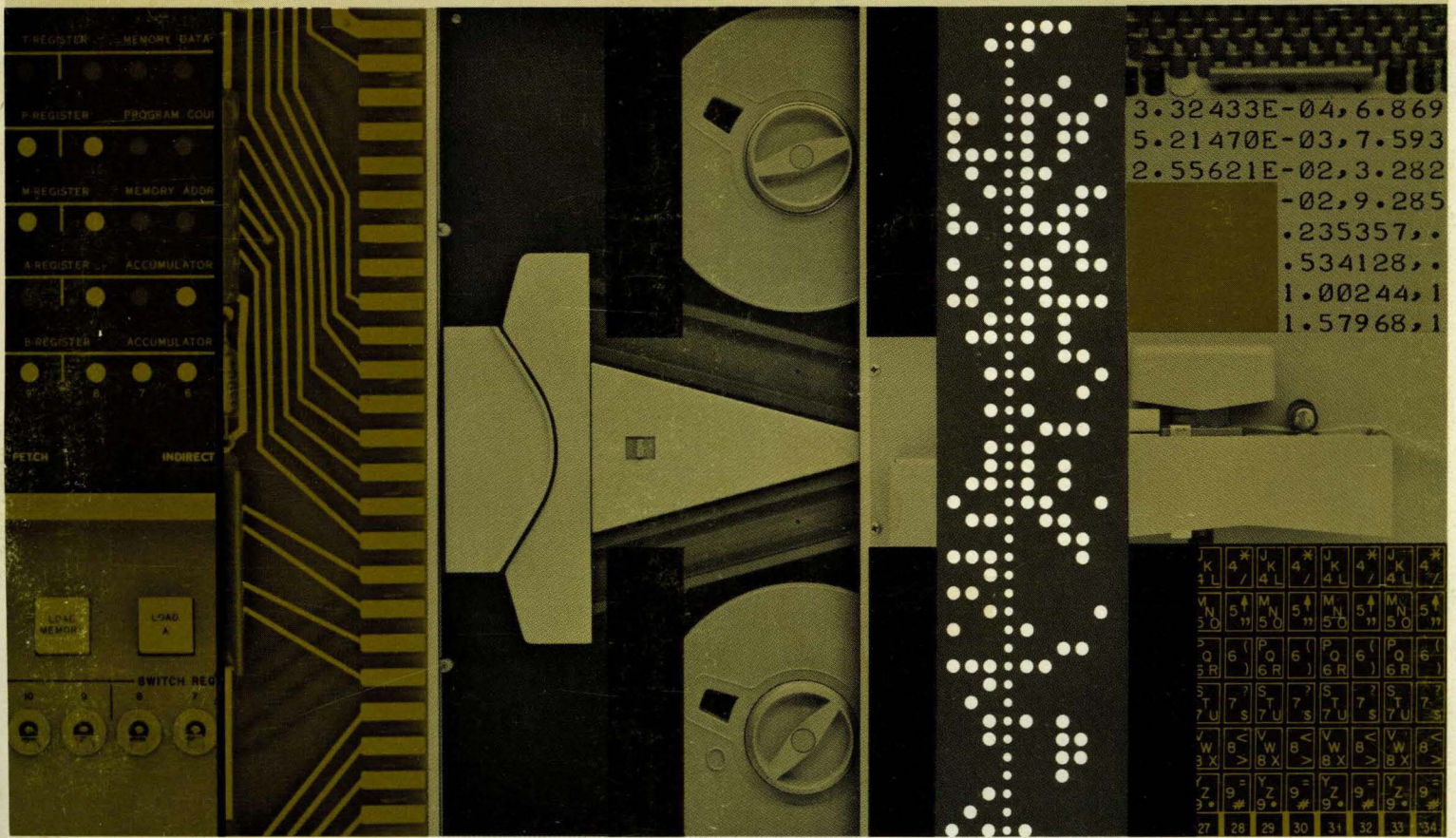


COMPUTER MAINTENANCE COURSE



3.32433E-04, 6.869
5.21470E-03, 7.593
2.55621E-02, 3.282
-02, 9.285
.235357, .
.534128, .
1.00244, 1
1.57968, 1

HEWLETT-PACKARD
COMPUTER MAINTENANCE COURSE

VOLUME IV
STUDENTS MANUAL

DIRECT MEMORY ACCESS

(HP STOCK NO. 5950-8706)

-NOTICE-

The information contained in this manual is for training purposes only. Consult the Hewlett-Packard documentation supplied with the computer for current information concerning the specific computer system furnished.

The information contained in this publication may not be reproduced in any form without the expressed consent of the Hewlett-Packard Company.

COPYRIGHT HEWLETT-PACKARD COMPANY 1969

11000 Wolfe Road, Cupertino, California 95014 Area Code 408 257-7000 TWX 910-338-0221



CONTENTS

FOREWORD

	Page
SECTION I GENERAL INFORMATION	
1-1. <u>INTRODUCTION</u>	1-1
1-2. Description	1-1
1-6. Major Advantages	1-1
1-8. <u>INSTALLATION</u>	1-2
1-9. Interface Kit HP 12578A	1-2
1-12. Installation Procedures	1-2
1-14. <u>FUNCTIONAL OPERATION</u>	1-3
1-15. General	1-3
1-20. Operation	1-4
SECTION II PROGRAMMING	
2-1. <u>INTRODUCTION</u>	2-1
2-2. General	2-1
2-4. Instructions and Control Words	2-1
2-7. <u>PROGRAMMING</u>	2-3
2-8. Program Control	2-3
SECTION III THEORY OF OPERATION	
3-1. <u>INTRODUCTION</u>	3-1
3-2. General	3-1
3-5. Orientation	3-1
3-8. <u>GENERAL THEORY OF OPERATION</u>	3-1
3-9. Operational Block Diagram	3-1
3-13. Overall DMA Operation	3-2
3-20. <u>DETAILED THEORY OF OPERATION</u>	3-5
3-21. General	3-5
3-23. Initialization	3-5
3-32. Input Data Transfer	3-6
3-44. Output Data Transfer	3-8
3-51. Data Transfer Completion Interrupt	3-10

CONTENTS (Cont'd)

SECTION IV DIAGRAMS		Page
4-1.	<u>GENERAL</u>	4-1
4-2.	Description	4-1

ILLUSTRATIONS

Figure No.	Title	Page
1-1	DMA Functional Block Diagram	1-4
2-1	DMA Instruction and Control Word Formats	2-2
3-1	DMA Operational Block Diagram	3-3/4
4-1	DMA Control Card Logic Diagram (Sheet 1 of 2)	4-3/4
4-1A	DMA Control Card Logic Diagram (Sheet 2 of 2)	4-5/6
4-2	DMA Address Encoder Card Logic Diagram	4-7/8
4-3	DMA Register Card Logic Diagram	4-9/10
4-4	DMA Character Packer Card Logic Diagram	4-11/12
4-5	Direct Memory Logic Card Diagram	4-13/14
4-6	DMA Initialization Timing Diagram	4-15/16
4-7	DMA Memory Input Timing Diagram	4-17/18
4-8	DMA Memory Output Timing Diagram	4-19/20
4-9	DMA Interrupt Timing Diagram	4-21/22

FOREWORD

THE HP DIRECT MEMORY ACCESS COURSE

The HP Direct Memory Access (DMA) Course has been developed, under supervision of the Cupertino Division Training Department, to teach service engineers and technicians the basic fundamentals of DMA operation and servicing. This basic training is provided HP customers and/or their certified representatives tuition-free. All necessary course materials and training manuals are also supplied and may be retained by the student.

This course assumes that the student has an elementary understanding of common logic symbology and basic electronics in general. As in any professional endeavor, however, proper and effective execution of the best-planned program requires practice, skill and cooperation. The student is encouraged to study, review and practice the course material until he is satisfied that he has mastered the basic rudiments of DMA operation and servicing.

THE STUDENTS TRAINING MANUAL

The objective of this Students Training Manual is to provide the student with an easily accessible reference manual which provides supplementary reading and study material, and complements the classroom lectures. The material presented in this manual, in general, follows the logical format used in the classroom and contains all the overhead visual slides that will be shown during the course.

The student is cautioned not to use this training manual as an operating or service manual. Those manuals are supplied with the computer documentation provided with all HP computer systems. The student should always consult the proper operating and service manual before attempting the operation, service or repair of any HP computer system. The information contained in this manual is for training purposes only.

SECTION INDEX

GENERAL INFORMATION

I

PROGRAMMING

II

THEORY OF OPERATION

III

DIAGRAMS

IV

general information



SECTION I

GENERAL INFORMATION

1-1. INTRODUCTION

1-2. DESCRIPTION

1-3. The Direct Memory Access (DMA) option enables the computer to transfer data directly between core memory and external devices at a maximum rate of 625,000 (16-bit) words per second, in block lengths from one-to-16,384 words. The DMA system consists of two separate, independent, high-priority control channels (DMA Channels 1 and 2). Either or both DMA Channels may be switched, under program control, between the computer's memory and any device normally serviced through the I/O channels on the computer mainframe or HP2150A I/O Extender Module.

1-4. To be placed in operation the two DMA Channels must first be "initialized" for a specific operating mode by instructions and control words in the main program. Data interchange then occurs automatically over DMA Channel 1 or DMA Channel 2 when a service request command signal is received from an I/O channel programmed to the DMA option. The DMA channel receiving the service request takes control of the Central Processor and I/O System, suspends the running program at the end of the current phase, and during the following machine cycle, generates a special Phase 5 memory cycle to read or write a word directly into or out of a predetermined memory location. At the end of the Phase 5 memory cycle (one complete machine cycle), control is returned to the Central Processor and I/O System, and the main program is automatically resumed at the point where it was suspended, without loss of continuity. A new Phase 5 cycle is initiated each time the I/O Channel signals to DMA that it is ready to input or output another word. When all data in a predetermined block has been transferred, the DMA Channel initiates a normal interrupt to a service subroutine.

1-5. An important feature of DMA is the capability of conserving memory space by "packing" two-8-bit input characters (bytes) into a 16-bit character word format and storing the word in a single memory location. Character words from memory can be divided into two separate 8-bit byte outputs by an "unpacking" process. Character packing/unpacking is a program option that is turned on or off by a single control bit during DMA Channel initialization, and can be used only with external devices that employ 8-bit characters (e. g., HP2737A/B Tape Readers and HP2752A/HP2754A Teleprinters).

1-6. MAJOR ADVANTAGES

1-7. There are several advantages to incorporating the DMA option in a computer system:

- a. Fast Data Transfer - under DMA control, data may be transferred between the computer memory and external devices at the rate of up to 625,000 (16-bit) words per second, which means the range of computer capabilities may be extended to include applications where data is generated at rapid rates and in large quantities. Also, data may be transferred in block lengths from one word to 16,384 words.
- b. Program Independent - data interchange occurs automatically as DMA "steals" memory cycles from the running program to read or write data directly from memory. By merely delaying regular program execution for the number of memory cycles "stolen", we can have regular program control with the fast access feature of the DMA option.

c. Dual Channel - the DMA option includes two separate and independent channels over which data may be transferred between the computer and any two peripheral devices interfaced with the computer. The two DMA channels may be switched, under program control, to serve as many devices as are connected into the computer's I/O system. The same interface is used whether the devices are serviced by DMA or program control. Dual channel operation combined with programmable switching and fast transfer adds versatility and greater workload capabilities to the computer system.

d. Selectable Packing and Unpacking - the DMA logic permits packing and/or unpacking 8-bit bytes automatically. Two bytes can be packed into each 16-bit memory word; thus, two characters of data can be placed in core memory using only one memory location and only one memory cycle. Also, 16-bit words can be unpacked into two 8-bit bytes. The packing feature actually conserves storage in memory and is directly applicable where HP Nine-channel Magnetic Tape Units, HP Teleprinters, or other 8-bit character HP peripheral equipment is used.

1-8. INSTALLATION

1-9. INTERFACE KIT HP12578A

1-10. When purchased as part of the initial computer order, the DMA option is installed in the computer at the factory and is ready for use upon receipt. The DMA option can also be field installed in computer models having serial number prefix 746- or subsequent by making the necessary jumper wire changes noted below, and then equipping the computer with the items supplied in DMA Modification Kit HP 12578A or HP 12578A-M1 (see Table 1-1). Modification Kit HP 12578A-M1 is used for modification of computer models not already equipped with Direct Memory Logic (DML) Card HP 02115-6044. It is also necessary to retrofit the computer with I/O Control Card, Board Revision 822 or later, if this revision is not already included in the basic card complement of the computer. The I/O Control Card is not supplied with either Modification Kit, and must be ordered separately if required.

- NOTE -

When the DMA option is installed in computers having serial number prefix 746- or subsequent, jumper wire connections affecting interrupt priority for select code addresses 6 and 7 must be changed on certain plug-in card assemblies within the computer. Computers having serial number prefixes prior to 746- require factory wiring modifications before the DMA option can be incorporated.

1-11. Table 1-1 lists the PC board complement required to incorporate the DMA option in all HP2115 and HP2116 series computers.

1-12. INSTALLATION PROCEDURES

1-13. To field install the DMA option, proceed as follows:

a. Check the interrupt priority system of the computer in accordance with instructions presented in Volume Three of your computer documentation. Make all jumper wire changes necessary to incorporate select code addresses 6 and 7 in the proper sequence within the interrupt priority network.

b. Check to ensure that the current drain on the power supply will not exceed the limits specified in Volume Three when the DMA option cards are added to the computer system.

c. Set the Computer POWER switch to OFF.

- d. Using the cards supplied in Modification Kit HP 12578A, and the specified I/O Control Card if required, configure the computer as indicated in Table 1-1.
- e. Set the POWER switch to the ON position and observe whether power turn-on is normal. If normal, conduct a performance test of the DMA option. If the test results are normal, the DMA option may now be programmed.

TABLE 1-1. DMA PC BOARD COMPLEMENT

Reference Designation		HP Part No.	Quantity Required or Supplied	Description
HP 2115	HP 2116			
A23	A120	02116-6203	1*	Printed Circuit Board Assembly: DMA Character Packer
A22	A119	02116-6204	1*	Printed Circuit Board Assembly: DMA Control
A21	A118	02116-6205	1*	Printed Circuit Board Assembly: DMA Address Encoder
A122, A123	A116, A117	02116-6206	2*	Printed Circuit Board Assembly: DMA Register
A20	A113	02115-6044	1**	Printed Circuit Board Assembly: Direct Memory Logic
A18	-	02115-6015	1***	Printed Circuit Board Assembly: I/O Control
-	A201	02116-6041	1***	Printed Circuit Board Assembly: I/O Control

*Supplied with Modification Kits HP 12578A and HP 12578A-MI
 **Supplied only with Modification Kit HP 12578A-MI
 ***Board revision 822 or subsequent required, but not supplied with Modification Kits

1-14. FUNCTIONAL OPERATION

1-15. GENERAL

1-16. A DMA Functional Block Diagram is given as Figure 1-1. There are three functional parts to the DMA option hardware:

- a. The Program Control Function: the Address Encoder card and the DMA Control card make up this functional part.

- b. The Word Count and Addressing Function: the DMA Register Cards make up this functional part.
- c. The Packing and Unpacking Function: the DMA Packer Card makes up this functional part.

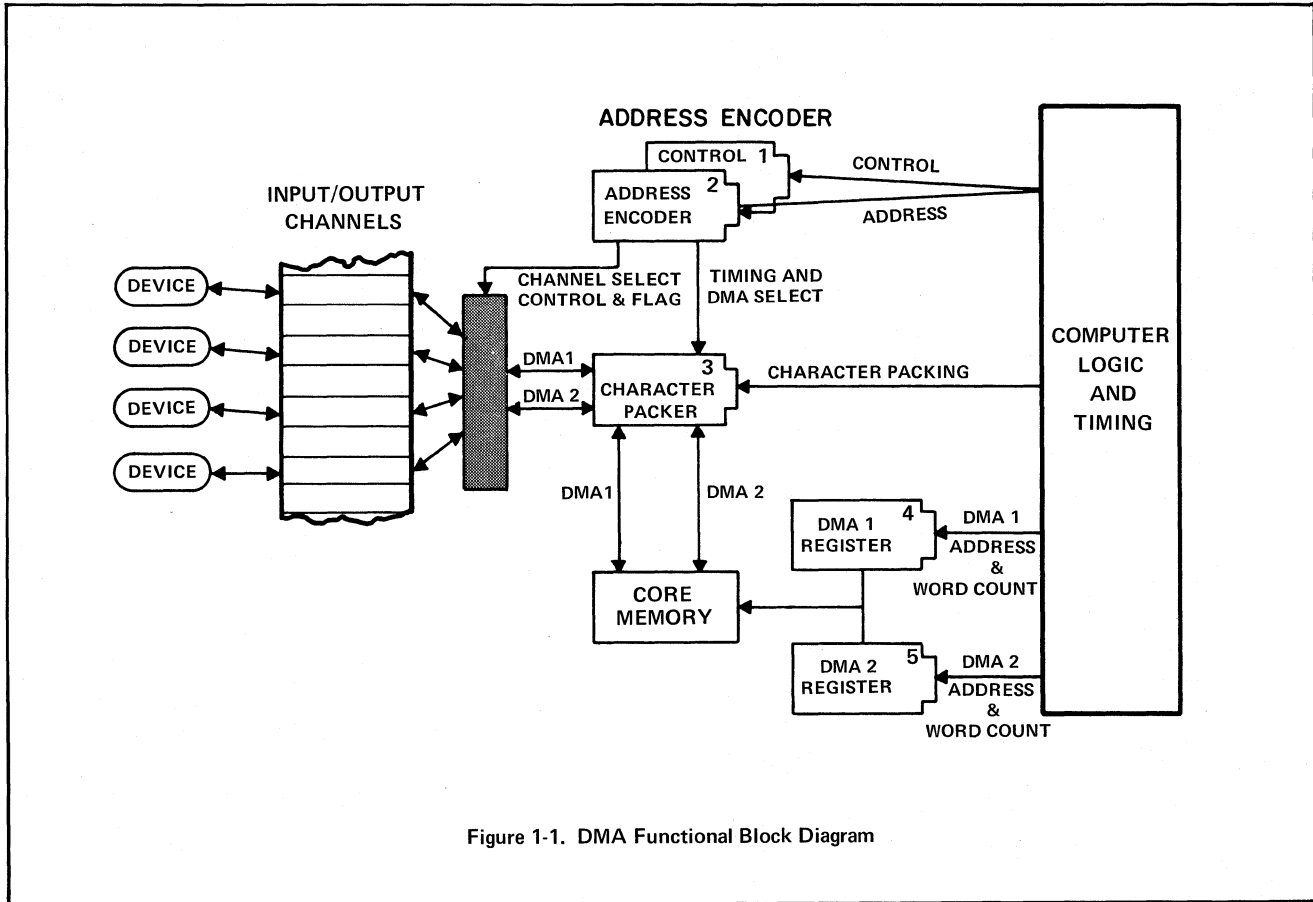


Figure 1-1. DMA Functional Block Diagram

1-17. Basically, the Program Control Function is the heart of the DMA option. It controls all DMA channel switching and furnishes control signals required for Phase 5 operation.

1-18. The Memory Control and Word Count Function is comprised of registers that give DMA access to the computer's memory and provide status and control of the data block length.

1-19. The Packing and Unpacking Function contains registers that permit special processing of 8-bit (byte) characters.

1-20. OPERATION

1-21. The DMA option adds a fifth phase to the normal four phase computer timing. This Phase 5 is a special memory cycle that requires one full machine timing cycle (T0 thru T7). Once initiated by a service request (SRQ), Phase 5 operation is automatic and independent of program control. Each Phase 5 cycle permits one data word (or character byte) to be exchanged directly between an external device and the computer's

core memory. It should be noted that the external device itself may be a memory unit (such as a Memory Disc or Drum).

1-22. Two high-priority DMA Phase 5 channels are shown in Figure 1-1. Both channels are functionally identical and, except that DMA Channel 1 has priority over DMA Channel 2, each operates independently of the other.

1-23. After being initialized by instructions and control words in the main program, the DMA channels operate independently of program control. When a service request is issued, a DMA channel suspends the main program for one machine cycle. While the main program is suspended, the computer is in Phase 5 and data is transferred to or from the external device.

1-24. After the data block has been transferred, a DMA interrupt allows the main program to continue from the point that it was suspended.

programming



SECTION II

PROGRAMMING

2-1. INTRODUCTION

2-2. GENERAL

2-3. The DMA option is programmed using the HP Assembler language. This language is explained in Volume I (HP Computer Maintenance Course). The descriptions in this section assume that the interrupt system has been enabled (STF 00 instruction), and that the DMA Channels will initiate an interrupt to a service subroutine when all words in the assigned data block have been transferred. However, data transfer may also be initiated using the wait-for-flag method (CLF 00 instruction) using the SFS or SFC instructions. This method, as well as the interrupt method, is fully explained in Volume One.

