A MEMORY ADDRESS SELECTION SYSTEM FOR A MAGNETIC DRUM DIGITAL COMPUTER

By

LAWRENCE GEORGE CLEMENS

Bachelor of Science

Oklahoma State University

Stillwater, Oklahoma

1961

Submitted to the faculty of the Graduate School of

the Oklahoma State University

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

May, 1963

A MEMORY ADDRESS SELECTION SYSTEM FOR A MAGNETIC DRUM DIGITAL COMPUTER

Thesis Approved:

Paul A. McCollum
Thesis Adviser

O. L. Leighbody

Everett MacVicar
Dean of the Graduate School

PREFACE

A quantity of computer components donated to the School of Electrical Engineering of the Oklahoma State University by the Continental Oil Company of Ponca City, Oklahoma, aided in the promotion of a project to design and construct a small digital computer for demonstration and instructional purposes in computer engineering courses. An overall organizational plan has been presented in a previous paper.(1). This paper will explain the considerations and technical details involved in the design, construction, and operation of a memory address selection system for the parallel readout magnetic drum memory of the OSTIC computer.[1]

I wish to express my thanks to my adviser, Professor Paul A. McCollum for his guidance and inspiration. They are sincerely appreciated.

Gratitude is also expressed to Bob Caswell and Joe Lewis of the Electrical Engineering Department of the Oklahoma State University for their technical assistance.

Lastly, the encouragement and support of my wife, Barbara, is eternally acknowledged.

[1] OSTIC - Oklahoma State Instructional Computer. This is a straight binary computer.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER I

INTRODUCTION

The optimum way in which the operating speed of a digital computer can best be utilized is to use the computer as a "stored program" computer. That is, a list of instructions, or program, along with initial data is stored in the memory of the computer. Control is then turned over to the computer which, starting with the first instruction, executes the entire program. Since each instruction is already stored internally, very little time is wasted by the computer being inoperative between execution of instructions, while the next instruction is being obtained.

In the Oklahoma State Instructional Computer, hereafter spoken of as OSTIC, a word is obtained from the memory during an **INSTRUCTION CYCLE** and placed in the **INSTRUCTION REGISTER**, to be interpreted as an instruction. During a **DATA CYCLE**, the word in the **INSTRUCTION REGISTER** is analyzed and executed. This may involve obtaining or storing a word (data) from or into the memory, depending upon the instruction. These two cycles are then repeated for each of the additional instructions in the internally stored program.

The purpose of this thesis is to explain the considerations and technical details involved in the design, construction, and operation of a memory address selection system for the parallel readout magnetic drum memory of the OSTIC.

CHAPTER II

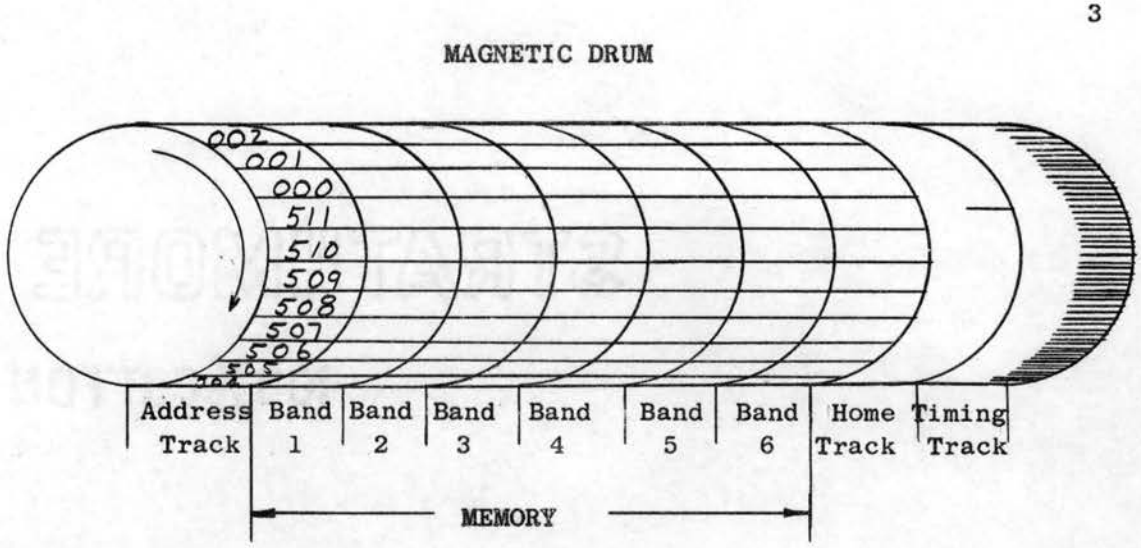
SYSTEMS ASPECT OF MEMORY ADDRESS SELECTION

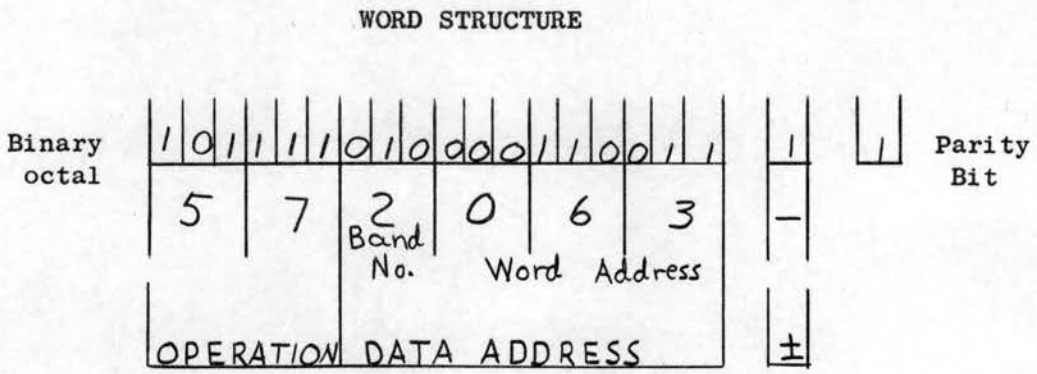
In the OSTIC, a rotating magnetic drum is used for the main memory. It rotates at a nominal 3450 rpm (17.4 milliseconds per revolution). A permanently machined timing track on one end of the drum serves as a source of timing pulses (2560 pulses per revolution). It was decided that words (either instructions or data) would be stored in parallel (all 20 bits of each word parallel to the drum axis). (1).

Figure 1 shows the matrix address structure of the drum memory, the word structure, and the timing pulses.(1). The two low-order bits are used as a sign bit and a parity bit. The remaining 18 bits can be interpreted (depending on the program) either as data or as an instruction. In the latter case, the six high-order bits specify the operation to be executed, the next three bits specify the drum band in which the operand is located, and the remaining nine bits specify the peripheral position on the drum of the operand.

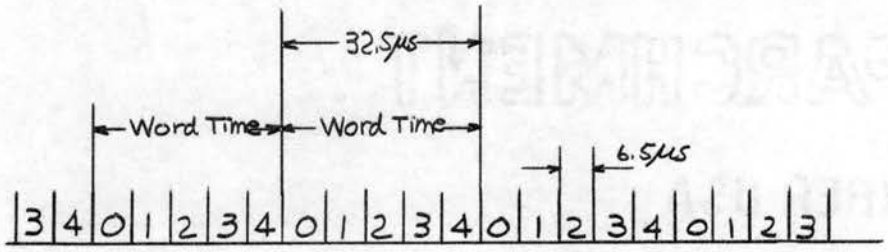
There are 512 memory locations in each of the bands ($2^9 = 512$). Corresponding to each of these memory locations is a nine-bit word address. A particular memory location is specified by the band number (selects a column of the matrix) and the word address (selects a row of the matrix). Details of recording the word address bits will be covered in CHAPTER VII.

The home pulse on the home pulse track immediately follows memory





INSTRUCTION FORMAT



TIMING PULSES

Figure 1. Drum Memory, Word Structure, and Timing Pulses

location 511 and immediately precedes memory location 0, as shown in Figure 1. (1).

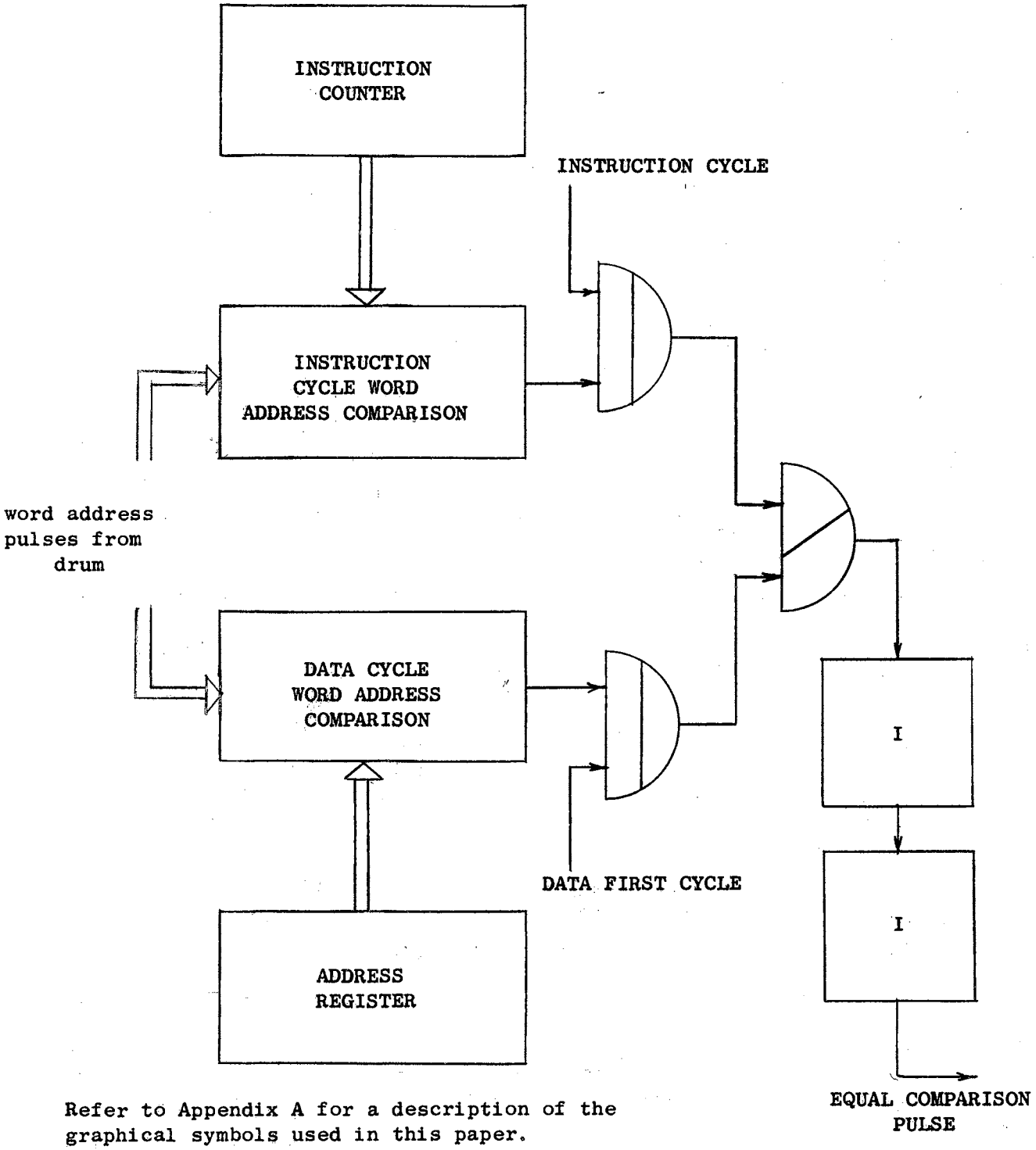
The address of the memory location whose contents are needed next is contained in the 12-bit INSTRUCTION COUNTER during an INSTRUCTION CYCLE, or in the ADDRESS REGISTER (12 low-order bits of the INSTRUCTION REGISTER) during a DATA CYCLE. Of these 12 bits, the nine low-order ones will be referred to as the word address bits, and the three high-order ones will be referred to as the band address bits.

Figure 2 illustrates the method of memory address selection. When a particular word address starts to pass under the word address read heads, this particular sequence of bits will be available as outputs from the word address READ AMPLIFIERS. This sequence will change each word time.

