AN ENGLISH AND ARABIC CHARACTER PRINTER

TWO PROBLEMS IN LIEU OF THESIS

Presented to the Graduate Council of the
North Texas State University in Partial
Fulfillment of the Requirements

For the Degree of

MASTER OF SCIENCE

By

Malek G. Abdel-Razzack, B. A.

Denton, Texas

December, 1976

PREFACE

This paper is presented in satisfaction of the requirement for two problems in lieu of thesis which are required for the degree, Master of Science. The two problems are: (1) to provide an electronic interface between the M6800 microprocessor and the printer; and (2) to design an Arabic character set and to provide the logic required for its implementation. As it would be artificial and impractical to document these problems separately, a single document is here provided.

In this paper, a general description of the M9601 A. B. Dick Videojet printer and of the Southwest Technical Products Motorola M6800 computer system are included. If the reader is, however, interested in more detailed information, he is advised to refer to the technical manuals described in the Bibliography.

Examining the Arabic alphabet shows that it is very similar to the Persian alphabet. With some very minor additions (three characters to be exact), this printer could be modified to become an English, Arabic, and Persian printer.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

CHAPTER I

INTRODUCTION

This paper concerns two problems. It consists on the one hand of the description of the design and construction of the electronic development needed to connect an A. B. Dick M9601 Videojet (TM) non-impact printer to a Southwest Technical Products version of the Motorola M6800 Microprocessor system, and on the other hand of increasing the character set of the printer to include the Arabic alphabet.

Chapter II contains a general description of the A. B. Dick M9601 non-impact printer and discusses its principles of operation. Chapter III is concerned with the general features of the Southwest Technical Products version of the Motorola M6800 microprocessor system, because they are related to the interfacing problem. The M6800 is, however, typical of the microprocessor, costing only a few dollars, whose sales should soon approach hundreds of thousands per year. Chapter IV describes how these two devices, the printer and the computer system, are interfaced. Finally, Chapter V describes the Arabic language and the design of the hardware needed to provide it for the M9601 Videojet printer.

Computer technology is not as advanced in the Middle-East as it is in the United States. In developing Arabic-English printers that can be easily connected to computers, a great service and encouragement to arouse interest in computer technology would be rendered. Since more than a hundred million people use the Arabic language, this sort of encouragement in using computers amoung them might eventually get them to render some contributions in the computer field.

CHAPTER II

•FEATURES OF THE A. B. DICK M9601 VIDEOJET PRINTER

The M9601 non-impact Videojet (TM) printer produces printed information from binary input data at the rate of 250 characters per second. This printer is a non-impact printer in the sense that it uses a Videojet (TM) assembly to produce the letters from the supplied input serial data with each letter formed from a 9 by 11 matrix of very tiny droplets, thus constituting 99 droplets for each character. This input data was originally supplied from computer via telephone by a 201A Dataset.[1] The codes for English conform to the USASCII[2] code.

The printer, including its electronics described below, was purchased and donated to the University by Professor Dan W. Scott. As illustrated in figure 1, the printer consists of two sections: a) A mechanical section to handle Videojet printer and paper advance controls and b) an electronic section consisting of digital integrated circuits to handle all data.

The mechanical section is illustrated in Figure 2. It consists of an ink tank, a pump, and a pressure regulator that regulates the ink pressure through a line filter to the printhead. The ink gutter drains the excess unused ink droplets

[1] A Southwestern Bell transmitter-receiver unit.

[2] An American Standards Committee on Information Interchange established the Standard Code for characters used in data communications. It is also referred to as ASCII.

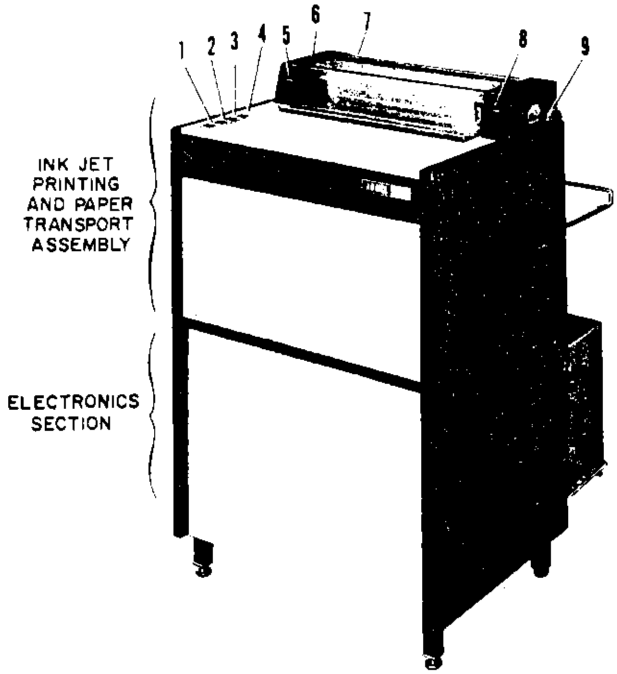


Fig. 1--The components of the printer source: A. B. Dick Videograph Systems Manual, pages 1-3. Actual size: 3:1/2' X 2' X 2'.

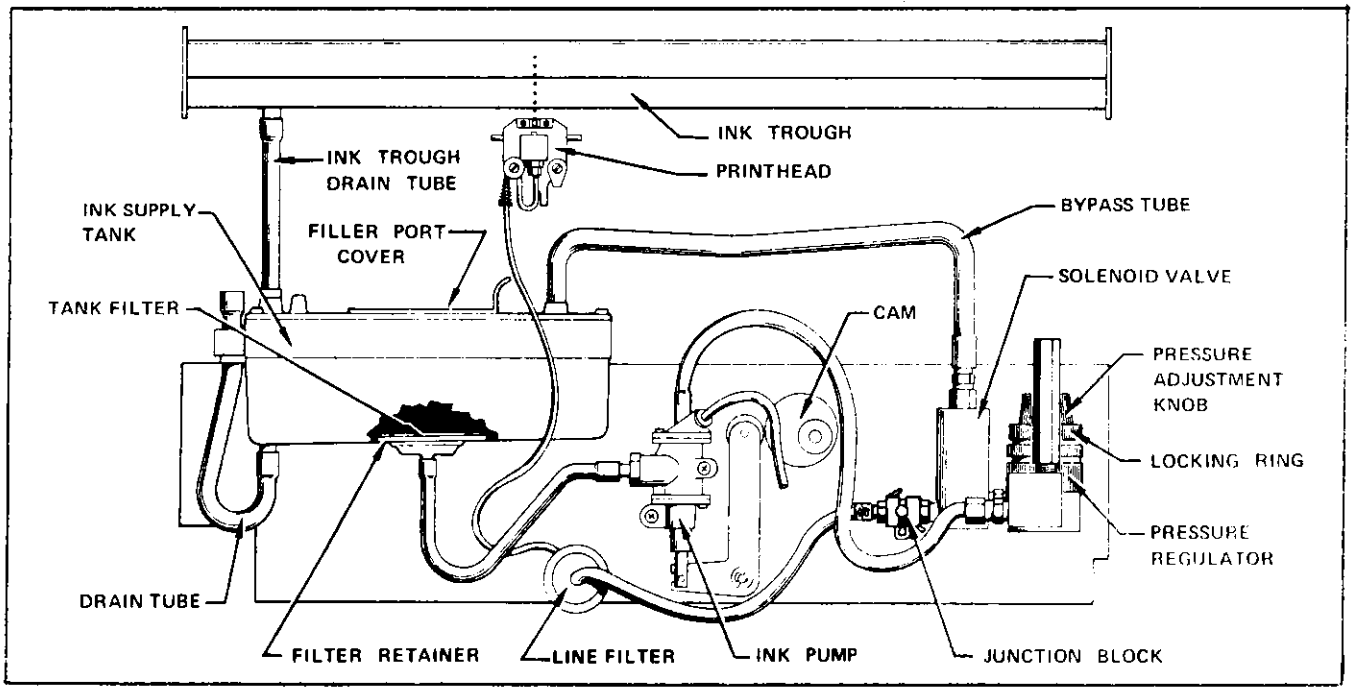


Fig. 2--The mechanical section source: A. B. Dick Videograph Systems Manual, pages 2-4.

back to the ink tank.

Principle of operation: The principle of operation is illustrated in Figure 3. The printer employs a small metal chamber having an orifice at one end, approximately .002 inches in diameter, which is energized by a source of ultrasonic energy (a piezoelectric crystal). This causes the ink to be discharged as a stream of droplets of the same size at a rate equal to the frequency of the energizing crystal (66 KHZ in this particular case). At a point slightly in front of the orifice, the droplets break away from the main ink stream where each droplet obtains an electrical charge according to the signal it receives from the character decoding circuitry depending on the character to be printed. The charged droplet then goes between a pair of high voltage deflection plates, where it is deflected on a printing medium such as paper, according to its previously received charge.

As illustrated in Figure 4, the electronic section consists mainly of eleven modules of integrated circuits labelled as J1 through J9 and J14, J15, with the following functions:

Module J1: This module acts as the interface circuit of the printer. It accepts serial data applied to the printer in eight bit segments, checks the parity, and recognizes the ASCII encoded directives[1] such as SOH, EM, EOT,...etc. Finally, any eight data bits for a particular character of the

[1] The meaning of these directives is explained in Chapter IV.