2-4. INSTRUCTIONS AND CONTROL WORDS

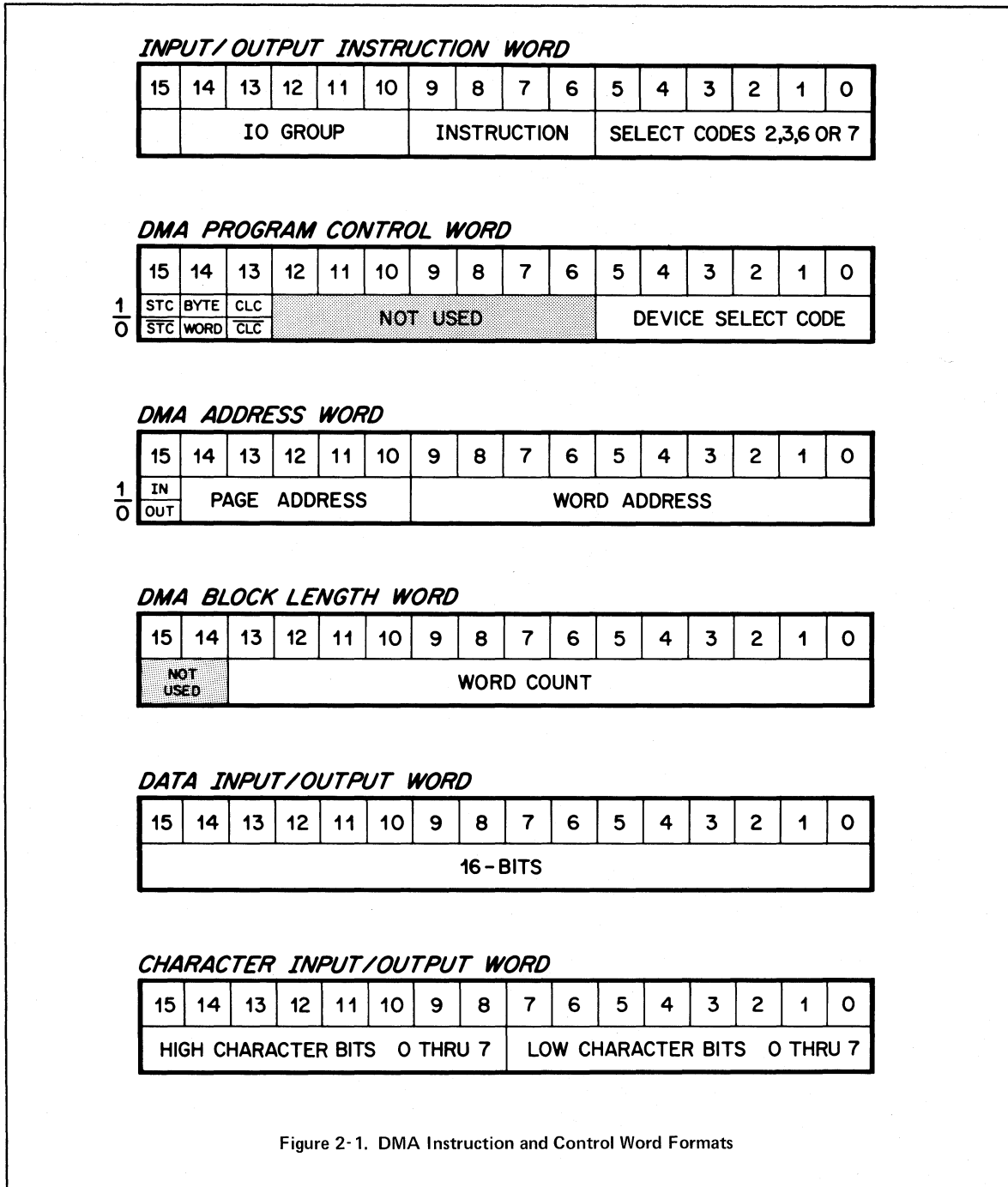
2-5. The DMA instruction and control word formats are shown in Figure 2-1. There are six basic formats to be considered:

a. Input/Output Instruction Words: I/O Group instructions (addressed to select codes 2, 3, 6, or 7) that permit the central processor to control the following DMA functions through the I/O select code addresses specified:

1. Select Code 2 (permits Control FF on DMA Channel 1 Register Card to be addressed by CLC and STC instructions)
2. Select Code 2 preceded by CLC instruction (permits DMA Channel 1 Memory Address Register to be addressed by an OTA instruction)
3. Select Code 2 preceded by STC instruction (permits DMA Channel 1 Word Count Register to be addressed by OTA and LIA instructions)
4. Select Code 3 (permits Control FF on DMA Channel 2 Register Card to be addressed by CLC and STC instructions)
5. Select Code 3 preceded by CLC instruction (permits DMA Channel 2 Memory Address Register to be addressed by an OTA instruction)
6. Select Code 3 preceded by STC instruction (permits DMA Channel 2 Word Count Register to be addressed by OTA and LIA instructions)
7. Select Code 6 (permits DMA Channel 1 switching functions to be addressed by OTA, CLC, STC, STF, SFC, and SFS instructions)
8. Select Code 7 (permits DMA Channel 2 switching functions to be addressed by OTA, CLC, STC, CLF, STF, SFC, and SFS instructions).

b. DMA Program Control Words: program constants that can be programmed to either DMA Data Channel (select code 6 or 7) to specify the following:

1. The I/O Channel select code address of the device to be serviced by the DMA Channel (bits 0 through 5)



2. Clear (turn off) control on device I/O Channel after last word or byte in data block is transferred (bit 13 = 1)
3. Do not clear control on device I/O Channel (Bit 13 = 0)
4. Use character packing mode if memory input transfer, or use character unpacking mode if memory output transfer (bit 14 = 1)
5. Word input/output mode (bit 14 = 0)
6. Set (turn on) control on device I/O Channel after each word or byte in data block is trans-

7. Do not set control on device I/O Channel (bit 15 = 0).
- c. DMA Address Words: program constants that can be programmed to either DMA Control Channel (select code 2 or 3) to specify the following:
 1. Starting memory address for first word of input/output data block (bits 0 through 14)
 2. Memory input from device I/O Channel (bit 15 = 1)
 3. Memory output to device I/O Channel (bit 15 = 0).
 - d. DMA Block Length Words: program constants that can be programmed to either DMA Control Channel (select code 2 or 3) to specify the number of words in data block. Word count is a decimal number expressed as the 2's complement of its positive binary equivalent.
 - e. DMA Input/Output Data Words: conventional data word format used to transfer data directly between the device I/O Channel and memory.
 - f. DMA Character Input/Output Words: word format used to transfer two character bytes, (8 bits) between memory and the DMA packing/unpacking function.
- 2-6. The following paragraphs explain how the instruction and control words are combined to program the DMA option.

TABLE 2-1. DMA CONTROL WORDS

DMA CHANNEL 1 CONTROL WORDS:			
CW1	OCT	160010	Assignment for DMA Channel 1 (ASGN1); specifies I/O Channel select code address (10_8), character byte transfer mode, STC after each byte is transferred, and CLC after final byte is transferred.
CW2	OCT	100200	Memory Address Register Control, DMA Channel 1 (MAR1); specifies memory input operation and starting memory address (200_8)
CW3	DEC	-50	Word Count Register control, DMA Channel 1 (WCR1); specifies the 2's complement of the number of character words in the block of data to be transferred (50_{10}).
DMA CHANNEL 2 CONTROL WORDS:			
CW4	OCT	020022	Assignment for DMA Channel 2 (ASGN2); specifies I/O Channel select code address (22_8), data word transfer mode, and CLC after final word is transferred.
CW5	OCT	010000	Memory Address Register control, DMA Channel 2 (MAR 2); specifies memory output operation and starting memory address (10000_8).
CW6	DEC	-4096	Word Count Register Control, DMA Channel 2; specifies the 2's complement of the number of words in the block of data to be transferred (4096_{10}).
CW7	OCT	140025	Assignment for Disc Memory; specifies a write (bit 15 = 1) on track 40_8 (bits 7 thru 13 = 40_8) beginning with sector 25_8 (bits 0 thru 6 = 25_8).

2-7. PROGRAMMING

2-8. PROGRAM CONTROL

2-9. A typical DMA program is exemplified in Tables 2-1 thru 2-3. The required control words shown in Table 2-1 are formulated using the DMA Program Control word format shown in Figure 2-1. Tables 2-2 and 2-3 list the instructions required for a memory input operation and memory output operation, respectively.

2-10. The first part of this presentation (Table 2-2) lists a memory input operation where it is desired to read a block of 100 bytes (50_{10} words) from a Paper Tape Reader and store the resulting character words (two bytes per word) in computer memory locations 200_8 through 261_8 . The second part of this presentation (Table 2-3) lists a memory output operation where it is desired to transfer a block of 4096_{10} words from computer memory locations $10,000_8$ through 17677_8 to a Disc Memory unit. This program assumes first: that the Paper Tape Reader is assigned to I/O channel select code 10_8 and that DMA Channel 1 will be used for the input transfer; and second, that the Disc Memory Unit is assigned to I/O channel select codes 22_8 and 23_8 and that DMA Channel 2 will be used for the output transfer.

TABLE 2-2. DMA INPUT PROGRAM

LABEL	OP CODE	OPERAND	REMARKS
			INITIALIZE DMA CHANNEL 1:
ASGN1	LDA	CW1	Fetches control word 1 (CW1) from memory and loads it in A-Register.
	OTA	6	Outputs CW1 to DMA Channel 1.
MAR1	CLC	2	Prepares DMA Channel 1 Memory Address Register to receive and store control word 2 (CW2).
	LDA	CW2	Fetches CW2 from memory and loads it in A-Register.
	OTA	2	Outputs CW2 to DMA Channel 1.
WCRI	STC	2	Prepares DMA Channel 1 Word Count Register to receive and store control word 3 (CW3).
	LDA	CW3	Fetches CW3 from memory and loads it in A-Register.
	OTA	2	Outputs CW3 to DMA Channel 1.
			START (STR1) DEVICE AND CHANNEL:
STR11	STC	10B,C	Initiate paper tape reader data transfer.
	STC	6B,C	Activate DMA Channel 1.
			DATA TRANSFER DMA CHANNEL 1:
:	:	:	Continue program while data transfer takes place.

TABLE 2-3. DMA OUTPUT PROGRAM

			INITIALIZE DMA CHANNEL 2:
ASGN2	LDA	CW4	Fetches control word 4 (CW4) from memory and loads it in A-Register.
	OTA	7	Outputs CW4 to DMA Channel 2.
MAR2	CLC	3	Prepares DMA Channel 2 Memory Address Register to receive and store control word 5 (CW5).
	LDA	CW5	Fetches CW5 from memory and loads it in A-Register.
	OTA	3	Outputs CW5 to DMA Channel 2.
WRC2	STC	3	Prepares DMA Channel 2 Word Count Register to receive and store control word 6 (CW6).
	LDA	CW6	Fetches CW6 from memory and loads it in A-Register.
	OTA	3	Outputs CW6 to DMA Channel 2.
ASGN23	LDA	CW7	INITIALIZE DISC MEMORY: Fetches control word 7 (CW7) from memory and loads it in A-Register.
	OTA	23B	Outputs CW7 to I/O Channel 23 _g .
			START CHANNEL AND DEVICE:
STRT2	STC	7, C	Activate DMA Channel 2.
	STC	22B	Initiate Disc Memory data transfer.
			DATA TRANSFER DMA CHANNEL 2:
.	.	.	Continue program while data transfer takes place.
.	.	.	
.	.	.	

theory of operation

SECTION III

THEORY OF OPERATION

3-1. INTRODUCTION

3-2. GENERAL

3-3. This section provides a general theory of operation for the DMA circuitry followed by a detailed theory of operation. The general theory of operation is based on the DMA Operational Block Diagram given in Figure 3-1. This diagram shows the main circuit functions comprising the DMA option and illustrates the relationship between the DMA hardware, the Central Processor Unit and the computer I/O system.

3-4. The detailed theory of operation is based on the logic diagrams, timing diagrams and logic equations given in Section IV. Unless otherwise specified, both the general and detailed theory descriptions are based on the programming information provided in Section II and assume that the program presented in Tables 2-1 thru 2-3 is in progress.

3-5. ORIENTATION

3-6. All required logic diagrams and timing diagrams necessary to follow the descriptions given in this section are located in Section IV. Figures 4-1 thru 4-5 diagram the DMA option hardware. Figures 4-6 thru 4-9 provide essential timing diagrams for understanding the initialization, control and data transfer operations using the DMA option. Table 4-1 then provides a glossary of all logic signal mnemonics shown on the logic and timing diagrams. This table provides a definition, the function and the source for each signal referenced on the diagrams.

3-7. Because of the DMA option's complexity, each logic card comprising the DMA hardware will not be discretely described. Rather, a DMA operation will be defined as consisting of four phases:

- a. Initialization
- b. Input Data Transfer
- c. Output Data Transfer
- d. Data Transfer Completion Interrupt

It should be noted, however, that either b or c above may be eliminated depending upon program requirements. All phases of DMA operation (listed above) will be described in the following paragraphs. This approach, rather than the discrete circuit approach, will serve to facilitate understanding the inter-relationships existing in the DMA hardware and will provide a more direct and simple path to understanding the overall DMA circuitry.

3-8. GENERAL THEORY OF OPERATION

3-9. OPERATIONAL BLOCK DIAGRAM

3-10. The DMA Operational Block Diagram is given as Figure 3-1. The following paragraphs implicitly reference this diagram. It must be noted that on this diagram, the mechanical switches are used to represent logical switching functions only and are not discrete components.

3-11. The DMA option adds a fifth phase to the four phase capability of the basic computer. Phase 5 is a special memory cycle that requires one machine timing cycle (T0 through T7). Once initiated by service request signals received from I/O channels programmed to DMA, Phase 5 operation is automatic and

independent of program control. Each Phase 5 cycle permits one word or character byte to be exchanged directly between an external device (tape reader, disc memory, magnetic tape unit) and the computer's memory.

3-12. The two high-priority DMA Phase 5 control channels are shown functionally on Figure 3-1. Both channels are functionally identical and, except for the fact that DMA Channel 1 takes priority over Channel 2, each operates independently of the other. After being initialized by instructions and control words in the main program, the DMA channels operate independently of program control. When a service request is received by a DMA Channel, it "suspends" the main program for one machine cycle to achieve data transfer, rather than by interrupting to a service subroutine. While the program is suspended, the computer is in Phase 5 and data is transferred to or from the external device. At the end of the Phase 5 cycle, the main program (delayed by one machine cycle) continues from the point that it was suspended, since the program registers in the Central Processor are not stepped during a Phase 5 cycle. A DMA interrupt occurs only after the Word Count logic signals that all words in a data block have been transferred.

3-13. OVERALL DMA OPERATION

3-14. The DMA channels are tied into the computer I/O addressing scheme through select codes 2, 3, 6 and 7. This is shown functionally by the Select Code switch in the I/O System. The Select Code switch is positioned by the six select code bits (0 through 5) contained in an input-output instruction word format (see Figure 2-1). The DMA select code assignments permit the DMA channels to be initialized by input-output (I/O Group) instructions and control words from the running program in the Central Processor. As shown by Figure 3-1, the channel functions of DMA Channel 1 are addressed through select codes 2 and 6, and those of DMA Channel 2 are addressed through select codes 3 and 7. Select codes 2 and 3 are normally called control channels while 6 and 7 are called data channels.

3-15. The DMA Channels are tied directly into the I/O System priority chain through select code addresses 6 and 7. Address 6 has the third highest priority in the I/O System (only Power Fail and Memory Protect are higher), and address 7 has the fourth highest priority. This gives DMA Phase 5, and DMA interrupt (FLG and IRQ lines), priority over all I/O Channel functions. It follows also that the two DMA channels cannot both operate at the same time, since DMA Channel 1 takes priority over DMA Channel 2. Addresses 2 and 3 are used for addressing purposes only and are not tied into the I/O priority system.

3-16. The DMA channels are tied to the Central Processor through the I/O bus output (IOBO) lines, I/O bus input (IOBI) lines, T-Register (TR) lines, direct memory address (DM) lines, the input/output (I/O group) instruction control lines, and the Phase 5 control lines. Initializing control words from memory are routed to the R-bus via the A or B Registers and are switched on to the IOBO bus by the Input/Output switching logic in the Central Processor. The control bits on the IOBO lines are then routed to the addressed DMA Channel logic by the Select Code switching logic in the I/O system. Note that the IOBO lines are also used to output data words to the I/O Channels when DMA is not in operation.

3-17. During Phase 5 output cycles, the Input/Output switching logic and Select Code switching logic are controlled by the Phase 5 control signals. Data words from memory are then routed to the DMA Channel directly from the T-Register via the TR lines. The data bits on the TR lines are switched on to the IOBO lines by the Service Select switching logic within the DMA Channel, and routed to the external device through the interface card located in the I/O Channel. The TR lines are routed through the Packing/Unpacking logic when character input/output words from memory are processed. For each character word received from memory the Unpacking logic outputs two separate 8-bit character bytes to the external device. Unpacking requires two Phase 5 cycles. The first Phase 5 cycle reads the character output word from memory, stores the low character bits in the Unpacking logic register, and outputs the high character bits to the external device. The second Phase 5 cycle outputs the low character bits to the external device.

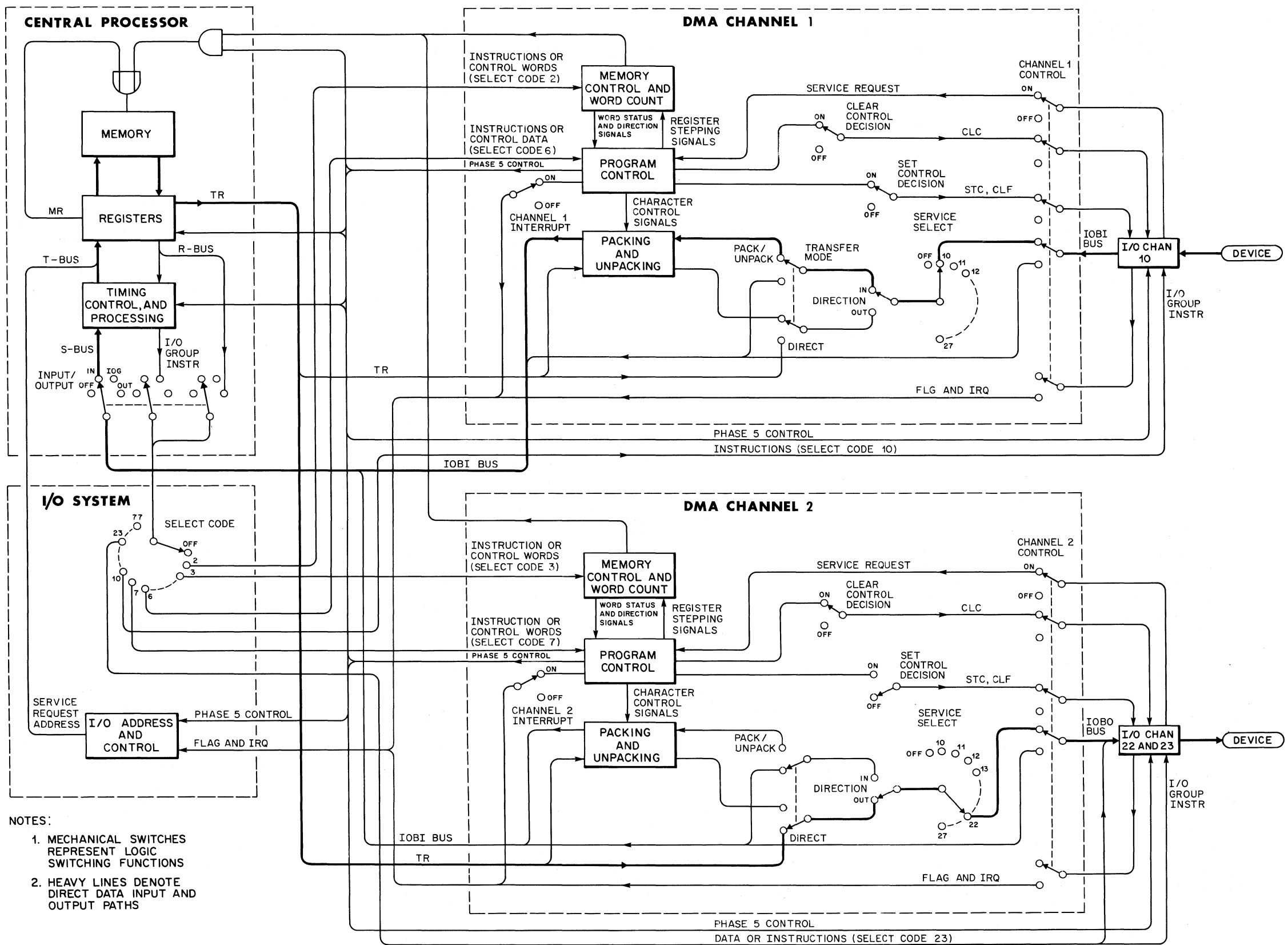


Figure 3-1. DMA Operational Block Diagram

3-18. During Phase 5 input cycles, the Input/Output switching logic and Select Code switching logic are also controlled by the Phase 5 control signals. The IOBI lines from the I/O Channel are now under control of the Service Select switch in the DMA Channel. The data bits on the IOBI lines are routed either directly to memory via the S-bus, T-bus, and T Register which are now under Phase 5 control, or in the case of character byte inputs, first through the Packing/Unpacking logic before being read into memory. Packing of input character bytes also requires two Phase 5 cycles. The first Phase 5 cycle transfers the high character bits into the Packing logic register where they are stored. The second Phase 5 cycle inputs the low character bits and packs them, together with the high character bits, into the character word format and transfers the character input word directly into memory.

