

**DA11-F unibus window
maintenance manual**

pdp11

digital

**DA11-F unibus window
maintenance manual**

1st Edition November 1972
2nd Printing April 1973

Copyright © 1972, 1973 by Digital Equipment Corporation

The material in this manual is for informational purposes and is subject to change without notice.

The following are trademarks of Digital Equipment Corporation, Maynard, Massachusetts:

DEC	PDP
FLIP CHIP	FOCAL
DIGITAL	COMPUTER LAB

CONTENTS

	Page	
CHAPTER 1	SYSTEM AND PHYSICAL DESCRIPTION	
1.1	General	1-1
1.2	Window Parameters	1-2
1.3	Physical Description	1-3
1.4	Typical Installation	1-5
1.5	Specification Summary	1-5
CHAPTER 2	THEORY OF OPERATION	
2.1	General	2-1
2.2	DA11-F Bus Cycles	2-1
2.2.1	DATO, DATOB Cycles	2-2
2.2.2	DATI, DATIP Cycles	2-3
2.2.3	Duplex Operation	2-3
2.2.4	DATO, DATOB Cycle Detailed Description	2-5
2.2.5	DATI, DATIP Cycle Detailed Description	2-7
2.3	DA11-F Block Diagram Description	2-9
2.4	Relocated Target Address	2-11
2.4.1	Window Size	2-11
2.4.2	Window Starting Address	2-12
2.4.3	Target Address	2-12
2.4.4	Target Address Calculation	2-13
2.5	Window Field Jumper Plugs	2-14
2.6	Window Bus Control	2-15
2.7	Data Path Block Diagram Description	2-17
2.8	Addressable Register Select Logic	2-19
2.9	Interrupt Control Logic	2-21
2.10	Power Fail	2-22
CHAPTER 3	PROGRAMMING INFORMATION	
3.1	General	3-1
3.2	Control and Status Register (CSR)	3-2
3.3	Output Data Buffer (DB)	3-5
3.4	Input Data Buffer (DB)	3-5
3.5	Displacement Address Register (DAR)	3-6
3.6	Relocation Address Register (RAR)	3-6
3.7	Starting Address Register (SAR)	3-8
3.8	Vector Address Register (VAR)	3-8
3.9	Programming Examples	3-8

CONTENTS (Cont)

	Page
3.9.1	Open-The-Window Program 3-8
3.9.2	Block Transfer Program 3-9
3.10	Error Reporting Procedures 3-9
CHAPTER 4	MAINTENANCE
4.1	General 4-1
4.2	Engineering Drawings 4-1
4.3	Installation 4-3
4.3.1	Customer Specifications 4-3
4.3.2	Implementing Customer Specifications 4-6
4.3.3	Starting Address Jumpers 4-7
4.3.4	Control and Status Register Address Jumpers 4-11
4.3.5	Vector Address Jumpers 4-12
4.3.6	Priority Jumper Plugs 4-12
4.3.7	Window Size Jumper Plugs 4-12
4.3.8	Mechanical Installation 4-17
4.3.9	Electrical Installation 4-17
4.3.9.1	Unibus Connection 4-17
4.3.9.2	Power Cable Connection 4-17
4.4	Preventive Maintenance 4-19
4.5	Corrective Maintenance 4-19
4.6	Diagnostic Programs 4-19
4.6.1	Bus Window Static Test 4-20
4.6.2	Bus Window Exerciser 4-20
4.7	Parts List 4-20
4.8	IC Spare Parts List 4-21

APPENDIX A IC DESCRIPTION

74H74-D-Type Edge-Triggered Flip-Flops
7488 Read-Only Memory
74123 Retriggerable Monostable Multivibrator
74151 8-Line to 1-Line Multiplexer
74174/74S174 Hex D-Type Flip-Flops
74175 Quad D-Type Flip-Flops

ILLUSTRATIONS

Figure No.	Title	Page
1-1	Unibus Window, Simplified Block Diagram	1-1
1-2	Relocated Target Address Concept	1-3

ILLUSTRATIONS (Cont)

