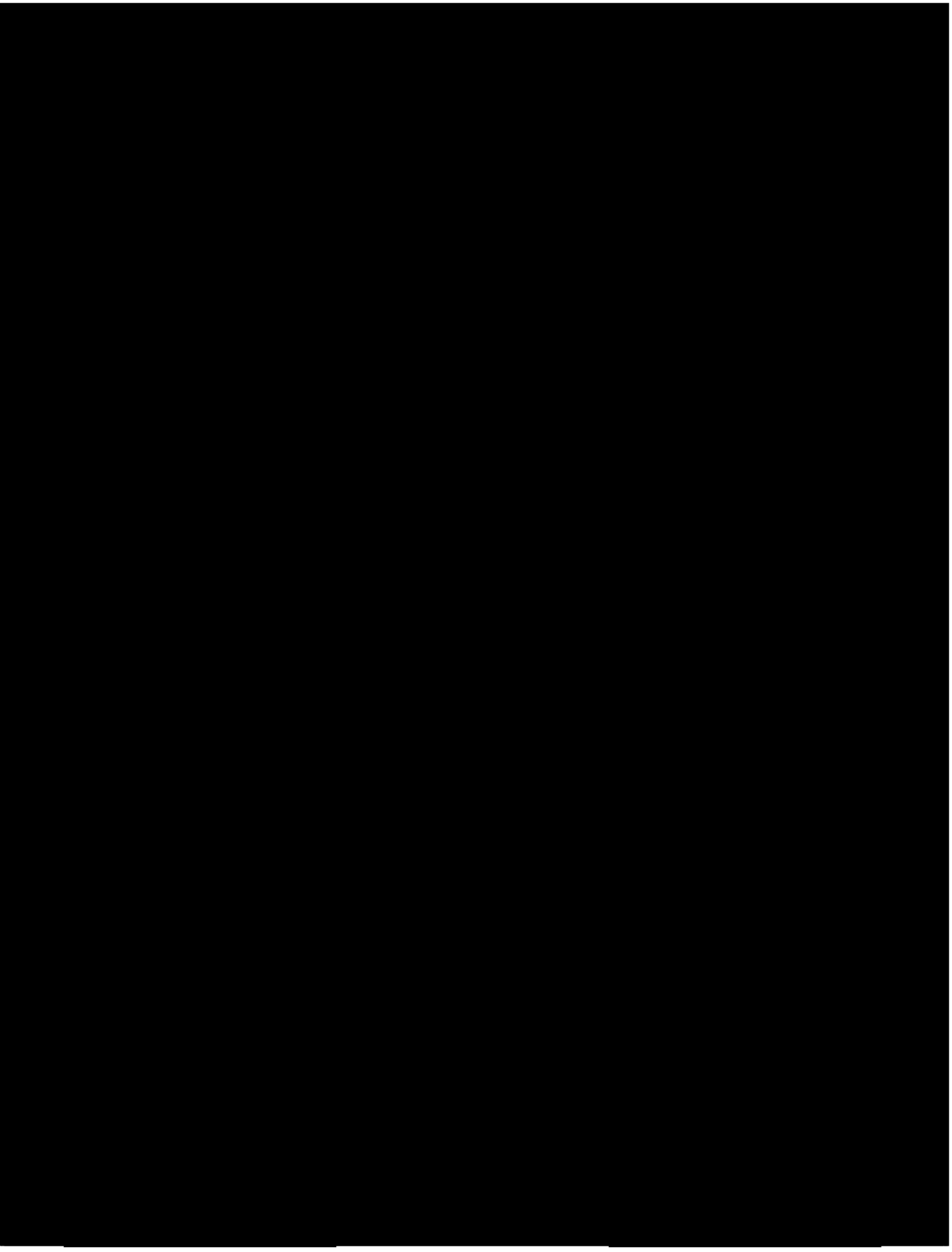


This blank page was inserted to preserve pagination.

A COMPUTER-CONTROLLED GRAPHICAL DISPLAY PROCESSOR

James Gerard Fiasconaro

June, 1970

PROJECT MAC

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Cambridge Massachusetts 02139

This empty page was substituted for a blank page in the original document.

A COMPUTER-CONTROLLED GRAPHICAL DISPLAY PROCESSOR*

Abstract

A cathode-ray tube (CRT) is frequently employed to display text and drawings generated by a digital computer. Unfortunately, all of the commercially available CRT display systems are either very expensive or have limited dynamic capability resulting from the use of some form of storage-type CRT. A need exists to develop a low-cost, relatively sophisticated display processor that can be used with a standard CRT monitor to display computer-generated pictures.

This thesis describes the design, construction and operation of such a display processor. The design was based on the following desired characteristics. The display processor should be able to intensify any one of the 1024 X 1024 raster points of the 10 X 10 inch display area. It should be able to draw straight lines of arbitrary length (up to 1024 raster units along each coordinate). It should be able to display any one of the 96 printable ASCII characters. And finally, it should provide two character sizes and eight intensity levels.

The prototype that was built based on the above characteristics was found to work quite well. The total parts cost for the display processor was only about $3,000. Thus it can be concluded that it was possible to achieve the goal of developing a useful low-cost display processor for a CRT display system.

*This report reproduces a thesis of the same title submitted to the Department of Electrical Engineering, Massachusetts Institute of Technology, in partial fulfillment of the requirements for the degree of Master of Science.

ACKNOWLEDGEMENT

The author wishes to express his sincere appreciation to Mr. John E. Ward for help and suggestions both in the supervision and writing of this thesis. Professor William L. Henke also deserves thanks for his advice and constructive criticism during this project. Thanks are also due to Mr. Albert Vezza for help during the initial phases of this project.

Credit is due to Mr. Roland Hill, Mr. Carlos Fernandez, and Mr. Keith North for their help in the construction of the hardware designed for this project. Credit is also due to Mr. I. E. Rollins for his help in the preparation of the circuit diagrams for this thesis.

Special appreciation is extended to Mrs. Brenda Dunteman for her assistance in the typing of this thesis.

The work reported herein was supported in part by the Air Force Cambridge Research Laboratories, Office of Aerospace Research, under Contract F19628-69-C-0044. This work was also supported in part by Project MAC (an M.I.T. research program sponsored by the Advanced Research Projects Agency, Department of Defense, under Office of Naval Research Contract Number Nonr-4102(01)) through its support of the Display Group of the Electronic Systems Laboratory at M.I.T. Reproduction in whole or in part is permitted for any purpose of the United States Government.

TABLE OF CONTENTS

CHAPTER I	INTRODUCTION	page	1
A.	Computer Graphics		1
B.	Examples of Display Systems		2
	1. The DEC 340		3
	2. The IBM 2250-III		4
	3. The ESL Display Console		5
	4. The ARDS		6
C.	Proposed Problem		8
CHAPTER II	GENERAL DESIGN CONSIDERATIONS		10
A.	Desired System Capabilities		10
B.	Input-Output Characteristics		12
	1. Input Command Formats		12
	2. Control Signals		13
	3. Deflection and Intensification Signals		15
C.	System Block Diagram		16
D.	System Building Blocks		19
CHAPTER III	IMPLEMENTATION AND OPERATION		22
A.	Interface		22
	1. Level Converters		22
	2. Cable Drivers		23
	3. Intensification Signal Clipper		23
B.	Input Register		24
C.	Control Unit		25

D.	Character Generator	page	27
	1. Read-Only Memory		27
	2. Character Generator Control Unit		29
	3. Matrix Generator		30
E.	Master Clock		31
F.	Line Generator		32
	1. BRM Registers		32
	2. BRM Counter		33
	3. Line Generator Control Unit		34
	4. Reset Switch		35
G.	Up-Down Counters		36
H.	Intensity Control		37
I.	Digital-to-Analog Converters		39
CHAPTER IV	SYSTEM PERFORMANCE		41
A.	Installation		41
B.	Sample Picture		44
C.	Display Speed		44
D.	Computer Speed		46
E.	System Improvements		47
CHAPTER V	CONCLUSIONS		49
APPENDIX A	CIRCUIT DIAGRAMS		52
APPENDIX B	DISPLAY PROCESSOR COST		68
BIBLIOGRAPHY			69

CHAPTER I

INTRODUCTION

A. COMPUTER GRAPHICS

The digital computer is undoubtedly the most powerful tool ever invented for computation and manipulation of data. Basic mathematical operations such as addition and multiplication can be performed at rates approaching one million operations per second. Furthermore the accuracy of the computer surpasses human capability by many orders of magnitude. In the relatively few years of their existence, computers have revolutionized many aspects of society and have opened new areas for study and advancement.

The usefulness of computers is highly dependent on the ease with which information can be given to the machine in a form that it can understand, and received from the machine in a form that humans can understand. Most input and output (I/O) done in the early days of computers and a great deal of the I/O done today is either typed or printed on paper, or it is punched on paper tape or cards.

The output, by and large, must be in some printed and/or drawn form to be easily intelligible. Unfortunately, typewriters and line printers have very poor picture drawing capability despite their excellent ability to produce alphanumeric information. Both devices are slow by computer standards, and the typewriter is slow even by human standards. In addition, it is not hard to imagine instances where one would like the output information to be in picture form. Here, both typewriters and line printers are useless.

Solutions to the above problem have taken the form of cathode-ray tube (CRT) display systems.[1] Such systems consist of a CRT, a display processor to control the CRT, a light-pen and/or other forms of graphical input devices, and a typewriter or teletype all under the control of and at the same time controlling the computer. Minimum capabilities for such display systems include the ability to display alphanumeric characters and straight lines on the face of the CRT. The position, size and brightness of the individual characters and line segments that make up the total picture should be controllable. A discussion of the capabilities of several representative display systems will be given in the next section.

First, however, one other attribute of display systems should be considered. This is their ability to produce an interactive environment in which a user may solve his problems. The user can provide input for the computer by typewriter or other input device. The computer then executes the prescribed program and displays the results on the CRT. Upon viewing the results, the user can decide if he wants to change any of the parameters of the problem and recalculate the results. These changes can be easily made with the available input devices or the present results can be accepted without change. A similar situation exists in the case of text editing. Here a page of text is rapidly displayed on the CRT and checked by the user. After making any necessary changes, the user can proceed to the next page. Such a procedure is much faster than editing text that is in printed or typed form.

B. EXAMPLES OF DISPLAY SYSTEMS

To illustrate the capabilities of CRT display systems, several examples will now be discussed.

1. The DEC 340

The first example is the DEC 340 incremental display system.[2] This system is designed to be driven directly by computer output commands and it can display selected points, draw straight lines, and generate alphanumeric characters.

The DEC 340 uses a round CRT with a 16-inch diameter. The usable display area is 9.375 X 9.375 inches and consists of a 1024 X 1024 point raster. (Only raster points can be intensified in the process of producing the resultant picture.) A light-pen is available as an option.

The 340 uses a 7-bit binary rate multiplier (BRM) to generate straight lines as a series of intensified points. Lines are specified by giving the change along the horizontal and vertical axes, ΔX and ΔY respectively, from the present beam location to the other end of the line. The maximum value of ΔX and ΔY that can be specified with a single 18-bit instruction is 255 raster units. One of eight different intensity levels can be specified for each line. Scale factors of 1, 2, 4, or 8 may be applied to the length of the line (the number of points making up the line remains constant). It takes 1365 microseconds for the 340 to draw an 8-inch line.

A stroke-type character generator is used in the DEC 340. Characters are specified by the 6-bit ASCII code, three characters per 18-bit instruction. Normal character size is 0.11 X 0.77 inch with scale factors of 2, 4, and 8 also available. Only one type font is available. The average time required to generate a character is 35 microseconds.

The basic DEC 340 display costs $25,800. A 64-character character generator costs an additional $7,700 and the light-pen option costs $1625. So the cost for this very modest display system is $35,125.

2. The IBM 2250-III

As a second example, consider the IBM 2250 Model III (which includes an IBM 2840 Model II display controller).[3] This display system has a refresher core memory, is capable of absolute and incremental line drawing, includes a light-pen, and has provisions for subroutining. It is more powerful than the DEC 340.

The IBM 2250-III uses a 21-inch rectangular CRT. The usable display area is 12 X 12 inches and consists of 1024 X 1024 raster points. A keyboard and set of 32 function switches are available as options. In addition, the 2250 uses a 16,384-word memory (18 bits per word) to store display commands. This allows the display system to operate independently of the central computer except when it is necessary to change the contents of the memory.

Lines can be drawn by the 2250 in one of two ways. First, the absolute screen locations of the end points of the line can be specified. This requires four memory locations since ten bits are required to specify X and Y at each end of the line. Second, lines can be drawn incrementally. In this case, one memory location contains ΔX and ΔY (six bits each plus sign) for each increment. The longest line that can be drawn this way is 63 raster units in the X and Y directions. In either case, only two intensity levels are available (off and bright).

The time required to draw an 8-inch line is about 70 microseconds. A 0.5-inch line can be drawn incrementally in 15.4 microseconds.

The 2250 uses an analog stroke character generator. Characters are specified by the 8-bit EBCDIC code, two characters per 18-bit instruction. Normal character size is 0.16 X 0.12 inch with scale factors of 1 and 1.5 available. Only one type font is available. The time required to display one character varies from 11.9 to 15.9 microseconds.

The 2250 display unit costs $76,800 (this includes absolute and incremental line capability and the light pen). The alphanumeric keyboard costs $3,600 and the function switches are $7,200. The 2840-II display controller (including the 16,384 word memory) costs $192,000. So the combined system cost is $278,800.

3. The ESL Display Console

The third display system to be considered is the ESL Display Console.[4] This was built by the Electronic Systems Laboratory at M.I.T. and is not available commercially. The ESL Console is a special-purpose computer which automatically converts three-dimensional drawing commands for picture elements and characters into arbitrary two-dimensional projections on the CRT. Real-time rotations, translations, and scale changes of displayed pictures are possible, even in a time-sharing environment. In addition, fully automatic light pen tracking is provided.

The ESL Console uses a 16-inch diameter round CRT with a usable area of 9.375 X 9.375 inches consisting of 1024 X 1024 raster points.

Line generation is incremental, similar to the DEC 340 described previously. System input devices include a typewriter, a teletype, a light pen, a bank of nine decimal switches, two banks of 36 toggle switches, 36 push-button function switches, three seven-bit shaft encoders, and a three-dimensional rate control joy stick (crystal ball). The system was originally designed for use with an IBM 7094 computer.

Line generation and image rotation are controlled by two sets of binary rate multipliers. Images can be rotated in three dimensions in real time by using the crystal ball mentioned above.

Characters can be generated by two different methods. The first method used a Straza Symbol Generator. This can produce 64 different characters matching those of the KSR-35 Teletype. In the second method, the CRT beam is stepped through the 35 points of a 5 X 7 raster by some special hardware. Intensity information is derived from 35 bits of a standard 36-bit long instruction. Using either method, four character sizes are available. Furthermore, character information bypasses the BRM's used for rotation so that characters always remain upright.

It should be obvious that the ESL Console is considerably more powerful than either of the other display systems discussed above. Because it is not marketed commercially, no exact price can be quoted. However, if it were marketed, the cost would probably be on the order of $100,000.

4. The ARDS

The final display system to be discussed is the ARDS (Advanced

Remote Display Station), developed jointly by the Electronic Systems Laboratory and Project MAC at M.I.T. [5] It can draw arbitrary straight lines and generate alphanumeric characters, and has a keyboard and optional joy stick (graphical input device). However, there are a few fundamental differences between the ARDS and the previously mentioned systems.

The first difference is the type of CRT used. The ARDS uses a 8.5 X 6.5 inch rectangular direct-view storage tube. The display area consists of 1081 X 1415 raster points. The use of the storage tube eliminates the problem of flicker that is inherent in standard CRT displays, but it does have its drawbacks. For example, the CRT does not have the capability for selective erasure, and so if a small portion of the picture is to be changed, the entire screen must be erased and the complete new picture drawn. This problem is aggravated somewhat by the second difference between the ARDS and the other displays; the way it is connected to the computer.

The ARDS, as the name suggests, was designed primarily for use in locations remote from the central computer. Typically, it is connected to the computer over narrow-bandwidth telephone lines. As a result, the speed of the display is limited by the speed of this connection. The character rate is 120 characters per second, thus it takes about 33 seconds to display a full screen of 4,000 alphanumeric characters. Thus, if only a few characters are to be changed, it is an inconvenience to have to wait for the entire picture to be reproduced.

The third difference between the ARDS and other display systems is cost. The ARDS terminal, with keyboard and joy stick costs about

$10,000. This is a substantial reduction over all refreshed-type displays. Several aspects of the ARDS hardware make this low price possible. Lines are generated incrementally using a single BRM. The character generator is a 9 X 7 raster type utilizing a read-only memory to store intensity information for each character. And finally, the ARDS produces its deflection voltages by analog integration of standard-shaped current pulses produced by the BRM and the character generator, rather than by up-down counters and D/A converters, as in the DEC 340 and ESL Display Console.

C. PROPOSED PROBLEM

It is easy to see that the problem with computer graphics is not one of capability, usefulness, or variety of hardware but rather one of the cost of the hardware. Given a large amount of money, one can buy any one of a number of sophisticated or not so sophisticated display systems. For quite a bit less money, one can buy an ARDS terminal but in doing so sacrifice the dynamic flexibility of a refresh-type CRT. The problem then was the following. Could a display system be built using a refresh-type CRT such that the finished system would be cost competitive with the ARDS?

The basic characteristics that such a system should have are the following. It should be able to plot arbitrary points on the CRT given the X and Y coordinates of the point. It should be able to draw arbitrary straight lines given the starting point and length along horizontal and vertical axes. Given the starting point, it should be able to produce any of the 96 printable ASCII characters. At least four intensity

levels should be available for points, lines and characters. And finally, it should be fast enough so that reasonably complex pictures can be generated without much flicker.

