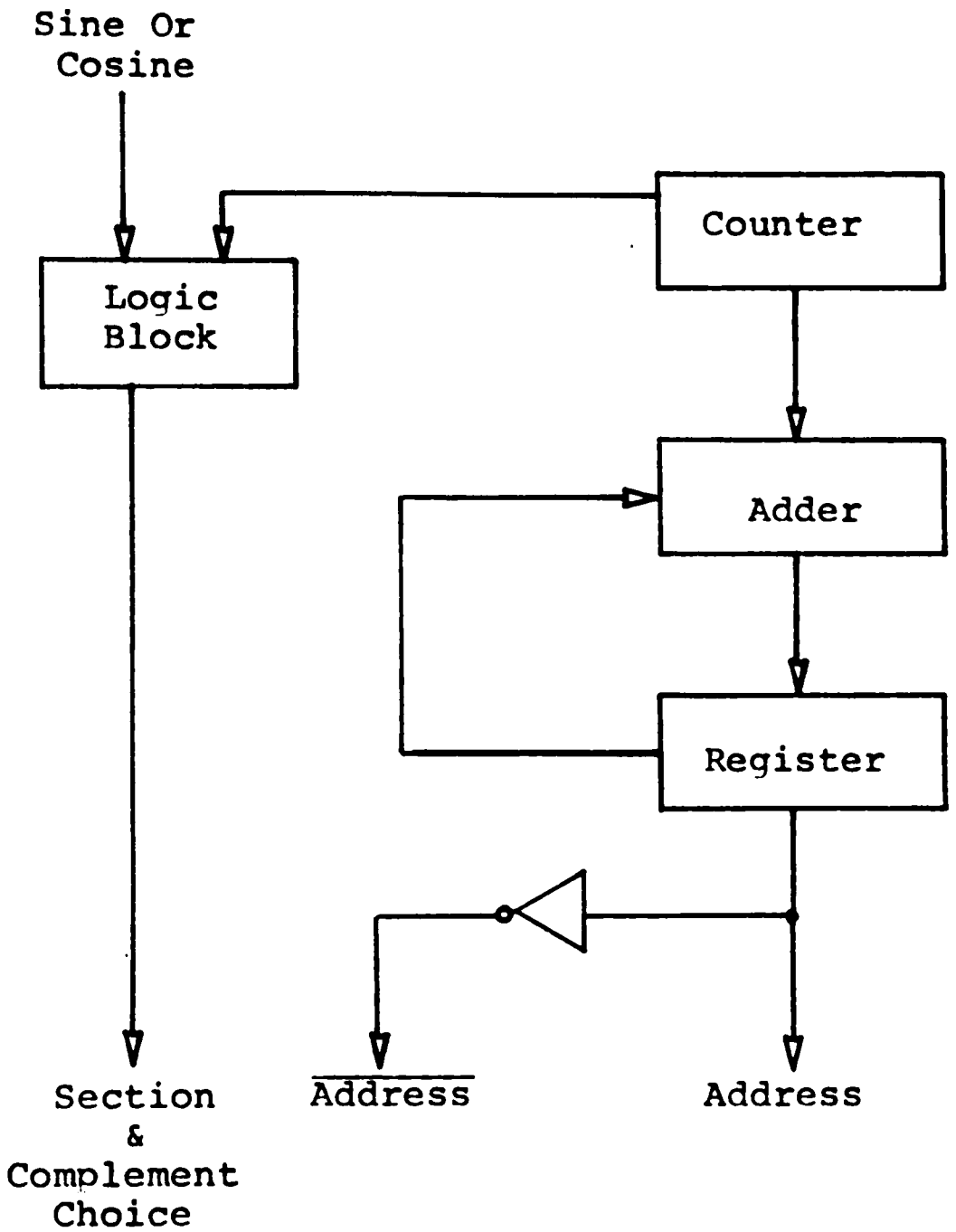


Figure 13. Address Unit Block Diagram.

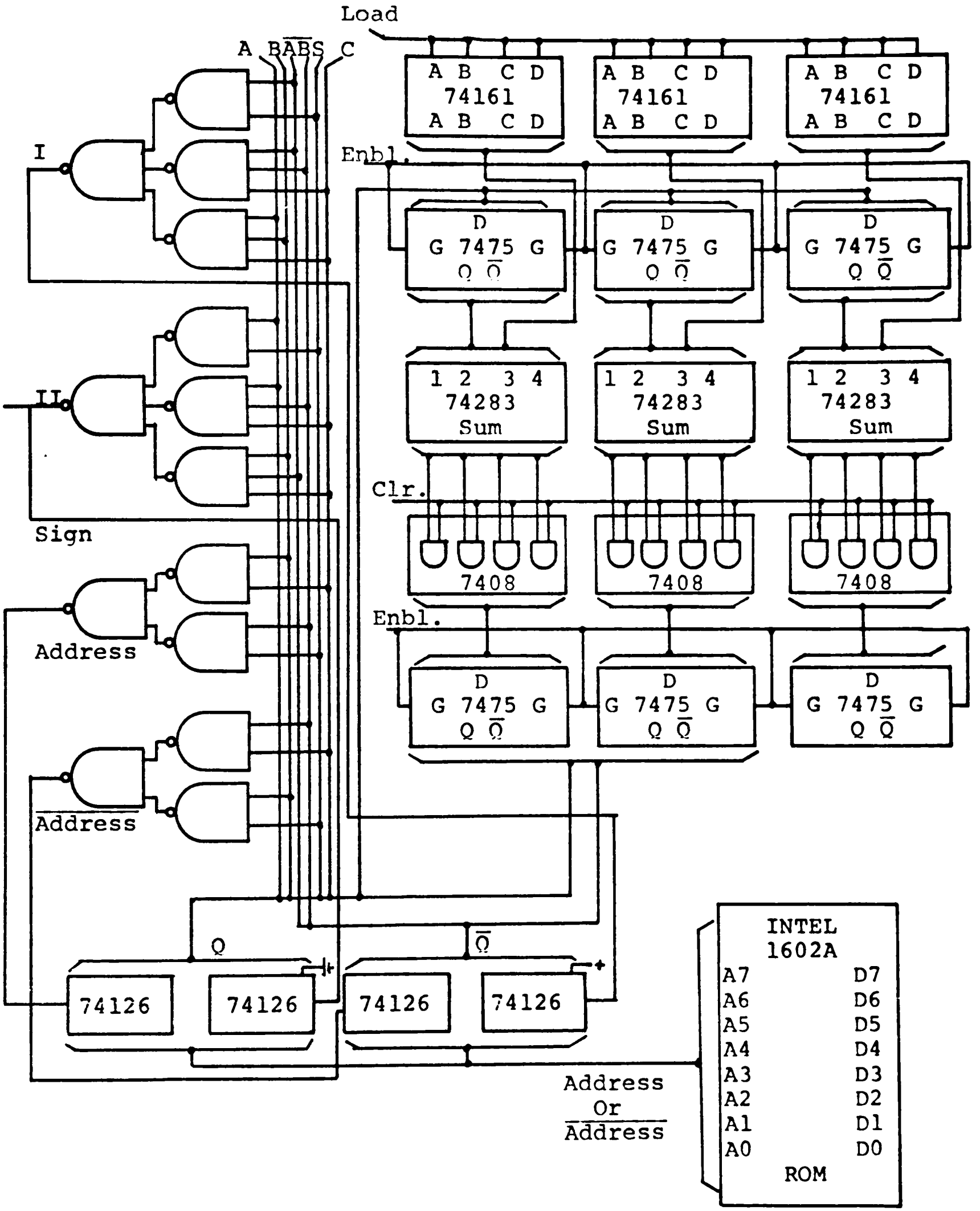
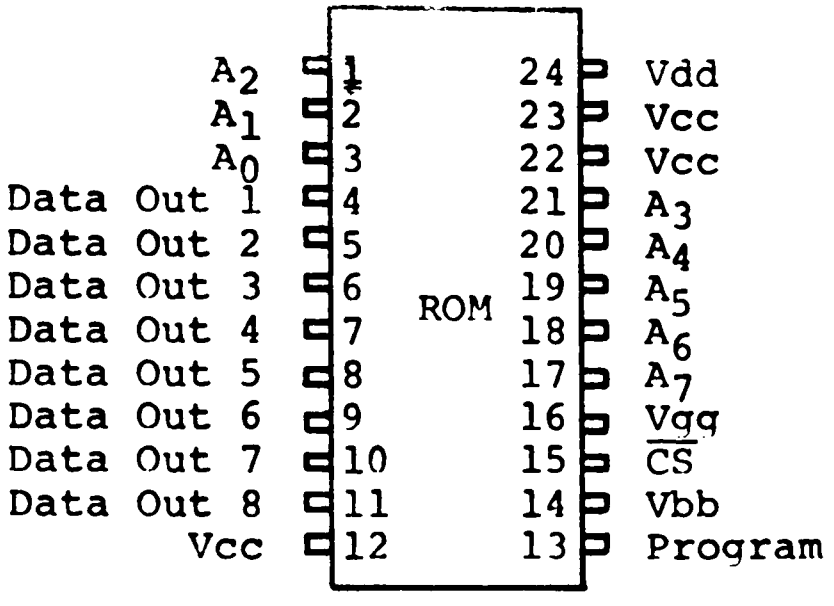


Figure 14. Detailed Address Generation Logic Diagram.

Pin Configuration



Pin Names

A_0-A_7	Address Inputs
CS	Chip Select
DO_1-DO_8	Data Output

Block Diagram

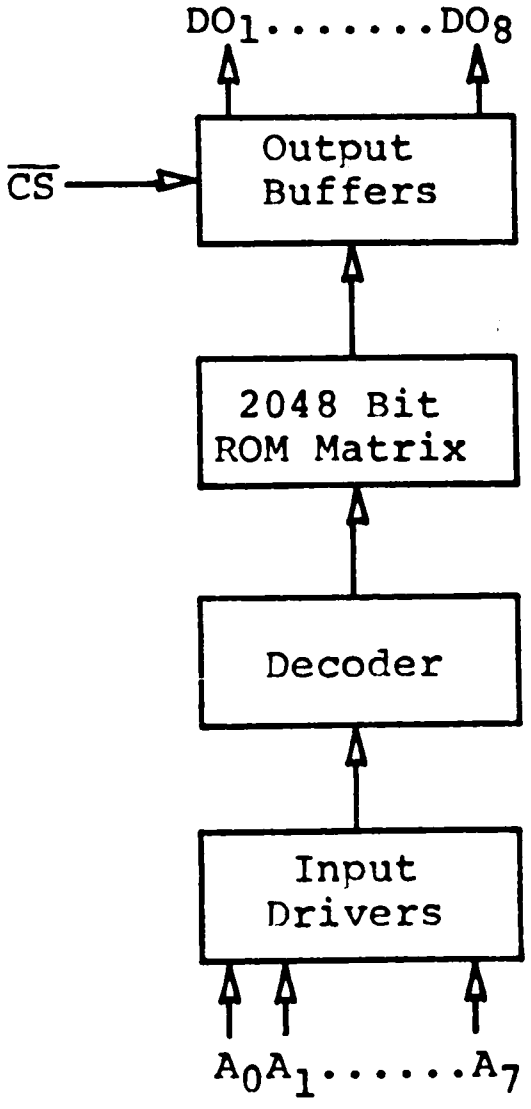


Figure 15. ROM Pin Diagram And Details.

