ISBN 0-315-61813-2

THEORY AND DESIGN OF A PRECISION DUAL SYSTEM TO MEASURE TIME INTERVALS IN NANOSECONDS AND AUTOMATICALLY SYNCHRONIZE FROM TWO TO FOUR EXTERNAL SYSTEMS

CENTRE FOR NEWFOUNDLAND STUDIES

TOTAL OF 10 PAGES ONLY
MAY BE XEROXED

(Without Author's Permission)

WILLIAM JACK P. HARVEY

THEORY AND DESIGN

OF

A PRECISION DUAL SYSTEM TO MEASURE TIME INTERVALS IN NANOSECONDS AND AUTOMATICALLY SYNCHRONIZE FROM TWO TO FOUR EXTERNAL SYSTEMS

BY

© William Jack P. Harvey, B.Sc. (Hon.)

A thesis submitted to the School of Graduate Studies in partial fulfillment of the requirements for the degree of Master of Science

Department of Physics
Memorial University of Newfoundland

February 1984

St. John's Newfoundland

National Library of Canada
Bibliothèque nationale du Canada

Canadian Theses Service
Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-61813-2

Canada

TO

The Living GODS

THEY created the universe

Created man

and

Gave man the ability to create

ABSTRACT

Verniers are employed widely in determining precise spatial measurements. A new time measuring system has been created, which may be revolutionary in nature, since it employs digital electronics and does not concern itself with coincident edges as do contemporary systems. The system may be the first of its kind, being a true digital vernier system, which is capable of:

1. determining the time delay inherent in a device or a transducer.
2. producing time delays precise to a nanosecond.
3. balancing the time delays of associated devices.

An analysis was made of the phase variations between the respective edges of the square pulses of two wave trains of different frequencies to determine if the principle of vernier measurement could be applied to create such an electronic system.

This led to the design of the device described herein, consisting of four parallel modules each capable of operating individually in either an automatic mode A, or a semi-automatic mode B. For mode A, time intervals between two external stimuli are automatically determined to a precision of one nanosecond by a module, while for mode B, required time delays are inserted at the operator's discretion. As soon as the time intervals have been

established, in either mode, the system then inserts the proper balancing time delays and initiates the four modules so that they produce properly timed pulses to external equipment according to a predetermined time sequence. Once set, the system is capable of automatic repetition.

The system has been designated as the "Precise Time Interval System" (PTIS).

ACKNOWLEDGEMENTS

I wish to express my heartfelt appreciation to the following persons:

My Grandparents, who in spite of the Great Depression and illness from World War I wounds, struggled to send me to High School,

and

My dear friends, Vickie and Roy Shawcross, who never failed to encourage and support me.

Dr. John Alcock, Head of the Laser Division, National Research Council, Ottawa, who took time and effort to listen to me and affirm the importance of the instrument herein described, when I needed consultation.

The electronics staff, Physics Department, National Research Council, Ottawa, for both testing the prototype oscillator and evaluating the system as to its feasability.

Dr. Frederick Aldrich, Dean of Graduate Studies, Memorial University, who listened, was convince, and advised me to finish the project, and did much to make the completion of this project possible.

Dr. James D. McAndrew, with whom I could always discus my problems, and whose advice was invaluable and

freely given.

Dr. David H. Rendell, Head of the Physics Department, Memorial University, for his assistance in bringing this work to a conclusion.

Dr. Bryan Latta, for his advice and patience, as my supervisor, in the final preparation of this thesis.

Dr. C.W. Cho for financial assistance during the early planning stages of this instrument.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Overview

There was a need to design a system capable of timing two or more electronic pulses to a high degree of precision. To define parameters for this undertaking, the following situation was considered. Two lasers are to be fired so that the pulse of light from one laser impinges on the plasma created by the pulse of light from the other laser with the concurrent operation of recording equipment, e.g. a streak camera.

If the outputs of two electronic devices are to occur simultaneously, one must find the total time delay for each device and its connecting transducer. Extra time delay can then be inserted in order to balance the time delays of the two devices.

The lifetime of a laser pulse is very short. If during this lifetime we require the occurence of a pulse from a second laser, the initiation time of each laser and any recording devices is critical.

In order to get maximum monochromaticity, a laser requires a pockels cell and a dye cell. The pockels cell, which is an optical shutter, must open just as the dye becomes completely bleached. Hence precision timing of the opening of this shutter, to nanoseconds, results in

a narrowing of the frequency spectrum of the laser to a single mode.

The system described in this thesis was designed for the dual purpose:

1. to measure the interval of time between two events, i.e. the occurence of two laser pulses, and
2. to create intervals of time,

all accurate to one nanosecond.

Because of the complexity of the system described, Fig. 1.1, a block diagram, is presented to aid in visualizing the relationships between the components which constitute the system. The components are integral circuits in their own right. The paths by which data may enter the Passive module are lettered from 1 to 3.

Path 1. data enters from Manual - 'manual digital switches';

Path 2. data enters from the External Interval Evaluation unit;

Path 3. data may enter an input of SYNC (synchronization) from either the Manual or External Interval Evaluation units and thence proceed through the A.L.U. Division (Arithmetic Logic Unit) to the Passive module.

Data, updated and stored by the Passive module, passes to the Active module which creates the required time interval.

The Manual insertion of data appears in

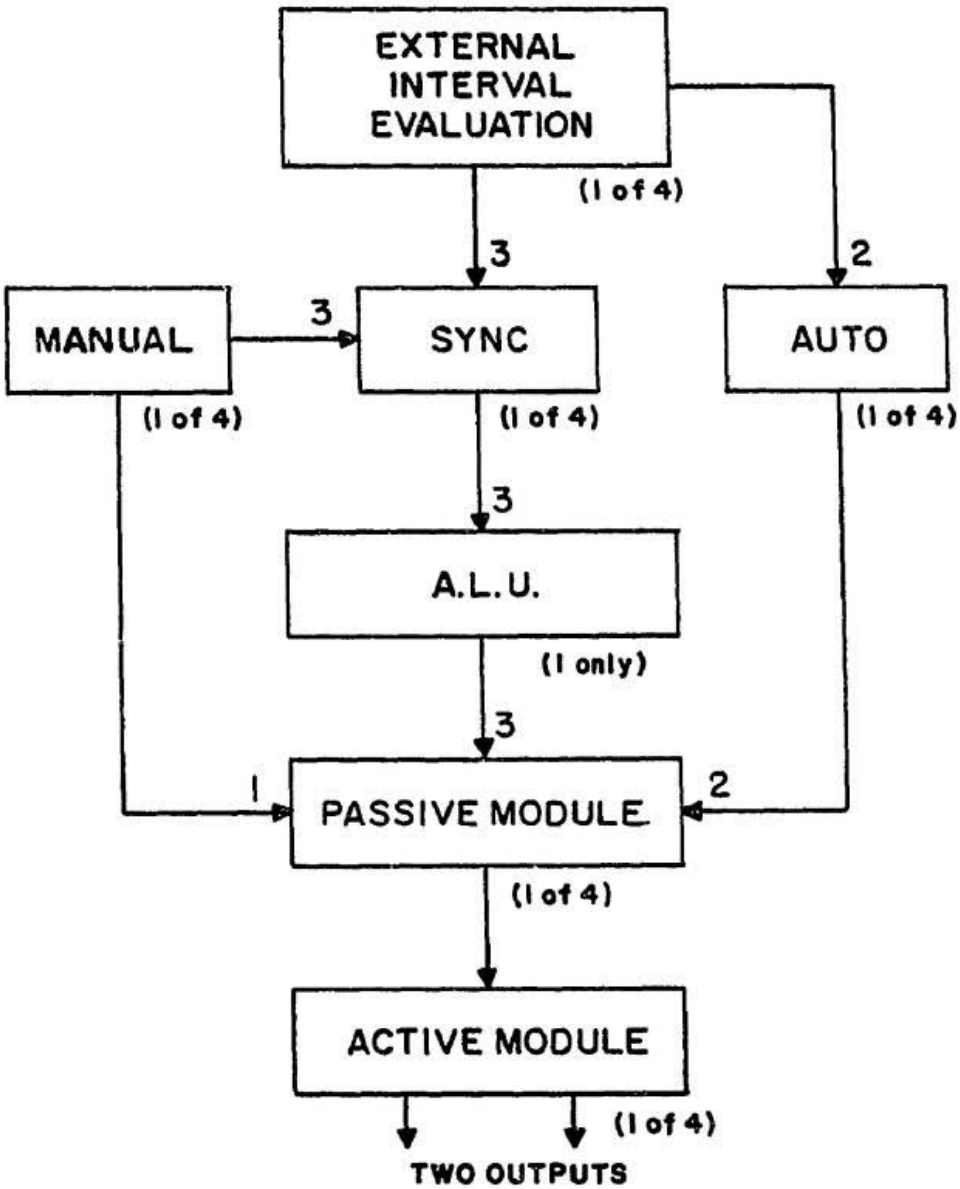


Fig. I.I BLOCK DIAGRAM OF TIME INTERVAL EVALUATION AND CREATION SYSTEM

Chapter 5, while External Interval Evaluation of data is covered in Chapter 4. The mode "SYNC" is dealt with in Chapters 2 and 11, while the A.L.U. is explained in Chapter 6. The Passive and Active modules are explained in Chapters 7 and 9 respectively.

But before progressing further, it is first necessary to state the principles underlying this system and to explain how the system came to be designed.

The digital logic employed in this thesis is based on the Gray code (1). The simple rule for generating the Gray code is as follows:

Begin with a state of all zeros. To get to the next state, always change the single least significant bit that brings you to a new state: i.e.-

0 = 0 0 0 0

1 = 0 0 0 1

2 = 0 0 1 0

To represent a number using only 0 and 1, we use the binary or base 2 number system. Each 0 or 1 multiplies a successive power of 2. For the number 94 we have:

$$94 = 1x2^6 + 0x2^5 + 1x2^4 + 1x2^3 + 1x2^2 + 1x2^1 + 0x2^0.$$

Now the input and output terminals for the digital counters employed in this system occur in the order given by:

$1x2^0 + 1x2^1 + 1x2^2 +$ etc from left to right, being represented by A, B, C, D, and Q_A, Q_B, Q_C, Q_D respectively as shown in Fig. 3.1b and Fig. 4.8. Thus in this paper

the standard binary system is written in the reverse order for the above number, i.e.-

$$0x2^0 + 1x2^1 + 1x2^2 + 1x2^3 + 1x2^4 + 1x2^5 + 1x2^6$$

For this thesis, a clock is defined as any device such as a counter, flip-flop, or specially designed circuit, etc that counts or keeps track of pulses either singularly or in predetermined groups.

The letter C, standing for clock, represents the train of square pulses or clock pulses emitted by an oscillator. If the pulses in this train are shifted by half a period, we obtain what is called the inverted clock, and this clock is represented by $\overline{C}$.

The term 'division' will be employed to indicate a major section of the system. A division is established by the assemblage of two or more sections in order to carry out a specific function. There are two functions performed by the system:- create required intervals, or measure existing intervals. This permits the use of terms such as 'section' or 'unit' etc to designate a subdivision or a smaller part of a division. Now repetition of names is allowed. For example, the A.L.U. Division contains an A.L.U. unit as a smaller subsection.

1.2 Spatial measurements

When we wish to determine minute spatial measurements such as decimal fractions of the smallest unit of our measuring tool, we add to our tool a short scale

which can move along our measuring tool adjacent to its scale markings. The length of this scale is chosen to be equal to nine tool scale divisions, and we divide this length into ten equal parts. This short minor scale is known as a vernier scale. The tool scale is the major scale.

We can now determine a length to one tenth of a small unit on our major scale. For example, in Fig. 1.2, arrow 1 indicates the last whole division on the major scale 20.5, while arrow 2 indicates the numerical value of the vernier distance between arrow 1 and the zero line of the vernier scale. Arrow 2 lines up five on the vernier scale with a scale marking on the major scale. Thus the vernier adds 0.05 to the major scale reading, giving us 20.55 as the determined measurement.

Here it is necessary to know,

1. the position of the last scale mark on the major scale before the zero line of the vernier scale, and

2. the vernier scale mark that lined up with a major scale mark.

Also it is noted that the figure 5 on the vernier scale lies outside the actual length being measured.

In to-day's contemporary electronic vernier interpolating systems, this is the very method employed. The matching up of scale divisions is referred to as finding coincident clock edges (1). Also, a measurement of a time interval is completed after the interval has terminated. Such systems do not therefore permit the concurrent measure-

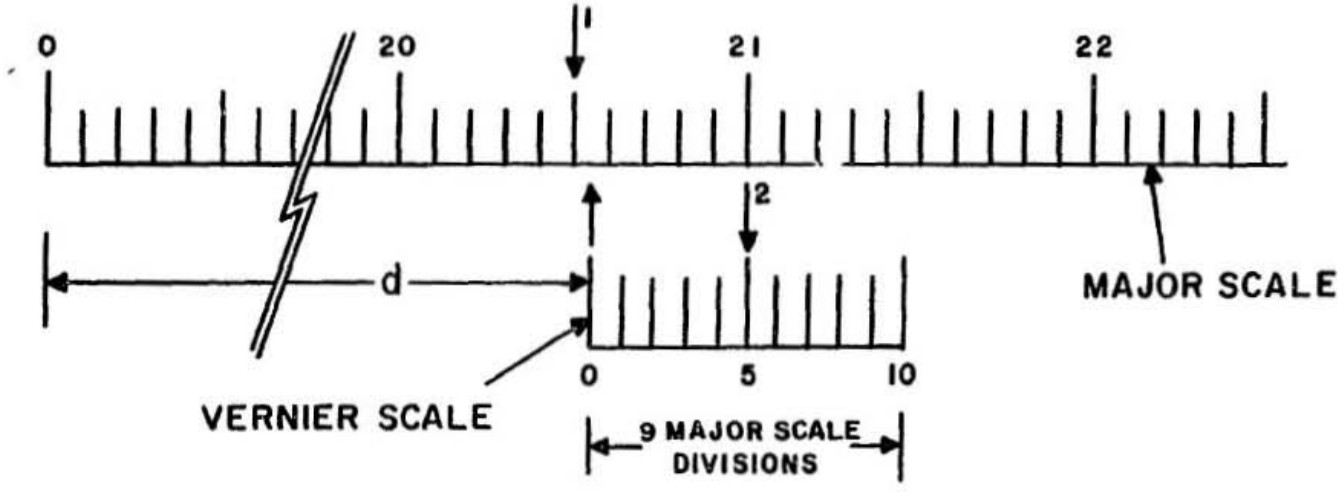


Fig. 1.2 SPATIAL MEASUREMENT

ment of time and adjustment of associated equipment to yield automation, and also, the precision is limited, especially by noise, internal or external to such systems.

Further, although the vernier scale moves along the major scale, it is not termed a sliding vernier scale because it can add to any major scale reading only the numbers ranging from 0.01 to 0.09 inclusive. A true sliding vernier is one whose range varies as it slides along the major scale. A possible example of a sliding vernier would be a scale moving in logarithmic space where its scale divisions vary as it moves through that space, so that the scale divisions always relate to that section of space it momentarily traverses.

1.3 Time determination of an event

To find where in time an event takes place, we require a tool to keep accurate appraisal of the passage of time, similar to the major scale of Section 1.2.

Although time is assumed to flow continuously and smoothly, our measuring devices give us a physical realization of time which is granular. That is, we relate time to the tick of a clock or to the appearance of a number on a luminous read-out. In olden times, the passage of time was related to the steady flow of sand from one glass vessel to another glass vessel, - very closely depicting our true sense of time as being continuous and regular in its dissipation.

In this instrument, we are both creating and measuring intervals. When creating intervals, the vernier must terminate on the end of the created interval, and in order to do so, it must commence within the interval. This is a new concept which leads to the reversed electronic vernier.

1.4 To create one or more events in time

Suppose we decide to cause the occurence of a flash of light somewhere at a predetermined time. How do we know when to apply the stimulus for the event to occur at the proper time? All systems have inherent delays that must be taken into account.

In case we wish to create two events separated from each other by a definite time interval, we still have the problem of when to initiate the first event and in addition we must be able to determine when to initiate the second event, so that the delays in both instruments are properly taken into account. That is the two instruments might have different reaction times.

1.5 A time measuring system

Sections 1.3 and 1.4 have each indicated the vital need for a time reference scale such as the major scale of Section 1.2 so that we can know where we are in time. The arbitrary zero of time is decided by the experiment.

Let us assume that we can somehow start time and accurately cut or set off known intervals of time on an

artificial time scale. But recall, our time measuring devices are granular, as is also true concerning spatial measurements. We attempt to overcome this granular aspect by decreasing the size of our granules. This in turn decreases our discrepancy in measuring, but never entirely removes it. Yet this approach can simulate a continual steady monitoring of time flow, especially if our granules become progressively smaller. That is, if we can vary the size of the granules at will, we can improve the time measuring to an acceptable level for any particular application.

Thus a measuring system is proposed which,

1. on its major scale records time in precise and repetitive time intervals, - fixed granules, and

2. on a second scale, a vernier, yields intervals whose widths are smoothly varying in order to measure the time between a point on the major scale and the time of an event occuring between points on that time scale.

Consider a train of square pulses being emitted by a crystal controlled oscillator of high frequency stability, as in Fig. 1.3a. Here all pulses have the same shape and are separated from each other by a constant interval of time T called the period of the wave. Consider the arrows shown on the vertical sides of the pulses. Those pointing upwards occur in time before those pointing downwards, since time increases as we go from left to right across the figure,

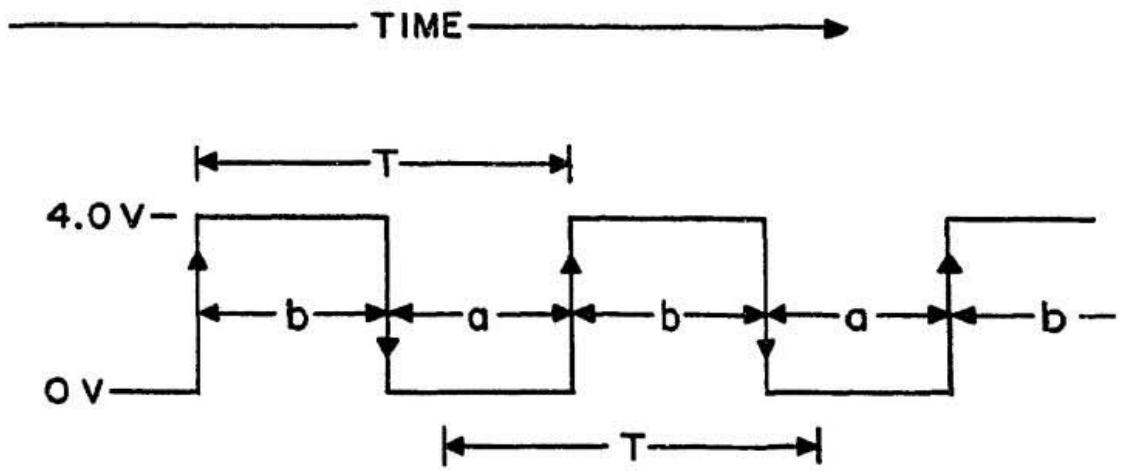


Fig. 1.3 a
WAVE TRAIN OF CONTINUALLY
RUNNING OSCILLATOR

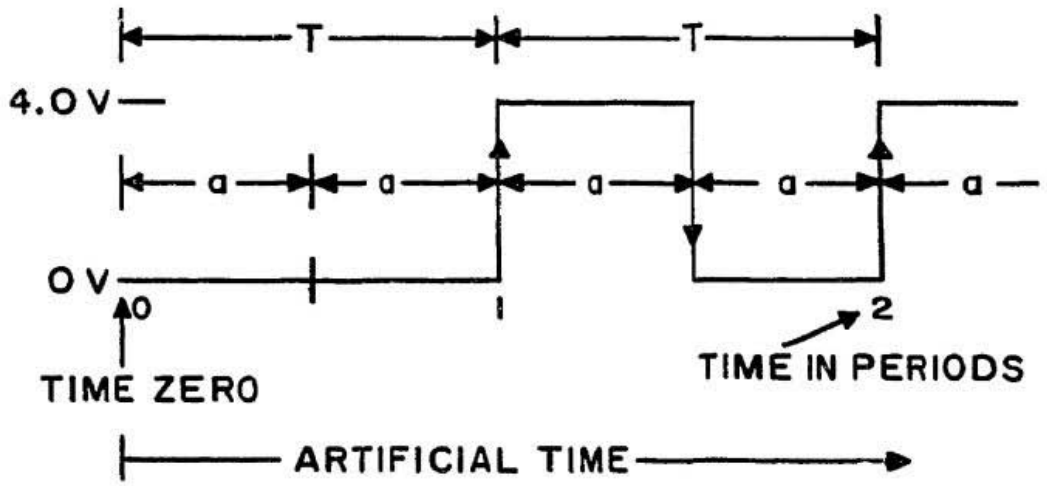


Fig. 1.3 b
WAVE TRAIN FOR GATED OSCILLATOR

or one may consider that the train of waves is approaching the observer from the right. The arrows pointing upwards indicate the leading or rising edges of the pulses. The other arrows indicate the trailing or falling edges of the pulses. These pulses will be termed clock pulses in this thesis because they are used to activate counters or clocks.

Inspection shows that the leading edge of any pulse occurs at the same precise time interval after the leading edge just prior to it. The same is true of the falling edges. Thus either edge can be used. Note that T gives the length or period of the wave, and T is fixed when the frequency of the oscillator has been chosen. Since the frequency tells us the number of pulses emitted per second, or here the number of rising or falling edges per second, the period of time between adjacent similar edges determines the unit of time on our fixed scale. Of course if we double the frequency, we cut the unit of time by two and this causes our measuring to be more precise.

Again in Fig. 1.3a, note that a pulse is formed by a voltage which may assume one of two possible values. It can be either +4 volts or 0 volts in magnitude. During the time interval 'a' the voltage is 0 volts, while during the interval 'b' it is +4 volts. When the time interval 'a' equals time interval 'b', the wave is said to have a 50% duty cycle. In this thesis, all wave trains have the 50% duty cycle. But other duty cycles could have been used.

Inspection of the wave train shown in Fig. 1.3a, indicates that zero time is not defined. In order to define a zero time, we start the oscillator. Now we must be able to count from this time zero via either the leading or trailing edges time intervals whose lengths correspond to the period of the oscillator. To do this we need an oscillator which when started will not put out its first leading or rising edge until one period elapses, as shown in Fig. 1.3b. Now each leading edge will correctly record the passage of time in intervals of our artificial clock. Let us call this clock a temporary clock, since it runs only when we wish it to run.

Thus we now have a scale in which we know time zero and we can accurately determine time if a time period ends on one of our clock edges. But how do we measure time between edges of two adjacent pulses ?

Consider Fig. 1.4. Here are depicted two wave trains of square pulses. Let the upper train of frequency f_o represent out time scale. The lower wave train has a frequency f slightly lower than f_o. Therefore the period of the lower train is slightly longer than that for the upper train. Assuming that we first look at the two wave trains at a time when the leading edges of the pulses in each train coincide, we can see what happens between the edges of pulses as time passes. Then let the pulses in each train be numbered starting with one, according to their occurence in

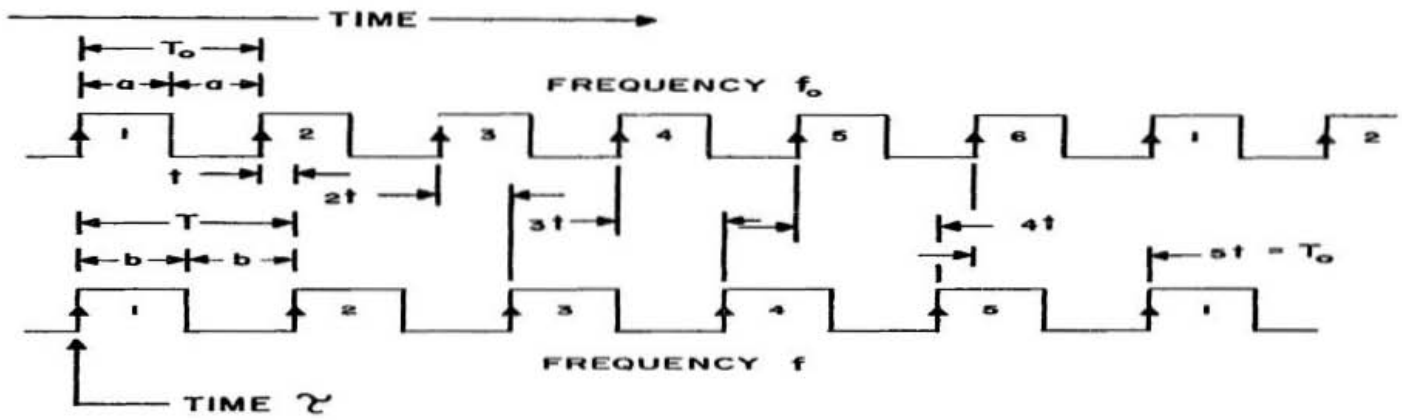


Fig. I.4
PHASE VARIATION WITH TIME FOR WAVE TRAINS OF FREQUENCIES f_o AND f

time. Again time flows from left to right in this diagram. Note that pulses 2, 3, 4, and 5 of the lower train lag progressively farther behind like numbered pulses in the upper train. A careful measurement shows that if the time difference between the arrival of pulses numbered 2 is t, then the differences bewteen pulses numbered 3, 4, 5, and 6 are 2t, 3t, 4t, and 5t respectively. But note that 5t is equal to the period of the wave of frequency f_o, and thus the two trains again have coincident leading edges. The difference in time between the occurence of two leading edges is termed the phase relationship between the two wave trains. Thus pulses numbered 1 have zero phase difference. Thus as time passes, the phase changes from zero phase to a maximum phase of one period, at which time we see zero phase again. But note that the phase changes by multiples of the first phase difference, i.e. that between pulses numbered 2.

If we can find a way to add these varying time intervals to our major scale, then we can measure the time of occurence of an event anywhere between either the leading or falling edges of adjacent pulses on our major scale by

1. adjusting the period of our master scale, and of course,

2. choosing the appropriate period and starting time of the second wave train, which will be called a vernier scale. Consider at some time τ, that one has counted up to some number N on the upper train and that the pulses numbered 1

in both trains in Fig. 1.4 have coincident leading edges. Then if one continues counting edges on the lower vernier train, the time increments that we are adding, can be altered by the amount t, where t = 1 ns here.

But again we will find that our vernier train of waves must commence when we decide it should commence. Thus we again need an oscillator which will give out its first leading edge one vernier period after the oscillator is started. This is necessary because we will count the edges of the pulses in the vernier wave train just as we will count the pulse edges of the master or major wave train, i.e. our system must count these edges since the system will be digital.

For the laser system that we were considering, we finally decided that one nanosecond (1×10^{-9} second) would be the smallest precise time interval to be measured. An extensive investigation of the variation of the phase differences between the edges of the pulses of two wave trains, as the frequency relationship between the two waves was varied, led to the decision that a frequency of 50 Mhz would be most appropriate to set the major time scale. This gave a major period or unit of 20 nanoseconds. Thus the oscillator frequency will be well within the speed capabilities of digital counters, these being TTL.

The vernier frequency was then set at $45.\dot{4}\dot{5}$ Mhz so that its period would be 22 nanoseconds. Thus this vernier has a resolution of 2 nanoseconds. If, as in the

case of the main oscillator, the vernier puts out its first leading edge 20 ns after being enabled, we can measure to an even number of nanoseconds. By adding the option of inserting a one nanosecond delay, we can also measure to an odd number of nanoseconds. Thus the use of the one nanosecond delay gives us an uncertainty which is not greater than 50 picoseconds.

If we had chosen the vernier frequency so that the differences in phase were multiples of one nanosecond, then we would have to employ nine vernier periods to resolve nine nanoseconds. By the method described in the foregoing paragraph, we can measure up to nine nanoseconds using a maximum of five vernier periods. Thus the time required to resolve the nanosecond units is greatly reduced.

Before proceeding, one must recall that for a spatial measurement, the vernier scale was positioned outside the interval being measured. Thus the zero line of the vernier terminated the interval being measured.

But here the vernier measurement must be completed concurrent with the end of the time interval being determined. This indicates the need for a reversed vernier. This aspect will be discussed in detail in Section 7.4.

Thus, this system will be different from contemporary systems, since the determination of a time interval will normally occur with the termination of the interval. However there is one case where the vernier measurement may come after the time interval being measured

has terminated, and this will be explained under the section entitled 'Automatic time interval evaluation (see Chapter 4).

CHAPTER 2

TIME CLASSIFICATION

2.1 Division of time

Each module of the time measuring system described herein is designed to cover a time range of from 1 to 9,999,999 nanoseconds.

Time is divided into three allotments: 1. hundreds, 2. tens, and 3. units of nanoseconds. The readout L.E.D.'s can display the decimal point to indicate the time in milliseconds but the system actually counts in nanoseconds.

The master oscillator frequency is 50 Mhz, establishing 20 nanoseconds as the period of the square wave. Five such periods will represent 100 nanoseconds. Thus if every fifth pulse is passed to a set of decade counters, they will register the passage of time in hundreds of nanoseconds. This will be illustrated under the section ' The hundred clock '. This leaves only the tens and units of nanoseconds to be counted

2.2 Two time counting divisions

Each module of the system is made up of two divisions. One division makes it possible to create required intervals of time, while the other division makes possible the determination of the time interval between two external events. Modules may be employed individually

or together in parallel or series.

2.2.1a. Time interval creation division

By means of this division, we can create one or two pulses per module to trigger external equipment. One can preset the module to process two time intervals:

1. the time interval before trigger pulse one occurs and,

2. the interval of time to exist between the first and the second trigger pulses.

In this division, trigger pulses can be separated by an interval as small as one nanosecond with an uncertainty less than 50 picoseconds.

2.2.1b. Cycling

The outputs of up to four modules may be operated either in series or parallel to give a wide range of time intervals between pulses, including the repetitive cycling of the pulses. Since each module can output two pulses, a maximum of eight pulses may be controlled.

2.2.2a. Measuring external time intervals

This division will measure the reaction time of an external system to a stimulus, i.e., the time of travel of a pulse along a cable of given length. It will also measure the interval of time between two external events, i.e., the time interval between the light flashes of two lasers. Again the degree of precision in making this determination is the same as for Section 2.2.1a.

Since this division is triggered by the leading edge of the pulse, the width of the pulse does not affect the accuracy of the determination of a time interval between two pulses.

2.2.2b. Parallel mode - automatic synchronization

When two or more modules are operated in parallel, up to four time intervals can be measured concurrently providing the means to automatically synchronize the operations of up to at least four devices, while a maximum number of eight is possible.

2.3 Modular time

Fig. 2.1 depicts a time scale in nanoseconds. The maximum number of digits for an input number is seven. We require one decade counter for each digit. Thus seven decade counters are required. For this reason, in the second line of Fig. 2.1, the time is parcelled out as powers of ten. Line three shows the three divisions for a module.

Fig.2.2 indicates the order in which the system handles the three time divisions for the time interval creation division. These time divisions are handled in time in the same sequence as they appear in Fig. 2.2, from left to right. Notice that 200 nanoseconds are cut out of the total hundreds count, and the tens are added after the units of nanoseconds have been established. The relationships between the 200 nanosecond subdivision and the units of nanoseconds as well as the tens of nanoseconds are carefully laid out under the section entitled ' Phasor vernier relationship '. Note that here the vernier is the

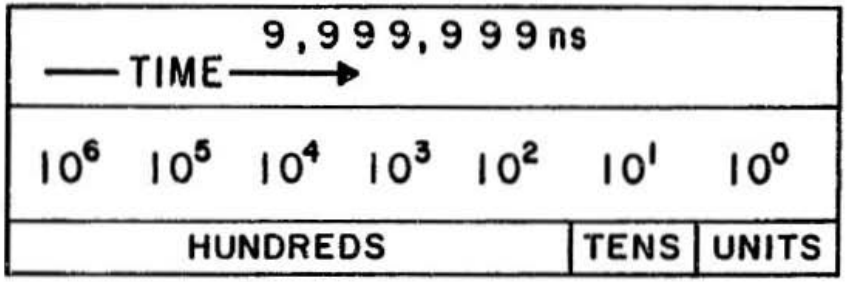


Fig. 2.1 CLASSIFICATION OF TIME

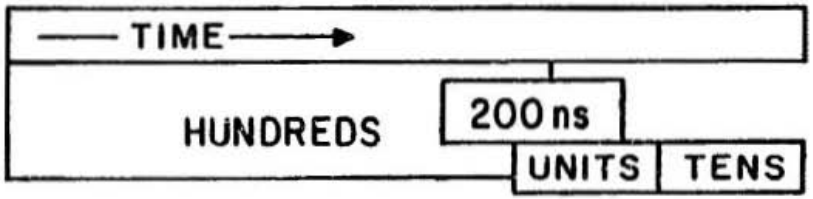


Fig. 2.2 TIME SEQUENCE, TIME INTERVAL CREATION

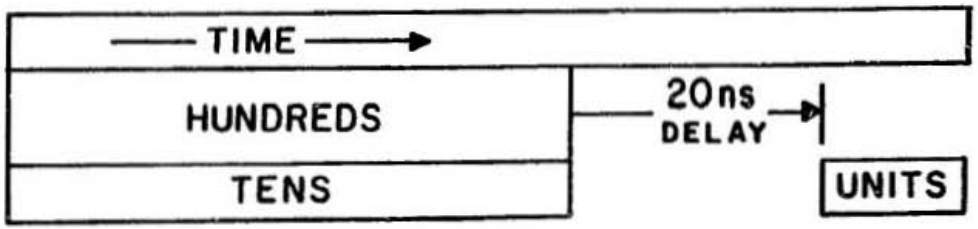


Fig. 2.3 TIME SEQUENCE, TIME INTERVAL EVALUATION

reversed vernier and it is also a sliding vernier.

For the time interval measuring division, the divisions of time are as shown in Fig. 2.3. Note that the hundreds and tens are determined concurrently, while the units are decided later. The units here are found by employing a delayed vernier. The precision of measurement is still one nanosecond with a discrepancy not greater than 50 picoseconds. This time system will be explained under the section ' Automatic time interval evaluation division.

As we have earlier noted, the present verniers are digital and thus are quite different from the verniers that are currently used in the electronic field.

CHAPTER 3

GENESIS OF A TIME MODULE

In the early planning stages, the task was to create a system which would provide an experimenter with the means of measuring off short intervals of time with high precision. Once the size of a required interval was established by calculation or estimation, it was entered into the system manually. This system is referred to as the time interval creation division.

It was realized that the above mode of operation would be time inefficient and to a degree uncertain because the operator would arrive at an optimum value of a time interval for a specific need by the hit and miss method of time variation and trial.

To provide a time efficient and more accurate system, the scope was broadened to include an automatic means of determining the required time interval together with automatic insertion of this data into the time interval creation division referred to above. The system, designed to achieve this result, is referred to as the time interval evaluation division.

A commencement was made with the study of digital decade and binary counters. The binary truth tables for these counters were reviewed. But this undertaking did not lead very far. It was necessary to observe the states and the changes in the states of the various sections of the

counters. To illustrate the problem, it will be well to consider a counter and to describe briefly its various functions and how these functions can be employed.

Digital counters: A digital counter is depicted by the rectangular box in Fig. 3.1a where the input, output, and supply terminals are shown as numbered black dots, the numbers ranging from 1 to 16. The correct orientation of such a circuit is indicated by a dot or small square impressed on the upper surface of the circuit package near its left hand edge just above the first terminal numbered 1. Fig. 3.1a is usually replaced by the schematic diagram of Fig. 3.1b. Here the first terminal, numbered 1, indicates the lower left hand corner of the package.

This diagram represents the 74S168 circuit, a fast TTL decade up-down digital counter, and all the terminals are named. The supply voltage for terminal 16 is +5 volts, and terminal 8, the ground terminal is identified by the symbol $\doteq$.

The remainder of the terminals can assume one of two voltage levels, namely 0 or +4 volts. These levels are shown diagrammatically in Fig. 3.1c. If the voltage state is 0, we simply draw a straight horizontal line which is our zero axis, as from a to b, while if the voltage state is +4, we draw a wide line above the zero axis, as from b to c, where the width of the line represents +4 volts. By means of such voltage level diagram, we can relate the voltage states of the different terminals to oneanother in

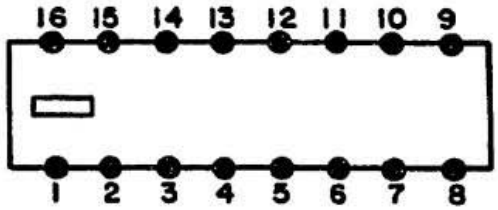


Fig. 3.1a PACKAGE DIAGRAM

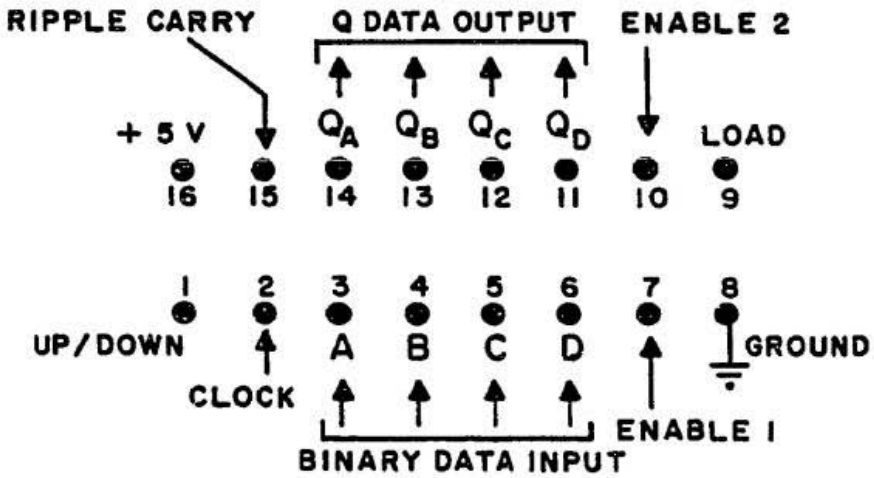


Fig. 3.1b SCHEMATIC DIAGRAM

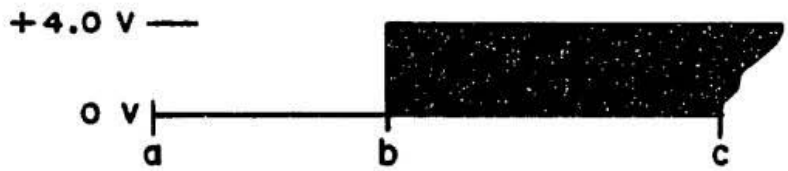


Fig. 3.1c VOLTAGE LEVEL DIAGRAM

time. Thus we generate a voltage-level state diagram for a static clock. If we wish to show these voltage-level state relationships as clock pulses arrive at the input clock terminal of the counter, we call the resulting diagram a pulse-time diagram. By means of this latter type of diagram, we can pictorially display how a circuit functions with time.

Returning to the digital counter, Fig. 3.1b, the functions performed by its various terminals may now be investigated (2).

Terminal 1 decides whether the counter adds or subtracts. The direction of counting is decided by the voltage applied to this terminal. If the voltage is +4 volts the counter counts up from 0 to 9, while if this voltage is 0 volts the counter counts down from 9 to 0, i.e. it continually subtracts one until zero is reached.