An EQUAL COMPARISON PULSE will be obtained when the word address bits at the word address READ AMPLIFIERS are the same as those in the INSTRUCTION COUNTER during an INSTRUCTION CYCLE, or in the ADDRESS REGISTER during a DATA FIRST CYCLE (portion of a DATA CYCLE). Since the memory locations corresponding to a particular word address pass under the memory read/write heads at the following T3 time, the EQUAL COMPARISON PULSE can be used as a gating signal for this T3 pulse, to produce a read/write pulse at the proper time.

Two comparison circuits are used, instead of one (with the INSTRUCTION COUNTER and ADDRESS REGISTER word address bits being switched), since fewer components are necessary.

The proposed method by which read/write operations are to be accomplished is shown in Figure 3. The band address selects the 20 read/write heads (energized by cathode followers) in one of the bands. An



Refer to Appendix A for a description of the graphical symbols used in this paper.

Figure 2. Memory Address Selection

EQUAL COMPARISON PULSE, a READ signal, and a T3 pulse gate the output of each of the 20 read amplifiers to the DATA CHANNEL and to the INSTRUCTION REGISTER. An EQUAL COMPARISON PULSE, a WRITE signal, a WRITE ZERO or WRITE ONE signal, and a T3 pulse cause the 20 write amplifiers to record a particular sequence of 20 bits in the proper memory location.

A READ signal is present if the computer is in an INSTRUCTION CYCLE, or in a DATA CYCLE if the OPERATION DECODER output indicates that a read operation is required. A WRITE signal is present if the computer is in a DATA CYCLE and the OPERATION DECODER output indicates that a write operation is required. The data being read into the memory will determine whether a WRITE ZERO or a WRITE ONE signal is present at the input gates for each of the write amplifiers.

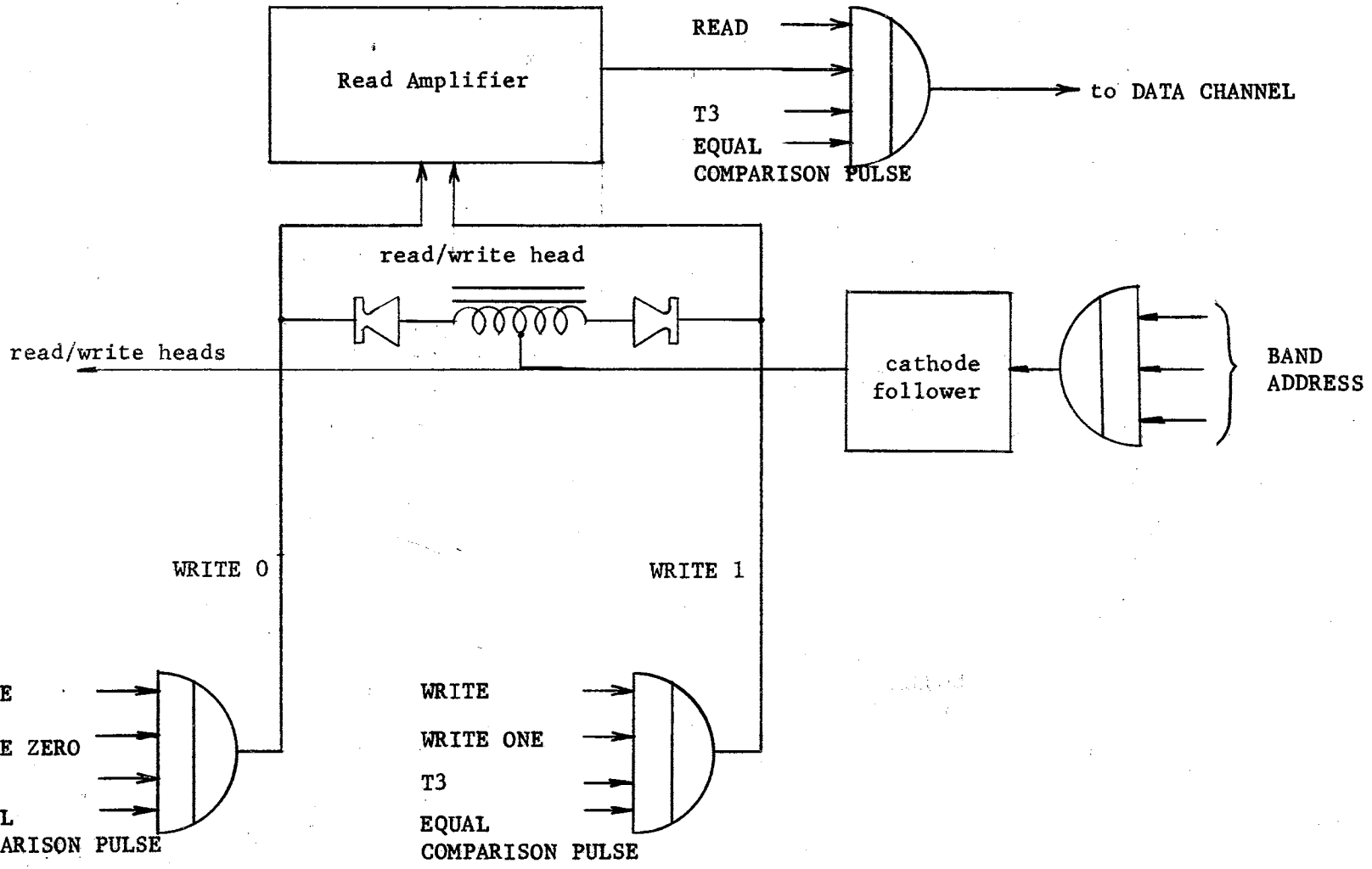


Figure 3. Proposed Method of Read/Write Operation

CHAPTER III

FLIP-FLOP/INVERTER MODULES

Since this is a binary computer, two distinct signal levels are needed to represent a logical 0 and a logical 1. Throughout the computer these are 90 and 110 volts, respectively. By using clamping diodes, voltages above 110 can be pulled down or clamped to 110, and voltages below 90 can be pulled up or clamped to 90. Thus, the flip-flops and inverters can be designed so that their plate voltage swings without clamping vary from much less than 90 to much more than 110; and then clamping provides the signal levels, with increased speed and reliability.

Figure 4 illustrates how plate clamping is effected. When a plate voltage above 110 is clamped, the current through the 12K plate resistor increases above the plate current, the excess current going to the 110 volt reference. When a plate voltage below 90 is clamped, the current through the plate resistor decreases below the plate current, the extra current coming from the 90 volt reference.

For purposes of analysis, a signal level of 90 volts will be referred to as low, conducting, turned on, or logical 0. A signal level of 110 volts will be referred to as high, nonconducting, turned off, or logical 1. This is the positive logic convention.

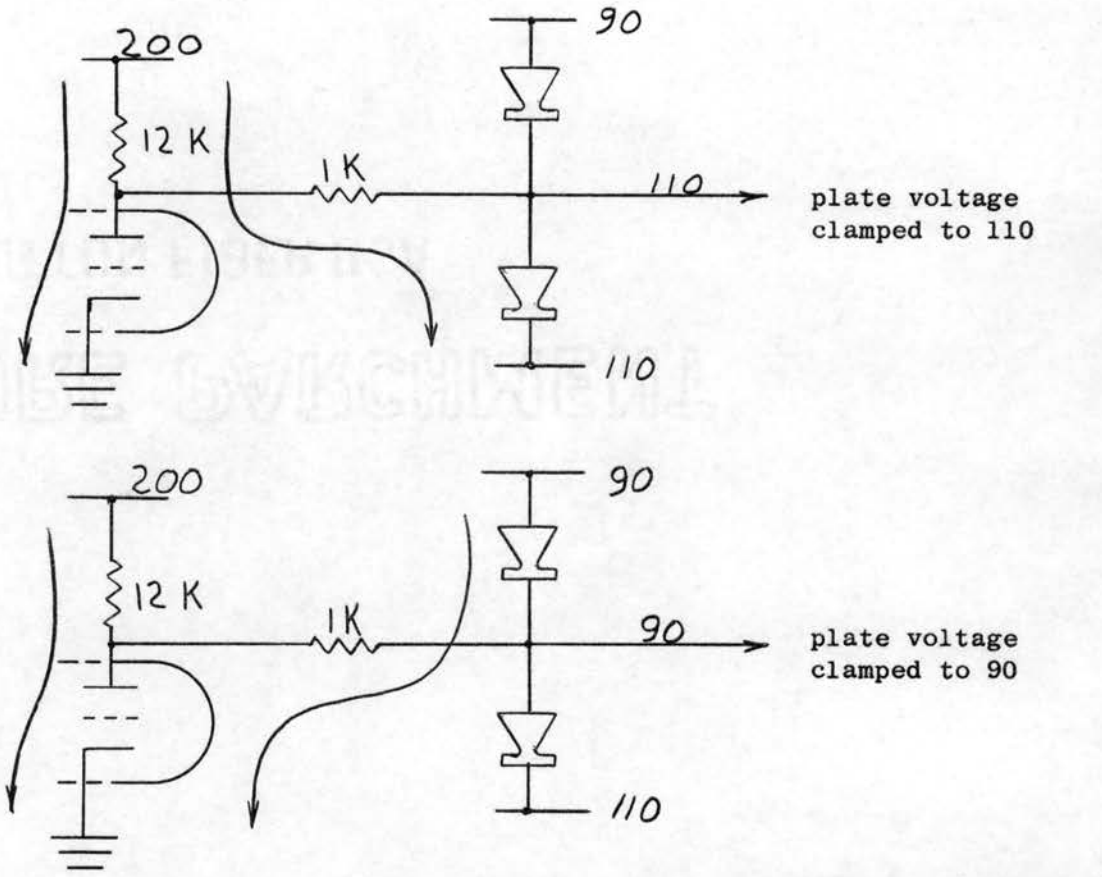


Figure 4. Plate Clamping

The basic inverter module circuit, with various inputs, is shown in Figure 5. Note that there are two independent inverters in each module. If any of the input lines in Figure 5 a are high, e_i will be high, the grid voltage will be clamped to slightly above 0 because of grid current, and the output will be low. The output will be high only if all of the input lines are either low or disconnected. In this configuration, the inputs perform the OR function, while the inverter along with the inputs performs the NOR function.