Such a display processor can and has been built as part of this thesis. Much of the hardware parallels in concept that used in the ARDS. One fundamental difference, however, was the need for faster operation because a refresh-type CRT is used instead of a storage tube. In addition, the display processor was built to interface directly with a small computer (a PDP-9) and not for operation over a telephone line. The remainder of this thesis is a description of the hardware implementation of the display processor, the environment in which it is used, and the capabilities and cost of the system.

CHAPTER II

GENERAL DESIGN CONSIDERATIONS

This chapter describes the overall considerations that led to the final design of the display processor. Included in this is a description of the desired system capabilities and the input-output characteristics of the processor. In addition, a block diagram of the system is given along with a description of the basic hardware building blocks.

A. DESIRED SYSTEM CAPABILITIES

Before any system can be designed, it is necessary to decide exactly what the completed system must do. And in deciding what a system must do it is necessary to know something about the existing systems with which the new system must interact. The display processor has to interact with three other systems. These are a PDP-9 computer, an ITT Model KM-105 CRT, and an existing point-plotting display system.

The existing display system is the DEC Type 34H point-plotting system that is available with the PDP-9 computer.[6] Any one of 1024 X 1024 raster points on the KM-105 CRT can be intensified by executing display instructions which transfer the desired X or Y coordinate from the accumulator to the appropriate D/A converter. The outputs of the D/A converters control the deflection of the CRT beam. Four intensity levels are provided by the system. Points can be plotted at approximately a 30 kHz rate.

The capabilities determined for the display processor include the following. First, the processor must be able to position the CRT beam

to any one of 1024 X 1024 raster points covering a 10 X 10 inch screen area. If the selected point is to be intensified, then the processor must wait long enough for the beam to settle at the new location before producing the intensification pulse. Second, the processor must be able to draw either intensified or unintensified straight lines of arbitrary length (up to 1024 raster units along either coordinate) given the starting point of the line and the length of the line along each coordinate. Third, the display processor must be able to generate all of the 96 printable ASCII characters given the starting point for the character. Fourth, the processor must provide two character sizes and seven unblanked intensity levels. And finally, it must provide a signal which the computer can test to determine whether or not to send a new command to the processor.

With the above capabilities determined, it was necessary to decide whether the display processor would be state oriented or op-code oriented. In a state-oriented system, the execution of each input instruction depends on the state of the system at the time. For example, if the processor were in line-drawing mode, then every input word would be interpreted as the length of a new line. This would continue until the processor was put into a new mode such as character mode where every command would be interpreted as a new character. Note that the same input word may produce a line in one case and a character in another.

In an op-code oriented system, each input word contains an op-code field and this op-code determines what is to be done with the new information. For example, if the op-code field contained the assigned code

for character generation, then the remainder of the input word would be interpreted as the code for the specified character and treated accordingly. In an op-code oriented system, a command that produces a line is always different from one that produces a character.

It was decided that the display processor would be op-code oriented for several reasons. First, for this particular application, there was nothing to be gained by the decrease in input word length that accompanies state-oriented operation. Second, an op-code oriented system is easier to use because commands can be given without regard for the state of the device at the time, i.e., it is not necessary for the computer to keep track of states established by previous instructions. With this decision, an additional requirement is that the processor must also be able to decode the op-code of each input word. More will be said about the input-word formats later.

B. INPUT-OUTPUT CHARACTERISTICS

With system capabilities determined, exact specifications can be given for the input signals and commands which the display processor must interpret and for the output signals which the display processor must produce.

1. Input Command Formats

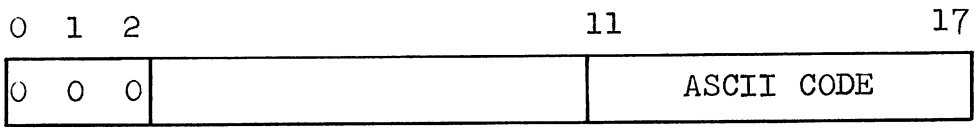
The PDP-9 has an 18-bit word length. All 18 bits can be transmitted in parallel to peripheral devices in two ways. Data is either transmitted from the accumulator by IOT instructions or from memory by special hardware that provides data channel operation. Thus 18-bit words are available to specify all of the input commands which the display processor must interpret.

The left-most three bits of each word contain the op-code. Op-code 000 (CHARACTER) indicates that the rest of the word is to be interpreted as the ASCII code for a character (see Figure 1). Op-codes 001 (LOAD ΔX) and 010 (LOAD ΔY) indicate that the word is to be interpreted as the X or Y component of the length of a line that is to be drawn starting at the present screen location. If the SIGN bit is a 1 the component is considered negative (positive X is to the right on the screen and positive Y is up). If the VIS bit is a 1 then the line will be visible. And if the START bit is a 1, the processor starts generating the line. Op-code 011 (SHORT LINE) is analogous to op-codes 001 and 010 except that both ΔX and ΔY are specified in one word. Op-codes 100 (SET X) and 101 (SET Y) indicate that the word is to be interpreted as the X or Y component of the absolute screen location to which the CRT beam is to be moved (the origin is in the lower left hand corner of the screen). If the WAIT bit is a 1, a 12 microsecond delay is triggered. If the FLASH bit is a 1, the specified point is intensified. And finally, op-code 110 (SET PARAMETERS) allows the user to change the intensity and character size. If $ENABLE_1$ is a 1, the new intensity information is entered. And if $ENABLE_2$ is a 1, the new character size is entered. Op-code 111 is currently unused.

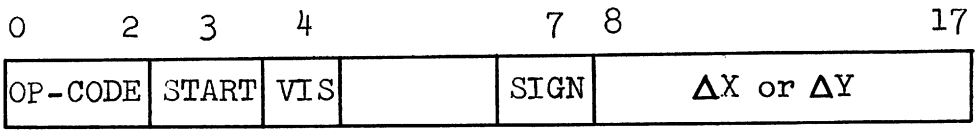
2. Control Signals

A number of control signals must be provided between the PDP-9 and the display processor. The first of these is the IOT4 pulse. This pulse is generated by the PDP-9 at those times when a new input word is available for the display processor. The IOT4 pulse is used to load

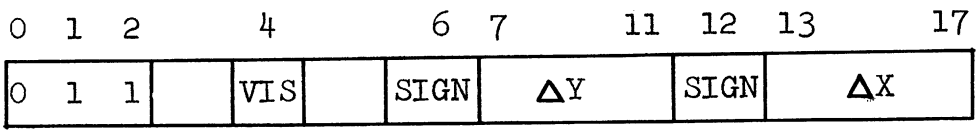
CHARACTER (000)



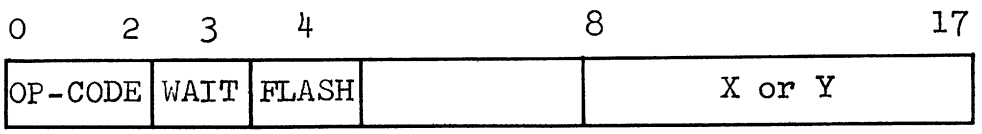
LOAD ΔX (001) or LOAD ΔY (010)



SHORT LINE (011)



SET X (100) or SET Y (101)



SET PARAMETERS (110)

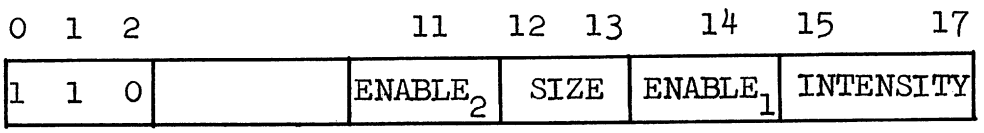


Figure 1. Command Formats

the new input word into the input register of the display processor. Another signal generated by the PDP-9 is the POWER CLEAR signal. This signal is used to reset the display processor when it is first turned on.

The display processor must be able to handle two other control signals in order to maintain compatibility with the existing display system. When the PDP-9 produces a SET X or a SET Y pulse (while executing Type 34H-display commands), the low-order ten bits of the input must be treated as if a SET X or SET Y command had been given along with the IOT4 pulse.

The only control signal that must be generated by the display processor is a WORD REQUEST signal. This signal does one of two things. It either tells the data channel hardware that the display processor is ready to accept a new input word or, if the display is being produced with IOT instructions, it provides the signal which allows the program to skip to the sequence which contains the IOT instruction. Both are functionally equivalent but data channel operation is faster.

3. Deflection and Intensification Signals

In addition to the one control signal produced by the display processor, there are three other output signals which the processor must produce. These are the deflection (X and Y) and intensification signals for the CRT. The KM-105 CRT monitor has a sensitivity of one volt per inch in the X and Y directions, with zero volts corresponding to the center of the screen. Thus the deflection voltage produced by the processor must range between $\pm$ 5 volts. The KM-105 intensity circuit

is such that zero volts will blank the screen and +10 volts will produce a bright spot.

Provision must also be made for the existing intensification signal from the Type 34H display controller. The circuit used to combine the two intensification signals will be described fully in the next chapter.

C. SYSTEM BLOCK DIAGRAM

In light of the preceeding discussion, it is possible to specify the major components of the system in block diagram form. This block diagram is shown in Figure 2, and a brief description of the function of each block is given below. A more detailed description will be given in Chapter III.

The primary function of the interface is level conversion between the -3 to 0 volt DEC logic levels and the 0 to +3 volt logic levels used by the display processor. The interface also includes cable drivers used for the WORD REQUEST, SET X, SET Y, POWER CLEAR, and IOT4 pulses.

The input register is the temporary storage area for each new input word. Other functions of the input register block include decoding each op-code and distributing information to the appropriate block for further processing. For example, if the op-code is 000 then the character generator must receive the ASCII code for the character to be generated.

The control unit has the job of knowing what the rest of the system is doing at all times and requesting that a new word be sent as soon as the input register is available to store it. For example, while the pro-

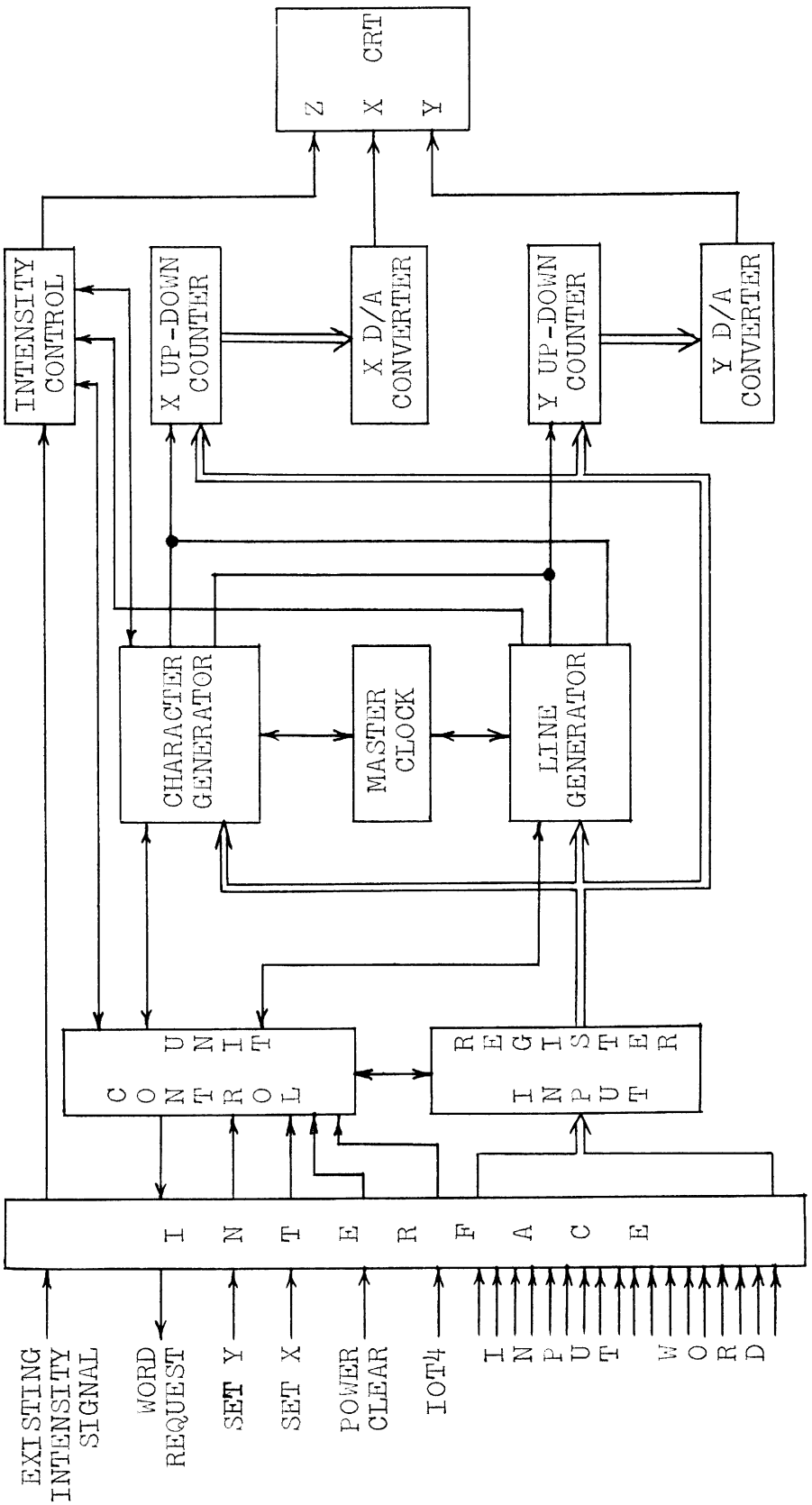


Figure 2. System Block Diagram

cessor is drawing a line the control unit requests a new word. As soon as the line generator signals the completion of the line, the new information that has been waiting in the input register can be sent to the appropriate place and a new request issued by the control unit.

The line generator is a binary rate multiplier (BRM) and a separate control unit. Its function is to send the correct number of pulses to the X and Y up-down counters during the production of each line. In addition, it sends timing signals to the intensification circuit so that the screen can be blanked while the D/A-converter outputs are changing.

The character generator includes a read-only memory, a matrix generator, and a control unit. The matrix generator steps the X and Y up-down counters (and thus the beam) through a 9 X 7 matrix of points for each character. The memory has stored within it the intensity pattern for each character, which is used to control the intensification of the beam during the generation of each character.

The master clock is a variable frequency astable multi-vibrator that controls the rate at which lines and characters are drawn. This includes the circuits that turn on, turn off, and interrupt the production of clock pulses.

The X and Y up-down counters are simultaneous-transition 10-bit binary counters. The counter outputs go to 10-bit digital-to-analog (D/A) converters that control the X and Y deflection of the CRT.

The intensification circuit produces the appropriate intensification pulses during the production of lines and characters, and while plotting individual points. In addition, it combines this signal with the inten-

sification signal of the existing system to maintain compatibility between the two systems.

D. SYSTEM BUILDING BLOCKS

Now that the system has been described from a functional point of view, it is possible to consider the individual components from which the system was built.

The majority of the system was built with Texas Instruments (TI) series 74 and 74H TTL logic in dual-in-line plastic packages. This was chosen for a number of reasons. TTL logic is low in cost, and it has high speed capabilities, good noise immunity, and high reliability. In addition, a wide variety of gates and flip-flops are available, providing great flexibility in the design of each specific circuit.

The dual-in-line packages are mounted on printed circuit boards that hold up to 16 packages (14 or 16 pins) and have 43-pin connectors. Connections on each board are made between Vector terminals. Connections between printed circuit board connectors are wire wrapped.

In addition to the logic gates and flip-flops, two other useful circuits are the astable and monostable multivibrator. These circuits were constructed using series 74 logic elements to facilitate mounting with other logic circuits and to insure electrical compatibility among the various elements.

The circuit for the astable multivibrator is shown in Figure 3. The circuit oscillates at about 12 MHz without the capacitors and the resistor. With the circuit as shown, the rate is variable between 10 MHz and 5 MHz with a pulse width of 30 nsec. This is the circuit used in the master clock of the display processor.