15. Two latches are shown, of twelve bits each. The post-address-adder latch provided a buffer between the adder and the ROM. The pre-address-adder latch stored the previous address while the succeeding address was being calculated, using that same previous address in the calculation. The and-gate buffers between adder and post-latch were used during "clear and preset" in the presetting post-latch to zero. The two sets of eight-bit address buffers had tri-state outputs and were used to gate either address or the complement of the address into the ROM.

The output of the post-address-adder latch varied between 0 and 512 in selected increments, according to the sample being processed. This output was formatted as shown:

$$\underbrace{2^8}_{A}\ \underbrace{2^7}_{B}\ \underbrace{2^6\ 2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0}_{\text{Address}},$$

where A and B bits are applied to the logic block, as indicated in Figure 13. The address equations implemented by the logic block are listed in Table 4 along with the resulting simplifications. Static logic hazards were neglected in this case because of the time lag before the address was needed. Approximately 1.4 microseconds were available for the address logic to choose the correct

TABLE IV

TABLE OF ADDRESS GENERATION EQUATIONS

ORIGINAL

$\overline{A}\cdot\overline{B}\cdot Sin$ = ROM Section I	$\overline{A}\cdot\overline{B}\cdot Sin$ = Address
$\overline{A}\cdot B\cdot Sin$ = ROM Section I	$\overline{A}\cdot B.Sin = \overline{Address}$
$A\cdot\overline{B}\cdot Sin$ = ROM Section II	$A\cdot\overline{B}.Sin$ = Address
$A\cdot B\cdot Sin$ = ROM Section II	$A\cdot B\cdot Sin = \overline{Address}$
$\overline{A}\cdot\overline{B}\cdot Cos$ = ROM Section I	$\overline{A}\cdot\overline{B}\cdot Cos = \overline{Address}$
$\overline{A}\cdot B\cdot Cos$ = ROM Section II	$\overline{A}\cdot B\cdot Cos$ = Address
$A\cdot\overline{B}\cdot Cos$ = ROM Section II	$A\cdot\overline{B}\cdot Cos = \overline{Address}$
$A\cdot B\cdot Cos$ = ROM Section I	$A\cdot B\cdot Cos$ = Address

BY FUNCTION

ROM Section I = $(\overline{A}\cdot\overline{B}\cdot Sin) + (\overline{A}\cdot B.Sin) + (\overline{A}\cdot\overline{B}\cdot Cos) + (A\cdot B\cdot Cos)$

ROM Section II= $(A\cdot\overline{B}\cdot Sin) + (A\cdot B\cdot SIN) + (\overline{A}\cdot B\cdot Cos) + (A\cdot\overline{B}\cdot Cos)$

Address = $(\overline{A}\cdot\overline{B}\cdot Sin) + (A\cdot\overline{B}\cdot Sin) + (\overline{A}\cdot B\cdot Cos) + (A\cdot B\cdot Cos)$

$\overline{Address}$ = $(\overline{A}\cdot B\cdot Sin) + (A\cdot B\cdot Sin) + (\overline{A}\cdot\overline{B}\cdot Cos) + (A\cdot\overline{B}\cdot Cos)$

Note: $Sin \triangleright \overline{Cos}$

$Cos \triangleright \overline{Sin}$

address and produce the corresponding output at the ROM output pins. With a one microsecond requirement by the ROM for the addressing delay, almost .4 microsecond was left for settling time and arithmetic. This was sufficient time neglect so that instantaneous static hazards could be neglected. Other than the actual ROM address output, the only information from the address generation section is the sign bit. This bit is expanded for the multiplication and carried on through for the adder arithmetic. Preset conditions were controlled by the "clear and preset" control from the timing generation block.

C. Input Interface and Analog to Digital Conversion

The input waveform required conversion from an analog signal into a digital signal before entry into the calculation process. The sampling intervals, determined by the timing generation section, was 1/512 second. As shown in Figure 10, the time delay between the A/D conversion timing command and the write-into-storage timing command was 2.85 microseconds. The last .4 microsecond of this period was more than sufficient for the arithmetic unit to perform the multiplication and addition, and settle. The remaining 2.45 microseconds was the time period allowed for analog to digital conversion.

An eight-bit analog-to-digital (Datel ADC-EH2) was used. The input voltage range was between +5 volts and -5 volts, implying that preconverter signal conditioning was required. The input impedance was 10 Kohms and the total conversion time was 2.4 microseconds.

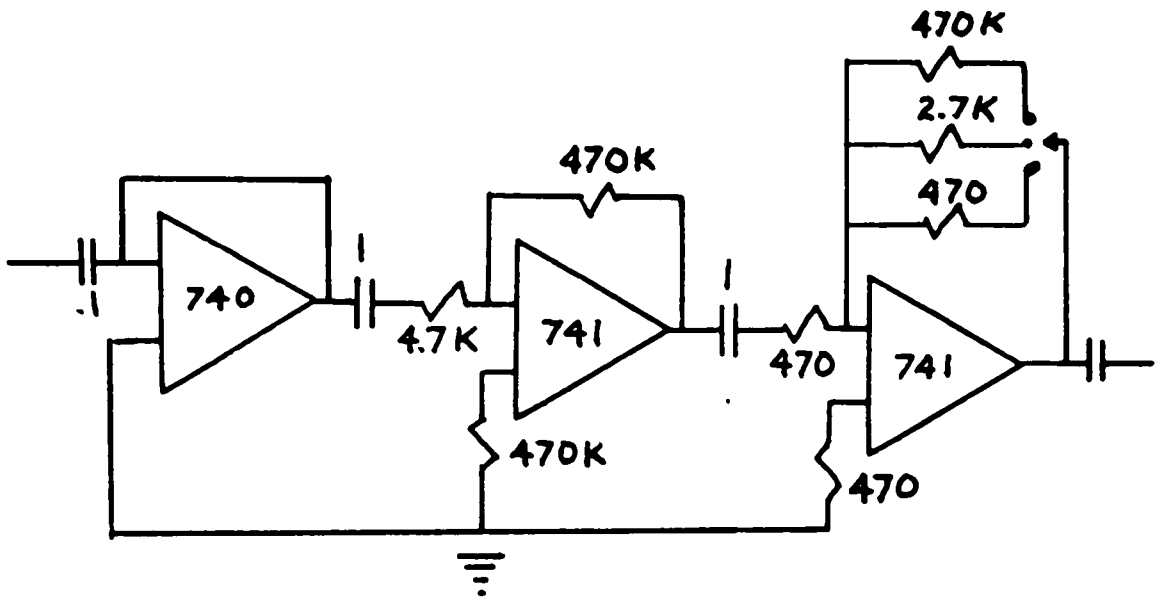
Figure 16 shows the layout of the preconverter signal conditioner. Output buffers increased the output driving capability of the A/D converter. The output of these buffers feed directly into the arithmetic section. Preconverter signal conditioning was necessary for three reasons:

(1) physiological signals are in the order of millivolts, and, in some cases microvolts, requiring at least several orders of magnitude of amplification, preceding conversion;

(2) the amount of amplification depends on the type of physiological signal being analyzed; and

(3) an extremely high input impedance is necessary to avoid losses in the coupling capacitor.

Capacitive coupling was chosen because dc levels are relatively unimportant from a physiological standpoint and



Note: Resistor values given in ohms
Capacitors in microfarads.

Figure 16. Preconverter Signal Conditioning.

dc offset errors from any preceding equipment would be eliminated.

The input impedance of a Fairchild A 740 operational amplifier in the unity gain, single input configuration is approximately one hundred megaohms. The use of a .1 microfarad coupling capacitor created at low frequency pole at

$$\omega = \frac{1}{RC} = .1 \text{ Hertz.}$$

The amplifier stages following the buffer had variable gains to accomodate three ranges of operation. The three ranges were intended to allow maximum accuracy over a specific range by utilizing the full scale accuracy advantages of the converter. The errors produced by the inherent operation of analog to digital conversion, as well as those resulting from limited bit truncation, are less important if the full scale swing of the range of the converter is used. The ranges available were from zero to 50 microvolts, zero to 1 millivolt, and zero to 50 millivolts.

Differential amplifiers and corresponding electrodes are not shown in Figure 16. Such support hardware is necessary for device operation: however, particular design will vary relative to the type of physiological signal being processed. Device limits were therefore established at the

input of the μA 740 operational amplifier.

D. Arithmetic Operations

The arithmetic operations of the DFT calculation were among the limiting factors of physical implementation. The speed at which the fundamental operations of addition and multiplication were carried out set the rate at which the input waveform could be sampled for real time processing. Four bit by four bit binary multiplier integrated circuits combined with fast look-ahead adders and arithmetic-logic units for partial product summation provided sufficient multiplier speed. Fast look-ahead adders were used for the addition operations. There were several multiplier chips reportedly available: however, those of sufficient speed were fewer and more difficult to find. The multiplication delay limit was 125 nanoseconds.

Two's complement arithmetic was chosen for device operation for several reasons. The first is that most commercially available, bipolar input, analog-to-digital converters provide their output in two's complement form along with a sign bit. (Table 5 provides a comparison of a 8-bit sign and magnitude output and an 8-bit two's complement output.) The second reason is that multiplication in two's complement provides an output in two's

TABLE V

TWO'S COMPLEMENT COMPARISON -4BIT EXAMPLE

DECIMAL	SIGN & MAGNITUDE	TWO'S COMPLEMENT	OFFSET BINARY
Full Scale +8			
+7	0111	0111	1111
+6	0110	0110	1110
+5	0101	0101	1101
+4	0100	0100	1100
+3	0011	0011	1011
+2	0010	0010	1010
+1	0001	0001	1001
0	0000	0000	1000
-1	1001	1111	0111
-2	1010	1110	0110
-3	1011	1101	0101
-4	1100	1100	0100
-5	1101	1011	0011
-6	1110	1010	0010
-7	1111	1001	0001
Full Scale -8		1000	0000

complement form. This convenient result permits the easy implementation of addition and subtraction by the use of an adder only.

Examples of two's complement arithmetic are given in Figure 17. Both addition-subtraction and multiplication are shown. The addition of two n-bit numbers, each consisting of n magnitude bits, none the less results in an n-1 bit sum, because of the requirement of one sign bit. Negative numbers in two's complement form are added with positive numbers to perform subtraction. The sign bits for positive and negative are one and zero, respectively. The resulting sign bit of the sum indicates whether the sum is positive or negative and whether the two's complement of the magnitude bits need be taken to produce a true binary magnitude.

The binary multiplication process used produced the fastest method for obtaining the product. There were other, less complex methods on from a hardware point of view, but these were too slow. Basically, the multiplier consisted of 4-bit by 4-bit binary multipliers which produced partial products; these were then combined by fast look-ahead adders to produce the product. The parallel nature of the multiplier was the reason for the short multiplication delay, relative to other methods.