		Page
1-3	DA11-F Module Utilization	1-4
1-4	DA11-F Functional Layout	1-4
1-5	Typical DA11-F Window Installation	1-6
2-1	Unibus Window, Simplified Block Diagram	2-2
2-2	DATO and DATOB Cycle	2-2
2-3	DATI Cycle	2-3
2-4	DATIP-DATO(B) Cycle (Read, Modify, Write)	2-3
2-5	Duplex DATO(B)/DATO(B) Cycles	2-4
2-6	Duplex DATIP-DATO(B)/DATO(B) Cycles	2-4
2-7	Duplex DATIP-DATO(B)/DATIP-DATO(B) Cycles	2-4
2-8	Detailed DATO, DATOB Cycle	2-6
2-9	Detailed DATI, DATIP Cycle	2-8
2-10	DA11-F Functional Block Diagram	2-10
2-11	Unibus Window Concept	2-11
2-12	Address Calculation Block Diagram	2-12
2-13	Jumper Plug Configuration	2-15
2-14	Bit 13 Jumper Plug Concept	2-16
2-15	Window Bus Control	2-16
2-16	DA11-F Data Paths	2-18
2-17	Address Select, Simplified Diagram	2-19
2-18	ROM Address Map	2-21
2-19	Interrupt Control for a Port	2-22
2-20	Power Fail Simplified Block Diagram	2-23
3-1	ACSR Bit Format	3-2
3-2	Cross Interrupt Logic	3-5
3-3	Output Data Buffer Bit Format	3-5
3-4	Input Data Buffer Bit Format	3-5
3-5	Displacement Address Register Bit Format	3-6
3-6	Relocation Address Register Bit Format	3-7
3-7	Starting Address Register Bit Format	3-8
3-8	Vector Address Register Bit Format	3-8
3-9	Open-The-Window Program	3-9
3-10	Block Transfer Program	3-10
4-1	DA11-F Configuration Chart	4-4
4-2	DA11-F Configuration Chart Example	4-5
4-3	Data Path Module Jumper Locations	4-7
4-4	Window Starting Address Chart	4-8
4-5	Example of Completed Window Starting Address Chart	4-9
4-6	Control Module Jumper Locations	4-11
4-7	CSR Address Chart	4-13

ILLUSTRATIONS (Cont)

		Page
4-8	Example of Completed CSR Address Chart	4-14
4-9	Vector Address Chart	4-15
4-10	Vector Address Chart	4-16
4-11	Typical Power Cable Connections for a DA11-F	4-18

TABLES

Table No.	Title	Page
2-1	Jumper Plug Selection	2-14
3-1	Register Addresses	3-1
3-2	Control and Status Registers WCSR 764XXX	3-3
3-3	Error Reporting Procedures After Illegal Window Channel Access	3-11
4-1	Decimal-To-Octal Address Conversion Chart	4-10
4-2	Window Size Chart	4-17

CHAPTER 1

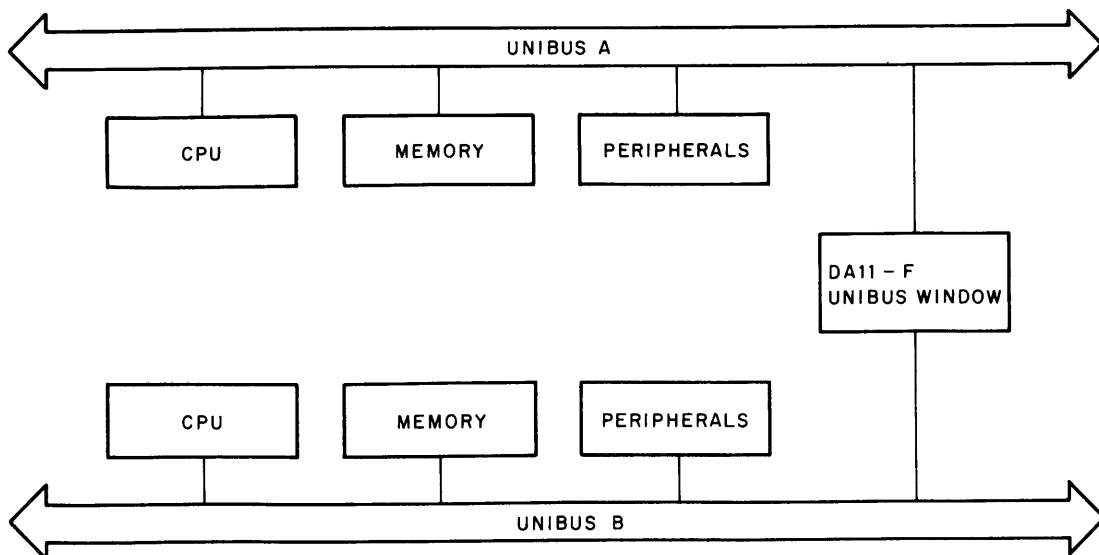
SYSTEM AND PHYSICAL DESCRIPTION

1.1 GENERAL

The DA11-F Unibus window option provides high-speed communication between two PDP-11 computer systems. The DA11-F is connected to the Unibuses of both computers as shown in Figure 1-1. For descriptive purposes, one Unibus is designated bus A and the second is designated bus B.