Terminals 3, 4, 5, and 6 are data input terminals for the binary data. They are lettered A, B, C, and D. Since they are usually shown on a schematic diagram in this order, I will lay out all my binary truth tables in this order so that they can be easily related to circuits to be presented in the explanation of the system described in this thesis. Also, since this is a design thesis, this approach is not out of order.

Table 3.2a gives the binary code for the digits 0 to 9, while Fig. 3.2b shows the voltage states for the data inputs A, B, C, and D for each of those digits. Thus

we can compare the data inputs both numerically and electronically. Note: In this thesis the order of the binary code is reversed from the standard order, i.e. the number 2 is 01.

In Fig. 3.1b, terminals 14, 13, 12, and 11 are lettered Q_A, Q_B, Q_C, and Q_D respectively and they are data output terminals. They can also serve as data storage and in this thesis will be called "Q data". They can be considered storage because if the clock input becomes inoperative, whatever states these terminal voltages are in at that time, those states are fixed. Thus we can use them as temporary storage space, and much use will be made of this property of counters in the design of this system.

Terminals 7 and 10 are called enable inputs. If both of these terminals see 0 volts, then the counter can count the clock pulses arriving at the clock input terminal and the clock is said to be enabled. If one or both of these terminals see +4 volts or any voltage greater than +0.8 volts, then the clock input cannot respond to any clock pulses, and the clock is said to be disabled (6).

Terminal 2 is the clock input to the counter.

The data which appears at the input data terminals is of no use to the counter until the counter sees that data in its Q data storage areas. This means that the data at the input terminals must be transferred to the Q data storage areas. Once this transfer of data has been made, then as the counter receives clock pulses, the Q data

DIGITS	A	B	C	D
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	1	1	0	0
4	0	0	1	0
5	1	0	1	0
6	0	1	1	0
7	1	1	1	0
8	0	0	0	1
9	1	0	0	1
	A	B	C	D

Fig. 3.2a NUMERICAL BINARY CODE DIGITS 0 TO 9

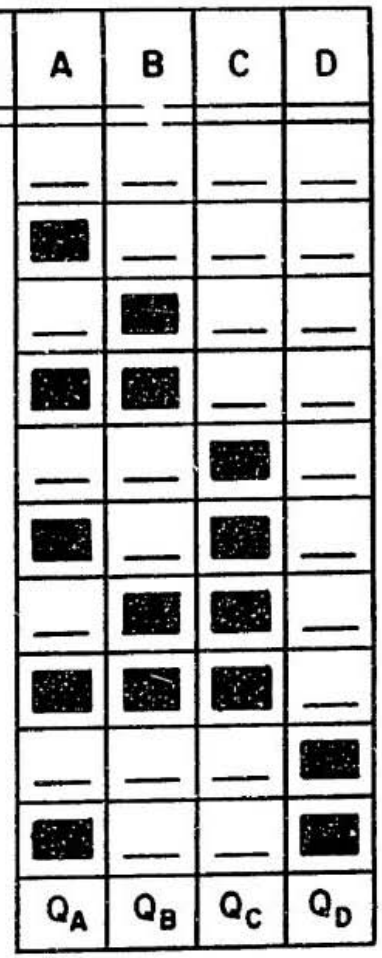


Fig. 3.2b INPUT AND Q OUTPUT DATA VOLTAGES, DIGITS 0 TO 9

follows the clock as shown in Fig.3.2b.

When the Q data storage assumes the states for 1 in the down counting mode or 9 in the up counting mode, then by means of the next clock pulse, the counter automatically resets its Q data storage to the proper states for the digits 9 or 0 respectively, using internal circuitry. At the same time that the states for 1 or 9 appear as Q data, terminal 15, called the ripple carry terminal, which is usually at +4 volts, changes its voltage state to 0 volts and on the next clock pulse returns to +4 volts, which voltage level will be fixed until the states for either 1 or 9 appear again in the Q data storage. Thus the counter can reset itself. But this resetting of the Q data may happen when we do not wish it to take place. For example, if we wish the counter to count by fives, in the down mode, then we set the data at the data input terminals to represent the number 4 and tie terminal 15 to terminal 9. Terminal 9 is termed 'load'. Now when terminal 15 changes its voltage state from +4 volts to 0 volts with the advent of Q data storage states for the digit 1, it causes the voltage of terminal 9 to change from +4 volts to 0 volts. This action is referred to by saying that terminal 15 has pulled down terminal 9, and it causes the counter to load its Q storage data to 4 on the next clock pulse. But it is most important to realize that terminal 9 was made to assume the 0 volts level well in advance of the next clock pulse in order for the clock input to be able to see the

rising edge of the resetting clock pulse. It is fully illustrated under the section ' Hundred clock ' why the counter was set to the number 4.

Again, we connect terminal 15 of our counter to terminal 10 of a second counter. If the clock input of this second counter is connected to the same clock line as the clock input of the first counter, this second counter will receive clock pulses synchronously with the first counter. Thus when terminal 10 of the second counter is pulled down by terminal 15, the second counter may count one pulse,- every fifth pulse as seen by the first counter. That is, the second counter counts every fifth pulse synchronously with the clock, and no time delays inherent in either counter has any detrimental effect on the counting. Although not obvious at first glance, the first counter does not count the fifth pulse, but simply uses it to reload its input data into its Q data storage.

Thus by employing terminal 15 and terminal 9, we can load our counter to any desired number from 0 to 9 for a decade counter, and from 0 to 15 for a binary counter, and again use terminal 15 to enable a second counter to record the next clock pulse by means of the resetting pulse.

Now this is fine if our system is already functioning as we have programmed it to function. But how do we start the system properly ? This is important because we do not want false counting to occur. For example, suppose we preset the data input terminals for the

data representing the digit 4 and then enable a pair of counters - counters with their clock inputs connected to the same clock line so that they will count synchronously, as mentioned above. What will our first counter do ? Well, its Q data states may be zero. If they are zero, the first pulse to the clock input terminal will load our input data into the Q data storage areas of the first counter, but the second counter will also see this loading pulse and count 1, which would incorrectly indicate that 5 pulses have taken place. But of course, the Q data storage areas may contain any set of states, which means that the incorrect count would also occur at some other time. Thus we must insure that the Q data storage of the counters assumes the input data states before counting starts.

To insure that the counters are loaded before counting starts, we make use of terminals 9, 10, and 2. We apply a low state voltage, 0 volts, to terminal 9 and sustain the high state voltage, +4 volts, at terminal 10 so that the counter cannot count, and then supply one clock pulse to the clock input terminal 2 in order to load the input data into the Q data storage areas. Now that the input data is correctly stored in the Q data areas, our counters will count accurately, when we enable the counters.

Flip-flops. These form a family of integrated circuits, and their importance to this system is next to that of the counters.

There is quite a wide range of circuits so named. Here I will refer to only two types, which will be employed in this system, which may function on the leading or trailing edge of the clock pulse.

In addition to a clock input, these devices have control inputs which may include any of the following inputs: clear, preset, J and K, and D.

They usually have two output terminals such that the voltage at one terminal is the inverse of that at the other terminal, i.e. +4 volts and 0 volts. These outputs are lettered Q and $\overline{Q}$ respectively.

The clear input is used to set the flip-flop so that the Q voltage output is 0 volts, while the preset input sets this output to +4 volts. Both of these settings are brought about by applying a narrow low state or zero voltage pulse to each of these terminals separately.

The D type flip-flop has the D input. Let the voltage of the D input be +4 volts. Then suppose that the D input voltage is allowed to become 0 volts between clock pulses while the Q output voltage is +4 volts. Then on the following or next clock pulse, the Q output voltage will change to 0 volts and disable the clock input, so that further clock pulses have no effect on the output states.

Again if the D input voltage is allowed to become +4 volts while the Q output voltage is 0 volts, then on the next clock pulse, the Q output voltage will change to

+4 volts. Again the clock input becomes disabled.

In either of the above two modes, the flip-flop acts as a switch which we can control by means of pulses or voltage levels supplied by other electronic equipment.

If the D input terminal is tied to the $\overline{Q}$ output terminal, then the flip-flop can function in such a manner that the Q output voltage can continually change back and forth from +4 to 0 volts with each succeeding clock pulse. When the D input is so maintained, the action of the flip-flop is referred to as toggling.

The device having J and K input controls is called a 'J-K Flip-flop'. Now if both of the J and K input voltages are maintained at +4 volts, this circuit toggles with each clock pulse arriving at the clock input, just as the D type flip-flop toggles. Note:- this flip-flop has two input control terminals and not one as in the case of the D type flip-flop. Thus we have an extra means of controlling the operation of this circuit.

If when the Q output voltage is 0 volts, and the J and K input voltages are +4 and 0 volts respectively, then the Q output voltage will change to +4 volts on the next clock pulse, and the clock input becomes disabled.

Likewise, if the Q output voltage is +4 volts when the J and K input voltages are 0 and +4 volts respectively, then the Q output voltage will change to 0 volts with the next clock pulse, and the clock input is disabled.

Now by the use of flip-flops, we can establish voltage levels which can be employed to control other circuits such that those circuits are active or idle.

But it is also possible to load or control the output volatges of flip-flops by employing the pulses put out by counters.

Next it was necessary to decide how the input time data would be handled. It was now a simple matter to count the hundreds of nanoseconds, or so it appeared at first glance.

The next step was to decide how the vernier scale was to be implemented. For instance, how and where would the vernier be started in order to determine a given unit of nanoseconds ? How long will it run and how will its operation be terminated ? All of these details are dealt with under the section 'Phasor-vernier relationship'.

Now in order to provide time for the vernier to function, 200 nanoseconds are cut off from the total hundreds of nanoseconds, as illustrated in Fig. 2.2. Once it has been decided where in time the vernier should start and how long it should run, the time data required for this to occur had to be rewritten in digital form, thus causing the creation of a machine language for the module.

Also under the section 'Shifter' an explanation is given as to why the tens of nanoseconds were to be added to the phasor data and thus bring about a sliding vernier.

Thus a module has a built in program that transforms the time data inserted into it. Originally digital switches were to be employed as the only means of inserting the time data for a proposed time interval. This time data was then to be transformed into machine language and stored in the Q data storage sections of the hundred, phasor, and vernier counters. It was then only necessary to trigger the system, whence the system would count down the time and put out a termination pulse at the proper time. Of course, to accomplish this, the hundreds counter counts down the hundreds and when 200 nanoseconds remain to be counted, the phasor starts. Now the phasor will count down until its Q data storage represents the digit 1, whence it will start the vernier. When the vernier's Q data storage represents the digit 1, it will put out a termination pulse.

In theory this was fine, and even during the temporary mock up, the system functioned perfectly for all time data inserted. But a careful investigation of time data revealed the need for more controls. What if there were no hundreds, or only one hundred period to count ? How would the phasor start ? How would the system arrange itself in order to function properly ?

As a solution to this problem, the Q clock and the Three clock were created and they function jointly with a dual option gate circuit.

Finally a complete module was designed that would meet all the time data requirements.

But there was one drawback to the system. Each time it was operated, it used up and lost the updated time data. This meant the input data would have to be transformed again after each run, thus limiting the rate at which re-runs may occur. To overcome this difficulty, the module was divided into two parts, A and B, where A is the active part used only during the actual interval determination, and part B is the passive section or machine language transformation division. Now section B transforms the input time interval data and stores it in its output storage areas so that the updated data can be loaded repeatedly and at high speed into the data inputs of the active section A.

Since the original requirement was to control the operation of three and possibly four pieces of equipment, of which two were lasers, a four channel system was formed by parallelling four modules.

But this called for a control section which could monitor each module automatically by giving out control pulses to each module and monitoring the output pulses from all modules, so that the system could reset itself after each run, preparing the system for a repeat run.

Note that up to this point in the design stage, all time data is to be inserted into the system by means

of digital switches. Of course the experimenter would have to calculate the time intervals to be inserted into each module.

Since it was necessary to determine the time interval between the occurence of discharges of two lasers, it was realized that some means was required to measure this interval of time. Once this time interval was known, it could be inserted directly into a module so that the two lasers might be fired automatically by the system and thus bring about the synchronization of the laser discharges without the need for the experimenter to calculate the interval by trail and error.

Thus the automatic time interval evaluation division was created and it is described in the section bearing that title.

To achieve a completely automatic system, an arithmetic logic unit was designed, which is in reality a small computer. By means of this unit, the time interval relative to each channel can be investigated to find the maximum time interval and also the differences in time between the maximum time interval and the remaining time intervals. Then this maximum time interval and the time differences are used to give a completely automatic time delay system. I have named the system:

'The precise time interval system'.

CHAPTER 4

THE AUTOMATIC TIME INTERVAL EVALUATION DIVISION

This division permits the direct measurement of time intervals between two or more events anywnere from one nanosecond to approximately 10 milliseconds in length with a discrepancy not greater than 50 picoseconds.

Fig. 4.1 is a block diagram indicating how the division functions. Just prior to timing an interval, the operator activates a set button which initiates the following:

1. The 'Control' unit puts out to this division a broad low state pulse A_o and a narrow high state clocking pulse B_o within the lifetime of pulse A_o. These two pulses appear at terminals numbered 9 and 2 of the five 'Hundred' counters and the 'divide by 4' counter respectively. These two pulses cause the Q data storages of these two sets of counters to assume the states for the digits 0 and 4 respectively.

2. B_o also sets the states of clocks A and B to the states for the digit 0, as well as presetting the pulse delay ciruit which pulls down 'enable 2' so that the five 'Hundred' counters are partially enabled.

3. The inverse of B_o, namely $\overline{B_o}$, is used to clear all the delay line output flip-flops so that their Q states are zero

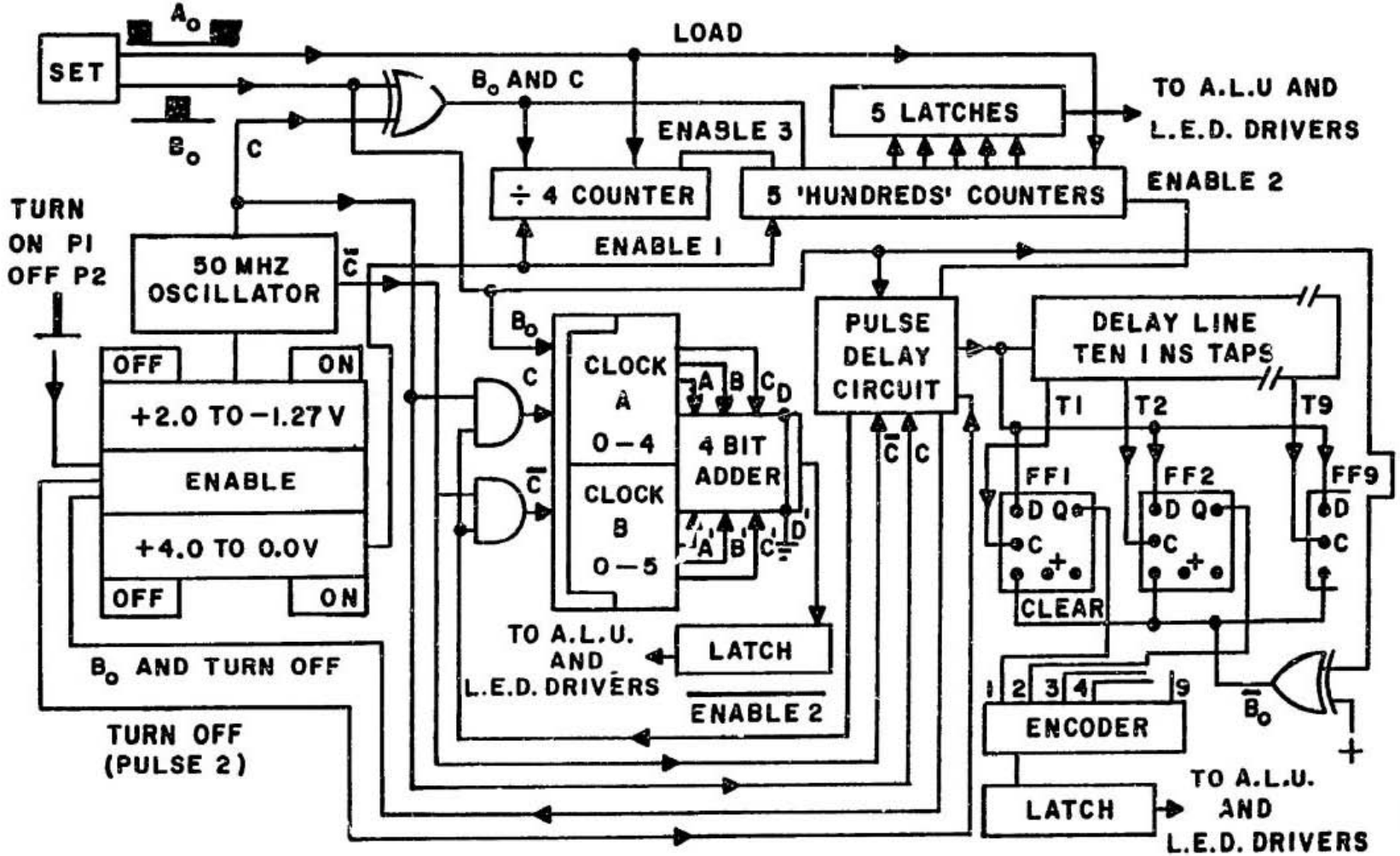


Fig. 4.1 AUTOMATIC TIME INTERVAL EVALUATION DIVISION

The system is now ready for the initiating pulse.

When the initiating pulse comes, it goes to the oscillator enable circuit. One half of this circuit now changes the oscillator's off-on control voltage from +2 to -1.27 volts, i.e. it pulls down the control voltage, and thus enables the oscillator. But at the same time, the other half of this circuit changes 'enable 1' control voltage from +4 to 0 volts, and thus fully enables the five 'Hundred' counters and the 'divide by 4' counter.

Note, that the first pulse received by the oscillator's enabling circuit arranges this circuit's input so that the next initiating pulse has no effect on this circuit, but it is passed directly to the pulse delay circuit.

The 'divide by 4' counter will now count down 4 pulses and put out the fifth pulse as a low state resetting pulse to the first counter of the five 'Hundred' counters as an 'enable 3' pulse, so that this counter may count one pulse synchronously with the clock to represent 100 ns. This 'divide by 4' counter uses this resetting pulse to reload its Q states for the digit 4 and thus continually recycle itself.

But clock A and the 'divide by 4' counter count synchronously the pulses put out by clock C, and clock A puts out a low resetting pulse to itself as well as to

clock B. This clock B counts up five pulses, from the inverted clock before it is reset to zero by the resetting pulse from clock A. Note that clock A counts the normal 50 Mhz pulses, while clock B counts the inverted 50 Mhz pulses. Consequently, as shown in lines 5 and 7 of Fig. 4.4, the states of clock B change at the mid-point of the states of clock A. The only exception occurs when the reset pulse arrives, causing the Q states of clocks A and B to change from the states for 5 and 4 respectively to the states for 0. Again because clock B changes states in the middle of the inverted clock period, states 5 and 0 last for half the 20 nanosecond period. Using this property, that the states of clock B change in the middle of the states of clock A, we may add the states of these two clocks as shown in line 10 of Fig. 4.4 to obtain the tens of nanoseconds. Fig. 4.2 is the schematic diagram for clocks A and B and also their resetting circuit.

We will now show how the states of clocks A and B are set. Consider Fig. 4.3. Now consider similar numbered lines in parts 'a' and 'b' of this figure. Note that lines numbered 1 indicate the ordering of pulses, and line 2, 'reset', shows the narrow resetting pulse for these two clocks.

Lines 3, 5, and 7 show the states required by Q_A, Q_B, and Q_C for each clock, in order to obtain the proper binary conditions for digits 0 to 4 and 0 to 5 in parts 'a' and 'b'.

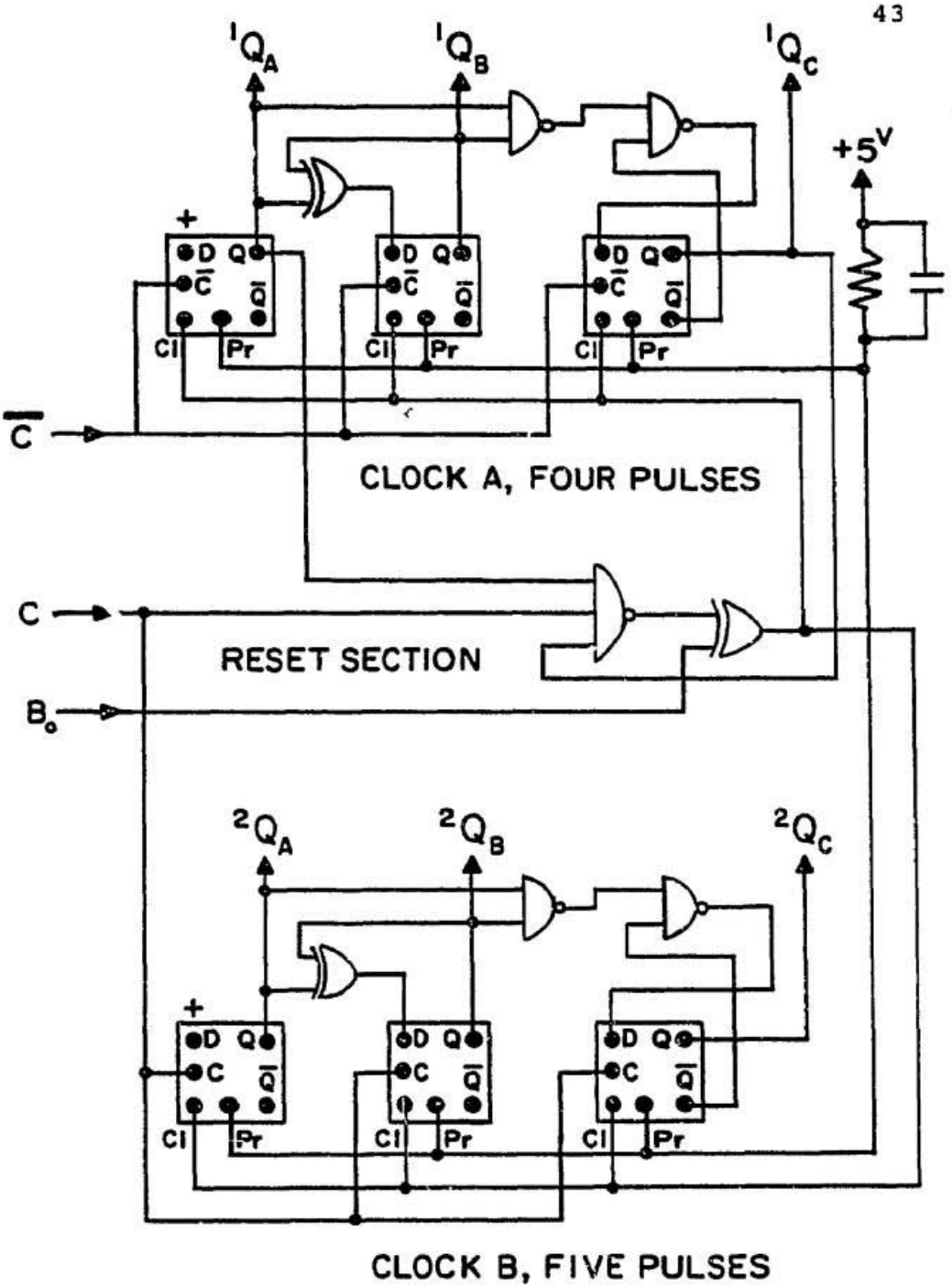
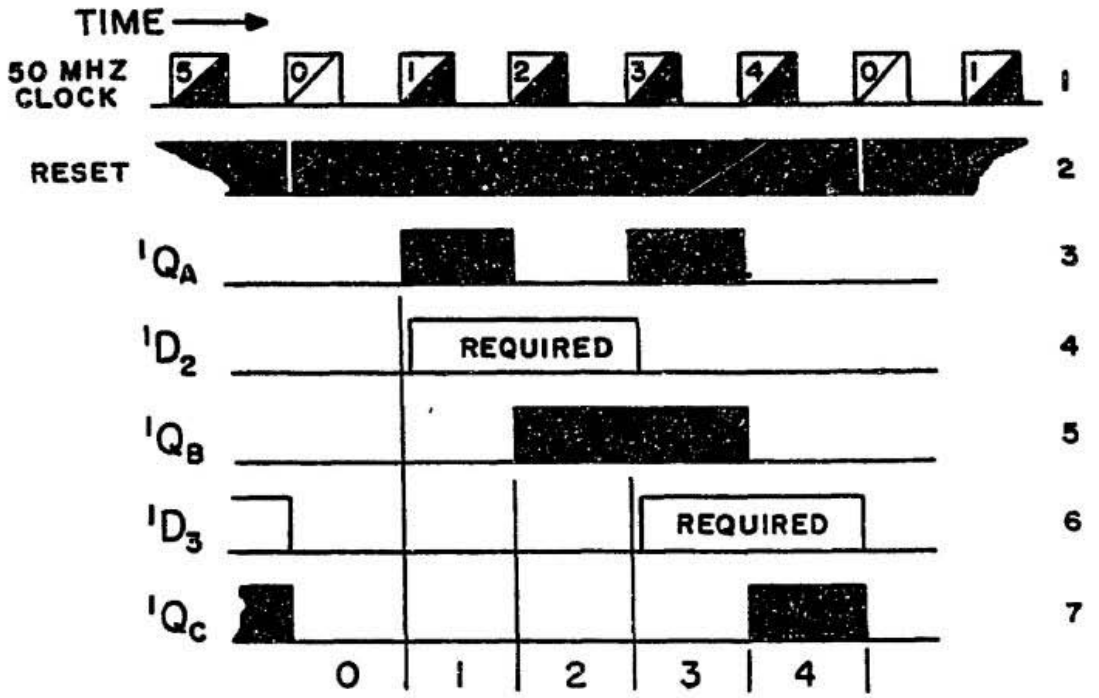
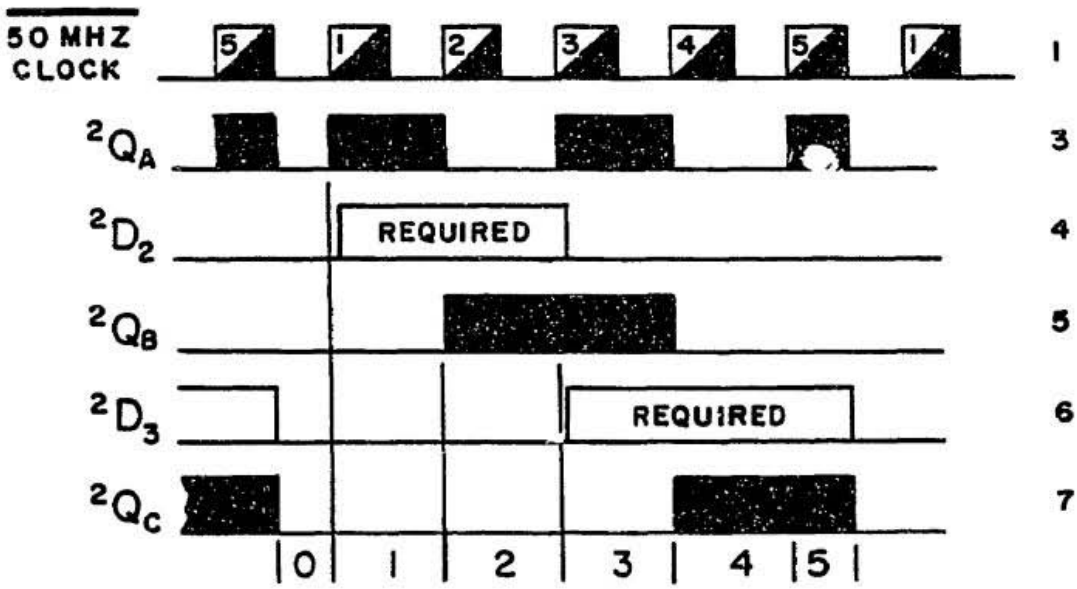


Fig. 4.2

SPECIAL AUTOMATIC SYNCHRONOUS CLOCKS



(a). CLOCK A, FOUR PULSES



(b). CLOCK B, FIVE PULSES

Fig. 4.3 Q DATA FOR SPECIAL AUTOMATIC SYNCHRONOUS CLOCKS

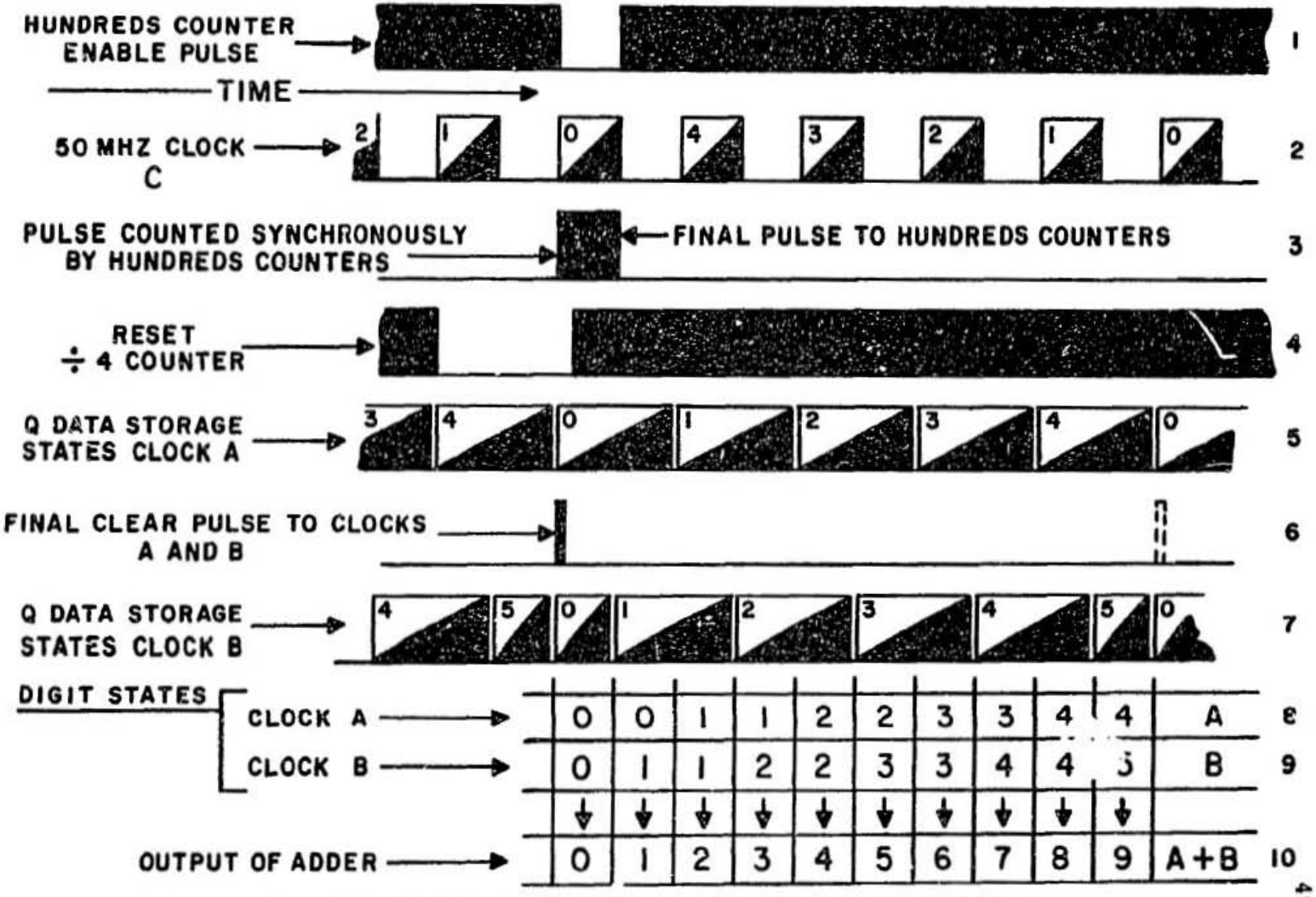


Fig. 4.4 TO FIND THE TENS OF NANOSECONDS

Lines 4 and 6 show states that these two circuits must see in each case to function properly. Clocks A and B, see Fig. 4.2, were designed to internally generate these two states.

Also note that in part 'b', both of the 0 and 5 digit states are states whose lifetimes are only half the length of those for the states for the digits 1, 2, 3, and 4.

We now consider the details involved in adding the states of clocks A and B to obtain the tens of nanoseconds. Fig. 4.4 shows the main oscillator C as reference and also gives the output states of clocks A and B relative to each other as well as to the clock C.

Line 4 of Fig. 4.4 shows the reset pulse for the 'divide by 4' clock which establishes the hundreds of nanoseconds.

Consider line 5. Here the state of clock A is 0 = (0 0 0 0) from the leading edge of the reset pulse line 6 to the leading edge of oscillator pulse numbered 4 in line 2. The state of clock B is 0 = (0 0 0 0) from the leading edge of the same reset pulse to the first leading edge of the inverted clock. Since these two states coexist for only 10 nanoseconds, they add to give zero tens of nanoseconds for the next 10 nanoseconds. In this manner, all the tens of nanoseconds are achieved. This is all summarized in the table given by lines 8, 9, and 10 of Fig. 4.4.

This sum is accomplished by the adder ciruit which occurs in schematic diagram Fig. 4.1.

In summary, first, it was necessary to use a 'divide by 4' counter to change the period of the clock pulses from 20 to 100 nanoseconds to obtain the 100 nanosecond count. Then, second, it was necessary to create two special clocks, A and B, to find the tens of nanoseconds, because the resetting pulse from the 'divide by 4' counter, line 4, could not be used to reset clock B. This pulse would lock out state 5 in line 7 if a decade counter were used, because line 4 which represents the ripple carry pulse used to reset the 'divide by 4' counter assumes a low state on state 4 which stays low until the next clock pulse. Also, the reset pulse from clock A, line 6, could not be used to reset the 'divide by 4' counter, because the rising edge of the clock has passed before terminal 15 would be pulled down by this pulse which is very narrow in time. Terminal 15 must be pulled down well in advance of the clock pulse used to reset the counter, in order for the counter to see the pulse's rising edge since this counter clocks only on the rising edge of that pulse. Therefore, the 'divide by 4' and the A and B clocks will have to be jointly presynchronized by the presetting pulse B_o.

Now as long as no termination pulse is received, the hundred counter will advance its count by 1 for each 100 nanosecond interval. The outputs of the adder for

clocks A and B will continually cycle from 0 to 9. Time problems due to the delay time inherent in the adder to respond to external stimuli will not cause any trouble because the adder output is not looked at until the interval determination has been completed, as will be amply clear shortly.

When the termination pulse comes, it goes to the pulse delay circuit, and performs the following functions. Consider Fig. 4.5. The Q output terminal of either flip-flop, FF1 or FF2, may change its voltage state from 0 to +4 volts depending upon which flip-flop sees +4 volts at its J input terminal. This will depend upon which of the clocks, C or $\overline{C}$, is in the +4 voltage level at the time the termination pulse occurs. Here note must be made that the Q output voltage states for these two flip-flops were both set to 0 volts by the set pulse B_o mentioned at the beginning of this chapter.

When a Q voltage state change takes place as mentioned in the foregoing paragraph, the output voltages of EXCLUSIVE-OR gates X2 and X3 are forced to change. The X2 gate changes its voltage state from 0 to +4 volts, while the X3 gate changes its voltage state from +4 to 0 volts. Thus the 'Hundred' and 'divide by 4' counters together with clocks A and B are all disabled. Thus the outputs of the 'Hundred' counters and the adder are now fixed, and may be read at leisure.

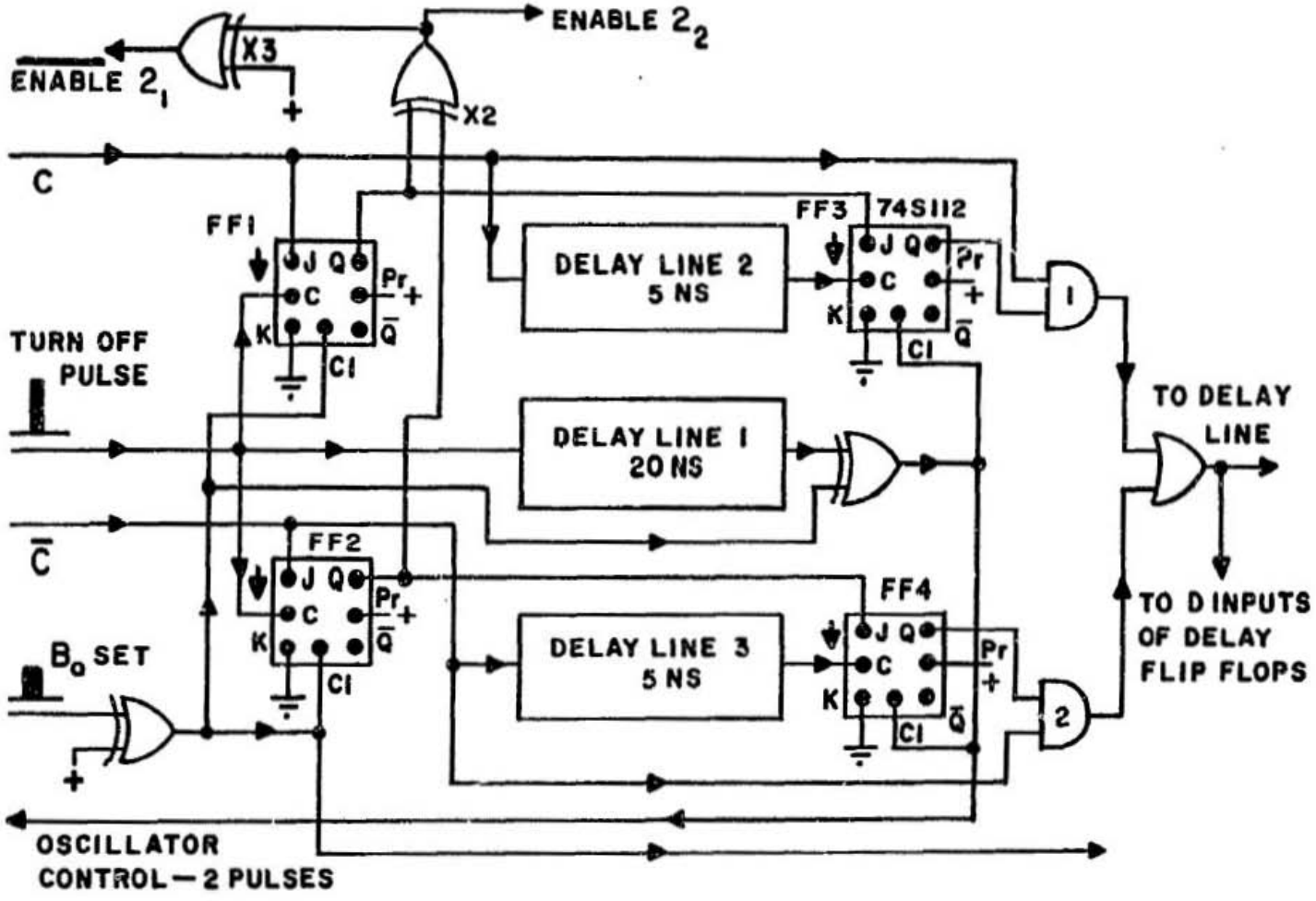


Fig. 4.5 PULSE DELAY CIRCUIT

Now referring to Fig. 4.6, consider the diagram which is actually a pulse-time chart. The top line shows the clock pulse available to a J input terminal of either of the flip-flops, FF1 or FF2. The diagram is true whether we consider the high clock state to be from clock C or clock $\overline{C}$, when the termination pulse occurs, because this circuit is symmetrical with respect to both these clocks.