If any of the input lines in Figure 5 b are low, e_i will be low,

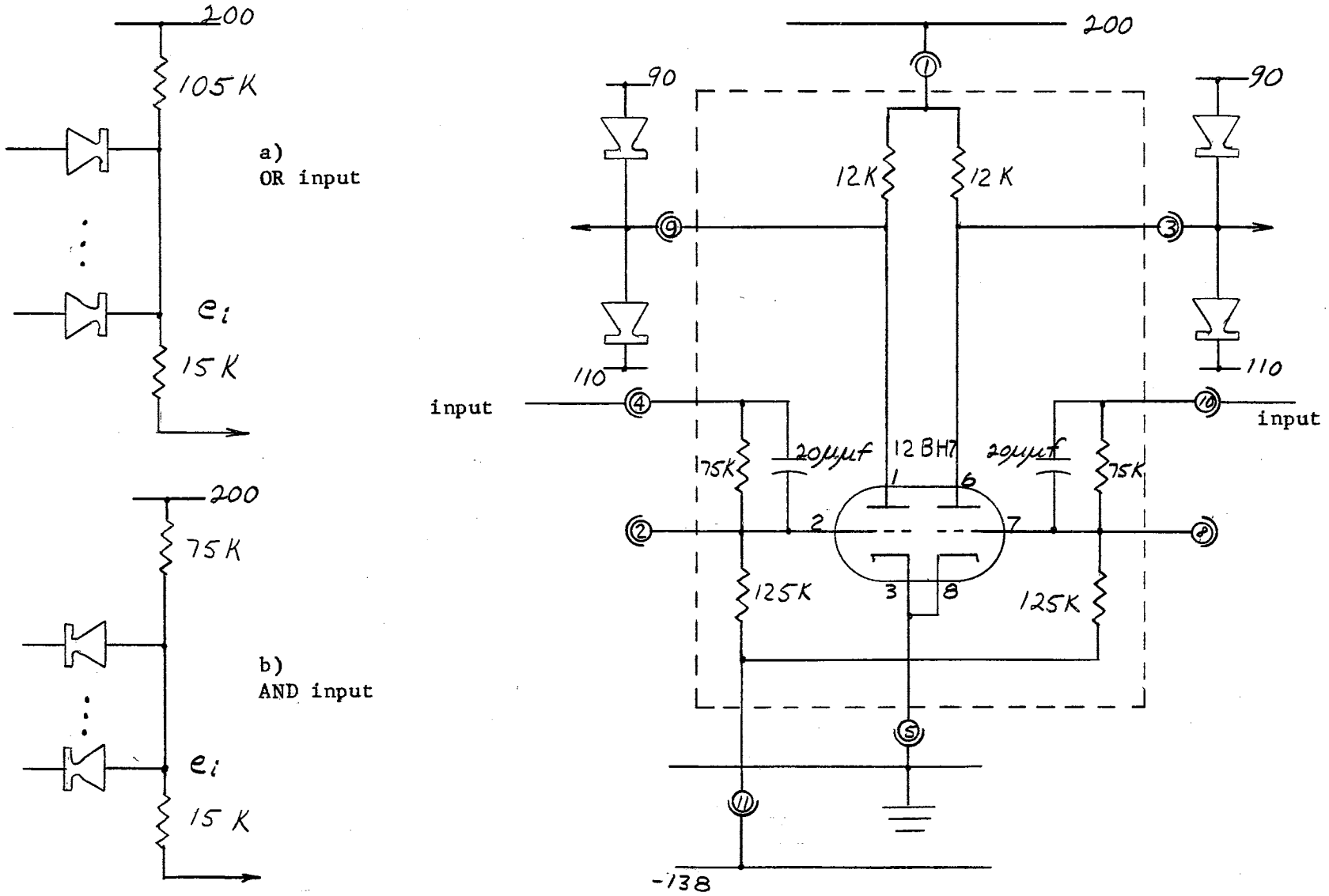


Figure 5. Inverter Module and Associated Input Logic

the grid voltage will be -6, and the output will be high. The output will be low only if all the inputs are either high or disconnected. In this configuration, the inputs perform the AND function, while the inverter along with the inputs performs the NAND function.

If negative logic is used for purposes of analysis (where 90 volts is represented as a logical 1 and 110 volts as a logical 0), in Figure 5 a the inputs perform the AND function, while the inverter along with the inputs performs the NAND function. In Figure 5 b the inputs perform the OR function, while the inverter along with the inputs performs the NOR function.

The aforementioned voltages were determined by voltage divider relationships, since the inverters do not load the diode logic outputs to any extent. These are valid if finite diode back resistance does not appreciably reduce the impedance seen by the output of the diode logic, and the 20$\mu\mu f$ capacitor is of the proper value. This capacitor is sometimes called a "speed-up capacitor" because it improves the speed of response of the circuit. Without this capacitor, the various interelectrode and stray capacitances appearing at the grid would have to be charged through the resistors in the voltage divider. This capacitor can be viewed as being in series with the net capacitive load appearing at the grid with the two capacitances acting like a voltage divider. (2). If the time constants of each part of this voltage divider are equal, the resistive voltage divider analysis is valid for transient conditions also. (3).

The best way to determine the value of this "speed-up capacitor" is to select the value for which the inverter output produces the best reproduction of a square wave input.

If multilevel diode logic precedes the inverter input, finite diode back resistance and loading may make it necessary to modify the inputs shown in Figure 5.

Upon examining Figure 5 and Figure 6, it can be seen that the only difference between the inverter module and the flip-flop module is the external pin connections.

The flip-flop, bistable multivibrator, or binary, consists of two inverters, connected such that the output of the first is the input to the second, and the output of the second is the input to the first. (3). This can be noted by observing the plate-to-grid coupling. The adjective, bistable, indicates that there are two stable conditions for this circuit. One condition is when one of the outputs is high and the other output is low. The other condition is just the reverse of this.

In addition to providing the output signal levels of 90 and 110 volts, plate clamping speeds up the transition between states (for the proper input signal) since the change in clamped plate voltages is less than for the unclamped case.

There are several different methods described in most texts on digital circuitry for triggering a flip-flop. The triggering circuit used by this author is shown in Figure 6 a. The input consists of pulses which can be obtained from a pulse generating circuit or from the output of a logic circuit. The differentiator output consists of a series of positive and negative spikes. When the input goes positive, the flip-flop is not affected since both of the steering diodes are reverse biased. When the input goes negative, the diode connected to the grid of the conducting tube will be forward biased, so this grid voltage will be pulled negative, increasing the plate voltage. It should be

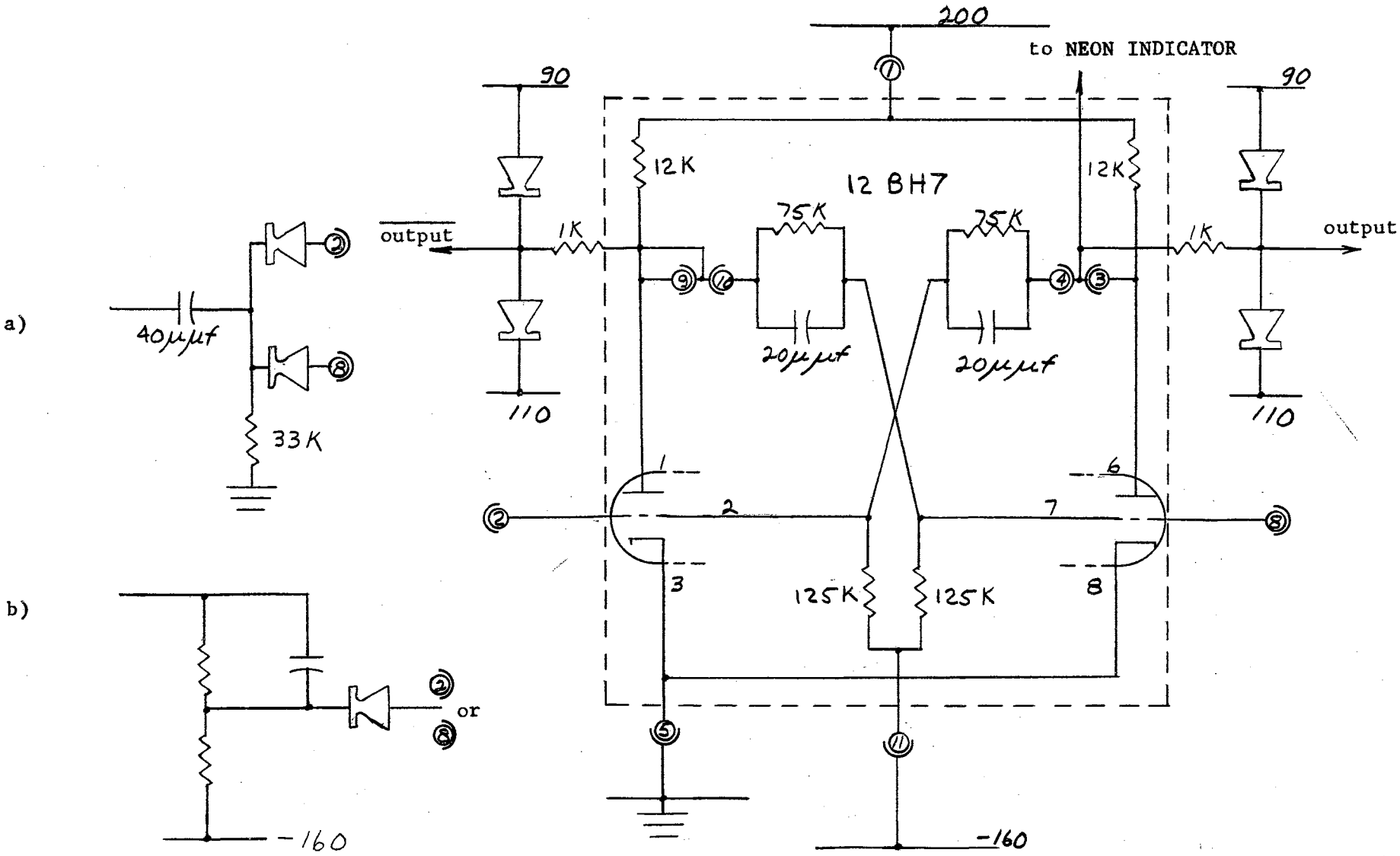


Figure 6. Flip-Flop Module and Associated Input Logic

noted that when this occurs, the differentiator output will be loaded considerably, causing the magnitude and decay time of the negative spike to be less than those of the positive spike. If the grid voltage is pulled far enough negative at a sufficiently rapid rate, the circuit regenerative action will occur, with the result that the flip-flop changes state.

If the negative spike outputs of the differentiator are not sharp enough, the circuit may be critical as to the magnitude of the negative spike that will cause triggering. As the negative spike decays, the steering diode connected to the grid being turned on must remain reverse biased, or the circuit regenerative action can be slowed down or stopped.

If positive pulses are on the differentiator input, it may be necessary to remove the positive spikes with a diode clipper. If narrow negative pulses are on the differentiator input, this will not be necessary since each positive spike will decay by the time the next negative spike occurs.

For resetting (making the output low) or setting (making the output high) the flip-flop, two inputs such as shown in Figure 6 b can be used. As long as the inputs are high, each diode will be reverse biased. When one of the inputs goes low the negative pulse applied to the cathode of that diode will pull the grid voltage negative and turn the tube off.

The resetting and setting operation can be accomplished manually, if desired, by momentarily grounding the proper grid.

CHAPTER IV

INSTRUCTION COUNTER

The INSTRUCTION COUNTER is a 12-bit binary counter, consisting of 12 flip-flop modules and their associated input logic. A block diagram is shown in Figure 7. The trigger, or toggle, input of each stage is connected to the output of the previous stage. That is, a stage will be triggered when the output of the preceding stage goes negative. There is one output pulse for every 2^n input pulses where n is the number of stages between the input and the output. Switch S1 shown in Figure 7 is for converting the three low-order stages from an 8:1 binary counter to a 5:1 counter, through the use of feedback. Applications of the INSTRUCTION COUNTER with this feature will be discussed in CHAPTER VII.

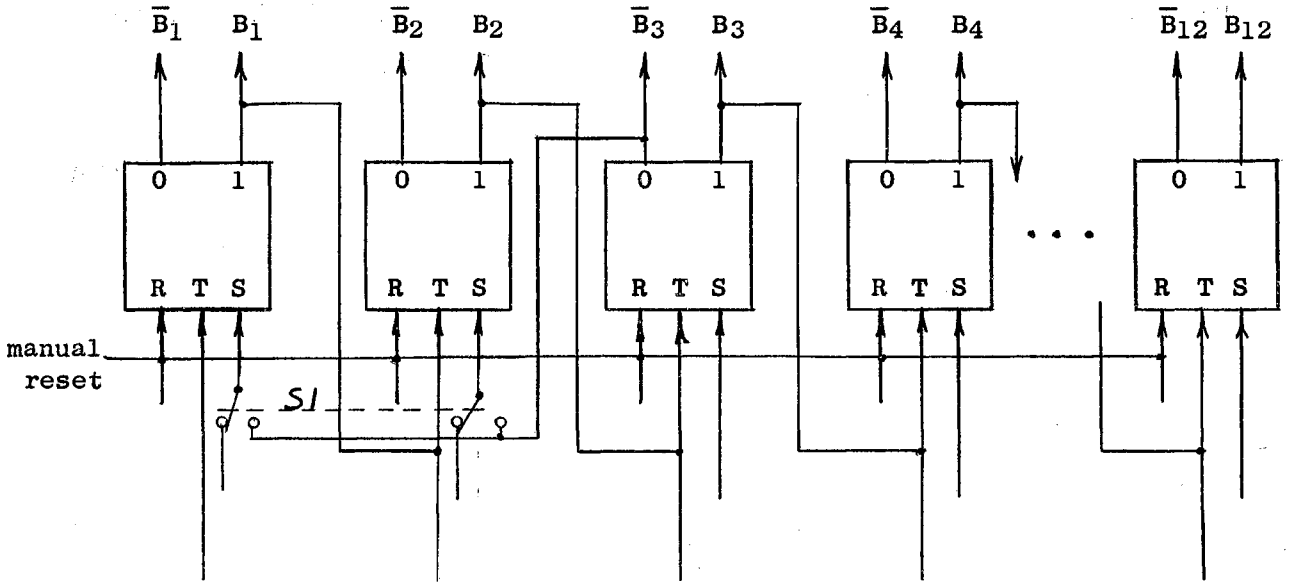


Figure 7. Block Diagram of INSTRUCTION COUNTER

After an instruction has been obtained from memory, analyzed, and executed, the next instruction in the program must be obtained. The INSTRUCTION COUNTER must be incremented so that the new address in the INSTRUCTION COUNTER is that of the next instruction in the program. Normally, the instructions are stored in sequential peripheral locations in the memory of the computer. If this is the case, the INSTRUCTION COUNTER should be incremented by one. Suppose, however, that an instruction has been executed which required 512 word times or less for acquiring the operand and executing the instruction. There may be a considerable (up to almost one drum revolution) time lapse after one instruction is executed until the next one is obtained. Using a variable increment feature, this instruction access time can be reduced to as little as one word time for sequences of certain instructions (when the time required to obtain an operand and complete the instruction is approximately the same for each of these instructions). (1).