ADDITION EXAMPLES

```
  -12   110100                    +12   01100
+ +10   001010                  + -10   10110
  -2    111110                    +2    00010
```

2's Comp. True Binary

MULTIPLICATION EXAMPLES

```
  3          0  0011          3     0  0011
           ⊕                      ⊕
 x5          0  0101        x-5    1  1011
+15             0011        -15       0011
                000                    011
                11                     00
                0                      1
             0  1111              1  0001
```

True Binary 2's Comp.

```
 -3          1  1101          -3     1  1011
           ⊕                      ⊕
 x5          0  0101        x-5    1  1101
-15             1101        +15       1011
                000                    000
                01                     11
                0                      1
             1  0001              0  1111
```

2's Comp. True Binary

Fig. 17. Two's Complement Examples

The following calculation illustrates multiplication of a negative number by a positive number using two's complement arithmetic.

$$\begin{array}{lll} & 2^n - A & \text{Negative number (n bits)} \\ \text{multiplied by} & & \\ & \underline{\qquad B} & \text{Positive number (n bits)} \\ & 2^nB-AB & \end{array}$$

If the provided result register is 2n bits then 2^nB will be equivalent to shifting B to the left by 2^n bits and will not appear in the result. The result is then -AB, which is a negative number, and which may be represented in two's complement form as 2^n -AB.

The multiplier design required a 16-bit by a 16-bit capability with a 16-bit result. The output from the analog-to-digital converter was 8-bits maximum, (7 significant bits and 1 sign bit). The output from the ROM section was 9-bits, (8 significant bits and 1 sign bit). Both outputs had their respective sign bit expanded such that the input to the multiplier array would be 16-bits. The extended sign bits were required to enable the parallel multiplication procedure to construct correctly the partial products. Consider, for example, the multiplication of -7 by + 3:

```
-7 = 1001
 3 = 0011
     1001
    1001
   0000
  0000
 00011011
```

However, -21 = 11101011. The multiplication did not produce the correct result because the sign bits of the negative number were not included. The correct calculation is illustrated below:

```
      ~~~~~~~~
-7 = 11111001
 3 =  0000011
     11111001
    11111001
   11111001
  1111101011
```

The bits indicated as "extended sign" merely provide the correct partial product elements to be produced.

In the device design, the two multiplier inputs resulted in a maximum possible output word length of 16-bits, therefore, a total partial product summation tree was not necessary. Only the least significant 16-bits of the possible product length of 32-bits were considered. For this reason, only those binary multipliers which produced partial product contributions to the least significant 16-bits were implemented. This procedure produced a half multiplier with a half partial product adder.

Figure 18 shows the basic block diagram of the entire machine and indicates required bit lengths throughout. The maximum value of the sine $(\frac{2\pi nk}{N})$ or cosine $(\frac{2\pi nk}{N})$ terms was 128 (the integer scaling factor). The maximum value of the digitized input waveform was limited to 100. The maximum product output of the multiplier output of 15 magnitude bits in two's complement form.

The output of the analog-to-digital converter was 7-bits of magnitude and 1-bit of sign in a two's complement format. The output of the ROM was 8-bits of magnitude with sign generation performed by the addressing logic. Table 6 shows the expansion of the sign bits of the two multiplier inputs to 9-bits and to 8-bits, respectively, for the converter and ROM. This expansion provided a 16-bit A/D converter output and a 16-bit ROM output in two's complement format for the multiplier. The output of the multiplier operating in two's complement arithmetic produced a 16-bit product which included 15 magnitude bits.

The values of the sine and cosine terms were scaled by 128 to allow integer generation. The output of the multiplier had the least significant 7-bits truncated in order to compensate for the scaling factor. Truncating the 7-bits resulted in the multiplier output being divided by

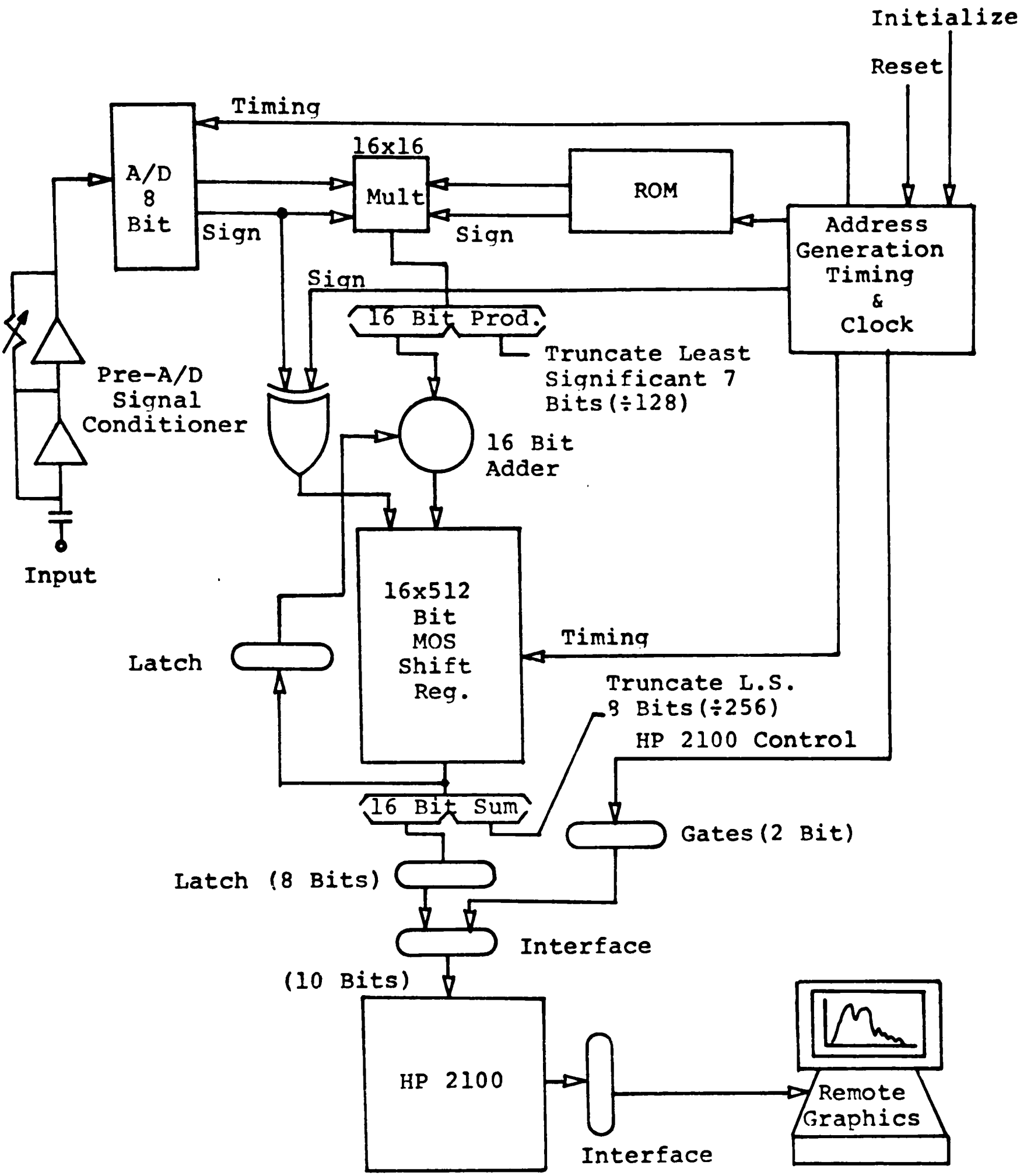


Figure 18. Final Block Diagram Showing Required Word Length.

128. The next most significant bits constitute the valid product term, which became the least significant 9-bits at the adder input. The sign bit which was expanded to 7-bits at the adder input was generated externally. The sign bit was the product of an exclusive or-gate with inputs from bit 8 of the analog-to-digital converter and the sign bit from the ROM addressing unit. The other adder input was the output of the circulating storage shift register. In actual calculations, after 512 register circulations, the adder will have added each partial product term from the multiplier to the correct position in memory. Each memory position required 512 additions to reach its total magnitude result. After truncation of the 7-bits of the multiplier output, the result remained an integer and permited, continuing integer operation.

The total arithmetic delay was less than 250 nanoseconds. This duration was within the allowed limits for the arithmetic operations. The overall function of the arithmetic section was to multiply the digital input value by the appropriate scaled sine or cosine term, truncate the product to compensate for the sine and cosine scaling, and add the result and the externally

generated sign bit to the correct memory location.

Figure 19 is a modified block diagram of the arithmetic section. The partial product layout of Table 6 provides details of line construction between binary multiplier integrated circuits and partial product tree adders. (The term "tree" refers to Wallace tree construction: a Wallace tree is a systematic partial product addition method.) Buffers were provided between the adder and the MOS circulating storage. The advantage of using powers of two sampling rate (512) and powers of two scaling factor (128) was the simplification of multiplication and division. The use of base two (binary) numbers in the digital implementation of the DFT calculation dictated the use of powers-of-two constants. The use of these constants did not cause a significant reduction in accuracy.

E. Circulating Storage And Output Buffers

Figure 5 shows the contents of the circulating storage memory. The elements of the circulating storage are the result of the method of calculation used in the DFT algorithm implementation. There were 512 multiplications and additions for each of the 512 samples of the input waveform. The 512 memory locations corresponded

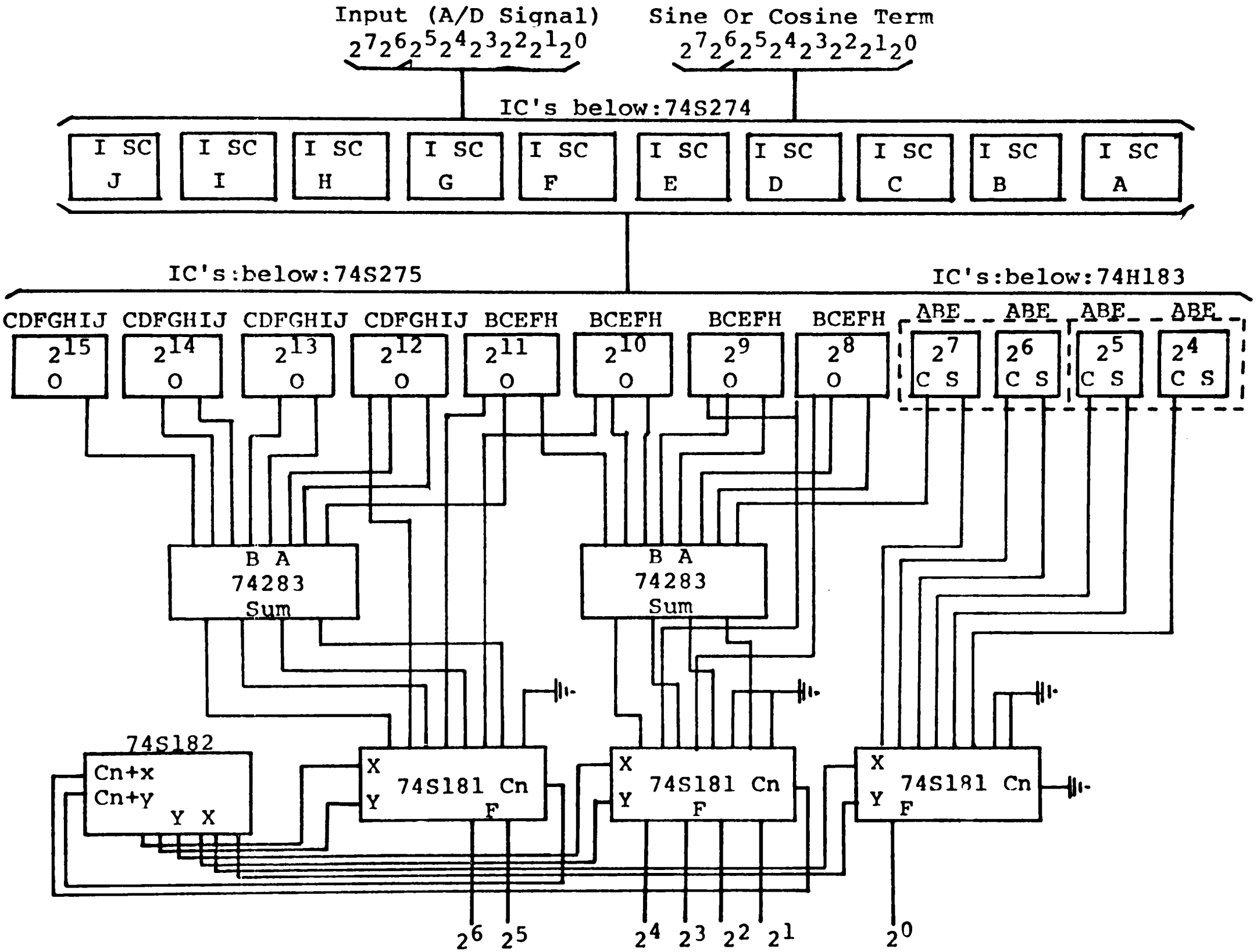


Figure (19a). Multiplier Detailed Logic Diagram Of Arithmetic Section.