Assume that clock C is in its high voltage state when termination occurs. Then Q_1 and J_3 will become positive due to the falling edge of the turn-off pulse. Note that flip-flops FF1 and FF2 change state with the falling edge of the clock pulse. Now flip-flop FF3 is set ready, and on the next falling clock edge, which is delayed 5 nanoseconds, Q_3 will change from 0 to +4 volts and Q_3 will hold this voltage state until the turn-off pulse exits from delay line 1 to clear this voltage state to 0 volts. The result of this action is that the time interval to be measured now appears as a delayed pulse whose width is equal to the interval of time between the two events.

This delayed pulse now passes into the final delay line, which has nine one nanosecond taps. At the same time, this pulse appears at the D input terminals of the delay line storage flip-flops - type 74S74 - while the outputs from the delay line taps appear sequentially in time at the clock inputs of these same flip-flops, at one

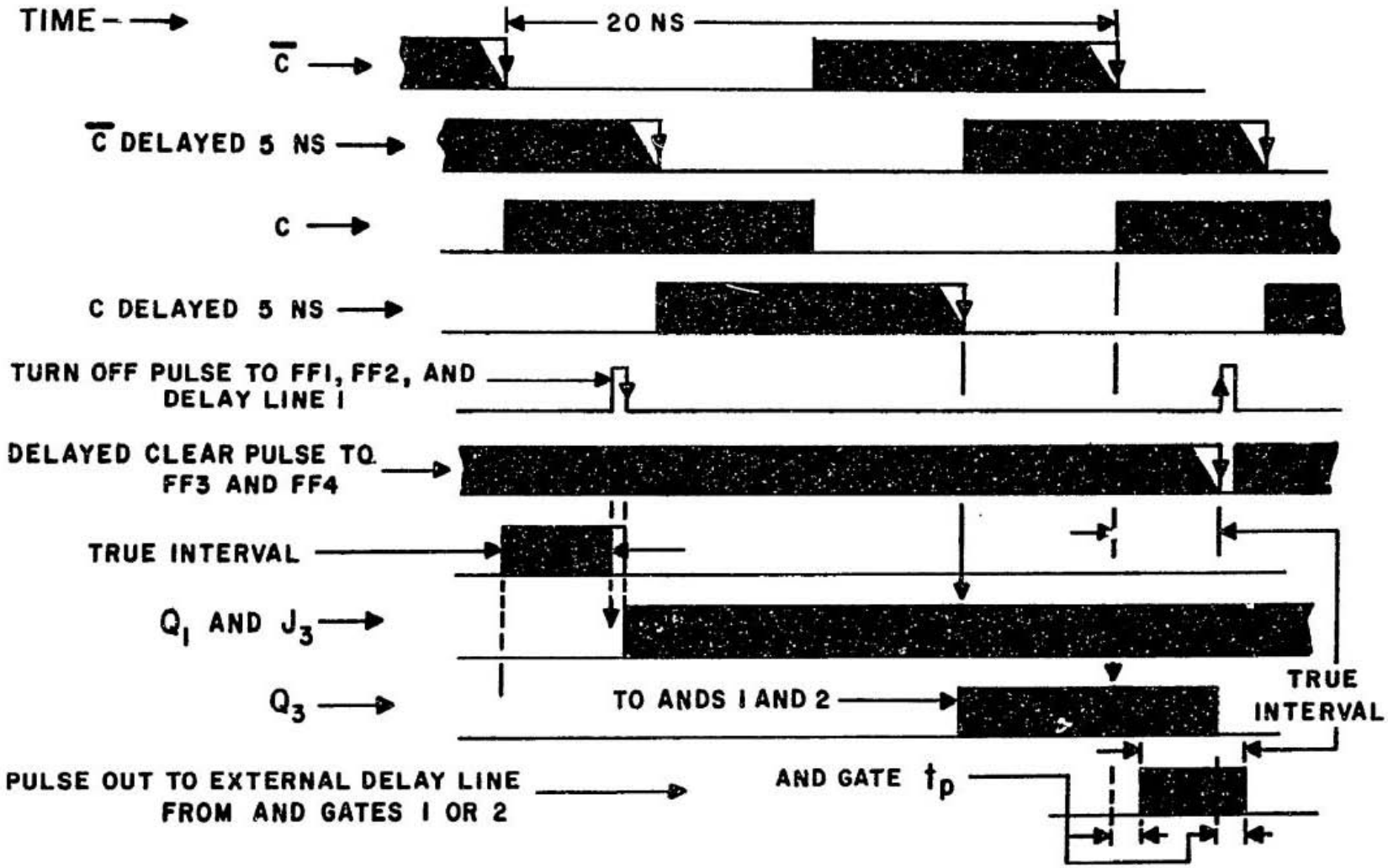


Fig. 4.6 PULSE TIME DIAGRAM FOR DELAYED PULSE CREATION CIRCUIT

nanosecond intervals. As long as the voltage at the D input terminals is positive, the delayed outputs of the delay line can set the Q output voltages to +4 volts. As soon as the delayed pulse vanishes from the D terminals, so that their voltage states become 0 volts, no further changes of the Q output voltages can take place. The width of the delayed pulse is now stored in the Q voltage states of these flip-flops. An encoder now sees all these Q voltage states, and its output will read from 0 to 9 and thus give the individual nanoseconds.

After the termination pulse has passed, and the units of nanoseconds have been decided, a loading pulse is delivered to all latches which then store the time data and make it available to the second input of the arithmetic logic division.

Fig. 4.7 indicates how the Q voltage states are seen by the encoder to give the units of nanoseconds.

Fig. 4.8 is the schematic diagram showing the construction of the 'Hundred' clock using five decade counters. Note that the section of the circuit labelled 'Five clock' is the 'divide by 4' circuit. Also note that this clock counts up since terminal 1 of each counter sees +4 volts.

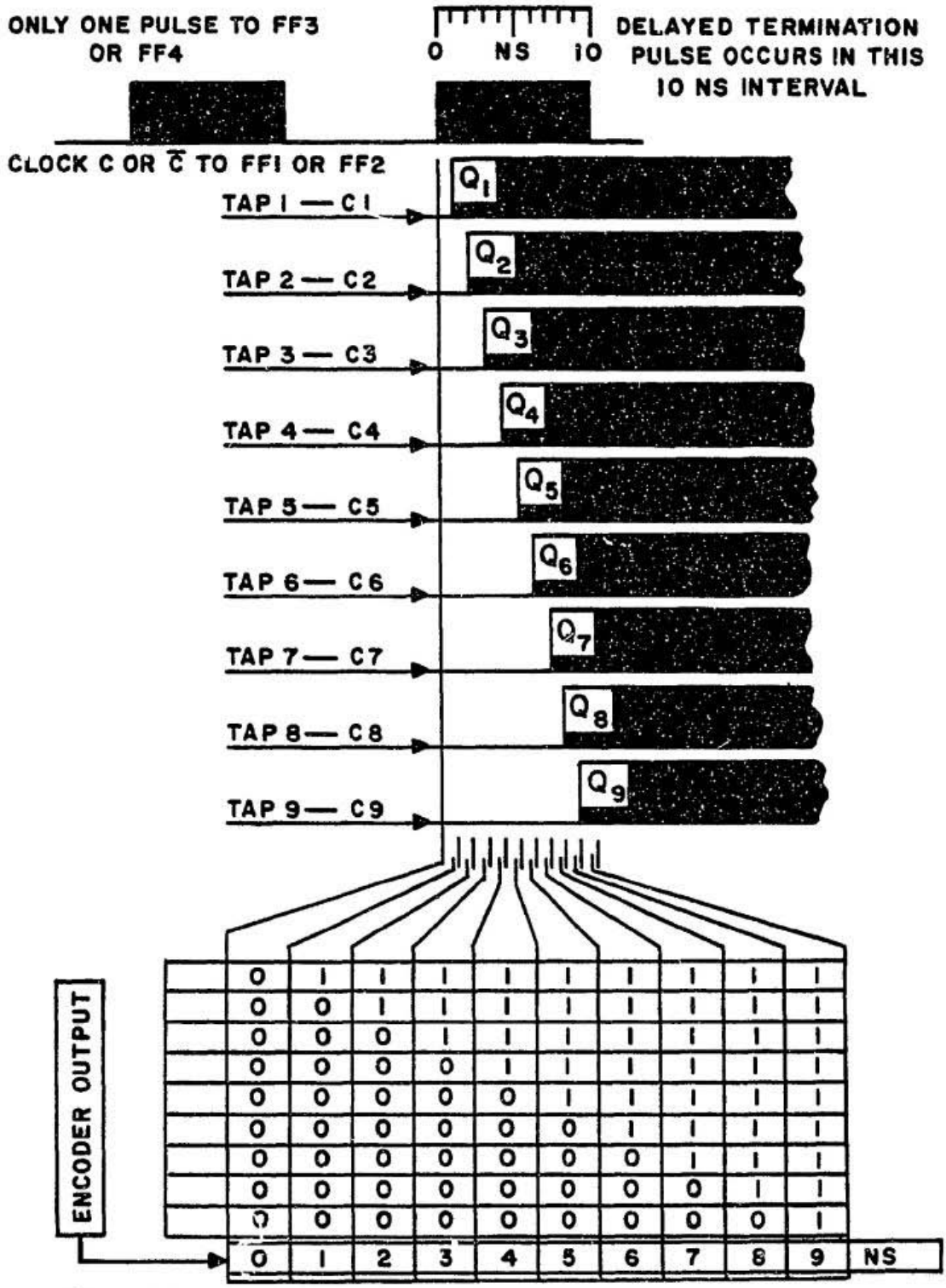


Fig. 4.7
DETERMINATION OF UNITS OF NANOSECONDS

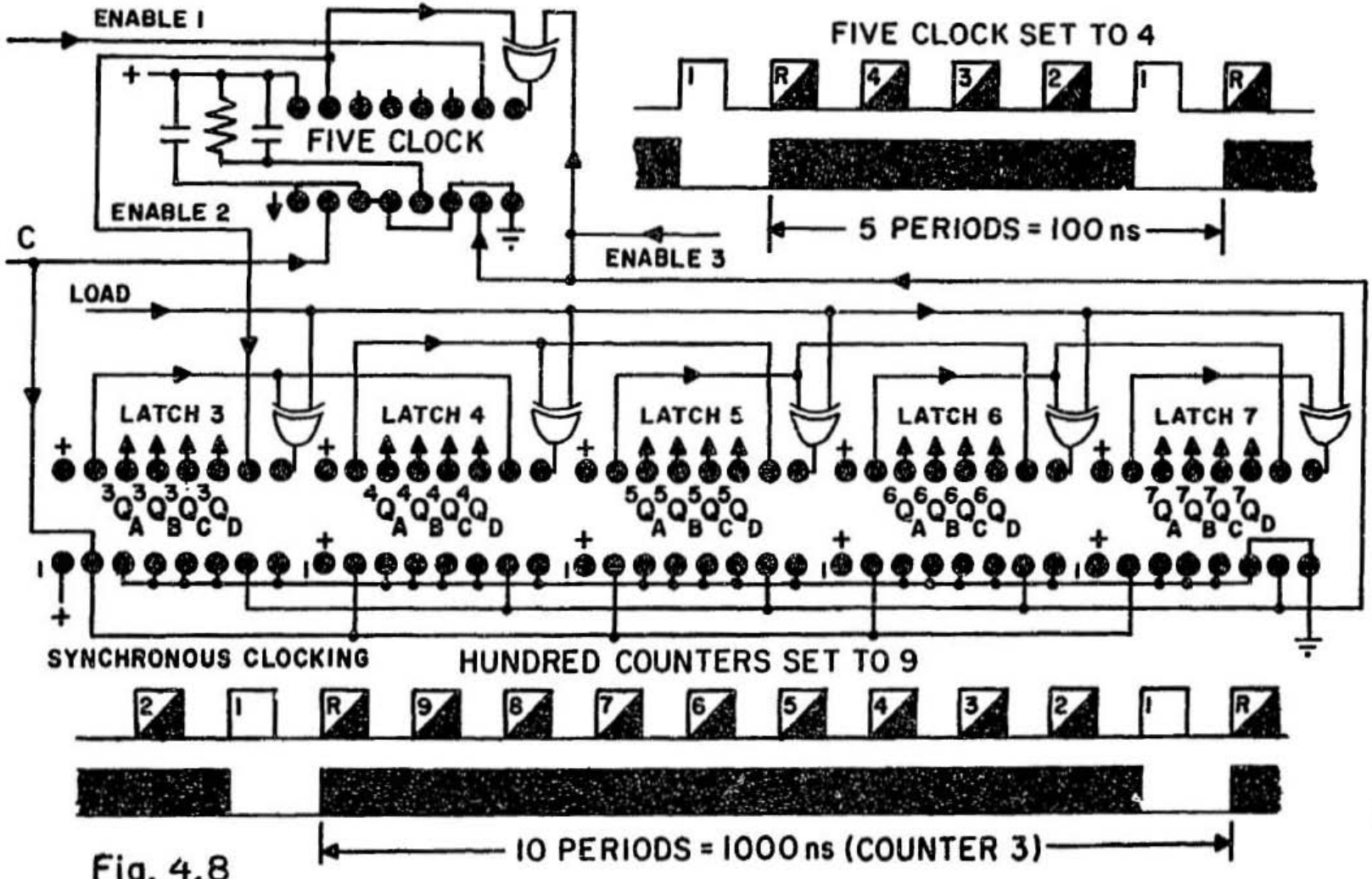


Fig. 4.8

SCHEMATIC AND PULSE-TIME DIAGRAMS, FIVE AND HUNDREDS COUNTERS

CHAPTER 5

MANUAL TIME INTERVAL INSERTION

Since a time interval can range from one nanosecond to 9,999,999 nanoseconds, seven digital manual switshes are employed to enter the periods of time preselected by the operator.

There are four sets of these switches, making it possible to enter up to four time intervals, each interval being of independent length.

When the numbers have been dialed into the system, they will appear at the A inputs of four registers which precede the four input registers of the arithmetic logic unit. They are automatically transferred to the output terminals of these registers for manual timing. At the same time, they appear at the A inputs of the arithmetic logic registers, but they are not stored in these registers unless the operator initiates the storage pulse by activating the automatic timing function.

These four sets of switches are located on the control panel in four horizontal rows, separated by horizontal rows of L.E.D.s, by means of which, the operator can quickly determine that the correct time data is entering each module.

The switches have 6 terminals; +5 volts input, ground, and four data voltage state output terminals.

CHAPTER 6

THE ARITHMETIC LOGIC DIVISION

When two or more external systems are to be operated in synchronization, the arithmetic logic division will consider the time delays required by each system to function, and make delta time delay adjustments in order that synchronization may occur. This is the function of the arithmetic logic division.

This division is a computer which can accept data from two to four inputs simultaneously, and each input can accomodate as many as seven digits. Each input is independent so that sources of inputs may be selected at random. Whichever source is to be used, it is selected by the appropriate panel button. The value for a specific delay time can be chosen from one of two sources:

1. the manual switches, or
2. the Q output storage areas of the Automatic time interval evaluation division.

Each input enters an input register and this register has two inputs entitled A and B respectively. Input A is for manual insertion of data and input B admits data from the time interval evaluation division. When a register control is low, the register accepts data from input A, while if this control is high, input B is accepted. After this control is set for each register being used, these registers will output the selected inputs on the falling edge of an activating

pulse. The outputs will now appear on the associated seven digit L.E.D. readout on the control panel, one readout per module, to verify to the operator that the system is receiving the desired inputs.

6.1 Phase one

Now the selected data appears at the outputs of the four input data registers which are numbered from one to four for easy reference, and we are ready to follow the sequence of operations carried out by the division.

Consider Fig. 6.1 which is a block diagram of the division. As shown, another pair of dual input registers, numbered five and six are required for the comparison of the outputs of registers one to four. The diagram shows registers one and three connected to inputs A and B of register five, while registers two and four are connected to inputs A and B of register six respectively.

When the controls of registers five and six are simultaneously set low, register five receives data from register one, and register six receives data from register two. Upon the trailing edge of the next clock pulse, these two registers output these data inputs, which now appear at inputs A and B of comparator one.

Upon the falling edge of the next clock pulse, the maximum of these two data inputs will be stored in register seven. If these two data are equal, then the data from register one will appear in register seven. While the

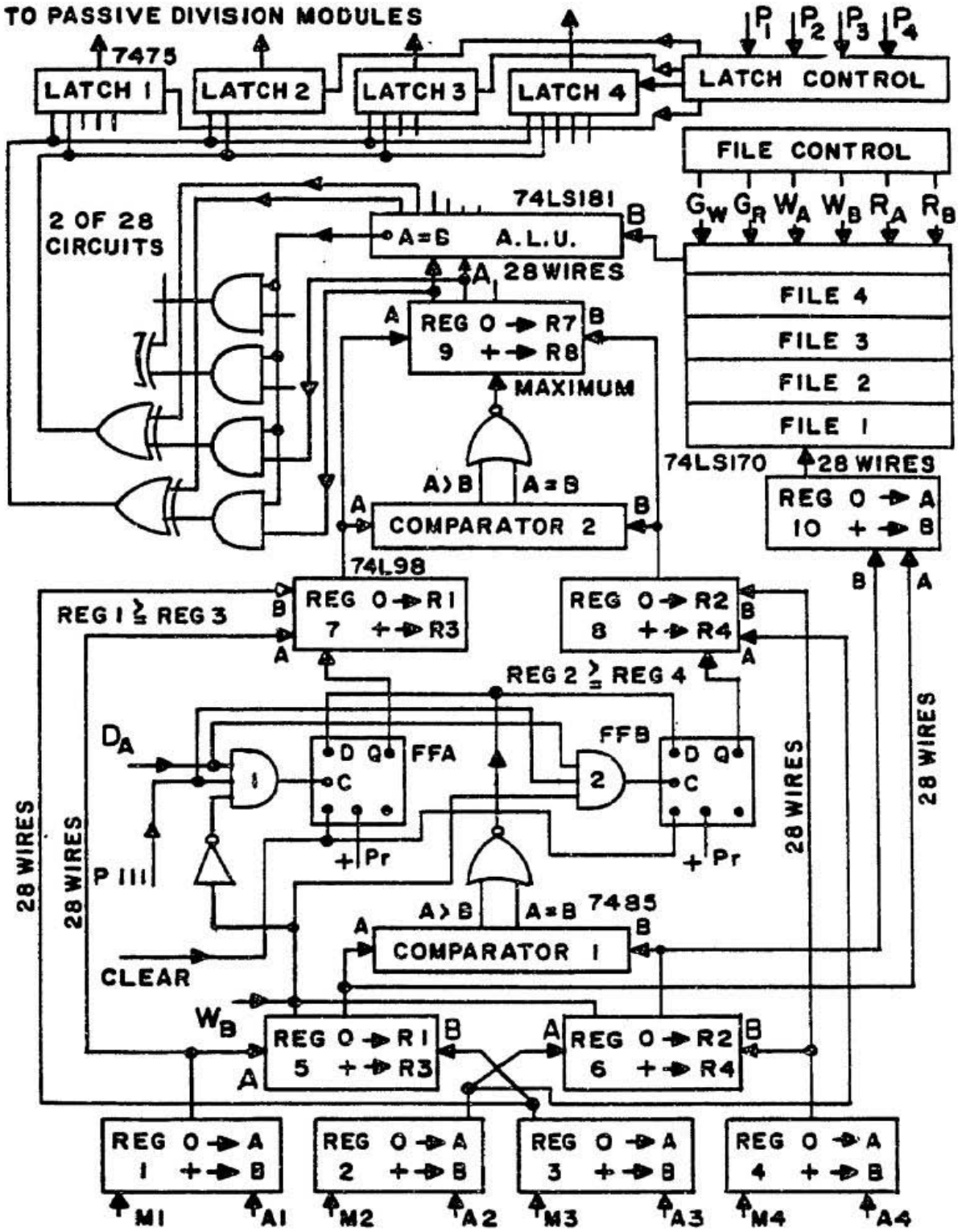


Fig. 6.1
BLOCK DIAGRAM OF ARITHMETIC LOGIC DIVISION

comparator is finding the maximum value, these two data are being stored simultaneously in files one and two, which act as memories.

The controls of registers five and six now automatically set high and comparator one compares the data inputs from registers three and four respectively. The maximum of these two data will now appear in register eight, and the input data from registers three and four will be stored in files three and four. Again if these input data are equal, data from register three appears in register eight.

Next comparator two compares the data outputs of registers seven and eight and selects the greatest one, which then appears in register nine and also at the input A of the arithmetic logic chip, A.L.U. Register nine acts as a memory. If these data are equal, data from register seven will appear in register nine.

But this maximum is not seen by the A.L.U. chip until the control of this chip is set in its proper state. The input control of the A.L.U. is M. When the state of M is low, this chip is operational. This control is set automatically by the division.

Part of the project involved fabrication of double sided circuit boards for the various sections of the apparatus. As an indication of the complexity of these circuits, the patterns for the A.L.U. boards are shown in

Appendix A

6.2 Phase two begins

File one data appears at the input B of the A.L.U. chip and the control of this chip is set low so that inputs A and B are compared, and if they are not equal, then they are subtracted and their difference appears in output latch one. If they are equal, then the data entered into output latch one will be zero. The output latches act as memories.

In this manner, all files are automatically read and compared to the input data from register nine, and the results stored in their respective output latches. Upon insertion of data into the final output latch, the division gives out a continuation pulse and shuts down.

This division will not be active again unless new data is inserted, because the output data is stored in the output latches where it is available for quick transfer to the Passive division without losing the data.

Now consider Fig. 6.2, a pulse time chart. By means of this chart, we can follow the division's method of operation step by step once the required data appears in the input registers one to four inclusive.

Note that the chart is divided into two phases, and that time increases as we go from left to right across the chart.

Line 1. The set or primary activation pulse, which occurs only once to initiate the system. This pulse is shown to appear coincident in time to some pulse in the wave train of pulses coming from the set-up clock. The method of accomplishing this will be given under 'Activation procedures'.

Line 2. The wave train of square pulses now enters the system from the set-up clock in the control section and becomes the vehicle to control the order of operations. Note that the set-up clock runs continuously, but it is only available to this division when the division is activated.

Line 3. The system's automatic low state setting pulse. This pulse is necessary to insure that all parts of the division start in the proper states relative to each other.

Line 4. Here the automatic run control state becomes high. The division will run until this control state becomes low.

Line 5. Here we have the set-up clock pulses which are seen by the division. Important pulses in the wave train either bear Roman numerals or are darkened to relate their connection with events depicted lower down in the chart. They appear only in phase one - the input phase.

Line 6. Here the wave train is inverted. The important pulses that appear in phase two - the output phase - are labelled 'G'.

Line 7. In the input phase, here, we see the times at which we have low states. This line depicts the changes of state of the function G_W with time, which is the write control. Whenever G_W is low, data can be written into the files one to four, if the conditions in Table 6.1 are met. Note that no clock pulse is required to transfer data into a file. But G_W must first be low. Also note that the pulse shape is changed for the output phase (phase two part of chart). A form of G_W that can be used in the system is illustrated in line 8.

Line 8. It shows the corrected wave for G_W. The low state at the beginning of line seven, before the division is activated, had to be removed and the varying states in the output phase had to be replaced by a steady high state. The file must not try to write data into a storage cell and read it at the same time.

Line 9. This is the preliminary set-up pulse train for the read control G_R which is unsuitable because it has a low state during phase one, where the state should be high, since no reading is to occur in this phase. Reading only takes place when G_R is low. Because this form of G_R is unuseable, we must modify it in the manner shown in line 10.

Line 10. This shows the corrected wave form for G_R. The part of the wave form in phase one also represents control M of the A.L.U. chip because it is in the high state: - the required state for control M to keep the A.L.U. chip

inoperative.

Line 11. In phase two we have four pulses which are required to transfer data to the latch outputs. Note that each pulse occurs when G_R is low - line 10 - and W_A and W_B have proper coded states for each file.

Line 12. This represents the state of control M which is low in phase one and high in phase two. Solid pulses are shown in order to compare related pulses in time on different lines of the chart. Since M is high, the state of M in phase two plays a part in shaping other pulse forms. Note that the related pulses in a vertical column are required jointly to obtain the necessary function.

Lines 13 and 14. These are the normal and inverted outputs of the first divide by two circuits.

Line 15. The inverse of G_w line 8 for phase one.

Line 16. The inverse of G_R line 10 for phase two.

Lines 17 and 18. These are the normal and inverted outputs of the second divide by two circuits.

Line 19. The third divide by two circuits - gives W_A in phase one, and R_A in phase two.

Line 20. The inverse of line 19 which is used for pulse shaping and timing.

Line 21. The fourth divide by two circuit which gives W_B in phase one, and R_B in phase two.

Line 22. This is the line for $\overline{M}$, the inversion of control M of the A.L.U. chip. The leading edge of this line at the end of phase two triggers the continuation pulse.

The solid pulses are identified across the bottom of the chart. These are useful in checking the step by step operations of the division.

Upon giving out the continuation pulse, this division becomes inoperative.

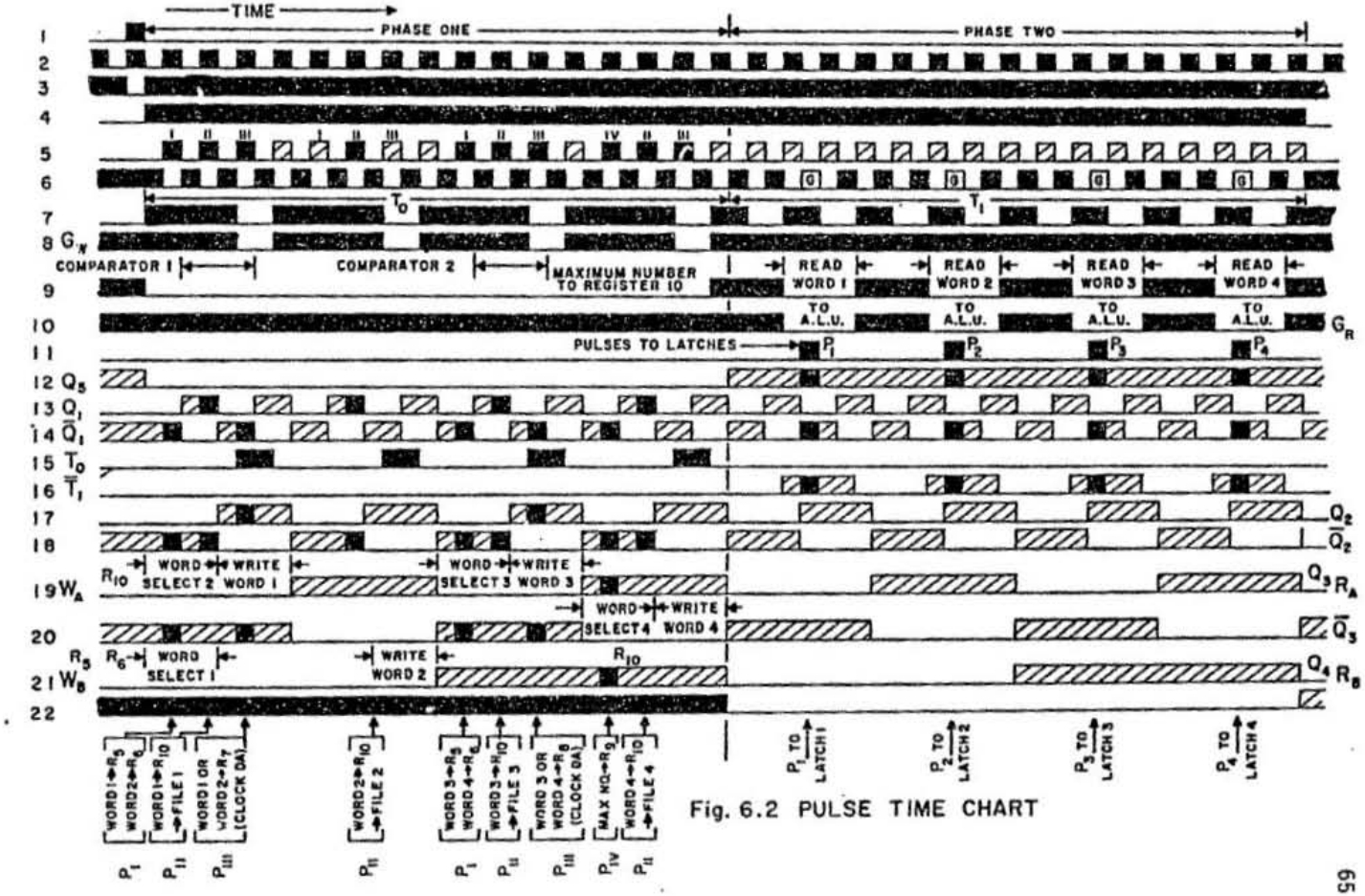


Fig. 6.2 PULSE TIME CHART

TABLE 6.1 A.L.U. FILE CONTROL

CONTROLS		WRITE - READ FILE STATES					
		FILE NUMBER				CONTROLS	
G_W	G_R	1	2	3	4		
		LOW	HIGH	LOW	HIGH	W_A	R_A
		LOW	LOW	HIGH	HIGH	W_B	R_B
LOW	HIGH	WRITE ONLY					
HIGH	LOW	READ ONLY					

CHAPTER 7

THE PASSIVE PHASE - TIME DATA TO MACHINE LANGUAGE

The time interval data outputs from either the Manual insertion division or the Arithmetic logic division are now available at the inputs of the various counters of the Passive division of each module, but loading pulses A_o and B_o are required to load this data into the Q data storage areas of these counters. Up to four modules may be operated in parallel concurrently. But since all the modules are duplicates, the operation of only one module will be described.

By carefully analyzing the operation of a digital counter such as the 74S168, it became obvious that the raw data for the time interval could not be loaded directly into the final Active division, due to peculiarities in the functioning of counters. Consider the following:

Recall that a time interval was to be created by the following sequential operation. The operator would load required data into a series of three counters by means of loading pulses A_o and B_o. Then he would initiate the steps to create the interval by turning on the first set of counters, which will count down until the Q data storage becomes zero, and these counters then turn on the next set of counters. Thus the counting of the interval continues

until the Q data storage of the last set of counters becomes zero, at which time the termination pulse is given out. The time interval is thus the time period between the initiation and termination pulses.

Two problems arise immediately; one with the manner of operation of counters, and two, the difficulty of initiating the counting of pulses using an endless train of pulses from a continually running oscillator.

As regards the first problem of the manner of operation of the counters being keyed on the Q data storage becoming zero, consider Fig. 7.1. Here the upper line shows the clock pulses being counted down, each pulse numbered from 10 to 1, just as we normally count. Note that there is no zero count. The next line indicates the changes in the Q data storage as the counting proceeds in the count down mode. The next line indicates the pulses still to come before reset of the counter. The bottom line shows the voltage variations of the ripple carry output terminal 15 with the counting. Note that the number of the Q data storage is set before the pulse of that number arrives. Also it is now clear that the pulse we would count as ten is the reset or zero pulse.

The ripply carry low state output pulse shown on line 4 is to be employed to activate numerous circuits, counters, and flip-flops, etc. Note that this pulse has two well defined edges, one falling and one rising. Either edge

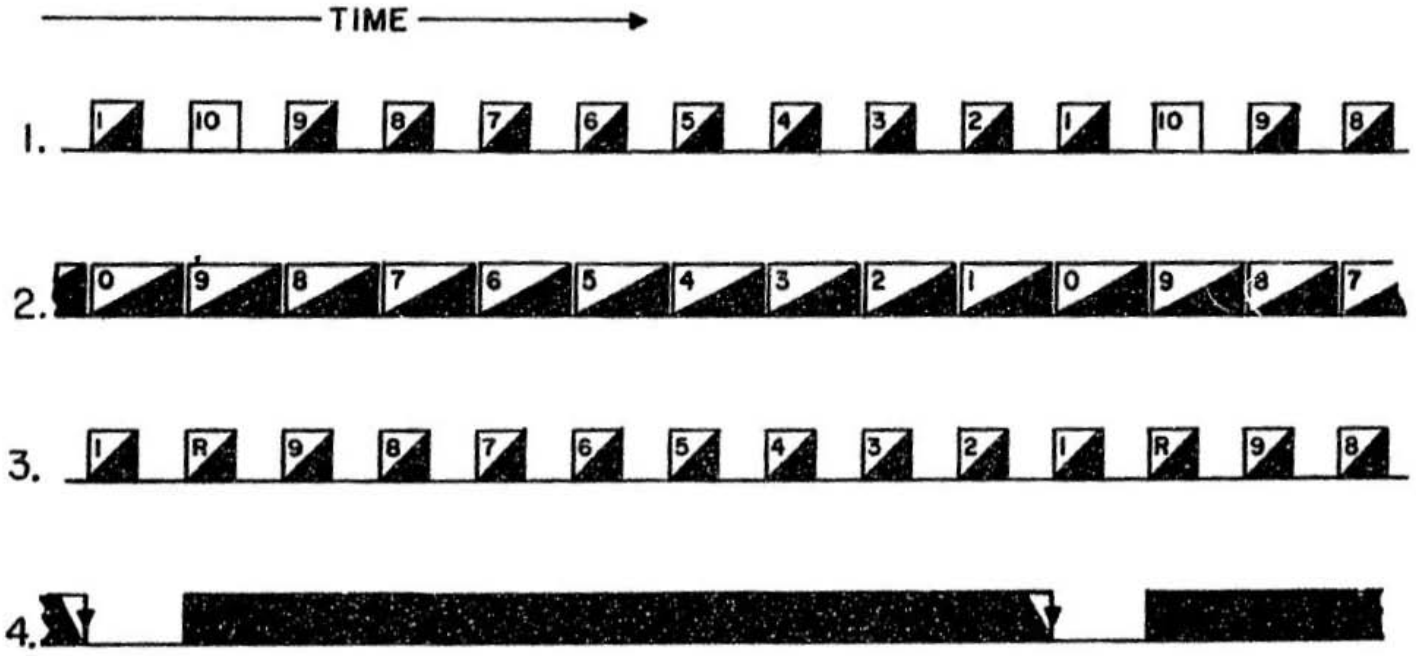


Fig. 7.1 RELATIONSHIP BETWEEN COUNTING, Q DATA STORAGE, AND RIPPLE CARRY ENABLING PULSES

may be used as a control edge; the edge chosen for a given situation depending jointly upon the time requirements and the characteristics of the necessary circuitry. Thus either edge can be a control which is used to turn on or off other circuits.

Problem one then involves choosing which edge to use for a specific counting method, and then adjusting the input binary data so that the Q data storage states representing state zero arrive at the proper time.

Problem two is associated with the time that is required to turn on or enable any electronic device. Thus if we can use the falling edge of the ripple carry pulse (low state, line 4) to enable the next piece of circuitry, we must make certain the next circuit is fully operational before the next rising or trailing edge of the clock pulse comes - the edge of importance depending upon which edge the next circuit is designed to see. It is not enough that the circuitry assumes the proper control voltage level before this clock edge comes. This control voltage level must be present and steady for some time before the clock edge comes or the next circuit may not respond to the clock edge (we say that the circuit did not see the required edge).

Assuming that the control voltage level can be ready in time, what other problems are involved? Lines 3 and 4 of Fig. 7.2 show the rising edges for two clocks. Those in line 4 occur later in time. Consider these clock pulses appearing at the inputs of two counters A and B respectively,

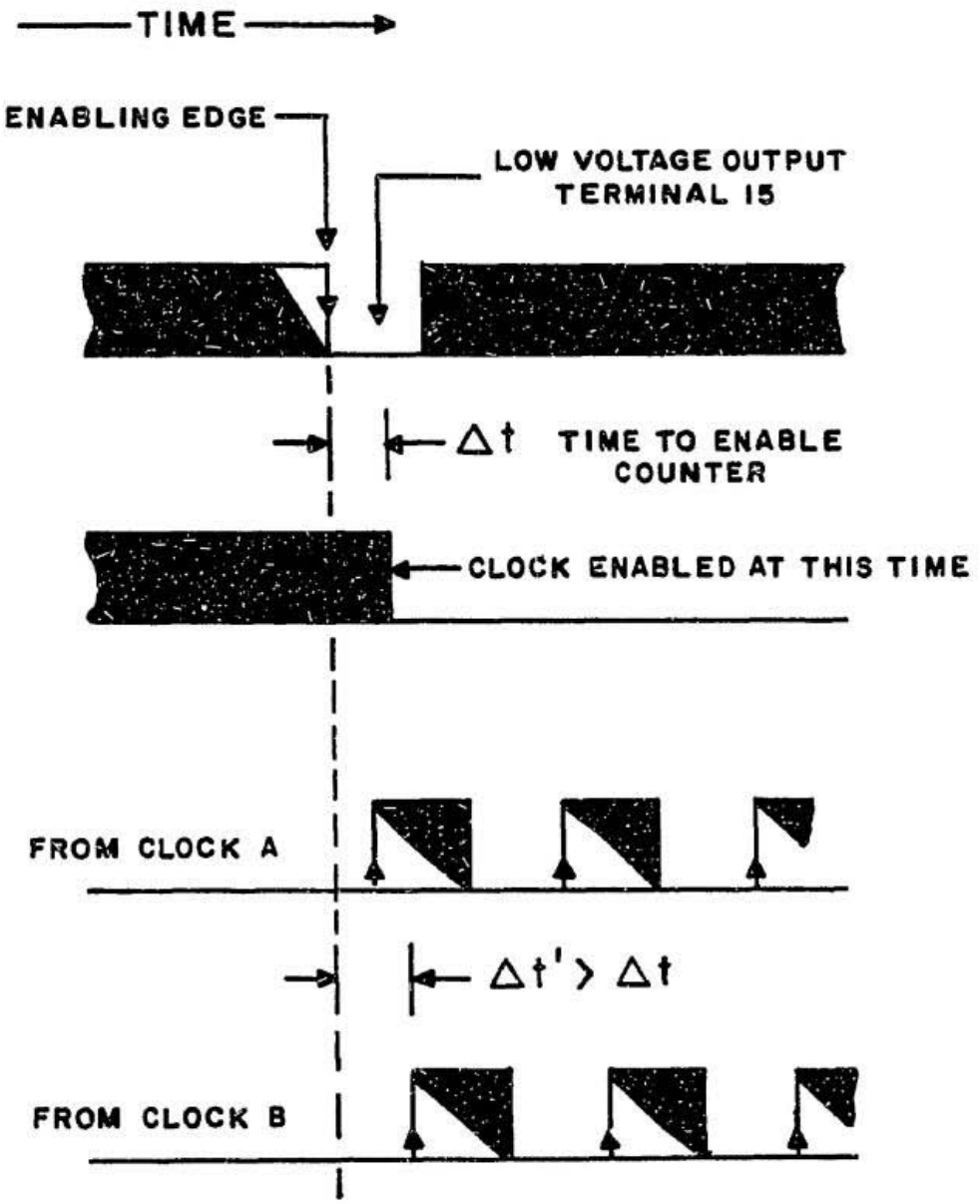


Fig. 7.2

CRITICAL TIMING OF CLOCKING EDGES

both counters being activated by the ripple carry pulse of line 1. Because of the time delay Δt (line 2) inherent in these counters, it is not possible for counter A to see the first rising edge of the clock in line 3. All edges must appear at some specific time interval after the falling edge in line 4, this time interval being greater than Δt. The duration time of the zero voltage level of the oscillator $\Delta t'$ must be greater than the interval turn on time Δt.