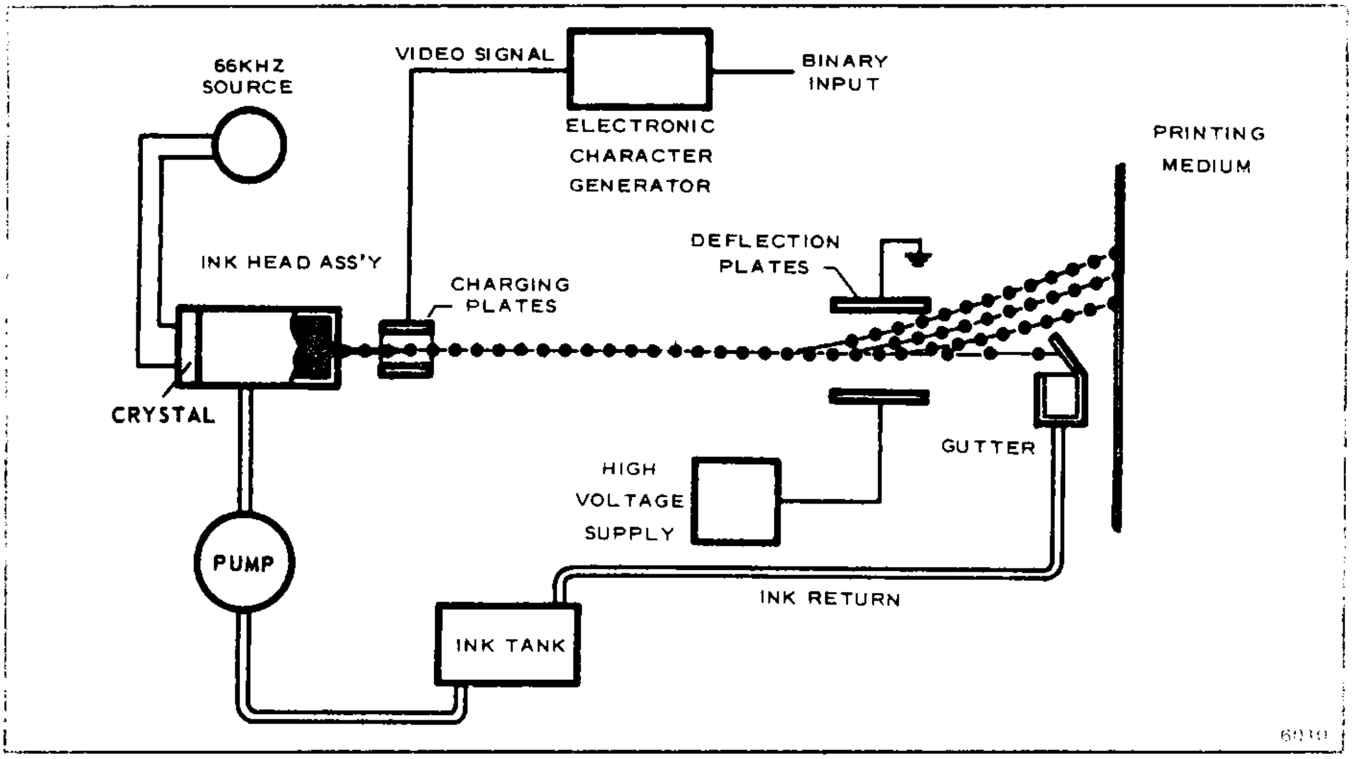


Fig. 3--Principle of operation components source: A. B. Dick Videograph Systems Manual, page 3-1.

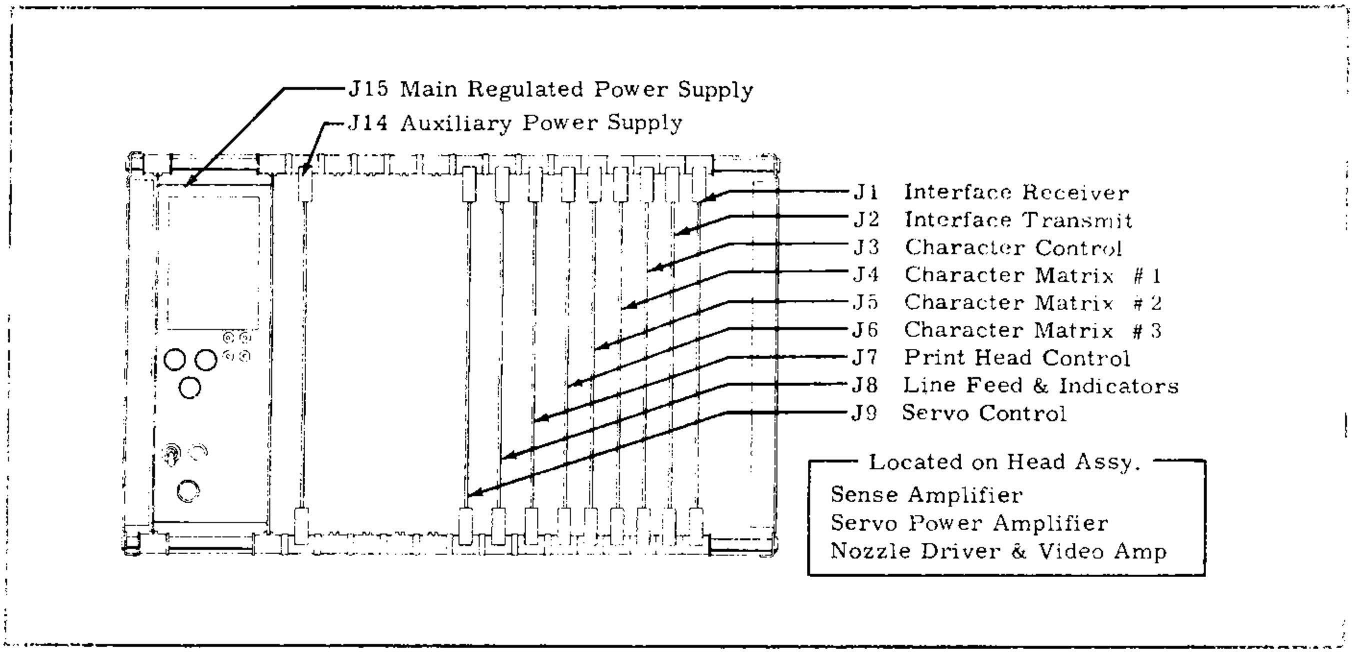


Fig. 4-- The electronic section source: A. B. Dick Videograph Systems Manual, page 2-1.

ASCII alphabet are routed, by this module, to the data register of module J3.

Module J2: This module acts as the transmission interface of the printer. It is used to convert any ASCII encoded directives to be transmitted from the printer such as ACK, NAK from the parallel format to the serial one, then transmit them to the Dataset.

Module J3: This module is divided into three sections with the following functions:

(a) Clocking and reset: The clock needed for advancing the ninety-nine coordinates of the 9 by 11 matrix and the modulating of the ink nozzle for drop generation is generated by this section. The basic frequency of this clock is 66KHZ. Besides the above clocking, a Master Reset clock is provided to keep all memory bistables on this module reset until five volts has been fully established.

(b) Generation of character coordinates: This section of the module has the function of breaking up the y-axis into eleven coordinate positions and the x-axis into nine coordinate positions. Every time interval during character generation will thus define one y position and one x position simultaneously.

(c) Control and buffer register: This section consists of a buffer and the necessary controls for the code entry of the character into the buffer. The control is necessary to indicate that a completion of scanning of one character and that the buffer is ready for another character has occured. Also simultaneously with the entry of the code for each character, a control train of ninety-nine pulses is produced corresponding to the ninety-nine dots of the matrix.

Modules J4, J5, J6: These three modules contain the required gates for producing all the characters used by the printer. These characters are formed by passing the x and y pulse trains through the necessary logic gates. To simplify matters, all these characters on the three boards are generated simultaneoulsy. The selection of a certain character is accomplished by the decoding of the character code. This particular decoding of the character set is also done through the use of some logic gates, on these three modules.

Module J7: This module is divided into four sections with the following functions:

(a) Drop phasing: This section is responsible for placing an electrical charge on each individual ink drop when this particular ink drop is about to separate from the ink stream. The amount of charge placed on each drop varies, depending on the character to be printed, with different deflections occuring with different charges.

(b) Character video generator: The function of the video character generator is to accept an input pulse train, control the amplitude of each pulse and then deliver the altered pulse train to a video amplifier.

(c) Carriage and printer ready: This section informs the Dataset of the carriage position and printer status thus synchronizing the data transfer between the printer and the Dataset.

(d) Fault control: This section causes a fault indicator switch to light up indicating a fault and stops the printer from accepting any more signals from the Dataset. This happens only if any of several menacing conditions exist, such as 'out of paper' or 'out of ink'.

Module J8: This module is divided into two sections with the following functions:

(a) Paper feed: This section causes the paper to advance one line width or two line widths depending on a manually set switch positioned on the console of the printer. This function occurs upon the receipt of the ASCII encoded directive LF, or if the carriage has moved to the right and has not returned to the left within one second.

(b) Indicator control: This section provides visual display of the status of the printer in the form of different light colors.

Module J9: The function of this module is to do the driving of the carriage back and forth upon the receipt of the ASCII encoded directives SOH, LF, and EM.