To utilize this feature, a SET INCREMENT instruction is included as an instruction in the program. Refer to Figure 8. The word address portion of this instruction contains a 1 in one position and 0's in the other eight positions. After this instruction is obtained from memory, the word address bits are placed in the INCREMENT REGISTER, and at the next T4 time of the current INSTRUCTION CYCLE an INSTRUCTION COUNTER INCREMENT PULSE is gated by the contents of the INCREMENT REGISTER to the trigger input of the appropriate stage of the INSTRUCTION COUNTER. Depending on which position of the word address of the SET INCREMENT instruction contains the 1, the INSTRUCTION COUNTER will be incremented by 2^0, 2^1, 2^2, 2^3, 2^4, 2^5, 2^6, 2^7, or 2^8 at T4 time during the last word time of each succeeding INSTRUCTION CYCLE. The programmer must, of

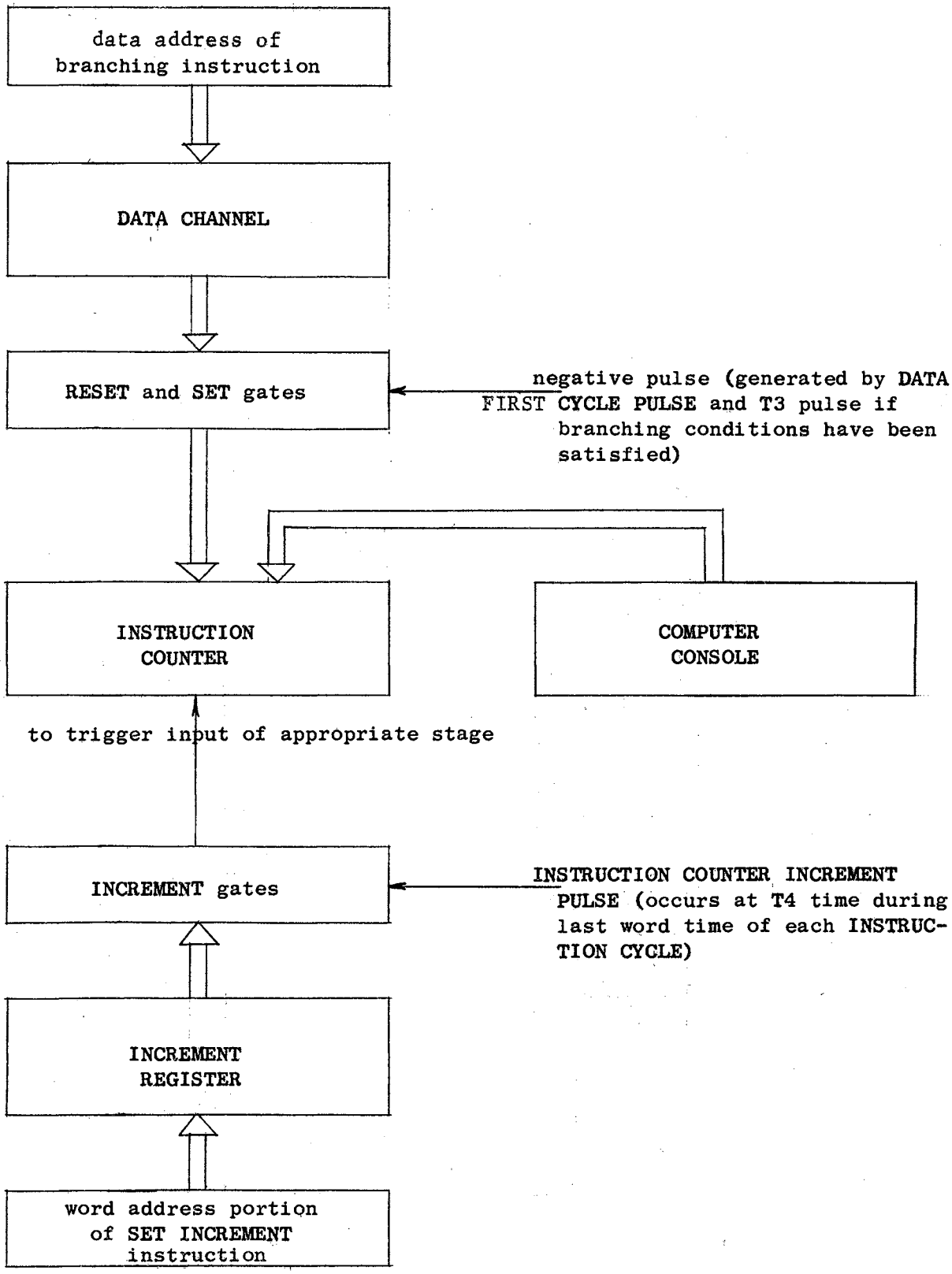


Figure 8. Address Modification in INSTRUCTION COUNTER

course, know the amount of time required to execute each instruction before full advantage can be taken of this feature.

Assuming that the complement of each position in the INCREMENT REGISTER is available and that the INSTRUCTION COUNTER INCREMENT PULSE is a negative pulse, the following logical expression will perform the desired function. I denotes the INCREMENT PULSE, R denotes one of the bits in the INCREMENT REGISTER, $\overline{R}$ denotes the complement of this bit, and $f_{trigger}$ denotes the desired triggering function.

I	R	$\overline{R}$	$f_{trigger}$
0	0	1	1
0	1	0	0
1	0	1	1
1	1	0	1

$$f_{trigger} = \overline{\overline{I}} + \overline{R}$$
$$= I + \overline{R}$$

That is, a particular stage will be triggered on the negative going part of the INSTRUCTION COUNTER INCREMENT PULSE if the complement of the bit in the corresponding position of the INCREMENT REGISTER is low.

Initially, before the INCREMENT REGISTER and other circuitry needed to implement the variable incrementing feature are available, the INSTRUCTION COUNTER can be made to have some fixed increment by letting a pulse that occurs at the end of each INSTRUCTION CYCLE trigger the appropriate stage of the INSTRUCTION COUNTER.

In some operations (such as branching), the address of the next instruction is to be taken from the data address of the branching instruction, rather than that generated by incrementing the INSTRUCTION COUNTER. (1). On one of these operations, if the branching condition is satisfied, this new address must be placed in the INSTRUCTION COUNTER before the next

INSTRUCTION CYCLE begins. Refer to Figure 8.

The data address is read onto the DATA CHANNEL and then, if the branching condition is satisfied, the DATA FIRST CYCLE PULSE and a T3 pulse will be used in generating a negative pulse which will be gated by the contents of the DATA CHANNEL to the proper reset and set inputs of the INSTRUCTION COUNTER.

Assuming that this negative pulse is available at the appropriate time, the following logical expressions will perform the desired functions. P denotes the negative pulse, D denotes one of the bits on the DATA CHANNEL, f_{reset} denotes the desired reset function for the corresponding stage, and f_{set} denotes the desired set function for the corresponding stage.

P	D	f_{reset}	f_{set}
0	0	0	1
0	1	1	0
1	0	1	1
1	1	1	1

$$f_{reset} = \overline{\overline{P}} + \overline{\overline{D}} = P + D$$

$$f_{set} = \overline{\overline{P}} + \overline{D} = P + \overline{D}$$

That is, each stage of the INSTRUCTION COUNTER will be reset (if a 0 is to be entered into that position) or set (if a 1 is to be entered into that position) on the negative going part of this negative pulse.

Since resetting a stage whose output is high will try to trigger the succeeding stage, this negative pulse must be from a low impedance source and of sufficient width to keep any stage from being triggered during this operation.

It may be desired to manually enter some new address into the INSTRUCTION COUNTER. This can be done by resetting the entire INSTRUCTION COUNTER and then setting particular stages. This can be implemented with pushbutton switches on the computer console.

A logic diagram for a single stage of the INSTRUCTION COUNTER is shown in Figure 9.

The circuit diagram for a single stage of the INSTRUCTION COUNTER is shown in Figure 10. The parts of the circuit shown in dashed lines (manual reset and set switches, and the reset and set inputs from the DATA CHANNEL) were not constructed, so ideas pertaining to these are merely suggested possibilities for realizing the full capabilities of the INSTRUCTION COUNTER.

Due to loading problems encountered when attempting to drive diode logic directly from the INSTRUCTION COUNTER outputs, a set of inverters (first level of diode logic consists of AND gates) was included to isolate the INSTRUCTION COUNTER outputs. Since no logic is associated with these inverter inputs, each inverter output produces the complement of its input.

The modular arrangement of the INSTRUCTION COUNTER is shown in Figure 11.