When we transfer the counting from one counter to a second counter, both of which count pulses from the same oscillator, then the half period of the oscillator's frequency must be greater then the period of time represented by the sum of the delay of the ripple carry pulse and the enabling delay of the second counter.

But when the second counter sees a different clock, a new problem arises as to where we enter the second wave train. Thus to transfer the counting of time from one set of counters to another set with a different clock, we must be able to:

1. control the start up time of the second clock which will supply pulses to our second counter, such that any delays in the trigger pulse are compensated for by adjusting the time interval before the first output pulse arrives,

2. control the turn on time of the second counter so that the delay in the trigger pulse and the internal enabling delay of this counter together are less than the second oscillator's turn on time.

Possible timing errors due to delays must be considered at each step. Examples of the solution to this problem appear under the sections 'The hundred clock' and 'The phasor - vernier relationship'.

The Passive division then must prepare the raw input binary data so that it appears in the form demanded by the Active division for a required time interval. To carry out this need, the Passive division has been designed to be an automatic time data to machine language conversion computer which automatically accounts for all delays in different components.

It must look at the data sectionally,- that is it has its input data divided into three classes:

1. hundreds
2. tens, and
3. units of nanoseconds.

Thus there are three systems of counters.

Here it is necessary to point out that in this division, these three systems of counters are similar to the three systems appearing in the Active division, but they only cycle once when updating input data. On the other hand, in the Active division, they cycle continuously. By this parallel arrangement, the Q data storage of these sets of counters first receive the raw binary time interval data either from:

1. the Automatic time interval evaluation outputs,

2. the Manual interval insertion outputs, or

3. the Arithmetic logic outputs.

Then the Passive function causes data to be added to or subtracted from the Q data to give the desired machine language for the Active division. Thus this machine language is stored in the Q data areas of the Passive division.

By controlling the counting direction terminal of each counter, we can cause each counter to:

1. count up and add data to existing data, or

2. count down and subtract data from existing data already stored in the Q data storage areas.

The amount of data added to or subtracted from the Q data areas of the various sets of counters is fully explained under the sections entitled 'Hundred', 'Shifter', 'Phasor', 'Vernier', and 'Zero' clocks.

The Passive division must also test the Q data storage of the set of five counters entitled 'Hundred' clock as to its data size, so that the Active division may perform one of three possible methods of counting down a required time interval, to be explained in the section 'Q clock'.

Thus the Passive division carries out a triple role:

1. it checks data to determine the mode of operation,

2. rewrites input time data into machine language, and

3. stores the information obtained by operations 1 and

2 in the Q clock and the Q data storage areas which act as temporary memories. The operations by which this triple role is fulfilled is called the 'set-up'.

Thus once the above information has been stored, it can be quickly and repeatedly transferred to the Active division, and not lost each time the Active division clocks out a given time interval.

To understand the various control systems making up this division, consider Fig. 7.3 which is a schematic diagram of a complete module. There are four such modules in parallel and the 'set-up' function controls these modules concurrently.

The Shifter counter and one counter of the five counters which constitute the 'Hundred' clock are shown near the bottom right hand corner of the diagram. The Phasor and Vernier counters are directly above these counters, while still higher up the diagram near its upper right hand corner is the Zero clock.

Note that all data to the 'Hundred' clock enter directly via terminals 3 to 6, being labelled a_n, b_n, c_n, and d_n where n has a value from 0 to 4 depending upon which counter you are considering.

But note that for the Shifter and Vernier clocks, where terminals 3 to 6 are lettered a''_o, b''_o, c''_o, d''_o, and a'_o, b'_o, c'_o, d'_o respectively, that a''_o and a'_o are not entered into the counters. Instead b''_o, c''_o, and d''_o as well as b'_o, c'_o, and d'_o are advanced in position of entry into the counters,

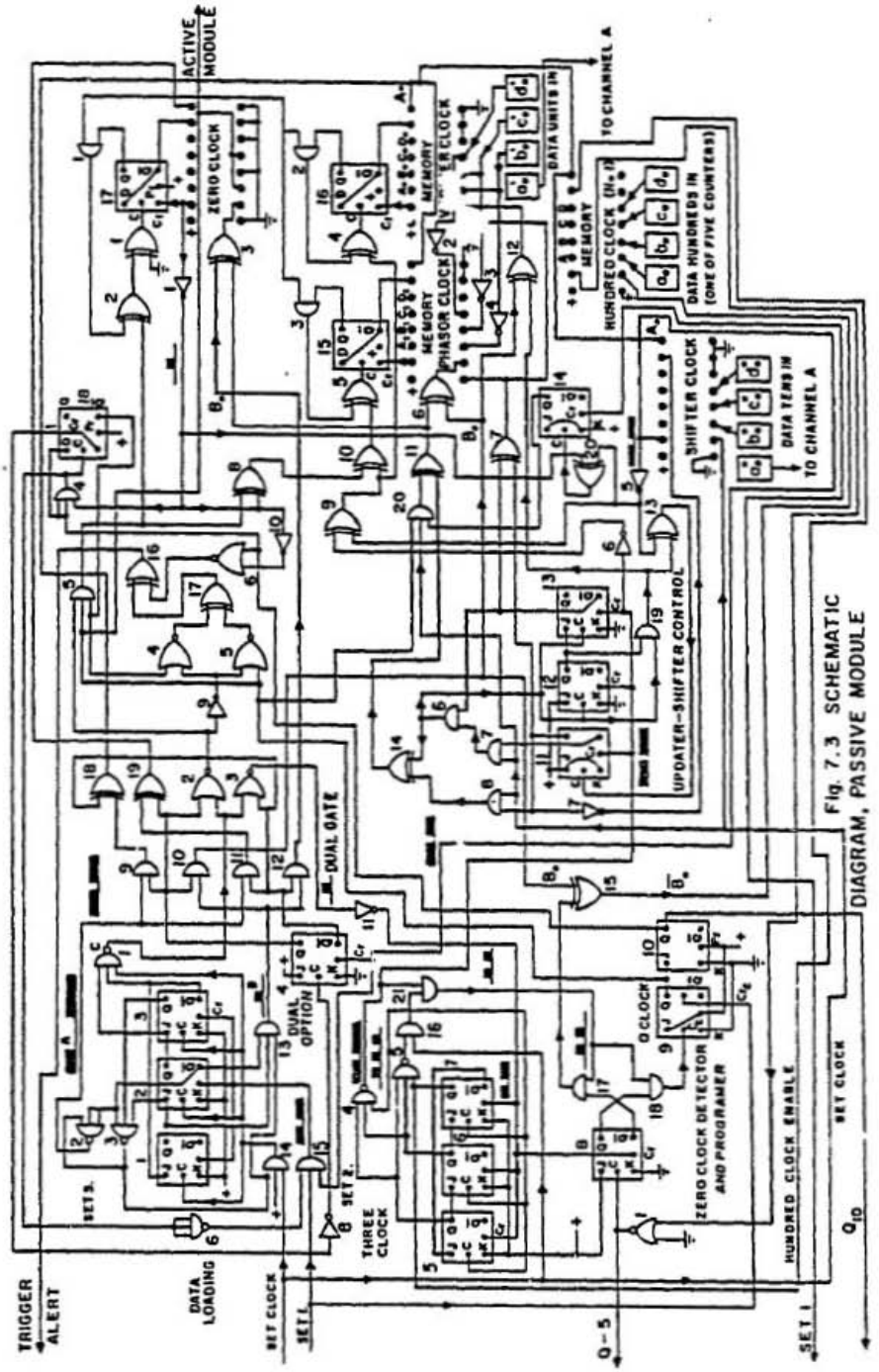


Fig. 7.3 SCHEMATIC DIAGRAM, PASSIVE MODULE

so that each input enters the terminal immediately ahead of the one it would normally enter. That is for b_o', b_o' enters terminal 3 instead of terminal 4. This leaves terminals 6 of these two counters with no data inputs, and so these two terminals are grounded.

Next notice that the a, b, and c data inputs to the Phasor are the data inputs to terminals 3, 4, and 5 of the Vernier inverted. Again terminals 6 are grounded. This method of inputting data to the Phasor will be justified when the data tables for the Phasor are considered later in this chapter.

Here it is only necessary to say that inputs a_o'' and a_o' are employed to tell the Active division which clock it is to see: the true clock, the inverted clock, or a delayed clock. This will also be clarified by the study of the data conversion tables.

Notice that directly above the Phasor, Vernier, and Zero clocks appear D type flip-flops. These flip-flops are controls which decide whether the counters count or remain idle.

Immediately to the left and slightly above the Shifter, there is a system of three J-K flip-flops and their controlling gates. This system connects to the Shifter via Inverter 5, and to the Phasor and Vernier clocks via:

1. the EXCLUSIVE-OR gate 7 which controls the count direction terminals 1, and

2. EXCLUSIVE-OR gates 6 and 12 which permit clock

pulses to be added to or subtracted from the Q data storage areas of the Phasor and Vernier, depending upon the voltage level applied to terminals 1 of these two clocks.

In the upper left hand corner of the diagram is the system of three flip-flops and associated gates called the 'data loading circuit', which when triggered, produces three pulses in sequence - a long low state A_o pulse, a positive short clocking pulse B_o that occurs during the lifetime of the pulse A_o, and a short low state pulse C_o, a continuation pulse after the lifetime of A_o as shown in Fig. 7.4.

The first two of these pulses are used to load the binary data at the inputs of the Phasor, Vernier, Shifter, and Hundred clocks into the Q data storage of these counters. The third pulse simply activates the next control circuit. Note that no data is loaded into the Q data storage areas of the Zero clock here.

Flip-flop 4 forms a dual gate. It is initially set in mode one, whence the continuation pulse C_o proceeds to the system of three flip-flops immediately below the loading circuit, called the 'Three clock'. This circuit is so named because it can put out only three pulses which are employed:

1. to test the number capacity of the 'Hundred' clock, and 2. to control a dual flip-flop circuit entitled 'Q clock', and

3. to provide a continuation pulse.

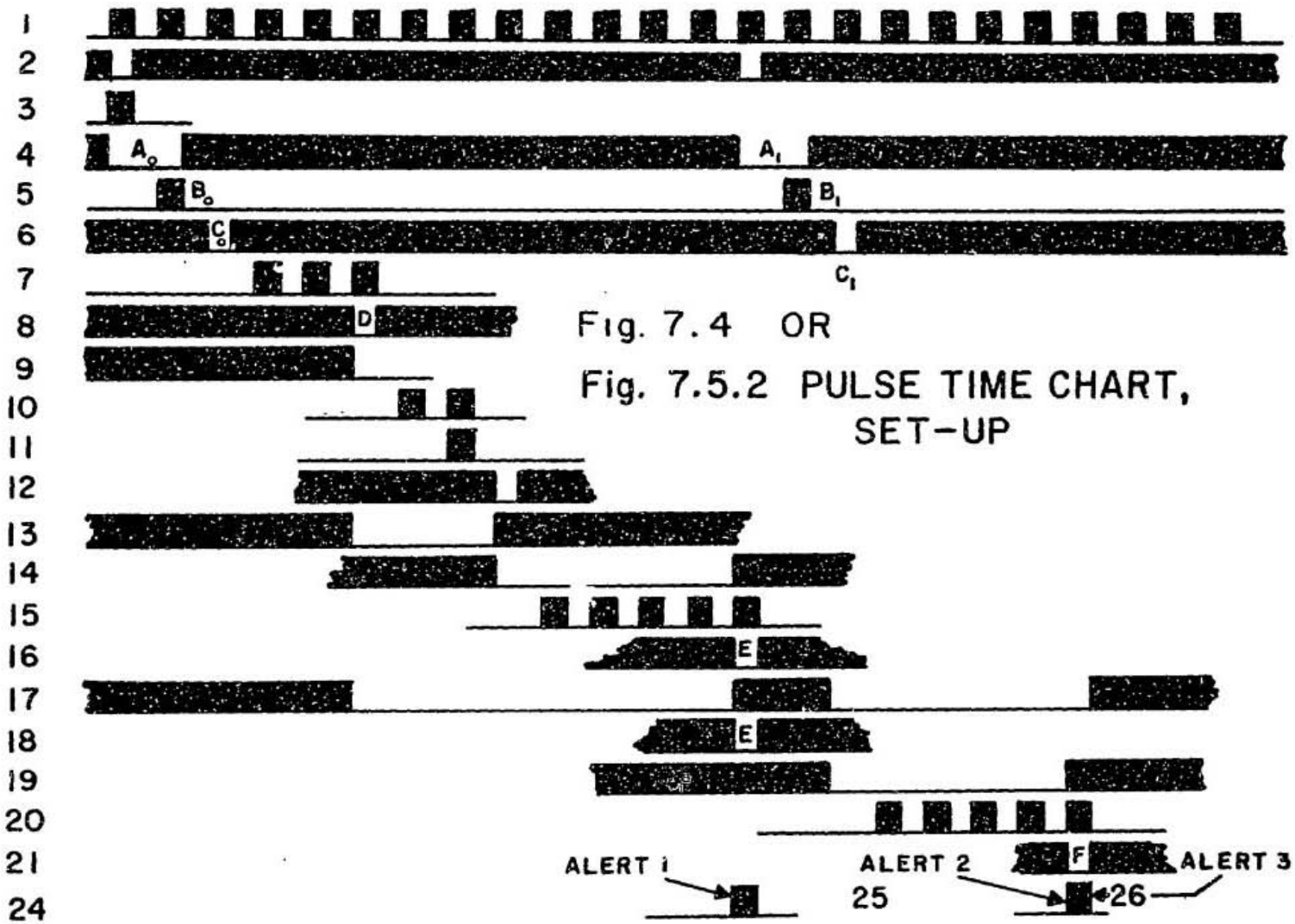


Fig. 7.4 OR

Fig. 7.5.2 PULSE TIME CHART, SET-UP

79-119

If the data in the 'Hundred' clock Q data represents a digit less than two, then the 'Q clock' becomes active.

This 'Q clock' controls all the circuitry in the upper half of the diagram lying between the loading circuit and the Zero clock. This circuitry is required to decide:

1. if the 'Q clock' will be used in the Active division, and 2. how it will be used - that is how the Q data storage of the shadow clock will be updated before being passed to the Q data storage of the Active division's Zero clock. This operation is discussed fully under the section 'Set-up'.

Now the third pulse, which is the continuation pulse, also provides an inverted pulse used to turn on the Shifter clock and the Shifter - Phasor - Vernier control circuit. This permits:

1. the Vernier and Phasor to decrease their data by one, and then

2. the data in the Q data storage areas of the Shifter to be added to the data in the Q data storage areas of the Phasor - thus shifting the data values in the Phasor Q data storage.

When the Shifter data value becomes zero, the Shifter puts out a low state pulse which is employed to turn off its control circuit.

By these means, the data input into each set of

clocks is updated and appears in the output of these shadow clocks ready for transfer to their active components.

In order to control the four parallel modules, as well as control and record the states in each of the modules, a special circuit entitled 'Control' has been created which monitors the actions of all circuits making up the entire time interval evaluation system. This control ensures:

1. that all steps are carried out properly in time,

and 2. that once the system is initiated, the output pulses are automatically obtained.

The control also makes the automatic adjustment so that the proper control is maintained even when as few as one module is employed.

The division will now be described completely under the headings:

1. The Hundred clock,
2. The Three clock,
3. The Q clock,
4. The vernier-phasor relationship,

and 5. The set-up function,

7.1 The Hundred clock

Consider Fig. 7.1.1. This is the schematic diagram for this circuit. This clock consists of five decade counters clocked synchronously in the down count mode. The clock in this module is similar to the one in the Active division.

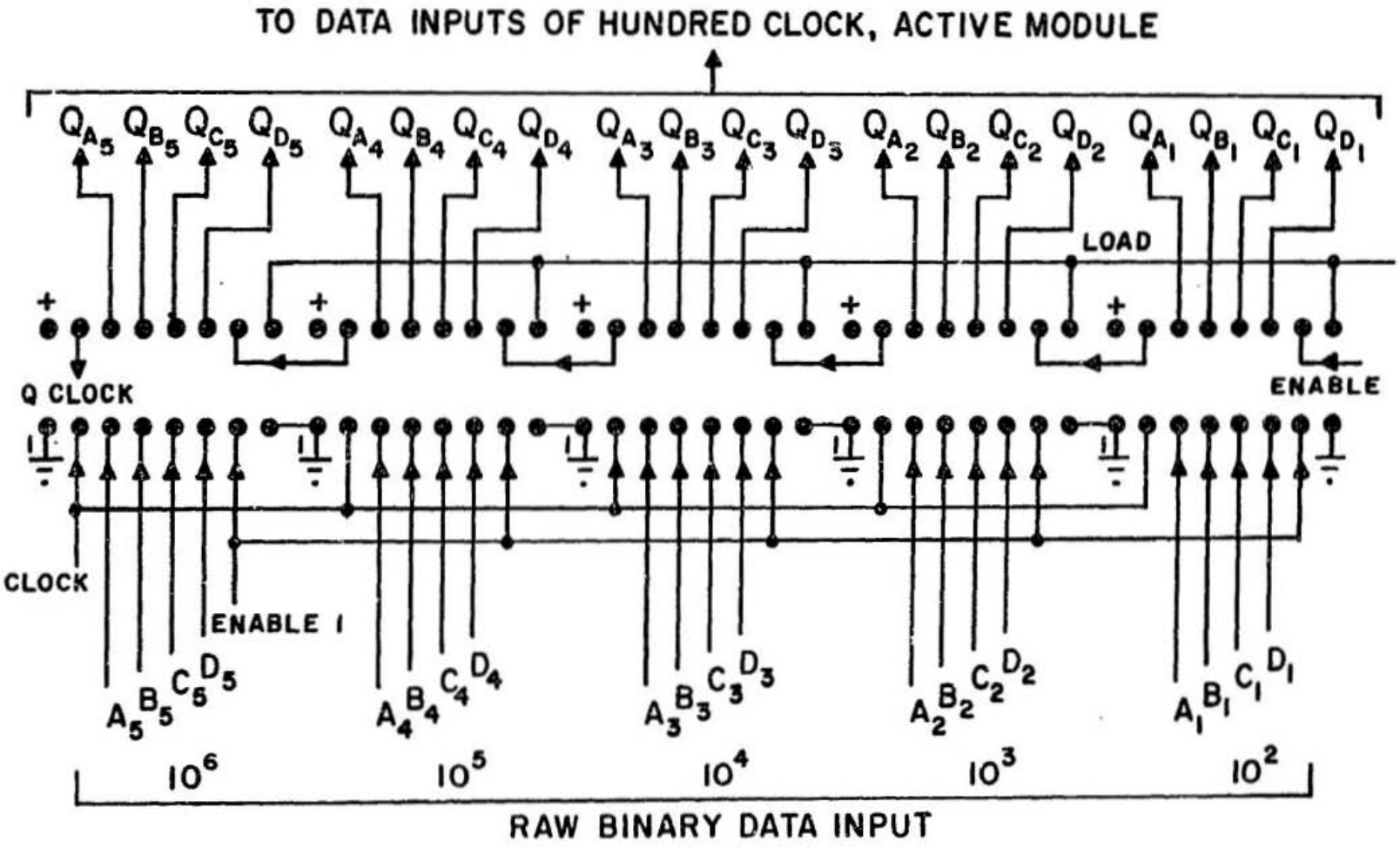


Fig. 7.1.1 HUNDRED CLOCK, PASSIVE MODULE

The binary input data enters the counters at inputs A_n, B_n, C_n, and D_n, and it is finally available in the Q data storage areas Q_{A_n}, Q_{B_n}, Q_{C_n}, and Q_{D_n} upon receipt of loading pulses A_o and B_o. The counter number is n.

In the passive module, this clock can receive only two pulses, and they come from the 'Three' clock. These two pulses are employed to determine whether the clock in the active module will count or remain idle. The criteria for the 'Hundred' clock to be used in the Active division is that the data input in the hundreds division must be greater than 2.

If upon receipt of two pulses, terminal 15 of the fifth counter "does not" put out a low voltage state, then:

1. the Q data storage is greater than the data for the digit 2, and

2. the Q data storage of this clock appears at the inputs of the active clock.

At the end of the set-up operation, this data will be loaded into the Q data storage areas of the active clock and that clock will be partially enabled by 'Control' at this time.

Note that this clock is enabled by the 'Three' clock when the Q output of flip-flop 7 assumes a low voltage state, and this state of the flip-flop will persist until the three pulses have been delivered by the 'Three' clock.

Recall that only two of these pulses may reach the inputs of the passive 'Hundred' clock.

The two cases possible, when terminal 15 of the fifth counter "does" put out a low voltage state, are dealt with under the section entitled, The Q clock.

Note that this clock counts by single periods in the passive module, since we are not timing in the Passive division, while in the Active and Time interval evaluation divisions, this clock counts by groups of five periods which constitute hundreds of nanoseconds. Consequently, the Passive division can be updated at high speeds which is important when used in conjunction with the Interval evaluation division. In both the Passive and Active divisions, this clock counts down, but in the Time interval evaluation division, it counts up.

7.2 The Three clock

This clock consists of flip-flops 5, 6, and 7 together with NAND gates 4 and 5 and AND gates 16 and 21. As stated earlier, this circuit must enable the 'Hundred' clock in the passive module, and while enabling it, deliver two pulses to it. The 'Three' clock will disable the 'Hundred' clock.

Note that when this clock is enabled by the narrow low voltage pulse from Inverter 11 during the first phase of the set-up procedure, the 'Q clock' gate (flip-flop 8) is also preset ready for the testing of the

'Hundred' clock.

7.3 The Q clock

This section explains how the 'Q clock' controls the 'Zero' clock. Consult Fig. 7.3.1 which is an enlargement of the 'Q clock' control section of Fig. 7.3 while reading this section. Some recapping will occur here in order to ensure that the complexity of the system is properly described and explained.

When set-up is activated, a low pulse C_o comes from NAND gate 1 and arrives simultaneously at terminal 1 of NOR gate 2 and terminal 2 of NOR gate 3. Since the voltage level at terminal 2 of NOR gate 2 is kept high by the output voltage of flip-flop 4 whose terminal Q is positive, this pulse C_o operates only on NOR gate 3 to send a positive pulse to Inverter 11. This inverter puts out a low voltage level pulse which clears flip-flops 5, 6, and 7 and also activates the 'Three' clock.

Then the 'Three' clock puts out 3 pulses of which the first two go to the 'Hundred' clock. If the Q data storage of this clock is 0, 1, or 2, then 2, 1, or 0 pulses respectively will be inputted to the 'Q clock' to establish the values of Q_1 and Q_2. The three possible conditions for the 'Q clock' are:

1. two input pulses, $Q_1 = +, Q_2 = +$
2. one input pulse, $Q_1 = +, Q_2 = 0$
3. no input pulse, $Q_1 = 0, Q_2 = 0$

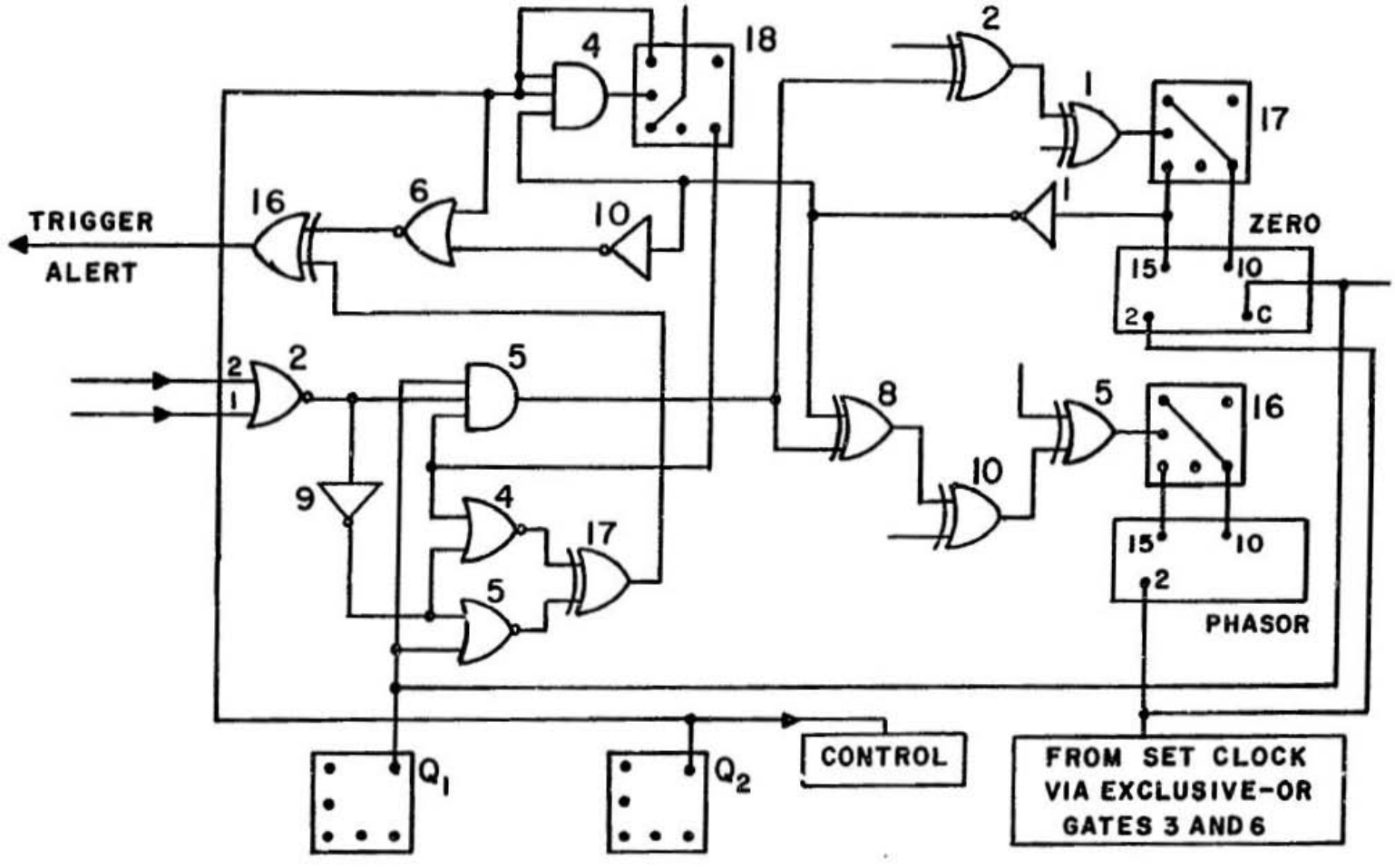


Fig. 7.3.1 Q CLOCK CONTROL FOR ZERO CLOCK

Recall that 200 nanoseconds were reserved for the operation of the vernier. Now since the data value in the 'Hundred' clock's Q data storage area can be 0, 1, or greater than or equal to 2, there are three possible conditions for which the Passive division must search. The 'Q clock' records which of the three conditions exists and causes the Passive division to update its output data storage so that the Active division will function properly.

Earlier it was explained that the vernier required only a maximum of 5 periods (100 ns) to determine the units of nanoseconds. This property of a maximum of 100 nanoseconds permits the operation time of the 'Phasor' to be reduced from 200 to 100 nanoseconds for the case when the Q data storage of the 'Hundred' clock is 1. For the case when this storage is zero, the operation time of the 'Phasor' is reduced by 100 nanoseconds and the conditions set up so that during the active counting period, the 'Zero' clock will count down 100 nanoseconds concurrently with the 'Phasor' and the output of the 'Zero' clock will establish zero time. Thus the time interval between this zero time and the time when the vernier output pulse occurs is the required time interval.

Pulse 3, inverted, now comes from NAND gate 4 and it clears flip-flops 11, 12, and 13, and this pulse also goes to Inverter 6. Inverter 6 puts out a positive pulse which turns on the 'Phasor' and 'Vernier' clocks via

flip-flops 15 and 16 and which also goes to EXCLUSIVE-OR gates 10 and 4, while at the same time causing the count down direction for these two clocks to change to the count up mode.

Now by means of the circuitry associated with flip-flops 11, 12, and 13, two and one pulses are added to the data stored in the Q data storage of the 'Phasor' and 'Vernier' clocks respectively. Then the 'Shifter' clock is turned on and its Q data is added to the Q data storage of the 'Phasor' clock. The low output level voltage from terminal 15 of the 'Shifter' when the data in the Q storage of the 'Shifter' becomes zero, acts via Inverter 5 to turn off the 'Phasor' and 'Vernier' clocks, and also to set flip-flop 4 so that its Q output assumes a low voltage level. Note that although the 'Vernier' clock was not shut off by its control flip-flop 16 while data was transferred from the 'Shifter' to the 'Phasor', it was inactive during this transfer being isolated from that data since it could only receive one pulse via the circuitry of flip-flops 11, 12, and 13.

When the 'Phasor' and 'Vernier' clocks are shut off by their control flip-flops, namely 15 and 16, their count mode changes back to the count down mode.

The low voltage level pulse that resets flip-flop 4 and which came from terminal 15 of tne 'Shifter' clock, also reactivates the Data loading circuit by clearing flip-flops 1, 2, and 3. This causes A_o, B_o, and C_o

to be generated again being redesignated as A_1, B_1, and C_1. Now A_1 and B_1 perform the function of loading the 'Zero' clock.

Now we come to the 'Zero' clock which can be active in two of three possible cases which are set up by the 'Q clock'.

Case 1. Q clock: $Q_1 = 0$, $Q_2 = 0$.

Since terminal 1 of NOR gate 3 is now held at a positive voltage level and terminal 2 of NOR gate 2 is at a low voltage level, pulse C_1 passes via NOR gate 2 and Inverter 9 to NOR gates 4 and 5. This pulse C_1 is now a low voltage pulse due to Inverter 9, and since $Q_1 = 0$, terminal 1 of NOR gate 5 is at a low voltage level, we have a high voltage pulse which passes through EXCLUSIVE-OR gates 17 and 16 to give a trigger pulse to 'Control'.

Case 2. Q clock: $Q_1 = +$, $Q_2 = 0$.

Now terminal 1 of NOR gate 5 and terminal 2 of AND gate 5 are both at a positive voltage level while terminal 2 of NOR gate 4 is still at a positive voltage level. Thus C_1 gets through Inverter 9 but is blocked by NOR gates 4 and 5. But because C_1 is here a positive pulse, it can pass via AND gate 5 to EXCLUSIVE-OR gates 2 and 5 to turn on the 'Zero' and 'Phasor' clocks respectively. Thus these two clocks will count concurrently until the Q data storage of the 'Zero' clock is zero. Note that the 'Phasor' is now in the count down mode, but the 'Zero' clock is

always in the count down mode. Also note that terminal 5 of the 'Zero' clock is joined to Q_1 of the 'Q clock' which is at a positive voltage level. This causes the data input of the 'Zero' clock to be five.

Terminal 15 of the 'Zero' clock puts out a low voltage level pulse to flip-flop 17 and Inverter 1 when the subtraction is completed, and thus:

1. turns off this clock via this flip-flop and
2. turns off the 'Phasor' via EXCLUSIVE-OR gate 5

while Inverter 10 sends a low voltage level pulse to NOR gate 6. From this gate, a high voltage level pulse goes via EXCLUSIVE-OR gate 16 to give the trigger alert pulse to 'Control'.

Case 3. Q clock: $Q_1 = +$, $Q_2 = +$.

All actions are the same as in case 2 until the low voltage level pulse arrives at terminal 2 of NOR gate 6. Terminal 1 of this gate is joined to output terminal Q of flip-flop 10, and Q is at a high voltage level; thus NOR gate 6 is inactive. But now inputs 2 and 3 of AND gate 4 are both at a high voltage level, so that the pulse from Inverter 1 passes through AND gate 4, sets flip-flop 18 so that its Q output is at a low voltage level, and retriggers flip-flops 1, 2, and 3 to give reloading pulses A_2, B_2, and C_2. A_2 and B_2 again load the Q data storage of the 'Zero' clock to five. Because terminal 2 of NOR gate 4 is at a low voltage level, C_2 passes via NOR gate 4 to EXCLUSIVE-OR gates 17 and 16 to give the trigger alert to 'Control'.

Because the Q data storage of the 'Zero' clock is now set at five, and Q_c is at a high voltage level, the state of Q_c will be transferred to the data input of the 'Zero' clock in the active module, and will finally be loaded into the Q data storage area of that clock by 'Control' to set that clock's Q data for the digit four after all alerted modules have been set-up.

7.4 The Phasor-Vernier relationship

The last 200 nanoseconds with the tens and units of nanoseconds are determined by a clock called a 'Phasor' and a second clock called a 'Vernier', both functioning in conjunction with each other. The first clock is called a phasor because it decides when the 'Vernier' should start and thus presets the phase of the termination vernier pulse relative to the 'Phasor'.

The Phasor-Vernier relationship is described in two steps.

1. Diagram 7.4.1 illustrates how the 'Vernier' is started by the ripple carry pulse put out by the 'Phasor' when the phasor's Q data storage becomes 1 with the clock pulse numbered 1, as shown in lines 7, 6, and 5 respectively.

2. Diagram 7.4.3 and following diagrams show what digits must be loaded into the 'Phasor' and 'Vernier' to establish different required time intervals

Consider Fig. 7.4.1.

Line 1. This line depicts a train of square waves being

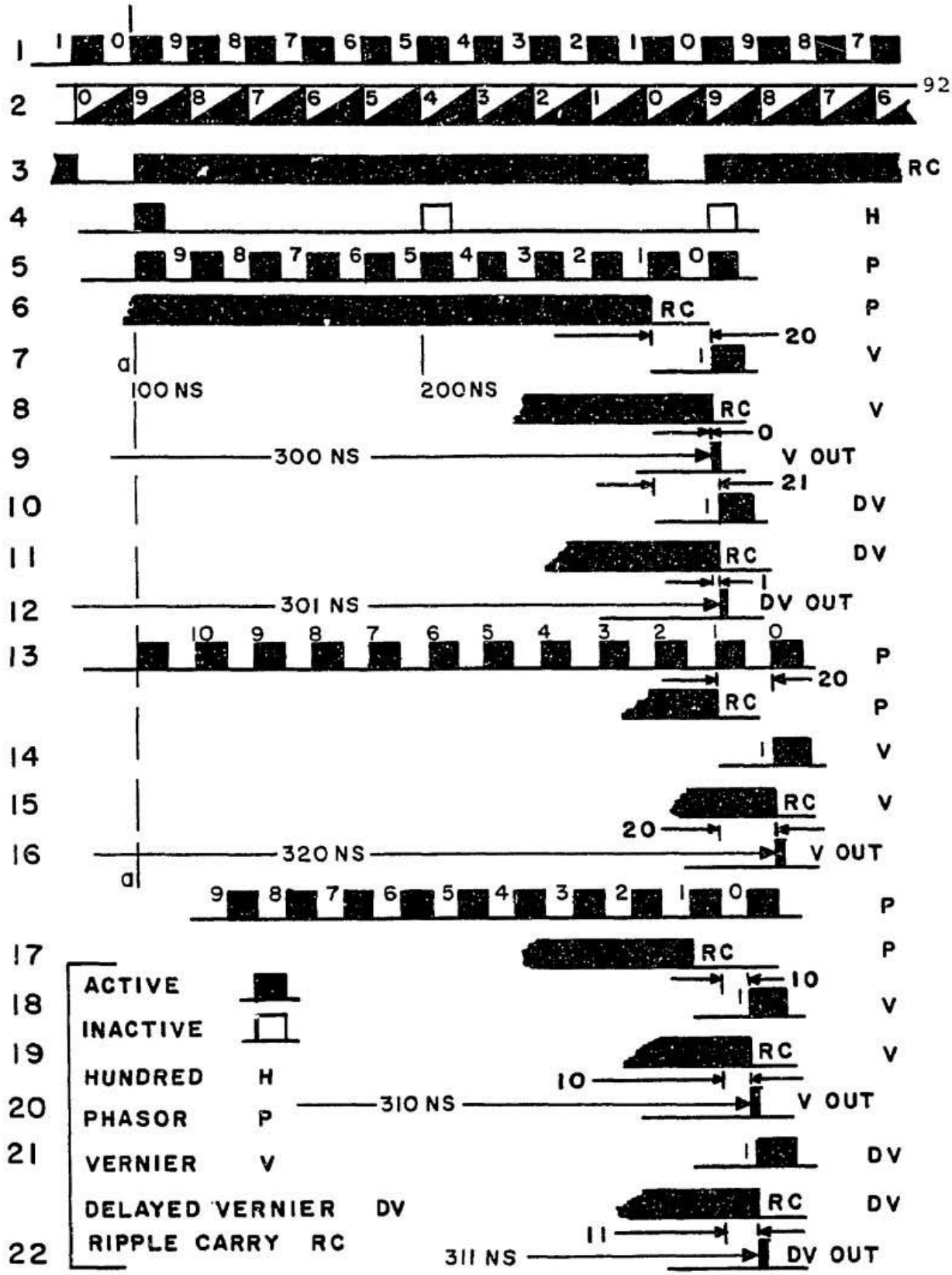


Fig. 7.4.1 PHASOR–VERNIER RELATIONSHIP

received by a digital decade counter preset to the number 9, and set in the count down mode. The numbers to the left of the pulses tell us the number of pulses that must come before the counter will reset its Q data storage states to the digit 9.

Line 2. This line gives the Q data storage states as a function of the count. This was explained in Fig. 7.1.

Line 3. Indicates that the high state of terminal 15 of this counter changes to a low state upon receipt of the next clock pulse after the Q data storage states change to the states for the digit 1. Note that this gives a falling edge to the state of terminal 15 very nearly coincident with the rising edge of pulse numbered 1 in line 1. This falling edge is required to activate oscillators, counters, flip-flops, etc. Also note the rising edge when the Q data storage states become the states for the digit 0.

Line 4. This line depicts the pulses from the 'Five' clock or the 'divide by 4' counter. As pointed out under the section "Hundred' clock, this gives a period of 100 nanoseconds.

Line 5. This line indicates that when the leading edge 'a - a' occurs, and when the 'Hundred' clock runs out, the 'divide by 4' counter becomes a 'divide by 1' counter and its output becomes the normal 50 Mhz wave train. These pulses go to the 'Phasor' which becomes active at the advent

of 'a - a'. Notice that the first ten pulses in this train are numbered backwards from 9 to 0 because the count is down. Also it is to be noted that the first pulse to be counted by the 'Phasor' comes 20 ns later than 'a - a', and its time of rising is independent of any delays in switching from the 'divide by 4' mode to the 'divide by 1' mode, the normal wave train. These ten periods after 'a - a' are each 20 ns in length, and they are required to complete the hundreds count and supply time for the 'Vernier' to function. Now suppose that we wish to clock 300 ns. There would be one pulse for the 100 ns due to the 'divide by 4' circuit to the 'Hundred' clock followed by ten 20 ns pulses to the 'Phasor'.

Line 6. Terminal 15 of the 'Phasor' changes its state to a low state when pulse numbered 1 is counted, and the 'Vernier' becomes operative due to the falling edge at terminal 15.