CHAPTER III

FEATURES OF THE SWTP M6800 MICROPROCESSOR

The Southwest Technical Products (SWTP) M6800 computer system is based upon the Motorola MC6800 microprocessor unit (MPU). The author of this paper assembled the computer system from a kit of parts supplied by Southwest Technical Productions, San Antonio, Texas. This computer system is illustrated in figure 5. The main board of this system consists mainly of the following components, with a brief description of the functions of these components:

(a) MC6800 MPU: This is an eight bit parallel microprocessor with addressing capability of up to 64K words. This microprocessor is self-contained and is capable of operating with virtually any MOS or standard TTL devices.

(b) MC6830 Read Only Memory (ROM): This previously factory programmed 1024 by 8 bit memory words contains a control program known as MIKBUG which acts as the operating systems of the computer.

(c) MC6810 Random Access Memory (RAM): This memory contains a total of 128 by 8 bit words of programmable memory, which can be accessed by the user and is mainly used by the operating system of this computer.

In addition to the above, a control interface is supplied, which can easily connect to a Teletype or other similar terminal to the system. Among the options availbale are memory

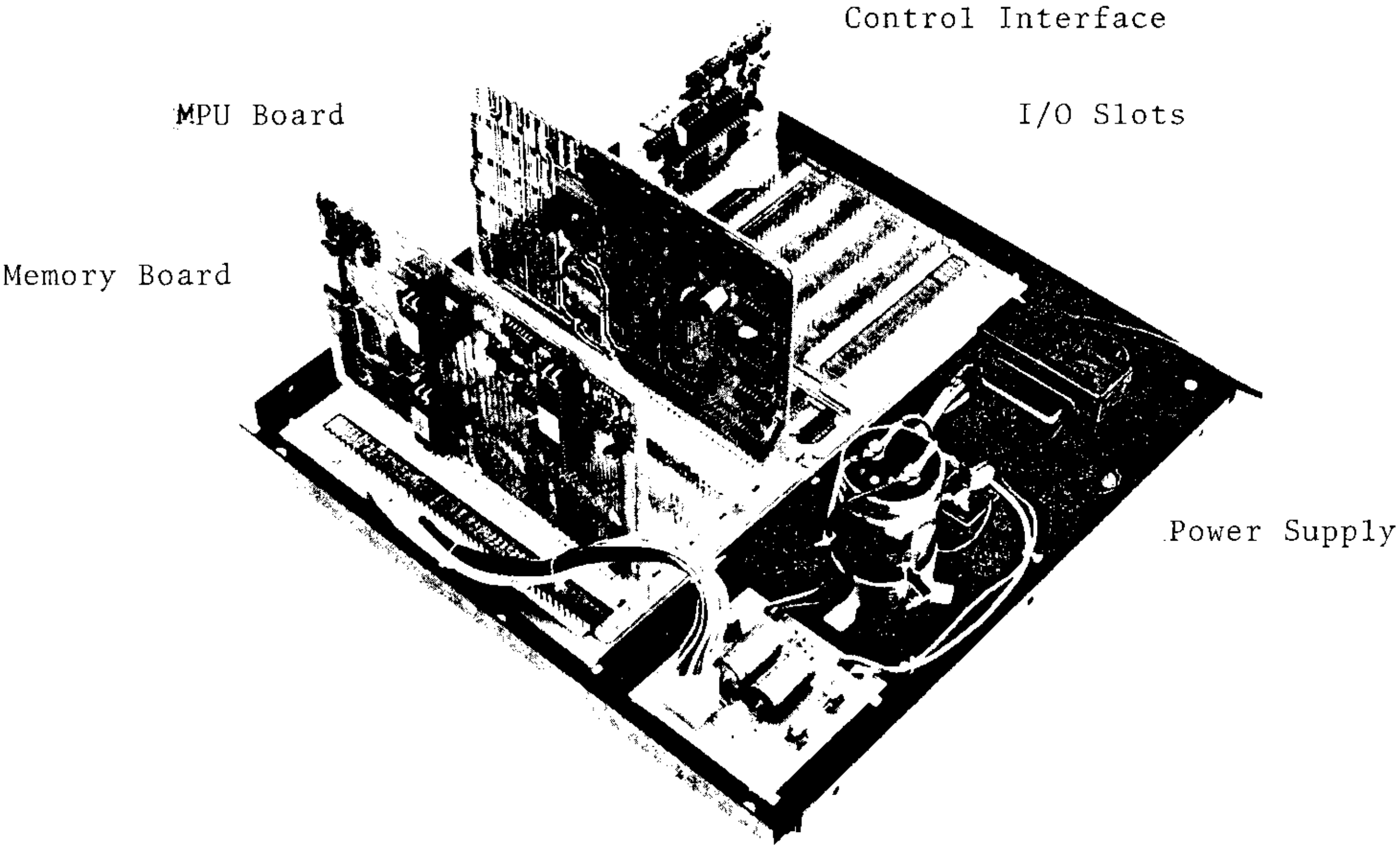


Fig. 5-- The SWTP 6800 computer system source: SWTP Computer Products, page 3.

slots for adding up to a maximum of six boards, thus giving a possible memory total of 24K of eight-bit words. Another option is some Input/Output (I/O) slots which allow adding a maximum of seven more I/O boards which have the ability of connecting some input or output units to the system. These limitations on the number of I/O boards and memory boards are due to the capability of the power supply provided with the computer system.

The different I/O devices that would be added to the system are tied directly to the bus network. This feature allows these I/O devices to respond to the MPU instructions in the same fashion as memory addresses do.

For the I/O boards, the user of this computer system has the option of connecting his I/O devices to the system using one of the following two chips:

(a) MC6820 Peripheral Interface Adapter (PIA): This chip is a programmable general purpose parallel interface device designed mainly to interface the MPU to peripherals requiring parallel data format.

The PIA is separated into two sides, referred to as the A side and the B side. Each side contains three registers as follows:

(i) The control register: This register is an addressable eight-bit register which could be loaded with a word defining the operation of the PIA lines.

(ii) The data directions register: This register is also addressable and defines whether the peripheral data lines are to be used as inputs or outputs with ones in the register designating output bit positions and zeros designating input bit positions.

(iii) The output data register: This eight-bit register temporarily holds the data to be transferred to the peripherals.

(b) MC6850 Asynchronous Communications Interface Adapter (ACIA): This chip is a programmable general purpose serial interface device disigned to interface the MPU to peripherals requiring asynchronous serial data format. The ACIA contains four registers as follows:

(i) The status register: This register is an eight-bit register with each bit concerned with some status of the other registers and the connected peripheral devices.

(ii) The receive data register and the transmit data register: These two eight-bit registers contain the data to be received and the data to be transmitted respectively.

(iii) The control register: This register mainly controls the operation of the receive and transmit data registers.

CHAPTER IV

INTERFACING THE M9601 PRINTER TO THE SWTP 6800 COMPUTER SYSTEM

In interfacing the M9601 printer to the SWTP 6800 computer system, two logic boards, one from the SWTP kit (MP-L parallel interface board) and the other, a self designed printer interface board, were used as follows:

MP-L, Parallel interface board: This board is a parallel interface board using the PIA chip, and can be connected to the computer system by plugging this board in any of the available slots for the I/O boards.

The logic drawing for this board is illustrated in figure 6. A set of elementary logic symbols is given in the appendix. The component parts of the SWTP MP-L board are as follows:

(1) IC1: This chip is the MC6820 PIA chip. As illustrated in figure 6, the left-hand side of the chip is connected to the MPU. It consists of three elements.:

(a) Eight bi-directional data lines labelled DO-D7: These data lines are used to transfer data back and forth between the PIA and the MPU. In this particular design, the ASCII code of the alphabet is what is being transferred from the MPU to the PIA.

(b) Five address lines labelled CSO, CS1, CS2/, RSO, RS1: these lines allow the selection of one of the different six registers within the PIA. (The notation "X/"