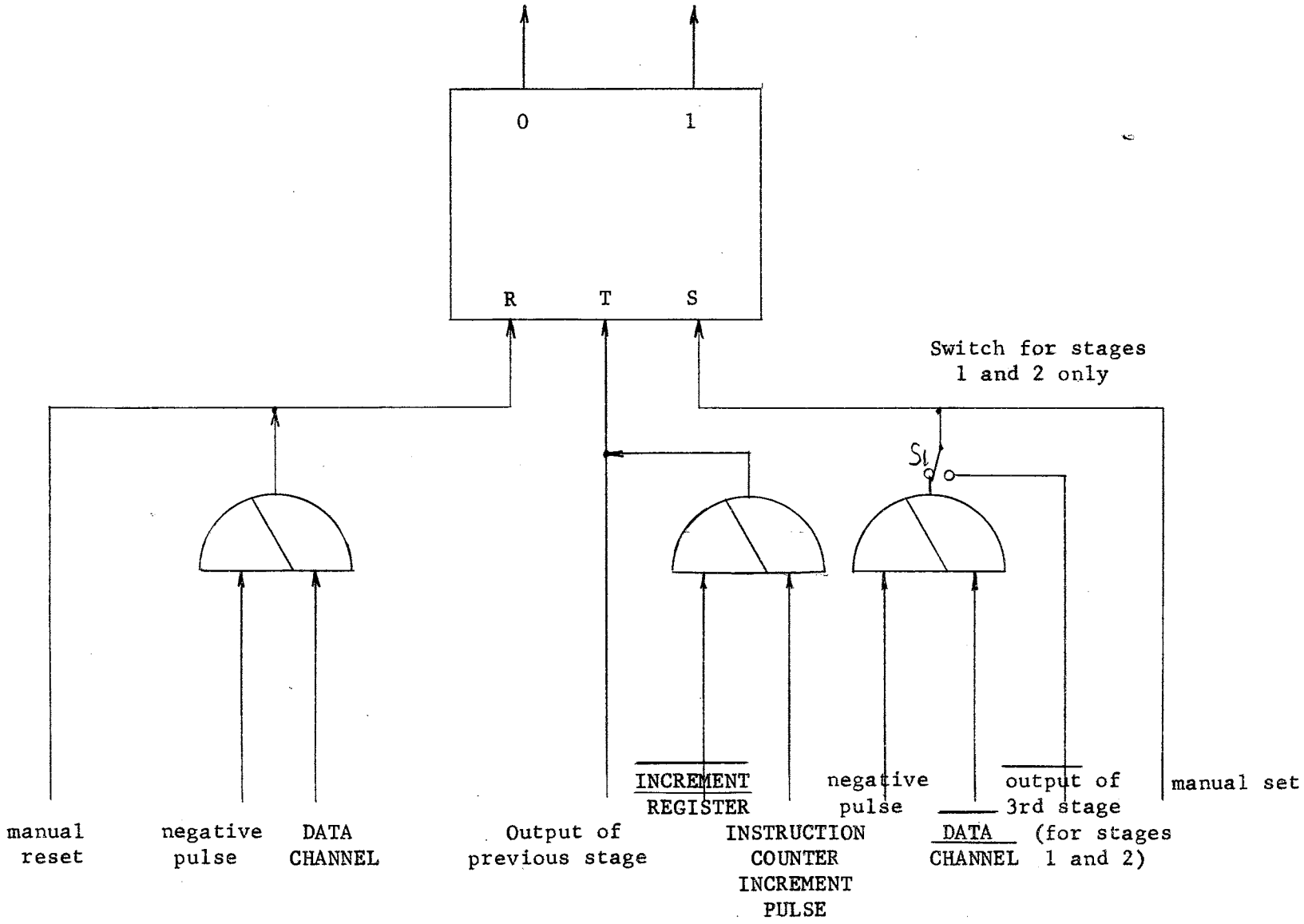


Figure 9. Logic Diagram for a Single Stage of INSTRUCTION COUNTER

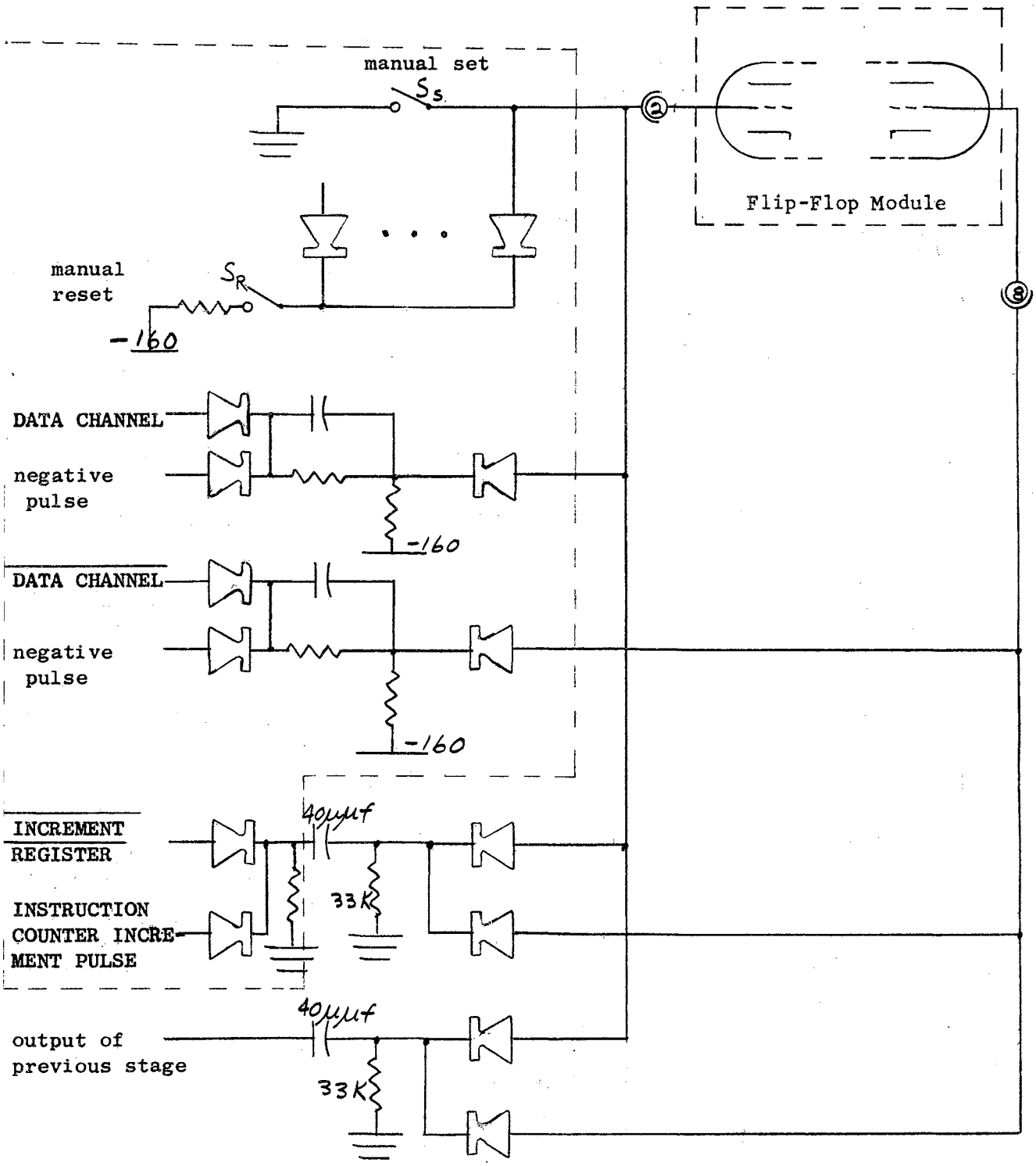


Figure 10. Circuit Diagram for a Single Stage of INSTRUCTION COUNTER

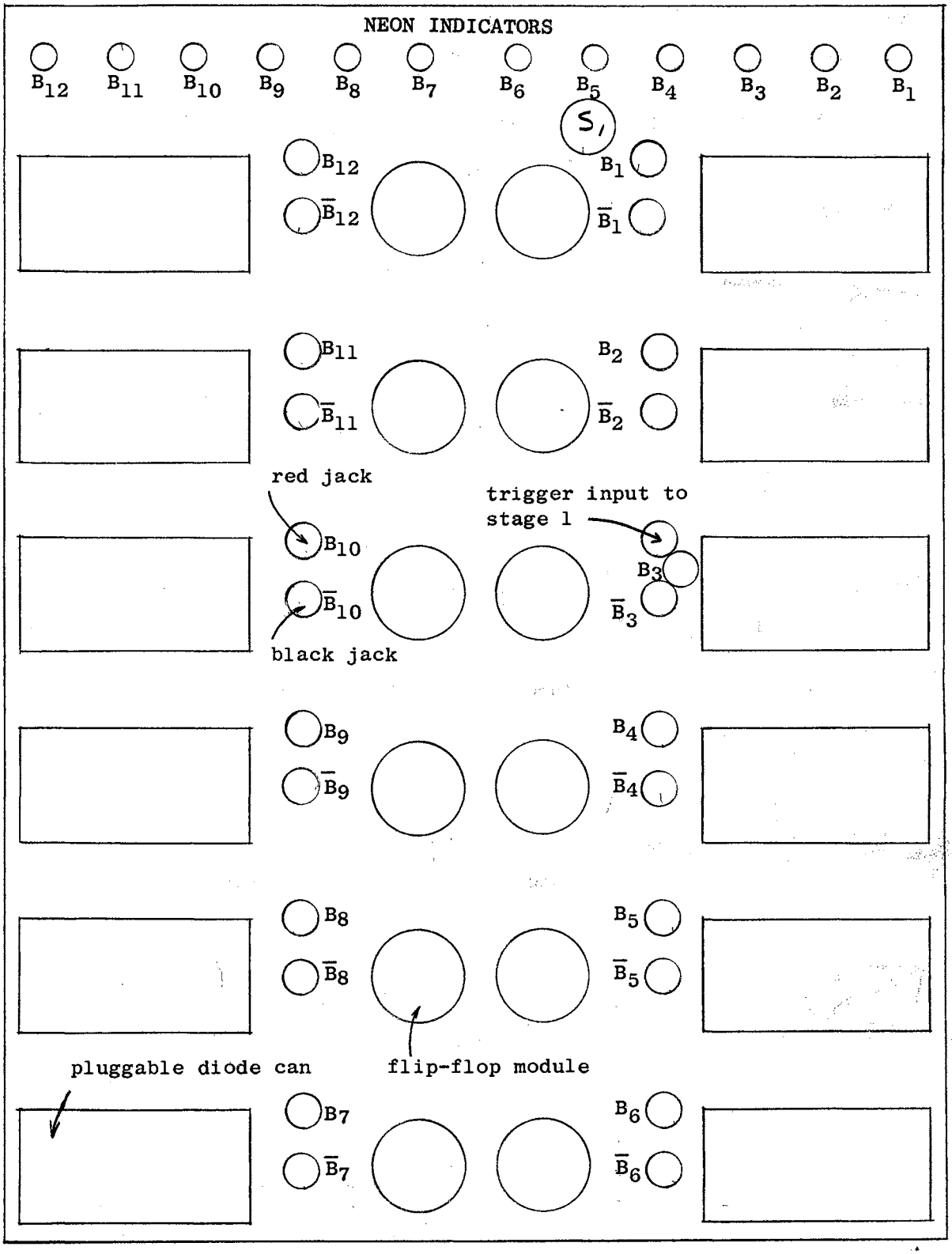


Figure 11. Modular Arrangement of INSTRUCTION COUNTER

CHAPTER V

WORD ADDRESS COMPARISON

Several factors are involved in determining what circuitry to use in implementing the word address comparison logic. If more than two levels of diode logic are used in succession, the output signal may have to be reshaped before being suitable for other use. Two loading conditions that may cause unreliable flip-flop operation are shown in Figure 12.

It can be seen that for a conducting tube whose output is an input to an AND gate, if more current is supplied through the input diode than is required from the 90 volt reference, the plate current will increase. This will decrease the plate voltage and may increase the size of the negative spike needed for triggering, or increase the time required to change the state.

The plate voltage of a nonconducting tube whose output is an input to an OR gate may decrease to a point below the clamping voltage if loaded too excessively.

If the impedance seen by an OR gate (fed by an AND gate) is shunted to any degree, possibly by finite diode back resistance, the output may be pulled low enough to reverse bias a normally conducting diode at one of the AND gate inputs.

The word address at the word address READ AMPLIFIERS, available for comparison at T1 time, will agree with that in the INSTRUCTION COUNTER (or ADDRESS REGISTER) when the following expression is equal to logical 1.

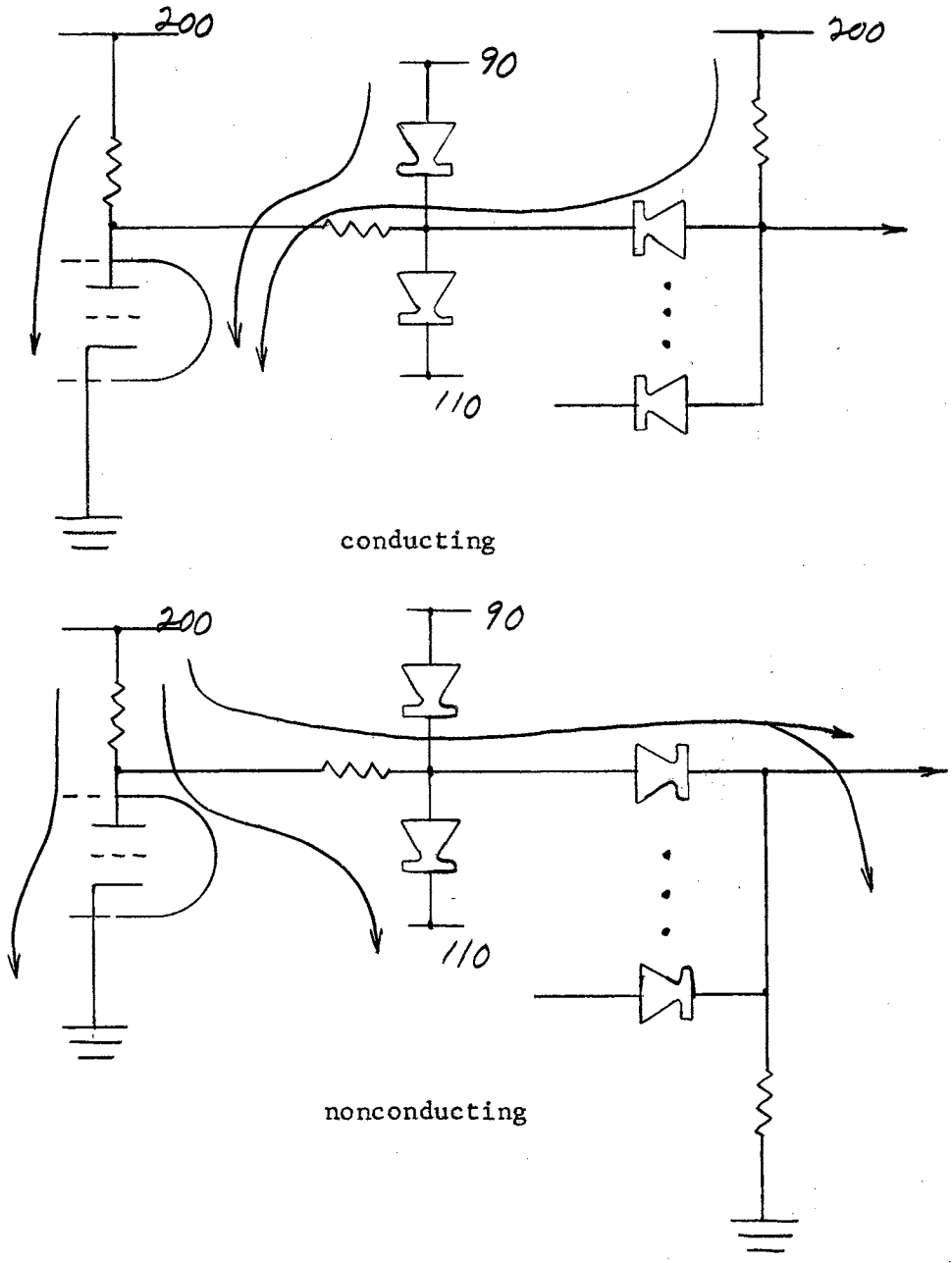


Figure 12. Flip-Flop Loading by Diode Logic

D_i denotes the bit in the ith position at the word address READ AMPLIFIERS, and B_i denotes the bit in the ith position of the word address in the INSTRUCTION COUNTER (or the ADDRESS REGISTER).