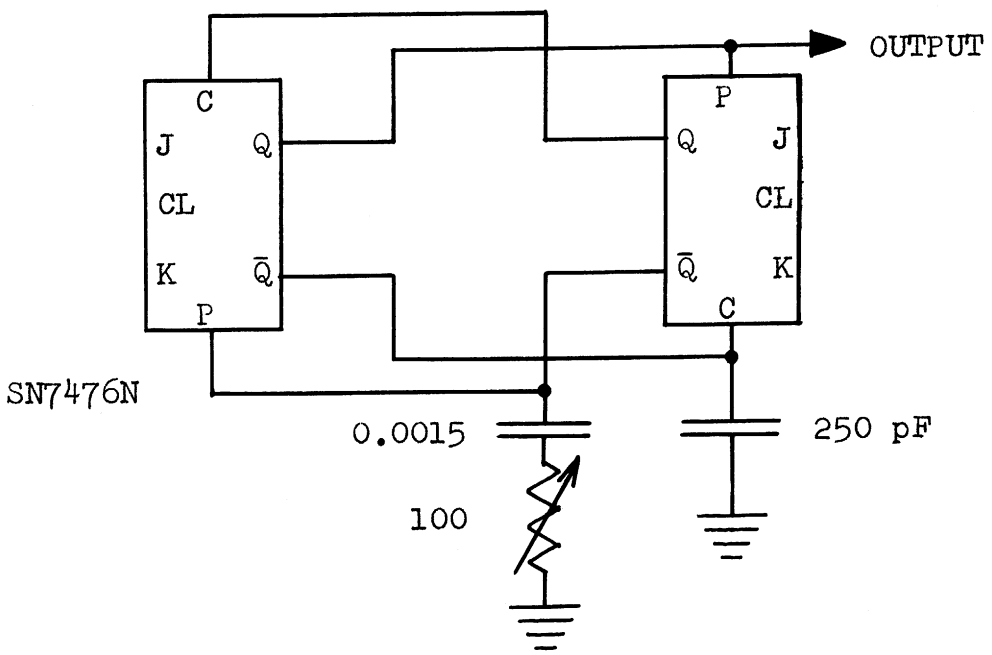


Figure 3. Astable Multivibrator

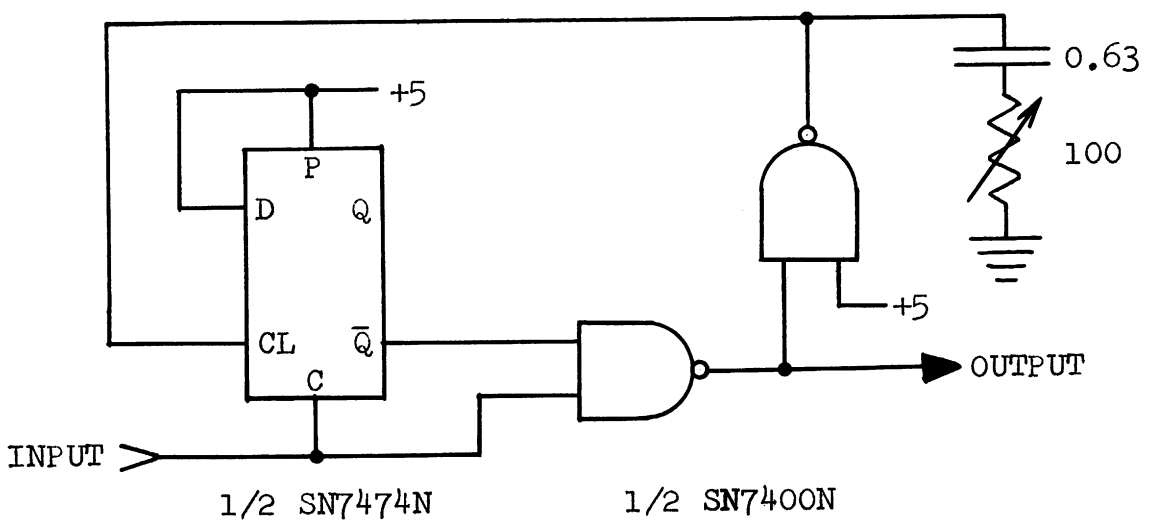


Figure 4. Monostable Multivibrator

The circuit for the one-shot is shown in Figure 4. The one-shot is triggered by a 0 to 1 transition and reset by a 1 to 0 transition. The width of the output pulse is variable over a wide range. The width is about 30 nsec without the resistor and capacitor. With the circuit as shown, the width can be varied out to about 12 microseconds.

Certain portions of the system (level converters, cable drivers, and D/A converters) were built with discrete components. This resulted from unavailability and/or high cost of off-the-shelf equivalent circuits. The circuits used will be described fully in the next chapter.

CHAPTER III

IMPLEMENTATION AND OPERATION

This chapter describes the implementation and operation of the display processor. The description is structured around the system block diagram given in Figure 2. Diagrams of the specific circuits used to implement each block are given in Appendix A.

A. INTERFACE

The interface performs three functions. First, it converts DEC -3 to 0 volt logic levels to TTL 0 to +3 volt logic levels, and vice versa. Second, it includes cable drivers for those signals that must be transmitted between the PDP-9 and the display processor without ringing. And third, it converts the existing CRT intensification signal to a form usable by the intensity control circuit of the display processor.

1. Level Converters

The circuit used to convert DEC logic levels to TTL logic levels is shown in Figure A1 (Figure 1 of Appendix A). The emitter-follower configuration is used to insure that the level converter has high input impedance. The ringing caused by this cable termination can be tolerated on the 18 INPUT WORD lines because the new data appears on the lines approximately two microseconds before it is loaded into the input register by the IOT4 pulse. By that time, all ringing has disappeared.

The fan-out of the DEC to TTL level converter is an important consideration. The converter must sink 1.6 mA of current at zero volts output for each TTL load that the converter drives. But to insure low

power drain on the -15 volt supply, each of the 22 converters used should be operated at as low a supply current as possible. Therefore, the converter is designed to drive one TTL load. In those cases requiring higher fan-out, use of a TTL inverter (or pair of inverters) is better than redesigning the converter for increased fan-out. Inverters are used with the SET X, SET Y, and IOT4 pulses to increase fan-out. The inverter used with the POWER CLEAR signal is necessary for inversion only and not for increased fan-out.

The TTL to DEC level converter is also shown in Figure A1. This circuit is used in only one application, the WORD REQUEST signal. The only non-obvious constraint on the design was the choice of power supply voltages. The converter is located in the PDP-9 and so the circuit was designed to operate with +10 and -15 volt supplies available in the PDP-9.

2. Cable Drivers

The circuits for the two types of cable drivers used are shown in Figure A1. In both of the circuits, either one or the other of the output transistors is always in saturation. This allows the circuit to source current at the more positive output level and to sink current at the less positive output level. Both circuits can drive 100 nsec-wide pulses over terminated cable.

3. Intensification Signal Clipper

The CRT intensification signal of the existing display system is a pulse varying from -15 volts (screen blanked) to +7 volts (screen unblanked). The duration of the pulse determines the intensity of the

spot on the screen. The easiest way to accomodate this signal in the display processor was to clip the pulse at ground and +5 volts to make it TTL-compatible and use the resulting pulse in the intensity control circuit that will be described later in this chapter. The clipping circuit is shown in Figure A1.

B. INPUT REGISTER

The input register is the temporary storage location for each input word received from the PDP-9. It consists of 18 D-type flip-flops. The six most significant bits (MSB's) are included in Figure A2 and the remaining bits are included in Figure A3.

D-type flip-flops have a definite advantage over J-K flip-flops for this application. If J-K flip-flops had been used, one of the following undesirable alternatives would have been required. First, 36 signals (J and K input for each flip-flop) could have been brought from the PDP-9 to the display processor, thus doubling the number of level converters required. Second, a set of 18 TTL inverters could have been used with the 18 lines to generate the complement of each line for the J and K inputs. Or third, each input line could have been brought to a J input and all of the K inputs grounded, thus requiring that the register be cleared before accepting each new input word. With D-type flip-flops, each of the 18 inputs is simply connected to the D input of the appropriate flip-flop.

The operation of the input register is straightforward. The CLOCK inputs of all the flip-flops are connected to the IOT4 pulse, and new data is loaded into the register on the leading edge of the IOT4 pulse.

The output of each bit of the input register is wired to the input of the appropriate bit in each of the other registers in the display processor.

A number of other functions that are closely associated with the input register are performed by circuits implemented with the input register. The first of these is the op-code decoder. It is included in Figure A2 and is the straightforward application of eight three-input AND gates. After the op-code is decoded, it is necessary to transfer the data to the appropriate section of the display processor. This is done by directing a pulse to the CLOCK inputs of the appropriate register through the set of NAND gates and buffers shown in Figure A2. The J-K flip-flops shown in Figure A2 are used to store the VIS and FLASH bits of the input word.

It is obvious from Figure 1 that the five least significant bits (LSB's) of ΔY, the sign of ΔY, and the sign of ΔX can be in two different places in the input register depending on the op-code. The choice is made by the set of two-wide two-input AND-OR (or equivalently NAND-NAND) gates shown in Figure A3.

The final circuit implemented with the input register is the three-bit intensity register shown in Figure A3. From a functional point of view, this register could have been located on the intensity control circuit card. However, after the essential parts of the intensity control circuit were implemented there was no room on that printed circuit board for the register. Therefore, it was placed with the input register.

C. CONTROL UNIT

The control unit supervises the operation of the display processor. The implementation is shown in Figure A4 and is best explained in terms

of the operation of the circuit. Each IOT4 pulse received from the PDP-9 sets a flip-flop in the control unit to 1. This has two effects. First, the WORD REQUEST signal is turned off. And second, a data transfer pulse is generated as soon as the display processor finishes what it was doing when the IOT4 pulse arrived. This transfer pulse is directed to the appropriate register through the gates in Figure A2 that were described earlier. In addition, the transfer pulse clears the flip-flop that the IOT4 pulse had set to 1, thus turning on the WORD REQUEST signal.

The data transfer pulse is also used to turn on the character generator, the line generator, and the set-point delay circuit. If the op-code of the input word is 000 (CHARACTER) then the character flip-flop is set to 1, turning on the character generator and blocking the production of data transfer pulses until the character generator is turned off. If the op-code is 011 (SHORT LINE), or if the op-code is either 001 or 010 (LOAD ΔX or ΔY) and the START bit of the input word is a 1, then the line flip-flop is set to 1, turning on the line generator (after a delay of 150 nsec which will be explained later) and blocking production of data transfer pulses. If the op-code is either 100 or 101 (SET X or Y) and the WAIT bit of the input word is a 1 then the set-point flip-flop is set to 1, triggering a 12-microsecond delay and blocking the production of data transfer pulses. This delay is the maximum time required for the CRT beam to settle in a new location after a large step is applied to the deflection inputs. The termination of the delay signals the intensity control circuit to intensify the spot (if the FLASH bit is a 1).

The final function of the control unit is to transfer the ten LSB's of the input word to the appropriate up-down counter when SET X or SET Y pulses are received. This is done in two steps. The input register is loaded on the leading edge of the SET X or SET Y pulses. On the trailing edge of the pulses, a one-shot is triggered to transfer the data to the appropriate up-down counter. These transfer pulses are ORed with the transfer pulses that would result from executing a SET X or SET Y command. The SET X and SET Y pulses also control the WORD REQUEST signal, setting it to 1 until the transfer has been completed.

D. CHARACTER GENERATOR

As mentioned previously, characters are produced by intensifying selected points of a 9 X 7 matrix. The major components of the character generator are a read-only memory (with its input and output registers), a matrix generator, and a control unit. Each of these component parts will be considered separately.

1. Read-Only Memory

The intensity pattern for each character is stored in a 10,240-bit (1024 words and 10 bits per word) read-only memory. The memory used was manufactured by Memory Technology, Inc. and is identical to the memory used in the first models of the ARDS. The memory requires 11 input signals. Ten of these are the complement of the desired memory address and the eleventh is a cycle-initiate signal (1 to 0 transition). The ten outputs produced by the memory are the contents of the addressed location.

The organization of the data in the memory is best understood by considering the ten-bit memory input register shown in Figure A5. The three LSB's of the register are a three-bit binary counter. The remaining seven bits serve simply as storage. Initially, the entire register is cleared. Next, the ASCII code for the desired character is loaded into the seven MSB's of the register and a cycle-initiate signal is produced. The memory output will contain the nine intensity bits of the first (left hand) column of the 9 X 7 matrix (starting at the top of the column) and one bit to indicate whether or not the matrix should be "dropped" (i.e., lowered two raster units) for lower case letters such as g, p, and q which extend below the normal bottom row of the raster. The second column of the matrix (starting at the bottom) is obtained by incrementing the three-bit counter by one and issuing another cycle-initiate signal. The counter is incremented again and another initiate signal given to obtain the third column, and so on.

At the appropriate times during the generation of a character (i.e., before each column) the memory output is loaded into the memory output shift register shown in Figure A5. As the column is produced, this register is shifted to the right and the output of the right-most bit is used to control the intensity of the CRT beam. Zeros are shifted into the register from the left so that it will be ready to accept the pattern for the next column of the matrix at the completion of the present column.

An exception to the standard procedure for generating a character is made if the ASCII code for a space is received. In this case the beam is not stepped through the entire matrix, but is moved directly to the starting location of the next character to save time.

2. Character Generator Control Unit

The control unit for the character generator is given in Figure A6. Again, the implementation is best understood in terms of the operation of the circuit.

When the character generator is turned on by the control unit, a one-shot immediately generates a pulse to clear the memory output shift register, issue a cycle-initiate pulse to the memory, set the X up-down counter so that it will count up, and trigger a 300-nsec delay (the access time for the memory). At the termination of the delay, the memory output is loaded into the memory output shift register and a pulse train is requested from the master clock.

If the matrix drop bit is a 0, the pulse train goes immediately to the matrix generator. If the drop bit is a 1, two pulses are sent to the Y up-down counter to shift the matrix down by two raster units before the character is generated. These two pulses are counted off by a simple two-bit binary counter.

After the complete matrix has been generated, the matrix generator sends a matrix-done signal to the character generator control unit. Upon receiving this signal, the character control unit sends two pulses to the X up-down counter to position the beam for the next character (two-unit intercharacter spacing). In addition, if the matrix was dropped at the start of the character, two pulses are sent to the Y up-down counter so that the beam will be on the same horizontal line that it was on prior to the generation of the character. The pulse train from the master clock is then turned off and a one-shot is triggered. This last

pulse is used to clear the matrix generator, clear the memory input register, and signal the control unit that the character is done.

3. Matrix Generator

The matrix generator (shown in Figure A7) consists primarily of mod-8 and mod-9 counters which break up the pulse train received from the master clock into one pulse train for the X up-down counter and one pulse train for the Y up-down counter. The first eight pulses decrement the Y counter, the next pulse increments the X counter, the next eight increment Y, the next one increments X, and so on. Note that each pulse that increments X also complements the direction of the Y up-down counter (which is initially set to down).

The matrix generator also controls the cycle-initiate signals and the pulses that increment the memory input register. The first pulse of each set of eight pulses that goes to the Y up-down counter increments the memory input register. The second pulse of each set of eight pulses is used to produce the cycle-initiate signal. While the matrix generator completes the rest of the column of the matrix, the memory is busy fetching the pattern for the next column.

The final job of the matrix generator is to control the loading and shifting of the memory output shift register. Two (alternating) signals from the intensity control unit are used to insure that loading and shifting are done while the screen is blanked. More will be said about these signals later but, briefly, what is done is the following. Initially, a flip-flop is set so that the first signal will produce a shift. The second signal loads the appropriate new value into the flip-flop so that

at the next occurrence of the first signal, loading or shifting will be carried out as required. This procedure continues throughout the generation of each character.

E. MASTER CLOCK

The primary function of the master clock (shown in Figure A8) is to supply pulse trains upon request to the line generator and the character generator and in so doing to control the rate at which lines and characters are produced.

The heart of the master clock is the astable multivibrator discussed in Chapter II (see Figure 3). The pulse rate is continuously adjustable between 10 MHz and 5 MHz. In addition, the rate can be divided by 1, 2, 4, or 8 by appropriately setting the toggle switches which control the frequency divider shown in Figure A8. Each stage of the frequency divider either allows every pulse to pass on to the next stage or allows every other pulse to pass depending on the switch setting.

Requests for pulses can arrive at the master clock at any time. If a request arrived during a clock pulse and was honored immediately, the first pulse might be very narrow and might produce errors in other parts of the display processor. To avoid this problem, the requests from the character and line generators and the clock pulses are combined in the synchronizer shown in Figure A8. The first clock pulse that occurs after (or possibly when) the request is made is used to set the appropriate synchronizer flip-flop. The next clock pulse that is produced will be the first pulse received by the device making the request. When the request signal returns to 0, the pulse train is dis-

continued. It is the job of the device making the request to see that this does not occur during a clock pulse.

The final part of the master clock is the interrupt control. This circuit detects when an attempt is made to draw a line extending past an edge of the 1024 X 1024 raster point area. When this happens, the up-down counter goes from all 1's to all 0's (or vice versa) and the clock is interrupted long enough for the beam to move to and settle at the opposite edge of the display area. After the delay, everything continues normally. The effect that this has on the displayed picture is called wrap-around. It is the simplist way to allow the user to see what he has drawn off the edge of the display area (so that he can make corrections, if desired).

F. LINE GENERATOR

As mentioned previously, the line generator consists of a dual 10-bit binary rate multiplier (BRM) and a control unit. The basic function performed by a BRM is the following. Consider the contents of a ten-bit register to be a binary integer N ($0 \leq N \leq 1023$). If a pulse train of 1023 pulses is applied to the input of the BRM, only N pulses will appear at the output of the BRM. So if the register contains ΔX (or ΔY), the number of BRM output pulses will equal ΔX (or ΔY). The BRM used in the display processor consists of two ten-bit registers (one for ΔX and one for ΔY), two BRM gating networks, and a single ten-bit binary counter with its associated logic networks.

1. BRM Registers

Each register used in the BRM is made of ten J-K flip-flops. The eight LSB's are shown in Figure A9 along with a portion of the BRM

gating. The remaining two bits of the registers are implemented with the BRM control unit and are shown in Figure A10. The registers were divided in this way simply because the full ten bits would not fit on one printed circuit board.

It is necessary to control the loading of the five LSB's of each register separately from the five MSB's because of the short line command format. In the case of a short line, only the five LSB's are altered. The five MSB's remain unaltered and each bit is a zero because the entire register is cleared after each line is drawn.

2. BRM Counter

The counter used in the BRM (see Figure A11) is a straight-forward ten-bit simultaneous-transition binary counter that counts the clock pulses applied to the BRM. However, it does have one special feature for this particular application. Up to nine of its LSB's can be set to 1 prior to the start of the count, and those bits that are set to 1 will not receive count pulses after the counting has begun. This permits the counter to be "shortened" from a ten-bit counter to a (10 - I)-bit counter, where I is the number of bits set equal to 1.