3-19. The Phase 5 control signals take complete control of the Central Processor and I/O System during Phase 5 input and output cycles. They effectively turn-off the I/O System during Phase 5 to prevent the I/O Channels from interrupting a DMA Phase 5 data transfer. They also control the operating mode of the Central Processor so that direct data paths are established between the I/O Channels and the computer's memory. In addition, they permit the DM lines from the Memory Control logic register to take over control of memory access from the M-Register (MR) lines during Phase 5. The Phase 5 control signals are referred to as "pseudo" signals because they have the same mnemonic labels and share the same lines in the backplane wiring as similar signals generated in the Central Processor or the I/O system. Pseudo signals perform essentially the same functions as their counterparts, but they originate in the DMA option circuits.

3-20. DETAILED THEORY OF OPERATION

3-21. GENERAL

3-22. This section provides a detailed description of DMA Operation. A DMA operation is defined as a four-phase running program. The four phases are:

- a. Initialization
- b. Input Data Transfer
- c. Output Data Transfer
- d. Data Transfer Completion Interrupt

This section provides a detailed description of each phase and attempts to explain the hardware logic necessary to successfully complete each phase. The programming terms and instructions used are contained and explained in Tables 2-1 thru 2-3. All necessary logic and timing diagrams are located in Section IV.

3-23. INITIALIZATION

3-24. During this discussion, DMA Channel 1 will be initialized for a memory input data transfer. The input device (Tape Reader) is assumed to be assigned Select Code 1 \emptyset . Reference Figures 4-1 thru 4-5 and Figure 4-6 during the following discussion.

3-25. Each time the computer is turned on, the POPIO and CRS signals are applied to the DMA Control Card (Figure 4-1), the DMA Address Encoder Card (Figure 4-2) and the DMA Register Card (Figure 4-3). The CRS signal clears the Control FF's 6 and 7 (CTF6 and CTF7) and the Transfer Enable FF's 1 and 2 (TEF1 and TEF2) located on the DMA Control Card. The POPIO signal applied to the DMA Address Encoder Card clears the Cycle Request FF's 1 and 2 (CRF1 and CRF2). On the DMA Register Cards 1 and 2, CRS clears the Control FF's 2 and 3 (CTF2 and CTF3), respectively. The DMA channels must be re-initialized whenever power is turned off and on.

3-26. The first step of initializing DMA Channel 1 is to have the main program perform a LDA instruction with CW1 and then an OTA with select code 6 (see Table 2-2). This routes CW1 to the Address Encoder Card (Figure 4-2) and the DMA Control Card (Figure 2-1). On the DMA Address Encoder Card, the lower six (6) bits of the DMA Control Word (CW1) are clocked into the Service Select Register FF's (SS10 thru

SS15) by IOG(B), IOO and SCM 0/SCL6. The Service Select Decoder decodes the SCM and SCL signals for the selected device with gates MC31A thru MC61B. Gates MC22A thru MC112A decode the SCL signals (for either DMA Channel) of the selected device. Gates MC54B, MC76A, MC74B and MC76B decode the SCM signals (for either DMA Channel). This select code is decoded and enables the Service Request Network (MC33C thru MC74A). Then, when the selected device sends a Service Request Signal (SRQ), and no higher priority device has interrupted, the Cycle Request FF (CR1) can be set at time period T6.

3-27. Bit 14 of CW1 specifies byte packing or unpacking. If bit 14 is true, it is clocked into the Character Mode FF (MC11A) by IOG(B), IOO and SCM0/SCL6. The output of the Character Mode FF enables gates MC16A thru MC46A, and gates MC17D thru MC37A, to output signals to implement packing/unpacking on the Character Packer card. The Character Mode FF also enables the Character Register stepping signals out of this network.

3-28. On the DMA Control Board (Figure 4-1) the SCM0/SCL6 and IOG(B) signals make the MC46D gate output a true signal. This true output is "anded" with IOO to set either the STC Decision FF or the CLC Decision FF, depending on the value of bits 13 and 15 of CW1. These decisions will enable the transfer signals and send the Set Control or Clear Control signals to the selected I/O device. These signals are optional and their state depends upon the device used with DMA.

3-29. The next step of initialization consists of loading the Memory Address Register with the starting address where the data is to be stored or retrieved. The Control FF (MC43), on the DMA Register Card (Figure 4-3), must be set or reset to load the Word Count or Memory Address respectively. The main program must issue a CLC instruction at select code $\emptyset 2$ to allow gating of the memory address into the Memory Address Register. The memory address contained in CW2 is loaded and output from the A Register. The gates MC72A thru MC83B directly set all flip-flops in the Memory Register that have a 1 bit on the IOBO lines. When PH5 is set, these signals will be gated to memory via the DM lines and the Direct Memory Logic card. This memory register is a counter and its contents will be incremented by each transfer operation. Bit 15 of CW2 is the direction bit and it controls the Direction FF MC33. This flip-flop determines whether DMA is outputting or inputting words from the computer's memory. (In this example, bit 15 is a 1 for an input operation.)

3-30. The third step of DMA initialization consists of loading the Word Count Register on the DMA Register Card with the block length. The Word Count Register is enabled by a STC instruction with Select Code $\emptyset 2$. Then the control word CW3 is loaded and output from the A Register to set the count. Before transfer of a word this register, holding the 2's complement of the word count, is incremented to decrease its value by one. When the counter is zero, the Overflow FF (MC51B) is set to generate a word count rollover. This signal enables the flag and interrupt signals to signal that the DMA channel operation is complete.

3-31. The last step of initialization consists of turning on DMA and the external device. The external device is turned on by a STC, CLF instruction programmed to its select code. The next STC, CLF instruction is programmed to select code 6. This sets the Control FF (MC84D), enables the DMA Interrupt, sets the Transfer Enable FF and enables setting the Cycle Request FF when a SRQ signal is received (see Figure 4-1) the STC6 instruction also clears the Cycle Divide FF (MC67A) initializing it to control the register stepping signals (see Figure 4-2).

3-32. INPUT DATA TRANSFER

3-33. The DMA Option may use direct word inputs to the computer (bit 14 = 0) or it may exercise the byte mode (bit 14 = 1). The byte mode is used exclusively with 8-bit character devices. This description assumes that the byte mode has been selected. Reference Figure 4-7 during the following discussion.

3-34. Anytime after initialization the selected device may have information available. At this time the input device's Flag FF (on the device interface card) is set and generates a SRQ signal. On the Address Encoder Card (Figure 4-2), the SRQ signal is enabled through the network decoder gates MC33C thru MC74A by the

output of the Service Select Register via MC14. If the PRL signal from Memory Protect is true, the TE1 signal from the Transfer Enable FF is true and at T6 the SRQ sets the Cycle Request FF MC106A (Figure 4-2). The output of MC97B is made false to disable DMA Channel 2.

3-35. The Cycle Request FF sends a true CR1 signal to enable the character in/out gates MC37C thru MC47C and the Cycle Divide FF (MC67A). These circuits control the register stepping signals. In the byte mode, the Word Count Register (Figure 4-3), must step only after the second character is available for transfer. The true CR1 signal also goes out of pin 34 to the DMA Control card (Figure 4-1). Here it sets the PH5 FF (MC25) at T7S of the current machine phase. The DMA channel goes into PH5 operation for the next machine cycle (T0-T7).

3-36. When PH5 is set the central processor and I/O system are under DMA control exclusively. Since byte packing requires two Phase 5 operations, a memory write cycle will occur on even Phase 5 operations. On even PH5 operations, the low order bits will be processed. During odd PH5 operations, the higher order bits are fed into the Character Packer Register. The PH5 control signals do the following:

- a. The true PH5 control signals disable the MR lines on the DML Card (Figure 4-5), through inverter MC76, and enable the DM lines from the Memory Address Register. This provides the Memory Address Register access to the computer memory to specify where the first character will be stored.
- b. The PH5 signal enables a false EPH (enable phase) signal to the computer's STG Card to block any phase decoding.
- c. The PH5 signal generates a true P123 signal to the STG card to enable the memory read and write timing signals, and to generate a true RST signal from the STG card.
- d. The PH5 signal sends a false P123 to the ERS card to prevent possible decoding of the data on the T Bus as an addressable register command.
- e. The PH5 signal prevents the I/O Address Card from generating an interrupt by clearing the Interrupt Control FF on the I/O Control Card to give a false ESR signal. It also disables any I/O Select Code decoding. During PH5 the HIS signal from the Address Encoder Card disables any interrupts from any device in the I/O system. The PH5 signal disables all signals, except timing signals and instructions from the computer, by enabling the false EPH signal to the STG card.
- f. The PH5 signal enables a true IIR signal to prevent any instruction decoding by the computer.

Due to the above control functions, the main program is now suspended and the I/O interrupt system is disabled. The true PH5 signal is also "anded" with the CR1 FF output and, via MC116B, enables the output of the Service Select Register to generate the SCM and SCL signals to the selected device (see Figure 4-2). Now DMA will send all the necessary control signals to this select code.

3-37. The first byte is now in the Buffer Register on the device's interface card. Meanwhile, on the DMA Control Card, the necessary "pseudo" control signals are being generated. At time period T2, the ENF signal is made true to enable the flag signals. Also, the PH5 and CR1 signals are enabling IOI, IOCI, ISG, ADF and SWST. These control signals do the following:

- a. The IOI and IOCI signals enable the IOBI lines in the computer so data on the IOBI lines can get on the S Bus.
- b. The signal ISG goes to the STG card to disable reading of core by making MST false during an input operation. This is specified by the Direction FF (DIN1) being true.
- c. The ADF signal is the function necessary to transfer the data from the S Bus onto the T Bus.
- d. The SWST signal enables the (STBT) signal in the computer to get the data into memory.

The DMA hardware must now store the first character into the Character Packer Holding Register

(Figure 4-4). This character will occupy the high order 8 bits.

3-38. At this point in time, data is on the IOBI lines to the Character Packer Card (Figure 4-4). Also, the Cycle Divide FF is reset and the CR1 FF is set on the Address Encoder Card (Figure 4-2) until time period T6. Since the Character Mode FF (MC11A) is set for byte mode operation, a true IHC1 signal is generated via MC17D. This true IHC1 signal gates the first character into the Character Packer Holding Register (Figure 4-4).

3-39. At time T3T4, another STC signal must be sent to the input device interface card to load another character. With the Word Count rollover signal (WCR1) false on the DMA Register Card (Figure 4-3), and the Cycle Request FF (CR1FF) still set on the Control Card (Figure 4-1), the STC Decision FF is set. Gate MC83A then outputs a true STC signal to the interface card. At T4T5 the Control Card also issues a CLF signal to clear the flag on the interface card. When the Flag FF on the device interface card is reset, the SRQ signal is removed from the SRQ decode network on the Address Encoder Card (Figure 4-2). With a false SRQ signal input, the Cycle Request FF is cleared at T6. Clearing the CR1 FF allows setting of the Cycle Divide FF. With the Cycle Divide FF set it enables generation of the stepping signals, but the now false CR1 prevents their generation. The false CR1 signal enables the PH5 FF to reset at T7S of this current Phase 5 operation.

3-40. Resetting the Phase 5 FF allows control of the computer and I/O system to resume under main program control because all DMA "pseudo" instructions are now false. DMA Channel 1 remains static while the main program is in progress. The next SRQ signal is received from the selected I/O device when its Flag FF is set. The Cycle Request FF (CR1) is set at T6. The CR1 signal sets PH5 FF at T7S of the current phase. The PH5 FF again generates the "pseudo" control signals to give DMA control of the computer and I/O system.

3-41. During the time that the CR1 FF was reset in the first PH5 cycle the Cycle Divide FF (MC67A) was set. When the CR1 FF is set by this SRQ signal again (the second), it enables the transfer signals to gate the high order bits onto the IOBI lines out of the Character Packer Register and the IOI "psuedo" signal gates the low order bits onto the IOBI lines. A true Cycle Divide FF CR1 FF, IOI, and Character-Mode FF, makes MC72C (on the Address Encoder Card) output a true IDW1 signal to strobe the high order bits onto the IOBI 8-15 lines. The low order bits are now on the IOBI 0-7 lines from the device's interface card.

3-42. After completion of each byte transfer, the Memory Address Register must be stepped to the next address. The true Cycle Divide FF and CR1 FF enable the CIN signal to the DMA Control Card to generate register stepping signals. The M Register bits are now inhibited by the PH5 signal. The PH5 signal enables the DM lines to the Memory Address Decoder card in the computer. The CIN, CR1, PH5 and DIN1 signals are combined on the DMA Control Card to generate the SMAR1 signal to step the Memory Address Register (trailing edge) and to gate the previously stored address onto the DM lines (leading edge).

3-43. The above process (para. 3-33 thru 3-42) is repeated until the Service Request signal for the final byte is received. The Word Count Register (Figure 4-3) is incremented just before the final character is transferred. This Word Count Register holds the 2's complement of the block length and is set to zero at the end of the block length. When rollover to zero occurs, the Overflow FF (MC51B) is set and generates a true WCR1 signal. This signal is routed to the Control Card where it inhibits generation of the STC and CLF signals.

3-44. OUTPUT DATA TRANSFER

3-45. As was the case for input data transfer, the output data transfer descriptions assume that the byte mode has been selected. The following discussion also assumes that the output device is programmed for DMA Channel 2 (see Table 2-3). In most respects, initializing a DMA output operation is the same as for an input operation. The essential differences will be pointed out. The following paragraphs reference Figures 4-1 thru 4-5 and Figure 4-8.

3-46 The main program loads the A Register with the program control word CW4. Then the control word is output to the DMA channel with an OTA instruction (Table 2-3). The CW4 control word sets the STC or CLC Decision FF on the DMA Control Card according to the value of bit 15 and 13, respectively (as for an input operation). CW4 also sets the Transfer Mode FF for byte mode operation (see Figure 4-1). The main program then issues a CLC to select code 3 to enable the DMA address word (CW5) to be programmed to the Memory Address Register (Figure 4-3). This sets the starting address of the data block into the Memory Address Register. The direction bit is a zero for output so the Direction FF (MC33A) is cleared. Meanwhile, CW4 has set the select code of the device to be used into the Service Select Register. The select code bits are clocked into the register as during an input operation. Next a STC is programmed to select code 3 to enable the Word Count Register to receive the Block Length Word. A LDA with CW6 and OTA to select code 3 loads the 2's complement of the number of words to be transferred into the Word Count Register. If the output device requires any special initializing signals more control words must be fetched from memory and routed to the device. The final step is to turn on the DMA channel and the I/O device by programming a STC, CLF to their respective select codes. DMA is now initialized for the output operation and the main program is kept running until a Service Request is sent to DMA.

3-47. When the device is ready to accept a character byte it sets the Flag FF on it's interface card. The Flag FF sends a SRQ signal to the DMA channel. The Cycle Request FF (CR2) is then set at T6 (see Figure 4-2). The CR2 signal sets PH5 at T7S of the current cycle and enables the packing/unpacking and stepping signals (see Figure 4-8). The true PH5 signal enables signals to the computer and I/O system to do the following:

- a. Inhibit the M Register bits on the MR lines and all other registers in the computer by generating a false EPH signal from the DML card (Figure 4-5).
- b. Strobe the memory address bits stored in the Memory Address Register onto the DML lines with the SMAR2 signal from the DMA Control Card. This enabling signal is (CR2) (PH5) ($\overline{\text{DIN2}}$) (COUT).
- c. Disable the I/O System with the true PH5 and HIS signals.
- d. Enable the memory timing circuits on the STG card with the true P123 signal from DML.
- e. Control the operation of the selected device by generation of DMA "pseudo" control functions.

3-48. The T Register Output lines are routed directly to the Character Packer Card (Figure 4-4). Two Phase 5 operations are necessary to unpack a 16-bit computer word. During the first PH5 operation: the true CR2 and the false Cycle Divide 2 FF makes gate MC47B output a false signal and MC66B output a true signal to give the true signal COUT from the Address Encoder Card (Figure 4-2). Then the true COUT signal is "anded" with PH5, CR2 and $\overline{\text{CIN2}}$ to give the true OUT signal from the DMA control Card (Figure 4-1). During the read time of the PH5 cycle the output of memory is put into the T Register by DMA enabling the read timing circuits with DMA "pseudo" control signal P123. The T Register lines are then routed to the Character Packer Card. The upper eight bits of the word are sent through the buffer gates MC56A - MC126A and the lower bits are sent to the buffer gates MC17A - MC47A. Now the gating signal OHC2 must become true to enable the lower bits into the unpacking register and the upper bits thru gates M65A-C and MC115A-C. This OHC2 signal is routed from the Address Encoder Card by the following signals:

- a. CM2FF
- b. CR2FF
- c. CD2

The CM2FF is set by the byte operation bit 14. The CR2FF is set by SRQ. $\overline{\text{CD2}}$ is true until CR2 is reset at T6 and the IOO and OUT signals are true. At T4T5 OHC2 gates the upper 8 bits out onto the IOBO

lines 0-7 and stores the lower 8 bits in the Character Packer Register until the next Phase 5 cycle. At the beginning of PH5, the Memory Address Register was enabled to the DM lines by the true SMAR2 signal. This SMAR2 signal is true when CR1FF, PH5FF, DIN2 and COUT are true. The address on the DM lines is the location of the first word to be output by DMA. At the trailing edge of SMAR2 the Memory Address Register is bumped 1 count. The Word Count Register is not incremented during this cycle because the stepping signal CIN is false.

3-49. At time periods T3T4 and T4T5 the DMA Control Card (Figure 4-1) issues the STC and CLF signals. These signals prepare the output device to accept another data byte. After reception of the character by the device another SRQ signal is sent to DMA. The SRQ signal sets PH5 at the end of the current phase and the PH5 control signals are again generated. It is now necessary to gate the low order bits stored in the Character Packer Register to the IOBO lines 0-7. The Word Counter must also be incremented. The necessary control signals are generated as follows:

The Cycle Divide FF was set when the CR2FF was reset at T6 of the previous PH5 cycle. With the CDF2 FF set, and CR2FF now set by the SRQ signal, the stepping signal for the Word Counter (CIN) is true. Therefore, the Word Counter is now incremented. The gating signal OLC2 required to transfer the low byte out of the Character Packer Register is now true because the CR2FF, CD2FF and the CM2FF, and the delayed IOO signal are all true on the Address Encoder Card (Figure 4-2). The bits in the Character Packer Register will be gated onto the IOBO bus lines 0-7. The DMA "pseudo" control signal (IOO) gates the data into the I/O interface card's Buffer Register. The OLC2 signal is delayed to make sure that the data is on the lines when the clock ends. This eliminates any possibility of dropping data bits.

3-50. The STC and CLF signals at T3T4 and T4T5 are once again generated to prepare the device to receive another character. At the end of this PH5 cycle, control is again returned to the main program. This procedure repeats itself until the Word Count Register (Figure 4-3) reaches zero. At this time the Word Count Overflow FF (MC51B) is set and sends a true WCR2 signal to the DMA Control Card (Figure 4-1). When the WCR2 signal comes true it initiates the completion interrupt for select code 7 by setting the Flag Buffer FF (MC45A/55B). This initiates the standard interrupt sequence and also clears the Transfer Enable FF to effectively turn DMA off. The WCR2 signal also enables the CLC Decision FF's output to turn the output device off if desired. At the end of the current PH5 cycle (plus one cycle) PH4 is set.

3-51. DATA TRANSFER COMPLETION INTERRUPT

3-52. The following paragraphs describe the DMA interrupt phase which is required at the end of data transfer to return I/O system control to the computer. These descriptions reference Figures 4-1 thru 4-5 and Figure 4-9.

3-53. The WCR1 and WCR2 signals (see para. 3-50) are used to enable the DMA interrupt phase. These signals are routed to the DMA control Card (Figure 4-1) and do the following:

- a. Inhibit generation of any further STC and CLF signals.
- b. Enable the CLC signal at T4T5 (via MC83B) to turn off the external device. This inhibits any SRQ signals until the device is re-initialized.
- c. At T6 of the current Phase 5, the CR1FF (or CR2FF) is cleared which allows the I/O System interrupt control to be enabled during the next Phase 1 machine cycle.
- d. Enable the generation of DMA Channel 1 "flag" and "interrupt" signals. With the transfer Enable FF (MC84C) set and the WCR1 (or WCR2) signal true the Flag Buffer FF MC45A/MC55B is set. This resets the TE1 FF at T4T5. At T2 (ENF), the Flag FF6 is set. With the Interrupt system enabled (IEN), and the Control FF1 and Flag FF6 outputting true signals, the IRQ FF is set at T5.