$$f = \prod_{i=1}^{9} (B_i D_i + \bar{B}_i \bar{D}_i)$$

Taking the complement of both sides produces

$$\bar{f} = \sum_{i=1}^{9} [(\bar{B}_i + \bar{D}_i)(B_i + D_i)]$$

$$= \sum_{i=1}^{9} (B_i \bar{D}_i + \bar{B}_i D_i)$$

Since $\overline{X + Y + Z} = \bar{X}\,\bar{Y}\,\bar{Z}$, the preceding expression can be rearranged as follows.

$$f = [\overline{(B_1\bar{D}_1 + \bar{B}_1 D_1 + B_2\bar{D}_2 + \bar{B}_2 D_2 + B_3\bar{D}_3 + \bar{B}_3 D_3)}$$

$$\overline{(B_4\bar{D}_4 + \bar{B}_4 D_4 + B_5\bar{D}_5 + \bar{B}_5 D_5 + B_6\bar{D}_6 + \bar{B}_6 D_6)}$$

$$\overline{(B_7\bar{D}_7 + \bar{B}_7 D_7 + B_8\bar{D}_8 + \bar{B}_8 D_8 + B_9\bar{D}_9 + \bar{B}_9 D_9)}]$$

Two circuits, each performing this logical function as shown in Figures 2 and 13 will, when their outputs are gated by the proper control signals, produce an EQUAL COMPARISON PULSE.

Figure 14 shows a portion of the circuit diagram for the preceding logical function.