The reason for doing this is the following. The time required to draw a line is essentially equal to the time required for the BRM counter to count through, and would normally be equal to 1023 pulse times, independent of line length. However, since the values of ΔX and ΔY are known before counting starts, the counter can be shortened so that fewer than 1023 (but greater than or equal to the larger of ΔX and ΔY) pulses are needed, so that the time required approximates the line length.

The procedure for shortening the counter is the following. If the MSB of ΔX and ΔY is 0, then the LSB of the counter is set to 1. If, in addition, the next MSB of ΔX and ΔY is 0, then the next LSB of the counter is also set to 1, and so on. Thus, if the larger of ΔX and ΔY is a B-bit binary number (excluding leading zeros) then only $2^B - 1$ pulses are necessary to increment the counter up to 1023. This function if performed by the initializing gating shown in Figure A11. The worst-case time required for a signal to propogate down the series of AND gates is about 100 nsec. It is primarily for this reason that the line generator is not turned on until about 150 nsec after the line flip-flop in Figure A4 is set to 1.

3. Line Generator Control Unit

The control unit for the line generator is shown in Figure A10. As soon as the line generator is turned on, it triggers a one-shot, which produces the pulse used to initialize the BRM counter. This pulse is also used to set the flip-flop that requests a pulse train from the master clock. The pulse train from the master clock is used to generate two pulse trains, namely count pulses which increment the BRM counter, and BRM input pulses (i.e. selected ones of these pulses will appear at the output of the BRM).

Operation of the BRM is based on the following algorithm. Consider a ten-bit register and a ten-bit counter. Let bit 0 represent the LSB and bit 10 represent the MSB. Then an output pulse should be produced if bit N ($0 \leq N \leq 10$) of the register is a 1 and the (10-N)th bit of the counter has just made a 0 to 1 transition. The number of pulses in the output pulse train will equal the value of the binary integer in the register.

This algorithm is implemented in the following manner. If the Nth bit of the register (ΔX or ΔY) is a 1 and the (10-N)th bit of the BRM counter is a 0 and all of the counter bits less than (10-N) are 1, then a BRM input pulse is allowed to pass through the BRM gating to the output of the BRM. The following observation shows that this implementation produces the desired result. If the (10-N)th bit of the counter is a 0 and all of the bits less than (10-N) are 1, then the next count pulse will produce a 0 to 1 transition in the (10-N)th bit and so the next BRM input pulse (which occurs simultaneously with the next count pulse) should and does appear at the output. Each BRM output pulse is ORed with the corresponding (X or Y) output pulse from the character generator and the result is applied directly to the input of the appropriate up-down counter.

The pulse train from the master clock is turned off by the clock pulse that occurs when the MSB of the BRM counter is a 1 and all of the other BRM counter bits are also 1 (i.e. the J-K input to the MSB is a 1). This last pulse also triggers a one-shot which clears the BRM registers and the BRM counter and tells the display processor control unit that the line is done.

4. Reset Switch

A push-button reset switch is provided on the front panel of the display processor (as shown in Figure A10). Depressing this switch initializes all relevant flip-flops in the display processor to the correct "cold start" value. This button can be used in place of the POWER CLEAR signal or as a general purpose panic button.

G. UP-DOWN COUNTERS

The digital representation of the location of the CRT beam is stored in the X and Y up-down counters (see Figure A13). Each counter is a simultaneous-transition, binary counter that can count either by one or by two. The output of each bit of each counter goes directly to the D/A converter that controls the deflection of the beam.

There are two ways to change the contents of the up-down counter. The first of these is to jam transfer a new value into the counter. This is done with the set of NAND gates shown in Figure A12. The outputs of the gates go to the PRESET and CLEAR inputs of the flip-flops making up the counter. When a load pulse occurs (anytime a set-point command or pulse is received) the ten LSB's of the input register are transfered to the up-down counter. The old value of the counter is lost.

The second way to change the contents of a counter is by producing a count pulse. The effect of this pulse is determined by the states of the two flip-flops shown in Figure A12. If the UP output is a 1, then the counter will count up. If the BY2 output is a 1, then the counter will count by two instead of by one.

The operation of the counter utilizes the fact that if the J and K inputs of a flip-flop are both 0, then the output is not changed by a clock pulse. But if the J and K inputs are both 1 then the output is complemented by a clock pulse. If the Q outputs of the flip-flops are considered to be labeled from Q_1 to Q_{10} (Q_1 is the LSB) then the J-K input of (N + 1)st flip-flop is given by

$$(Q_1 * Q_2 * Q_3 * \ldots * Q_N * \mathrm{UP}) + (\overline{Q}_1 * \overline{Q}_2 * \overline{Q}_3 * \ldots * \overline{Q}_N * \mathrm{DOWN})$$

where * is logical AND and + is logical OR. Clearly, each flip-flop will make the proper transition at each count pulse.

The particular arrangement of gates used to obtain the J-K inputs for each flip-flop was dictated by speed considerations. The longest propogation path is eight gates or about 50 nsec (from the output of the LSB to the input of the MSB). If the count-pulse width is held to the minimum allowable (20 nsec), the counter will count at a 10 MHz rate (the design objective). Operating within the display processor, the maximum counting rate is about 9 MHz. This is still quite a bit faster than the CRT can handle but it does mean that the display speed will not be limited by the digital hardware.

In order to make the counter count by two, it is necessary to make the nine MSB's think the LSB is a 1 if the counter is counting up and a 0 if the counter is counting down. The LSB remains unchanged at whatever value it happens to have while the counter is counting by two.

H. INTENSITY CONTROL

The circuit used to control the intensity of the CRT beam is shown in Figure A14. It consists essentially of four subsections. There are three timing circuits that control the width of the intensification pulse for points, lines, and characters. And there is a three-bit D/A converter which controls the height of each intensification pulse.

As mentioned previously, the display processor control unit signals the intensity control unit approximately 12 microseconds after a new set point value is loaded into either the X or Y up-down counter. This signal triggers a one-shot in the intensification circuit. The width of

the output pulse of the one-shot determines the width of the intensification pulse for the point. The trailing edge of this pulse triggers another one-shot which signals the control unit that the set-point is done.

The timing circuit for lines is more complicated than the one for set-points because the pulses in the output pulse train of the BRM are not necessarily evenly spaced. So the following procedure is used to intensify the points of a line.

Each BRM pulse that is sent to the BRM gating networks is also sent to the intensity control circuit. Each pulse does two things. First, it triggers a one-shot whose output pulse width determines the length of time that the beam will be blanked between BRM pulses. And second, each BRM pulse sets a flip-flop to zero. If, however, a count pulse is received by either the X or Y up-down counter, this flip-flop is set to 1 by the count pulse, overriding the effect of the BRM pulse. The trailing edge of the one-shot pulse is used to load the value of the flip-flop into a second flip-flop whose output now determines if the beam will be blanked or unblanked. In this way, all points in the line will be equally bright. If an invisible line is to be drawn, this process continues but the 0 level from the VIS bit overrides everything and no intensification pulses are produced. Similarly, during wrap-around interrupts, the intensity output is forced to zero to blank the screen.

The beam intensity for characters is controlled by the pulse train received by the matrix generator. Each pulse turns off the beam for a controllable amount of time while the D/A output changes value. In addition, the leading edge of the one-shot output pulse which turns off the beam also signals the matrix generator to shift or load the memory

output shift register. The trailing edge of the one-shot pulse triggers another one-shot whose output pulse is used by the matrix generator to load the appropriate value into the flip-flop that controls the loading and shifting of the memory output shift register. Because the intensity circuit performs this function, the memory output shift register will be changed only while the screen is blanked. The screen is unblanked only if the character generator has requested a pulse train from the master clock, and the matrix generator is on, and the output of the memory output shift register is a 1, and the output of the one-shot that controls the width of the intensification pulse is a 1.

The three controlling signals for points, lines, and characters are ANDed together and then ANDed with the existing intensification signal. The resulting signal controls the DTL NAND gate whose output determines the height of the intensification pulses. If each of the controlling signals is a 1, the output transistor is in saturation and the intensification signal is effectively zero volts. If any one of the controlling signals goes to 0, then the output transistor comes out of saturation and the output voltage is determined by the output voltage of the three-bit D/A converter that is controlled by the intensity register. The height of the intensification pulse can vary from nearly zero volts to 10.5 volts in seven 1.5 volt steps.

I. DIGITAL-TO-ANALOG CONVERTERS

The outputs of each up-down counter are converted to an analog deflection voltage by a ten-bit D/A converter (shown in Figure A15). The major parts of the D/A converter are the switches, the resistor ladder network, the reference voltage regulator, and the output op-amp.

Each of the switches in the D/A converter is a DTL NAND gate designed to be compatible with TTL logic circuits and represent one standard TTL load. The output of the switch is either $V_{CE(SAT)}$ of the transistor (0.2 volts) or V_{REF} (the reference voltage) plus V_D (the voltage drop across a silicon diode).

The reference voltage is supplied by a Fairchild uA723 voltage regulator. With the configuration shown in Figure A15 the reference voltage can be varied from 7.2 to 8.1 volts.

The analog output is produced by an R-2R ladder network. The value 1000 Ohms was chosen as a compromise between speed (i.e. the time constant associated with resistors of the ladder network and stray capacitance of the circuit) and loading (i.e. the current drawn by the ladder network when the ladder input voltage is $V_{REF} + V_D$). The value 1995 Ohms was chosen to compensate for the output impedance of the transistor in saturation and the output impedance of the voltage regulator. For the configuration shown, the impedance seen by each switch looking into the ladder network is the same and equal to 3000 Ohms.

The output of the ladder network is added to an offset voltage at the summing node of the op-amp. This offset voltage is necessary because the CRT is designed for deflection signals between -5 volts and +5 volts. The output of the op-amp provides the deflection voltage for the CRT.

CHAPTER IV

SYSTEM PERFORMANCE

The two previous chapters described the specification, design, implementation and operation of the display processor. This chapter describes the performance of the display processor as a part of the total system that includes the PDP-9 computer and the ITT Model KM-105 CRT. In addition, some possibilities for improving the performance of the display processor are discussed.

A. INSTALLATION

Photographs of the installation of the display processor are given in Figures 5 and 6. The power supplies for the display processor are mounted on the panel at the top of the 19-inch rack. The rest of the hardware (with the exception of the cable drivers and level converters located in the PDP-9) is contained in the single cardfile installed just above the CRT. The read-only memory for the character generator is a single large card mounted on the shelf that is suspended from the cardfile.

The low volume of the display processor hardware is due mainly to the use of integrated-circuit logic elements. The use of such circuits allows considerably higher circuit density than is possible with circuits constructed of discrete components. Circuit density was an important consideration in the choice of the form of those circuits of the display processor that are made of discrete components. An example of this is the DEC to TTL level converter (shown in Figure A1). Two of these

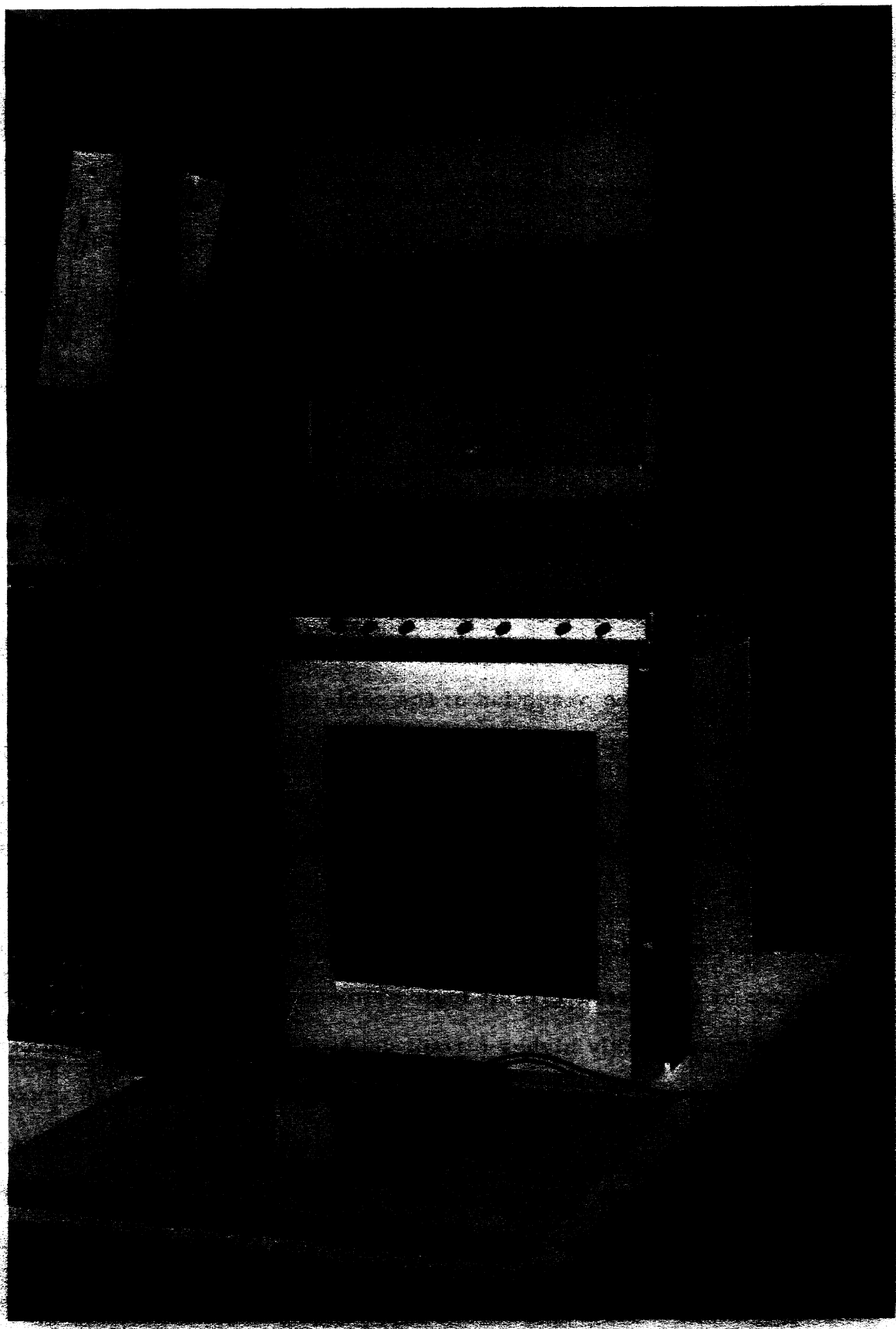


Fig. 5 Front View of Installation

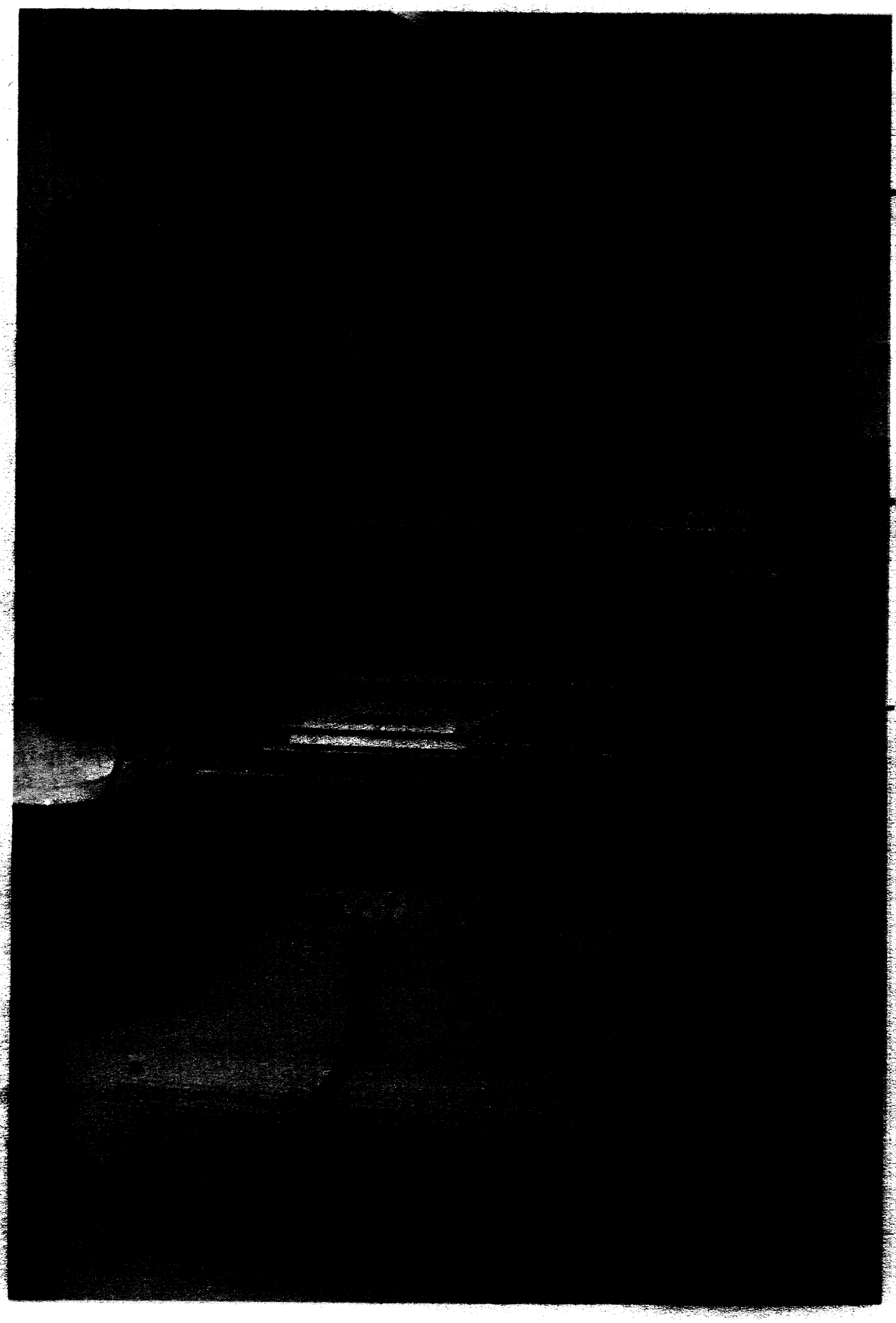


Fig. 6 Rear View of Installation

circuits can be mounted in the space required by a 16-pin dual-in-line package. A total of 24 of these circuits plus several other circuits are mounted on a single printed circuit board.