3-54. The FLG 0 and IRQ6 signals are sent to the I/O Address Card in the computer. The "flag" and IRQ signals try to generate the "interrupt" signal but the I/O Control Card disables the interrupt system for at least one phase to allow execution of at least one main program instruction following a Phase 5 operation. At the end of this machine phase, DMA can interrupt. The computer is set to PH4 and the DMA trap cell location is forced into the M Register. The JSB for DMA completion is the next fetched instruction. The last character in the data block is transferred by generation of the IHC1, IDW1, IOI and other control signals as any other character is transferred. In the DMA completion subroutine, DMA is turned off and must be re-initialized before being placed into operation again.

diagrams

SECTION IV

DIAGRAMS

4-1. GENERAL

4-2. DESCRIPTION

4-3. This section provides the logic and timing diagrams referenced throughout the theory of operation descriptions given in Section III. This section also provides a glossary of all logic mnemonics given on the diagrams (see Table 4-1). This glossary provides a definition, the function and the source for each logic signal.

THIS PAGE INTENTIONALLY BLANK

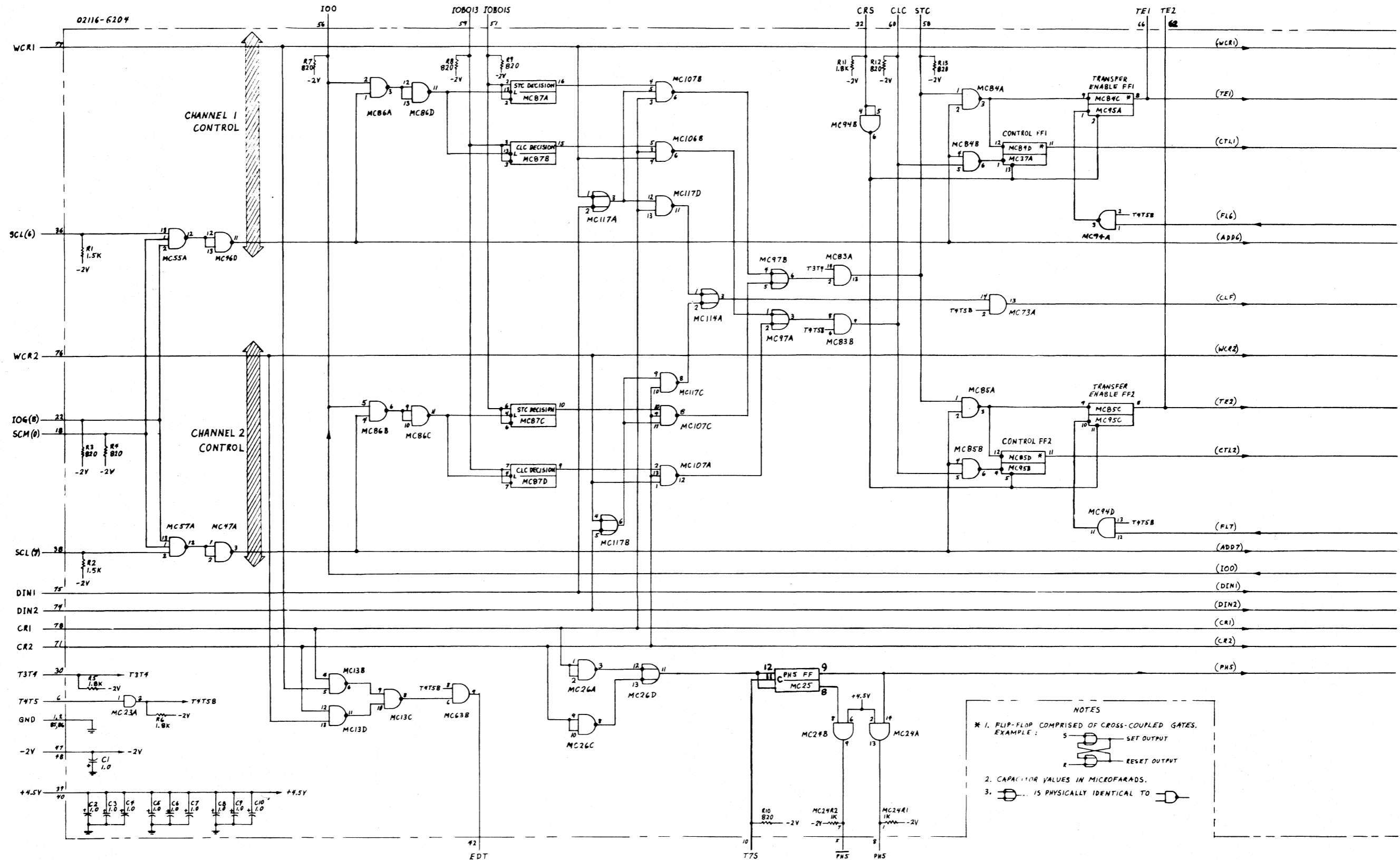


Figure 4-1. DMA Control Card Logic Diagram (Sheet 1 of 2)

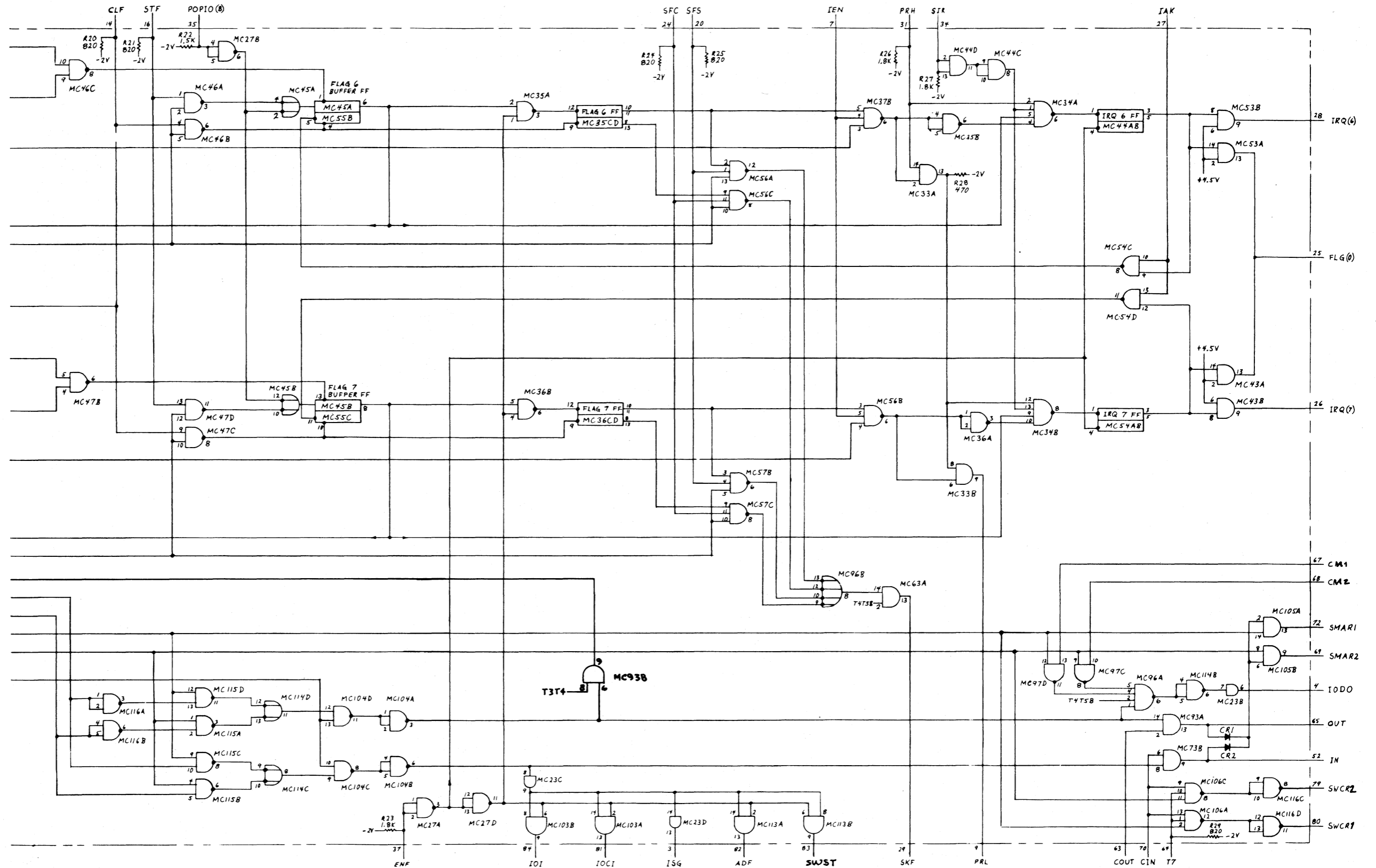


Figure 4-1A. DMA Control Card Logic Diagram
(Sheet 2 of 2)

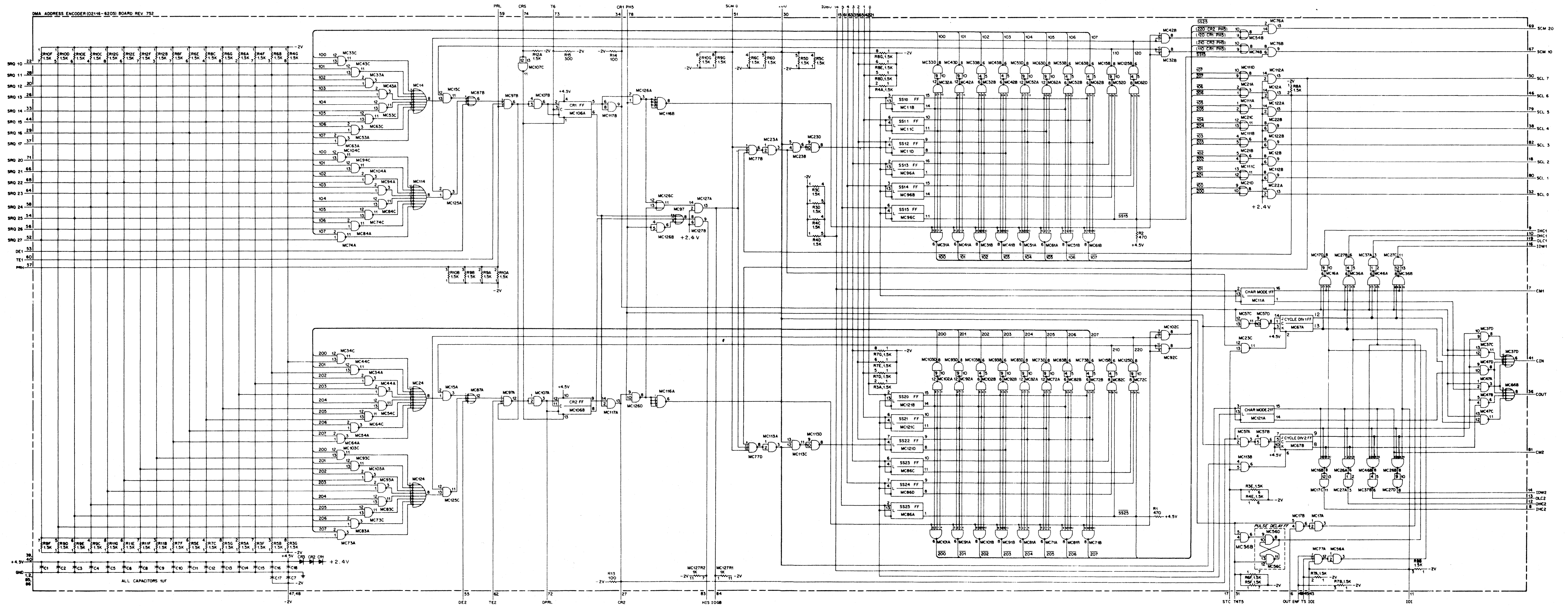


Figure 4-2. DMA Address Encoder Card Logic Diagram

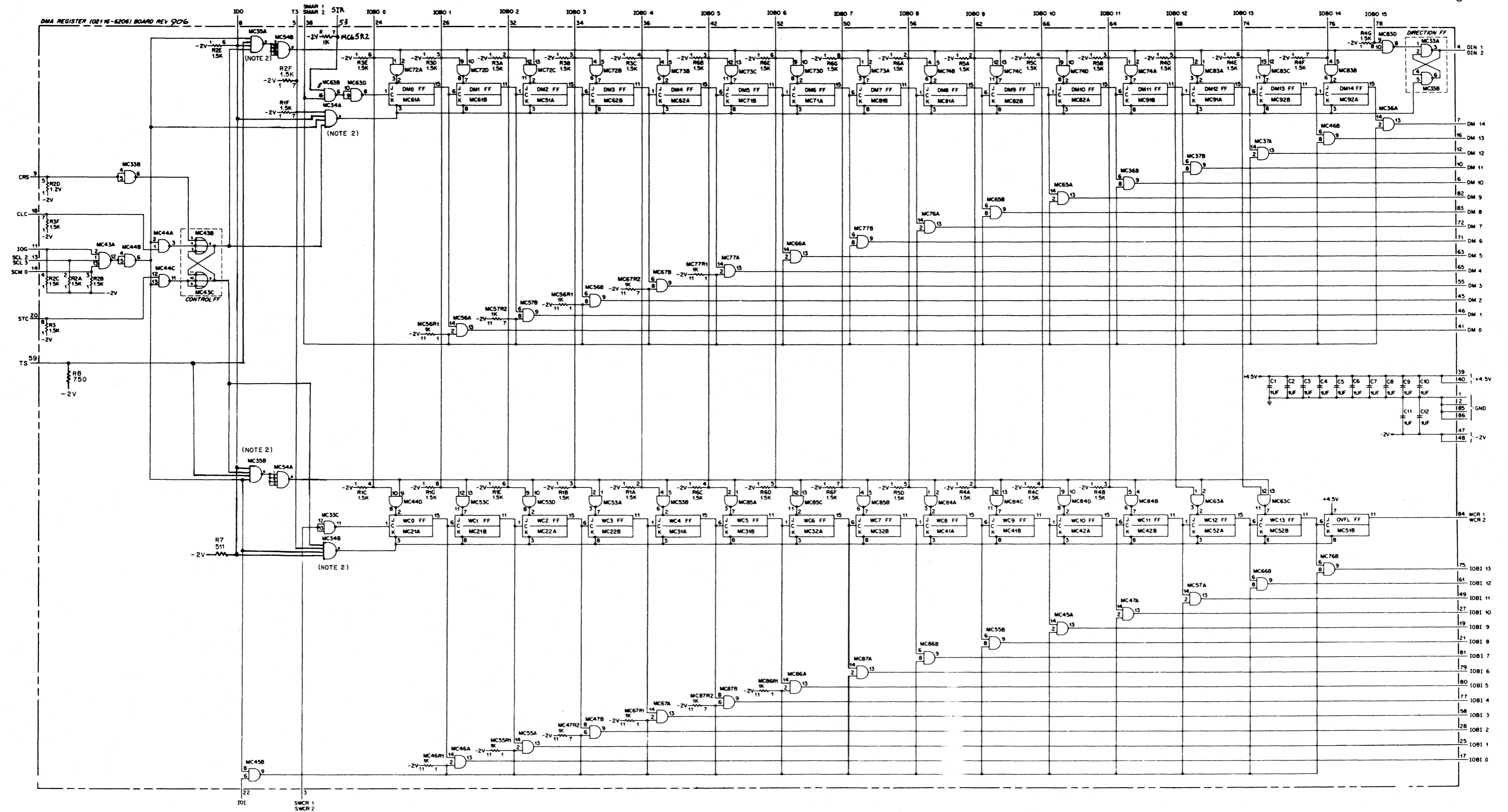


Figure 4-3. DMA Register Card Logic Diagram

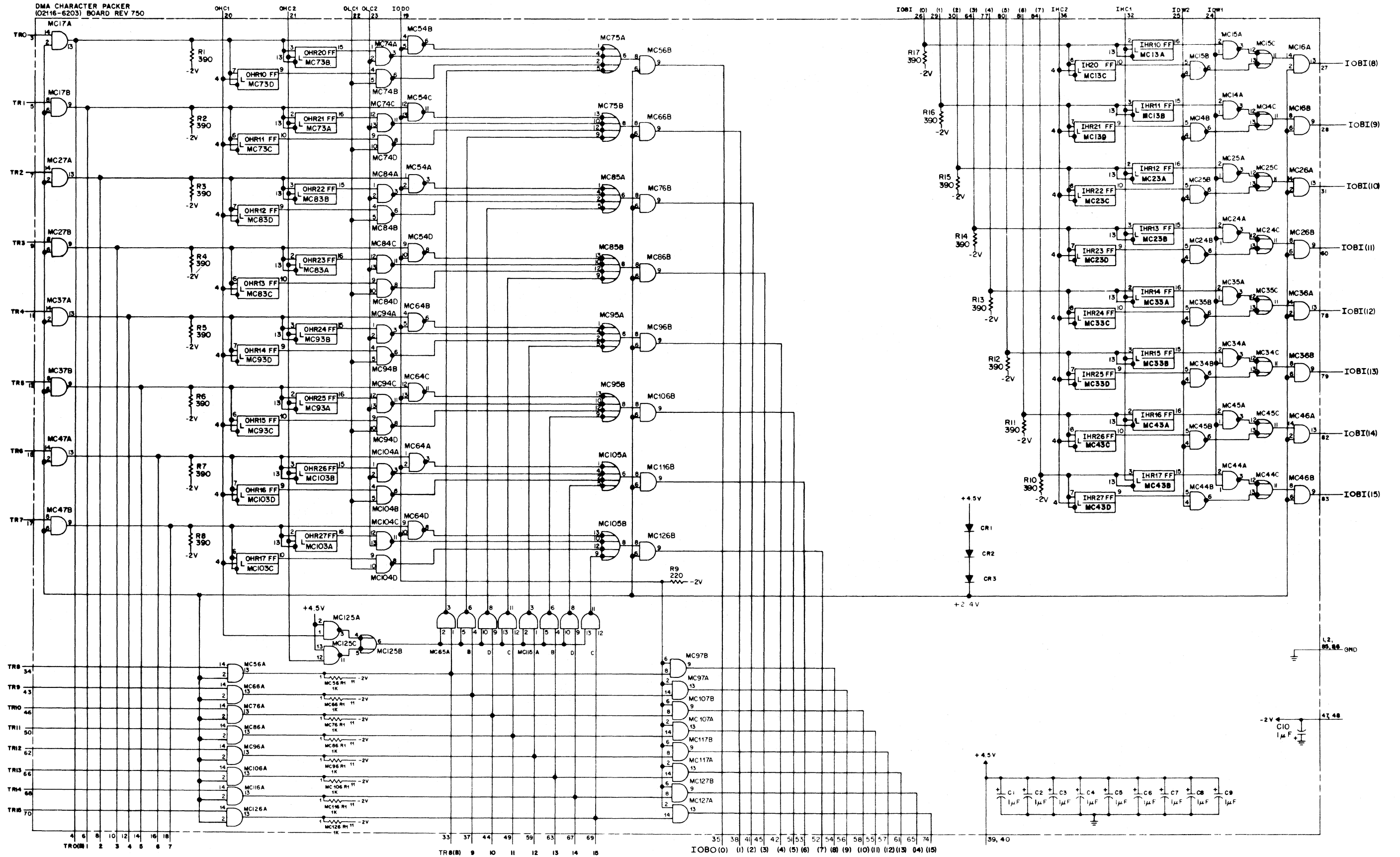
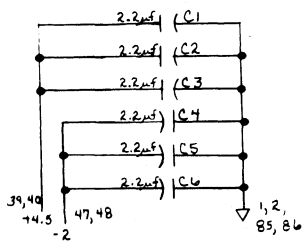
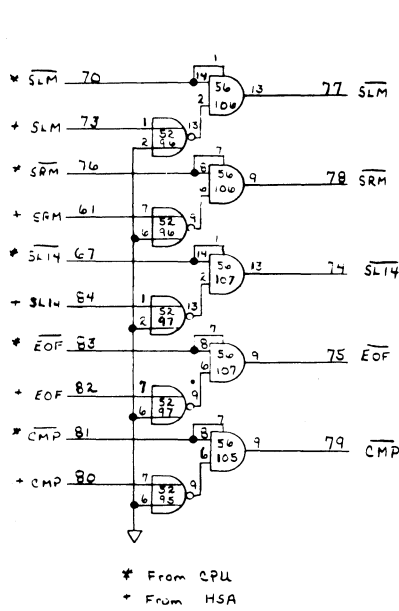