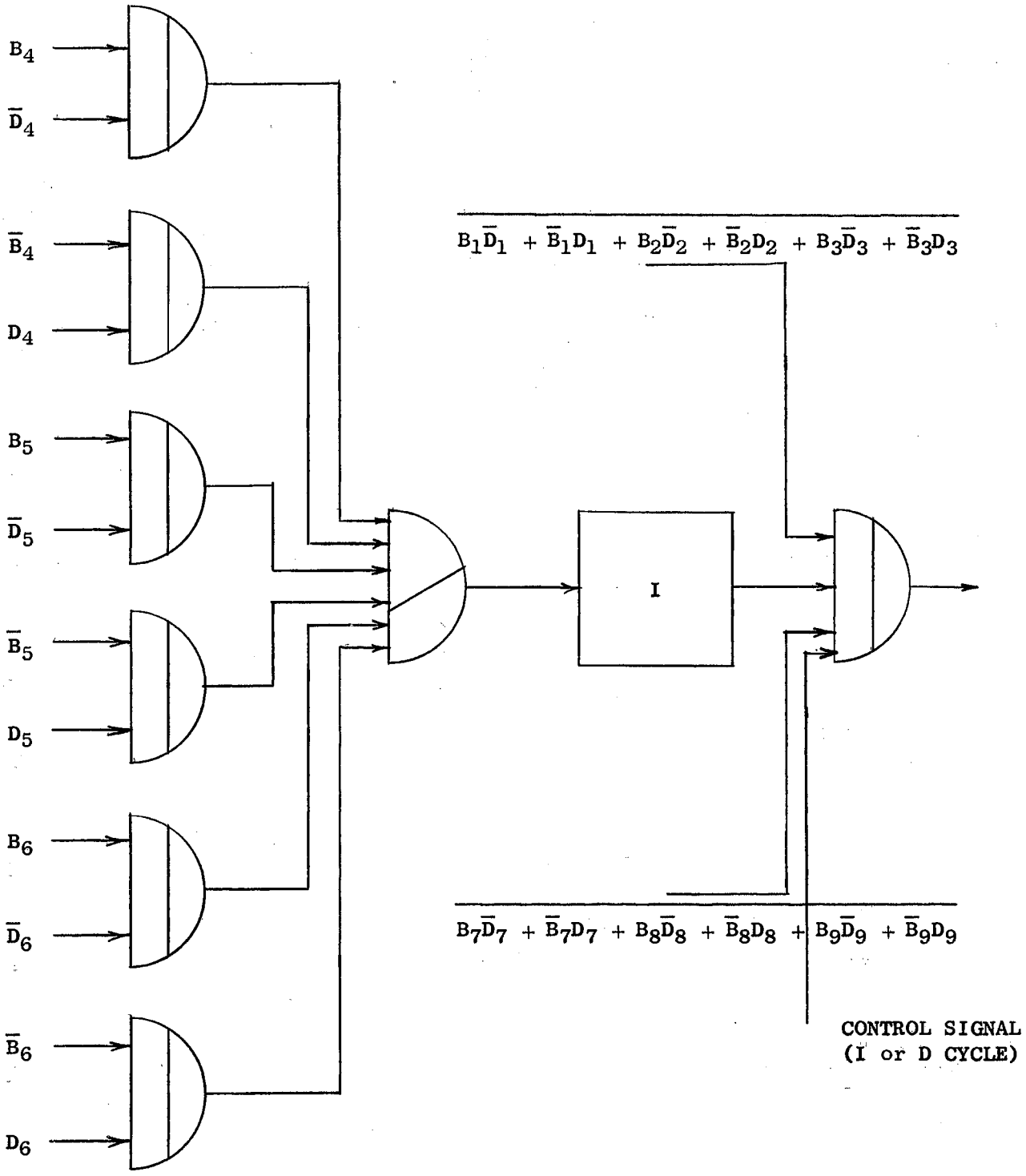


Figure 13. **WORD ADDRESS COMPARISON** Logic

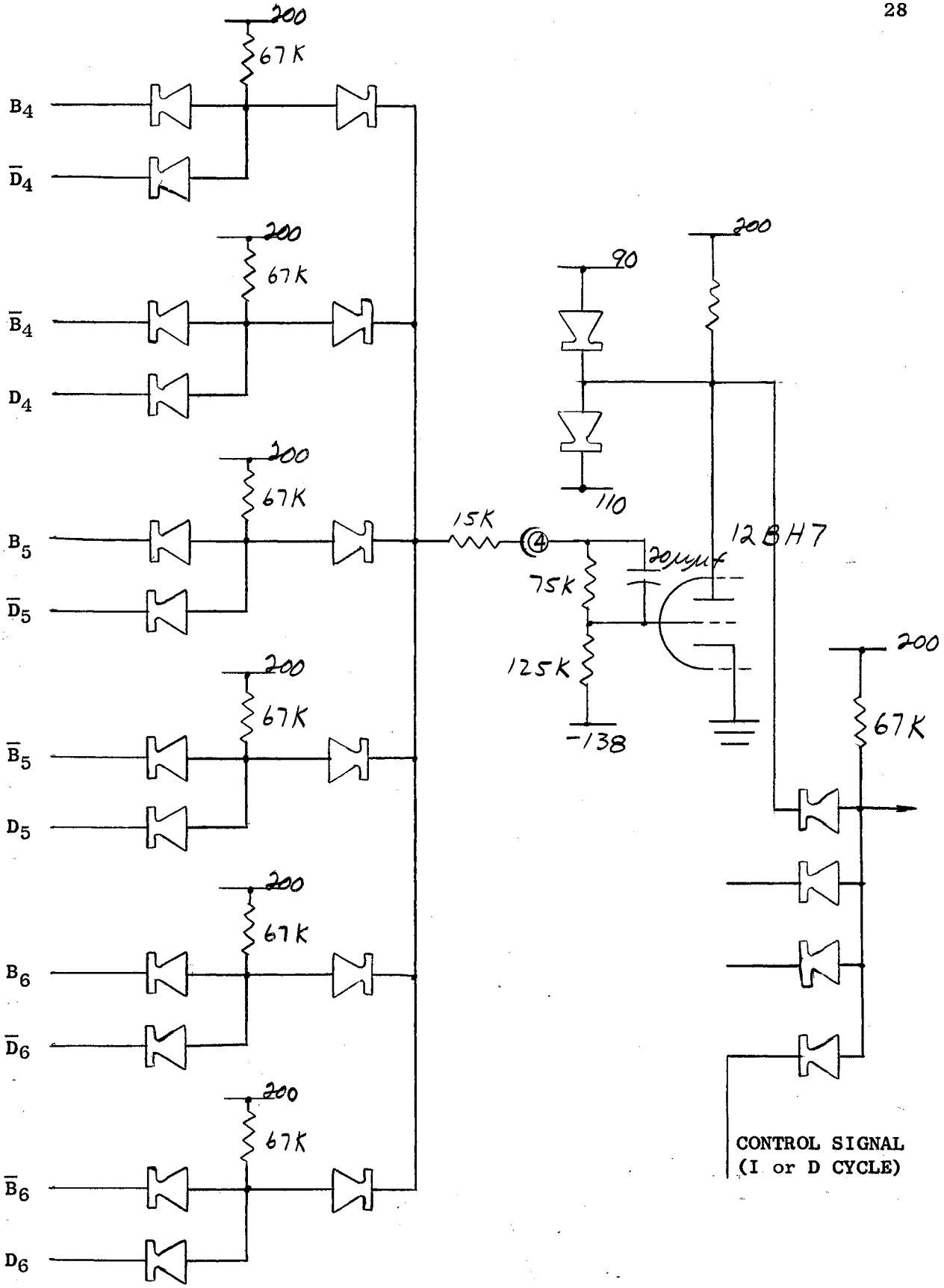


Figure 14. Partial Circuit Diagram for WORD ADDRESS COMPARISON Logic

CHAPTER VI

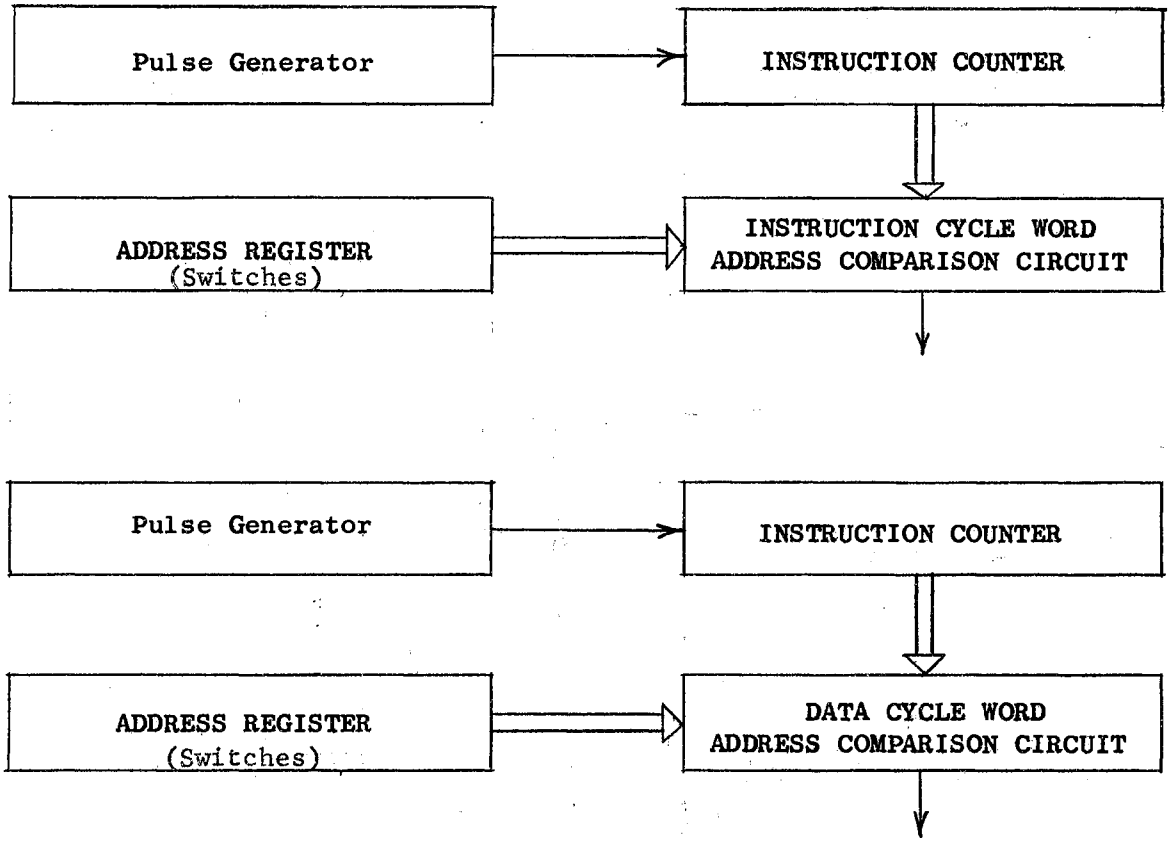
TESTING

Initial testing of the WORD ADDRESS COMPARISON CIRCUIT was made in the following manner, one half being tested at a time, as indicated in Figure 15. The word address READ AMPLIFIERS were simulated by the INSTRUCTION COUNTER, since the outputs of the INSTRUCTION COUNTER when driven by a pulse generator at the proper pulse repetition rate are identical to those of the word address READ AMPLIFIERS (except for delay in the counter and when the word address is changing). Nine double pole - double throw switches, simulating 90 or 110 volts for each bit and its complement, replaced the INSTRUCTION COUNTER when testing the INSTRUCTION CYCLE part of the WORD ADDRESS COMPARISON CIRCUIT, or the ADDRESS REGISTER when testing the DATA CYCLE part of the WORD ADDRESS COMPARISON CIRCUIT. Alternatively, the switches can simulate the word address READ AMPLIFIERS, with the INSTRUCTION COUNTER being used as the other set of inputs for either part of the WORD ADDRESS COMPARISON CIRCUIT. The control signal inputs shown in Figure 13 were left open.

Waveforms similar to those expected in actual operation were observed. An EQUAL COMPARISON PULSE was obtained every 512 word times. The duration was one word time.

Some output waveforms can occur using this testing method that will not occur in normal operation of the WORD ADDRESS COMPARISON CIRCUIT. Suppose, for example, that the second and third position bits agree, the

a)



b)

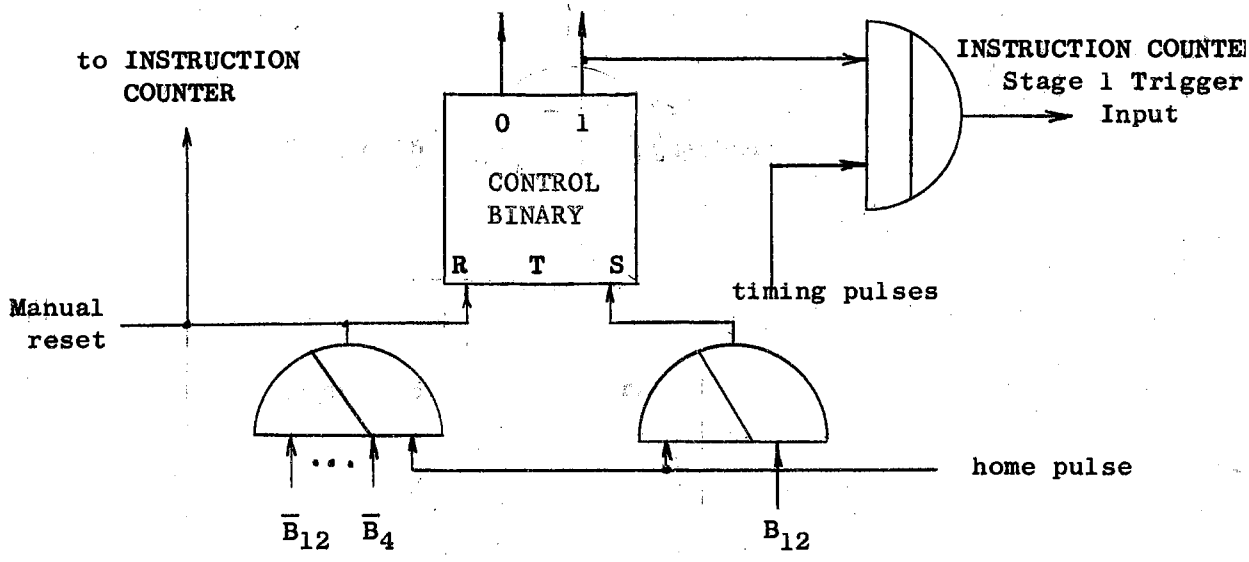


Figure 15. Testing Arrangements

low order bit in the INSTRUCTION COUNTER is high, and the low order bit in the switches is low. There is only one AND gate whose output is high, for these three low order stages. Refer to Figure 13. The next triggering pulse will cause the bit in the low order position of the INSTRUCTION COUNTER to change. While it is changing, and before the second stage of the INSTRUCTION COUNTER is triggered by this change, the output of this AND gate will be going low. It will be pulled back up when the second stage is triggered, however. Due to delay in the INSTRUCTION COUNTER and particular switch settings then, pulses of approximately one microsecond width can appear as outputs from the WORD ADDRESS COMPARISON CIRCUIT.

If wide positive pulses clamped to a 90 volt level are applied through an OR gate, whose other input is the EQUAL COMPARISON PULSE, to the trigger input of the INSTRUCTION COUNTER low-order stage, the INSTRUCTION COUNTER will be triggered until coincidence occurs, the gate being closed by the EQUAL COMPARISON PULSE going high. At this point the neon indicators on the INSTRUCTION COUNTER chassis and the switches should indicate the same word addresses. They may not agree because of delay in the INSTRUCTION COUNTER and particular switch settings, as mentioned in the preceding paragraph.

The preceding method of testing can also be used to determine the maximum delay in the INSTRUCTION COUNTER and the WORD ADDRESS COMPARISON CIRCUIT, which occurs when the word address in the INSTRUCTION COUNTER is changing from 1···1 to 0···0. The word address in the switches can be set to 1···1 and the pulse repetition rate of the triggering signal varied until it is slow enough for this "turn-off" action to occur. For these switch settings, the extra pulses are not obtained from the WORD ADDRESS COMPARISON CIRCUIT output.

It is most essential that some relatively simple methods be available for testing the operation of the INSTRUCTION COUNTER and the WORD ADDRESS COMPARISON CIRCUIT when the OSTIC is operational.

When it is turned off, the method just described for initial testing can be applied. If the computer is turned on, but has not started executing the stored program, this same method can be used, without simulation.

If the EQUAL COMPARISON PULSE is disconnected from the read amplifier input gates, so as not to change the contents of the manually set ADDRESS REGISTER, and the control signal inputs in Figure 13 are left open, the outputs of each part of the WORD ADDRESS COMPARISON CIRCUIT will coincide, if the word addresses in the INSTRUCTION COUNTER and ADDRESS REGISTER are the same.

A simple method that can be used to determine the unit reliability of the INSTRUCTION COUNTER as a binary counter along with either part of the WORD ADDRESS COMPARISON CIRCUIT, is as follows. The EQUAL COMPARISON PULSE is applied to the trigger input of the low-order stage of the INSTRUCTION COUNTER. After the initial coincidence occurs, the negative going slope of the EQUAL COMPARISON PULSE will trigger the INSTRUCTION COUNTER. After a short delay, depending on how many stages are caused to trigger, the INSTRUCTION COUNTER will have been incremented by one. The word address at the word address READ AMPLIFIERS will also, in effect, have been incremented by one, so another coincidence occurs, and the foregoing action is repeated. The EQUAL COMPARISON PULSE output will be at 110 volts, except when the INSTRUCTION COUNTER is changing states, or the word address at the word address READ AMPLIFIERS is changing.

One method for testing the INSTRUCTION COUNTER makes use of the 5:1 modification feature since, if the three low-order stages function as a

5:1 counter, and the nine high-order ones as a binary counter, the entire counter will start over once every drum revolution. The logical diagram of Figure 15 b applies if negative timing pulses, and a negative home pulse occuring simultaneously with the T3 pulse of word time 511, are available. The entire INSTRUCTION COUNTER as well as the CONTROL BINARY are reset manually. Since the outputs of the INSTRUCTION COUNTER are not high, the next home pulse will set the CONTROL BINARY. After some delay (the T3 pulse will already have gone negative and will not have triggered the INSTRUCTION COUNTER), the output of the CONTROL BINARY will go high, gating succeeding timing pulses to the trigger input of the INSTRUCTION COUNTER low-order stage. The next home pulse (or the simultaneous T3 pulse) will be the last pulse to trigger the INSTRUCTION COUNTER. The neon indicator lights on the INSTRUCTION COUNTER chassis will now all be lighted, unless the INSTRUCTION COUNTER did not function properly. It may be noted that the CONTROL BINARY is not reset by the initial home pulse, after manual resetting, but is reset by the succeeding home pulse, after the simultaneous timing pulse has been gated to trigger the INSTRUCTION COUNTER.

Initially, the INSTRUCTION COUNTER was tested by using a pulse generator to trigger the input stage. For testing the reset inputs and the set inputs, some means of applying one negative pulse at a time is needed. This can be accomplished by momentarily touching the input to the output of a pulse generator whose period is slow enough to make this a valid test procedure.

To test the overall memory address selection system, probably the simplest and most obvious way is to store a word in some location and then retrieve it. This might be done by stepping through a program one

cycle at a time and observing neon indicator lights on the computer console. Alternatively, the operator might use part of the program to compare the stored word and the retrieved word. This would, however, involve use of the ACCUMULATOR and the circuits used in testing whether the various branching conditions have been satisfied.

The memory address comparison circuits are tested to an extent in another way; since, unless an EQUAL COMPARISON PULSE is obtained, the next instruction or the operand will not be obtained.

CHAPTER VII

RECORDING WORD ADDRESS PULSES ON THE DRUM

Figure 16 indicates how the modified INSTRUCTION COUNTER affects the counting operation. Each time the complement of the third stage output goes negative, the two low-order stages are set, the effect being the same as if three additional input pulses were applied. The entire counter is thus made to start over every 2560 input pulses.