B. SAMPLE PICTURE

A photograph of a picture generated by the display processor is shown in Figure 7. The curved lines in the photograph are actually made from many short, straight-line segments.

C. DISPLAY SPEED

The speed at which pictures can be generated is limited by the slowest device in the total display system. In this case, the slowest device is the CRT. It was found experimentally that the individual points of lines and characters could not be generated faster than 1.4 microseconds per point. At this rate, normal-size characters are quite legible. However, the legibility of the double-size characters is reduced slightly because the individual dots making up the characters are not in the desired locations. This is caused by the slow response of the CRT to a step input corresponding to moving the beam two raster units.

The time required to produce a character is determined primarily by the period of the pulse train produced by the master clock and the total number of steps in the matrix. At 1.4 microseconds per step, the total time required to generate any character (except a space) is 105 microseconds. As pointed out previously, the entire matrix is not generated for a space, and the beam is simply moved to the starting location of the next character. The time required to produce a space is 20 microseconds.

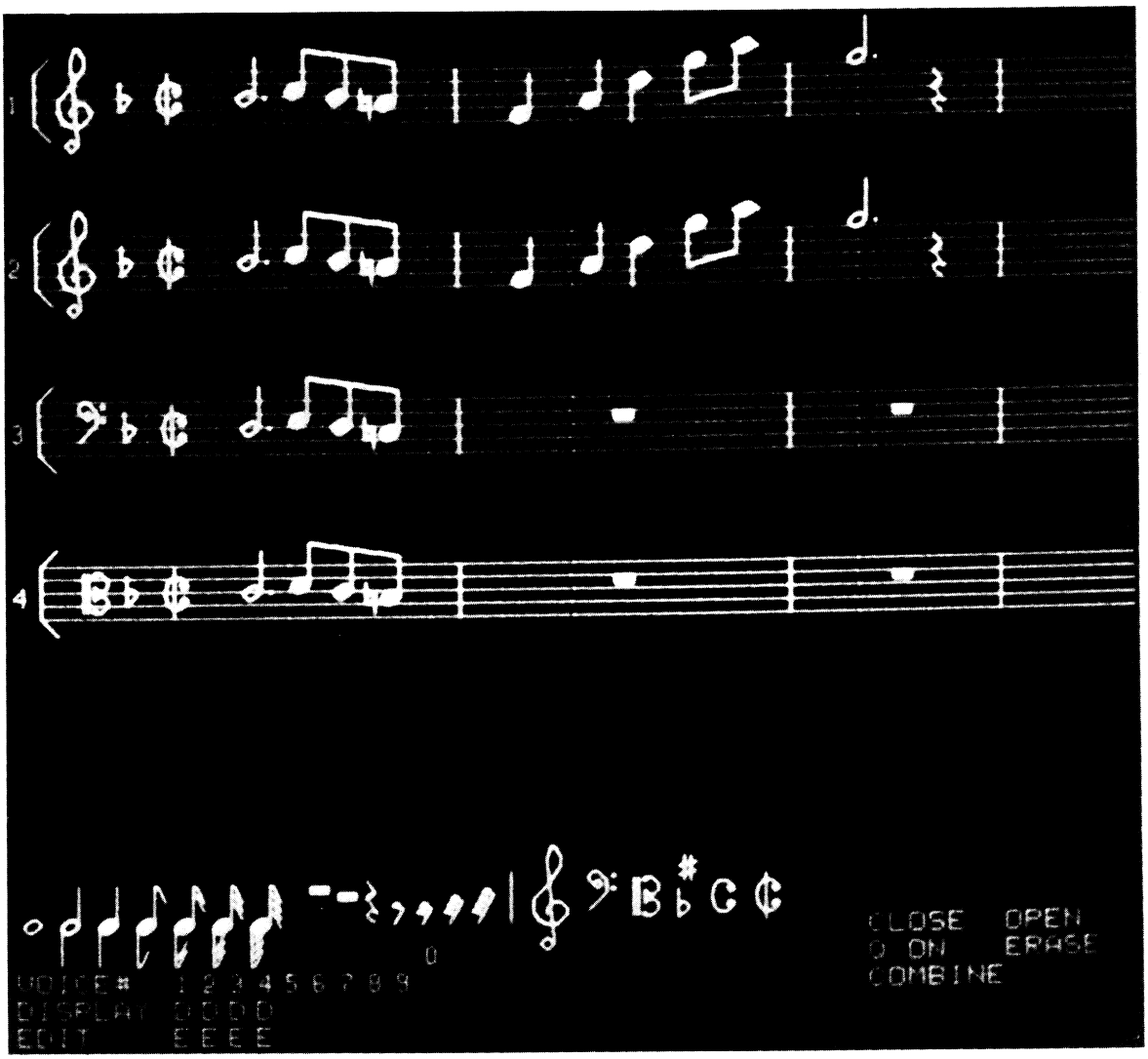



Figure 7. Sample Display

The time required to produce a line is determined by the values of ΔX and ΔY and the period of the pulse train produced by the master clock. If the larger of ΔX and ΔY is a B-bit binary number (excluding leading zeros), then the time required to produce the line is essentially $1.4(2^B - 1)$ microseconds. This has the following implication. If one desires, for example, to draw a horizontal line 128 raster units long, he should be aware that a line 127 raster units long would take half as long to produce. Note also that an additional factor-of-two increase in the line drawing speed can be obtained by specifying a line half as long as the desired length and setting the up-down counters to count by two. For example, specifying a length of 63 instead of 128 units and double size instead of normal size would produce a line essentially equal in length to 128 units long (but with half the resolution) and it would take only one-fourth as long to produce the line as it would to produce one 128 raster units long.

D. COMPUTER SPEED

As mentioned previously, it is possible to operate the display processor either under program control or under data channel control, and in either case, the computer can usually supply data as fast as it can be used by the display processor. The primary times that the speed of the computer becomes a limiting factor are those times when the display processor executes commands that do not turn on the line generator or the character generator, or trigger the set-point delay circuit. Examples of such cases are op-code 110 commands (SET PARAMETERS), op-code 100 or 101 commands (SET X or Y) in which

the WAIT bit is a zero, and op-code 001 or 010 commands (LOAD ΔX or ΔY) in which the START bit is zero. The time required for the display processor to execute these commands and request a new input word is approximately 100 nsec. In these cases, the display processor will be inactive while waiting for the computer to send the next command. If the display processor is being controlled by a program, this delay will be about 15 microseconds. If the display processor is being operated by the data channel, this delay will be about 4 microseconds.

E. SYSTEM IMPROVEMENTS

The primary areas where improvements could be made in the display processor are the character generator, and the CRT deflection signals. The improvement needed is an increase in speed. The character generator could be improved in one of two ways. First, the rate of the master clock could be varied during the production of each character, increasing the rate if the matrix points are unintensified and decreasing the rate for the intensified points. This would require that the intensification circuit for characters be modified slightly also to insure proper operation of the character generator. The second approach would be to use a stroke-type character generator which moves the beam over those points that will be intensified and not over the unintensified points unless such a move is necessary to get to the next intensified segment of the character. Such a character generator is more costly than the matrix-type character generator and it would require a read-only memory whose contents are completely different from the contents of the memory used.

An alternative solution to the speed problem would be to modify the output D/A converters so that they would produce large spikes (of the appropriate polarity) each time the D/A converter changes value. The purpose of the spike would be to increase the initial acceleration of the beam so that it would move to the next location more rapidly. The amplitude and width of the spikes would have to be adjusted experimentally to some optimum value. Clearly, this solution would increase both the line drawing rate and the rate of character generation. In addition, the spikes could be produced by some separate circuit and added to the output at the summing node of the deflection op-amps (thus leaving the D/A converters unchanged).

The manner of handling point commands (SET X or Y) could be made more sophisticated if the delay before intensifying the point were made proportional to the distance between the present beam location and the new location. This would probably be a relatively complex task to perform and would add significantly to the cost of the display processor.

One final improvement would be to make the hardware sensitive to such commands as carriage return, line feed, backspace, tab, and so forth. Here again, though, the cost would be increased (possibly significantly) for each command incorporated into the hardware.

CHAPTER V

CONCLUSIONS

The goal set at the outset of this project was to produce a relatively sophisticated, medium performance display processor at a lower cost than commercially available units. As pointed out in Chapter I, there are many commercially available units ranging in price from around $40,000 to several hundred thousand dollars. It is clear that for a large amount of money one can buy any one of a number of very sophisticated display systems. It is also clear that for a small amount of money one can buy or build some rather unsophisticated display systems. So in considering and evaluating any display system one must consider both the cost and the capability of the system.

The capabilities of the display processor built as part of this project have been considered in detail in the preceeding chapters. At this time the cost of the system will be considered. The hardware used in constructing the display processor can be divided into several categories. These are: integrated circuits, the read-only memory, operational amplifiers, printed circuit boards and connectors, power supplies and discrete components. The total cost of the hardware is approximately $3,000. A more detailed breakdown of the cost is given in Appendix B.

But before one can compare this cost to the cost of commercially available display systems it is necessary to estimate what this system would sell for if marketed. The cost of the CRT must also be included

in this price because it is included in the price of all of the commercially available models. Allowing $5,000 for the CRT and estimating a marketing price of 4 times the parts cost, the total cost for the display processor would be about $17,000. This is less than half of the cost of the least expensive commercially available systems with comparable performance and more than an order of magnitude less than many commercially available systems. This is not to say that all of these systems are equivalent but simply to say that it has been possible to produce a reasonable display processor for much less than any of the commercially available models.

As with any hardware project, however, it is impossible to know what changes could be made in the design because of the continuing changes in the state of the art. Advances in medium-scale integration and large-scale integration over the next few years will almost certainly make it possible to obtain larger and larger subsections of the digital logic circuits on a single chip. This would mean a decrease in the initial cost of the circuit, a decrease in the number of connections necessary to make the completed device, and an increase in the reliability of the subsection and the completed device.

Another aspect of the advancing state of the art in electronics is the emergence of integrated circuit read-only memories. Any form of character generator requires a read-only memory to store some pattern for each displayable character. The memory used in the display processor is a braided read-only memory and it costs approximately $0.10 per bit. However, it is not unreasonable to predict that within

the next five years the cost of integrated-circuit memories will be as low as $0.01 per bit. This should permit a great saving in display systems built in the future.

In addition to the above extensions of the present state of the art, it is also possible that some revolutionary developments may occur that would greatly change some or many of the considerations that presently determine how display systems function. The victim of such a development might well be the only non-solid-state part of the present systems, the CRT. Possible future alternatives to the CRT include crossed-grid electroluminescent matrices, magneto-optic panels, and injection electroluminescent diode matrices.[7] All of these devices would eliminate the need for D/A conversion but would pose a possibly larger problem of decoding N-bit binary numbers into 2^N individual drive signals, one for each row and column of a 2^N by 2^N display area. However, it may be possible to substitute some new device for the presently-used up-down counters and get around the problem of decoding the value of the counter. This is purely speculation. However, as technology advances, bold new solutions to already solved problems must certainly be tried in the hope of pushing the state of the art even further. For, that which is revolutionary today may easily be commonplace tomorrow.

APPENDIX A

This appendix contains the circuit diagrams for the display processor. All logic elements are Texas Instruments (TI) series 74H or series 74 circuits in dual-in-line plastic packages. Gates are represented by standard symbols and all inputs are shown so the particular type of gate is apparent from the symbol. There are several types of gates that are available in both the high-speed and the standard-speed variety (e.g. SN74H20N and SN7420N). In these cases, the high-speed gates are marked with an H. All AND gates are SN74H11N and are not marked with an H since there is no SN7411N. The NAND buffers SN74H40N and SN7440N are marked with HB and B respectively. Each flip-flop is represented by a rectangle. Only three types were used, namely SN7473N, SN7474N, and SN7476N. The type should be apparent from the symbols used in the diagrams. The letter C is used for the CLEAR input of the flip-flops, P for the PRESET input (if present), and CL for the CLOCK input. Technical data for all of the logic elements is available in TI Catalog CC201, dated 1 August 1969.