Figure 4-4. DMA Character Packer Card Logic Diagram



Unless Otherwise Specified:
 All Resistors are 220 ohms, 1/4 W, 5%

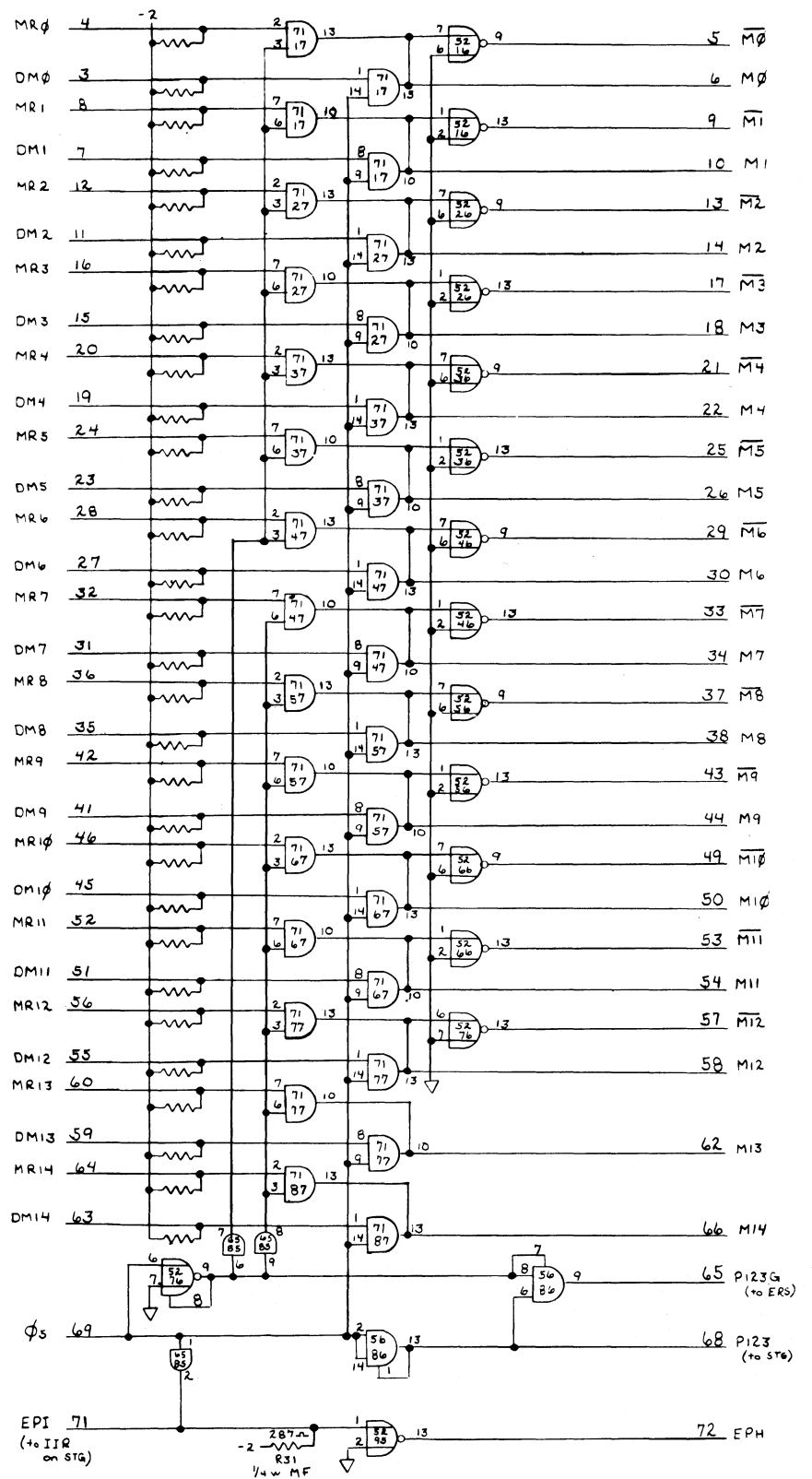
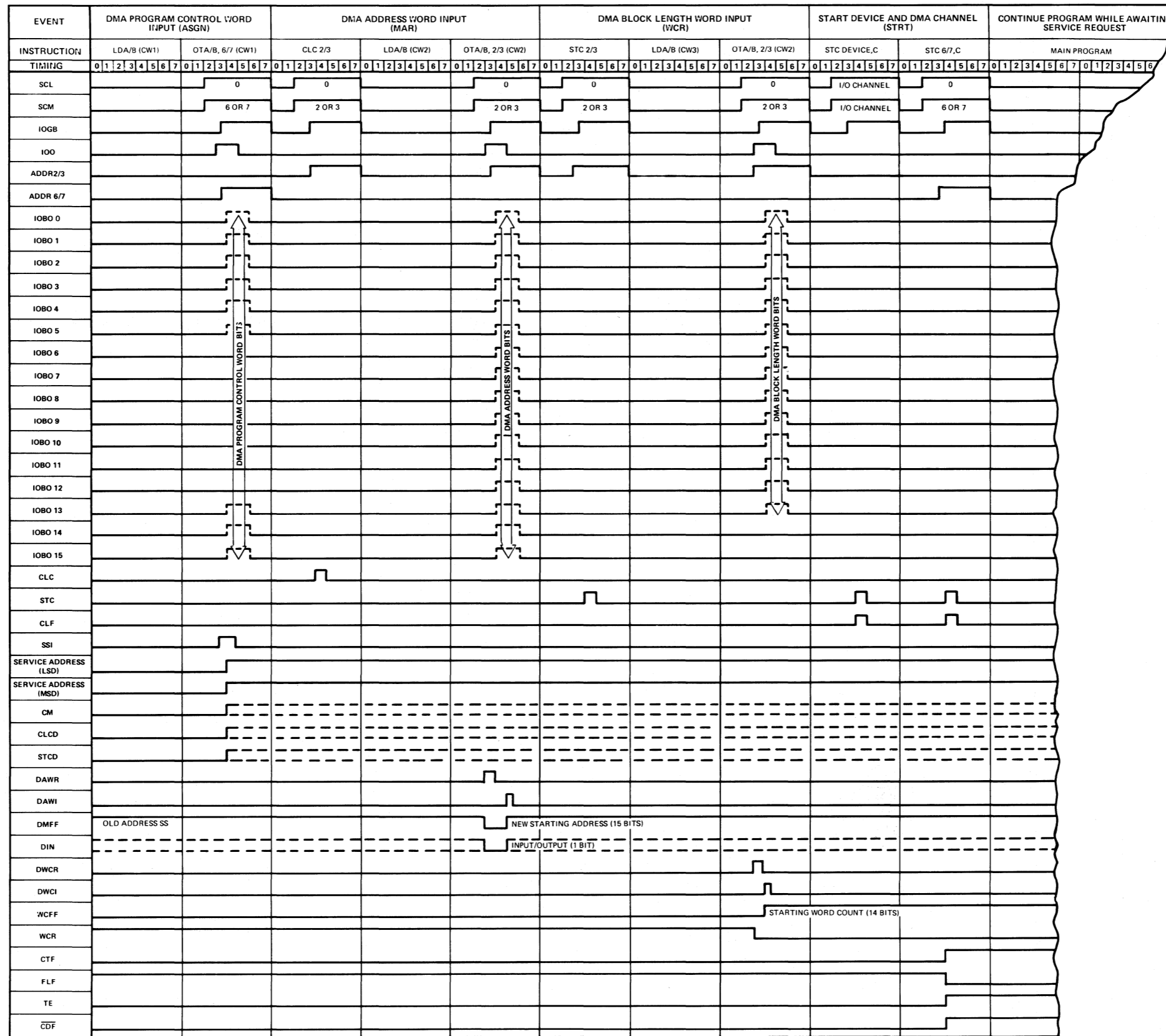
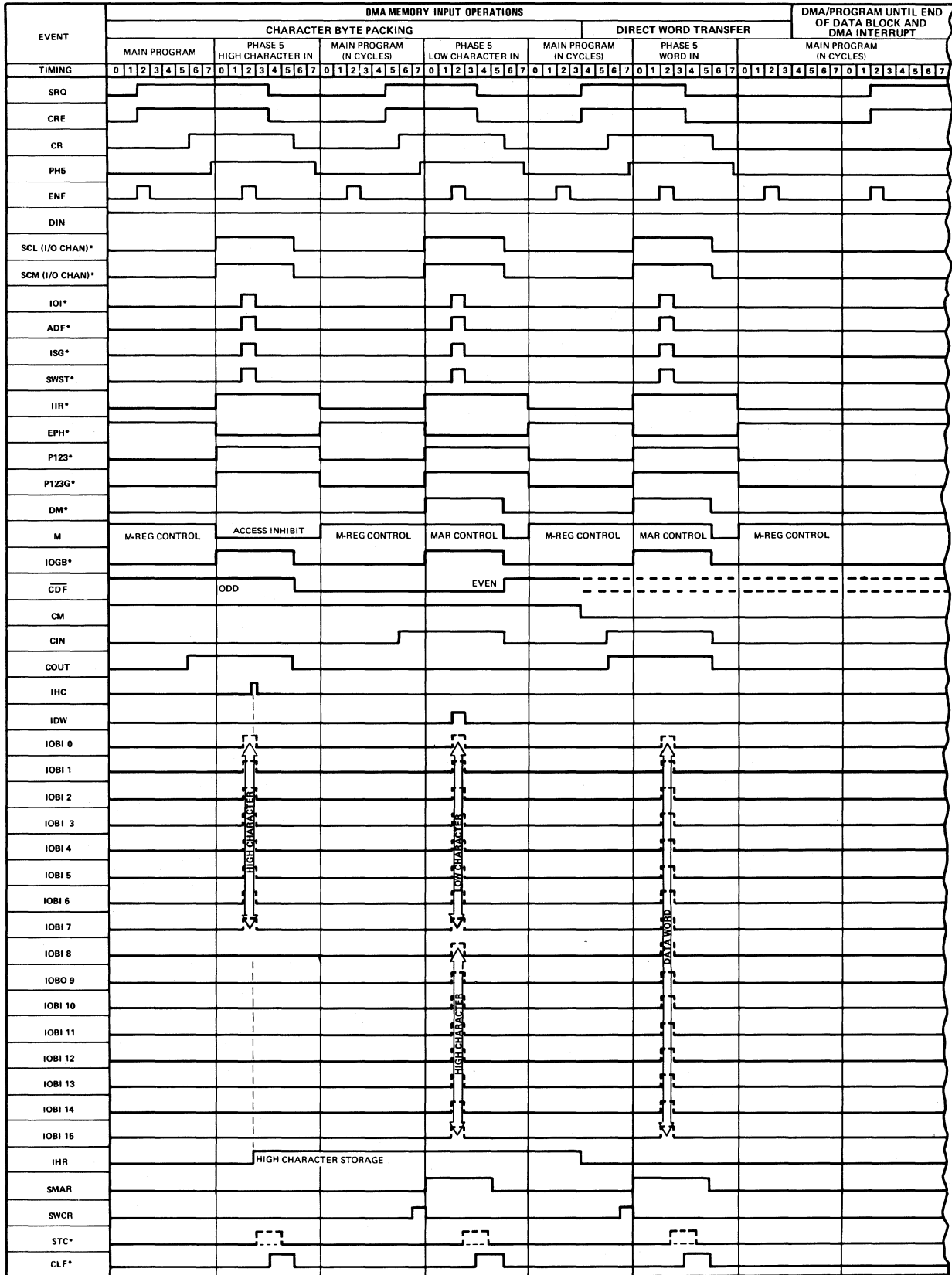


Figure 4-5. Direct Memory Logic Card Diagram



2047-1

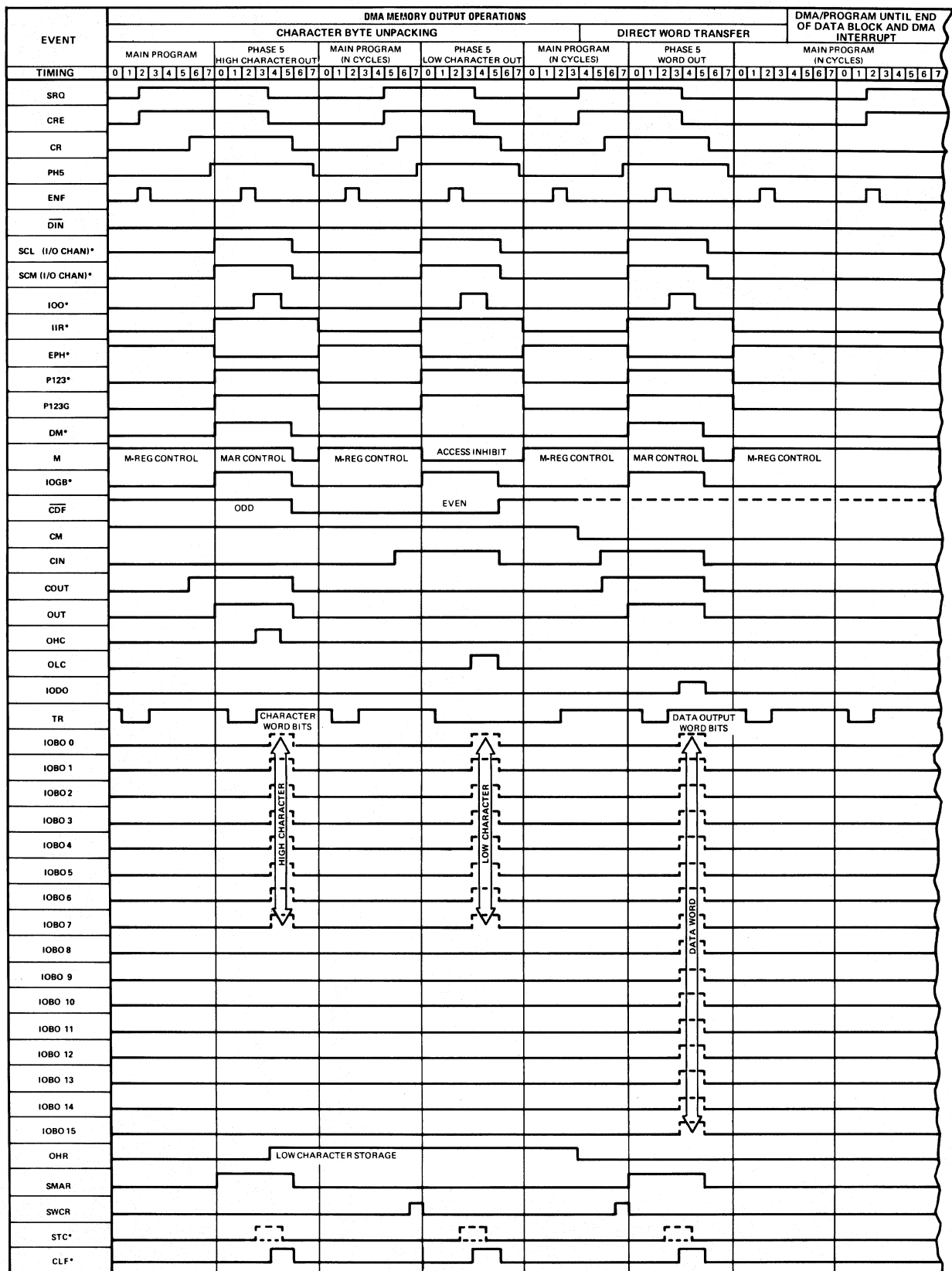
Figure 4-6. DMA Memory Input Timing Diagram



*DENOTES DMA CONTROL FUNCTION

2047-2

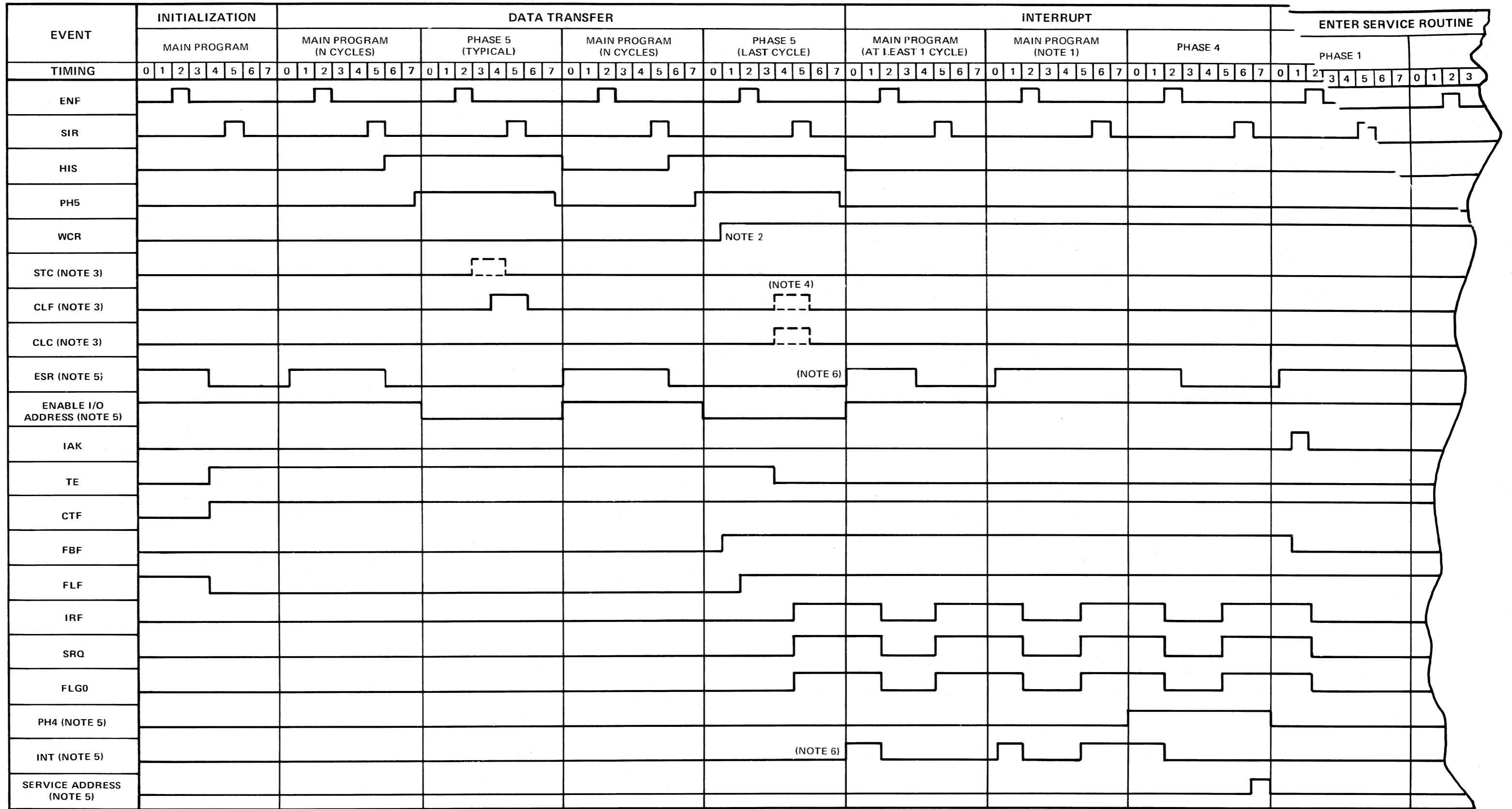
Figure 4-7. DMA Memory Input Timing Diagram



*DENOTES CONTROL FUNCTION

2047-3

Figure 4-8. DMA Memory Output Timing Diagram



- NOTES:
- Timing shown is based on the assumption that one of the conditions specified in note 6 below was present during the preceding machine cycle. Otherwise phase 4 would have set during this cycle rather than the next.
 - The exact time that the WRC signal goes high depends on inherent circuit delays. However, switching should not occur later than the end of T1.
 - DMA control functions.
 - Generated at this time during data output mode only.
 - Refer to volume III for detailed information concerning the generation and control of signal.
 - This signal assumes the low state during the occurrence of T7Ts if STF, CLF, STF, CLC, JMP,I, or JSB,I instructions are in progress, if phase 5 is set, or if a higher priority interrupt is being serviced. The low state prevents phase 4 from being set.