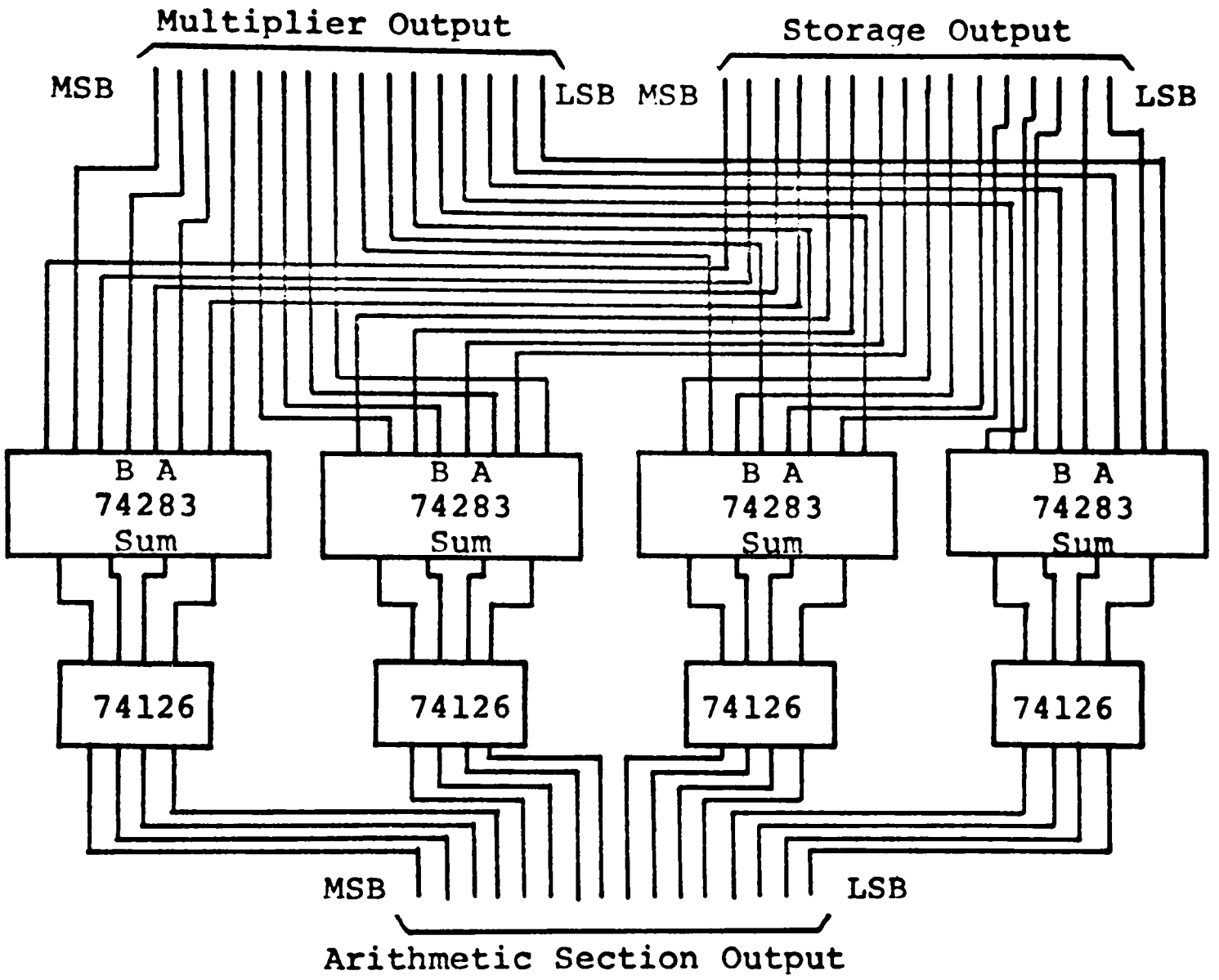


Figure 19(b). Adder Detailed Logic Diagram.

TABLE VI

TABLE OF PARTIAL PRODUCT TREE LAYOUT

(A)

Input	2^0	2^1	2^2	2^3				
Sine	2^0	2^1	2^2	2^3				
PPC	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

(B)

Input	2^3	2^2	2^1	2^0				
Sine	2^7	2^6	2^5	2^4				
PPC	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4

(C)

Input	2^3	2^2	2^1	2^0				
Sine	2^{11}	2^{10}	2^9	2^8				
PPC	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^9	2^8

(D)

Input	2^3	2^2	2^1	2^0				
Sine	2^{15}	2^{14}	2^{13}	2^{12}				
PPC	2^{15}	2^{14}	2^{13}	2^{12}				

(E)

Input	2^7	2^6	2^5	2^4				
Sine	2^3	2^2	2^1	2^0				
PPC	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4

(F)

Input	2^7	2^6	2^5	2^4				
Sine	2^7	2^6	2^5	2^4				
PPC	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^9	2^8

(G)

Input	2^7	2^6	2^5	2^4				
Sine	2^{11}	2^{10}	2^9	2^8				
PPC	2^{15}	2^{14}	2^{13}	2^{12}				

(H)

Input	2^{11}	2^{10}	2^9	2^8				
Sine	2^3	2^2	2^1	2^0				
PPC	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^9	2^8

(I)

Input	2^{11}	2^{10}	2^9	2^8				
Sine	2^7	2^6	2^5	2^4				
PPC	2^{15}	2^{14}	2^{13}	2^{12}				

(J)

Input	2^{15}	2^{14}	2^{13}	2^{12}				
Sine	2^3	2^2	2^1	2^0				
PPC	2^{15}	2^{14}	2^{13}	2^{12}				

PPC - Partial Product Contribution

Sine - Sine or Cosine Term

to the 256 real and imaginary component pairs of the DFT. Upon completion of the calculation, the real and imaginary part of the magnitude of the spectrum component at 1 Hertz were in pair location 256. The other magnitude component pairs were stored in order, with the real and imaginary part of the magnitude of the spectrum component at 256 Hertz at pair position 1. For each of the samples the circulating memory made one complete circulation. As each location appeared at the output, the adder produced a new value for the location resulting from the sum of the preceding value and the product term of the multiplier array. An 8-bit latch holds the value of the shift register output until addition is complete and the correct item for memory entry is shifted into position.

The circulating storage received correct clock levels and timing commands from the timing generation section. The length of the two-phase non-overlapping MOS level clock pulses was critical, with a minimum pulse duration of 200 nanoseconds. During the calculation and data entry phase of the device operation, the memory was circulated at 2.63×10^5 Hertz, and circulated at one-half that value during the read-out phase. The read-out phase refers to data transfer to the HP 2100.

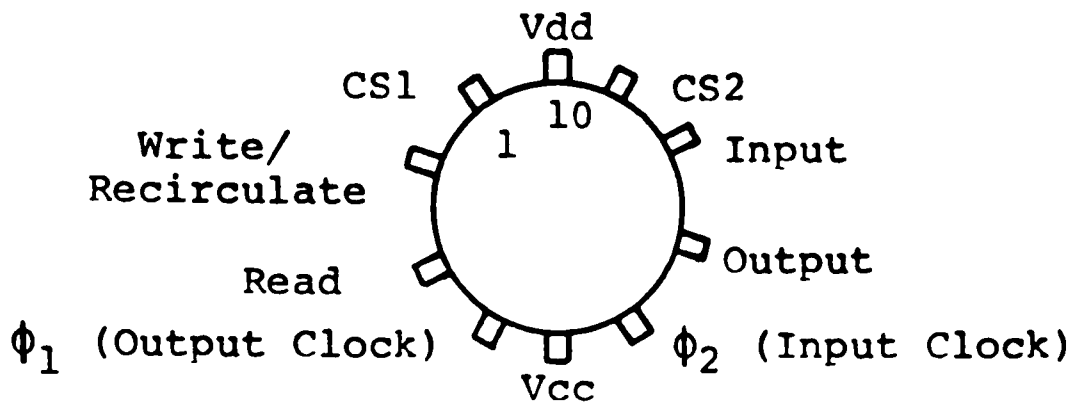
The slower circulating rate allowed the HP 2100 7.6 microseconds per data word to latch and store the information. Each data word was held by the output data latch buffers for the 7.6 microseconds upon command from the timing section.

The maximum possible total per memory location was 100 (input limit) multiplied by the resulting scaling factor of 256 (Nyquist frequency). Design for the worst case required a 16-bit word size and 512 locations. The scaling factor was a direct result of the DFT algorithm implementation. To implement the circulating storage 16, 512 dynamic recirculating storage shift registers were used (Intel Model 1405-A). The individual logic diagram is shown in Figure 20. Internal recirculation permitted flexibility for possible read-out delay without loss of information. The minimum clock data repetition for the shift registers of 200 Hertz at 25^{o}C was well below the lowest clock rate used.

The complete logic diagram for the memory and output buffers is shown in Figure 21. The appropriate latch and clocking coordination was critical. Loss of recirculation and clocking results in total data loss. The Hp 2100 control lines are also shown in Figure 21.