Lines 7 & 8. Now time zero is to be indicated by the vernier driver output pulse, and if there are no tens or units of nanoseconds, counting must terminate with the first edge of the 'Vernier' coincident on the zero state of the 'Phasor'. Since terminal 15 of the 'Vernier' can also go to the low state on the next clock pulse after its Q data states have become the states for the digit 1, and since the 'Vernier' is also in the down counting mode, we see that the 'Vernier' data input must be set to one in order to have time zero take place at the proper time. Referring to the

vernier oscillator section, it is stated there that when that oscillator is enabled, 20 ns will elapse before the first rising edge appears, after which all rising edges occur at intervals of 22 nanoseconds. Thus, if the 'Phasor' is set at nine, its terminal 15 will put out a falling edge when its Q data states change from the states for the digit 1 to the states for the digit 0. This activates the vernier oscillator to give a leading edge 20 ns after it has been enabled, thus causing the 'Vernier' to change the state of its terminal 15 to a low state. The falling edge occuring with this change of voltage level indicates time zero count for the 300 ns. This indicates that the 'Vernier' should have been preset to have the Q data states representing the digit 1.

Line 9. The vernier driver puts out a positive pulse so that its leading edge occurs at time zero. This is the undelayed vernier system.

Lines 10 & 11. If we wish to add 1 ns to the 300 ns time interval, we delay the activating signal to the 'Vernier' by 1 ns. Then the state of vernier terminal 15 will change to the low state 1 ns after the hundreds count has been completed. This is the delayed vernier system.

Line 12. The vernier driver output pulse occurs 1 ns after the hundreds count has been completed. Thus we have counted 301 nanoseconds.

Line 13. If one is added to the 'Phasor' input data, so as to increase the count from 9 to 10, then the phasor terminal 15 will put out its low voltage state one period or 20 ns after the time the hundreds count has run out. This would add 20 ns to the 300 ns giving 320 ns for the undelayed vernier system. For the delayed vernier system, we would have 321 ns. But since we now require to count the number 10 instead of the number 9, we must use a binary counter here instead of a decade counter. This gives us a new counting range extending from 15 to 0 consecutively in the count down mode.

Line 17. If now we delay the enabling signal to the 'Phasor' by 10 ns while the 'Phasor' is set at 9, we will cause the phasor terminal 15 to change its voltage state to the low state 10 ns after the hundreds count has been completed. Then the delayed phasor system and the undelayed vernier system combination gives 310 ns.

Line 20. The vernier driver output pulse appears at the end of 310 ns.

Lines 21 & 22. For the delayed phasor system and the delayed vernier system combination, we see the vernier driver output pulse coming at the end of 311 ns.

In this manner, any time period can be measured. This is part of the explanation of the sliding vernier. Thus each digit added to the phasor storage will alter the phasor counting time by 10 or 20 ns. But note that this change in

the phasor count will be arranged by the 'Shifter'. The 'Shifter' clock contains the information as to whether the 'Phasor' operates on the normal or inverted clock. This permits the 'Phasor' to count either the odd or even number of tens. This was mentioned earlier, and will be fully explained shortly. With regard to this point, note that in line 17, the 'Phasor' saw the inverted clock.

Now we turn to the machine language required to carry out all the foregoing time interval generating operations. This language or code will make it possible to keep track of the system. Code tables, Table 7.4.1 to Table 7.4.6, are employed to show how this code was designed and to explain how the system functions.

Consider Table 7.4.1. The digits, 0 to 9, are each designated for digital circuits by a four element code, employing only two numbers, 0 and 1, as shown in Table 7.4.1, which gives the digital representation for each of these digits in our reversed order.

Notice that if column one is ignored, then there are duplicate rows of combinations of zeros and ones. That is for the digits 0 and 1 we have (0 0 0) and for the digits 2 and 3, we have (1 0 0). Thus by cutting off the first digit, a new code can be formed. Since we use 4 bit words, a null digit, say 0, is added to the remaining trios of digits. This additional fourth digit is needed to meet the requirements of the counter's four inputs. Then the

TABLE 7.4.1 BINARY CODE FOR DIGITS 0 TO 9

DIGIT N	CODE
0	0 0 0 0
1	1 0 0 0
2	0 1 0 0
3	1 1 0 0
4	0 0 1 0
5	1 0 1 0
6	0 1 1 0
7	1 1 1 0
8	0 0 0 1
9	1 0 0 1

first digit, which we cut out of the code, can be used to determine whether the normal or inverted clock is employed. By this means, we end up with Split Code 1 given in Table 7.4.2.

Thus when a digit, say N_o, is dialed into the system, the first part of the code for that digit, let it be lettered X_o, decides whether the system sees 0 or 1, i.e. - whether the count is even or odd. If it sees 0, then only the even digits are possible, i.e. - 0, 2, 4, 6, and 8. If it sees 1, then the possible digits are 1, 3, 5, 7, and 9.

The remaining numbers in the split code, plus the dummy 0 in the fourth position, may now represent one of the digits 0, 1, 2, 3, and 4. Let Y_o stand for this digit. But this has made a new code as shown in Split Code 2 of Table 7.4.2. Here we now have:

0 = (0 0), 1 = (1 0), 8 = (0 4), 9 = (1 4), etc. Thus we now let N_o = (X_o Y_o).

Now in the case where we call for 0 ns, we require the 'Vernier' to go low on 1. Then to the second member of the split code, Y_o, one is added, changing the code for 0 from 0 = (0 0) to 0 = (0 1). This is the true code for 0 nanoseconds. In this manner, Split Code 3 was created, and it is given in Table 7.4.2. Let Y_1 stand for this new member of the new split code, i.e. -

$$N_1 = (X_o \; Y_1).$$

TABLE 7.4.2 SPLIT CODES 1, 2, AND 3

	SPLIT CODE 1			SPLIT CODE 2		SPLIT CODE 3	
	N_o			N_o		N_1	
	SPLIT CODE			SPLIT CODE		SPLIT CONE	
	X_o	Y_o		X_o	Y_o	X_o	Y_1
	STATE		DIGIT	STATE	DIGIT	STATE	DIGIT
0	0	0 0 0 0	0	0	0	0	1
1	1	0 0 0 0	0	1		1	
2	0	1 0 0 0	1	0	1	0	2
3	1	1 0 0 0	1	1		1	
4	0	0 1 0 0	2	0	2	0	3
5	1	0 1 0 0	2	1		1	
6	0	1 1 0 0	3	0	3	0	4
7	1	1 1 0 0	3	1		1	
8	0	0 0 1 0	4	0	4	0	5
9	1	0 0 1 0	4	1		1	

Now we need a code for the "Phasor' so that its terminal 15 will put out a low state voltage at the proper time - when its Q data states represent the digit 1. Notice that from line 4 of the pulse time chart, Fig. 7.4.1. the 'Phasor' must be set to 9 causing the 'Phasor' to count 9 x 20 = 180 ns, while the 'Vernier' is to count the remaining 20 nanoseconds.

Now returning to Split Code 1, Table 7.4.2, and remembering that the extra column is set to 0, take inversions of the first three digits of that code, i.e.- for 0 ns, 0 = (0 0 0 0) becomes 0 = (1 1 1 0), to get a new partial code for this time of 0 ns. Now notice that according to the binary codes given in Table 7.4.1, this new partial code stands for the digit 7. But recall that for 0 ns, the 'Phasor' was to be set at 9. Thus we must add 2 to this partial code to get the required number 9. That is: 7 + 2 = (1 1 1 0) + (0 1 0 0) = (1 0 0 1) = 9. Recall, all binary numbers are written here in reverse order so that they match the order of the counter's inputs.

Now add 2 to each line of the complement of Y_o of Table 7.4.3. This indicates how we get a new code that will set the 'Phasor' so that its terminal 15 will put out a low voltage state at the right time. In this manner, the preliminary phasor code $N_2 = (X_o \; Y_2)$, Y_2 coming from Table 7.4.3. Note that the second member of this code ranges from 9 to 5 inclusive.

TABLE 7.4.3 PRELIMINARY PHASOR CODE

Y_o	COMPLEMENT OF Y_o		ADD 2 Y_2
0 0 0 0	1 1 1 0	7	9
0 0 0 0	1 1 1 0		
1 0 0 0	0 1 1 0	6	8
1 0 0 0	0 1 1 0		
0 1 0 0	1 0 1 0	5	7
0 1 0 0	1 0 1 0		
1 1 0 0	0 0 1 0	4	6
1 1 0 0	0 0 1 0		
0 0 1 0	1 1 0 0	3	5
0 0 1 0	1 1 0 o		

Now consider the tens of nanoseconds. Split Code 2 of Table 7.4.2 shows the code for the tens developed to give a short code involving either 0 or 1 in the first position, and one of 0, 1, 2, 3, or 4 in the second position. That is $N_o = (X_o \; Y_o)$. Now note that Y_o is the digit that appears in the shifter data input storage.

Then to develop the full phasor code, we take X_o to indicate even (0) or odd (1) tens, and Y_o is added to Y_2. Recall that the digits 0 to 4 represented in Split Code 2 of Table 7.4.2 by Y_o are added to the phasor data by means of the 'Shifter' clock. The full phasor code with shifter data added is given by Table 7.4.4. Now N_o becomes:

$$N_3 = (X_o \; Y_3).$$

Table 7.4.5 gives examples of the joining together of two codes to get a complete code for the "Precise Time Interval System", while Table 7.4.6 lists the codes for all periods of time ranging from 0 to 99 nanoseconds. Recall that the hundreds of nanoseconds are supplied by the 'Hundred' clock.

The pulse time chart, Fig. 7.4.2 illustrates the principle behind the 'Vernier'. Note that, as the number of required nanoseconds increases, ranging from 0 to 9, the 'Vernier' must start earlier in time. Also note that the first line shows only the last five pulses supplied to the 'Phasor'. These pulses demonstrate the relationship of the 'Vernier' to the 'Phasor', showing how the correct vernier

TABLE 7.4.4 FULL PHASOR CODE

UNITS NS	X_o	Y_2	Y_o SHIFTER				
			+0	+1	+2	+3	+4
0	0	9	9	10	11	12	13
1	1						
2	0	8	8	9	10	11	12
3	1						
4	0	7	7	8	9	10	11
5	1						
6	0	6	6	7	8	9	10
7	1						
8	0	5	5	6	7	8	9
9	1						

TABLE 7.4.5 SAMPLE OF FULL CODE

NS	PHASOR CODE	VERNIER CODE	FULL CODE
0	(0 9)	(0 1)	(0 9 0 1)
10	(1 9)	(0 1)	(1 9 0 1)
1	(0 9)	(1 1)	(0 9 1 1)
11	(1 9)	(1 1)	(1 9 1 1)
22	(0 9)	(0 2)	(0 9 0 2)
32	(1 9)	(0 2)	(1 9 0 2)
85	(0 11)	(1 3)	(0 11 1 3)
95	(1 11)	(1 3)	(1 11 1 3)

TABLE 7.4.6 FULL CODE FOR 0 TO 99 NANOSECONDS

NS	CODE	NS	CODE	NS	CODE	NS	CODE	NS	CODE
0	(0 9 0 1)	20	(0 10 0 1)	40	(0 11 0 1)	60	(0 12 0 1)	80	(0 13 0 1)
1	(0 9 1 1)	21	(0 10 1 1)	41	(0 11 1 1)	61	(0 12 1 1)	81	(0 13 1 1)
2	(0 8 0 2)	22	(0 9 0 2)	42	(0 10 0 2)	62	(0 11 0 2)	82	(0 12 0 2)
3	(0 8 1 2)	23	(0 9 1 2)	43	(0 10 1 2)	63	(0 11 1 2)	83	(0 12 1 2)
4	(0 7 0 3)	24	(0 8 0 3)	44	(0 9 0 3)	64	(0 10 0 3)	84	(0 11 0 3)
5	(0 7 1 3)	25	(0 8 1 3)	45	(0 9 1 3)	65	(0 10 1 3)	85	(0 11 1 3)
6	(0 6 0 4)	26	(0 7 0 4)	46	(0 8 0 4)	66	(0 9 0 4)	86	(0 10 0 4)
7	(0 6 1 4)	27	(0 7 1 4)	47	(0 8 1 4)	67	(0 9 1 4)	87	(0 10 1 4)
8	(0 5 0 5)	28	(0 6 0 5)	48	(0 7 0 5)	68	(0 8 0 5)	88	(0 9 0 5)
9	(0 5 1 5)	29	(0 6 1 5)	49	(0 7 1 5)	69	(0 8 1 5)	89	(0 9 1 5)
10	(1 9 0 1)	30	(1 10 0 1)	50	(1 11 0 1)	70	(1 12 0 1)	90	(1 13 0 1)
11	(1 9 1 1)	31	(1 10 1 1)	51	(1 11 1 1)	71	(1 12 1 1)	91	(1 13 1 1)
12	(1 8 0 2)	32	(1 9 0 2)	52	(1 10 0 2)	72	(1 11 0 2)	92	(1 12 0 2)
13	(1 8 1 2)	33	(1 9 1 2)	53	(1 10 1 2)	73	(1 11 1 2)	93	(1 12 1 2)
14	(1 7 0 3)	34	(1 3 0 3)	54	(1 9 0 3)	74	(1 10 0 3)	94	(1 11 0 3)
15	(1 7 1 3)	35	(1 8 1 3)	55	(1 9 1 3)	75	(1 10 1 3)	95	(1 11 1 3)
16	(1 6 0 4)	36	(1 7 0 4)	56	(1 8 0 4)	76	(1 9 0 4)	96	(1 10 0 4)
17	(1 6 1 4)	37	(1 7 1 4)	57	(1 8 1 4)	77	(1 9 1 4)	97	(1 10 1 4)
18	(1 5 0 5)	38	(1 6 0 5)	58	(1 7 0 5)	78	(1 8 0 5)	98	(1 9 0 5)
19	(1 5 1 5)	39	(1 6 1 5)	59	(1 7 1 5)	79	(1 8 1 5)	99	(1 9 1 5)

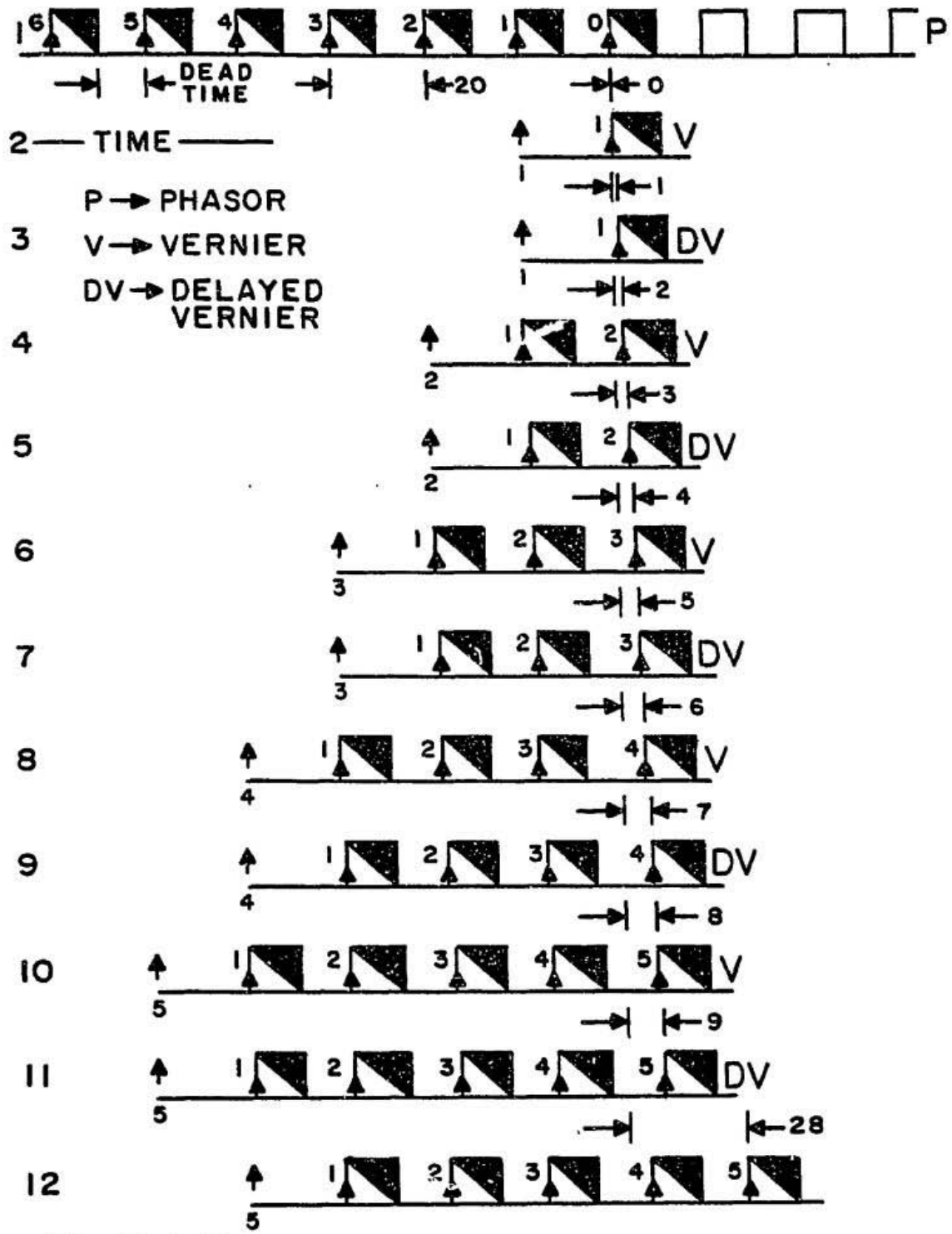


Fig. 7.4.2

PHASOR VERNIER PULSE TIME CHART

time is found. One can easily see the number of vernier pulses needed to cause a digital time shift between the leading edge of the zero phasor pulse and the required leading edge of the 'Vernier'.

The vernier pulses carry two sets of numbers, one set above and another set below each wave train. The upper number gives the number of pulses needed to give a required shift in time, while the lower ones tell to what number the 'Vernier' should be set. Recalling that terminal 15 of the 'Phasor' must put out a low voltage state when the phasor Q data states represent the digit 1 in order to enable the 'Vernier', we see from this chart that for a 2 ns shift, that pulse numbered 2 will have to be renumbered to 1. In like manner for a 4 ns shift, although the chart shows the 'Vernier' being enabled on pulse numbered 4, the 'Phasor' will have to look at this pulse as if it were numbered 1. This simply means that as the number of nanoseconds required increases, the phasor data is reduced. Thus the 'Vernier' is started earlier on the phasor train of pulses.

The pulse time chart, Fig. 7.4.3 shows the phasor pulses correctly numbered for 0, 2, 4, 6, and 8 ns. Here the 'Phasor' is set at 9, 8, 7, 6, and 5 respectively. Since the 'Phasor' enables the 'Vernier' when its Q data states change from the states representing the digit 1 to the states for the digit 0, and that on the next pulse the 'Phasor' is disabled, we see that the 'Phasor' does not always count down the full 200 ns after the 'Hundred' clock

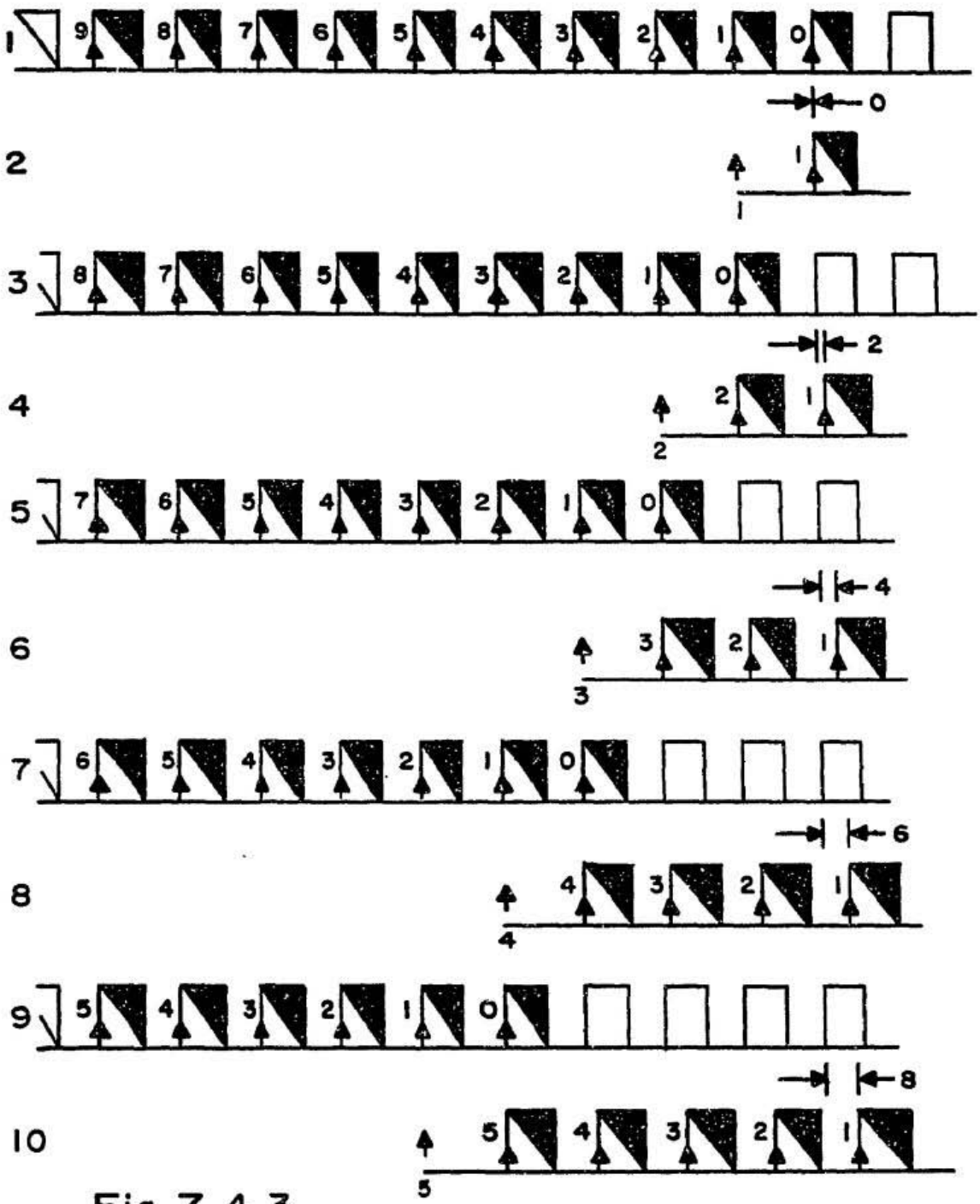


Fig. 7.4.3
TRUE PHASOR VERNIER PULSE NUMBERING

runs out. Thus whatever time is left of these 200 ns, it is covered by the vernier action of the 'Vernier'. Remember that the first vernier pulse is due 20 ns after the 'Vernier' is enabled. It is important to note that since the 'Phasor' foreshortens its scale while the 'Vernier' lengthens its scale, we have in fact a double vernier, creating the equivalent of a sliding vernier. By this method, any time period from 0 to 200 ns can be obtained. Also, the only delays to affect our results are:

1. the time of rise of the first leading edge from the 50 Mhz slave oscillator, and

2. the time of rise of the first leading edge from the vernier slave oscillator. The delay in the rising of this edge is controlled by the digital delay line to within a discrepancy of 50 picoseconds.

All other delays in switching, such as those occuring in the switching from the 'divide by 4' mode to the 'divide by 1' mode, are unimportant, since these delays occur during the inactive or dead time of the oscillator pulses. These delays are less than 3 ns, while from the pulse time chart, Fig. 7.4.2, the dead times are shown to be 10 and 11 ns for the 50 Mhz and vernier slave oscillators respectively. The counters are only active on the leading edge of the clock pulse.

Note again that the 'Vernier' must occupy some of the time set aside for the phasor-vernier operation.

Note that in the case of time intervals less than 199ns, the 'Hundred' clock is idle in the Active division. In order to accomodate such time intervals, periods that are less than 199 ns must be divided into two allotments.

1. 199 to 99 nanoseconds.
2. 99 to 0 nanoseconds.

Examination of Fig. 7.4.2 will show that there are always at least five phasor pulses available to start the 'Vernier'. Now if we subtract five from the phasor data storage, we are removing 100 ns from the 200 ns - the time allotted to the 'Phasor'. Since the 'Vernier' can give a maximum period of 99 ns, we can now easily obtain 199 ns. This brings us to the reason for the 'Q clock', which tests the 'Hundred' clock and controls the 'Zero' clock depending upon the data input to the 'Hundred' clock. This has been mentioned earlier. The testing of the 'Hundred' clock is further dealt with under the two sections: The Hundred clock, and The Q clock.

It remains to be stated that the Q clock has two outputs, Q_1 and Q_2. When Q_1 = + (high) and Q_2 = 0 (low), the data input to the 'Zero' clock in the Passive division is set at five. Then during the set-up operation, five is subtracted from the data sum of the 'Phasor' and 'Shifter' data by means of the 'Zero' clock which can activate the 'Phasor' clock and allow the subtraction of data from the storage of that clock until the zero clock data becomes zero. Thus the 'Q clock' automatically controls the

'Phasor' clock. Note that for all time delay intervals greater then 99 ns, there are no system delays affecting the counting, and the delay counting can be stimulated by an external trigger (see Fig. 7.4.4).

Now if we require periods of time ranging from 99 to 0 ns, then the 'Q clock' will give:

$$Q_1 = + \text{ (high)}, \quad Q_2 = + \text{ (high)}.$$

In this case, five will first be subtracted from the data storage of the 'Phasor' as above, and the input of the 'Zero' clock is once more set to five. Now 'Control' will set the data input to the 'Zero' clock in the Active division to four using the state of Q_c and load four into the Q data storage of this clock in the Active division.

Now during the actual delay time counting, the active zero clock will start and run concurrently with the 'Phasor'. The falling edge of the high voltage state of terminal 15 of the 'Zero' clock occurs when the Q data storage becomes zero, but the positive rising edge of this voltage level coincident with the resetting pulse will now indicate the commencement of the interval. The interval will be terminated by the leading edge of a positive pulse stimulated by the occurence of the Q data low voltage level output of the 'Vernier'. The advent of these two pulses defines the interval. Some examples of intervals so defined are shown in Fig. 7.4.5, namely for 0, 4, and 8 nanoseconds.

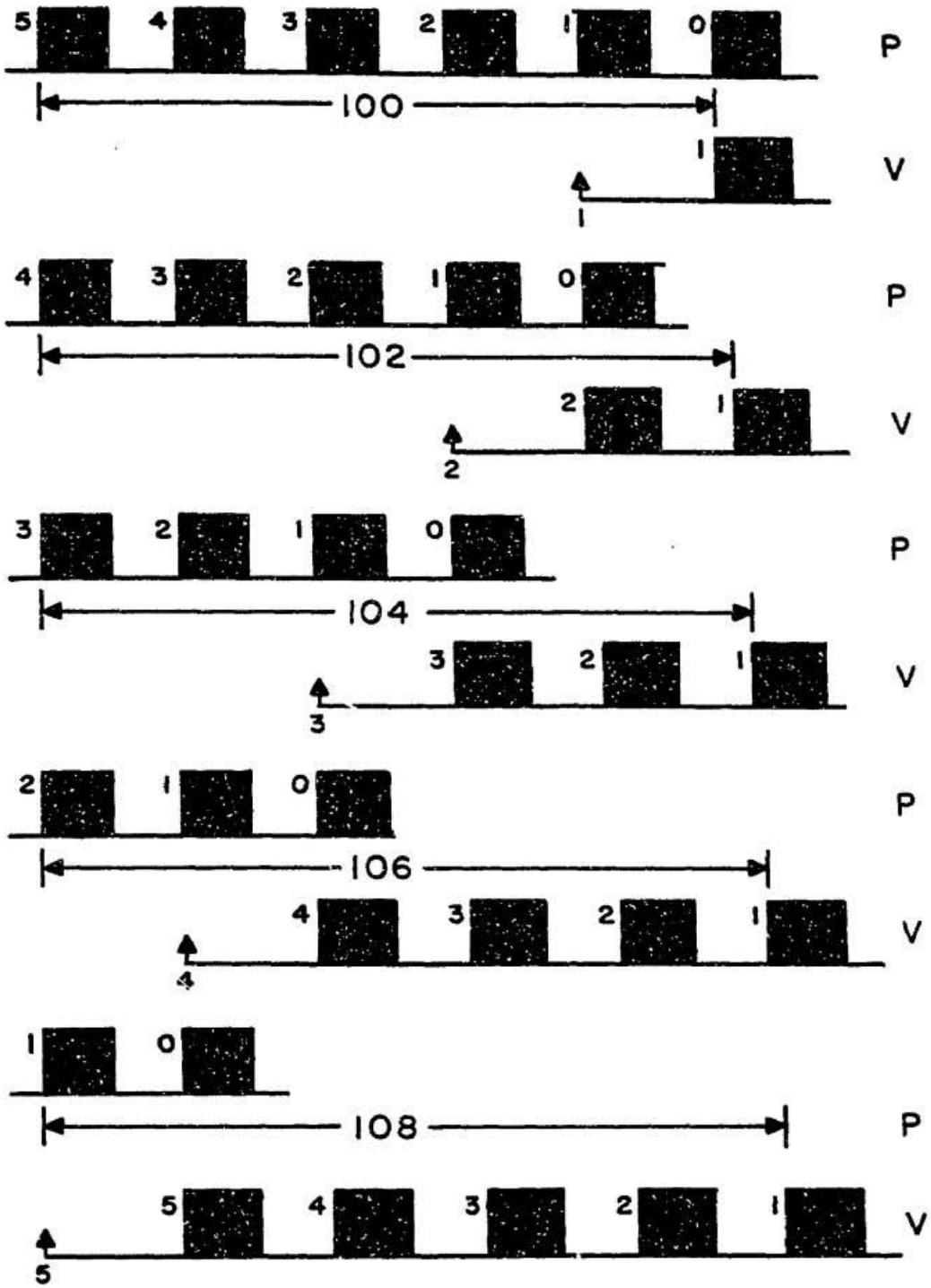


Fig. 7.4.4

REDUCTION OF PHASOR OPERATIONAL TIME

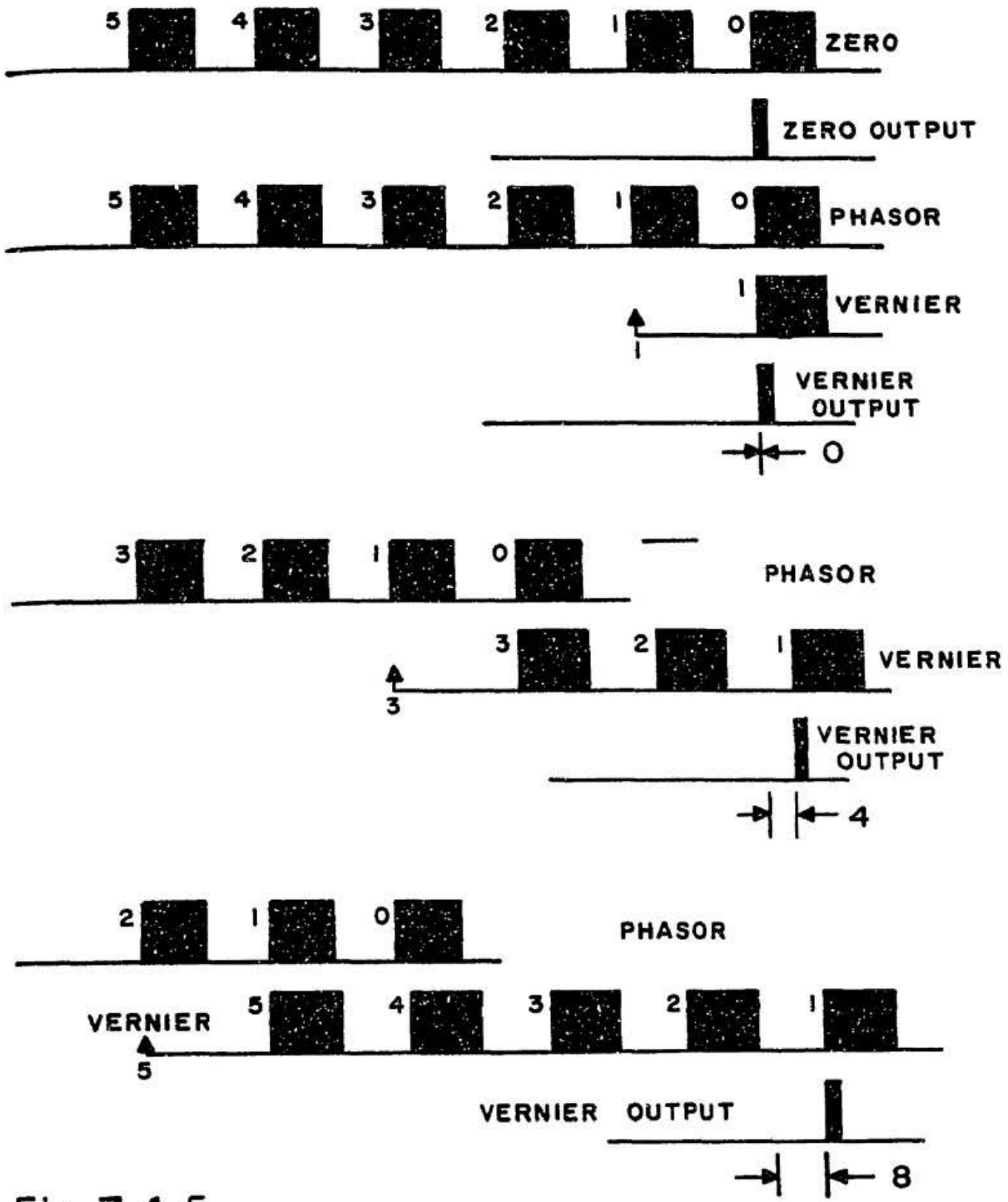


Fig. 7.4.5
TIME INTERVAL DEFINITION FOR 0, 4, AND 8 NANOSECONDS

7.5 The set-up function

The set-up operation converts the input data into machine language. This system is unique in that an electronic sliding vernier has been invented. The 'Zero' clock is also a vernier system. Recall that for this paper, a clock is defined as any device such as a counter, flip-flop, or specially designed circuit, etc. that counts or keeps track of pulses singularly or in predetermined groups.

The set-up source is a high speed square wave oscillator located in the control division. This oscillator supplies the wave train that is used to time all set-up functions, and further it provides loading pulses A_o and B_o for the Active division. The frequency can be as high as 1 Mhz, but the present value is 600 KHz. This oscillator becomes active when inserted into the passive modules, and will continue to operate until all modules are updated. It is isolated from each module as soon as that module is fully updated.

During the time that the set-up function is active on any module that is alerted, the following operations take place in each module independently. That is we have parallel operation, but the specific steps are occuring not necessarily simultaneously in different modules. Fig. 7.5.1 will assist in following the various stages in the process of updating a module.

Stage 1. The dual gate selection option 1 is chosen, the

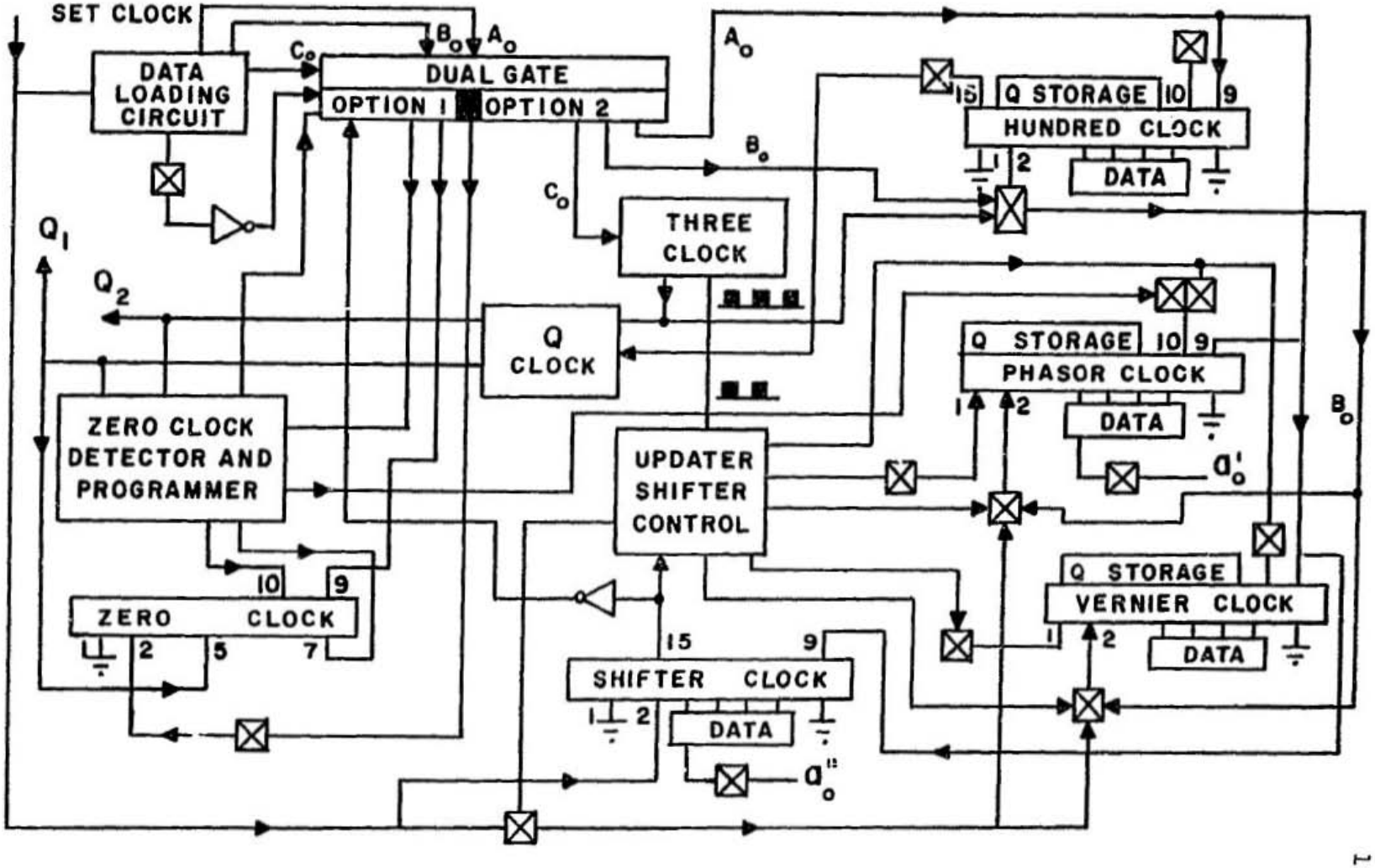


Fig. 7.5.1 BLOCK DIAGRAM OF A PASSIVE MODULE

'Q clock' set, and the set-up source inserted into each module simultaneously.

Stage 2. Time data is loaded into the Hundred, Phasor, Vernier, and Shifter clocks via pulses A_o and B_o. Note that the 'Zero' clock has no data loaded into its Q data storage at this time.

Stage 3. 'Q clock' adjustments, if any, are made, thus deciding whether the input to the 'Zero' clock is 0 or 5.

Stage 4. The 'Phasor' and 'Vernier' clocks are adjusted upwards by two and one counts respectively by the Updater unit.

Stage 5. The data in the Q data storage of the 'Shifter' clock is added to the Q data storage of the 'Phasor' clock via the Shifter control.

Stage 6. Case 1. The 'Zero' clock input is 0 for case 1 of the 'Q clock' ($Q_1 = 0$, $Q_2 = 0$). The option 1 is maintained, and a trigger pulse alerts 'Control' that the module is ready.