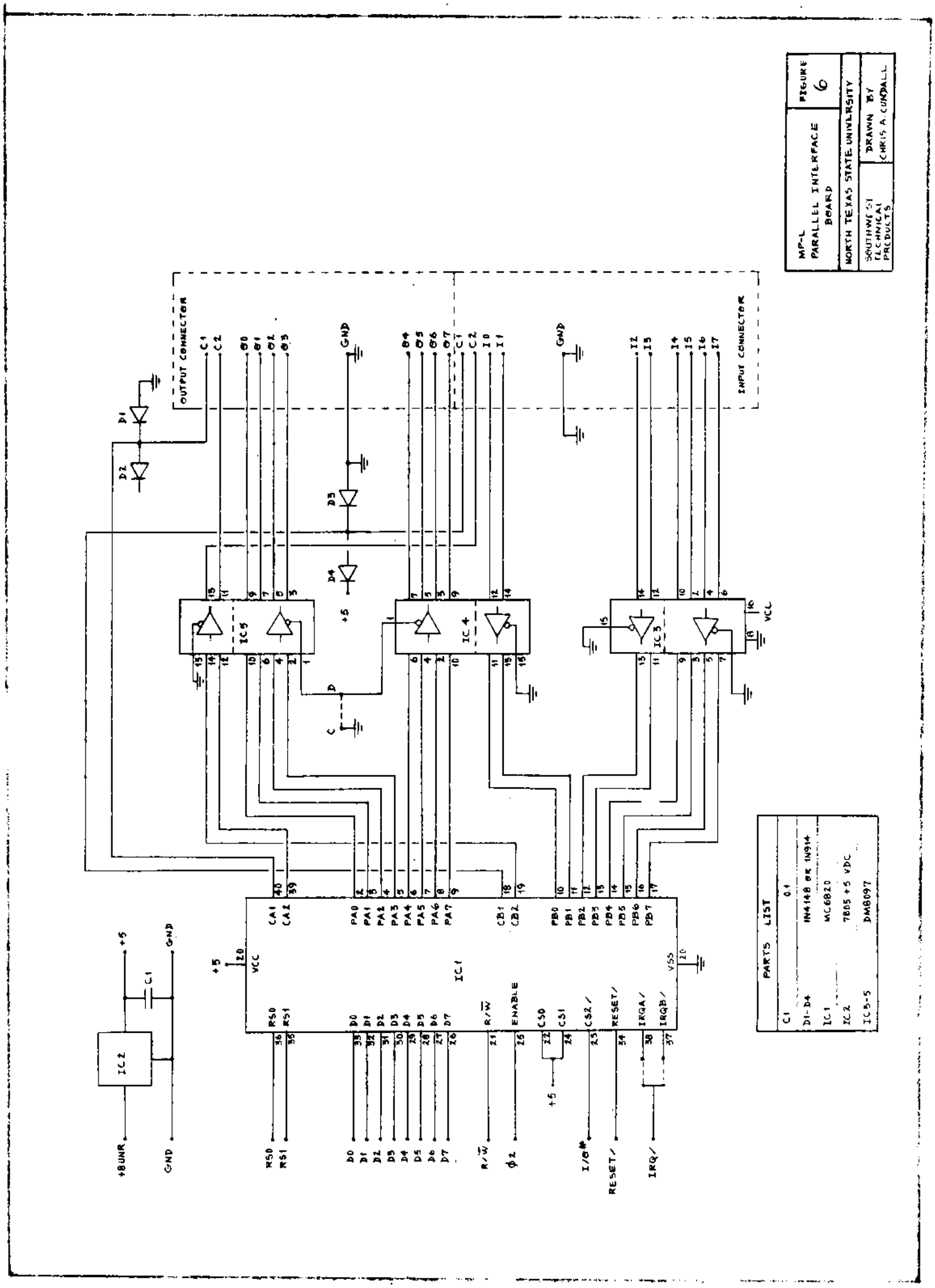


Fig. 6-- The MP-L parallel interface board source: Southwest Technical Products Manual.

means "X" complemented".)

(c) Four control lines labelled R(W/), PHI2, RESET/,IRQ/: R(W/) defines the direction of the data flow between the PIA and the MPU. PHI2 provides the timing signal to the PIA. RESET/ is used to reset the PIA to an all zero condition and IRQ/ causes an MPU interrupt when logically false.

The right-hand side of the PIA consists of the following:

(a) Eight peripheral data lines labelled PAO-PA7: These lines are used to supply data from the PIA to some peripheral unit. In this case, the eight-bit ASCII code is what is transferred to the printer.

(b) CA1 and CA2 : In this particular application CA1 is used as an input to the PIA, signalling the computer of the ready and busy condition of the printer. CA2, in this application, is used as a strobing signal to control the loading of the eight bits of the character code into the buffer register of the printer.

(2) IC2: This integrated circuit is a voltage regulator with the function of producing regulated five volts from the unregulated eight volts of the computer.

(3) IC3 through IC5: These three integrated circuits are hex tri-state buffers through which the eight bits go before being sent to the printer. It is a good idea to have those buffers to obtain a good, strong signal (fast rise time, no reflections) on the other end if a long cable is used to interconnect the computer and the printer.

In this application only the A side of the PIA is used. For this reason, PBO-PB7 are ignored and IC3 is not necessary.

Printer interface logic board: The main function of this board is to accept the codes from the PIA board of the computer, to recognize certain directives, and to transfer the character codes to the buffer register of the J3 module. Using this direct approach bypasses the use of modules J1 and J2 for faster operation and provides a more appropriate application of the code to include codes for the Arabic alphabet.

The logic drawing for this board is illustrated in figure 7. The two directive codes that are important to this application are SOH and EM, however provisions for two extra directives, labelled as EOT and NULL, are provided for optional future use. SOH, which stands for "start of heading", causes the initiation of the movement of the printer carriage, and EM, which stands for "end of message", causes the return of the carriage to its initial position. LF, for "line feed", is not necessary because this function occurs mechanically after the carriage has moved to the right and has not returned to the left after one second. Also, transmitted directives such as NAK, for "negative acknowledge", are not necessary for this application.

If the data received by the printer has neither of these two directive codes values, each bit position is complemented and directed to the corresponding position in the buffer register of module J3, which accepts data in a parallel complemented form. CA2, which is supplied by the computer, is also

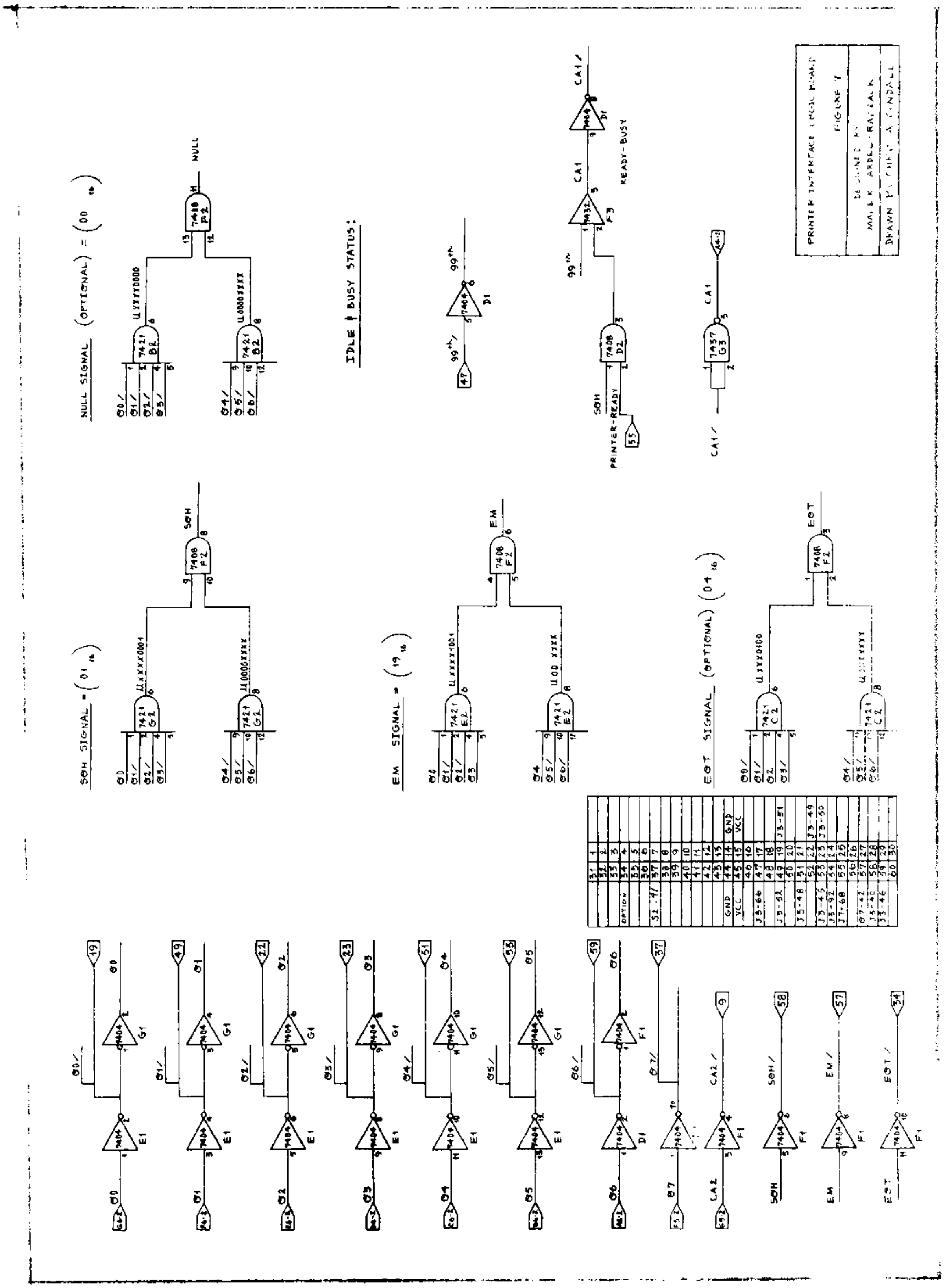


Fig. 7-- Printer interface logic board

complemented and directed to module J3 to act as a register strobe for the character value. CA1 becomes true either initially, when the printer is ready and when first turned on and has already received the SOH signal, or when the printer receives the 99th signal, indicating that the printer has just got through printing the last bit of a character. Control of CA1 by the printer allows it to cause a program interrupt, informing the MPU that the printer is ready for the next character.