The algorithm calculation introduced a scaling

Pin Configuration



Logic Diagram

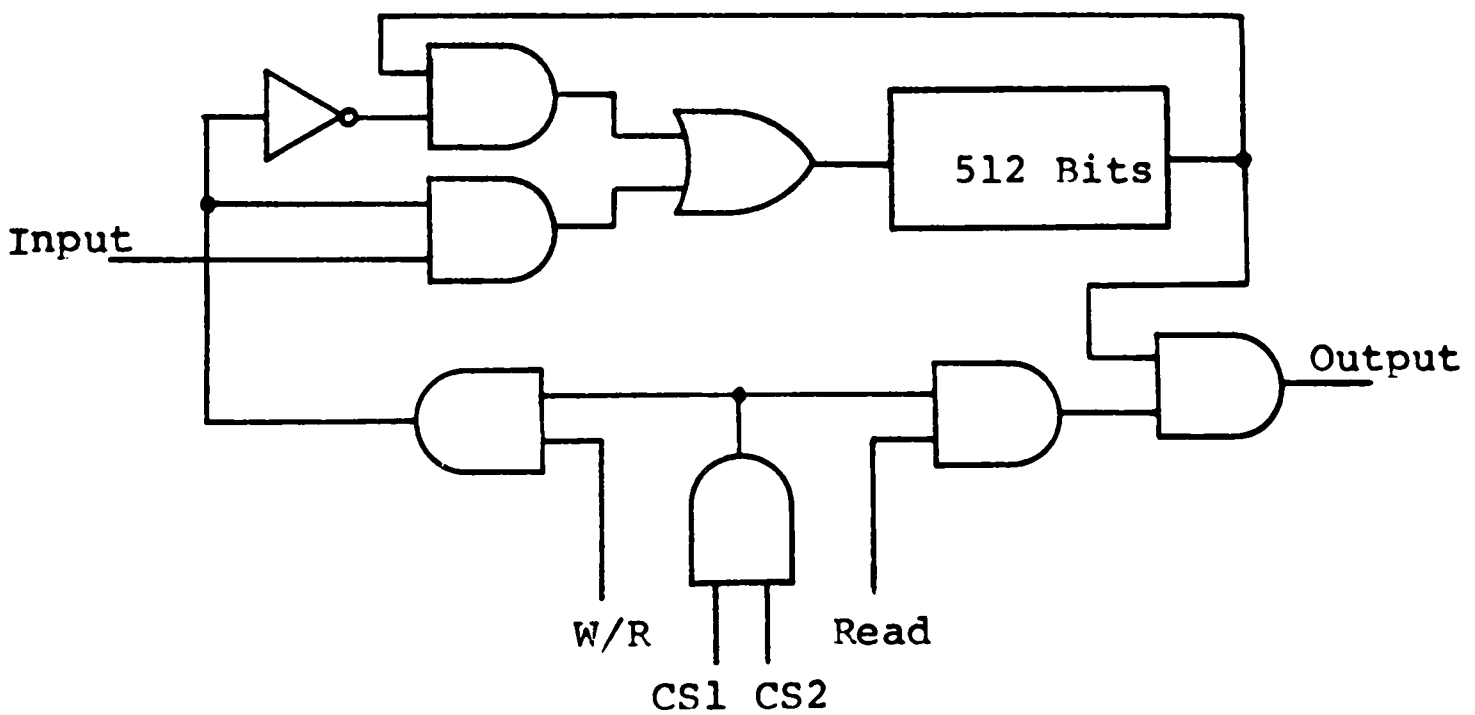


Figure 20. Logic Diagram For One 1405-A Shift Register.

Round chips:1405-A

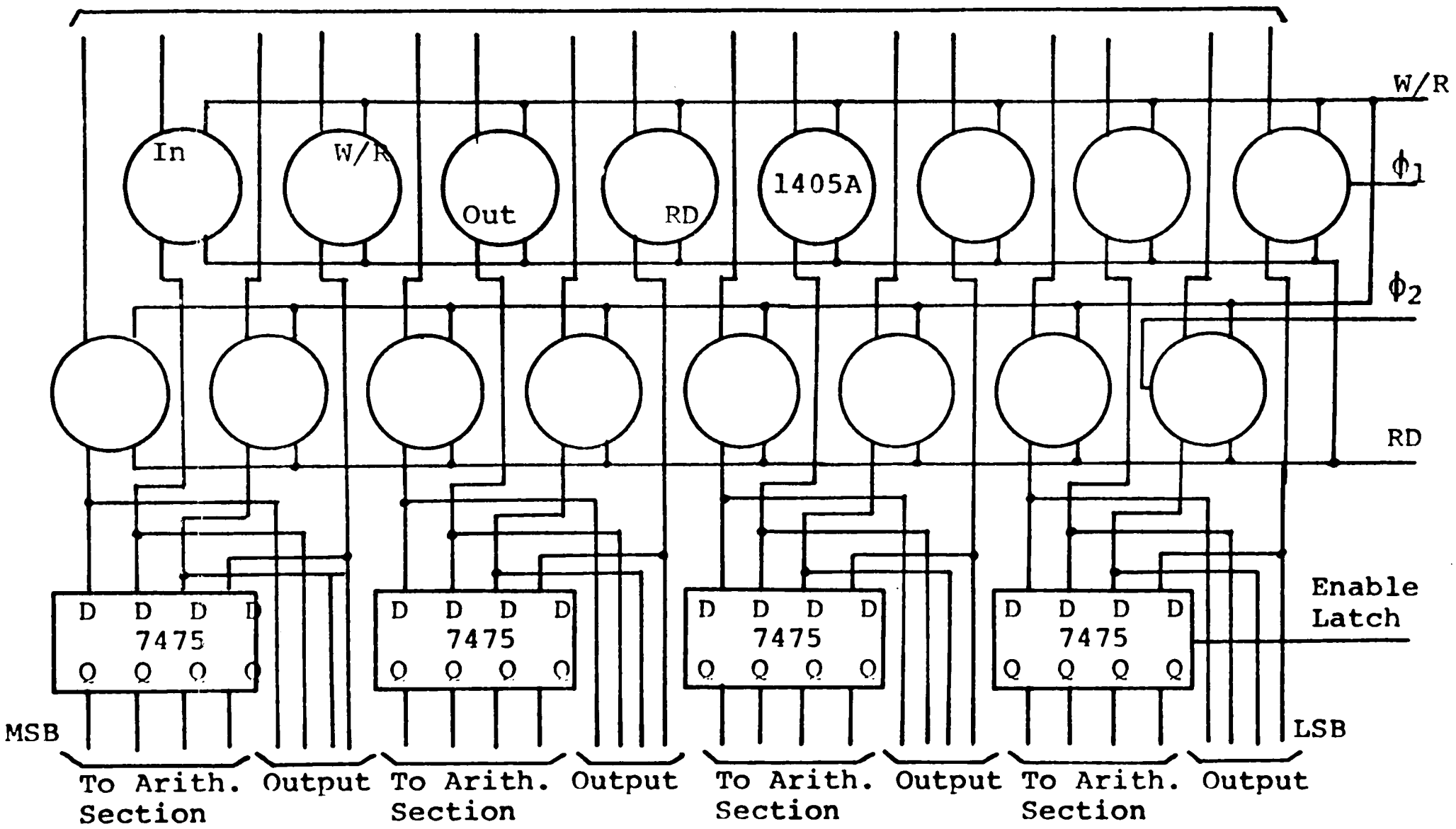


Figure 21(a). Circulating Storage Detailed Logic Diagram.

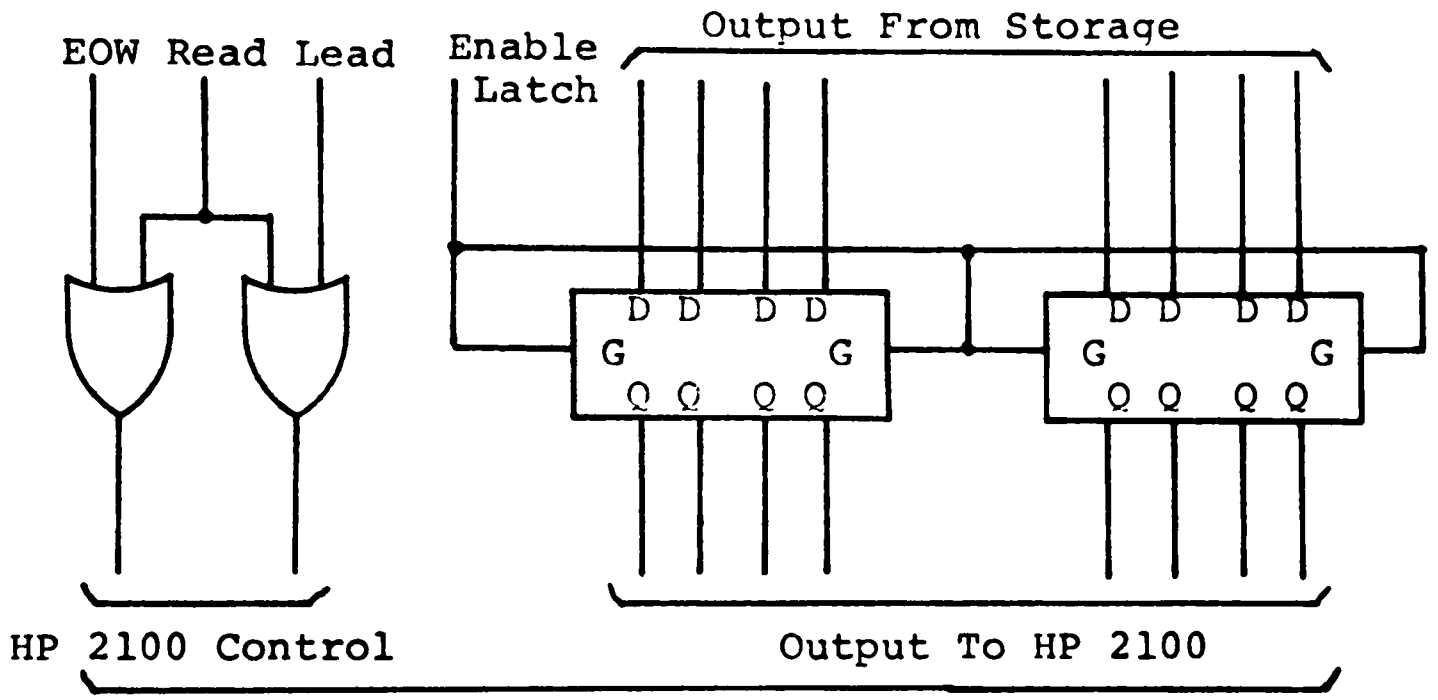


Figure 21(b). Output Buffers Of Storage Section.

factor related to the Nyquist frequency. To correct for the scaling factor, the least significant 8 output bits were truncated preceding the final output latch, dividing the final output by 256. The final output delivered to the HP2100 was made up of 10 bits. Two bits were the control lines informing the HP 2100 of device status. Eight of the bits were magnitude components of either a real or imaginary portion of a spectrum term. The truncation is indicated in Figure 18.

F. HP 2100 Minicomputer Use and General Software Approach

The actual support software for implementation of the final arithmetic functions by the Hewlett-Packard 2100 minicomputer were not included in this project. However, a very general approach and suggested guidelines are presented here.

The HP 2100 was included in the calculation procedure because of the necessity of computer support for display purposes. The minicomputer can be interfaced to the device *via* standard TTL compatible cards. Status commands produced by the timing generation section appear in parallel format along with the output data. The commands are: "ready", which indicates that a data word is about to appear at the output; "read",

which indicates that the data appearing at the device output are valid and should be read and stored; and "end of word", which indicates that data at the output of the device may be changing and are not necessarily valid. Command codes are shown in Figure 22, along with a total system block diagram. The data can be entered into the minicomputer at a maximum rate of one word per 7.6 microseconds. The 512 data words appear as successive real and imaginary component pairs and in sequential order from pair 1 through pair 256.