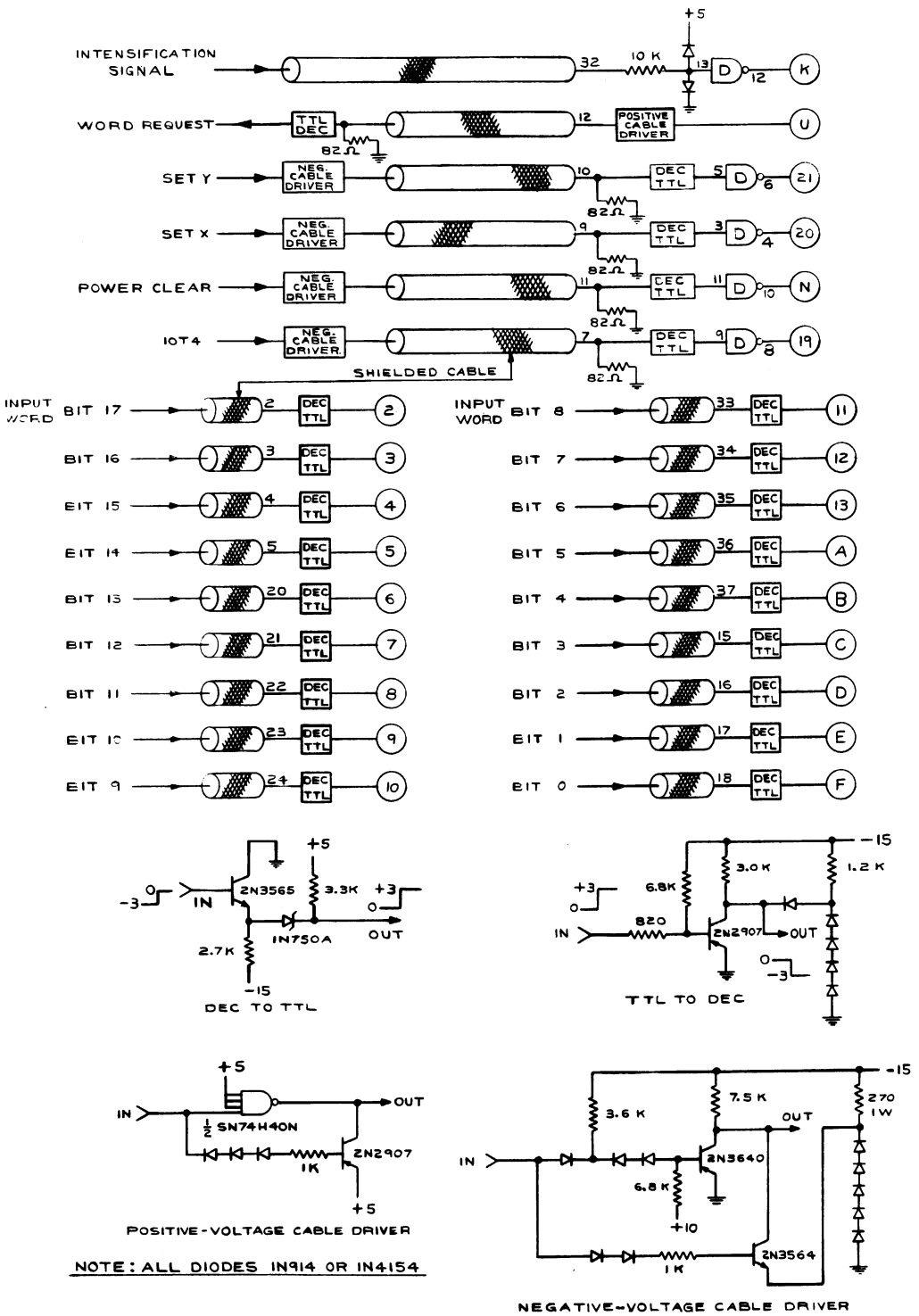


Figure A1. Interface for Display Processor

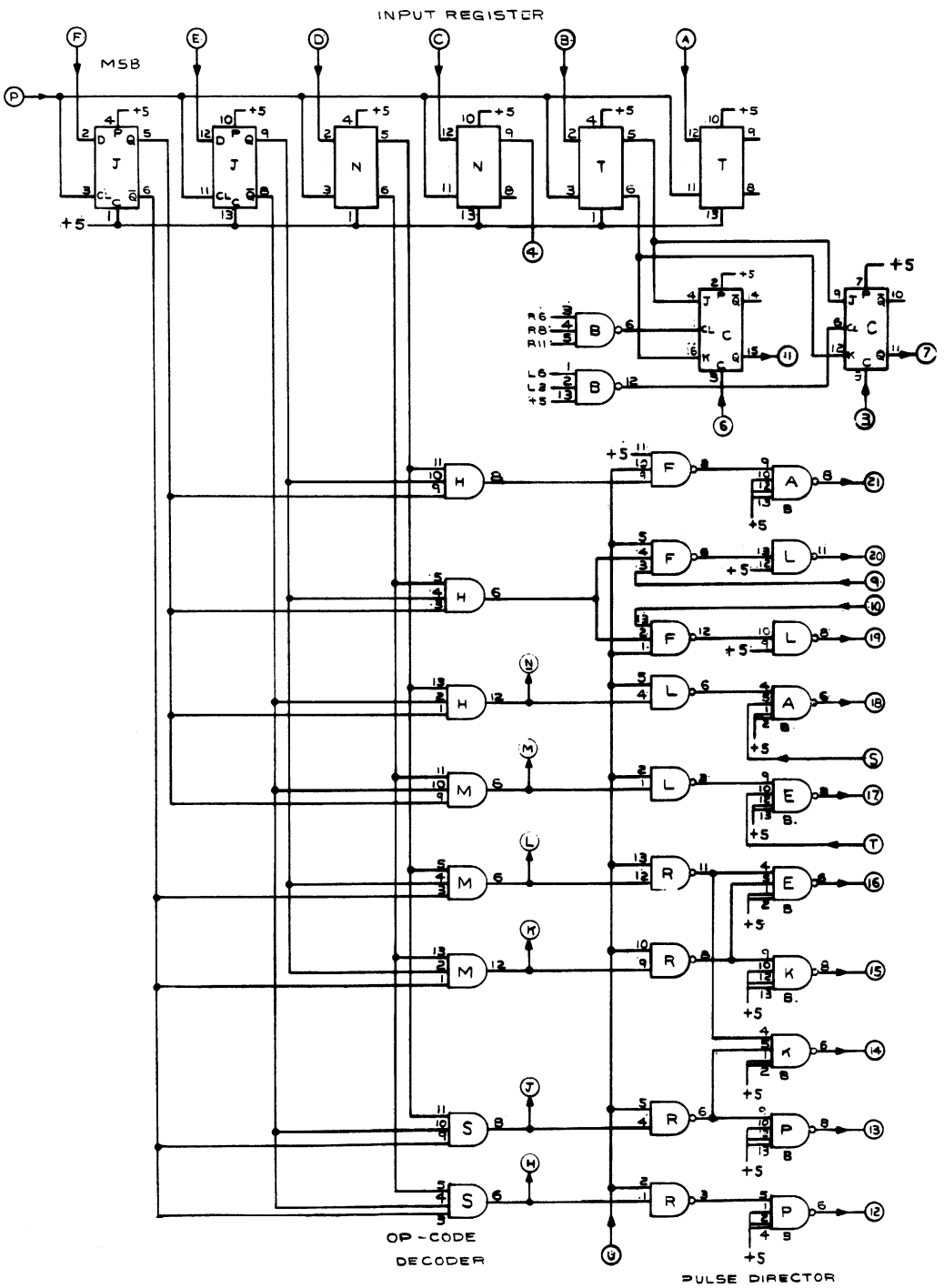


Figure A2. Printed Circuit Board Two

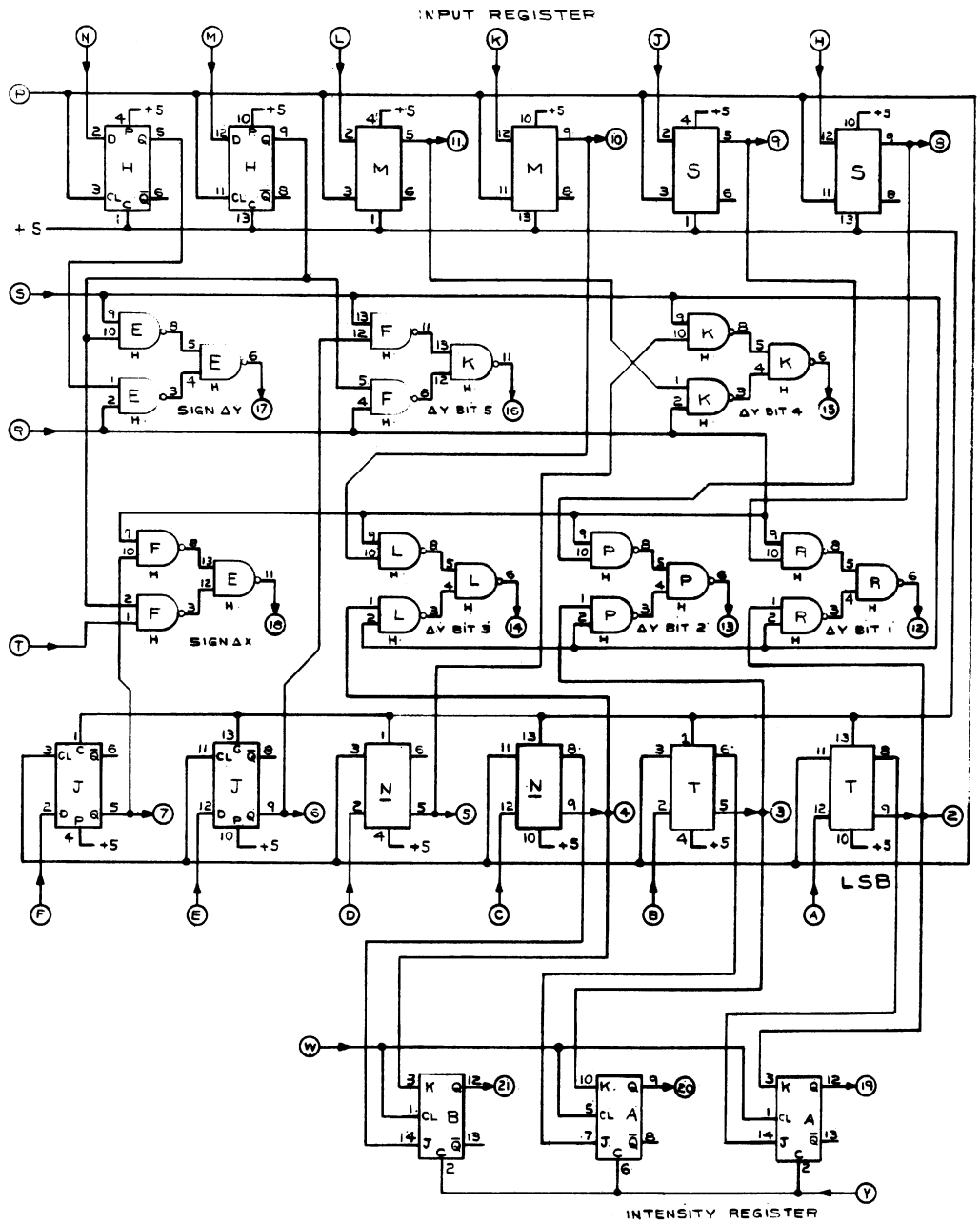


Figure A3. Printed Circuit Board Three

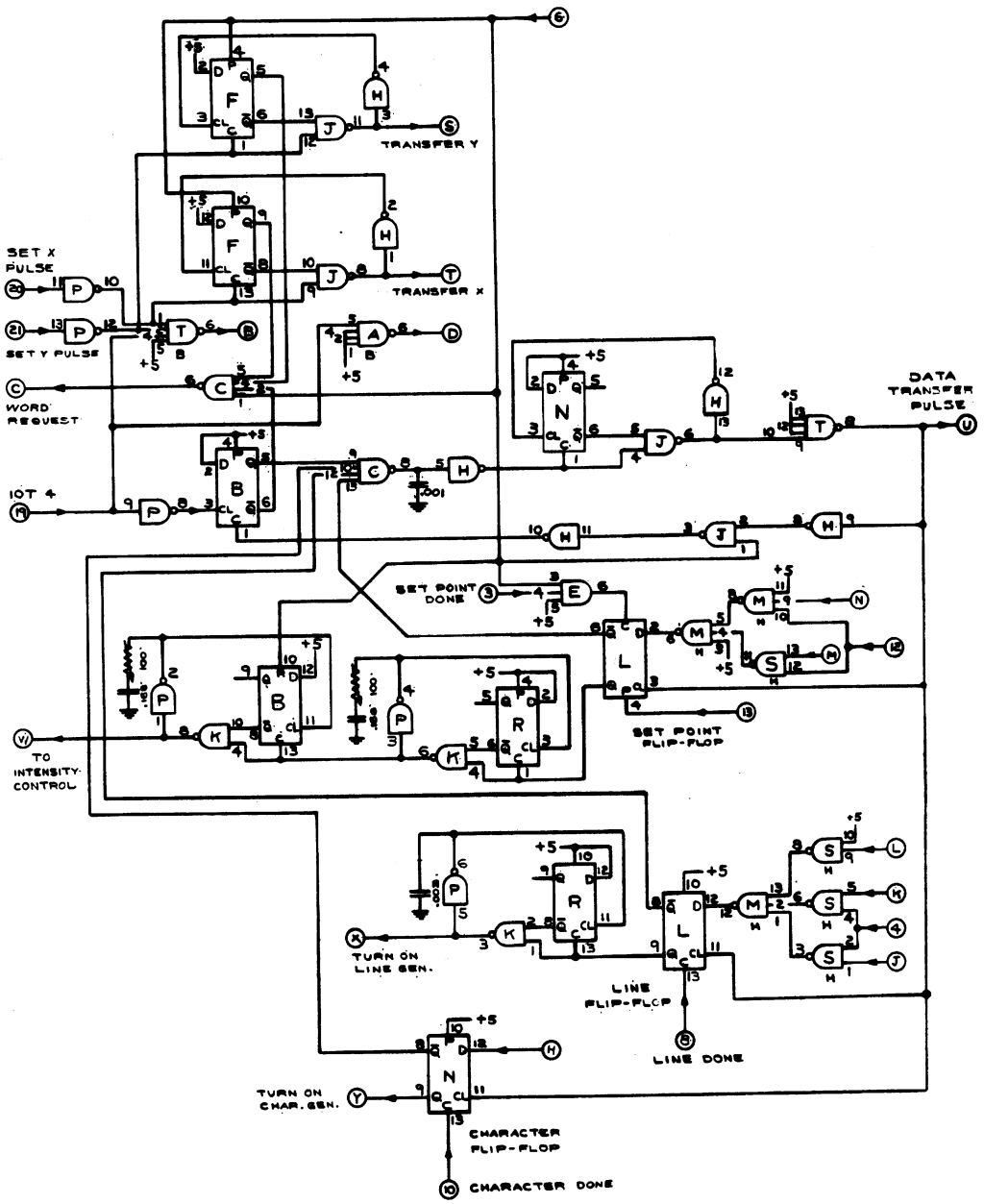


Figure A4. Printed Circuit Board Four

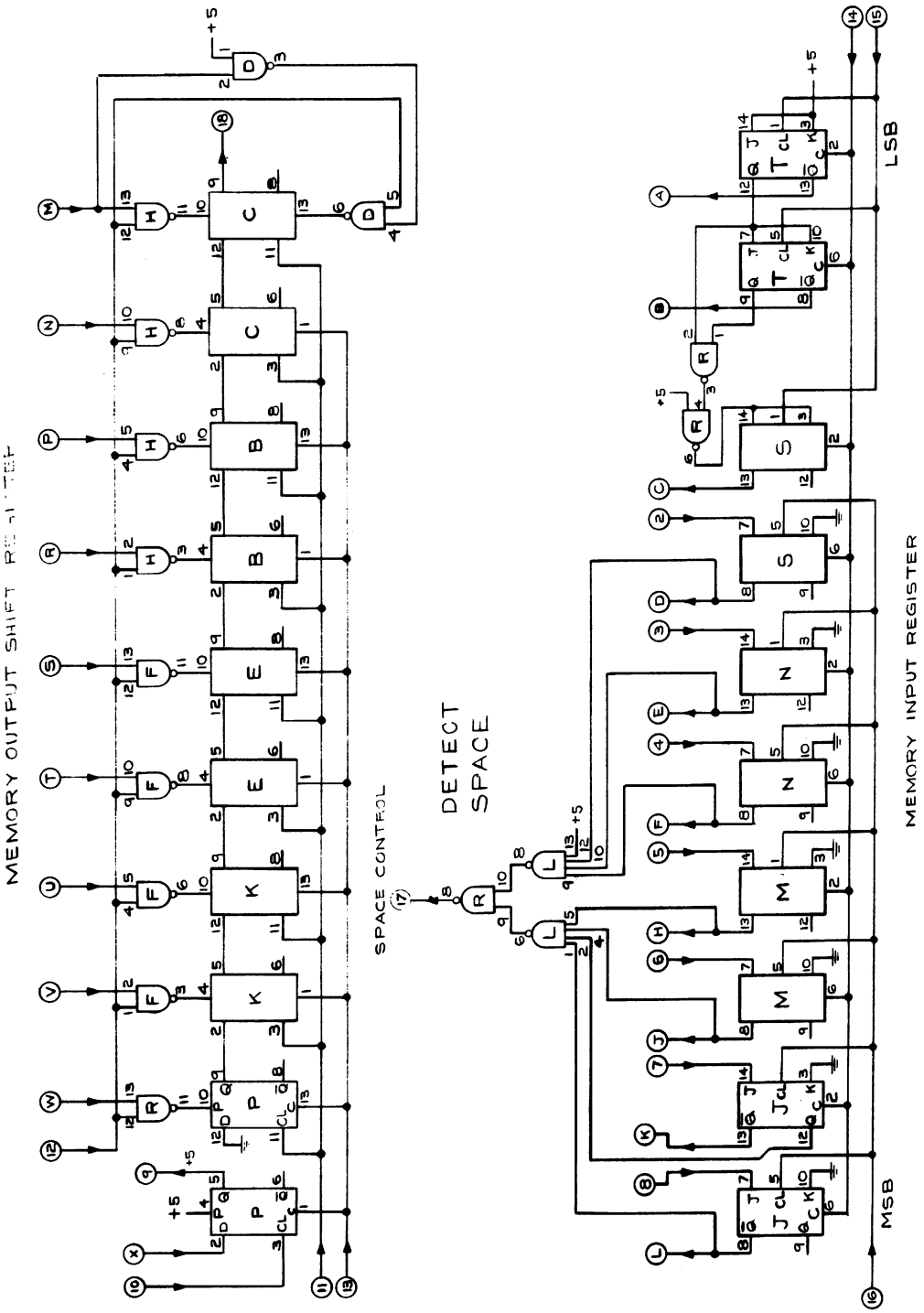


Figure A5. Printed Circuit Board Five

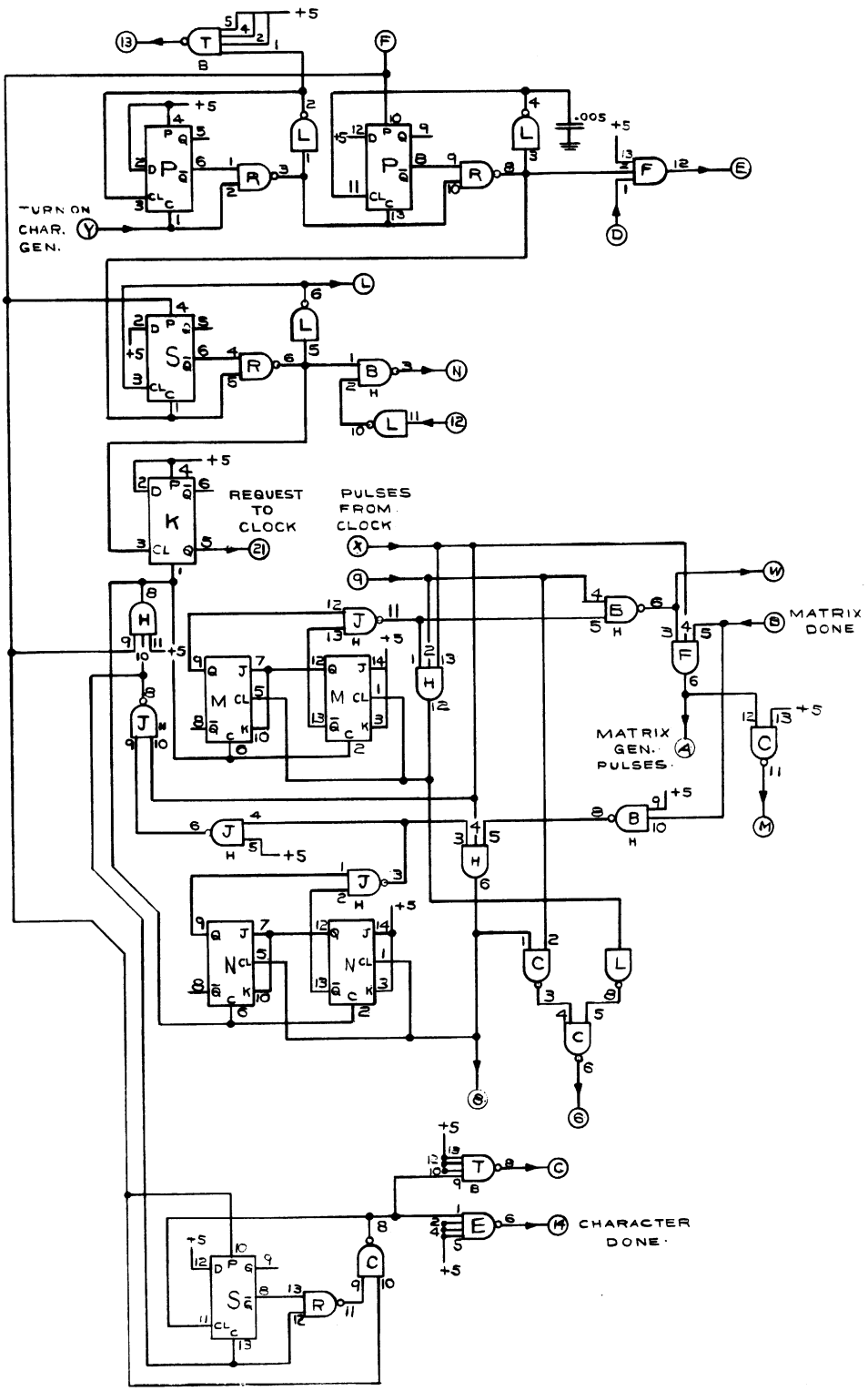


Figure A6. Printed Circuit Board Six

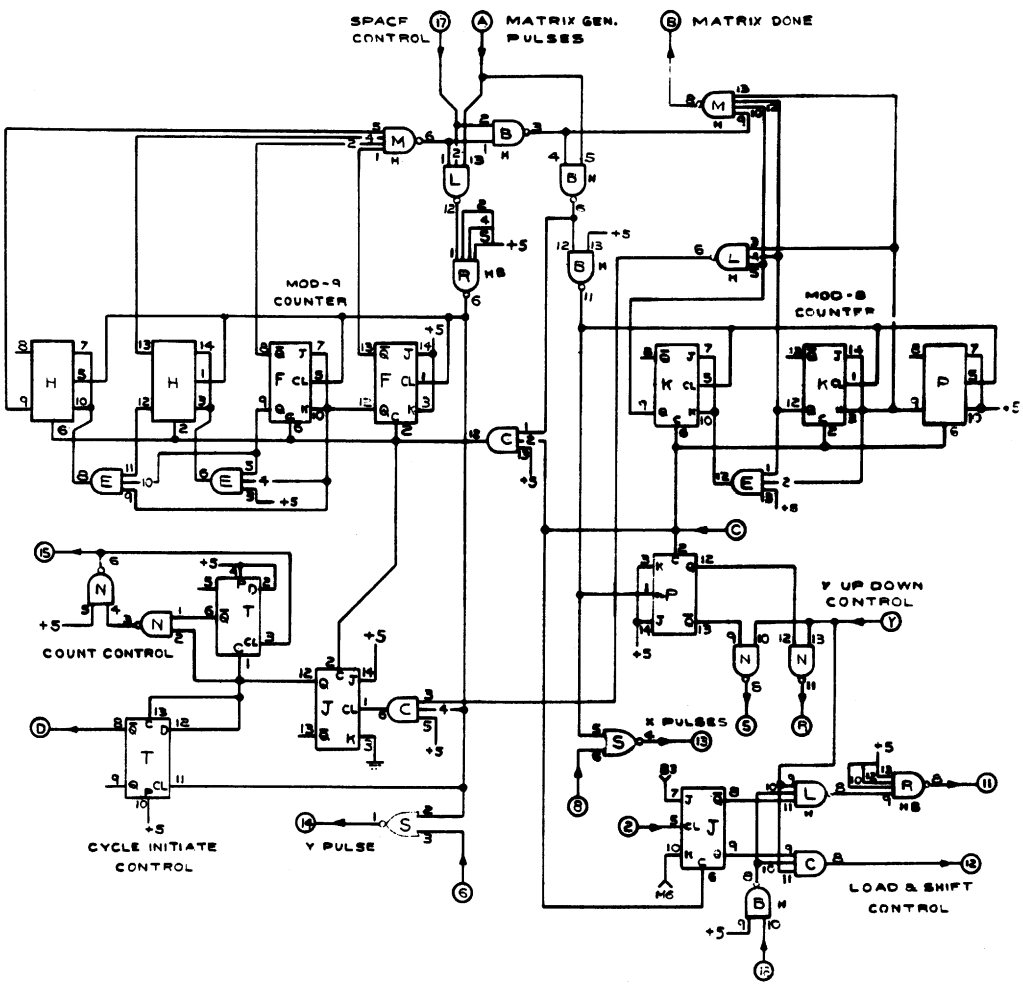


Figure A7. Printed Circuit Board Seven

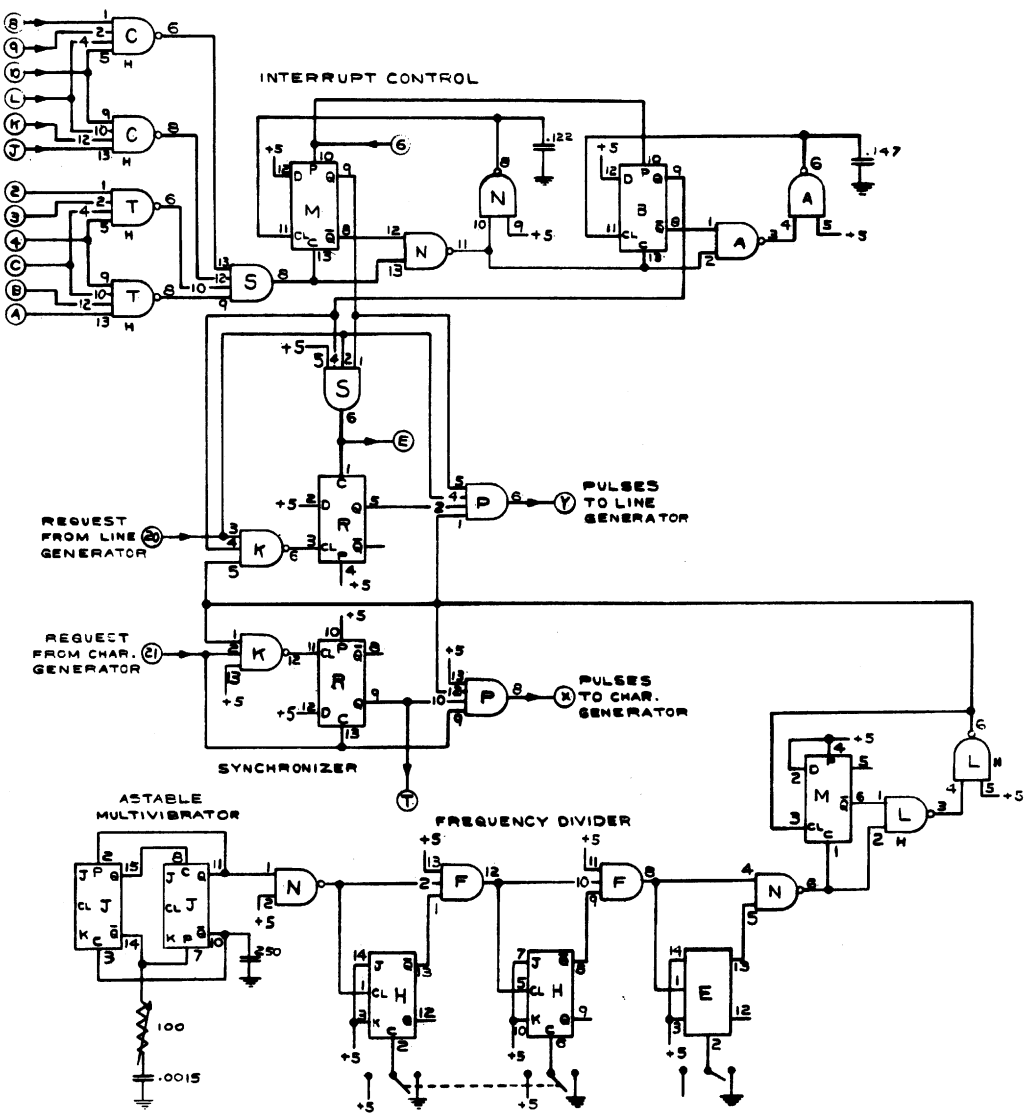


Figure A8. Printed Circuit Board Eight

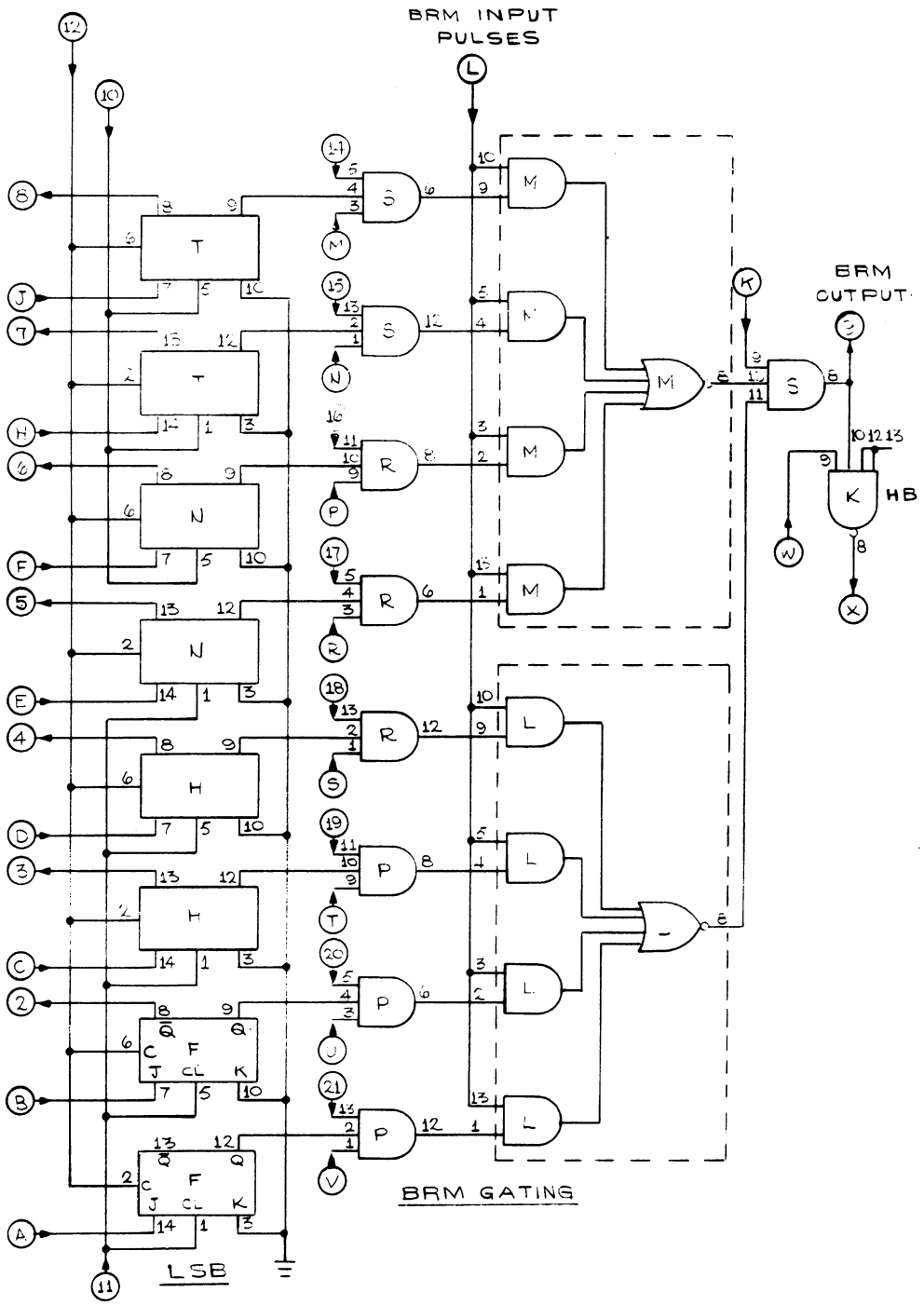


Figure A9. Printed Circuit Boards Nine and Ten

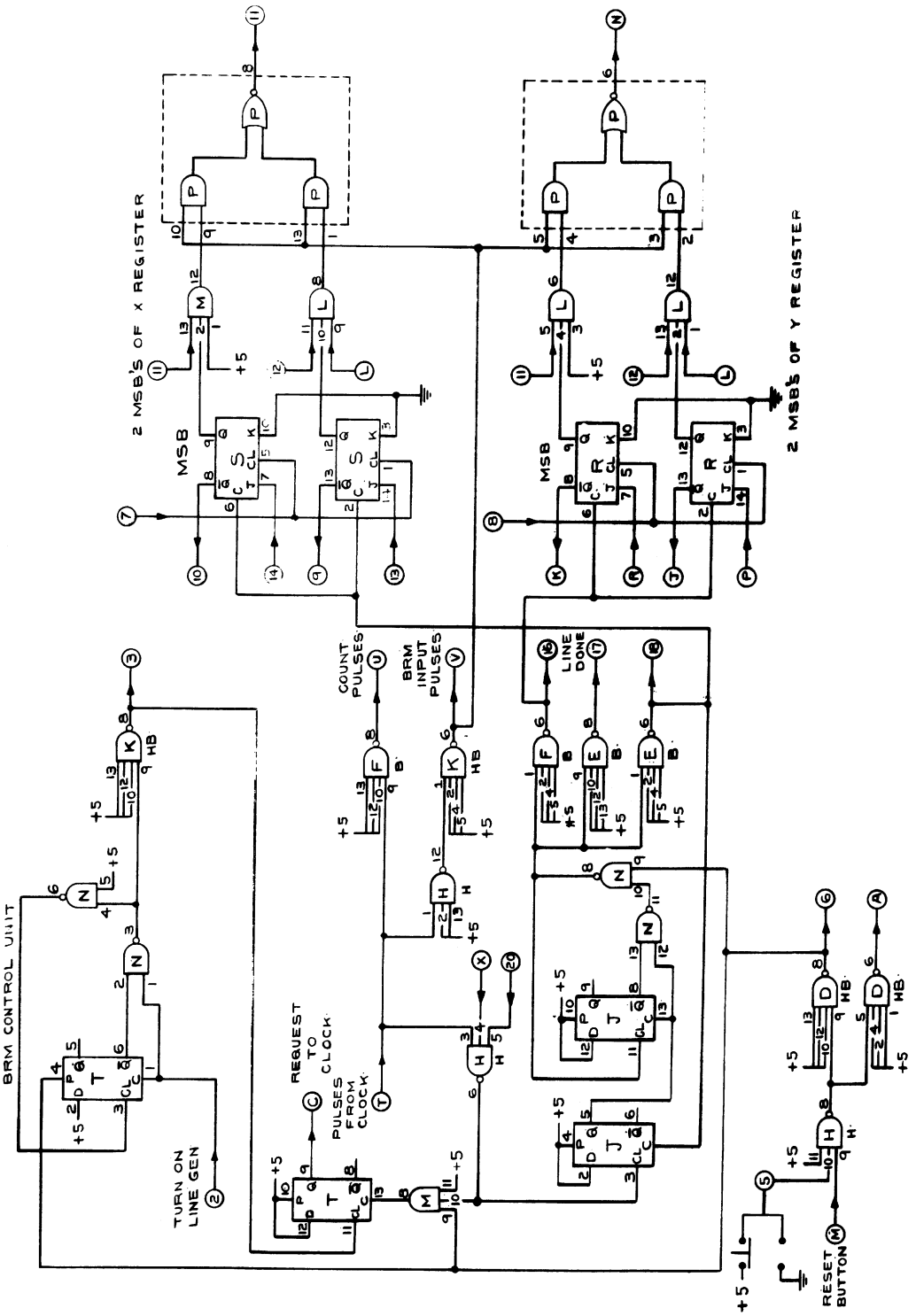


Figure A10. Printed Circuit Board Eleven

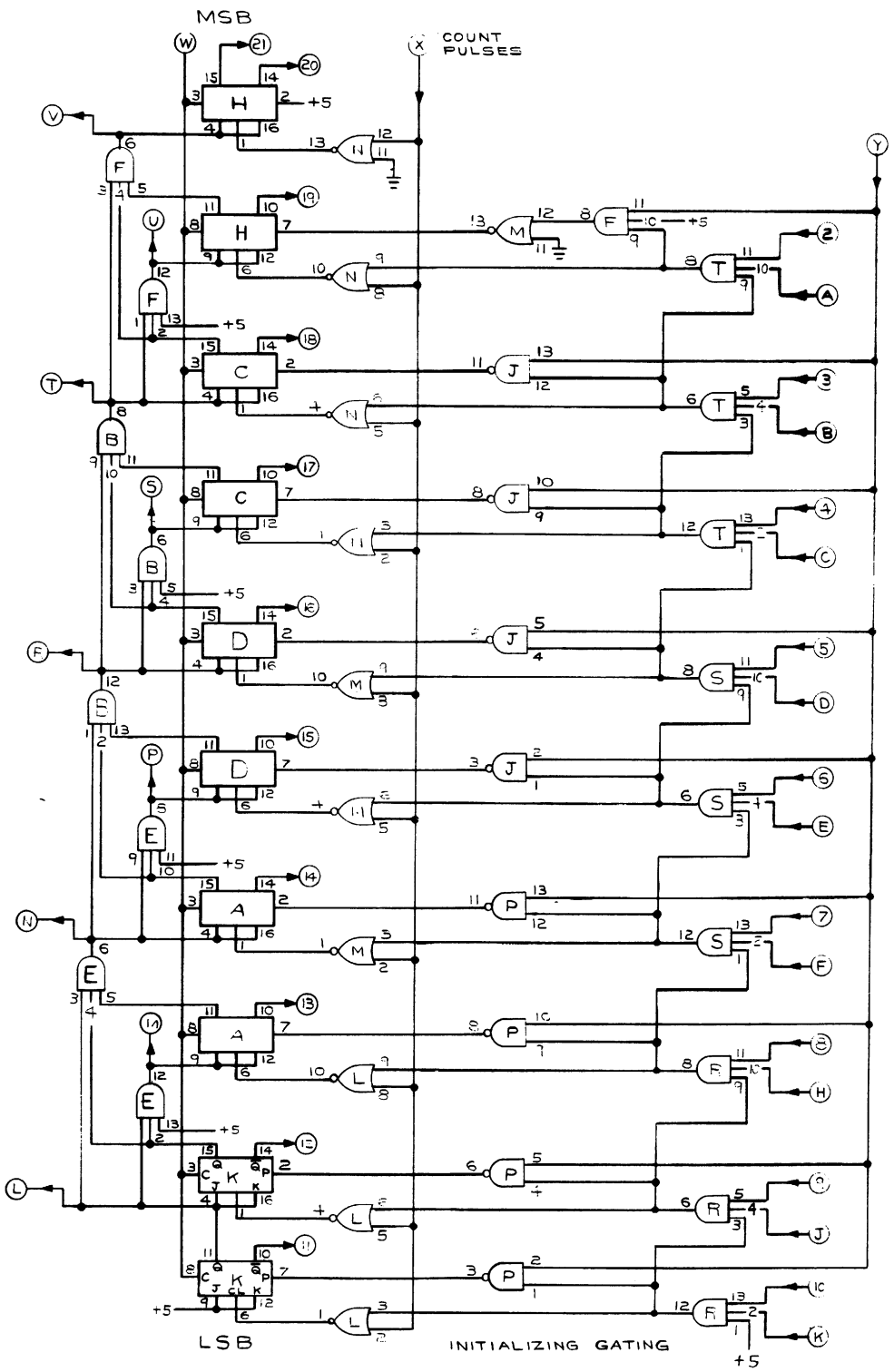


Figure A11. Printed Circuit Board Twelve

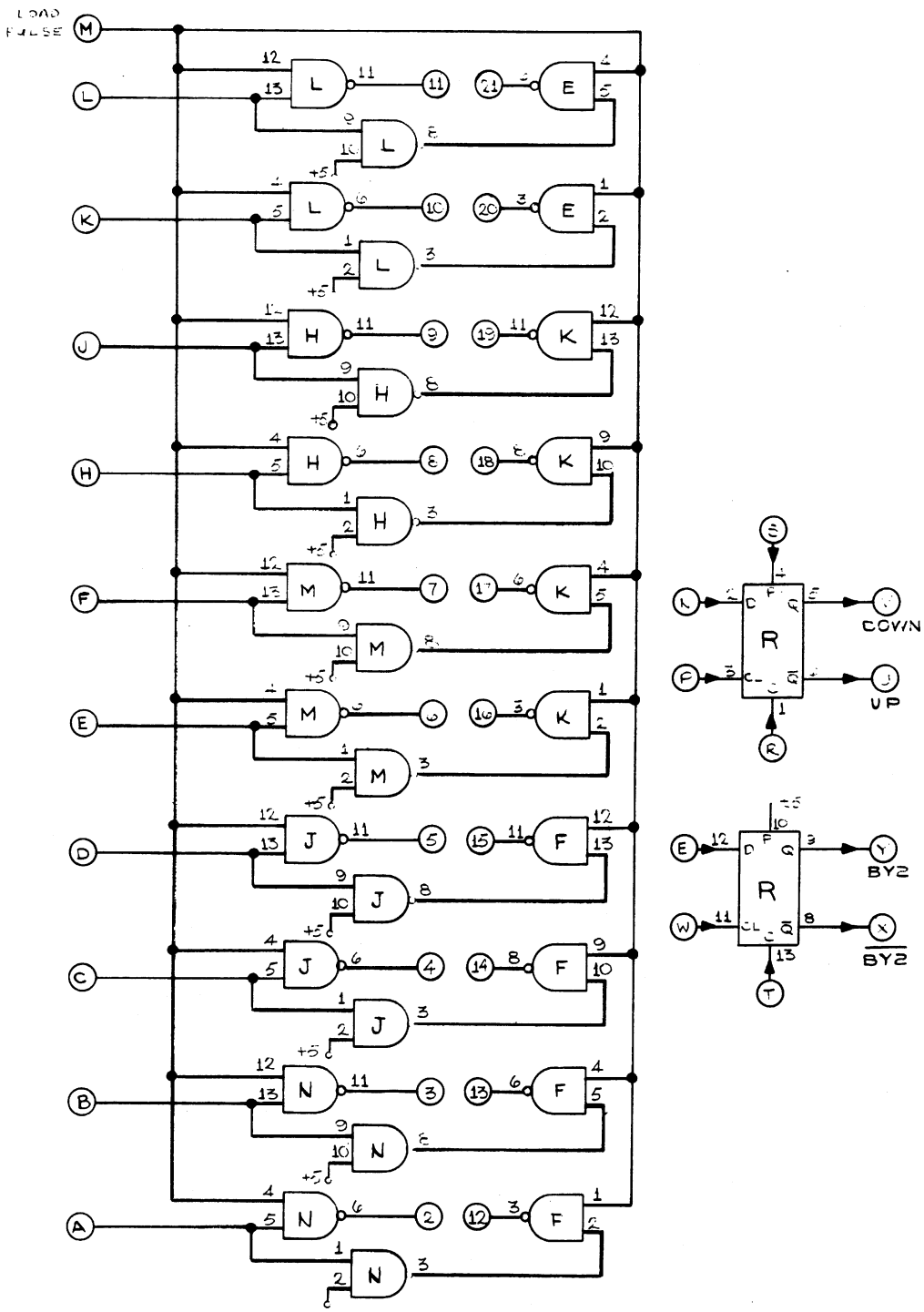


Figure A12. Printed Circuit Boards Thirteen and Fourteen

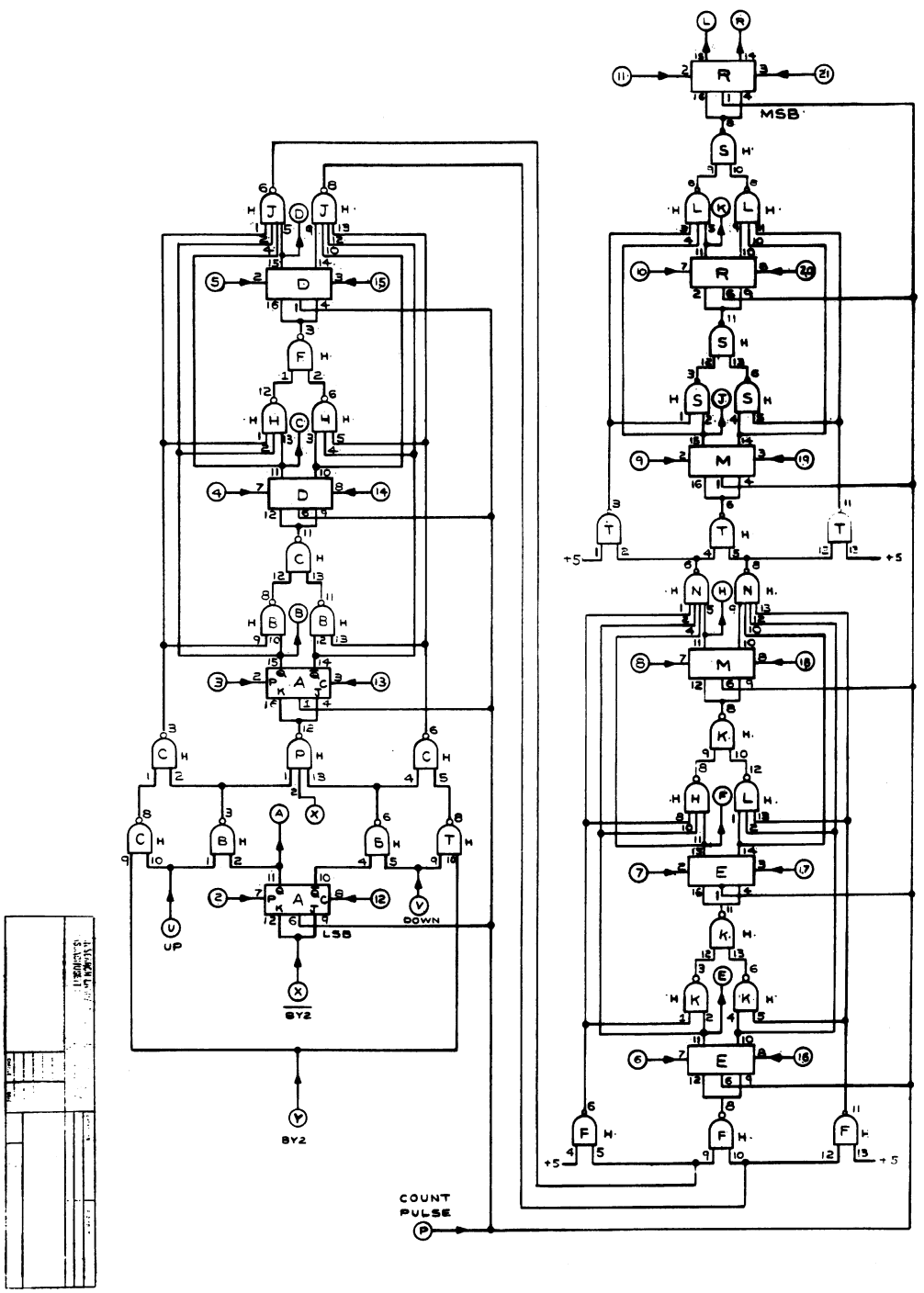


Figure A13. Printed Circuit Boards Fifteen and Sixteen

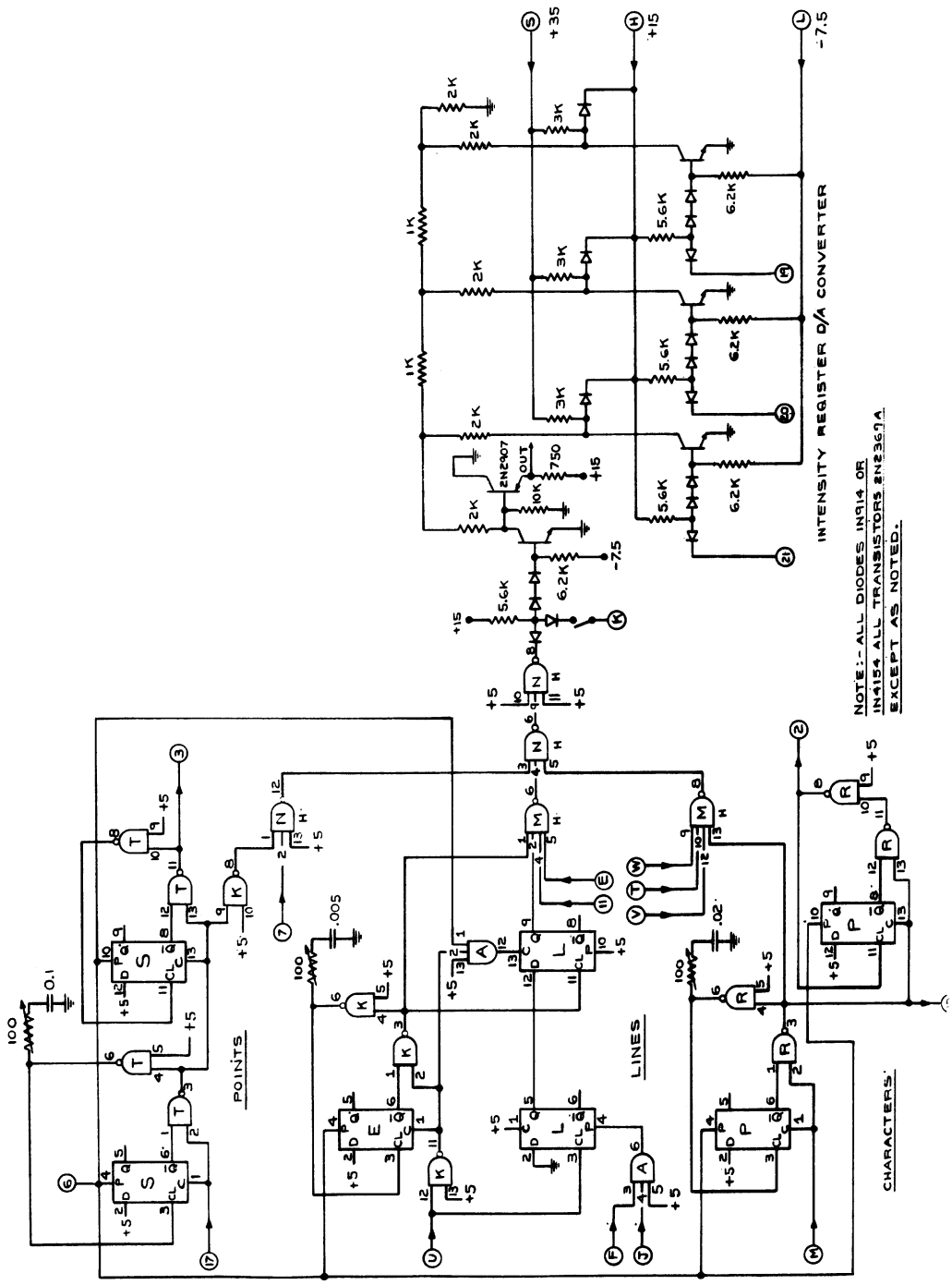


Figure A14. Printed Circuit Board Seventeen

APPENDIX B

DISPLAY PROCESSOR COST

The following is a list of the costs of the major components used to construct the display processor.

Integrated Circuits	
1. Digital (275)	$700
2. Linear (5)	25
Read-Only Memory	975
Operational Amplifiers (2)	110
Printed Circuit Boards (20)	225
Printed Circuit Board Connectors (20)	125
Power Supplies (3)	385
Discrete Components	300
TOTAL	$3,045

BIBLIOGRAPHY

1. Ninke, W.H., "Man-Computer Graphical Communication," in F.F. Kuo and J.F. Kaiser (eds.), System Analysis by Digital Computer, John Wiley and Sons, Inc., New York, 1966.

2. Adams Associates Incorporated, "DEC 340 Precision Incremental CRT Display," The Computer Display Review, 1967.

3. Adams Associates Incorporated, "IBM 2250 Model III Display Unit," The Computer Display Review, 1967.

4. Stotz, R.H. and Ward, J.E., "Operating Manual for the ESL Display Console," ESL Internal Memorandum 99442-M-129, March 9, 1965.