CHAPTER V

THE ENGLISH/ARABIC CHARACTER GENERATION

As explained in the previous chapters, each character, when printed, consists of a 9 by 11 (x, y) matrix of tiny droplets, where the x coordinate consists of nine signals, and for each x there are eleven y signals. To obtain an ink droplet at a particular position in the matrix, the x and y signals defining that position must be logically true. Thus, to obtain the signal train for each character, the necessary x and y signal sequences for that character are gated together.

In the original design of the printer electronics, the function of the character generator is to generate all the pulses for all the characters simultaneously. The next step is selecting a particular character. This is done by decoding the ASCII code for the English character set which is illustrated in figure 8. The decoding is done through the use of NAND gates on the different bit positions of the ASCII code to obtain nine lines, labelled as A decoder lines, and eight lines, labelled as B decoder lines, where each character will have only one particular pair of A and B decoder lines corresponding to its ASCII code. The pulse train for each character is ANDed with its A and B decoder lines and the outputs of these ANDs are tied together forming a common output line labelled PC1. Selection of a particular character occurs only

BIT b8	0	0	0	0	0	0	0	0
Values b7	0	0	0	0	1	1	1	1
b6	0	0	1	1	0	0	1	1
b5	0	1	0	1	0	1	0	1
b b b b 1 2 3 4								
0 0 0 0				0	@	P		p
1 0 0 0			!	1	A	Q	a	q
0 1 0 0			"	2	B	R	b	r
1 1 0 0			#	3	C	S	c	s
0 0 1 0			$	4	D	T	d	t
1 0 1 0			%	5	E	U	e	u
0 1 1 0			¢	6	F	V	f	v
1 1 1 0			/	7	G	W	g	w
0 0 0 1			(	8	H	X	h	x
1 0 0 1			)	9	I	Y	i	y
0 1 0 1			*	:	J	Z	j	z
1 1 0 1			+	;	K	[	k	
0 0 1 1			,	<	L	∼	l	
1 0 1 1			-	=	M	]	m	
0 1 1 1			.	>	N	^	n	
1 1 1 1			/	?	O	-	o	

Fig. 8-- The character set ASCII code of the printer (Lower case English letters are shown although the existing printer prints upper case in the place.

when the A decoder line and the B decoder line for that particular character are true, hence, in other words, the ASCII code for that character is provided, and as a result, the signal train for that particular character is passed on to PC1.

The ASCII code is logically defined to provide for upper case letter and lower case letter codes. By changing the value of the sixth bit (b6) from 0 to 1, the lower case of that same letter is obtained. This printer, however, does not produce lower case letters, substituting instead the corresponding upper case letters.

We now consider the design modifications to the original electronics to permit the printing of Arabic characters. These designs were implemented, and as described below, partially checked out.

Examining the ASCII code for the English alphabet shows that the value of bit eight is always zero. This feature is used to derive a fairly similar code for the Arabic alphabet with the main difference that the value of bit eight is 1. This proposed Arabic code is demonstrated in figure 9A. Since there are no lower case and upper case letters in Arabic, this code seems appropriate. It is also important to mention that certain of the Arabic characters are not included in the electronic design for implementing the Arabic character set because these characters already exist in the English character set (such as punctuations and some Arabic numbers), and hence could be shared.

b8	1	1	1	1	1	1	1	1
b7	0	0	0	0	1	1	1	1
b6	0	0	1	1	0	0	1	1
b5	0	1	0	1	0	1	0	1
b b b b 1 2 3 4								
0 0 0 0	ا	ج	ز	ض	ﻎ	ﻟ	ﻳ	،
1 0 0 0	ب	ﺣ	س	ط	ف	م	ي	؛
0 1 0 0	ﺑ	ح	ﺳ	ظ	ﻓ	و	٢	؟
1 1 0 0	ت	ﺧ	ش	ء	ق	ﻧ	٣	
0 0 1 0	ﺗ	خ	ﺷ	ع	ﻗ	ن	٤	
1 0 1 0	ث	د	ﺻ	ﻊ	ن	ﻫ		
0 1 1 0	ﺛ	ذ	ص	ﻏ	ﻛ			
1 1 1 0	ﺟ	ر	ﺿ	غ	ل	ﻻ	ه	

Fig. 9a--The proposed Arabic alphabet code.

The fact that this particular printer uses a 9 by 11 matrix, made it very appropriate to produce the Arabic alphabet, which due to graphic features could not be clearly produced from the very commonly used 5 by 7 dot matrix. The actual dot representation is illustrated in figure 9b.

The logic drawings for the Arabic alphabet decoding and generation are illustrated in figures ten throught fourteen. The same principle of character generation is also used in this case. The logic behind this particular alphabet generation is divided into four sections with the following functions:

(a) Buffers and drivers: The purpose of this section is to receive the 9 x pulses and the 11 y pulses from the printer, buffer them and complement them. These signals are needed in several locations on the logic board, hence, buffering them ensures a strong signal (fast rise time, and no reflections) wherever needed. Figure 10 demonstrates this function.

(b) Character code decoding: This section uses two 9301 integrated circuits (1 of 10 decoders). Each decoder decodes four bits of the character code supplied by the printer in the form of D1 through D7 and D8/. This enables only one combination of their outputs, AA and BB, to define this particular character. This function is illustrated in figure 14.

(c) Character generation: Again, all these Arabic characters are generated simultaneously with the English

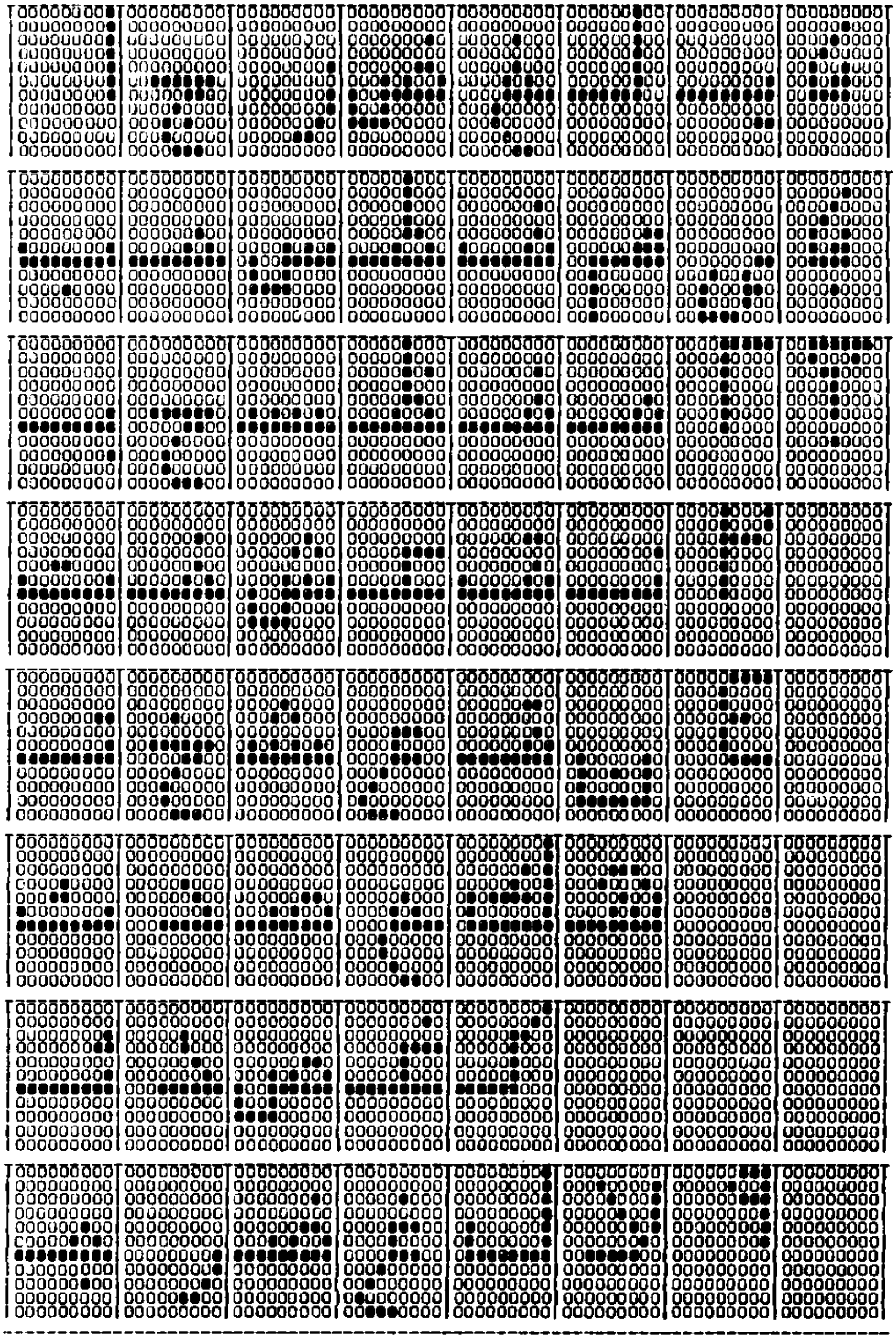


Fig. 9b-- Actual dot representation of the proposed Arabic alphabet.