Once the 512 data words have been received and stored in the minicomputer, several options can be applied. A minimum delay of 1 second will elapse before a new data set will be ready for entry into the minicomputer. The software can be constructed to produce a power versus frequency spectrum or a voltage magnitude versus frequency spectrum. Each spectrum can be immediately displayed or stored for averaging with subsequent spectra. A power spectrum requires that each real and imaginary pair have the respective components squared and added. This spectrum represents some power delivered to a unit load and would actually be the voltage magnitude squared. The nature of the subject prevents current or impedance

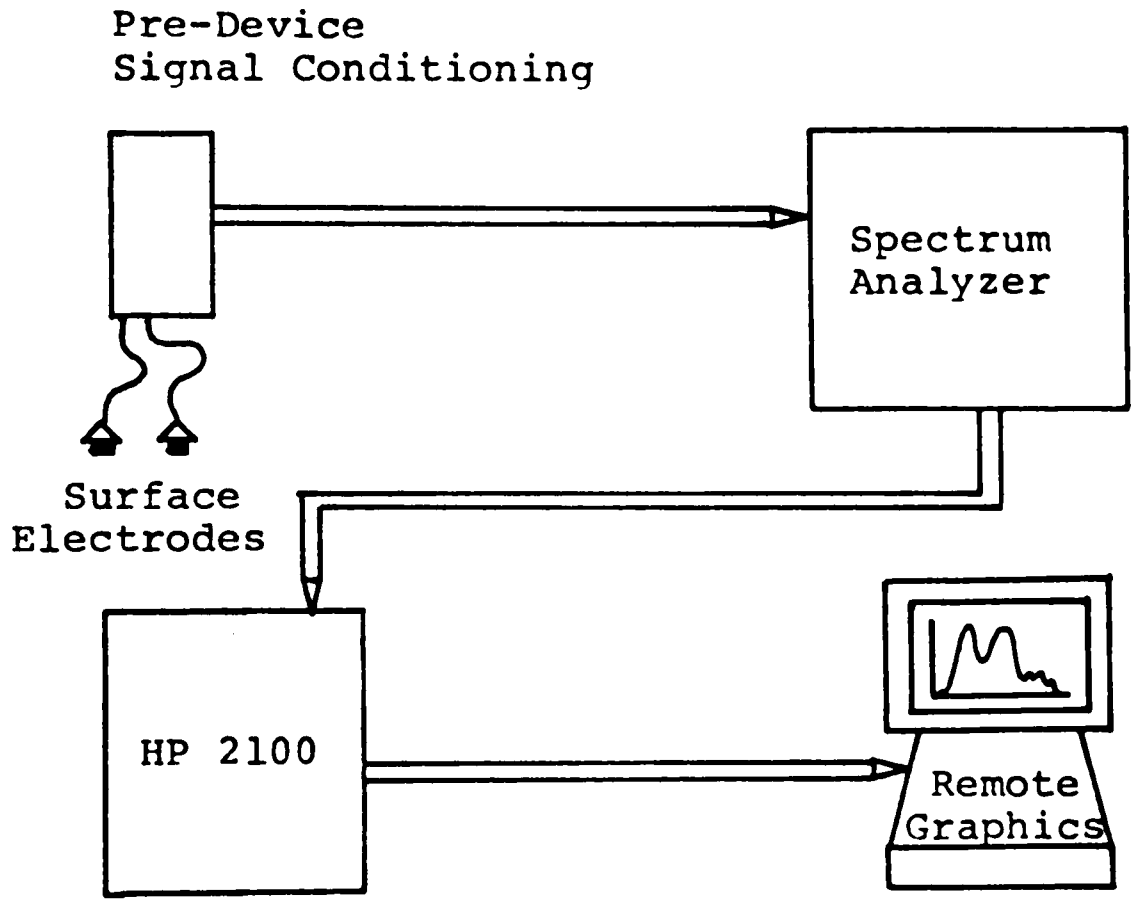
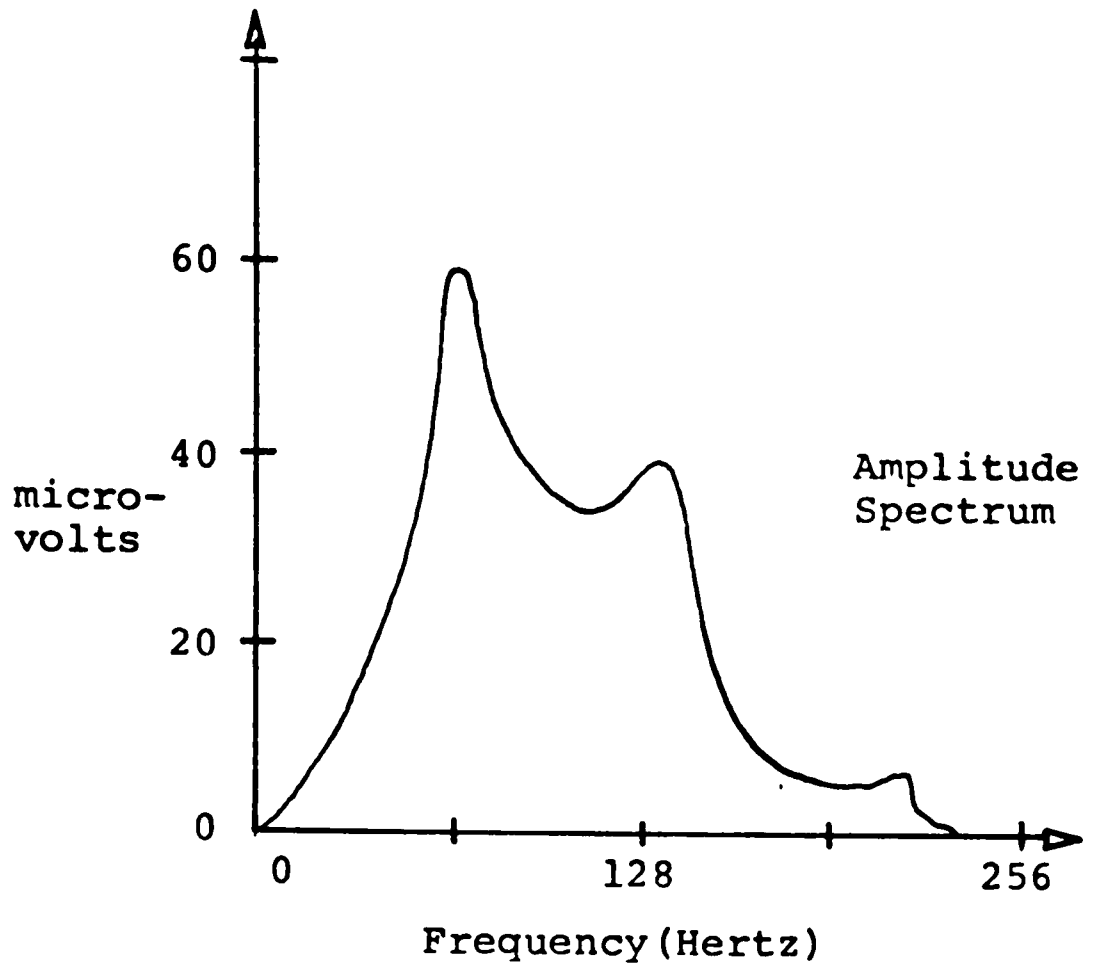


Figure 22. System Block Diagram.

measurement. A voltage magnitude spectrum requires the same operations as a power spectrum, and the additional operation that each sum have its square root obtained.

Successive averaging of spectra provides useful information in the cases of evoked responses or periodic waveforms. For example, if the same segment of successive heartbeats were the signal being examined, then averaging will eliminate extraneous instantaneous noise. Averaging is not an effective noise removal technique for non-periodic waveforms, averaging under these conditions destroys the information.

Once mathematical operations have been performed on the data, display software programs must take the generated spectrum values and provide a visual representation. An effective display of the resultant spectrum generated in a device calculation is essential in successful application of the device. A Tektronics 4010 remote graphics terminal has been interfaced with the HP 2100. Figure 23 shows several representations which could be displayed on the terminal; display consists of a simple plot routine. The output of the HP 2100 spectrum will be scaled according to the range settings on the preconversion signal conditioners. Entry of range



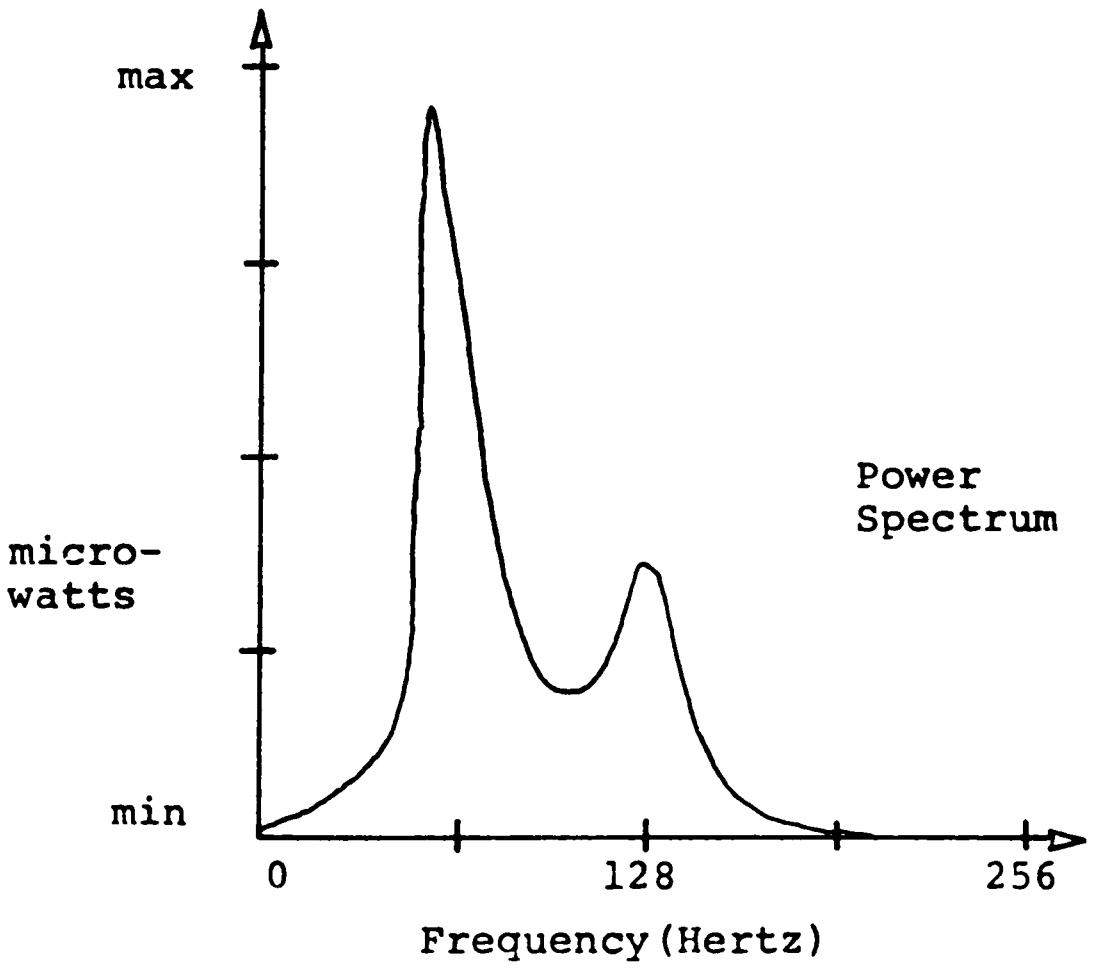


Figure 23. Spectrum Display Format Examples.

information must be manually supplied to the HP 2100.