Case 2. If the input to the 'Zero' clock is 5, then option 2 of the dual gate is chosen and five is now subtracted from the Q data storage of the 'Phasor'. Now referring to the section entitled Q clock, this is the case for Q_1 = + (high), and Q_2 = 0 (low). Then the trigger pulse alerts 'Control that the module is ready.

Case 3. In this case, Q_1 = + (high) and Q_2 = + (high). Now option 2 is chosen by the dual gate, and after 5 has been subtracted from the Q data storage of the 'Phasor' clock as in case 2, then the active zero clock memory is loaded with the data Q_1 = + in order to control the input of the 'Zero' clock of the Active division. Then the trigger pulse is sent to 'Control'.

Upon receipt of all four trigger pulses, or one from each alerted module, 'Control' isolates the set-up oscillator from all passive modules, and loads the updated time data into the Q data storage areas of the Hundred, Phasor, Vernier, and Zero clocks in the active modules.

The set-up operation is now completed. Since the new data is stored in the Q data storage areas of the Hundred, Phasor, Vernier, and Zero clocks of the passive modules, this data can be reloaded into the Q data storage areas of the active modules using two loading pulses from 'Control'. If S type digital integrated circuits are used in the construction of the system, frequencies up to 70 Mhz may be employed.

The firing pulse enables each slave 50 Mhz oscillator in each alerted module.

Consider Fig. 7.5.2. Here in chart form, all the operations involved in the set-up function are clearly laid out on numbered lines in order of their occurence vertically, while the time appears horizontally across the

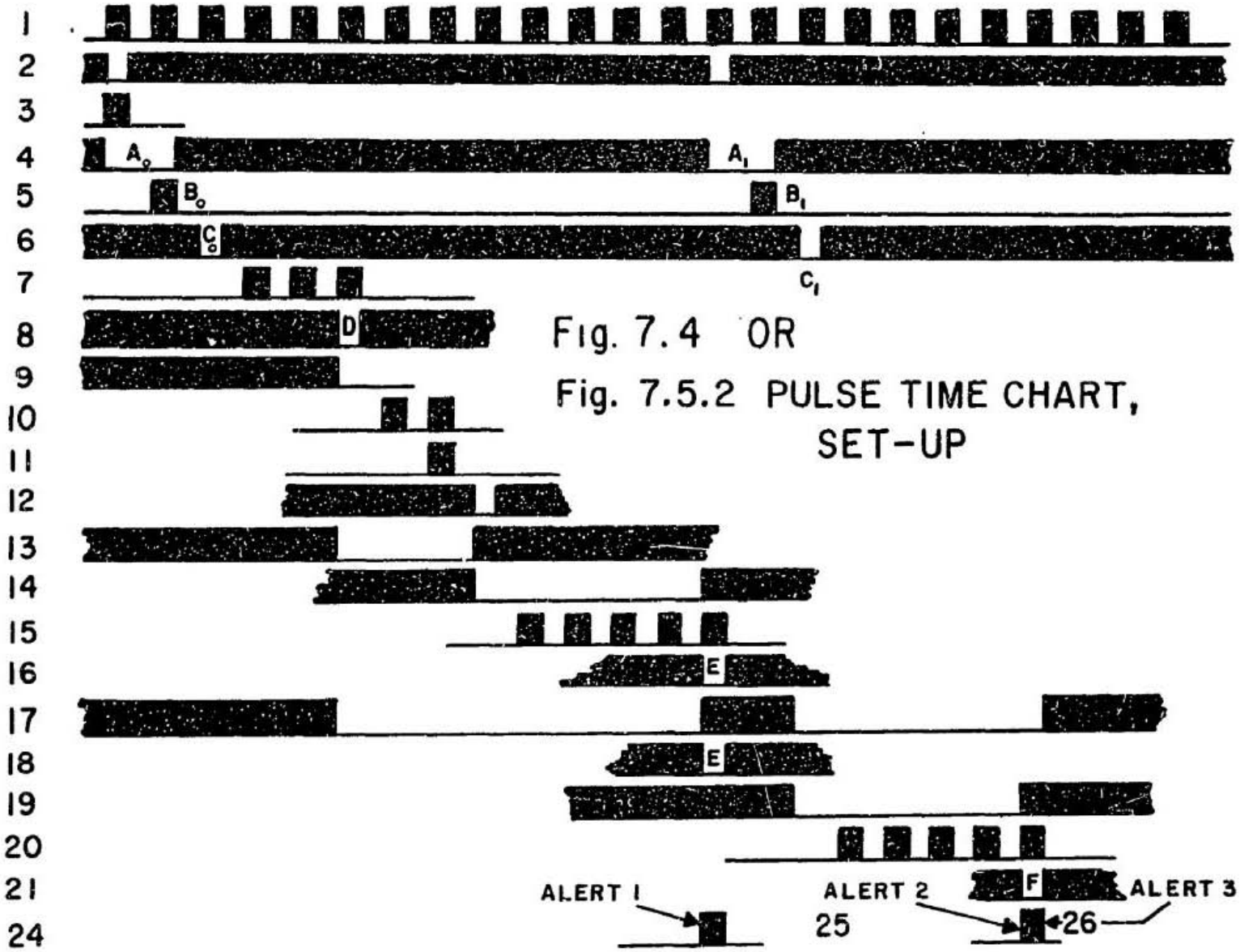


Fig. 7.4 OR

Fig. 7.5.2 PULSE TIME CHART, SET-UP

79-119

page for each event, time being zero at the left hand margin of the chart.

Line 1. The set-up oscillator inserted into each passive module.

Line 2. Inverted pulse, Set pulse 1. Required to make certain that the various circuits in each module start in the correct modes. Activation of data loading.

Line 3. The pulse of line 2 inverted. Option 1, count periods of 100 nanoseconds.

Line 4. The broad low voltage state pulse A_o delivered to terminals 9 of the Hundred, Phasor, Vernier, and Shifter clocks.

Line 5. The narrow high voltage state clocking pulse B_o delivered to terminals 2 of the clocks named in line 4. Loading of data into the Q data storage areas is completed.

Line 6. A narrow low voltage state pulse C_o is provided as a continuation pulse. It automatically activates the next circuit - the 'Three' clock.

Line 7. The 'Three' clock puts out three positive pulses, of which two pulses are employed to test the data in the 'Hundred' clock storage areas and set the 'Q clock'.

Line 8. This is the inversion of the third pulse from the 'Three' clock. It is lettered 'D'. It is also a continuation pulse used to activate the Updater circuit.

This pulse also causes the 'Phasor' and 'Vernier' clocks to:

1. have their counting directions changed from the down to the up counting mode, and
2. each clock is turned on.

Lines 10 & 11. The Preshifter clock causes two pulses and one pulse to be added to the Q data storage of the 'Phasor' and 'Vernier' clocks respectively. Then the 'Vernier' clock changes back into the count down mode and is isolated.

Line 12. The Preshifter clock now sends a low voltage state pulse to the 'Shifter' clock, turns it on, and causes the data in the Q data storage of the 'Shifter' clock to be added to the data storage of the 'Phasor' clock. Note that at this time the 'Zero' clock is idle because it is isolated from all the above operations.

Line 16. The low voltage state pulse E put out by the 'Shifter' clock at zero count on the 'Shifter clock. Option 2 of the dual gate is chosen.

Line 24. If $Q_1 = 0$ and $Q_2 = 0$, pulse E alerts 'Control' that the module is prepared.

Line 18. If $Q_1 = +$ and $Q_2 = 0$, pulse E reactivates the data loading circuit, yielding pulses A_o, B_o, and C_o, now redesignated as A_1, B_1, and C_1 for clarity. The set-up clock is still connected to the 'Phasor' and 'Zero' clocks,

so that pulses A_1 and B_1 reach terminals 9 and 2 of the 'Zero' clock respectively, causing 5 to be placed in the Q data storage of this later clock. C_1 is a continuation pulse which then turns on the 'Zero' and 'Phasor' clocks. Since both of these clocks are in the down counting mode, 5 is subtracted from the Q data storage of the 'Phasor' clock. Note that these pulses represent a time of 100 ns, the period of time by which the running time of the 'Phasor' will be shortened in the actual timing operation by the active module.

Line 21. When the Q data storage of the 'Zero' clock becomes zero, this clock puts out a low pulse F via its terminal 15, and this pulse turns off both the 'Phasor' and 'Zero' clocks. In addition, this pulse alerts 'Control' that the module is ready (Ready Alert 2).

Line 26. If Q_1 = + and Q_2 = +, then the chain of events for lines 19 and 20 take place, except that the low voltage state pulse F now goes to the zero clock memory in the Passive division. This memory will set the 'Zero' clock in the active module to the required mode. 'Control' also receives pulse F as ready Alert 3.

When all modules are ready, the set-up clock is disconnected from the passive modules, and the updated data is loaded into the Q data storage areas in the active modules by means of two loading pulses supplied by 'Control'.

CHAPTER 8

A HIGH SPEED PRECISION GATED OSCILLATOR

This oscillator was designed to operate at median frequencies of either 50 Mhz to give a period of 20 nanoseconds, or at $45.\dot{4}\dot{5}$ Mhz to give a period of 22 nanoseconds.

The chief and most vital requirement of the oscillator was that when it is gated on, or enabled, the rising edge of the first clock pulse must appear early enough in time so that its actual time of appearing at the output terminals of the circuit may be adjusted to give a 20 nanosecond delay, which will match the period of the 50 Mhz oscillator. This would permit the counting of intervals of 20 ns to give an accurate account of time, commencing with the enabling of the oscillator. This also makes it possible to commence counting time at any time and have the uncertainty removed that would exist if we tried to break into a wave train of pulses being emitted by a continually operating oscillator.

The second important required characteristic was that the leading edge of the first pulse should not be distorted.

Third, all pulses thereafter should occur at precise intervals of 20 or 22 nanoseconds respectively.

It was not possible to find a commercial TTL

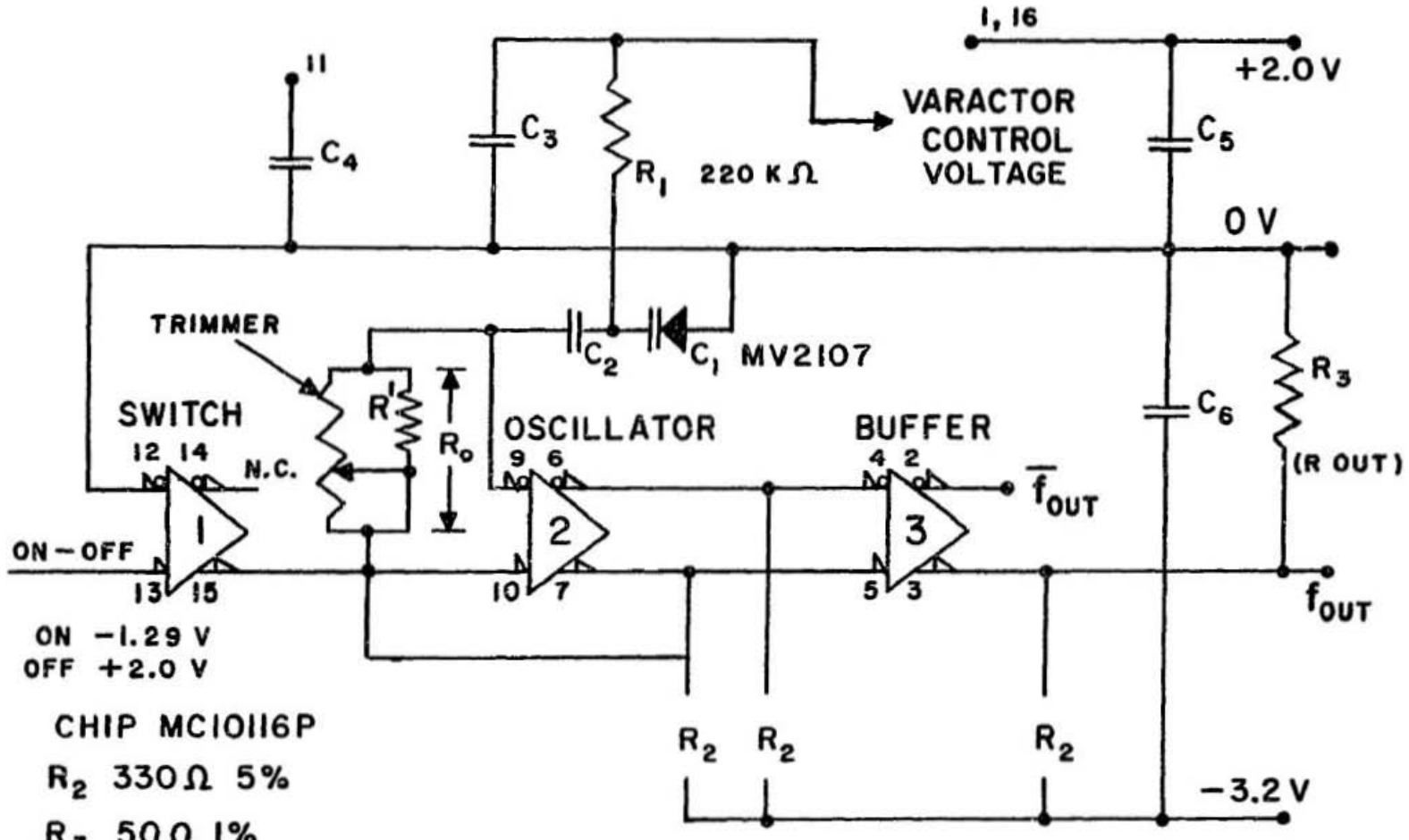
chip that would fulfill these requirements. Either the first pulse took too long to occur after the oscillator was enabled, or the leading edge of the first pulse was not regular.

Eventually it was decided to attempt the design of the required oscillator based upon the operational characteristics of a differential amplifier. Finally the line receiver MC10116P, which incorporates three identical differential amplifiers in one chip, was investigated. (5, 7, 8).

Fig. 8.1 is a schematic diagram of the final oscillator. Amplifier 2 was used to design a high speed Schmidt trigger by first reducing the resistance of the feedback line to virtually zero ohms.

The tuning of this oscillator is accomplished by the use of varactor MV2107 in series with a pair of 10 pf capacitors. The control voltage for this varactor may be adjusted from +0.6 volts to +10 volts.

The tuning resistor consists of a 100 ohm resistor in parallel with a 5K ohm variable resistor, and the 100 ohm resistor is connected from the moveable tap to one end of the variable resistor. The open end of the variable resistor is then joined by a piece of wire to the center tap in order to remove possible reflections from an open resistor wire end. By means of this assembly, the tuning resistance can be varied from 98 ohms to very nearly



CHIP MC10116P

R_2 330Ω 5%

R_3 50Ω 1%

C_2 TWO 10pf SILVER MICA (IN PARALLEL)

C_3 C_4 C_5 C_6 0.1 μf.

Fig. 8.1 HIGH SPEED PRECISION GATED OSCILLATOR

zero ohms.

The third amplifier serves as a buffer. It also acts as a pulse shaper since it improves the rate of fall of the trailing edge of the pulse and thus makes the pulse more symmetrical.

The first amplifier acts as the turn on or gating switch. The voltage supply and all control voltages such as the varactor control voltage are well buffered by the use of capacitors.

When finally functional, the prototype performed as follows. The on-off switching levels were, -1.27 volts and +2.0 volts respectively. The output wave train was as shown in Fig. 8.2. In this diagram, the oscillator appeared to be gated on 4.5 ns after the gate was closed. The square wave oscillator used to switch the oscillator on had a slow rate of fall for the falling edge of its pulse and this made it difficult to read the time of actual turn on of the oscillator.

Nevertheless, the first pulse rose in 14.7 ns after the aforementioned gating, and thus was within the 20 ns period required. It appeared that if the gating pulse were a fast pulse, that is a pulse with a steep falling edge, this 4.5 ns period might be reduced to 3 ns. If this reduction is possible, it could be very beneficial because this saving in time could be used to absorb time delays in the output transducers. If proper test equipment had been available to me, I would have replaced the MC10116P

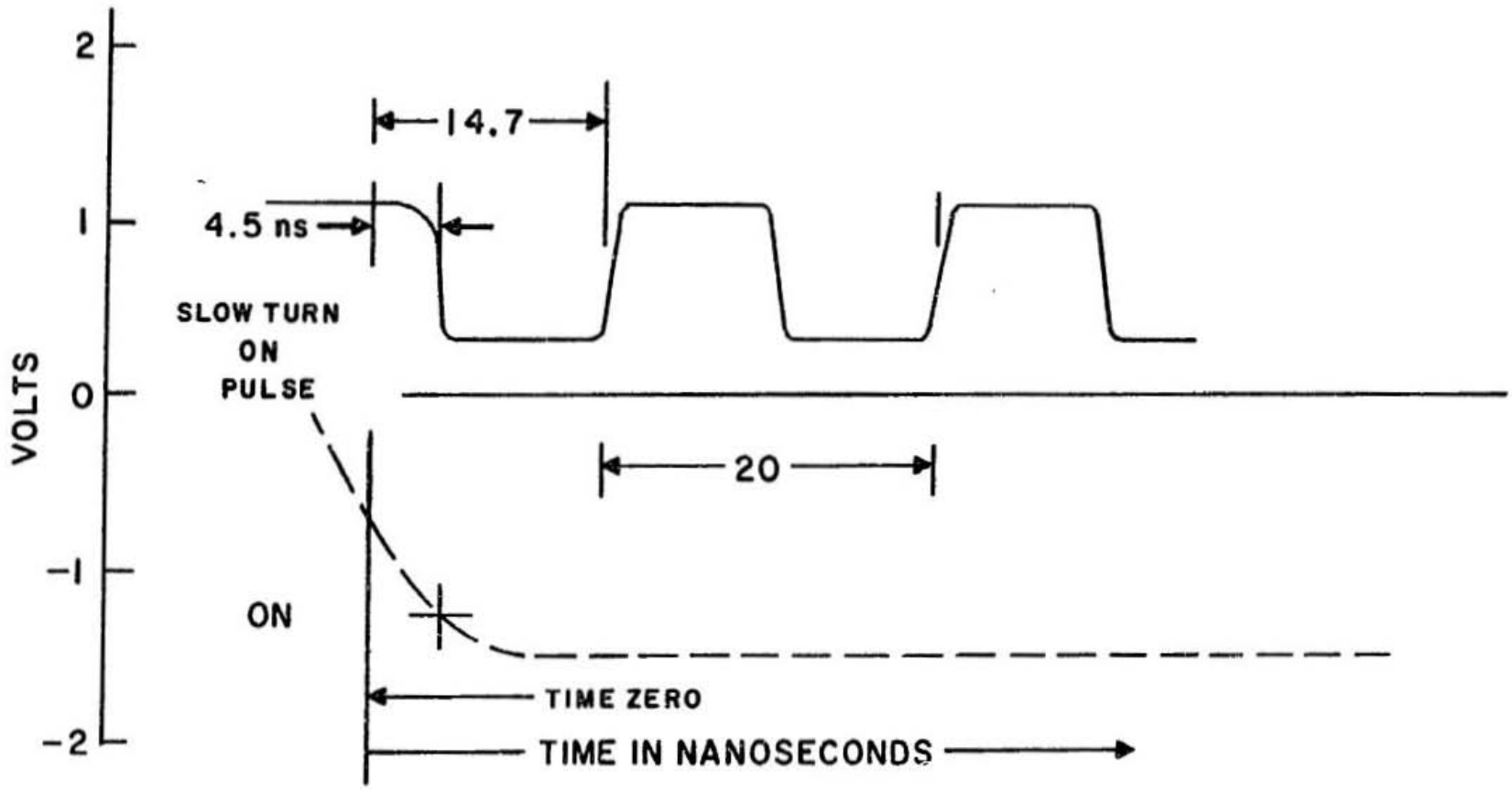


Fig. 8.2

PULSE TIME RELATIONSHIP FOR THE GATED 50 Mhz OSCILLATOR

with the MC10616 because each amplifier in this chip has a 1.8 ns propagation delay instead of 2 ns for the chip used. Since the terminals of these chips are identical, one can be substituted for the other in the prototype circuit. But equipment was not available to carry testing any further.

As shown in Fig. 8.2, the first pulse must be quite regular and it should approach a square wave in shape. Also all the following pulses appeared at 20 ns intervals thereafter.

The turn off time of the oscillator is not of importance, because the gating pulse which turns it off also disables all the counters at the same time.

As a favour to me, the Physics Department of The National Research Council carried out further tests on the prototype. Their report shows that they found the oscillator to be vary stable as to its frequency. Further, the turn-on and turn-off delays of the gated oscillator were very stable, being independent of:

1. the frequency of the gating pulse, which varied from 500 Hz to 5 Mhz during their testing, and

2. the length of time the power was applied to the oscillator before the gating pulse came, the test time ranging from 10 seconds to one hour during their testing.

Photographs a, b, and c given in Fig. 8.3 are NRC's test oscilloscope traces (12) for the oscillator. Photo (a) shows the response to a repeated gating pulse, while photo (b)

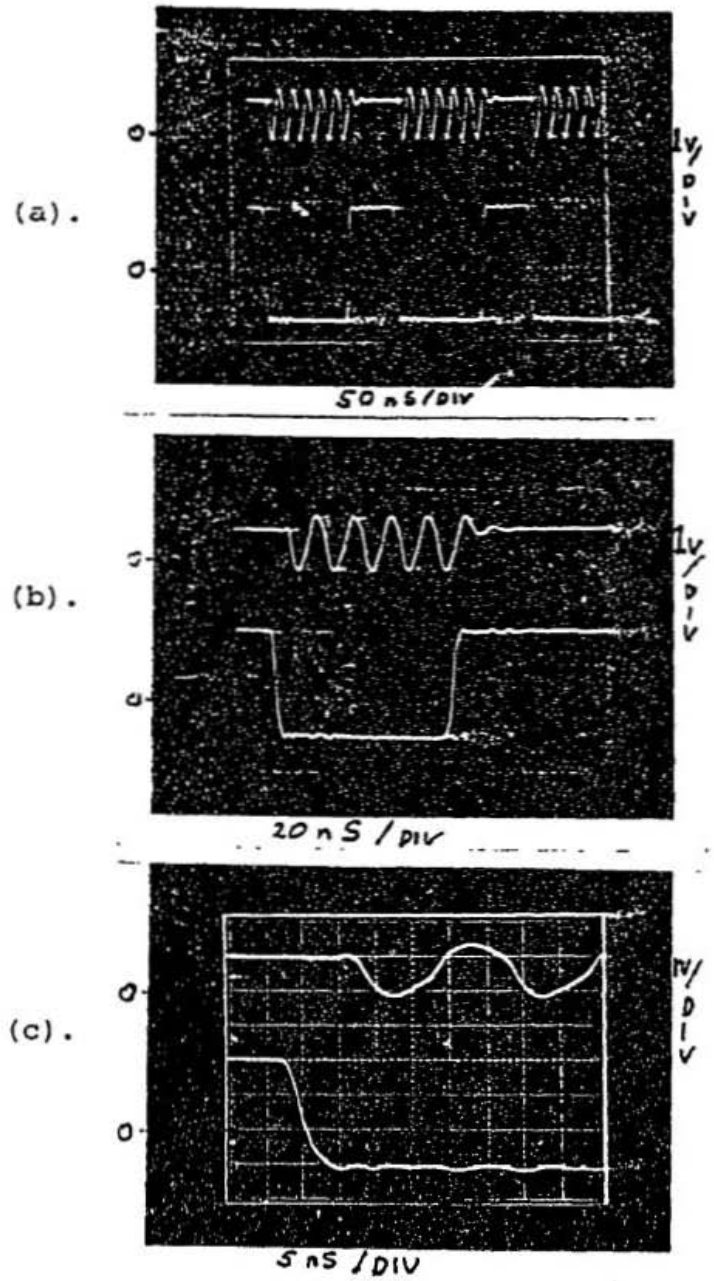


Fig. 8.3 NRC's TEST OSCILLOSCOPE TRACES

shows the oscillator's output relative to one gating pulse (frequency of gating pulse is 5 Mhz).

A further inspection of photographs (a) and (b) indicates that the gated oscillator puts out a rising edge after the voltage level of the gating pulse becomes positive with respect to the turn-on voltage level of the oscillator. This rising edge of the oscillator's output, although undesireable, causes no problems because the gating pulse employed in the system also disables the counters. Thus the counters do not see this last oscillator rising edge.

Photo (c) shows the oscillator's output when employing a gating pulse with a 5 ns/Div sweep speed. The start-up delay can be measured from this photograph, and this delay was shown to depend upon the trigger voltage level of amplifier 3, shown in Fig. 8.1, which indicates that the turn-on time is somewhat adjustable. For example, from Fig. 8.3. (c) we obtain a start-up delay time of 4.5 nanoseconds. The oscillator pulse shape appears to be distorted in Fig. 8.3 (c). This distortion is due to the scope that was employed by N.R.C. (12).

We investigated the oscillator's period as a function of the control voltage. Table 8.1 indicates how the frequency varied with variation of the varactor control voltage above and below +1.0 volts. Fig. 8.4 indicates that the shift is not linear but it is symmetrical about this

TABLE 8.1 PERIOD VERSUS VARACTOR CONTROL VOLTAGE

PERIOD	CONTROL VOLTAGE
ns	Volts
20.8	0.5
20.4	1.0
19.5	1.5
19.2	2.0
19.0	2.5
18.9	3.0

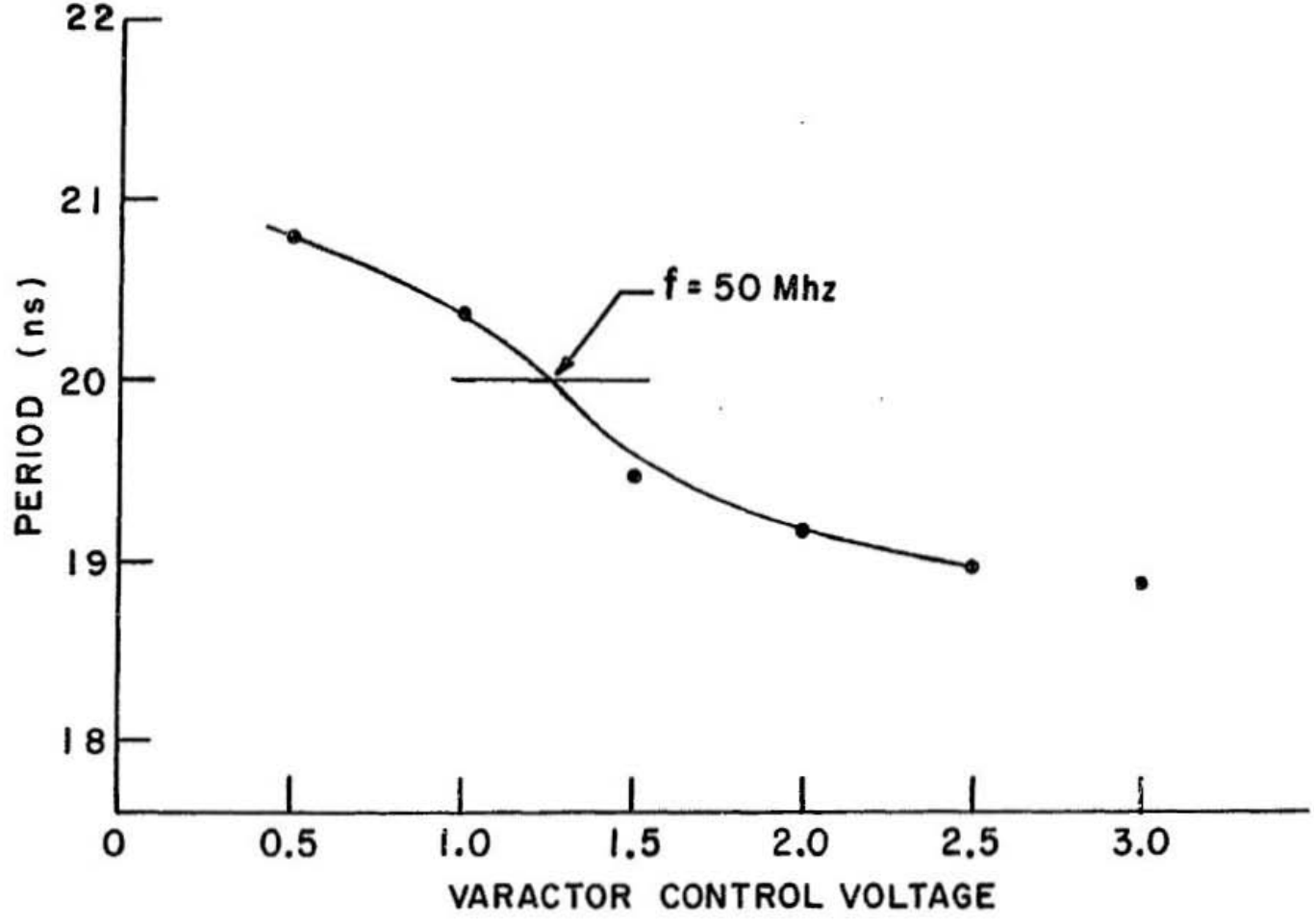


Fig. 8.4 FREQUENCY VERSUS VARACTOR CONTROL VOLTAGE

voltage in both directions. The maximum variation in the period is 0.8 ns around the 20 ns point. This is excellent because it also indicates that a slight variation in the varactor control voltage will not permit the oscillator's frequency to depart suddenly from the median frequency, i.e. - that is, run away. Special circuitry for maintaining the varactor voltage at a constant controlled value will be shown in Fig. 10.1 and discussed there. Hence the variation in the frequency will be eliminated.

The unit was mounted on a double sided copper printed circuit board designed by me, the underside acting as a ground plane.

Note that Fig. 8.4 indicates that the tuning resistor value used (see Fig. 8.1) for the 50 Mhz oscillator will have to be adjusted to give the vernier frequency of $45.\dot{4}\dot{5}$ Mhz, since the setting that was used to produce the data for the 50 Mhz oscillator (see Fig. 8.4) will not allow one to obtain the 22 ns period. To achieve both settings, the trimmer or variable resistor should be at least a 10,000 ohm variable resistor.

The output wave train of the oscillator circuit shown in Fig. 8.2 must be further adapted before it can be used by the various circuits in which it will be employed.

Consider Fig. 8.5. This figure is the schematic diagram that shows the circuitry required to adapt the wave train for the 'Hundred' and 'Phasor' clocks. Note

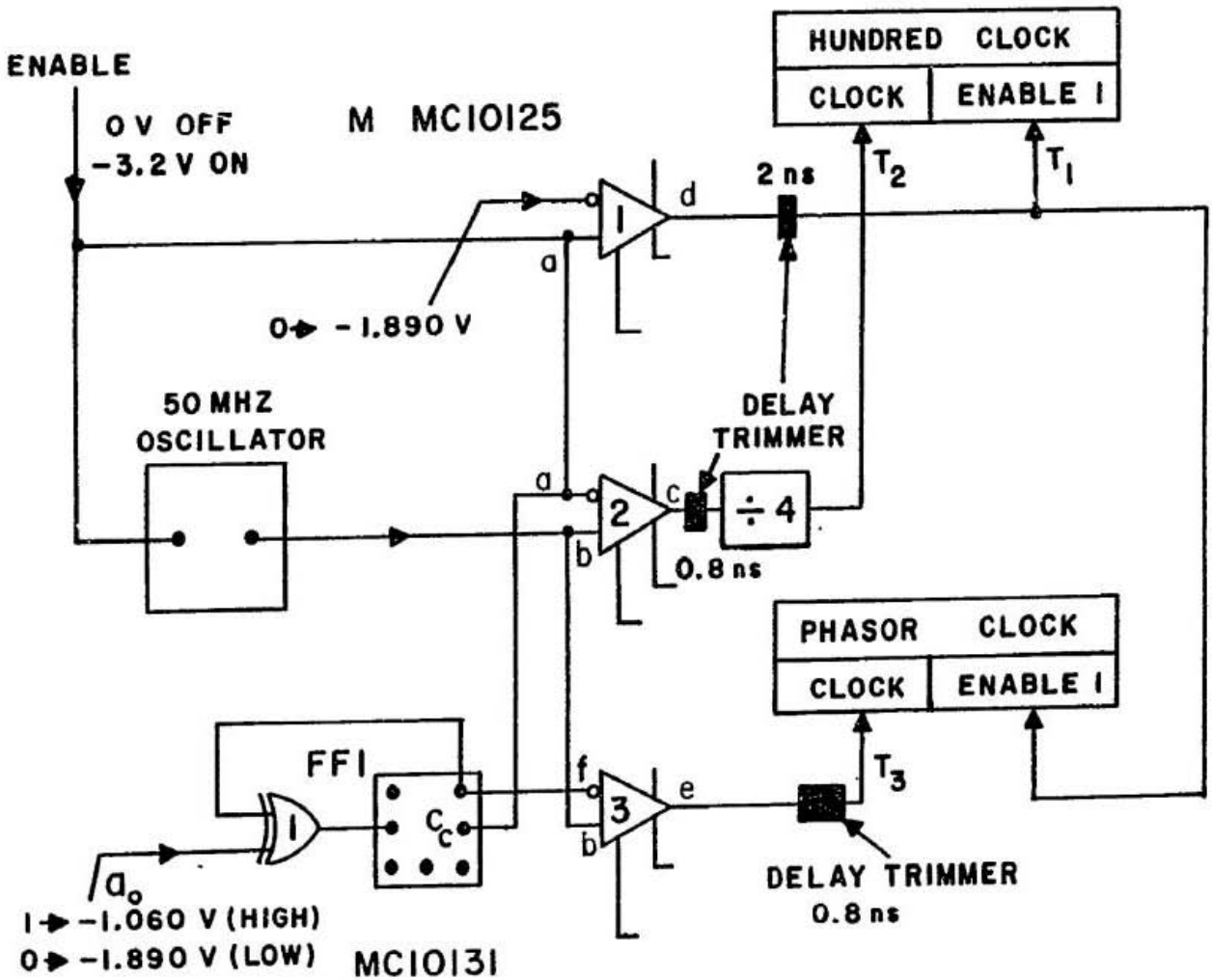


Fig. 8.5 50 MHZ SLAVE OSCILLATOR ADAPTATION CIRCUIT

that terminals T_1 and T_2 go to the 'Hundred' and the 'Five' clocks respectively in both the Active division and the Automatic time interval evaluation division. Note that the 'Five' clock is really the 'divide by 4' counter.

Terminals T_1 and T_3 go to the 'Phasor' clock in the modules of the Active division.

Note that delay lines are indicated just before the output terminals T_2 and T_3. These delay lines can be used to trim the first period to exactly 20 nanoseconds. If the MC10616P chip is employed, this trimming may amount to as much as 1.5 ns. Then a digital delay line could be used to insert a 1 ns delay, and a copper line trimmed to insert the rest of the required delay to give the true 20 ns period for the rise of the leading edge of the first pulse.

Fig. 8.6 explains the need for the circuitry shown in Fig. 8.5. First note that the vertical broken line near the left side of the diagram indicates a zero of time. This line tells us where time zero is located. Consider the following:

Line 1. This is the actual voltage wave form of the output of the oscillator both before and after it has been enabled. Note that to the left of our zero time line, this level is high, and this high level persists in being high for 4.7 ns after the circuit has been enabled. This high level during the 4.7 ns period must be removed or we will

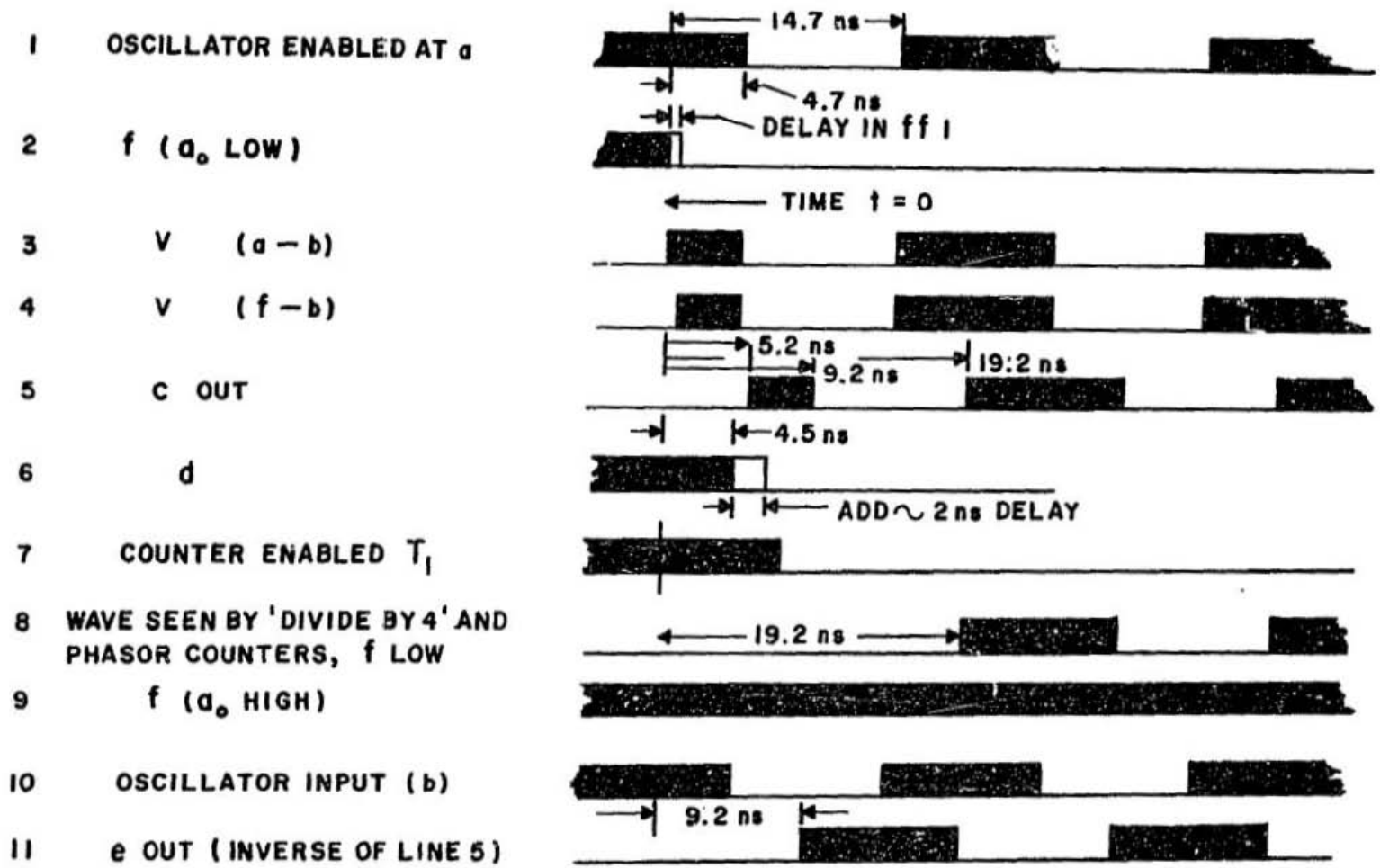


Fig. 8.6 ADAPTATION OF OSCILLATOR OUTPUT WAVE FOR HUNDRED AND PHASOR CLOCKS

not be able to load our data into our counters because their clock levels are high. Remember that a short clocking pulse is required to load the data into the Q data storage areas of each clock. This is our 50 Mhz wave train.