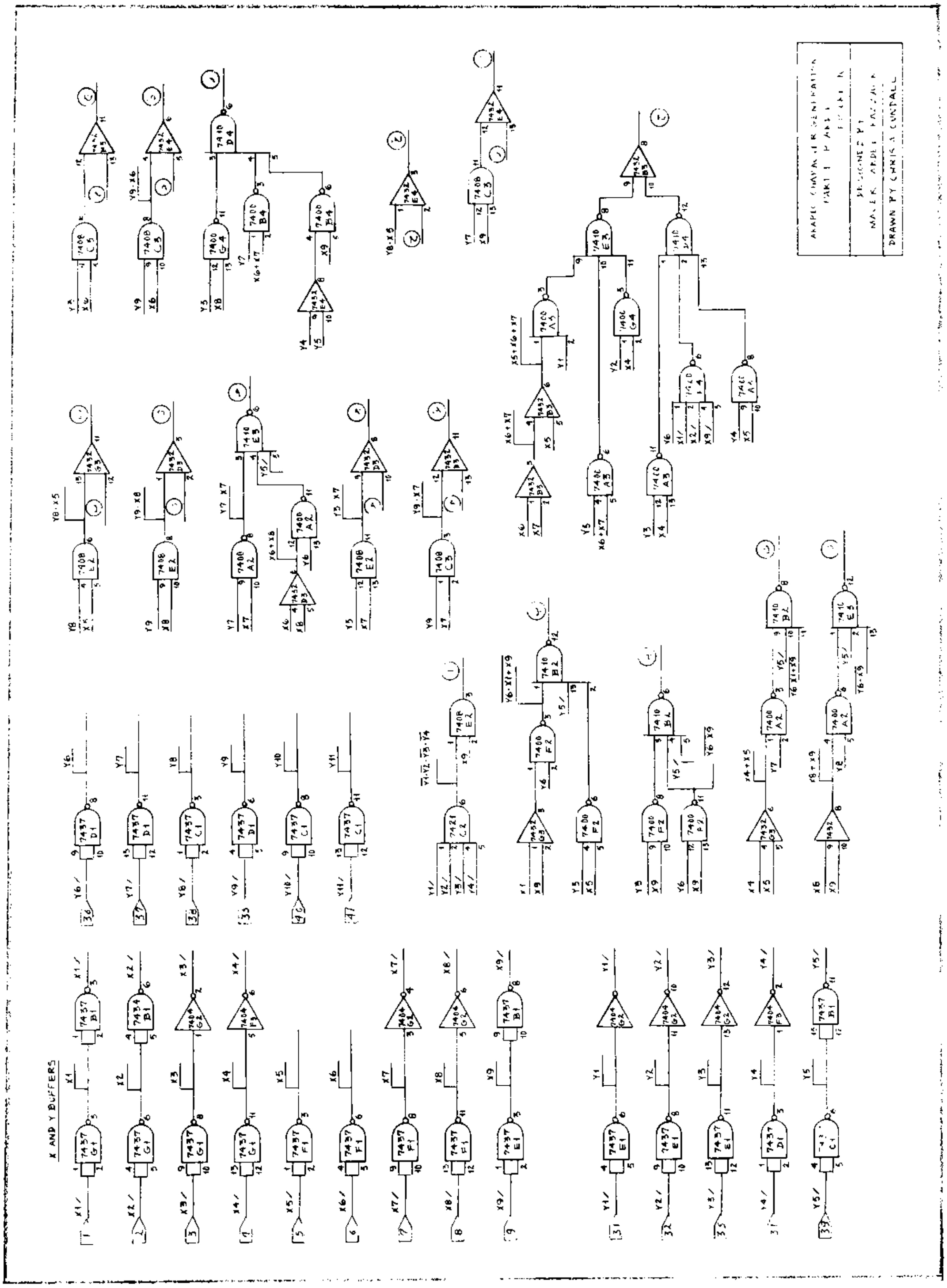


Fig. 10--Arabic character generation, Part 1

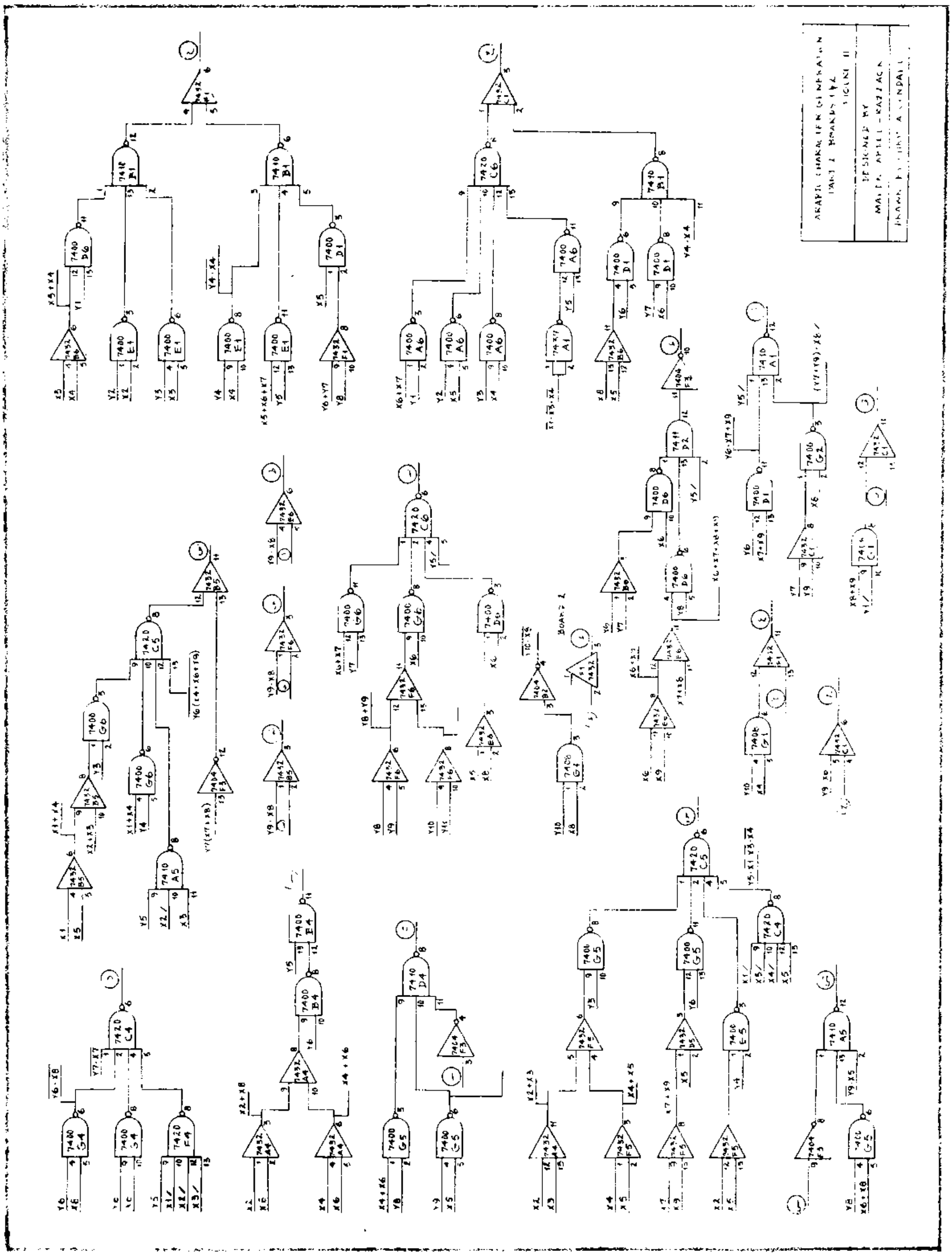


Fig. 11--Arabic character generation, Part II

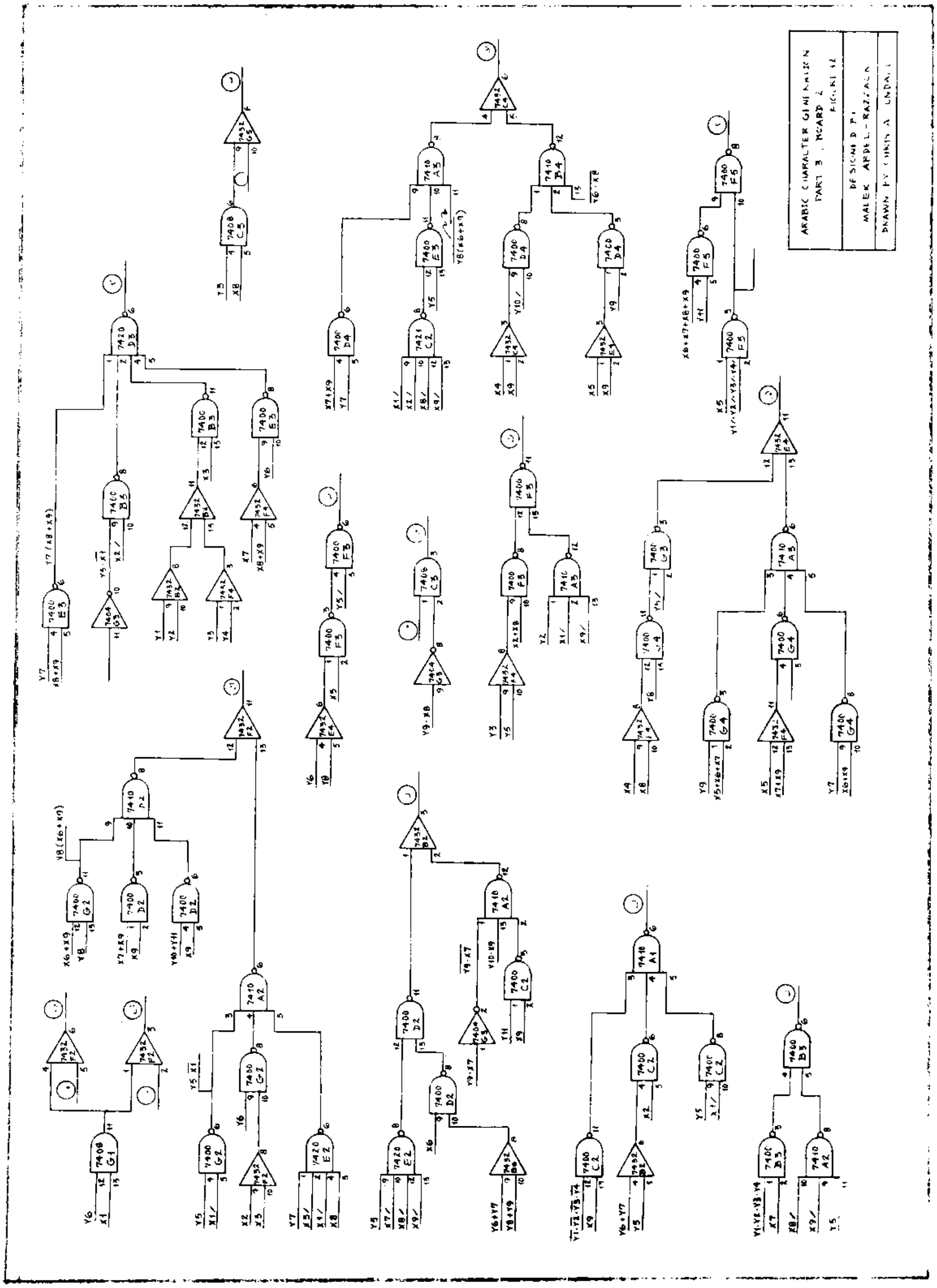


Fig. 12-- Arabic character generation, Part III

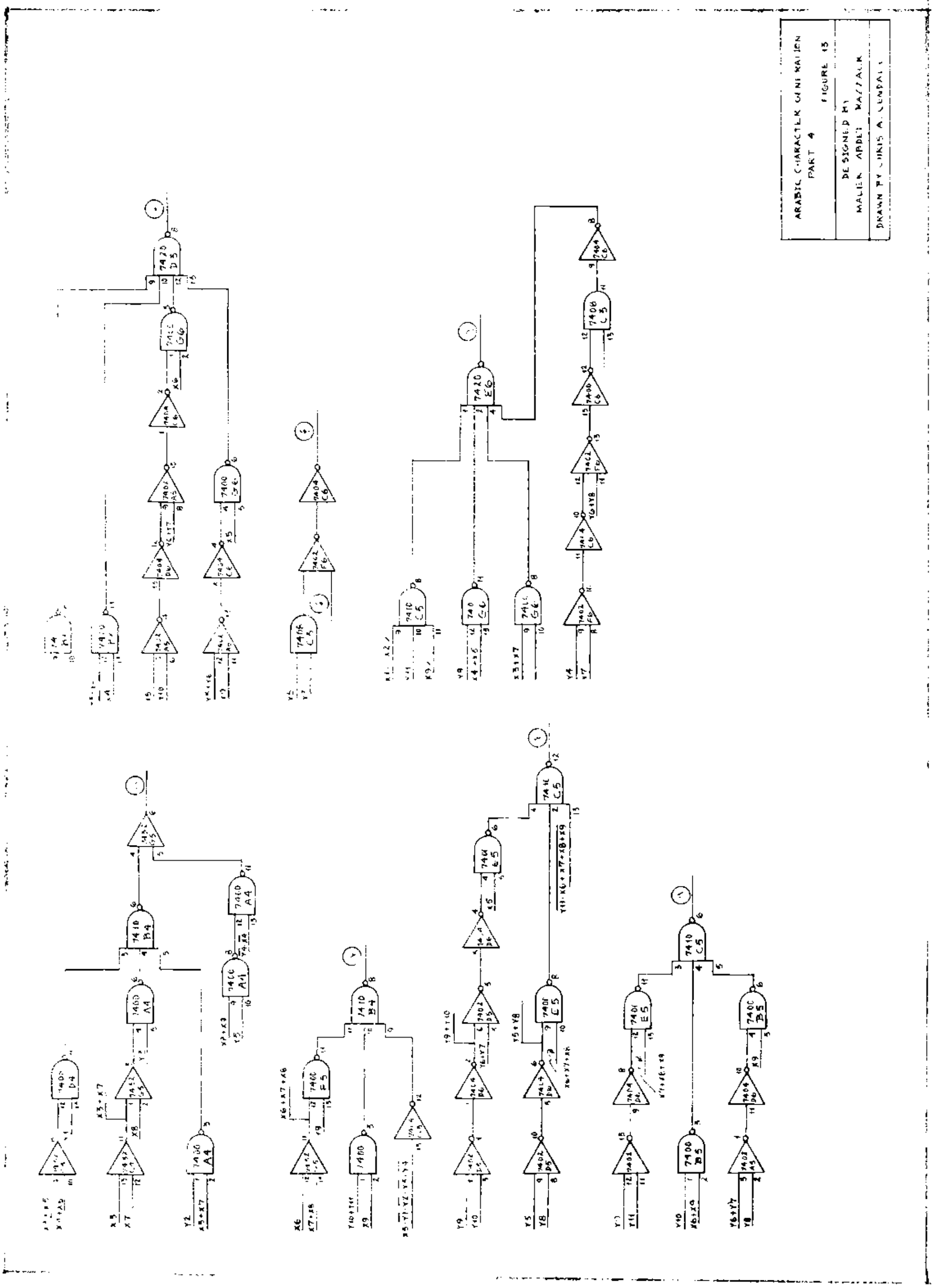


Fig. 13--Arabic character generation, Part IV