G. Power Supply Requirements and Miscellaneous Layout Notes

Three regulated supply voltages are required by the device: standard TTL logic integrated circuits require 5 volts; MOS circulating shift registers require dual power supplies of +5 volts and -9 volts; and the analog-to-digital converter and preconversion operational amplifiers require dual power supplies of +15 volts and -15 volts. The 5 volts, ±15 volts, and 9 volt supplies are manufactured by Standard (SPS-40-5), Datel (BPM/15/100), and Powermate (RC-9), respectively.

Several additional layout considerations are shown in Figure 24 along with the general physical layout and front panel design. A standard TTL design practice is using .47 μfarad capacitors across power outputs and ground terminals of the integrated circuits. The purpose of the capacitors is to eliminate power supply dips during switching.

General Board Locations

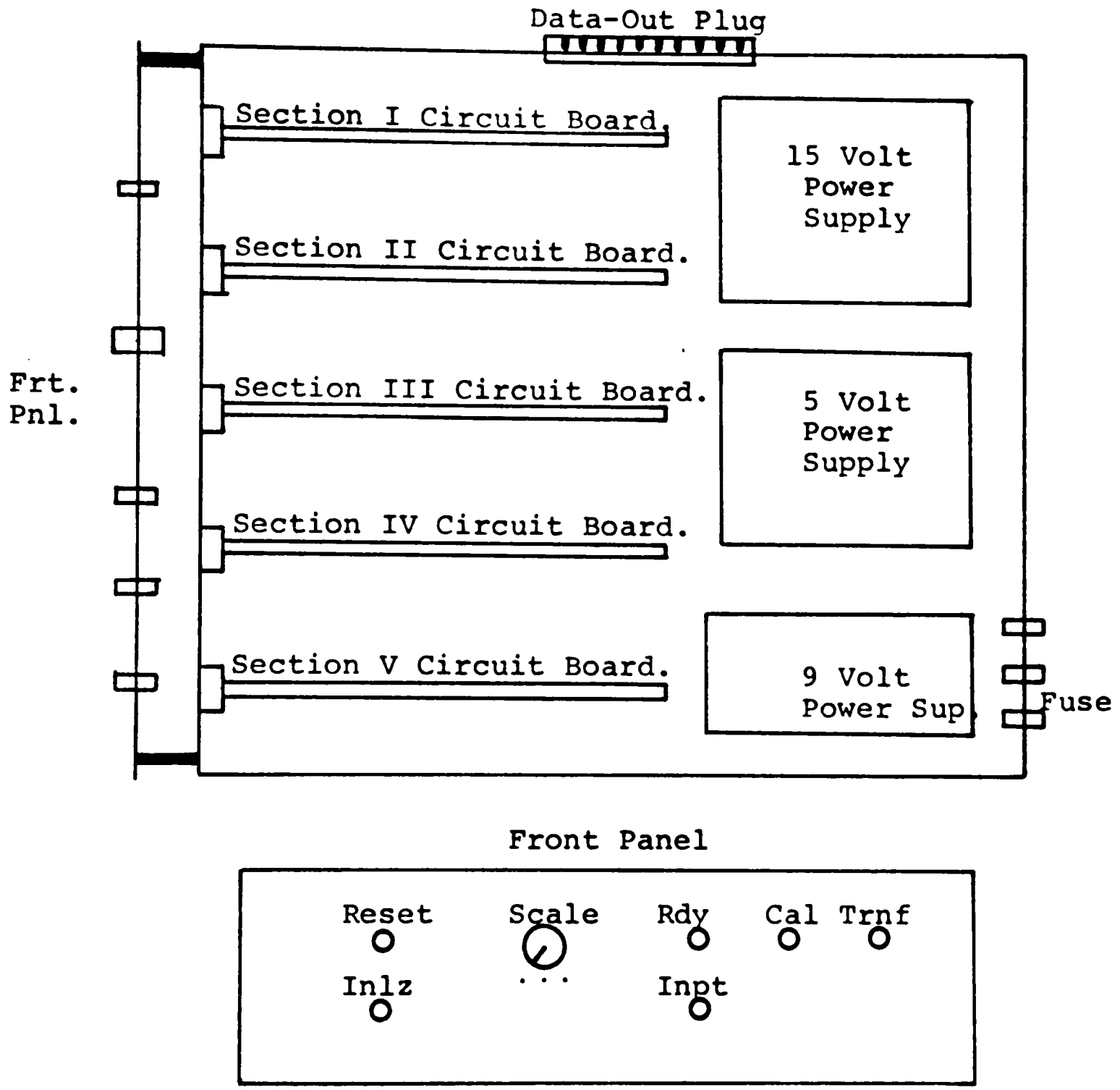


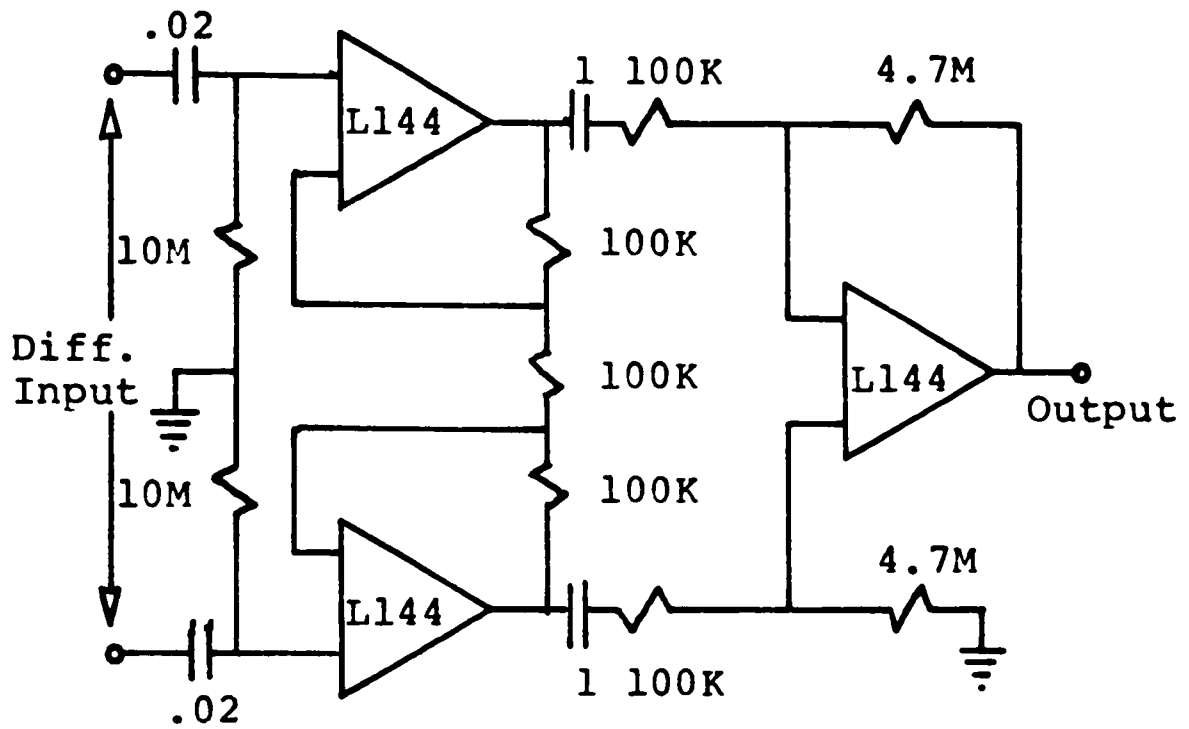
Figure 24. General Section Orientations And Front Panel Design.

CHAPTER V

OPERATION PROCEDURES, CALIBRATION AND PERIPHERAL REQUIREMENT

Operation of the device requires very little manual external control. The device is manually reset; a ready light indicates the device is ready for calculation. Calculation and sampling are begun simultaneously when the TTL logic level goes high at the "initialize" input. Once a calculation process begins, the automatic sequence of events proceedes without interruption to display if all sections are operating correctly. Two milliseconds are required to dump the calculation into the HP 2100. The total sampling interval is one second. The device must be externally triggered to provide synchronization with physiological events. Calibration is obtained by using a standard signal of known spectral content. The peripheral equipment used with the device, other than the HP 2100 and the graphics terminal, consists of electrodes and signal conditioners. An example of a possible front end is shown in Figure 25. The signal conditioners consists of three differential amplifiers in a single package.

NASA Front End



Note: Capacitance given in
microfarads and
resistance in ohms.

Gain: 1000
Linearity: 1%
Input Z: 20M
Output Z: 100
Bandwidth: 1 Hz-2500 Hz

Figure 25. EEG Operational-Amplifier Front End.

CHAPTER VI

POTENTIAL SYSTEM IMPROVEMENTS

The spectrum analyzer design was the product of economic and application compromises. The specific purpose for which the machine was designed, permitted several design compromises which are not desireable in a more generalized machine. Since relative comparisons and the general characteristics of a frequency spectrum are required, rather than accuracy, the error produced by the design here would probably be unacceptable in another device. The frequencies of interest, relative to the overall physiological picture, are in the range from 0 to 256 Hertz. A more generalized machine would probably require a much wider frequency range. The use of the graphics terminal, via the Hp 2100, allowed the minicomputer to be used also for final arithmetic operations and possible averaging techniques. The machine can be improved by increasing accuracy, range, and mobility. These improvements are not necessarily limited to the implementation of a more generalized machine. They could apply as well to improvement of the present design for the

intended use.

A. Accuracy Improvement

Increased accuracy is primarily a function of the word length or number of bits carried throughout the device. Accuracy also depends on the resolution of the analog-to-digital conversion process. Computer simulations of a twelve bit floating point arithmetic machine produced errors of less than 1% (twelve significant bits and 3 bits of exponent). Computer simulation of 16-bit integer arithmetic machines also produced only small errors. Although economics dictated the 8-bit design implementation for this project, the simulations indicate that, if accuracy were a paramount design criterion, the necessary design requirements could be met by additional logic. The type of arithmetic used (integer or floating point) is not the major limitation on accuracy. Equally small error tolerance would be possible with each type; however, a floating point machine would be slower, although more accurate, than an equal-bit-word-length integer machine.

The resolution of the analog-to-digital converter is merely a word length related accuracy variable. The smaller the change at the input of the converter which will result in an output change, relative to the over-

all range, the greater the resolution and the less the error. Digitizing an analog wave form inherently produces errors, but the higher the resolution of the converter, the greater degree of accuracy possible. If the sample rate were in the order of the time of the conversion rate, a sample-and-hold circuit could improve the conversion process. The key limitations on accuracy are economics and the usual sacrifice in calculation speed and subsequent sacrifice in frequency range.

The usefulness of real time frequency analysis has always been limited because of the small range of frequencies which could be examined at a specified resolution. The particular application of physiological spectrum generation requires that spectral components be as close as possible. Samples cannot be processed in real time before they actually appear, so that the minimum distance between spectral components is 1 Hertz.

B. Frequency Range Improvement

There were several factors limiting the frequency range which could be analyzed. The speed at which stored sine and cosine values could be accessed from the MOS ROM was a major limitation. The speed at which the basic arithmetic operations could be performed limited device

range. Read out delay to the HP 2100 and inherent TTL logic propagation delays also reduced operational speed, but less than the previous factors.