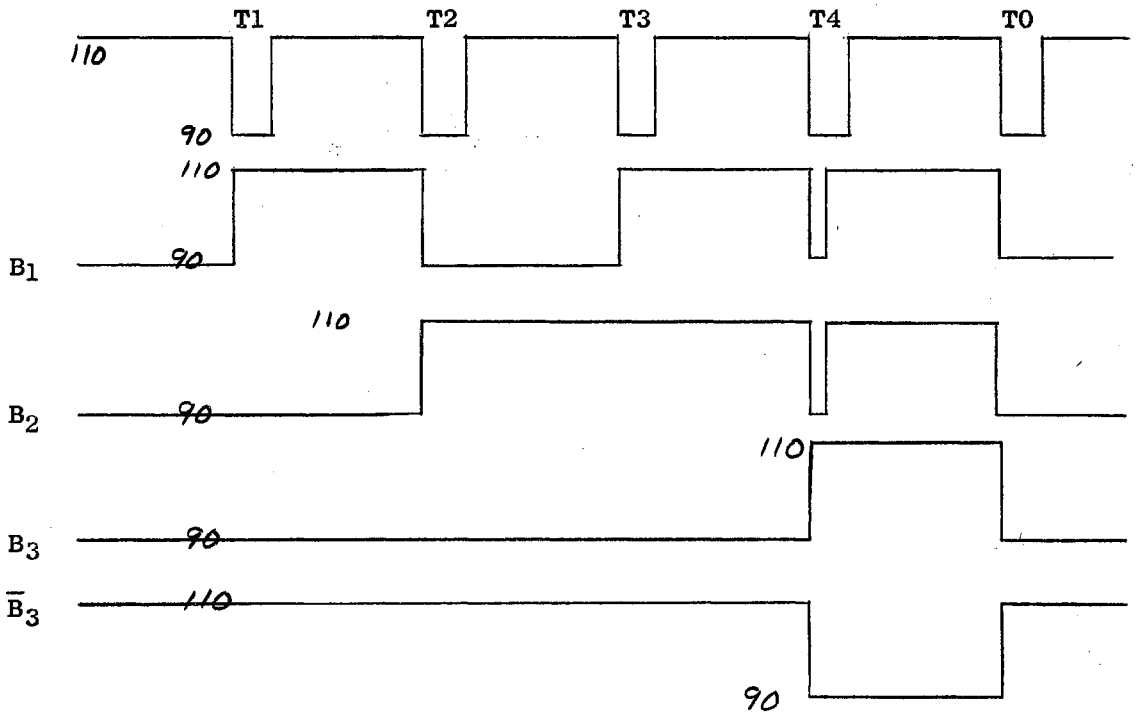


Figure 16. 5:1 Counter Timing Diagram

It was mentioned in CHAPTER V that the bits at the word address READ AMPLIFIERS will be available for comparison at T1 time and in CHAPTER II that memory read/write operations will occur at T3 time. For this to be so, the word address pulses must be recorded with respect to the timing pulses, starting prior to each T1 pulse and ending after the following T3 pulse. Figure 17 shows one method of accomplishing this. Initially the entire INSTRUCTION COUNTER is manually reset. The timing pulses from the drum are then used to trigger the low-order stage of the INSTRUCTION COUNTER. The output of the third stage of the INSTRUCTION COUNTER goes negative on every fifth input pulse from the drum and triggers the fourth stage as well as the pulse generator, whose delay and pulse width can be adjusted to provide a positive output pulse suitable for use as a recording pulse.

For the recording arrangement shown in Figure 17, this recording pulse should end before the next triggering pulse from the 5:1 counter occurs. The output of each of the nine high-order stages of the INSTRUCTION COUNTER can be considered as a WRITE ONE signal and the complement of each of these outputs as a WRITE ZERO signal. Nine write amplifiers are connected as shown, with the T3 pulse inputs replaced by the recording pulse, the WRITE signal inputs disconnected, and the EQUAL COMPARISON PULSE inputs disconnected. The bits in the INSTRUCTION COUNTER thus determine whether each recording pulse causes a 0 or a 1 to be recorded in the corresponding bit positions for each word address.

After the word address pulses have been recorded there must be a timing reference to insure that the timing pulses and the word address pulses maintain their specified time relationship. This can be done with a home pulse which, when the computer is started each time will set

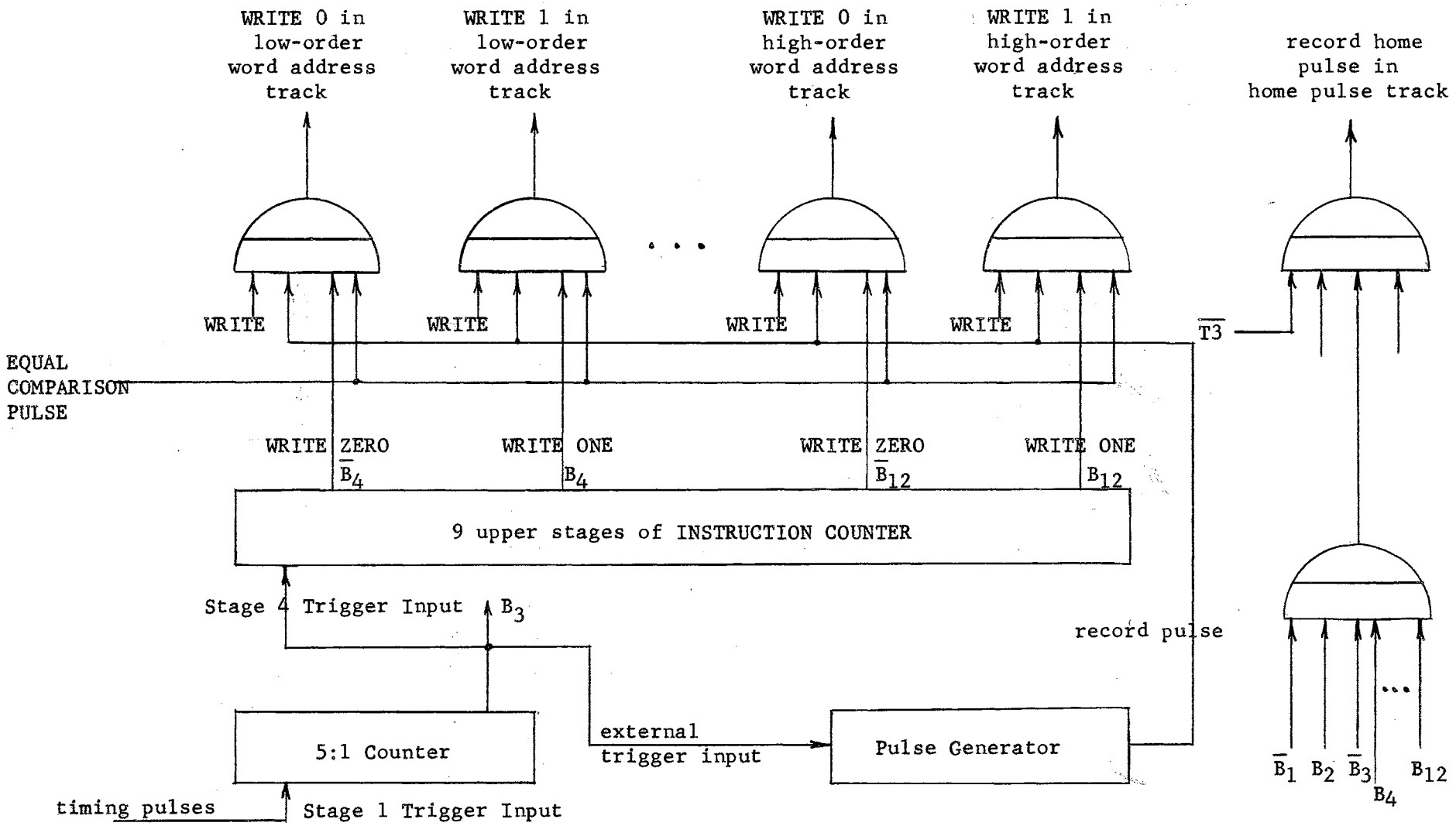


Figure 17. Recording Word Address Pulses and Home Pulse on Drum

this time relationship. That is, the home pulse can be recorded so as to occur simultaneously with one of the five timing pulses immediately preceding memory location 0. When the computer is turned on each time, the first timing pulse (the one occurring simultaneously with the home pulse) will first be available when the initial home pulse occurs.

For recording the home pulse so that it and the T3 pulse immediately preceding memory location 0 occur simultaneously, the arrangement shown in Figure 17 can be used. Note that the timing pulse applied to the home pulse write amplifier input gate must be a positive pulse. The other two inputs to this gate are left open.

SUMMARY

The systems aspect of memory address selection, as it pertains to the parallel readout magnetic drum for a straight binary computer, the OSTIC, was investigated. It was decided that a versatile INSTRUCTION COUNTER would be needed, in addition to an ADDRESS REGISTER and a WORD ADDRESS COMPARISON CIRCUIT, which compares the word address bits from the currently available drum memory location with those in the INSTRUCTION COUNTER or the ADDRESS REGISTER.

The variable incrementing feature of the INSTRUCTION COUNTER will enable the programmer to minimize the instruction access time for sequences of certain instructions. A new instruction address may be entered into the INSTRUCTION COUNTER from the DATA CHANNEL, for use in branching instructions. The INSTRUCTION COUNTER can also be manually addressed from the computer console.

Suggestions were made pertaining to the signals and logical circuitry needed to make these features operational, with respect to the entire computer system.

The WORD ADDRESS COMPARISON CIRCUIT will produce an EQUAL COMPARISON PULSE whenever the word address of the memory location being sought and that of the drum memory location currently available for read/write operations are the same. This pulse can be used as a gating signal for the following T3 timing pulse to control read/write operations.

The memory address selection system was tested, with satisfactory

results. Additional tests for maintenance and trouble shooting of this system were suggested.

A method was also proposed for initially recording the word address pulses and the home pulse on the drum, using the drum timing pulses and the modified INSTRUCTION COUNTER.

SELECTED BIBLIOGRAPHY

(1). Fike, John Lee, Jr. Philosophy and System Organization of a Small Digital Computer. Stillwater, Oklahoma: 1962

(2). Richards, R. K. Digital Computer Components and Circuits. Princeton: D. Van Nostrand Company, Inc., 1957

(3). Millman, Jacob and Herbert Taub. Pulse and Digital Circuits. New York: McGraw-Hill Book Company, Inc., 1956

Bukstein, Ed. Digital Counters and Computers. New York: Rinehart and Company, Inc., 1960

Flores, Ivan. Computer Logic: The Functional Design of Digital Computers. Englewood Cliffs, New Jersey: Prentice Hall, Inc., 1960

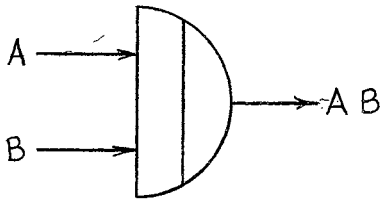
Phister, Montgomery, Jr. Logical Design of Digital Computers. New York: John Wiley and Sons, Inc., 1958

APPENDIX A

GRAPHICAL SYMBOLS

Listed below are the graphical symbols used in the logical diagrams accompanying this paper.

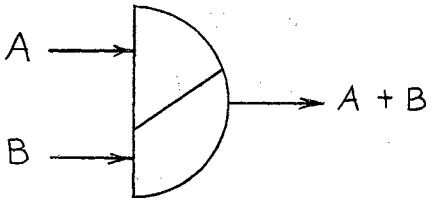
AND gate



A	B	AB
0	0	0
0	1	0
1	0	0
1	1	1

If the positive logic convention is used, the AND gate output will be high only if all of the inputs are high. If A is high, and B is alternately high and low, these pulses will be gated through by A.

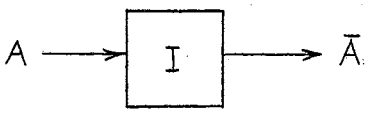
OR gate



A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1

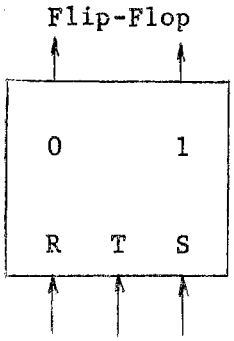
If the positive logic convention is used, the OR gate output will be low only if all of the inputs are low. If A is low, and B is alternately high and low, these pulses will be gated through by A.

INVERTER



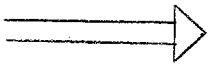
A	$\bar{A}$
0	1
1	0

The complement of the input is produced by the inverter.



The 0 line is the $\overline{\text{output}}$ and the 1 line is the output. R, T, and S denote the reset, trigger, and set input lines, respectively.

Parallel transmission of data is represented as follows.



VITA

Lawrence George Clemens

Candidate for the Degree of

Master of Science

Thesis: A MEMORY ADDRESS SELECTION SYSTEM FOR A MAGNETIC DRUM DIGITAL COMPUTER

Major Field: Electrical Engineering

Biographical:

Personal Data: Born in Ponca City, Oklahoma, February 24, 1939, the son of Harold L. and Bernice A. Clemens.

Education: Attended grade school in Ponca City, Oklahoma; graduated from Ponca City High School in 1957; attended Rice University; received the Bachelor of Science degree from the Oklahoma State University, with a major in Electrical Engineering, in May, 1961; completed requirements for the Master of Science degree, with a major in Electrical Engineering, in May, 1963.

Professional Experience: Employed by Boeing Airplane Company during the summer of 1960, as a student engineer; employed by the General Electric Company during the summer of 1961, as an electrical engineer; employed by the School of Electrical Engineering of the Oklahoma State University from September, 1961 until May, 1962, as a Graduate Assistant.

Professional Organizations: Eta Kappa Nu; Institute of Radio Engineers