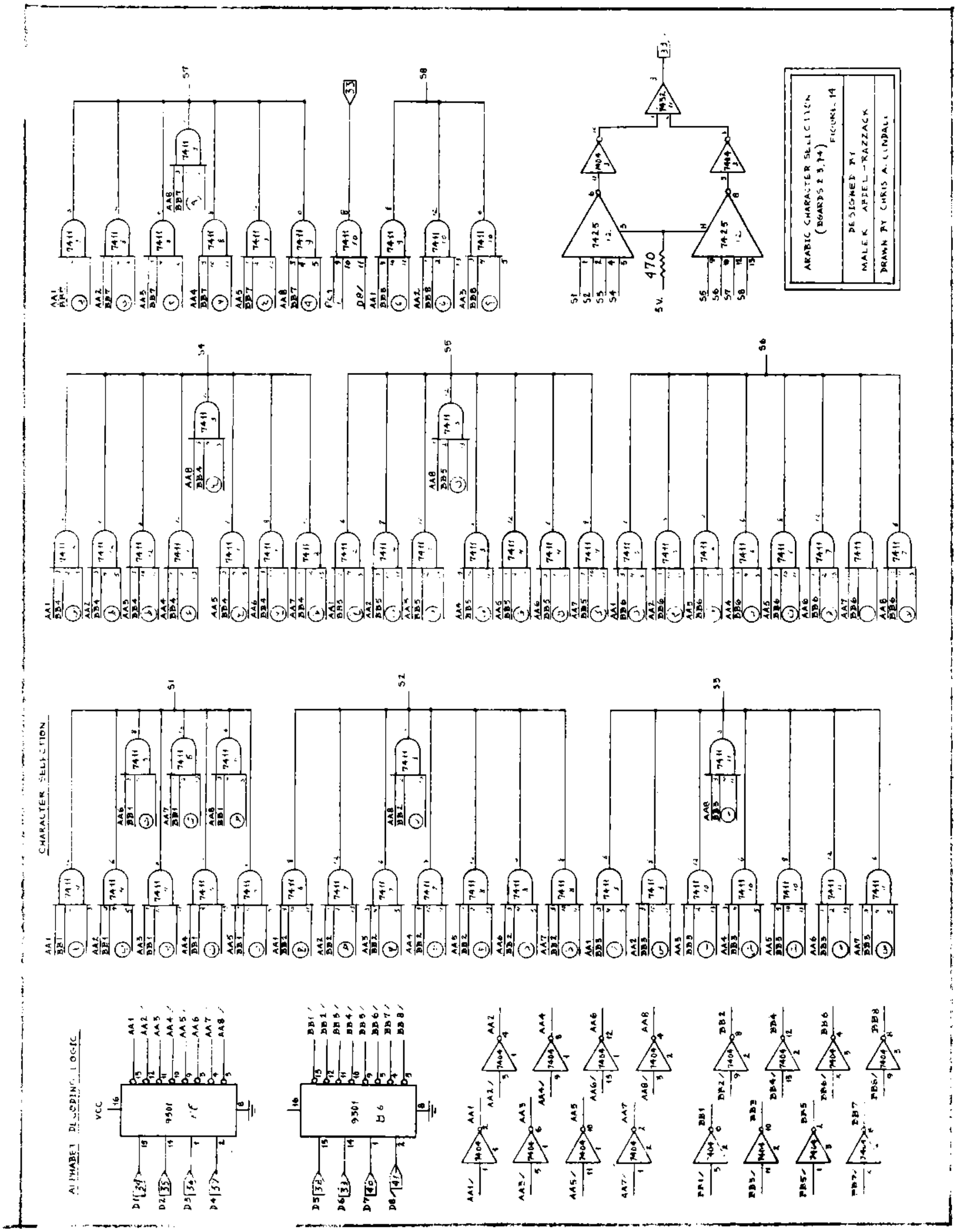


Fig. 14--Arabic character selection

characters. They are also generated by passing the necessarry x and y signal trains corresponding to the required ink droplets in the 9 by 11 dot matrix, through some Small Scale Integration (SSI) gates, mainly NANDs and ORs. The character generation is illustrated in figures 10 through 13.