A compromise between speed (frequency range) and accuracy produced the present design. Sacrificing accuracy for range is not necessarily the only compromise possible. Accuracy could be maintained and range increased by the implementation of the scheme shown in Figure 26. The result of the implementation would be increased device operation speed and a greater frequency range, but at an increase in cost. The implementation requires two devices. One device would calculate the real portion of the spectrum component and the other device would calculate the imaginary portion. The resulting increase in speed would double the frequency range of the resulting spectrum analyzer. Using reasonable circuit modifications, the ultimate limitation for the calculation phase is the physical operation speed of individual integrated circuits, assuming display methods do not slow down the process.

C. Mobility Improvement

Economics, availability of the HP 2100, and original display criteria dictated the design presented here. Device mobility was essentially eliminated because of the

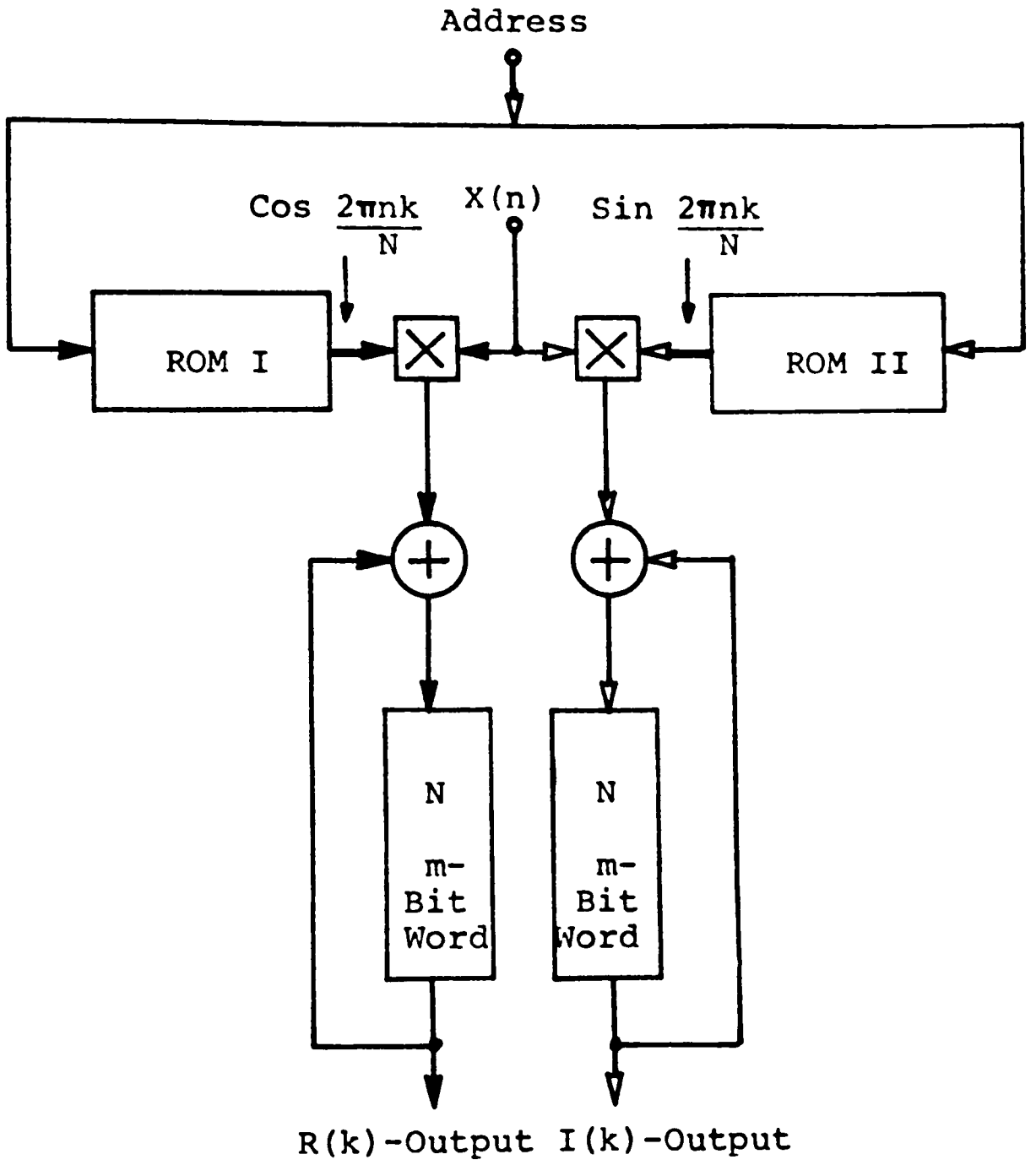


Figure 26. Speed-up Techniques.

incorporation of the HP 2100 for final arithmetic operations and display. The minicomputer can be eliminated from the overall system by the additon of an arithmetic section. The additional arithmetic section would perform the squaring and addition operations required for a power spectrum.

As Figure 27 shows, the output of an analyzer having total internal arithmetic functions can be displayed on a standard storage oscilloscope. The output could be converted from digital values to analog voltage levels and fed into the vertical control of the oscilloscope. With sweep time across the screen correctly chosen, the spectrum would appear and remain stored. The horizontal sweep duration should be exactly equal to the time required to produce all the component values, thereby producing a full screen spectrum.

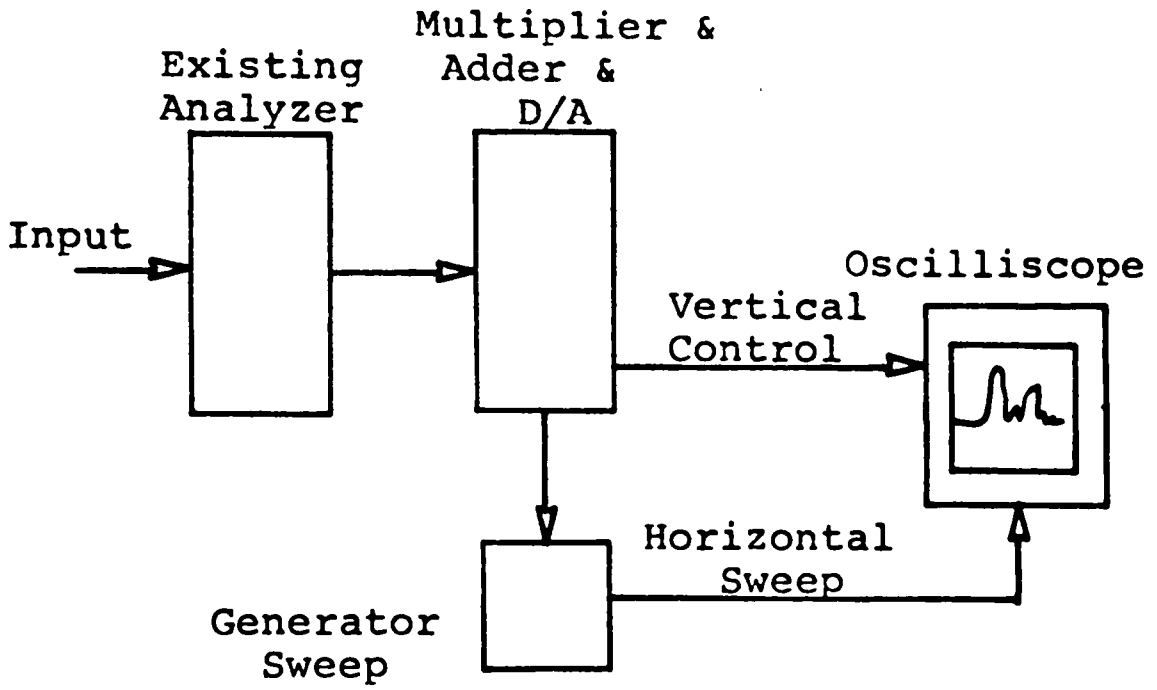


Figure 27. Mobility Improvement Block Diagram.

CHAPTER VII

CONCLUSIONS

The discrete Fourier transform implementation presented in here was the result of cost and design compromises. The circuitry was as efficient and as effective as possible under the existing, declared or implied, constraints. There are several changes which would greatly increase the usefulness and applicability of the spectrum analyzer, but each of these would also increase the cost, whether the improvements involved increased word length for greater accuracy, or faster components for extended frequency mobility. Although the device, in its present design, is of restricted application, it can be operated successfully; however, the improvements suggest that the present design is a first generation approach.

REFERENCES

1. Montcastle, Vernon B., M.D., Medical Pysiology, Vol. II, Twelfth Edition, C.V. Mosby Company, St. Louis, 1968.
2. Javid, Mansour and Brenner, Egon, Analysis and Transmission and Filtering of Signals, McGraw Hill, New York, 1963.
3. Steiglitz, Kenneth, An Introduction To Discrete Time Systems, John Wiley & Sons, New York, 1966.
4. Kuo, F.F. and Kaiser, J.F., Systems Analysis By Digital Computer, John Wiley & Sons, New York, 1966.
5. "Audio Frequency Spectrum Analysis", IEEE Transactions On Audio And Electro Acoustics, Vol. AU-18, No. 2, June 1970.

APPENDIX

APPENDIX

PART LIST

Quan.	Number	Description
1	SN74121N	Single Monostable Multivibrator
3	SN74H103N	Dual Neg. Edge Triggered JK FlipFlops with clear
2	SN74164N	8-Bit SI PO Shift Registers
8	SN74161N	Asynchronous 4-Bit Binary Counter
6	Sn7408N	Quad 2-Input Positive And Gate
3	SN7432N	Quad 2-Input Positive or Gate
2	SN7404N	Hex Inverters
2	SN7425N	Dual 4-Input Positive Nor
3	Sn7410N	Trip 3-Input Nand
6	SN7400N	Quad 2-Input Hand
1	SN55180S	Dual TTL-MOS Level Converter
8	Sn7475N	4-Bit Bistable Latch
9	SN74283N	4-Bit Binary Adder With Fast Look Ahead Carry
12	SN74126N	Quad Tri-State Bus Buffer
10	SN74S274N	4x4 Binary Multiplier
8	SN74S275N	7-Bit Slice Wallace Tree Adder
2	SN74H183N	Dual Carry Save Full Adder
3	SN74S181N	ALU/Function Generator
1	SN74S182N	Look Ahead Carry Generator
1	1602-A	(Intel) Electrically Progtamable ROM
16	1405-A	(Intel) 512 Bit Recirculating MOS Shift Reg.

APPENDIX CONTINUED:

Quan.	Number	Description
1	ADC-Eh2	(Datel) 2.4 s A/D Converter (8-Bits)
1	740	FET Input Operational Amplifier
1	741	Operational Amplifier (General Purpose)
1		Oscillator Control Crystal
1		+9v±5% (500ma) Power Supply
1		±15v±5% (100ma) Power Supply
1		+5v±5% (2a) Power Supply
42		14 PIN Dual-In-Line Package I.C. Sockets
34		16 PIN Dual-In-Line Package I.C. Sockets
10		20 PIN Dual-In-Line Package I.C. Sockets
4		24 PIN Dual-In-Line Package I.C. Sockets
17		10 PIN Round Sockets
1		Special Sockets For A/D Converter
1		Container Case or Front Panel For Rack Mount
5		32 PIN Cinch Edge-Board-Connectors
		Miscellaneous Wire & Circuit Boards