2047-4

Figure 4-9. DMA Interrupt Timing Diagram

TABLE 4-1. DMA OPTION INPUT-OUTPUT SIGNALS AND EQUATIONS

Signal Mnemonic and Equation	1. Definition 2. Source 3. Function
ADDR2 = (IOGB) (SCM0) (SCL2)	1. Select Code Address 2 Enabling Signals Decoded 2. DMA Register Card 1 3. *Enables DMA Channel 1 circuits addressed by select code 2
ADDR3 = (IOGB) (SCM0) (SCL3)	1. Select Code Address 3 Enabling Signals Decoded 2. DMA Register Card 2 3. *Enables DMA Channel 2 circuits addressed by select code 3
ADDR6 = (IOGB) (SCM0) (SCL6)	1. Select Code Address 6 Enabling Signals Decoded 2. DMA Encoder Card and DMA Control Card 3. *Enables DMA Channel 1 circuits addressed by select code 6
ADDR7 = (IOGB) (SCM0) (SCL7)	1. Select Code Address 7 Enabling Signals Decoded 2. DMA Encoder Card and DMA Control Card 3. *Enables DMA Channel 2 circuits addressed by select code 7
ADF = (PH5) [(CR1) (DIN1) + (CR2) (DIN2)] (ENF)	1. ADD Function 2. DMA Control Card 3. **Enables ADD function on Arithmetic Logic Card to establish direct path between IOBI lines and T-Register during phase 5 data input cycles
CDF1 = (CR1) (PH5) $\overline{\text{CDF1}}$ = (ADDR6) (STC) = (CR1) (PH5)	1. Cycle Divider FF, DMA Channel 1 2. DMA Address Encoder Card 3a. *Set: Enables low character transfer during even (second, fourth, etc.) phase 5 cycles 3b. *Clear: Enables high character data transfer during odd (first, third, etc.) phase 5 cycles
CDF2 = (CR2) (PH5) CDF2 = (ADDR6) (STC) = (CR2) (PH5)	1. Cycle Divider FF, DMA Channel 2 2. DMA Address Encoder Card 3a. *Set: Enables low character transfer during even (second, fourth, etc.) phase 5 cycles 3b. *Clear: Enables high character data transfer during odd (first, third, etc.) phase 5 cycles

*Internal card function; not routed through backplane wiring

**Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$\text{CIN} = (\text{CR1}) (\text{CDF1}) + (\text{CR1}) (\overline{\text{CM1}})$ $= (\text{CR2}) (\text{CDF2}) + (\text{CR2}) (\overline{\text{CM2}})$	<ol style="list-style-type: none"> 1. Character In 2. DMA Address Encoder Card 3a. Controls generation of register stepping signals for DMA Channel 1 3b. Controls generation of register stepping signals for DMA Channel 2
$\text{CLC} = (\text{IOG}) (\text{T4}) (\overline{\text{TR1}}) (\text{TR8}) (\text{TR7}) (\text{TR6})$ $= [(\text{CLCD1}) (\text{CR1}) (\text{WCR1}) + (\text{CLCD2}) (\text{CR2}) (\text{WCR2})] (\text{T4T5B})$	<ol style="list-style-type: none"> 1. Clear Control Signal 2a. Shift Logic Card 3a. Clears I/O Control FF addressed by program instructions 2b. DMA Control Card 3b. **Clears I/O Control FF addressed by DMA Channel 1 or 2
$\text{CLCD1} = (\text{ADDR6}) (\text{IOO}) (\text{IOBO13})$ $\overline{\text{CLCD1}} = (\text{ADDR6}) (\text{IOO}) (\overline{\text{IOBO13}})$	<ol style="list-style-type: none"> 1. Clear Control Decision FF, DMA Channel 1 2. DMA Control Card 3a. *Set: Enables generation of pseudo CLC signal by DMA Channel 1 when word count reaches zero 3b. *Clear: Inhibits generation of pseudo CLC signal by DMA Channel 1
$\text{CLCD2} = (\text{ADDR7}) (\text{IOO}) (\text{IOBO13})$ $\overline{\text{CLCD2}} = (\text{ADDR7}) (\text{IOO}) (\overline{\text{IOBO13}})$	<ol style="list-style-type: none"> 1. Clear Control Decision FF, DMA Channel 2 2. DMA Control Card 3a. *Set: Enables generation of pseudo CLC signal by DMA Channel 2 when word count reaches zero 3b. *Clear: Inhibits generation of pseudo CLC signal by DMA Channel 2
$\text{CLF} = (\text{IOG}) (\text{T4}) (\text{TR9})$ $= [(\overline{\text{WCR1}}) (\overline{\text{DIN1}}) (\text{CR1}) + (\overline{\text{WCR2}}) (\overline{\text{DIN2}}) (\text{CR2})] (\text{T4T5B})$	<ol style="list-style-type: none"> 1. Clear Flag 2a. Shift Logic Card 3a. Clears Flag FF addressed by program instructions 2b. DMA Control Card 3b. **Clears Flag FF addressed by DMA Channel 1 or 2
$\text{CM1} = (\text{SSI1}) (\text{IOBO14})$ $\overline{\text{CM1}} = (\text{SSI1}) (\overline{\text{IOBO14}})$	<ol style="list-style-type: none"> 1. Character Mode FF, DMA Channel 1 2. DMA Address Encoder Card 3a. Set: sets character data transfer mode and enables generation of character packing or unpacking control signals 3b. Clear: sets word data transfer mode and controls generation of register stepping signals for DMA Channel 1 during this mode

*Internal card function; not routed through backplane wiring

**Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$CM2 = (SSI2) (IOBO14)$ $\overline{CM2} = (SSI2) (\overline{IOBO14})$	<ol style="list-style-type: none"> 1. Character Mode FF, DMA Channel 2 2. DMA Address Encoder Card 3a. Set: sets character data transfer mode and enables generation of character packing or unpacking control signals 3b. Clear: sets word data transfer mode and controls generation of register stepping signals for DMA Channel 2 during this mode
$COUT = (CR1) (\overline{CDF1}) + (CR1) (\overline{CM1})$ $= (CR2) (\overline{CDF2}) + (CR2) (\overline{CM2})$	<ol style="list-style-type: none"> 1. Character Out 2. DMA Address Encoder Card 3a. Controls generation of register stepping signals for DMA Channel 1 3b. Controls generation of register stepping signals for DMA Channel 2
$CR1 = CRF1$	<ol style="list-style-type: none"> 1. Cycle Request, DMA Channel 1 2. DMA Address Encoder Card 3. Controls phase 5 FF
$CR2 = (CRF2) (\overline{CRF1})$	<ol style="list-style-type: none"> 1. Cycle Request, DMA Channel 2 2. DMA Address Encoder Card 3. Controls phase 5 FF
$CRE1 = [(SRQ10) (100) + (SRQ11) (101) + (SRQ12) (102) + (SRQ13) (103) + (SRQ14) (104) + (SRQ15) (105) + (SRQ16) (106) + (SRQ17) (107)] (110) + [(SRQ20) (100) + (SRQ21) (101) + (SRQ22) (102) + (SRQ23) (103) + (SRQ24) (104) + (SRQ25) (105) + (SRQ26) (106) + (SRQ27) (107)] (120)$	<ol style="list-style-type: none"> 1. Cycle Request Enable, DMA Channel 1 2. DMA Address Encoder Card 3. *Signals presence of service request input signal from I/O Channel currently under control of DMA Channel 1
$CRE2 = [(SRQ10) (200) + (SRQ11) (201) + (SRQ12) (202) + (SRQ13) (203) + (SRQ14) (204) + (SRQ15) (205) + (SRQ16) (206) + (SRQ17) (207)] (210) + [(SRQ20) (200) + (SRQ21) (201) + (SRQ22) (202) + (SRQ23) (203) + (SRQ24) (204) + (SRQ25) (205) + (SRQ26) (206) + (SRQ27) (207)] (220)$	<ol style="list-style-type: none"> 1. Cycle Request Enable, DMA Channel 2 2. DMA Address Encoder Card 3. *Signals presence of service request input signal from I/O Channel currently under control of DMA Channel 2

*Internal Card function; not routed through backplane wiring

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$CRF1 = (\overline{CRE1}) (\overline{PRH6}) (\overline{TE1}) (T6T7)$ $\overline{CRF1} = (CRS) + [(\overline{CRE1}) + (\overline{PRH6}) + (\overline{TE1})] (T6T7)$	<ol style="list-style-type: none"> 1. Cycle Request FF, DMA Channel 1 2. DMA Address Encoder Card 3a. *Set: generates CR1 signal 3b. *Clear: controls generation of HIS signal
$CRF2 = (\overline{CRE2}) [(\overline{CRE1}) + (\overline{PRH6}) + (\overline{TE1})] (TE2) (T6T7)$ $CRF2 = (CRS) + [(\overline{CRE1}) (\overline{PRH6}) (\overline{TE1}) + (\overline{CRE2}) + (\overline{TE2})] (T6T7)$	<ol style="list-style-type: none"> 1. Cycle Request FF, DMA Channel 2 2. DMA Address Encoder Card 3a. *Set: generates CR2 signal 3b. *Clear: controls generation of HIS signal
$CRS = POPIO (\text{buffered}) + CLC0$	<ol style="list-style-type: none"> 1. Control Reset to I/O Registers 2. I/O Control Card 3. Resets DMA Cycle Request and all I/O Control FF's when power is applied, or CLC instruction for select code 0 is executed
$CTF2 = (ADDR2) (STC)$ $\overline{CTF2} = (CRS) + (ADDR2) (CLC)$	<ol style="list-style-type: none"> 1. DMA Register Control FF, Channel 1 2. DMA Register Card 1 3a. *Set: enables DMA Word Count Register loading 3b. *Clear: enables DMA Direct Memory Address Register loading
$CTF3 = (ADDR3) (STC)$ $\overline{CTF3} = (CRS) + (ADDR3) (CLC)$	<ol style="list-style-type: none"> 1. DMA Register Control FF, Channel 2 2. DMA Register Card 2 3a. *Set: enables DMA Word Count Register loading 3b. *Clear: enables DMA Direct Memory Address loading
$CTF6 = (ADDR6) (STC)$ $\overline{CTF6} = (CRS) + (ADDR6) (CLC)$	<ol style="list-style-type: none"> 1. DMA I/O Control FF, Channel 1 2. DMA Control Card 3a. *Set: enables DMA Channel 1 interrupt 3b. *Clear: inhibits DMA Channel 1 interrupt

*Internal card function; not routed through backplane wiring

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$CTF7 = (ADDR7) (STC)$ $\overline{CTF7} = (CRS) + (ADDR7) (CLC)$	<ol style="list-style-type: none"> 1. DMA I/O Control FF, Channel 2 2. DMA Control Card 3a. *Set: enables DMA Channel 2 3b. *Clear: inhibits DMA Channel 2
$DAWI1 = (\overline{CTF2}) (ADDR2) (IOO) (TS)$	<ol style="list-style-type: none"> 1. DMA Address Word Input Control, Channel 1 2. DMA Register Card 7 3. *Strobes DMA address word bits on IOBO lines 0 thru 15 into Direct Memory Address Register FF's and DMA Input (Direction) FF
$DAWI2 = (\overline{CTF3}) (ADDR3) (IOO) (TS)$	<ol style="list-style-type: none"> 1. DMA Address Word Input Control, Channel 2 2. DMA Register Card 2 3. *Strobes DMA address word bits on IOBO lines 0 thru 15 into Direct Memory Address Register FF's and DMA Input (Direction) FF
$DAWR1 = (\overline{CTF2}) (ADDR2) (IOO) (T3)$	<ol style="list-style-type: none"> 1. DMA Address Word Reset Control, Channel 1 2. DMA Register Card 1 3. *Clears Direct Memory Address Register FF's and DMA Input (Direction) FF
$DAWR2 = (\overline{CTF3}) (ADDR3) (IOO) (T3)$	<ol style="list-style-type: none"> 1. DMA Address Word Reset Control, Channel 2 2. DMA Register Card 2 3. *Clears Direct Memory Address Register FF's and DMA Input (Direction) FF
$DIN1 = (DAWI1) (IOBO15)$ $\overline{DIN1} = (DAWR1)$	<ol style="list-style-type: none"> 1. DMA Input (Direction) FF, Channel 1 2. DMA Register Card 1 3a. Set: sets input data transfer mode 3b. *Clear: sets output data transfer mode

*Internal Card function; not routed through backplane wiring
 ***For future use

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	1. Definition 2. Source 3. Function
$DIN2 = (DAWI2) (IOBO15)$ $\overline{DIN2} = (DAWR2)$	1. DMA Input (Direction) FF, Channel 2 2. DMA Register Card 2 3a. Set: sets input data transfer mode 3b. *Clear: sets output data transfer mode
$DM0-14 = (DMFF0-14) (SMAR1)$ $= (DMFF0-14) (SMAR2)$	1. Direct Memory Address, Bits 0 thru 14 2. DMA Register Card 1 or 2 3a. **Direct Memory Address from DMA Channel 1 3b. **Direct Memory Address from DMA Channel 2
$DMFF0-14 = (DAWI1) (IOBO0-14)$ $\overline{DMFF0-14} = (DAWR1)$	1. Direct Memory Address Register FF's 0 thru 14, DMA Channel 1 2. DMA Register Card 1 3a. *Set: stores direct memory starting address for DMA Channel 1 3b. *Clear: all DMA Channel 1 Direct Memory Address Register FF's reset
$DMFF0-14 = (DAWI2) (IOBO0-14)$ $\overline{DMFF0-14} = (DAWR2)$	1. Direct Memory Address Register FF's 0 thru 14, DMA Channel 2 2. DMA Register Card 2 3a. *Set: stores direct memory starting address for DMA Channel 2 3b. *Clear: all DMA Channel 2 Direct Memory Address Register FF's reset
$DWC11 = (CTF2) (ADDR6) (IOO) (TS)$	1. DMA Word Count Input Control, Channel 1 2. DMA Register Card 1 3. *Strobes DMA Word Count bits on IOBO lines 0 thru 13 into Word Count Register FF's
$DWC12 = (CTF3) (ADDR7) (IOO) (TS)$	1. DMA Word Count Input Control, Channel 2 2. DMA Register Card 2 3. *Strobes DMA Word Count bits on IOBO lines 0 thru 13 into Word Count Register FF's
$DWCR1 = (CTF2) (ADDR2) (IOO) (T3)$	1. DMA Word Count Reset Control, Channel 1 2. DMA Register Card 1 3. *Clears Word Count Register FF's and Overflow (Rollover) FF

*Internal card function; not routed through backplane wiring

**Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$DWCR2 = (CTF3) (ADDR3) (IOO) (T3)$	<ol style="list-style-type: none"> 1. DMA Word Count Reset Control, Channel 2 2. DMA Register Card 2 3. *Clears Word Count Register FF's and Overflow (Rollover) FF
EDT (not used)	
$ENF = T2$ (buffered)	<ol style="list-style-type: none"> 1. Enable Flag 2. I/O Control Card 3. Clocking control signal to DMA Control Card and DMA Address Encoder Card
$EPH = \overline{PH5}$	<ol style="list-style-type: none"> 1. Enable Phase 2. Direct Memory Logic Card 3. **Enables Phase FF output gates on Timing Generator Card when present; inhibits Phase FF output gates when absent
$FBF6 = [(POPIO) + (TE1) (WCR1) + (ADDR6) (STF)] (FBF6)$ $\overline{FBF6} = [(IAK) (IRQ6) + (ADDR7) (CLF)] (FBF6)$	<ol style="list-style-type: none"> 1. Flag Buffer FF I/O Channel 6 2. DMA Control Card 3a. *Set: enables Flag 6 FF, DMA Channel 1 3b. *Clear: inhibits Flag 6 FF, DMA Channel 1
$FBF7 = [(POPIO) + (TE2) (WCR2) + (ADDR7) (STF)] (FBF7)$ $\overline{FBF7} = [(IAK) (IRQ7) + (ADDR7) (CLF)] (FBF7)$	<ol style="list-style-type: none"> 1. Flag Buffer FF I/O Channel 7 2. DMA Control Card 3a. *Set: enables Flag 7 FF, DMA Channel 2 3b. *Clear: inhibits Flag 7 FF, DMA Channel 2
$FLF6 = (FBF6) (ENF)$ $\overline{FLF6} = (ADDR6) (CLF)$	<ol style="list-style-type: none"> 1. Flag FF I/O Channel 6 2. DMA Control Card 3a. *Set: enables interrupt request from DMA Channel 1 3b. *Clear: inhibits interrupt request from DMA Channel 1
$FLF7 = (FBF7) (ENF)$ $\overline{FLF7} = (ADDR7) (CLF)$	<ol style="list-style-type: none"> 1. Flag FF I/O Channel 7 2. DMA Control Card 3a. *Set: enables interrupt request from DMA Channel 2 3b. *Clear: inhibits interrupt request from DMA Channel 2

*Internal card function; not routed through backplane wiring

**Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$FLG0 = (IRQ6) + (IRQ7)$	<ol style="list-style-type: none"> 1. Flag from I/O Channel 6, DMA Channel 1 or 2 2. DMA Control Card 3. Clocks generation of Interrupt signal on I/O Address Card
$HIS = (CR1) + (CR2) + (PH5)$	<ol style="list-style-type: none"> 1. Hold Interrupt System 2. DMA Address Encoder Card 3. Inhibits interrupt system during phase 5 data transfer cycles
$IAK = (\overline{ICF}) (PH1) (T1)$	<ol style="list-style-type: none"> 1. Interrupt Acknowledge 2. I/O Control Card 3. Clocks reset of Flag Buffer 6 and Flag Buffer 7 FF's on DMA Control Card
$IDW1 = (CM1) (CR1) (CDF1) (IOI)$	<ol style="list-style-type: none"> 1. Input Data Word from DMA Channel 1 2. DMA Address Encoder 3. Strokes high character data bits stored by Input Holding Register onto IOBI lines 0-7
$IDW2 = (CM2) (CR2) (CDF2) (IOI)$	<ol style="list-style-type: none"> 1. Input Data Word from DMA Channel 2 2. DMA Address Encoder 3. Strokes high character data bits stored by Input Holding Register onto IOBI lines 0-7
$IEN6 = (ISEF)$	<ol style="list-style-type: none"> 1. Interrupt Enable, I/O Select Codes 6 and 7 2. I/O Control Card 3. Interrupt Enable for DMA Channels 1 and 2
$IHC1 = (CM1) (CR1) (\overline{CDF1}) (IOI) (ENF) (TS)$	<ol style="list-style-type: none"> 1. Input High Character to DMA Channel 1 2. DMA Address Encoder 3. Strokes data bits on IOBI lines 0-7 into Input Holding Register for DMA Channel 1
$IHC2 = (CM2) (CR2) (\overline{CDF2}) (IOI) (ENF) (TS)$	<ol style="list-style-type: none"> 1. Input High Character to DMA Channel 2 2. DMA Address Encoder 3. Strokes data bits on IOBI lines 0-7 into Input Holding Register for DMA Channel 2
$IHR10-17 = (IOBO0-7) (IHC1)$ $\overline{IHR10-17} = (\overline{IOBO0-7}) (IHC1)$	<ol style="list-style-type: none"> 1. Input Holding Register FF's, DMA Channel 1 2. DMA Character Packer Card 3a. *Set: register FF stores binary 1 high character data bit 3b. *Clear: register FF stores binary 0 high character data bit

*Internal card function; not routed through backplane wiring

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
<p>IHR20-27 = (IOBO0-7) (IHC2)</p> <p>$\overline{\text{IHR20-27}} = (\overline{\text{IOBO0-7}}) (\text{IHC2})$</p>	<ol style="list-style-type: none"> 1. Input Holding Register FF's, DMA Channel 2 2. DMA Character Packer Card 3a. *Set: register FF stores binary 1 high character data bit 3b. *Clear: register FF stores binary 0 high character data bit
<p>IIR = (PH5)</p>	<ol style="list-style-type: none"> 1. Inhibit Instruction Register Output Signal 2. Direct Memory Logic Card 3. **Inhibits generation of Instruction Register enabling signal on Timing Generator Board
<p>IN (not used)</p>	
<p>IOBI0-7 = (Refer to applicable I/O Interface Card Manual)</p> <p>= (WC0-7) (ADDR2) (IOI)</p> <p>= (WC0-7) (ADDR3) (IOI)</p>	<ol style="list-style-type: none"> 1. I/O Bus Input, Lines 0-7 2a. I/O Interface Cards 3a. Word or low character input bits 0-7 from I/O Channels to S-Bus lines 0-7, or high character input bits 0-7 to DMA Character Packer Card 2b. DMA Register Card 1 3b. Word count bits 0-7, DMA Channel 1, to S-Bus lines 0-7 2c. DMA Register Card 2 3c. Word count, bits 0-7, DMA Channel 2, to S-Bus lines 0-7
<p>IOBI8-13 = (Refer to applicable I/O Interface Card Manual)</p> <p>= (IHR10-15) (IDW1)</p> <p>= (IHR20-25) (IDW2)</p> <p>= (WC8-13) (ADDR2) (IOI)</p> <p>= (WC8-13) (ADDR3) (IOI)</p>	<ol style="list-style-type: none"> 1. I/O Bus Input, Lines 8-13 2a. I/O Interface Card 3a. Word input bits 8-13 from I/O Channel to S-Bus lines 8-13 2b. DMA Character Packer Card 3b. High Character input bits 0-5 from DMA Channel 1 to S-Bus lines 8-13 2c. DMA Character Packer Card 3c. High character input bits 0-5 from DMA Channel 2 to S-Bus lines 8-13 2d. DMA Register Card 1 3d. Word count bits 8-13, DMA Channel 1, to S-Bus lines 8-13 2e. DMA Register Card 2 3e. Word count bits 8-13, DMA Channel 2, to S-Bus lines 8-13