Line 2. Terminal f of translator 3 is held high, but goes low with a_o since f must assume the same level as a_o. Note the delay due to the reaction time of flip-flop 1. The voltage level of line 1 appears at input b of translator 2, while the voltage levels of lines 1 and 2 appear at inputs b and f respectively of translator 3.

Lines 3 & 4. These lines show the voltage differentials between the levels in lines 1 and 2 as seen by translators 2 and 3 at their input terminals. Note that we now have a narrow high voltage level just after the circuit has been enabled. We can not use lines 3 and 4 for the oscillator's output because of this narrow high voltage level.

Line 5. This is the same wave train as for line 4 coming out of terminal c of translator 2. Note that the whole train is delayed by 4.5 ns, which is the delay in the propagation of the input differential voltage through the translator. We still can not use this as our output wave train since this first short pulse must still be eliminated.

Line 6. This is the enabling pulse that emerges from terminal d of translator 1. Note the delay of 4.5 ns. We need to add approximately 2 ns to this delay by means of a

trimmer delay line to ensure that the 'Hundred' clock does not see the first rising edge in line 5.

We need a delay of 0.7 ns to eliminate the reaction of the counter to the first rising edge on line 5. In addition we require a 0.8 ns delay trimmer to adjust the first output period to 20 ns following translators 2 and 3, as will be indicated below. Therefore a total delay of at least 1.0 ns is required, and 2 ns have been chosen to give a margin of error.

This means that if terminal d is connected to the terminal 'enable 1' of a counter, then that counter will appear to be enabled 6.5 ns after the external enabling pulse comes from an external circuit as shown in the top left of Fig. 8.5.

Line 7. This voltage line tells us when the counter is enabled. Note that the falling edge here comes after the falling edge in line 6. This is excellent, because the counter will now be enabled after the rising edge of the narrow high level pulse of line 5. Thus the counter will not see this rising edge and it will not count this edge.

Line 8. This is the wave train as now seen by the counter. Note that the first rising edge now comes 19.2 ns after the zero time of the external enabling edge. We can insert a trimmer delay line and add 0.8 ns delay to give a true 20 ns period for the first period. Thus we have our normal 50 Mhz wave train in the proper form.

Now the 'Phasor' can operate on this wave train or on the inverse of this train. But it will require more than just the inversion. In addition to the inversion, it requires an extra pulse to occur 10 ns before the first rising edge of the normal wave train.

Consider the following:

Line 9. This line shows that f remains high since a_o is high. We wish the voltage level at f to be decided by the input data voltage a_o. Note that this voltage a_o appears at one input of EXCLUSIVE-OR gate 1. Using this gate, we can cause the Q output state of flip-flop 1 to assume the same voltage level as at a_o. But we want the Q output voltage level to remain high until after the oscillator has been enabled, and then let it follow a_o. We can do this by tying the common clock input of flip-flop 1 to terminal a of translator 1. Note, Q has its voltage level preset high automatically before the enabling of the oscillator.

Now if a_o has a low voltage level, then as soon as the common clock input level goes to the low voltage level with the enabling low voltage level, the flip-flop will follow a_o in approximately 1 ns, and this change in voltage level will cause the output of translator 3 to be that of line 8, which is the normal wave train.

Line 11. But if a_o is at a high level, then the output will be the output of terminal e, which is now just the

inversion of the oscillator wave delayed by 4.5 ns by the translator. Note that the first rising edge of this inverted wave train rises at approximately 9.2 ns after the enabling of the oscillator, i.e 4.7 ns plus 4.5 ns. Thus a trimming delay line can be added to obtain the required 10 ns interval here.

In this manner either the normal or the inverted wave trains can be chosen for the 'Phasor'; the train chosen depending only upon the voltage level of a_o.

Now consider Fig. 8.7. This is the schematic diagram of the circuit which adapts the output of the vernier slave oscillator for the vernier clock. Here only the inverted output of the vernier slave oscillator is seen by the 'Vernier' clock. But we will require two different clock trains. We create the second train by passing the first train through a 1 ns delay line.

By using the inverted wave train which has its first rising edge after 4.7 ns, there is ample time (about 15.3 ns) to permit the insertion of a switching circuit formed by AND gates 1 and 2 and OR gate 1. The translator and the delay trimmer will bring the total delay up to 20 ns. The EXCLUSIVE-OR and flip-flop FF1, together with this switching circuit, make it possible to choose which one of the two wave trains will be inputted into MC10125, which is translator 2. The choice is decided by the voltage level of the input terminal of the EXCLUSIVE-OR gate which depends

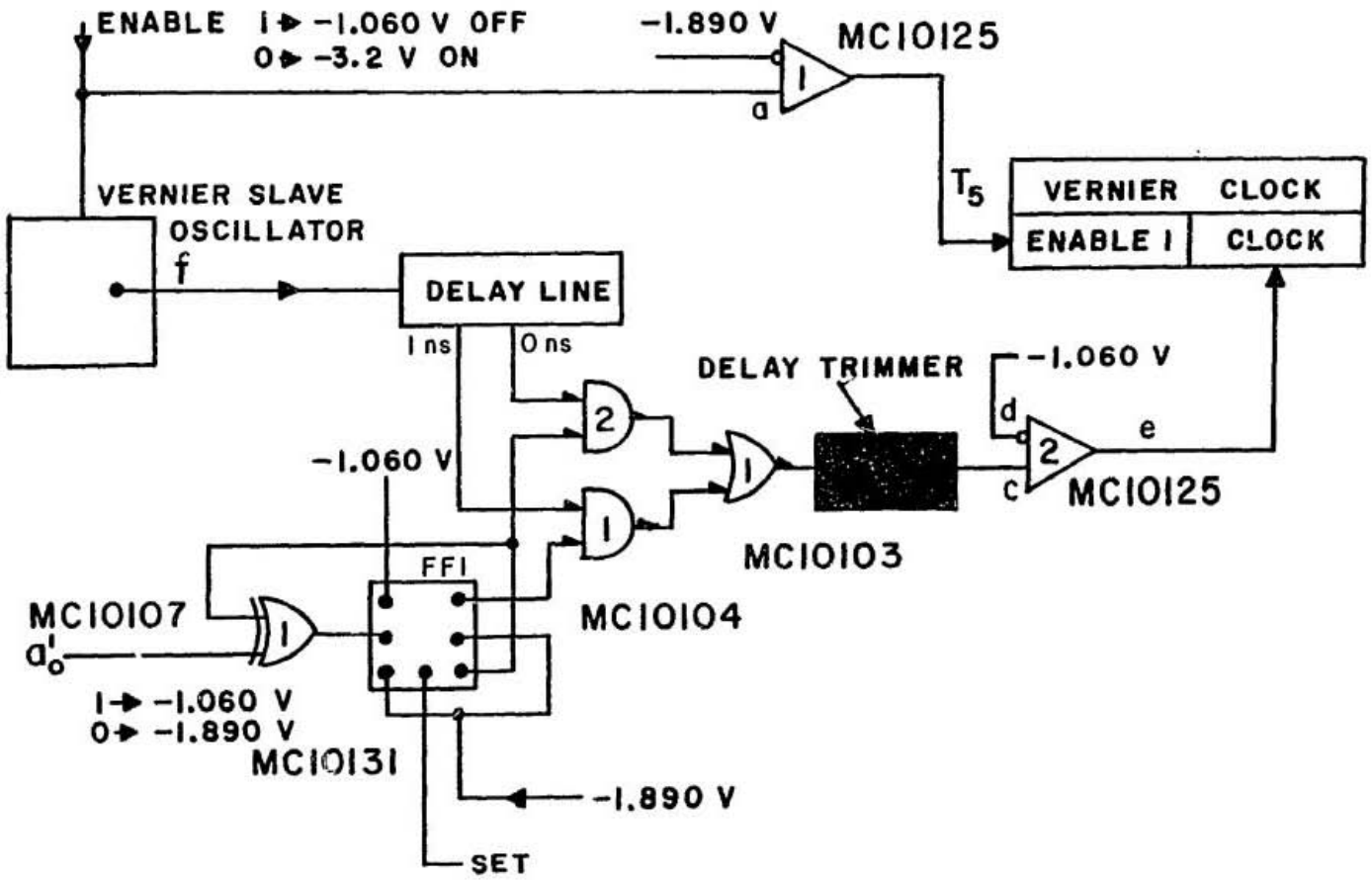


Fig. 8.7 ADAPTATION CIRCUIT FOR VERNIER SLAVE OSCILLATOR

upon the voltage level of a_o'.

Fig. 8.8 shows how this circuit is able to present the desired wave train to the 'Vernier' clock.

Line 1. Here we have the external enabling voltage level from the 'Phasor' at the input of the vernier slave oscillator and also at terminal a of translator 1 at tir zero.

Line 2. This line depicts the normal pulse pattern leaving the oscillator, commencing before time zero.

Line 3. Here we see the first falling edge entering terminal c of translator 2. The wave has passed through the delay line and the switching circuit, and thus has been delayed an additional 10.7 ns.

Line 4. The steady high voltage level of terminal d of translator 2.

Line 5. This is the voltage differential $\triangle V$ between terminals c and d of translator 2 which is delayed by the sum of the delays of the vernier slave oscillator and of the switching circuit.

Line 6. Here, 19.9 ns after the enabling voltage level occured, $\triangle V$ emerges from terminal e of translator 2.

Line 7. This shows the same wave train leaving translator 2, but delayed by 1 ns if the train has been delayed by the optional 1 ns delay as determined by gate 1 and flip-flop FF1.

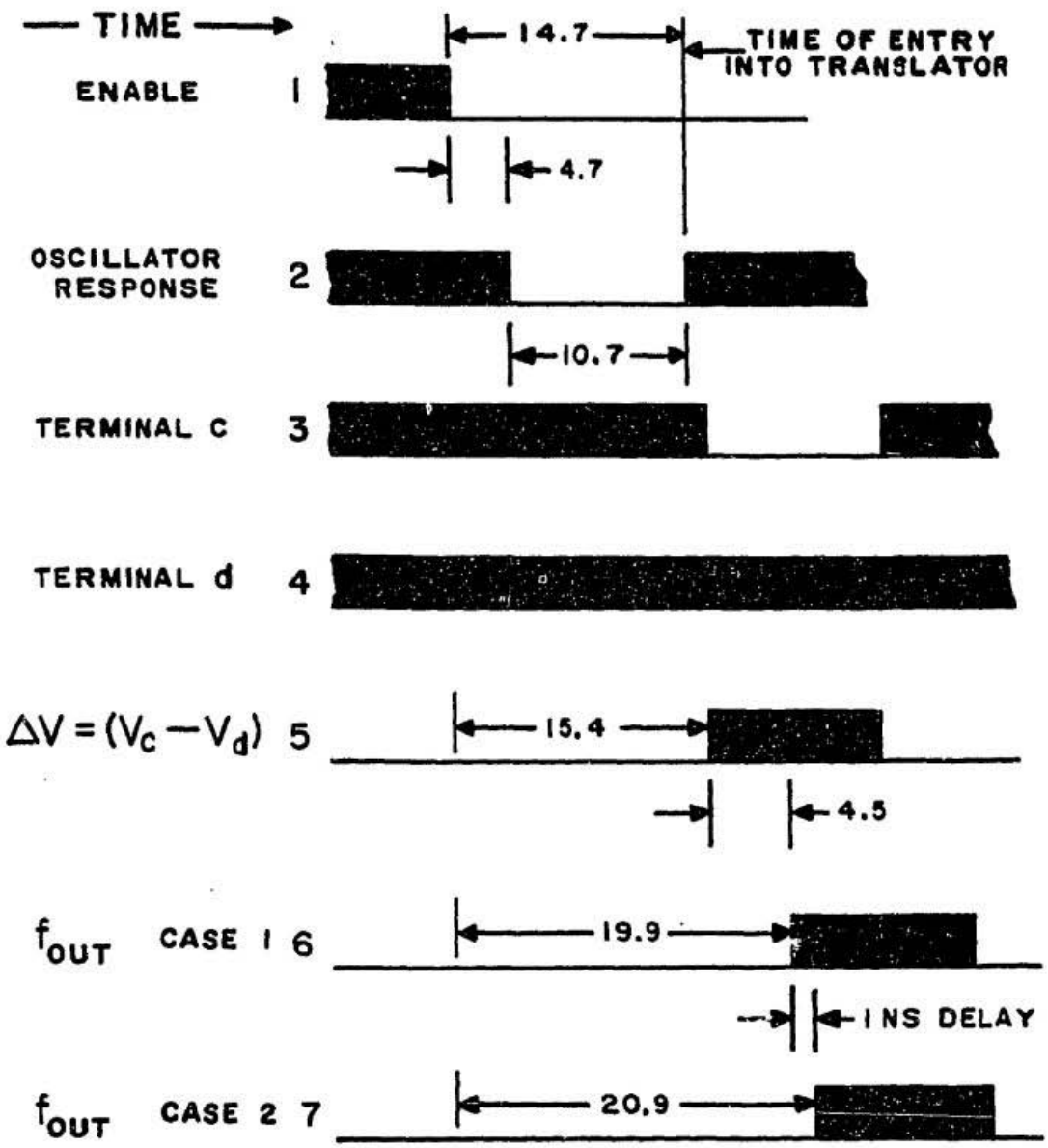


Fig. 8.8

PULSE TIME RELATIONSHIP
FOR THE VERNIER CLOCK

As shown in Fig. 8.7, a trimmer delay line may be inserted after translator 2 in order to adjust the period to 20 ns (or 21 ns if the 1 ns delay option is chosen).

Recall that although the vernier oscillator has a period of 22 ns, the time of rise of the first leading edge is required at 20 ns after the occurance of the enabling voltage level for the normal wave train.

CHAPTER 9

THE ACTIVE DIVISION: HIGH SPEED CLOCKING

Block diagram, Fig. 9.1 depicts the arrangement of components that make up the circuitry of the Active division, and also shows the paths taken by the various pulses. Fig. 9.2 is the schematic diagram for this division. Also note that Fig. 9.1 shows the 50 Mhz and the vernier slave oscillators, which are located in each module.

By means of Fig. 9.1, one can see that when the trigger pulse t_o comes, it enables the 50 Mhz oscillator by means of control C_1, which then puts out frequency f_o. C_2 is the control for the 'divide by 4' circuit. C_2 was set by the Passive division in order to give every fifth pulse to the 'Hundred' clock - that is if the shadow 'Hundred' clock's data did not become zero during the test by the 'Q clock'. If that Q data is not zero, the 'Hundred' clock will count, but if it should be zero, this clock will be idle. When this clock is idle, C_2 causes the passage of the undivided oscillator frequency to the 'Phasor' clock. Control C_3 is the flip-flop which will disable the 'Hundred' clock when its Q data becomes zero.

C_5 is the control, set by Q_5 of the Passive division, which decides whether the 'Zero' clock will remain idle or count down 5 pulses concurrently with the 'Phasor' in order, in this case, to establish a reference time zero

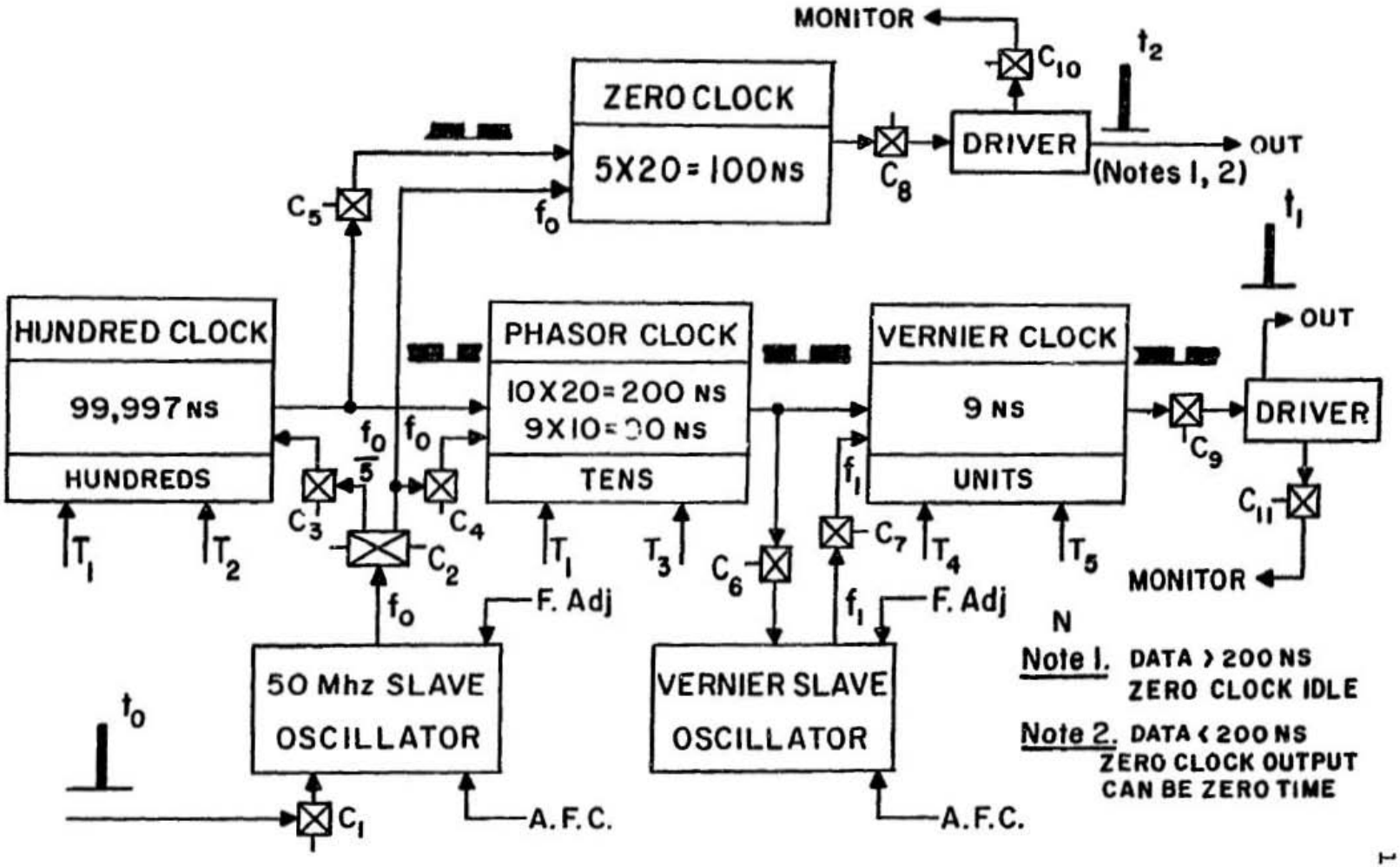


Fig. 9.1 BLOCK DIAGRAM, ACTIVE DIVISION MODULE (HIGH SPEED)

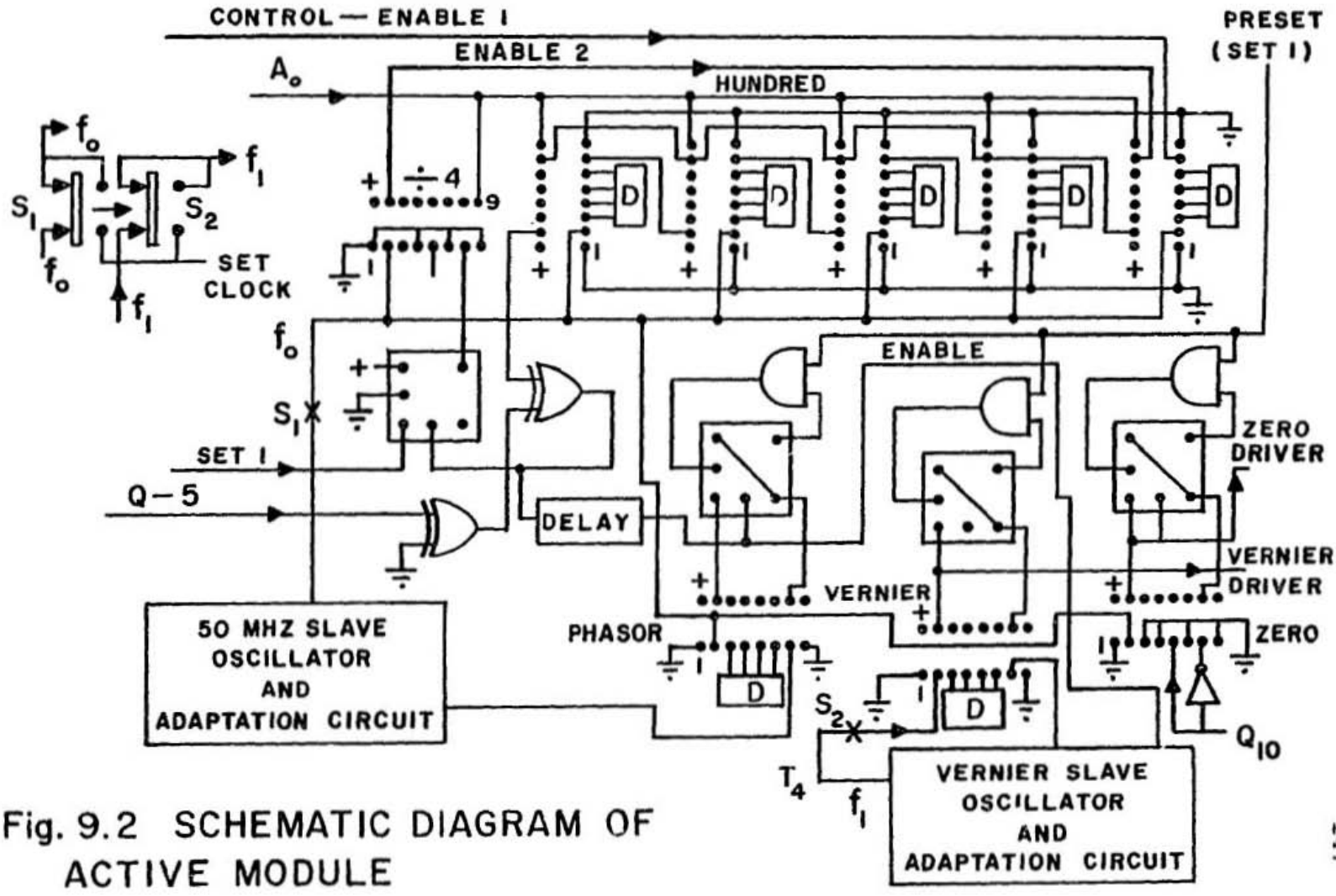


Fig. 9.2 SCHEMATIC DIAGRAM OF ACTIVE MODULE

for the 'Vernier' when time intervals shorter than 100 ns are required. In this case the time interval between this 'Zero' clock output pulse and the 'Vernier' output pulse is given by $t_1 - t_2$. Control C_8 simply disables the 'Zero' clock when its Q data becomes zero.

Controls C_4 and C_7 likewise disable the 'Phasor' and 'Vernier' clocks respectively when their Q data become zero. Control C_6 enables the vernier oscillator when the 'Phasor' clock is disabled. When the vernier Q data states become zero, an output pulse occurs via the associated driver at time t_1 giving a time interval, $t_1 - t_o$.

Controls C_{10} and C_{11} allow the outputs of the 'Zero' and 'Vernier' clocks to be monitored by the Control division, so that 'Control' may reload all Q data areas automatically, and set each alerted module ready for another time interval run, when monitoring signals have been received from the alerted modules by 'Control'.

As shown in Fig. 9.1, the 'Hundred' clock will put out a ripple carry pulse to control the activation of both the 'Zero' and 'Phasor' clocks, as conditions require. The 'Phasor' clock will put out a similar control pulse to initiate the 'Vernier' clock and the vernier slave oscillator. Finally, the 'Zero' and 'Vernier' clocks will put out similar control pulses to their respective drivers.

The two slave oscillators each have a frequency adjustment so that each frequency may be adjusted up or

down slightly about a central or median frequency.

By means of the two controls, the automatic frequency control A.F.C. and the frequency adjustment F. Adj., one can set both oscillators to be exactly on their respective frequencies. The A.F.C. control comes from the Master frequency division and will be discussed in Chapter 10.

Since these oscillators turn on very quickly, it may be possible in some cases to adjust the delay lines associated with these oscillators so as to compensate for the time delays inherent in the output drivers employed for a specific piece of external equipment. In this manner, depending upon the driver used, one may be able to balance out the driver's time delay.

Note that the use of E.C.L. logic throughout this apparatus would eliminate switches S1 and S2 and all the translators in Fig. 9.2. In addition, the vernier used to measure the units of nanoseconds in the Interval evaluation division would run concurrently with clocks A and B of Fig. 4.1, so that when the termination pulse comes all time measurements (100^s, 10^s, units) would be determined concurrently and instantaneously (8).

Referring to Fig. 9.2, note the mechanical switches S1 and S2 shown in the upper left hand corner of this diagram. These switches were originally intended to be digital switches, which were to perform two functions.

1. to effect the loading of data into the Q data storage of the various clocks by passing the clocking pulse B_o from the set clock during the loading cycle, and

2. to permit the passage of clock pulses to the above clocks during the active clocking cycle.

Since digital switches are permanently a part of the circuit in which they are employed, their reaction or delay times are added to the total existing delay time of that circuit, and their delay times cannot be removed. But there is no room left timewise to accomodate this additional delay inserted by the use of digital switches. If one switches entirely to E.C.L. circuitry, eliminating the translators and their 4.5 ns delays, then digital switches can be used for S1 and S2.

CHAPTER 10

MASTER FREQUENCY CONTROL

The heart of the system is a precision crystal controlled oscillator operating at 50 Mhz with a temperature stability of 5×10^{-7} percent and a change in frequency with time given by 2×10^{-6} hz/year (1×10^{-8}hz/day average), and frequency adjustment range sufficient to compensate for 5-10 years of crystal aging: setable to 1×10^{-7}.

Two master oscillators are phase locked to this oscillator, both of which are running continuously. One master operates at 50 Mhz giving a period of 20 ns while the second one operates at $45.\dot{4}\dot{5}$ Mhz for a period of 22 ns.

The voltage outputs of the low pass filters of these two phase locked loops are used respectively to vary the reverse voltages applied to the varactors substituted in place of the usual timing capacitors. In this manner, smooth fine frequency tuning is obtained by each master, together with stable frequencies.

But in addition to accurate frequencies, we obtain relatively steady voltages due to the varactors of the master oscillators, and these voltages are used to control the varactors in the 50 Mhz and vernier slave oscillators that start up or turn off for the actual time delay or time interval determination.

Fig. 10.1 is a block diagram of this control system. The frequency control, and the range control for each slave oscillator is located on each oscillator board, and may be used jointly to vary the frequency, so as to set each oscillator to its proper operating frequency value, while its varactor is under the influence of the control voltage put out by its master control oscillator.

The foregoing gives a very sensitive system, which, although it adjusts itself with sufficient speed, does not tend to overshoot the equilibrium frequency value. This is because the varactor reverse voltage must follow the voltage given out by the respective master control oscillator.

If the master control oscillators are located adjacent to the active and time interval evaluation modules respectively, then deviations from the required frequencies by the slave oscillators due to temperature variations can be minimized, resulting in a steady system.

So far we have discussed the normal phase-locked system, shown in the lower half of Fig. 10.1. For such a phase-locked system, there is a small frequency variation or equivalently a small voltage variation. The normal variation in frequency would be much less than that shown in Fig. 8.4, since the allowable voltage variation for the varactor must be less than $\pm$ 0.1 volts around its desired operating voltage.

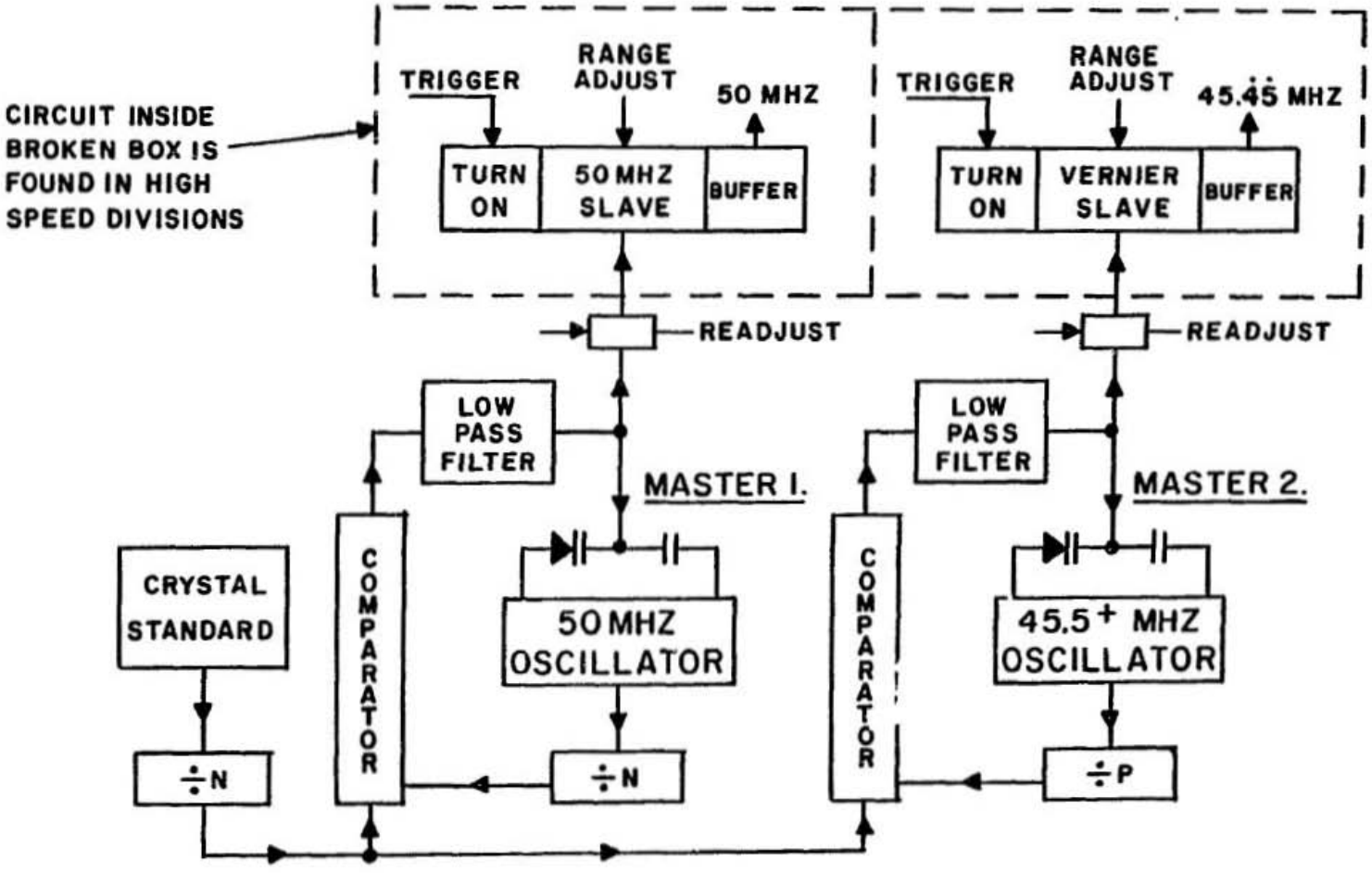


Fig. 10.1 PRECISION HIGH FREQUENCY VOLTAGE CONTROLLED AND CRYSTAL CONTROLLED OSCILLATORS

The term 'jitter' applies to the periodic variation of the frequency above and below the synthesized output frequency of a phase-locked loop. This variation occurs, because in a phase-locked loop, there has to be some phase difference between the edge of the reference pulse and the edge of the respective oscillator pulse at the inputs of the phase detector in order to establish the error voltage necessary to keep the oscillator operating close to the desired frequency. This is a dynamic system and it must act in the above manner.

In reference 11, page 97 it states:

"- as the microprocessor keeps track of all the various division ratios, it becomes apparent that the resolution for the direct synthesizer in the 2 to 3 Mhz range, has to be better than 1 Hz in order to get 0.1 Hz resolution up to 100 Mhz at the final output frequency, and 1 Hz from 100 to 1300 Mhz."

However even this small jitter can be a problem in high precision timing. Thus as shown in Fig. 10.1, the output voltage from the low pass filter of the crystal controlled phase-locked loop is supplied to a circuit named 'readjust'. This circuit contains a differential voltage compensating circuit which removes the unwanted voltage variations, and leaves a steady mean voltage level to tune the diode of each slave oscillator. There is no phase detector in the slave oscillator circuits to affect their output frequencies. In this manner, all jitter is removed from the outputs of the slave oscillators that do the actual timing. (11).

CHAPTER 11

CONTROL

This system allows the operator to choose which of the possible modes of operation will be implemented once the operator has made a selection. This division controls the complete time measuring system automatically.

11.1 Choosing the source of data input

Consider Fig. 11.1. Note that 'Control' offers only three choices when the operator chooses the data source:

1. control level one,
2. control level two, and
3. set.

Control level one is set either to 0 or +4 volts by means of the Manual-Auto time button. When the nameplate of this switch is lighted, the Automatic time interval evaluation mode is activated and the output is +4 volts; otherwise the voltage output is 0 volts and the Manual insertion mode by means of the digital switches is operative.

In like manner, the switch bearing the title - SYNCH - the name standing for Automatic synchronization, will light when synchronization is desired and again the output will be +4 volts. If the nameplate is not lighted, we have 0 volts, and non synchronization. This is control level two.

These two controls make it possible to choose

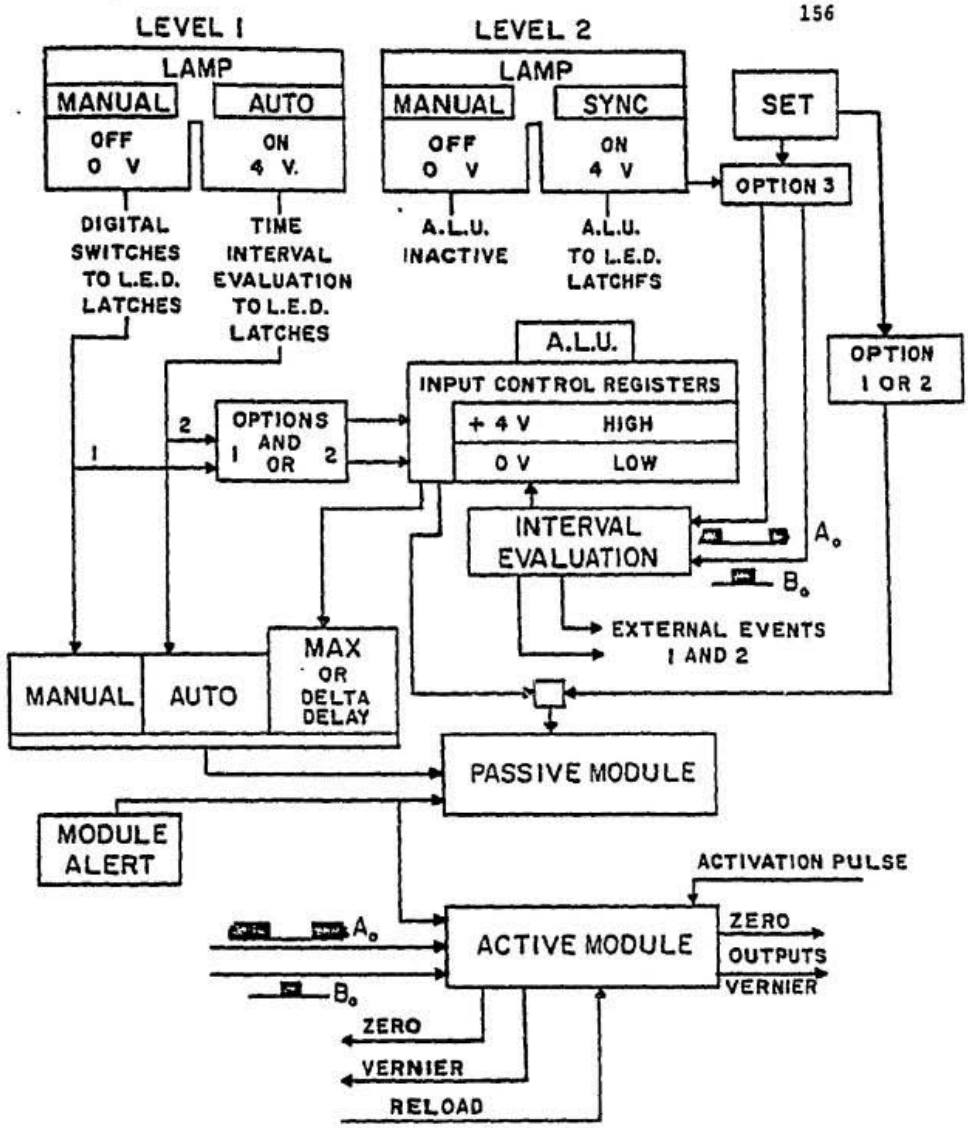


Fig. II.1 BLOCK DIAGRAM OF CONTROL

the input intervals that will:

1. synchronize external equipment, or
2. operate external equipment according to some predetermined schedule (not necessarily synchronized).

Synchronization may come about by using the Time interval evaluation mode, time intervals between external events which it measures, or the operator may arrange for synchronization using manually inserted data that has been predetermined in some manner.

The operator has multiple choices of how to inspect the data entering the Passive division. He can cause data from any of the three input sources to be displayed on the L.E.D. readouts:

1. manual switches,
2. Time interval evaluation, or
3. the A.L.U. output.

If the A.L.U. has been employed and it has calculated the maximum and the difference intervals, then the operator can switch the output display to any of the three inputs and not lose the data output of the A.L.U. division.

The operator will first set the alert buttons (not shown) to engage modules and then choose the desired inputs by setting the above mentioned switches. He will then activate the 'set' button. This button must not be set until data has been selected, because data must appear at the inputs of the various clocks of the passive modules

before the set pulse 1 arrives. After the set button has been activated, the system is automatic.

Fig. 11.1 also shows that if the SYNC mode is preset, option 3, then the set pulse from the set button goes to either the Interval evaluation module or to the A.L.U. The A.L.U. will put out a continuation pulse to start the passive modules when it has completed the arithmetic logic operation.

If the non synchronization mode is chosen, then the set pulse passes via option 1 or option 2 and starts the passive modules.

11.2 Preparing the data

As soon as the continuation pulse from the A.L.U. or the set pulse enters the passive modules, the modules will begin to receive pulses from the set clock in 'Control', because either of these two pulses also activates that clock as they enter the passive modules. This set clock will now run until the system is completely prepared.

Consider Fig. 11.2. A passive module will send to 'Control' one of three possible trigger alert pulses to tell 'Control' that it is prepared.