(d) Particular character selection: The pulse train for a particular character is selected by ANDing the AA decoder line, the BB decoder line, and the pulse train for that particular character together. This allows only that signal train to be sent to the common output train PC1. This function is illustrated in figure 14.

This scheme described creates one problem, however. This problem rises because in the English character set decoding, done by the printer, only seven bits of the character code were used. Bit eight, originally used for parity (error detection), was not used and it was made to have the value 0 always. As a result, in some cases, when specifying an Arabic character with the same first seven bits sequence as an English character, both of their outputs are gated to the common output PC1, thus producing an erroneous character. To overcome this, an AND gate was used in figure 14 to intercept PC1, which is produced from the English alphabet in module J6. The function of the AND gate is to allow the routing of that particular PC1 to module J3, where required, only when D8 is false (has a logical 0 value). If D8 is true (has a logical 1 value), then PC1 is routed from the Arabic character generator, thus solving this particular problem.

This method of implementing the Arabic alphabet was selected for its compatibility with the existing method, for the low cost of the Small Scale Integration gates (composed mainly of NORs and NANDs), and for the availability of these gates.

One more issue needs to be discussed for programming purposes. The printer carriage moves from left to right corresponding to the direction in which English is written, or printed. In the case of Arabic, however, it is written and printed from right to left. This issue is easily solved in writing the computer program for handling the printer device. The service routine normally assigns a memory area, known as a buffer, to keep the characters in a queue, waiting for the printer to be ready. When ready, the characters are delivered to the printer in a First In First Out (FIFO) order. In the Arabic printing mode, the service routine should simply deliver the characters to the printer in a Last In First Out (LIFO) order.

Finally, we consider the current status of the design and implementation. The hardware was assembled as designed in the logic drawings and then connected to the printer. The printer interface board was tested by sending the directives SOH and EM from-the computer to the terminal, and the desired results were obtained, proving the validity of the design. For the character generation and selection, the logic boards were hand checked to show that the desired characters are the ones obtained and selected. The only two things that remain to be done are to obtain printed information in Arabic and English, and develop the required service program. This has not been done yet because of a mechanical failure of the ink nozzle of the printer.

SIGNAL	ORIGIN Board - Pin #	DESTINATION Board - Pin #
00/	A1-19	J3-51
01/	A1-49	J3-52
02/	A1-22	J3-49
03/	A1-23	J3-50
04/	A1-51	J3-48
05/	A1-53	J3-45
06/	A1-52	J3-46
07/	A1-37	B2-41
Vcc	J15-32	A1-15,45; B1-15,45; B2-15,45; B4-1; B3-1
GND	J3-98	A1-14,44; B1-14,44; B2-14,44; B4-3; B3-3
SOH/	A1-58	J3-40
EM/	A1-57	J7-42
99th/	J3-66	A1-47
X1/	J3-68	B1-1
X2/	J3-80	B1-2
X3/	J3-79	B1-3
X4/	J3-82	B1-4

Fig. 15-- Wiring list, Part I

SIGNAL	ORIGIN Board, Pin #	DESTINATION (s) Board - Pin #
X5/	J3-81	B1-5
X6/	J3-84	B1-6
X7/	J3-77	B1-7
X8/	J3-65	B1-8
X9/	J3-67	B1-9
y1/	J3-91	B1-31
y2/	J3-87	B1-32
y3/	J3-88	B1-33
y4/	J3-85	B1-34
y5/	J3-86	B1-39
y6/	J3-83	B1-36
y7/	J3-78	B1-37
y8/	J3-75	B1-38
y9/	J3-76	B1-35
y10/	J3-69	B1-40
y11/	J3-70	B1-41
D1	J3-60	B3-34

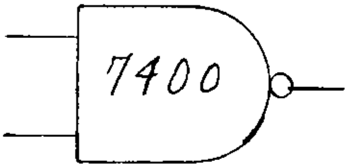
Fig. 16--Wiring list, Part II

SIGNAL	ORIGIN Board - Pin #	DESTINATION Board - Pin #
D2	J3-58	B3-35
D3	J3-57	B3-36
D4	J3-55	B3-37
D5	J3-47	B3-38
D6	J3-53	B3-39
D7	J3-54	B3-40
IPC1	J6-79	B4-34
PC1	J3-74	B4-33

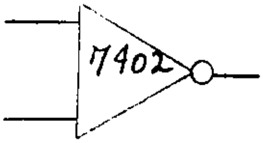
Fig. 17--Wiring list, Part III

APPENDIX I

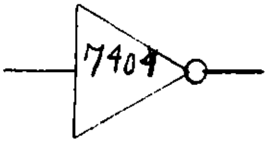
The logic symbols and device part numbers used in the design are:



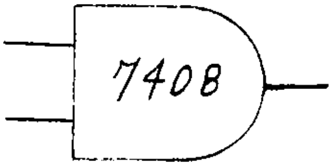
Quadruple 2-input NAND gate



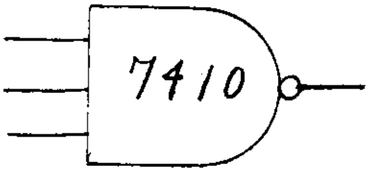
Quadruple 2-input NOR gate



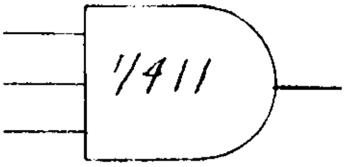
Hex inverter



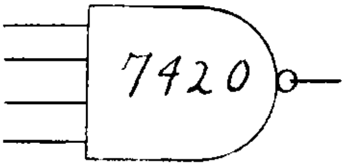
Quadruple 2-input AND gate



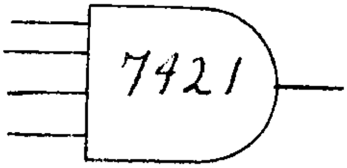
Triple 3-input NAND gate



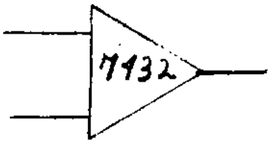
Triple 3-input AND gate



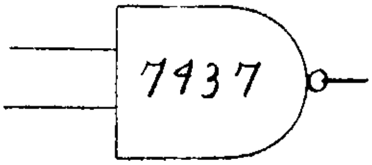
Dual 4-input NAND gate



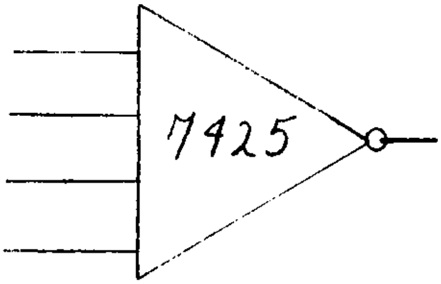
Dual 4-input AND gate



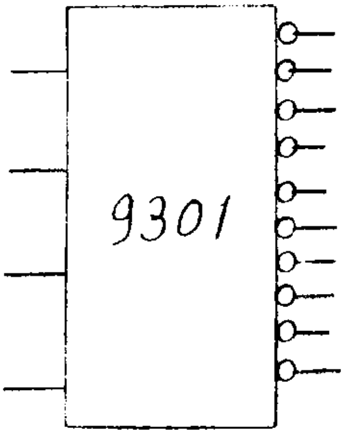
Quadruple 2-input OR gate



Quadruple 2-input NAND gate / Buffer



Dual 4-input NOR gate



1 - of 10 decoder (8 outputs used).

APPENDIX II

THE HARDWARE WIRING LIST

The hardware for the printer interface was assembled on a 60-pin general purpose logic board labelled A1. The hardware for the character generation was assembled on two 60-pin general purpose logic boards labelled B1 and B2, while the character selection hardware was assembled on two 44-pin general purpose boards labelled B3 and B4.

All these boards were connected to the rear of the electronic section of the printer. Bearing in mind that the electronic circuitry of the printer exists on logic boards J1 through J9, the modified wiring list is shown in figures 15 through 17.

BIBLIOGRAPHY

A. B. Dick Company, _Videojet Vertical Forms Control Instructions Manual_, Chicago, Videograph Marketing services, 1971.

A. B. Dick Company, _A. B. Dick Videograph Systems Manual_, Chicago, Videograph Marketing Services, 1971.

Microcomputers Applications Engineering, _M6800 Microprocessor Applications Manual_, Motorola Inc., 1975.

Southwest Technical Products Corp., _SWTP Computer Products_, San Antonio, Texas, SWTP Corporation, 1975.

Fairchild Semiconductors, _The Fairchild Semiconductor TTL Data Book_, Mountain View, Fairchild Semiconductors.