*Internal card function; not routed through backplane wiring
 **Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	1. Definition 2. Source 3. Function
<p>IOBI14, 15 = (Refer to applicable I/O Interface Card Manual)</p> <p>= (IHR16, 17) (IDW1)</p> <p>= (IHR26, 27) (IDW2)</p>	<p>1. I/O Bus Input, Lines 14 and 15</p> <p>2a. I/O Interface Cards</p> <p>3a. Word input bits 14 and 15 from I/O Channel to S-Bus lines 14 and 15</p> <p>2b. DMA Character Packer Card</p> <p>3b. High character input bits 6 and 7 from DMA Channel 1 to S-Bus lines 14 and 15</p> <p>2c. DMA Character Packer Card</p> <p>3c. High character input bit 6 and 7 from DMA Channel 2 to S-Bus lines 14 and 15</p>
<p>IOBO0-7 = (RB0-7) (IOCO)</p> <p>= (TR0-7) (IODO)</p> <p>= (TR8-15) (OHC1 + OHC2)</p> <p>= (OHR10-17) (OLC1)</p> <p>= (OHR20-27) (OLC2)</p>	<p>1. I/O Bus Output Lines 0-7</p> <p>2a. Arithmetic Logic Board</p> <p>3a. Output from R-Bus lines 0-7</p> <p>2b. DMA Character Packer Card</p> <p>3b. Word output bits 0-7 from DMA Channel 1 or 2</p> <p>2c. DMA Character Packer Card</p> <p>3c. High character output bits 0-7 from DMA Channel 1 or 2</p> <p>2d. DMA Character Packer Card</p> <p>3d. Low character output bits 0-7 from DMA Channel 1</p> <p>2e. DMA Character Packer Card</p> <p>3e. Low character output bits 0-7 from DMA Channel 2</p>
<p>IOBO8-15 = (RB8-15) (IOCO)</p> <p>= (TR8-15) (IODO)</p>	<p>1. I/O Bus Output Lines 8-15</p> <p>2a. Arithmetic Logic Card</p> <p>3a. Output from R-Bus lines 8-15</p> <p>2b. DMA Character Packer Card</p> <p>3b. Word output bits 8-15 from DMA Channel 1 or 2</p>
<p>IOCI = (PH5) [(CR1) (DIN1) + (CR2) (DIN2)] (ENF)</p>	<p>1. I/O Control Input</p> <p>2. DMA Control Card</p> <p>3. **Same as **IOI</p>
<p>$IODO = \left[\begin{array}{l} \overline{CM1} (CR1) + \overline{CM2} (CR2) \\ (CR1) \overline{DIN1} + (CR2) \overline{DIN2} \end{array} \right] (PH5) (T4T5B)$</p>	<p>1. I/O Data Out (DMA Channel 1 or 2)</p> <p>2. DMA Control Card</p> <p>3. Strokes word data bits TR0 thru TR15 onto IOBO lines 0-15</p>

**Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$\text{IOGB} = (\text{IOG delayed 100 ns})$ $= (\text{CR1}) (\text{PH5}) + (\text{CR2}) (\text{PH5})$	<ol style="list-style-type: none"> 1. IOG Buffered 2a. I/O Control Card 3a. Select code address enabling signal for I/O channels addressed under program control 2b. DMA Address Encoder Card 3b. **Select code address enabling signal for I/O channels addressed by DMA Channel 1 or 2
$\text{IOI} = (\text{IOG}) (\text{T4T5}) (\text{TR8}) (\overline{\text{TR7}}) + (\text{SEO}) (\text{T2})$ $= (\text{PH5}) [(\text{CR1}) (\text{DIN1}) + (\text{CR2}) (\text{DIN2})] (\text{ENF})$	<ol style="list-style-type: none"> 1. I/O Input 2a. Shift Logic Card 3a. Enables IOBI lines under program or switch register control 2b. DMA Control Card 3b. **Enables IOBI lines under control of DMA Channel 1 or 2
$\text{IOO} = (\text{T3T4}) (\text{TR8}) (\text{TR7}) (\overline{\text{TR6}}) [(\text{IOG}) + (\text{SRG}) (\text{T3})]$ $= (\text{T3T4}) (\text{PH5}) [(\text{CR1}) (\overline{\text{DIN1}}) + (\text{CR2}) (\overline{\text{DIN2}})]$	<ol style="list-style-type: none"> 1. I/O Output 2a. Shift Logic Card 3a. Enables IOBO lines under program control 2b. DMA Control Card 3b. **Enables IOBO lines under control of DMA Channel 1 or 2
$\text{IRF6} = (\text{SIR}) (\text{PRH6}) (\text{FBF6}) (\text{IEN6}) (\text{FLF6}) (\text{CTF6})$ $\overline{\text{IRF6}} = (\text{ENF})$	<ol style="list-style-type: none"> 1. Interrupt Request FF, I/O Channel 7 2. DMA Control Card 3a. *Set: generates IRQ6 and FLG0 signals for DMA Channel 1 3b. *Clear: inhibits IRQ6 and FLG0 signals for DMA Channel 1
$\text{IRF7} = (\text{SIR}) (\text{PRH7}) (\text{FBF7}) (\text{IEN6}) (\text{FLF7}) (\text{CTF7})$ $\overline{\text{IRF7}} = (\text{ENF})$	<ol style="list-style-type: none"> 1. Interrupt Request FF, I/O Channel 7 2. DMA Control Card 3a. *Set: generates IRQ6 and FLG0 signals from DMA Channel 2 3b. *Clear: inhibits IRQ6 and FLG0 signals from DMA Channel 2
$\text{IRQ6} = \text{IRF6}$	<ol style="list-style-type: none"> 1. Interrupt Request, I/O Channel 6 2. DMA Control Card 3. Interrupt Request to I/O Address Card from DMA Channel 1

*Internal card function; not routed through backplane wiring
 **Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$IRQ7 = IRF7$	<ol style="list-style-type: none"> 1. Interrupt Request, I/O Channel 7 2. DMA Control Card 3. Interrupt Request to I/O Address Card from DMA Channel 2
$ISG = (PH5) [(CR1) (DIN1) + (CR2) (DIN2)] (ENF)$	<ol style="list-style-type: none"> 1. Inhibit Strobe Generator 2. DMA Control Card 3. **Inhibits generation of Memory Strobe Timing (MST) signal on Timing Generator Card
$M0-14 = (MR0-14) (\overline{PH5})$ $= (DM0-14) (PH5)$ $\overline{M0-12} = (\overline{MR0-14}) (\overline{PH5})$ $= (DM0-14) (PH5)$	<ol style="list-style-type: none"> 1. Memory Address Bits 2. Direct Memory Logic Card 3a. Provides binary 1 memory address bits during program control 3b. **Provides binary 1 memory address bits during DMA Channel 1 or 2 control 3c. Provides binary 0 memory address bits during program control 3d. **Provides binary 0 memory address bits during DMA Channel 1 or 2 control
$OHC1 = (CM1) (CR1) (\overline{CD1}) (IOO) (OUT)$	<ol style="list-style-type: none"> 1. Output High Character from DMA Channel 1 2. DMA Address Encoder 3. Strobes high character data bits from TR8 thru TR15 onto IOBO lines 0-7, respectively; strobes low character data bits from TR0 thru TR7 into Channel 1 Output Holding Register
$OHC2 = (CM2) (CR2) (\overline{CD2}) (IOO) (OUT)$	<ol style="list-style-type: none"> 1. Output High Character from DMA Channel 2 2. DMA Address Encoder 3. Strobes high character data bits TR8 thru TR15 onto IOBO lines 0-7, respectively; strobes low character data bits from TR0 thru TR7 into Channel 2 Output Holding Register
$OHR10-17 = (TR0-7) (OHC1)$ $\overline{OHR10-17} = (\overline{TR0-7}) (OHC1)$	<ol style="list-style-type: none"> 1. Output Holding Register FF's, DMA Channel 1 2. DMA Character Packer Card 3a. *Set: register FF stores binary 1 low character data bit 3b. *Clear: register FF stores binary 0 low character data bit

*Internal card function; not routed through backplane wiring

**Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$\text{OHR20-27} = (\text{TR0-7}) (\text{OHC2})$ $\overline{\text{OHR20-27}} = \overline{(\text{TR0-7})} (\text{OHC2})$	<ol style="list-style-type: none"> 1. Output Holding Register FF's DMA Channel 2 2. DMA Character Packer Card 3a. *Set: register FF stores binary 1 low character data bit 3b. *Clear: register FF stores binary 0 low character data bit
$\text{OLC1} = (\text{CM1}) (\text{CR1}) (\text{CDF1}) (\text{T4T5 Delayed})$	<ol style="list-style-type: none"> 1. Output Low Character from DMA Channel 1 2. DMA Address Encoder Card 3. Strokes low character data bits stored by Output Holding Register onto IOBO lines 0-7
$\text{OLC2} = (\text{CM2}) (\text{CR2}) (\text{CDF2}) (\text{T4T5 Delayed})$	<ol style="list-style-type: none"> 1. Output Low Character from DMA Channel 2 2. DMA Channel 2 3. Strokes low character data bits stored by Output Holding Register onto IOBO lines 0-7
$\text{OUT} = (\text{COUT}) (\text{PH5}) [(\text{CR1}) (\overline{\text{DIN1}}) + (\text{CR2}) (\overline{\text{DIN2}})]$	<ol style="list-style-type: none"> 1. Output High Character 2. DMA Control Card 3. Enables generation of OHC1 signal
$\text{P123} = \text{PH5}$	<ol style="list-style-type: none"> 1. Phase 1, Phase 2, or Phase 3 2. Direct Memory Logic Card 3. **Simulates P123 output on Timing Generator Card
$\text{P123G} = \overline{\text{PH5}}$	<ol style="list-style-type: none"> 1. Phase 1, Phase 2, or Phase 3 2. Direct Memory Logic Card 3. **Inhibits A/B Address Flip-Flop on Shift Logic Card during phase 5 cycles
$\text{PH5} = [(\text{CR1}) + (\text{CR2})] \text{T7TS}$ $\overline{\text{PH5}} = \overline{\text{T7TS}}$	<ol style="list-style-type: none"> 1. Phase 5 FF, DMA Channel 1 or 2 2. DMA Control Card 3a. Primary DMA control signal; turns-on phase 5 direct memory access cycle and controls pseudo operation of Central Processor and I/O Control System 3b. *Turns-off phase 5
$\text{POPIO} = \text{T5} [(\text{POFP}) + (\text{PRS}) (\overline{\text{SWC}})]$	<ol style="list-style-type: none"> 1. Power On Pulse to I/O 2. Timing Generator Card 3. Resets Flag Buffer 6 FF and Flag Buffer 7 FF when power is applied

*Internal card function; not routed through backplane wiring
 **Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$\text{PRH6} = \text{PRL5}$	<ol style="list-style-type: none"> 1. Priority High I/O Channel 6; Priority Low I/O Channel 5 2. Power Fail Control Card 3. Enables DMA Channel 1 cycle request and interrupt request
$\text{PRH7} = \text{PRL6} = (\text{PRH6}) [(\overline{\text{FLF6}}) + (\overline{\text{CTF6}}) + (\overline{\text{IEN6}})]$	<ol style="list-style-type: none"> 1. Priority High I/O Channel 7; Priority Low I/O Channel 6 2. DMA Control Card 3. *Enables DMA Channel 2 cycle request and interrupt request
$\text{PRH10} = \text{PRL7} = (\text{PRH7}) [(\overline{\text{FLF7}}) + (\overline{\text{CTF6}}) + (\overline{\text{IEN6}})]$	<ol style="list-style-type: none"> 1. Priority Low I/O Channel 7; Priority High I/O Channel 10 2. DMA Control Card 3. Enables I/O Channel 10
$\begin{aligned} \text{SCL0} &= (\overline{100}) + (\overline{200}) \\ &= (\overline{\text{TR2}}) (\overline{\text{TR1}}) (\overline{\text{TR0}}) (\text{PH5}) \end{aligned}$	<ol style="list-style-type: none"> 1. Select Code Least Significant Octal Zero 2a. DMA Address Encoder Card 3a. **Address enabling signal to all I/O channels, addressed under control of DMA Channel 1 or 2, having least significant select code digit zero 2b. I/O Address Card 3b. Address enabling signal to all I/O channels, addressed under program control, having least significant select code digit zero
$\begin{aligned} \text{SCL1} &= (\overline{101}) + (\overline{201}) \\ &= (\overline{\text{TR2}}) (\overline{\text{TR1}}) (\text{TR0}) (\overline{\text{PH5}}) \end{aligned}$	<ol style="list-style-type: none"> 1. Select Code Least Significant Octal One 2a. DMA Address Encoder Card 3a. **Address enabling signal to all I/O channels, addressed under control of DMA Channel 1 or 2, having least significant select code digit one 2b. I/O Address Card 3b. Address enabling signal to all I/O channels, addressed under program control, having least significant select code digit one
$\text{SCL2} = (\overline{102}) + (\overline{202})$	<ol style="list-style-type: none"> 1. Select Code Least Significant Octal Two 2a. DMA Address Encoder Card 3a. **Address enabling signal to all I/O channels, addressed under control of DMA Channel 1 or 2, having least significant select code digit two

*Internal card function; not routed through backplane wiring

**Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$\text{SCL2} = (\overline{\text{TR2}}) (\overline{\text{TR1}}) (\text{TR0}) (\overline{\text{PH5}})$	<ol style="list-style-type: none"> 1. Select Code Least Significant Octal Two (cont) 2b. I/O Address Card 3b. Address enabling signal to all I/O channels, addressed under program control, having least significant select code digit two
$\begin{aligned} \text{SCL3} &= (\overline{\text{I03}}) + (\overline{\text{203}}) \\ &= (\overline{\text{TR2}}) (\text{TR1}) (\text{TR0}) (\overline{\text{PH5}}) \end{aligned}$	<ol style="list-style-type: none"> 1. Select Code Least Significant Octal Three 2a. DMA Address Encoder Card 3a. **Address enabling signal to all I/O channels, addressed under control of DMA Channel 1 or 2, having least significant select code digit three 2b. I/O Address Card 3b. Address enabling signal to all I/O channels, addressed under program control, having least significant select code digit three
$\begin{aligned} \text{SCL4} &= (\overline{\text{I04}}) + (\overline{\text{204}}) \\ &= (\text{TR2}) (\overline{\text{TR1}}) (\overline{\text{TR0}}) (\text{PH5}) \end{aligned}$	<ol style="list-style-type: none"> 1. Select Code Least Significant Octal Four 2a. DMA Address Encoder Card 3a. **Address enabling signal to all I/O channels, addressed under control of DMA Channel 1 or 2, having least significant select code digit four 2b. I/O Address Card 3b. Address enabling signal to all I/O channels, addressed under program control, having least significant select code digit four
$\begin{aligned} \text{SCL5} &= (\overline{\text{I05}}) + (\overline{\text{205}}) \\ &= (\text{TR2}) (\overline{\text{TR1}}) (\text{TR0}) (\overline{\text{PH5}}) \end{aligned}$	<ol style="list-style-type: none"> 1. Select Code Least Significant Octal Five 2a. DMA Address Encoder Card 3a. **Address enabling signal to all I/O channels, addressed under control of DMA Channel 1 or 2, having least significant select code digit five 2b. I/O Address Card 3b. Address enabling signal to all I/O channels, addressed under program control, having least significant select code digit five
$\begin{aligned} \text{SCL6} &= (\overline{\text{I06}}) + (\overline{\text{206}}) \\ &= (\text{TR2}) (\text{TR1}) (\overline{\text{TR0}}) (\text{PH5}) \end{aligned}$	<ol style="list-style-type: none"> 1. Select Code Least Significant Octal Six 2a. DMA Address Encoder Card 3a. **Address enabling signal to all I/O channels, addressed under control of DMA Channel 1 or 2, having least significant select code digit six 2b. I/O Address Card 3b. Address enabling signal to all I/O channels, addressed under program control, having least significant select code digit six

**Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$SCL7 = (\overline{107}) + (\overline{207})$ $= (TR2) (TR1) (TR0) (\overline{PH5})$	<ol style="list-style-type: none"> 1. Select Code Least Significant Octal Seven 2a. DMA Address Encoder Card 3a. **Address enabling signal to all I/O channels, addressed under control of DMA Channel 1 or 2, having least significant select code digit seven 2b. I/O Address Card 3b. Address enabling signal to all I/O channels, addressed under program control, having least significant select code digit seven
$SCM0 = (\overline{TR5}) (\overline{TR4}) (\overline{TR3}) (\overline{PH5})$	<ol style="list-style-type: none"> 1. Select Code Most Significant Octal Zero 2. I/O Address Card 3. **Address enabling signal to all I/O channels, addressed under program control, having most significant select code digit zero
$SCM1 = (\overline{TR5}) (\overline{TR4}) (TR3) (\overline{PH5})$ $= [(110) (CR1) (PH5) + (210) (CR2) (PH5)]$ $(\overline{SSFF15})$	<ol style="list-style-type: none"> 1. Select Code Most Significant Octal One 2a. I/O Address Card 3a. Address enabling signal to all I/O channels, addressed under program control, having most significant select code digit one 2b. DMA Address Encoder Card 3b. **Address enabling signal to all I/O channels, addressed by DMA Channel 1 or 2, having most significant select code digit one
$SCM2 = (\overline{TR5}) (TR4) (\overline{TR3}) (\overline{PH5})$ $= [(120) (CR1) (PH5) + (220) (CR2) (PH5)]$ $(\overline{SSFF25})$	<ol style="list-style-type: none"> 1. Select Code Most Significant Octal Two 2a. I/O Address Card 3a. Address enabling signal to all I/O channels, addressed under program control, having most significant select code digit two 2b. DMA Address Encoder Card 3b. **Address enabling signal to all I/O channels, addressed by DMA Channel 1 or 2, having most significant select code digit two
$SFC = (IOG) (\overline{TR8}) (TR7) (\overline{TR6})$	<ol style="list-style-type: none"> 1. Skip if Flag Clear, Decoded 2. Shift Logic Card 3. Tests flag status on DMA Control Card under program control
$SFS = (IOG) (\overline{TR8}) (TR7) (TR6)$	<ol style="list-style-type: none"> 1. Skip if Flag Set, Decoded 2. Shift Logic Card 3. Tests Flag status on DMA Control Card under program control

**Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Diagnostic 2. Source 3. Function
SIR = T5 (buffered)	<ol style="list-style-type: none"> 1. Set Interrupt Request 2. I/O Control Card 3. Enables interrupt logic on DMA Control Card
$\text{SKF} = (\text{ADDR6}) (\text{SFC}) (\overline{\text{FLF6}}) +$ $(\text{ADDR6}) (\text{SFS}) (\text{FLF6}) +$ $(\text{ADDR7}) (\text{SFC}) (\overline{\text{FLF7}}) +$ $(\text{ADDR7}) (\text{SFS}) (\text{FLF7})$	<ol style="list-style-type: none"> 1. Skip on Flag Signal 2. DMA Control Card 3. Flag status signal to Control Processor from DMA Channel 1 or 2
$\text{SMAR1} = (\text{CR1}) (\text{PH5}) (\overline{\text{DIN1}}) (\text{COUT})$ $= (\text{CR1}) (\text{PH5}) (\text{DIN1}) (\text{CIN})$	<ol style="list-style-type: none"> 1. Step Memory Address Register, DMA Channel 1 2. DMA Control Card 3a. Increments Direct Memory Address Register (on DMA Register Card 1) by one after completion of phase 5 word or high character output transfer 3b. Increments Direct Memory Address Register (on DMA Register Card 1) by one after completion of phase 5 word or low character input transfer
$\text{SMAR2} = (\text{CR2}) (\text{PH5}) (\overline{\text{DIN2}}) (\text{COUT})$ $= (\text{CR2}) (\text{PH5}) (\text{DIN2}) (\text{CIN})$	<ol style="list-style-type: none"> 1. Step Memory Address Register, DMA Channel 2 2. DMA Control Card 3a. Increments Direct Memory Address Register (on DMA Register Card 2) by one after completion of phase 5 word or high character output transfer 3b. Increments Direct Memory Address Register (on DMA Register Card 2) by one after completion of phase 5 word or low character input transfer
SRQ10-17, 20-27 = FLFi	<ol style="list-style-type: none"> 1. Service Request Signals to DMA 2. Flag FF on Interface Card in I/O Channel i 3. Sets Cycle Request FF of DMA Channel 1 or 2 on DMA Address Encoder Card
SSFF10-15 = (SSI1) (IOBO0-5)	<ol style="list-style-type: none"> 1. Service Select Register FF's, DMA Channel 1 2. DMA Address Encoder Card 3a. *Set: register FF stores binary 1 select code bit making up address of I/O channel to be serviced by DMA Channel 1