11.3 Loading active modules

When all alerted modules have carried out the above step, 'Control' will activate its data loading circuit and load data appearing at the inputs of the active clocks of each module into their Q data storage areas. Then

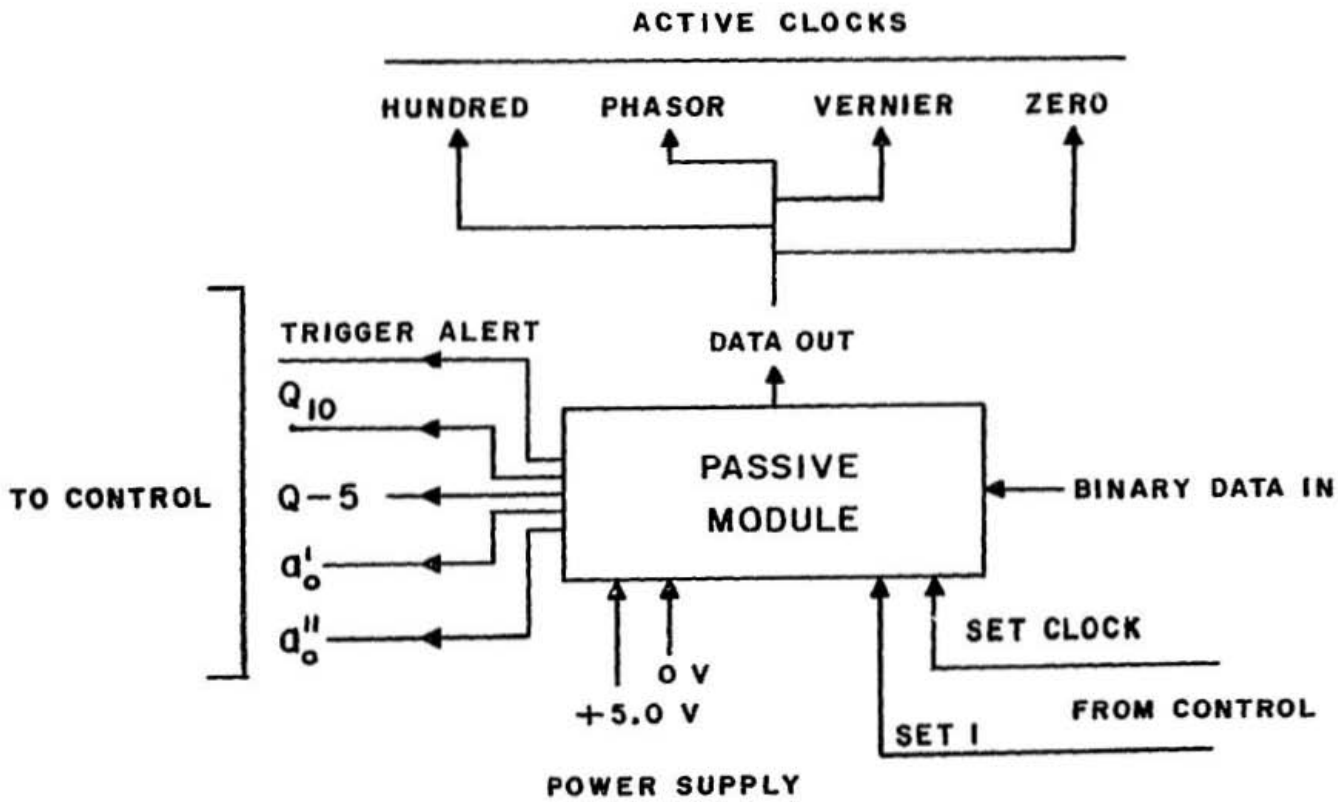


Fig. II.2 PASSIVE MODULE, TERMINAL INPUTS AND OUTPUTS

'Control' becomes inactive until the next prompt for action is received from one of the modules or from the operator.

'Control' also receives data such as the states of Q, Q-5 setting, a_o' and a_o'' for each passive module and transfers it to the respective active module at the proper time.

Consider Fig. 11.3. When each alerted active module has received the above data from 'Control', it remains inactive until it receives the external enabling pulse. The module will then count down the required time interval and put out a low voltage level pulse to its output driver or drivers. Each module has two outputs - the vernier and zero driver outputs. Each of these outputs can send a pulse to 'Control'. 'Control' knows whether to look for one or two pulses from any module according to the data that the operator has entered into the system.

'Control' will carry out these latter steps repeatedly until the operator changes the input data.

When the operator wishes to find the time interval between two external events, he will first activate the Auto button and then activate the set button. Auto stands for Automatic evaluation. Then 'Control' will activate its data loading circuit and will clear all clocks and flip-flops in the Time interval evaluation division modules. After each module has completed the determination of its time interval, it will send a pulse to 'Control'.

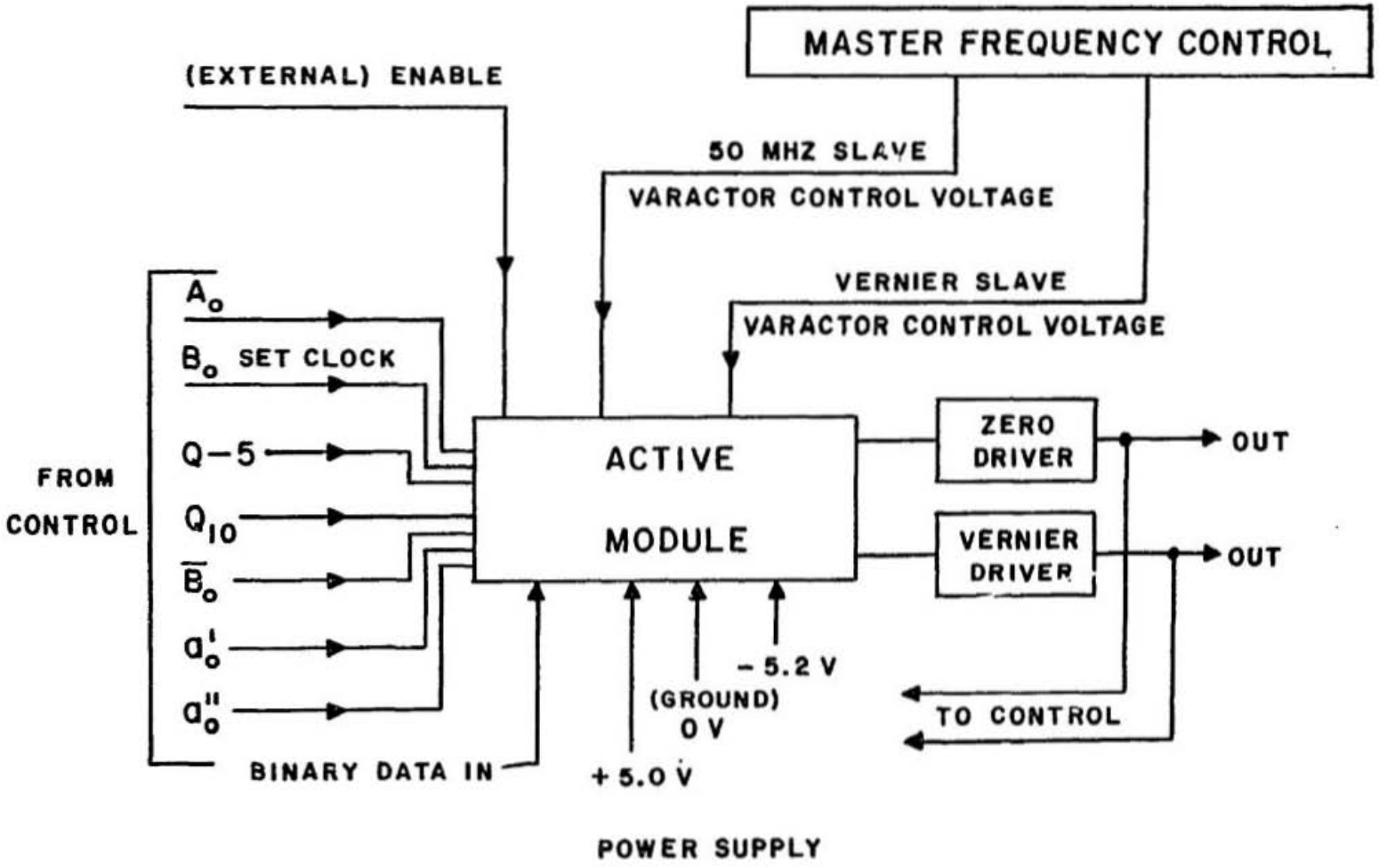


Fig. 11.3 ACTIVE MODULE, TERMINAL INPUT AND OUTPUTS

Then 'Control' will send a latch pulse to all alerted modules so that the determined time data will be locked into each output latch. But this pulse comes only after all alerted modules have sent their completion pulses to 'Control'.

Finally, if the operator wishes to cycle his output, he may use the output of one module as the trigger for the input of another module. This injection of an output signal into an input was originally to be handled by switches on the operator's panel, and this meant that the operator only had to activate the proper switches, and not use external coupling because of time delays always associated with such connections.

CONCLUSION

The nature of this project has been such as to require continual updating of the circuitry as the design progressed, in order to obtain the most efficient system as regards time, and also to remove from the system any aspect which might cause a malfunction of the system.

But a point in time was eventually reached when further improvement had to come to a halt because the rapid progress in the art of micro chips goes on ad finitum and time was of the essence.

Because of a malfunction of our fast scope, and delays in seeking external testing, the oscillator's parameters were not known until late in the project. Since we had to design for the worst possible results, TTL circuitry was used. Had we known that the oscillator would function as well as it was eventually shown to function, we would have used ECL logic. This would have resulted in a simplification of many of the circuits.

In addition, the detailed circuits for the 'control' division have been designed, but the details of their designs are not included here. In this case, only the block diagrams are given. Because the oscillator performed to specifications, the control circuits are being redesigned to take advantage of the performance of the oscillator. Consequently, since the present circuits will

be obsolete shortly, there is no advantage of including them here. Although these circuits are not included here, an electronics engineer can, after careful scrutiny of all fully described divisions, lay out the required circuits.

The circuits for each division have been laid out and tested at low speed to ensure that the circuits are logically correct. Also the various divisions have been interconnected and tested to confirm that interconnecting logic is correct. The A.L.U. division has been committed to printed circuit boards. Besides designing these circuit boards, the technical details of their actual production have been mastered by the author. Layouts for the other printed circuits have also been designed. However because of the oscillator test results, there are some layout modifications, and consequently the final printed circuit board designs have not been finalized.

Besides these tests that were conducted in our laboratory, N.R.C. tested the oscillator. Their results, that were discussed in Chapter 8, indicate that the high speed oscillator does function according to the design specifications.

In addition, N.R.C. Physics Division studied the circuits that have been designed. In Peter Grant's report (12), several points were made:

1. With regard to the vernier design, he states

" The design which you have submitted is a novel approach

to the best of my knowledge."

This statement has been verified by a computer based search of the literature.

2. With regard to the special oscillator and the time vernier, he states

"The principal elements of your system, that is the Special Oscillator and Time Vernier will, when properly constructed, permit generation of the time delays with $\pm$ 1 ns precision.

The $\pm$ 1 ns is a limitation due to your present clock rates and could be reduced. It is not a fundamental limit."

It should be noted that we have achieved 1 ns timing using a 50 Mhz clock which has a 20 ns period. We have a factor of 20 improvement using a vernier. At the same time, we have minimized the technical difficulties, that would be involved if one just increased the clock frequency. On the other hand if we did increase the clock frequency, one could in principle use a vernier again to increase the precision past that of the clock's period. Since the major and vernier frequencies can be freely chosen, and the Active division is a simple straight forward design, it is possible to approach 100 picosecond intervals using printed circuit boards. If the Active divisions were incorporated into a micro chip using the new ballistic-transport devices, operation at frequencies up to 200 GHz is possible. To date switching speeds of 20 picoseconds have been recorded and speeds as high as 10 picoseconds are expected this year. These devices

operate at room temperatures. These new submicron circuits are reported in the February edition of 'High Technology' (1983), Volume 3. No 2, page 62 under Perspectives, entitled: "Submicron circuits beat Josephson Chips". With the advent of these devices, it appears that periods as short as one picosecond might be attained. This being the case, this system has many applications in which it can form the important nucleus. Some examples are:

1. High speed detection, which will make it possible to obtain full knowledge as to the speed, direction of motion, as well as position with time of a moving object. This system can be augmented by a special computer that would predict future positions of an object or objects, using information gathered by the detection division. This makes possible the interception of moving objects and if need be, their destruction.

2. High speed measurement of distance which when coupled with a special computer will allow docking in space to be controlled automatically so that as one craft docked with another, its speed is reduced in proportion to the distance of their separation, making possible safe docking with no jarring or bumping of crafts.

3. Other adaptations occur in robotics, required in all branches of science and industry for the betterment of life on this planet.

Thus the size of time intervals detected or

created using the design principles underlying this system is limited only by the art of the micro chip technology.

(1). Paul Horowitz and Winfield Hill. The Art of Electronics, Cambridge University Press, New York, 1980.

(2) Designing with TTL Integrated Circuits, edited by Robert L. Morris and John R. Miller, McGraw-Hill Book Company, 1971.

(3). L.W. Anderson and W.W. Beeman. Electric Circuits and Modern Electronics, Holt, Rinehart and Winston Inc., New York, 1973.

(4). Jacob Millman. Microelectronics, Digital and Analog Circuits and Systems, McGraw-Hill Book Company, New York, 1979.

(5). U. Tietze and Ch. Schenk. Advanced Electronic Circuits, Springer-Verlag, Berlin, 1978.

(6). The TTL Data Book for Design Engineers, Second Edition. Texas Instruments. 1981.

(7). William R. Blood Jr. MECL System Design Handbook, Third Edition, Motorola Semiconductor Products Inc., 1980.

(8). Semiconductor Data Library, Technical Information MECL 10,000, MECL III, Phase-Locked Loop Components, Motorola Semiconductor Products Inc., 1974.

(9). Linear Data Book, National Semiconductor Corp., 1976.

(10). Delay Lines, Active Filters, Passive Filters, Data Delay Devices Inc., Clifton, N.J.

(11). Ubrich L. Rohde. Digital PLL Frequency Synthesizers, Theory and Design, Prentice-Hall, Inc., Englewood Cliffs, N.J. 1983.

(12). Private Communications, National Research Council, Physics Division, Ottawa.

APPENDIX A

This appendix contains reduced copies of the circuit patterns for the Arithmetic Logic Division which indicate the complexity of the circuit boards. Double sided boards are required.

The following comments apply to all circuits and are not specific to the A.L.U. Division circuits.

When committing the designs that have been given here to circuit boards for fast timing circuits, many problems arise such as ensuring that clock pulses arrive in phase at the inputs of different units. This requires the careful positioning of the units forming a circuit, so that path lengths may be minimized and sometimes equallized in length.

The width of and the separation between circuit lines are very critical as given in the literature for high speed circuits. Also the circuit boards are made especially for high frequency applications, and they have a uniform dielectric constant throughout the entire board. The theory for transmission lines is further invoked by using one side or part of a side of a board as a ground plane beneath the high frequency circuit lines in order to eliminate reflections that appear as ringing and false spikes that together will cause false counting. Furthermore, all boards must be etched cleanly, giving sharp lines

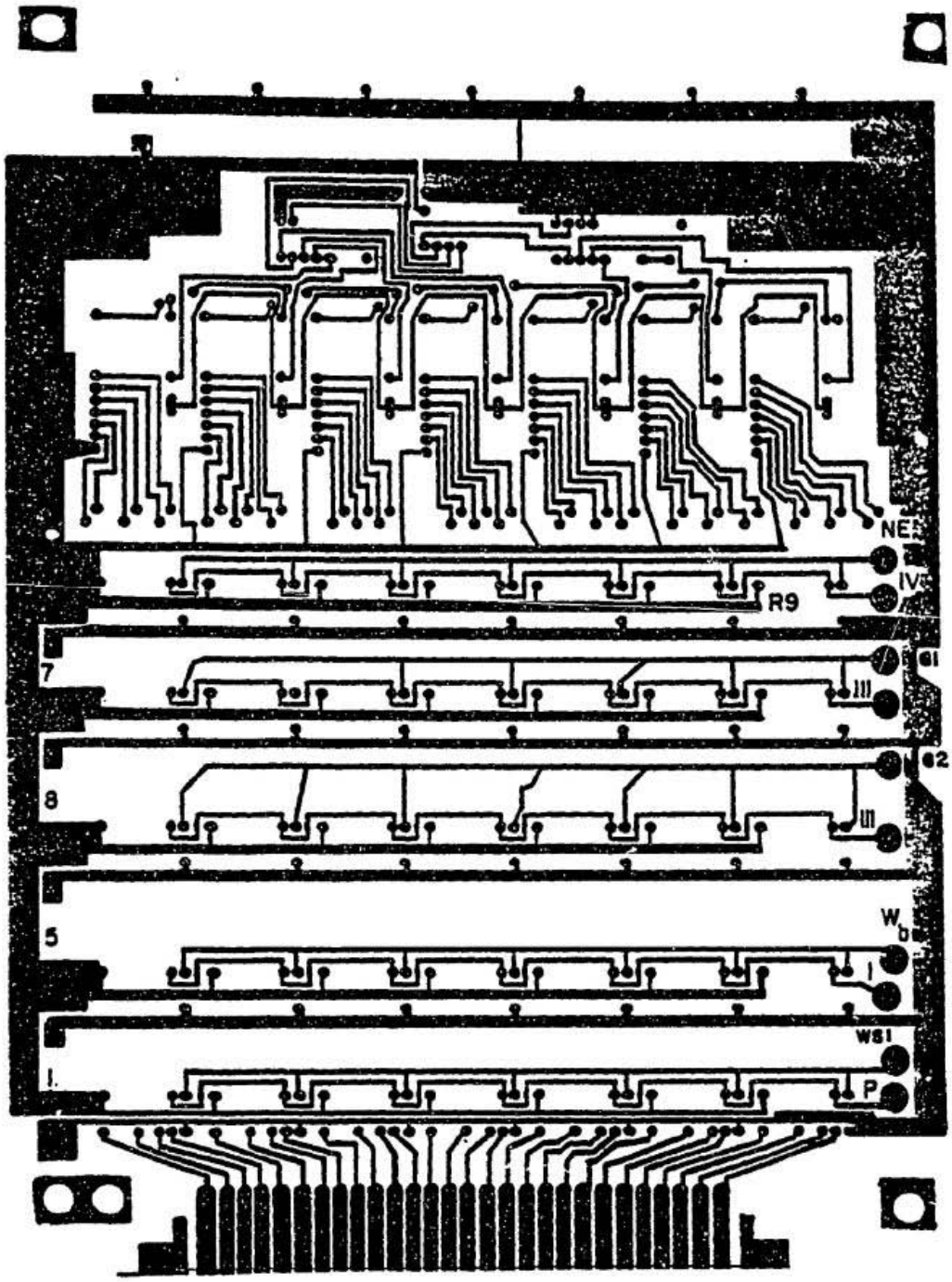


Fig. A1. UPPER SIDE, MAIN A.L.U. PANEL DRAWING

and no islands of residual copper to add capacitance to the circuits.

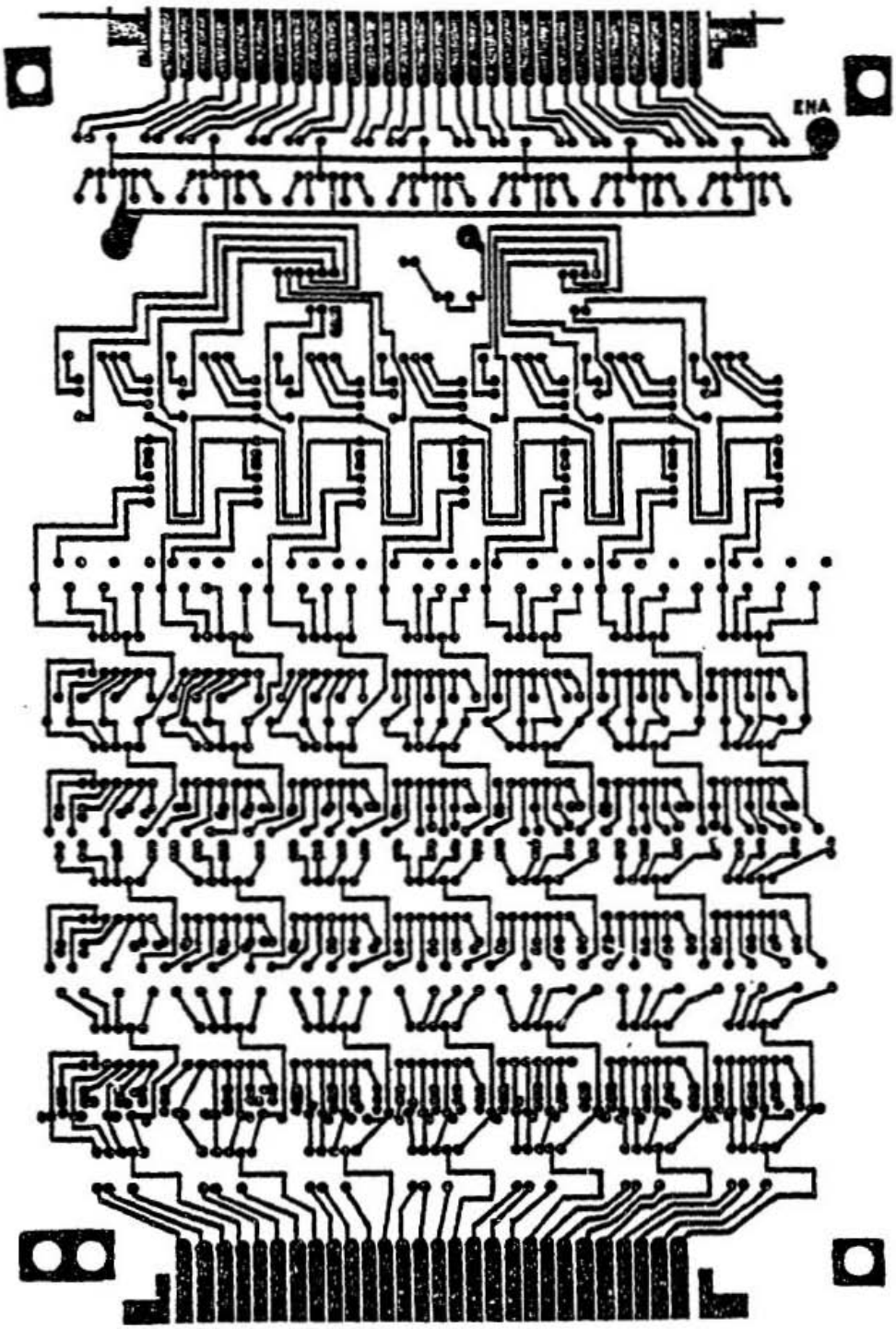


Fig. A2. UNDERSIDE, MAIN A.L.U. PANEL DRAWING

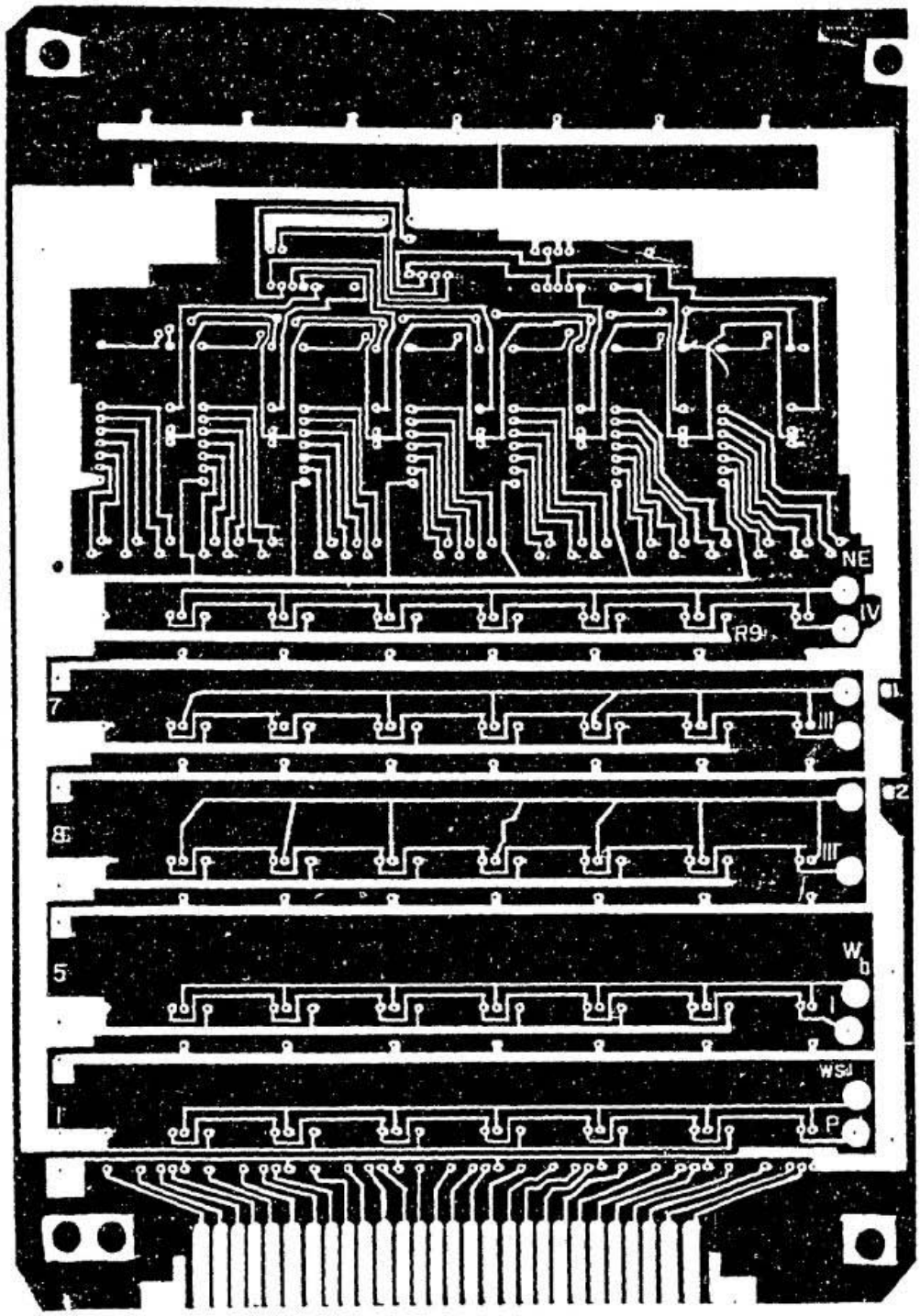


Fig. A3. NEGATJVE OF Fig. A1 FOR COPPER BOARD

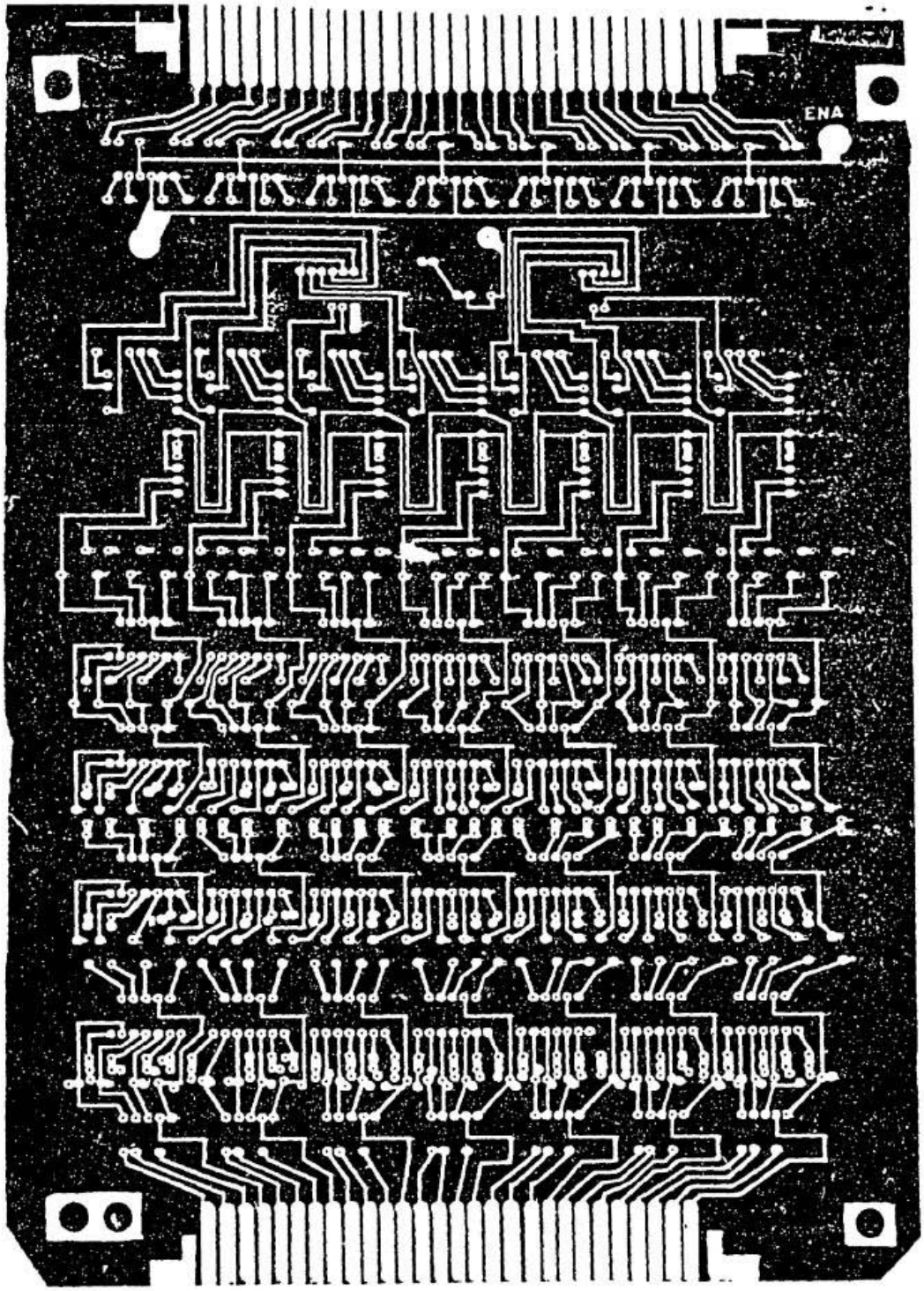


Fig. A4. NEGATIVE OF Fig. A2 FOR COPPER BOARD

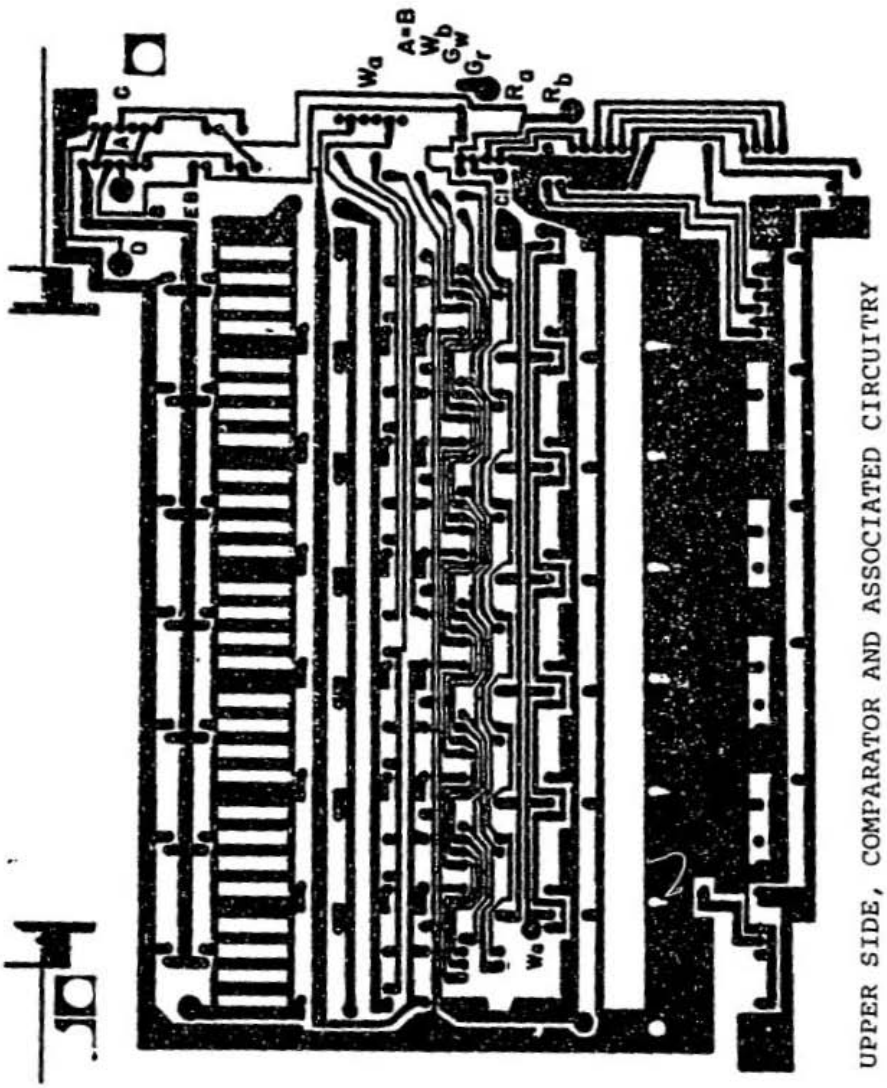


Fig. A5. UPPER SIDE, COMPARATOR AND ASSOCIATED CIRCUITRY DRAWING FOR A.L.U DIVISION

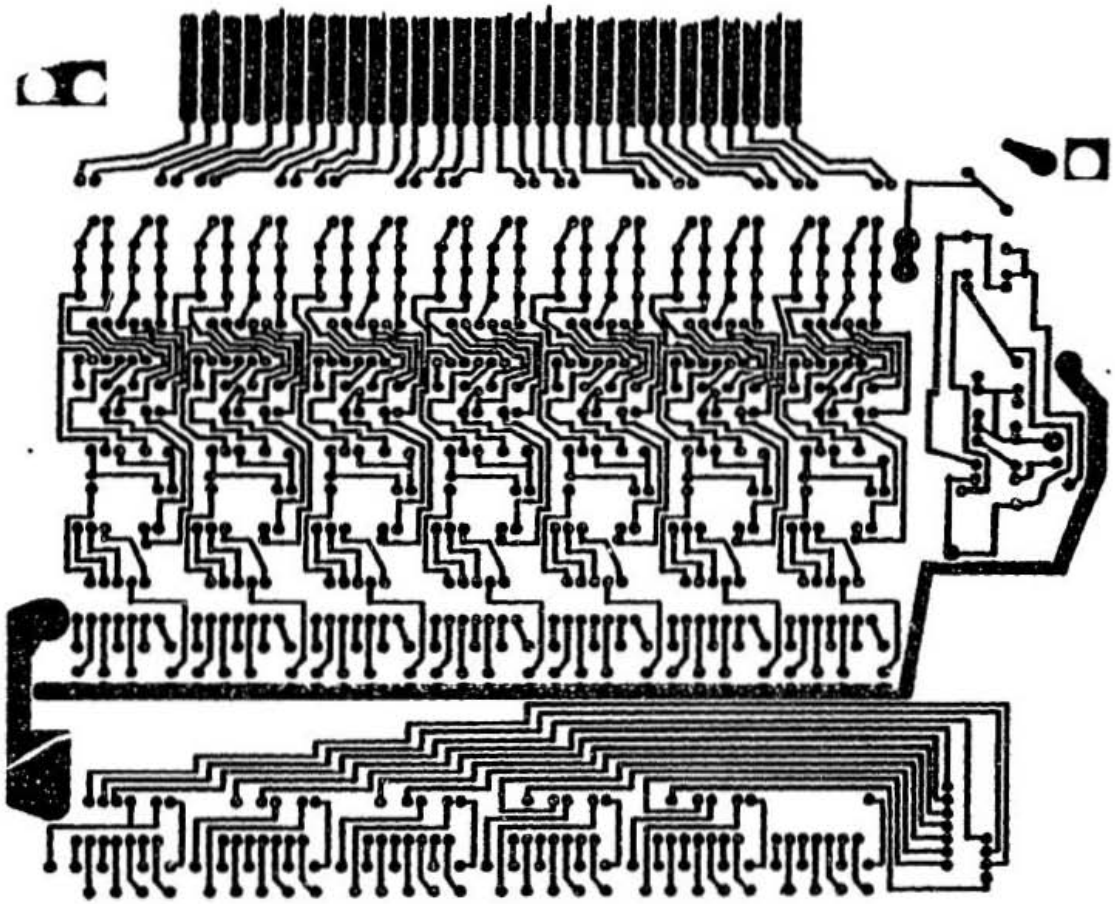
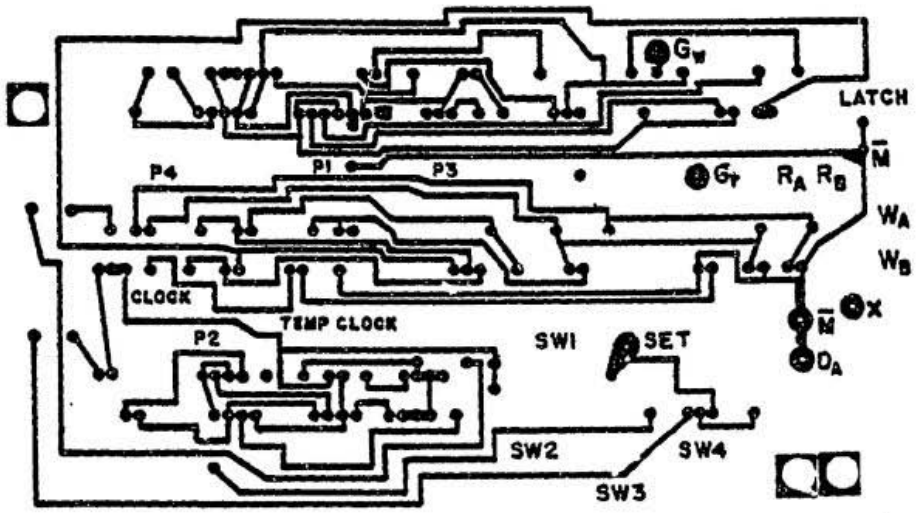


Fig. A6. UNDERSIDE, COMPARATOR AND ASSOCIATED CIRCUITRY DRAWING FOR A.L.U. DIVISION

UPPER SIDE



UNDERSIDE

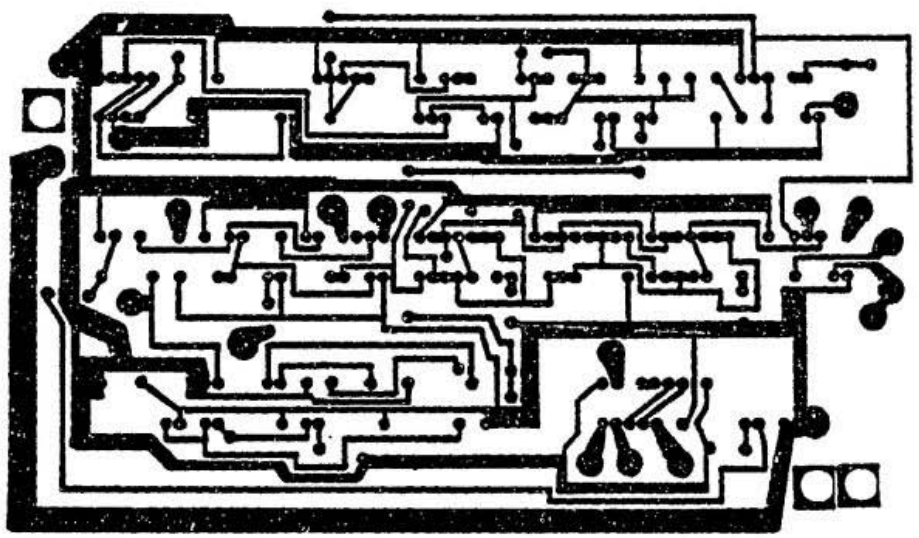


Fig. A7. CONTROL PANEL