5. Stotz, R.H., "A New Display Terminal," Computer Design, April, 1968.

6. Machine Manual, PDP-9 Digital Computer, Digital Equipment Corporation, Maynard, Massachusetts.

7. Hobbs, L.C., "Display Applications and Technology," Proceedings of the IEEE, December, 1966.

This empty page was substituted for a blank page in the original document.

CS-TR Scanning Project
Document Control Form

Date : 2/8/96

Report # LCS-TR-71

Each of the following should be identified by a checkmark:
Originating Department:

☐ Artificial Intellegence Laboratory (AI)
☒ Laboratory for Computer Science (LCS)

Document Type:

☒ Technical Report (TR) ☐ Technical Memo (TM)
☐ Other:_____

Document Information

Number of pages: 76 (82-IMAGES)
Not to include DOD forms, printer instructions, etc... original pages only.

Originals are:
☐ Single-sided or
☒ Double-sided

Intended to be printed as :
☐ Single-sided or
☒ Double-sided

Print type:
☐ Typewriter ☐ Offset Press ☐ Laser Print
☐ InkJet Printer ☒ Unknown ☐ Other:_____

Check each if included with document:
☒ DOD Form ☐ Funding Agent Form ☒ Cover Page
☐ Spine ☐ Printers Notes ☐ Photo negatives
☐ Other: _____

Page Data:

Blank Pages (by page number): _____

Photographs/Tonal Material (by page number): 42-43, 45

Other (note description/page number):

Description :	Page Number:
IMAGE MAPS (1-76) UNNUMBERED TITLE & BLANK PAGES,	
V-VIII, 1-69, UNNUMBERED BLANK.	
(77-82) SCANCONTROL, COVER, DOD, TRGT'S (3)	

Scanning Agent Signoff:
Date Received: 2/8/96 Date Scanned: 2/14/96 Date Returned: 2/15/96

Scanning Agent Signature: Michael N. Cook

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Massachusetts Institute of Technology Project MAC	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP None

3. REPORT TITLE

A Computer-Controlled Graphical Display Processor

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

MS Thesis, Dept. of Electrical Engineering, M.I.T.

5. AUTHOR(S) *(Last name, first name, initial)*

Fiasconaro, James G.

6. REPORT DATE June, 1970	7a. TOTAL NO. OF PAGES 77	7b. NO. OF REFS 7