*Internal card function; not routed through backplane wiring

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$\overline{\text{SSFF10-25}} = (\text{SSI1}) (\overline{\text{IOB00-5}})$	<ol style="list-style-type: none"> 1. Service Select Register FF's, DMA Channel 1 (cont) 3b. *Clear: register FF stores binary 0 select code bit making up address of I/O channel to be serviced by DMA Channel 2
$\text{SSFF20-25} = (\text{SSI2}) (\text{IOB00-5})$ $\overline{\text{SSFF20-25}} = (\text{SSI2}) (\overline{\text{IOB00-5}})$	<ol style="list-style-type: none"> 1. Service Select Register FF's, DMA Channel 2 2. DMA Address Encoder Card 3a. *Set: register FF stores binary 1 select code bit making up address of I/O channel to be serviced by DMA Channel 2 3b. *Clear: register FF stores binary 0 select code bit making up address of I/O channel to be serviced by DMA Channel 2
$\text{SSI1} = (\text{ADDR6}) (\text{IOO})$	<ol style="list-style-type: none"> 1. Service Select Input Control, DMA Channel 1 2. DMA Address Encoder Card 3. *Strobes select code bits and character mode bits of DMA control word into Service Select Register and Character Mode FF, respectively, of DMA Channel 1
$\text{SSI2} = (\text{ADDR7}) (\text{IOO})$	<ol style="list-style-type: none"> 1. Service Select Input Control, DMA Channel 2 2. DMA Address Encoder Card 3. *Strobes select code bit and character mode bit of DMA control word into Service Select Register and Character Mode FF, respectively, of DMA Channel 2
$\text{STC} = (\text{IOG}) (\text{T4}) (\overline{\text{TR11}}) (\text{TR8}) (\text{TR7}) (\text{TR6})$ $= [(\text{STCD1}) (\text{CR1}) (\overline{\text{WCR1}}) (\overline{\text{DIN1}}) + (\text{STCD2}) (\text{CR2}) (\overline{\text{WCR2}}) (\overline{\text{DIN2}})] (\text{T3T4B})$	<ol style="list-style-type: none"> 1. Set Control Signal 2a. Shift Logic Card 3a. Sets I/O Control FF addressed by program instructions 2b. DMA Control Card 3b. **Sets I/O Control FF addressed by DMA Channel 1 or 2
$\text{STCD1} = (\text{ADDR6}) (\text{IOO}) (\text{IOB015})$ $\overline{\text{STCD1}} = (\text{ADDR6}) (\text{IOO}) (\overline{\text{IOB015}})$	<ol style="list-style-type: none"> 1. Set Control Decision FF 2. DMA Control Card 3a. *Set: enables generation of STC signal by DMA Channel 1 3b. *Clear: inhibits generation of STC signal by DMA Channel 1

*Internal card function; not routed through backplane wiring

**Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$STCD2 = (ADDR7) (IOO) (IOBO15)$ $\overline{STCD2} = (ADDR7) (IOO) (\overline{IOBO15})$	<ol style="list-style-type: none"> 1. Set Control Decision FF 2. DMA Control Card 3a. *Set: enables generation of STC signal by DMA Channel 2 3b. *Clear: inhibits generation of STC signal by DMA Channel 2
$STF = (IOG + SRG) [(T3) (\overline{TR8}) (\overline{TR7}) (TR6)]$	<ol style="list-style-type: none"> 1. Set Flag Signal 2. Shift Logic Card 3. Sets Flag FF addressed by program instruction
$SWCR1 = (CR1) (CIN) (T7)$	<ol style="list-style-type: none"> 1. Step Word Count Register, DMA Channel 1 2. DMA Control Card 3. Increments Word Count Register (on DMA Register Card 1) after completion of phase 5 word or character transfer by DMA Channel 1
$SWCR2 = (CR2) (CIN) (T7)$	<ol style="list-style-type: none"> 1. Step Word Count Register, DMA Channel 2 2. DMA Control Card 3. Increments Word Count Register (on DMA Register Card 2) after completion of phase 5 word or character transfer by DMA Channel 2
$SWST = (PH5) [(CR1) (DIN1) + (CR2) (DIN2)] (ENF)$	<ol style="list-style-type: none"> 1. Switch Store in T-Register 2. DMA Control Card 3. **Enables Store T-Bus in T-Register signal (STBT) on Instruction Decoder Card
$TE1 = (ADDR6) (STC)$ $\overline{TE1} = (CRS) + (FBF6) (T4T5B)$	<ol style="list-style-type: none"> 1. Transfer Enable FF, DMA Channel 1 2. DMA Control Card 3a. Set: enables DMA Channel 1 to accept service requests from selected I/O channels 3b. Clear: inhibits service request processing by DMA Channel 1
$TE2 = (ADDR7) (STC)$ $\overline{TE2} = (CRS) + (FBF7) (T4T5B)$	<ol style="list-style-type: none"> 1. Transfer Enable FF, DMA Channel 2 2. DMA Control Card 3a. Set: enables DMA Channel 2 to accept service requests from selected I/O channels 3b. Clear: inhibits service request processing by DMA Channel 2

*Internal card function; not routed through backplane wiring
 **Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
<p>TR0-15 = T-Register FF's 0-15</p>	<ol style="list-style-type: none"> 1. T-Register Bits 0 through 15 2. Arithmetic Logic Card 3. Word or character data input to DMA Character Packer Card
<p>TR0B-15B = TR0-15 (buffered)</p>	<ol style="list-style-type: none"> 1. T-Register Bits 0 through 15 Buffered 2. DMA Character Packer Card 3. Buffered TR source between TR lines and IOBO lines
<p>WCFF0-13 = (DWCI1) (IOBO0-13) <u>WCFF0-13</u> = (DWCR1)</p>	<ol style="list-style-type: none"> 1. Word Count Register FF's 0 thru 13 DMA Channel 1 2. DMA Register Card 1 3a. *Set: stores word count for DMA Channel 1 3b. *Clear: all DMA Channel 1 Word Count Register FF's reset
<p>WCFF0-13 = (DWCI2) (IOBO0-13) <u>WCFF0-13</u> = (DWCR2)</p>	<ol style="list-style-type: none"> 1. Word Count Register FF's 0 thru 13 DMA Channel 2 2. DMA Register Card 2 3a. *Set: stores word count for DMA Channel 2 3b. *Clear: all DMA Channel 2 Word Count Register FF's reset
<p>WCR1 = WCFF13 <u>WCR1</u> = DWCR1</p>	<ol style="list-style-type: none"> 1. Word Count Rollover (Overflow) FF, DMA Channel 1 2. DMA Register Card 1 3a. Set: signals completion of block data transfer by DMA Channel 1 when Word Count Register content steps to zero 3b. *Clear: Overflow FF reset
<p>WCR2 = WCFF13 <u>WCR2</u> = DWCR2</p>	<ol style="list-style-type: none"> 1. Word Count Rollover (Overflow) FF, DMA Channel 2 2. DMA Register Card 2 3a. Set: signals completion of block data transfer by DMA Channel 2 when Word Count Register content steps to zero 3b. *Clear: Overflow FF reset
<p>100 = (<u>SSFF10</u>) (<u>SSFF11</u>) (<u>SSFF12</u>)</p>	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 0), DMA Channel 1 2. DMA Address Encoder Card 3. *Enables SRQ 10 and 20 coincidence gates

*Internal card function; not routed through backplane wiring

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$\overline{100} = (\overline{SSFF10}) (\overline{SSFF11}) (\overline{SSFF12}) (CR1) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 0), DMA Channel 1 2. DMA Address Encoder Card 3. *Generates **SCL0 output
$101 = (\overline{SSFF10}) (\overline{SSFF11}) (SSFF12)$	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 1), DMA Channel 1 2. DMA Address Encoder Card 3. *Enables SRQ 11 and 21 coincidence gates
$\overline{101} = (\overline{SFF10}) (\overline{SFF11}) (SFF12) (CR1) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 1), DMA Channel 1 2. DMA Address Encoder Card 3. *Generates **SCL1 output
$102 = (\overline{SSFF10}) (SSFF11) (\overline{SSFF12})$	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 2), DAM Channel 1 2. DMA Address Encoder Card 3. *Enables SRQ 12 and 22 coincidence gates
$\overline{102} = (\overline{SSFF10}) (SSFF11) (\overline{SSFF12}) (CR1) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 2), DMA Channel 1 2. DMA Address Encoder Card 3. *Generate **SCL2 output
$103 = (\overline{SSFF10}) (SSFF11) (SSFF12)$	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 3), DMA Channel 1 2. DMA Address Encoder Card 3. *Enables SRQ 13 and 23 coincidence gates
$\overline{103} = (\overline{SSFF10}) (SSFF11) (SSFF12) (CR1) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 3), DMA Channel 1 2. DMA Address Encoder Card 3. *Generate **SCL3 output
$104 = (SSFF10) (\overline{SSFF11}) (\overline{SSFF12})$	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 4), DMA Channel 1 2. DMA Address Encoder Card 3. *Enables SRQ 14 and 24 coincidence gates
$\overline{104} = (SSFF10) (\overline{SSFF11}) (\overline{SSFF12}) (CR1) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 4), DMA Channel 1 2. DMA Address Encoder Card 3. *Generates **SCL 4 output

*Internal card function; not routed through backplane wiring

**Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
105 = (SSFF10) ($\overline{\text{SSFF11}}$) (SSFF12)	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 5), DMA Channel 1 2. DMA Address Encoder Card 3. *Enables SRQ 15 and 25 coincidence gates
$\overline{105}$ = (SSFF10) ($\overline{\text{SSFF11}}$) (SSFF12) (CR1) (PH5)	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 5), DMA Channel 1 2. DMA Address Encoder Card 3. *Generates **SCL 5 output
106 = (SSFF10) (SSFF11) ($\overline{\text{SSFF12}}$)	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 6), DMA Channel 1 2. DMA Address Encoder Card 3. *Enables SRQ 16 and 26 coincidence gates
$\overline{106}$ = (SSFF10) (SSFF11) ($\overline{\text{SSFF12}}$) (CR1) (PH5)	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 6), DMA Channel 1 2. DMA Address Encoder Card 3. *Generates **SCL 6 output
107 = (SSFF10) (SSFF11) (SSFF12)	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 7), DMA Channel 1 2. DMA Address Encoder Card 3. *Enables SRQ 17 and 27 coincidence gates
107 = (SSFF10) (SSFF11) (SSFF12) (CR1) (PH5)	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 7), DMA Channel 1 2. DMA Address Encoder Card 3. *Generates **SCL 7 output
110 = ($\overline{\text{SSFF13}}$) ($\overline{\text{SSFF14}}$) ($\overline{\text{SSFF15}}$)	<ol style="list-style-type: none"> 1. Detect Service Request (MSD), DMA Channel 1 2. DMA Address Encoder Card 3. *Enables SRQ 110 coincidence gates
$\overline{110}$ = ($\overline{\text{SSFF13}}$) ($\overline{\text{SSFF14}}$) ($\overline{\text{SSFF15}}$) (CR1) (PH5)	<ol style="list-style-type: none"> 1. Service Select Code Output (MSD 0), DMA Channel 1 2. DMA Address Encoder Card 3. *Generates **SCM 0 output
120 = ($\overline{\text{SSFF13}}$) (SSFF14) ($\overline{\text{SSFF15}}$)	<ol style="list-style-type: none"> 1. Detect Service Request 2. DMA Address Encoder Card 3. *Enables SRQ 120 coincidence gate

*Internal card function; not routed through backplane wiring

**Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$\overline{120} = (\overline{SSFF13}) (\overline{SSFF14}) (\overline{SSFF15}) (CR1) (PH5)$	<ol style="list-style-type: none"> 1. Detect Select Code Output (MSD 1), DMA Channel 1 2. DMA Address Encoder Card 3. *Generates **SCM 1 output
$200 = (\overline{SSFF20}) (\overline{SSFF21}) (\overline{SSFF22})$	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 0), DMA Channel 2 2. DMA Address Encoder Card 3. *Enables SRQ 10 and 20 coincidence gates
$\overline{200} = (\overline{SSFF20}) (\overline{SSFF21}) (\overline{SSFF22}) (CR2) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 0), DMA Channel 2 2. DMA Address Encoder Card 3. *Generates **SCL 0 output
$201 = (\overline{SSFF20}) (\overline{SSFF21}) (\overline{SSFF22})$	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 1), DMA Channel 2 2. DMA Address Encoder Card 3. *Enables SRQ 11 and 21 coincidence gates
$\overline{201} = (\overline{SSFF20}) (\overline{SSFF21}) (\overline{SSFF22}) (CR2) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 1), DMA Channel 2 2. DMA Address Encoder Card 3. *Generates **SCL 1 output
$202 = (\overline{SSFF20}) (\overline{SSFF21}) (\overline{SSFF22})$	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 2), DMA Channel 2 2. DMA Address Encoder Card 3. *Enables SRQ 12 and 22 coincidence gates
$\overline{202} = (\overline{SSFF20}) (\overline{SSFF21}) (\overline{SSFF22}) (CR2) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 2), DMA Channel 2 2. DMA Address Encoder Card 3. *Generates **SCL 2 output.
$203 = (\overline{SSFF20}) (\overline{SSFF21}) (\overline{SSFF22})$	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 3), DMA Channel 2 2. DMA Address Encoder Card 3. *Enables SRQ 13 and 23 coincidence gates
$\overline{203} = (\overline{SSFF20}) (\overline{SSFF21}) (\overline{SSFF22}) (CR2) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 3), DMA Channel 2 2. DMA Address Encoder Card 3. *Generates **SCL 3 output

*Internal card function; not routed through backplane wiring
 **Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$204 = (\overline{SSFF20}) (\overline{SSFF21}) (\overline{SSFF22})$	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 4), DMA Channel 2 2. DMA Address Encoder Card 3. *Enables SRQ 14 and 24 coincidence gates
$\overline{204} = (SSFF20) (\overline{SSFF21}) (\overline{SSFF22}) (CR2) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 4), DMA Channel 2 2. DMA Address Encoder Card 3. *Generates **SCL 4 output
$205 = (SSFF20) (\overline{SSFF21}) (SSFF22)$	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 5), DMA Channel 2 2. DMA Address Encoder 3. *Enables SRQ 15 and 25 coincidence gates
$\overline{205} = (SSFF20) (\overline{SSFF21}) (SSFF22) (CR2) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 5), DMA Channel 2 2. DMA Address Encoder 3. *Generates **SCL 5 output
$206 = (SSFF20) (SSFF21) (\overline{SSFF22})$	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 6), DMA Channel 2 2. DMA Address Encoder Card 3. *Enables SRQ 16 and 26 coincidence gates
$\overline{206} = (SSFF20) (SSFF21) (\overline{SSFF22}) (CR2) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 6), DMA Channel 2 2. DMA Address Encoder Card 3. *Generates **SCL 6 output
$207 = (SSFF20) (SSFF21) (SSFF22)$	<ol style="list-style-type: none"> 1. Detect Service Request (LSD 7), DMA Channel 2 2. DMA Address Encoder Card 3. *Enables SRQ 17 and 27 coincidence gate
$\overline{207} = (SSFF20) (\overline{SSFF21}) (SSFF22) (CR2) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (LSD 7), DMA Channel 2 2. DMA Address Encoder Card 3. *Generates **SCL 7 output
$210 = (\overline{SSFF23}) (\overline{SSFF24}) (\overline{SSFF25})$	<ol style="list-style-type: none"> 1. Detect Service Request (MSD 0), DMA Channel 2 2. DMA Address Encoder Card 3. *Enables SRQ 210 coincidence gate

*Internal card function; not routed through backplane wiring

**Pseudo control function

TABLE 4-1. (Cont'd)

Signal Mnemonic and Equation	<ol style="list-style-type: none"> 1. Definition 2. Source 3. Function
$\overline{210} = (\overline{SSFF23}) (\overline{SSFF24}) (\overline{SSFF25}) (CR2) (PH5)$	<ol style="list-style-type: none"> 1. Service Select Code Output (MSD 0), DMA Channel 2 2. DMA Address Encoder 3. *Generates ** SCM 0 output
$220 = (\overline{SSFF23}) (SSFF24) (\overline{SSFF25})$	<ol style="list-style-type: none"> 1. Detect Service Request (MSD 1), DMA Channel 2 2. DMA Address Encoder 3. *Enables SRQ 220 coincidence gate
$\overline{220} = (\overline{SSFF23}) (SSFF24) (\overline{SSFF25}) (CR2) (PH5)$	<ol style="list-style-type: none"> 1. Detect Select Code Output (MSD 1), DMA Channel 2 2. DMA Address Encoder 3. *Generates **SCM 1 output

*Internal card function; not routed through backplane wiring.
 **DMA control function.

World-Wide Hewlett-Packard Sales & Service

Call your HP Computer Specialist
at any of these convenient locations:

UNITED STATES

ALABAMA Huntsville Tel: (205) 881-4591	DELAWARE Wilmington Tel: (302) 655-6161	MASSACHUSETTS Lexington Tel: (617) 861-8960	NEW YORK Albany Tel: (518) 869-8462 Endicott Tel: (607) 754-0050 Poughkeepsie Tel: (914) 454-7330 Rochester Tel: (716) 473-9500 Roslyn, Long Island Tel: (516) 869-8400 Syracuse Tel: (315) 454-2486	OREGON Portland Tel: (503) 292-9171
ARIZONA Scottsdale Tel: (602) 945-7601 Tucson Tel: (602) 298-2313	FLORIDA Miami Shores Tel: (305) 754-4565 Orlando Tel: (305) 841-3970 St. Petersburg Tel: (813) 391-0211	MICHIGAN Southfield Tel: (313) 353-9100	MINNESOTA St. Paul Tel: (612) 645-9461	PENNSYLVANIA Monroeville Tel: (412) 271-0724 West Conshohocken Tel: (215) 248-1600
CALIFORNIA North Hollywood Tel: (213) 877-1282 Palo Alto Tel: (415) 327-6500 Sacramento Tel: (916) 482-1463 San Diego Tel: (714) 223-8103	GEORGIA Atlanta Tel: (404) 436-6181	MISSOURI Kansas City Tel: (816) 333-2445 St. Louis Tel: (314) 962-5000	NORTH CAROLINA High Point Tel: (919) 882-6873	TEXAS Richardson Tel: (214) 231-6101 Houston Tel: (713) 667-2407 San Antonio Tel: (512) 434-4171
COLORADO Englewood Tel: (303) 771-3455	ILLINOIS Skokie Tel: (312) 677-0400	NEW JERSEY Paramus Tel: (201) 265-5000 Cherry Hill Tel: (609) 667-4000	OHIO Cleveland Tel: (216) 884-9209 Dayton Tel: (513) 298-0351	UTAH Salt Lake City Tel: (801) 486-8166
CONNECTICUT East Hartford Tel: (203) 289-9394 Norwalk Tel: (203) 853-1251	INDIANA Indianapolis Tel: (317) 546-4891	NEW MEXICO Albuquerque Tel: (505) 255-5586 Las Cruces Tel: (505) 526-2485	OKLAHOMA Oklahoma City Tel: (405) 848-2801	VIRGINIA Richmond Tel: (703) 282-5451
	LOUISIANA Kenner Tel: (504) 721-6201			WASHINGTON Bellevue Tel: (206) 454-3971
	MARYLAND Baltimore Tel: (301) 944-5400 Rockville Tel: (301) 948-6370			

CANADA

ALBERTA Edmonton Tel: (403) 482-5561	MANITOBA St. James Tel: (204) 786-7581	ONTARIO Ottawa Tel: (613) 722-4223 Toronto Tel: (416) 249-9196	QUEBEC Pointe Claire Tel: (514) 697-4232
BRITISH COLUMBIA Vancouver Tel: (604) 731-5301	NOVA SCOTIA Halifax Tel: (902) 455-0511		

CENTRAL AND SOUTH AMERICA

ARGENTINA Buenos Aires	BRAZIL Sao Paulo Rio de Janeiro	MEXICO Mexico City	VENEZUELA Caracas
----------------------------------	--	------------------------------	-----------------------------

EUROPE

BELGIUM Brussels	FRANCE Orsay Lyon	Düsseldorf Frankfurt Hamburg München	NETHERLANDS Amsterdam	SWITZERLAND Zurich Meyrin-Geneva
DENMARK Naerum	GERMANY Berlin W Böblingen	ITALY Milan Rome	NORWAY Haslum	UNITED KINGDOM Slough, Bucks
FINLAND Helsinki			SWEDEN Mölnal Solna	

AFRICA, ASIA, AUSTRALIA

AUSTRALIA Melbourne Sydney Adelaide Perth, W. A.	JAPAN Osaka Nagoya Tokyo	NEW ZEALAND Wellington	SOUTH AFRICA Cape Town Johannesburg
---	--	----------------------------------	--

FOR EUROPEAN AREAS NOT LISTED, CONTACT:

Hewlett-Packard S.A., Rue du Bois-du-Lan 7,
1217 Meyrin-Geneva, Tel: (022) 41 54 00

FOR OTHER AREAS NOT LISTED, CONTACT:

Hewlett-Packard Export Marketing, 3200 Hillview Ave.,
Palo Alto, California 94304, U.S.A.; Telex: 034-8461,
Cable: HEWPACK Palo Alto.



DIGITAL COMPUTERS

HEWLETT  PACKARD

11000 WOLFE ROAD, CUPERTINO, CALIFORNIA 95014, TELEPHONE 408 257-7000, TWX 910-338-0221
EUROPE: 1217 MEYRIN-GENEVA, SWITZERLAND ● CABLE "HEWPACKSA" TEL. (022) 41.54.00