8a. CONTRACT OR GRANT NO. F19628-69-C-0044, Nonr-4102(01) b. PROJECT NO. c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) MAC TR-71 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*	

10. AVAILABILITY/LIMITATION NOTICES

Distribution of this document is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Office of Aerospace Research, Air Force Cambridge Research Labs. and Advanced Research Project Agency

13. ABSTRACT

This report describes the design, construction and operation of a low-cost, relatively sophisticated display processor that can be used with a standard cathode-ray tube (CRT)monitor to display computer-generated pictures. The design was based on the following desired characteristics. The display processor should be able to intensify any one of the 1024 X 1024 raster points of the 10 X 10 inch display area. It should be able to draw straight lines of arbitrary length (up to 1024 raster units along each coordinate). It should be able to display any of the 96 printable ASCII characters. And finally, it should provide two character sizes and eight intensity levels.

The prototype that was built based on the above characteristics worked quite well. The total parts cost for the display processor was about $3,000.

14. KEY WORDS

Graphical Display Processor, Inexpensive Graphical Display Processor Line Generator, Character Generator, Binary Rate Multiplier, Computer Driven Display, Random Position Display, Medium Performance Graphical Computer Display

DD FORM 1 NOV 68 1473 (M.I.T.)

UNCLASSIFIED
Security Classification

Scanning Agent Identification Target

Scanning of this document was supported in part by the **Corporation for National Research Initiatives,** using funds from the **Advanced Research Projects Agency** of the **United states Government** under Grant: **MDA972-92-J1029.**

The scanning agent for this project was the **Document Services** department of the **M.I.T Libraries.** Technical support for this project was also provided by the **M.I.T. Laboratory for Computer Sciences.**

Scanned

Date: 2/26/1996

M.I.T. Libraries
Document Services