In order to provide high-speed data transfer between the computers, the DA11-F translates a bus cycle on one Unibus to a bus cycle on the other Unibus. In doing this, the DA11-F conforms to standard Unibus protocol.

The DA11-F is symmetrical and bilateral, i.e., data transfers may originate on either bus and data may flow in either direction. In order to accomplish this, the window contains two identical channels going in opposite directions. The logic for the two channels provides duplex capability allowing simultaneous transactions through the window from each Unibus. In addition to the channels, the window includes addressable registers and interrupt controls for full program operation by each computer.



11 - 1362

Figure 1-1 Unibus Window, Simplified Block Diagram

Some significant features of the DA11-F Unibus window are tabulated below.

- High-speed interprocessor communications (through-the-window transactions).
- Random access to virtual memory space. The user can address memory that is actually attached to a different processor.
- Duplex operation – data transfers can occur simultaneously in both directions.
- Minimum program overhead – once opened the window is transparent to the user program.
- Data transfers between peripherals and memory can pass through the window automatically.
- Window field is adjustable in size and can be located anywhere in full 128K address space.
- Data and interrupts can be transferred between computers under program control.
- Extensive built-in debugging aids.
- Compact circuitry comprising one system unit for ease of installation.
- Interchangeable modules for ease of maintenance.
- Uses system unit mounting box internal power supply.
- Compatible with any PDP-11 processor.

It is assumed that the reader is familiar with PDP-11 Unibus theory and is acquainted with the information contained in the following manuals:

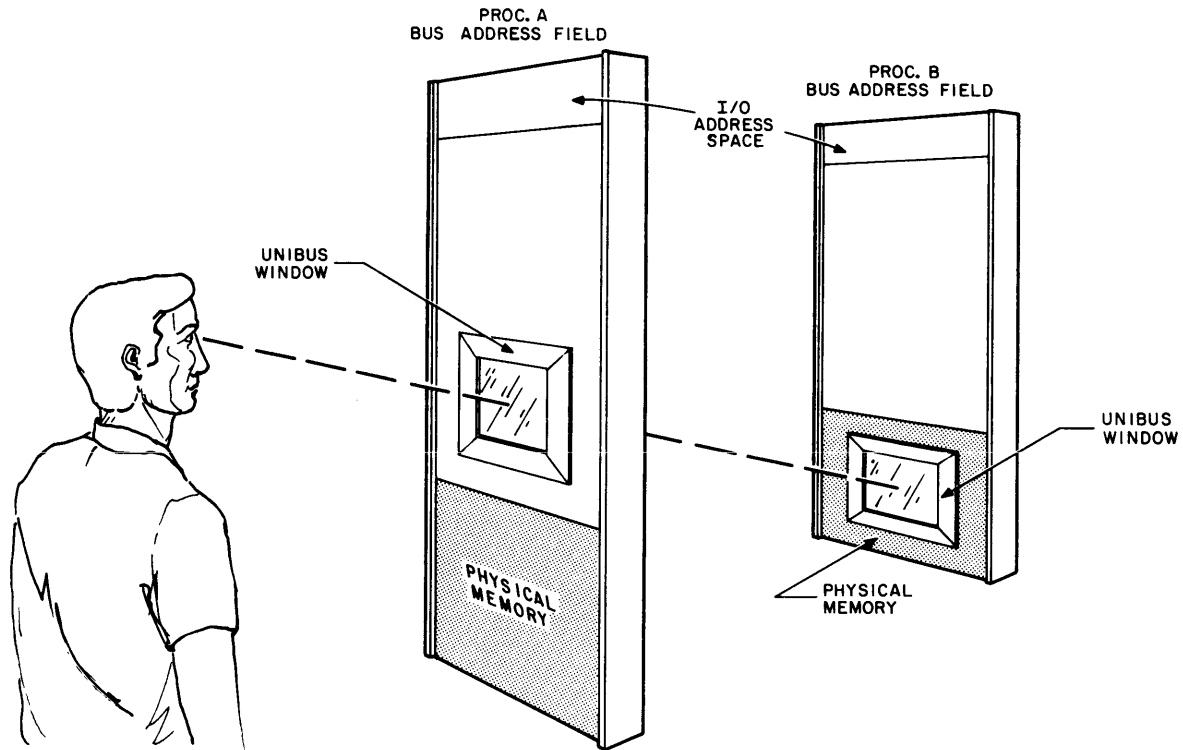
PDP-11 Processor Handbooks

PDP-11 Peripherals and Interfacing Handbook

1.2 WINDOW PARAMETERS

The originator bus is defined as the bus connected to the master initiating a bus cycle, and the target bus is defined as the bus connected to the slave device being addressed by the master. The window starting address on the originator bus can be anywhere in the full 128K of bus address space and is determined by a set of window address select jumpers. The target address, which is the address of the location containing the slave device, is relocatable from the window starting address and must be within the physical memory space of the target bus. This has the effect of making physical address space on the target bus appear as virtual address space on the originator bus (Figure 1-2). The relocated address is normally set under program control.

The size of the window field may be 1/2K, 1K, 2K, 4K, 8K, 16K, or 32K words and is determined by jumper selection.



11-1360

Figure 1-2 Relocated Target Address Concept

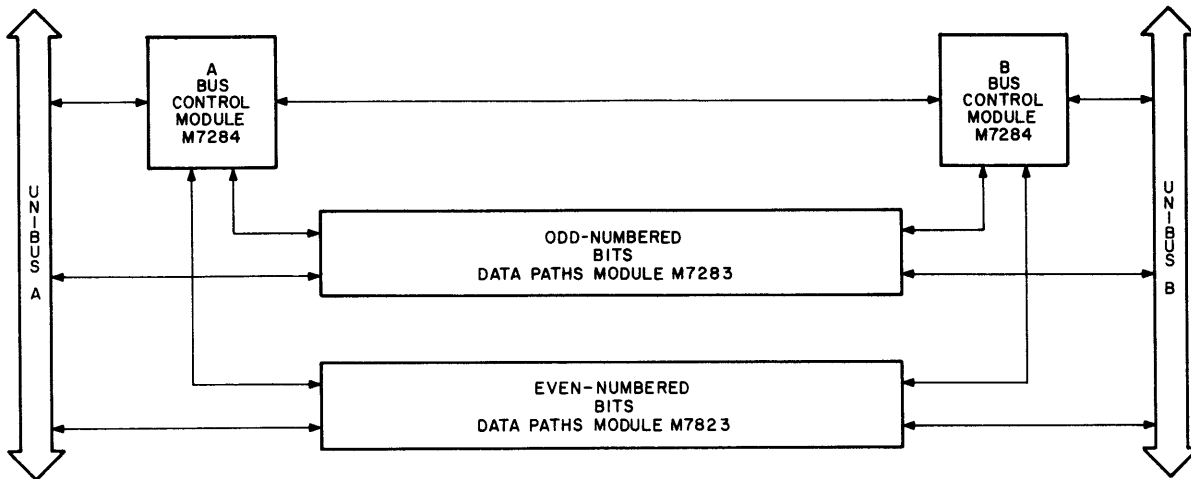
1.3 PHYSICAL DESCRIPTION

The DA11-F is mounted in a single system unit and consists of four quad-height boards and two pair of Unibus cables (A BUS IN and A BUS OUT; B BUS IN and B BUS OUT). The two inner modules contain the address and data path logic between the buses. The outside modules contain the bus control logic. One module controls the A bus and the other the B bus (Figure 1-3). Note that the Unibus connector pairs are located above the modules. Figure 1-4 shows a functional arrangement of the DA11-F. The data and address path logic is subdivided into "even bits" and "odd bits." The "even bits" board contains the even-numbered bit data and address logic for both the A and B buses, while the "odd bits" board contains the odd-numbered bit data and address logic for both buses. The A and B control modules interact with both data modules and also interact with each other.

The DA11-F is installed in any PDP-11 mounting box that can accommodate hex-height modules and draws power from the +5V regulator in the box's power supply.

	1	2	3	4
A	A Bus In	B Bus Out	B Bus In	A Bus Out
B				
C	M7284	M7283	M7283	M7284
D	A Bus Control	Bus to Bus Paths	Bus to Bus Paths	B Bus Control
E		Odd Numbered Bits	Even Numbered Bits	
F				

Figure 1-3 DA11-F Module Utilization



11-1506

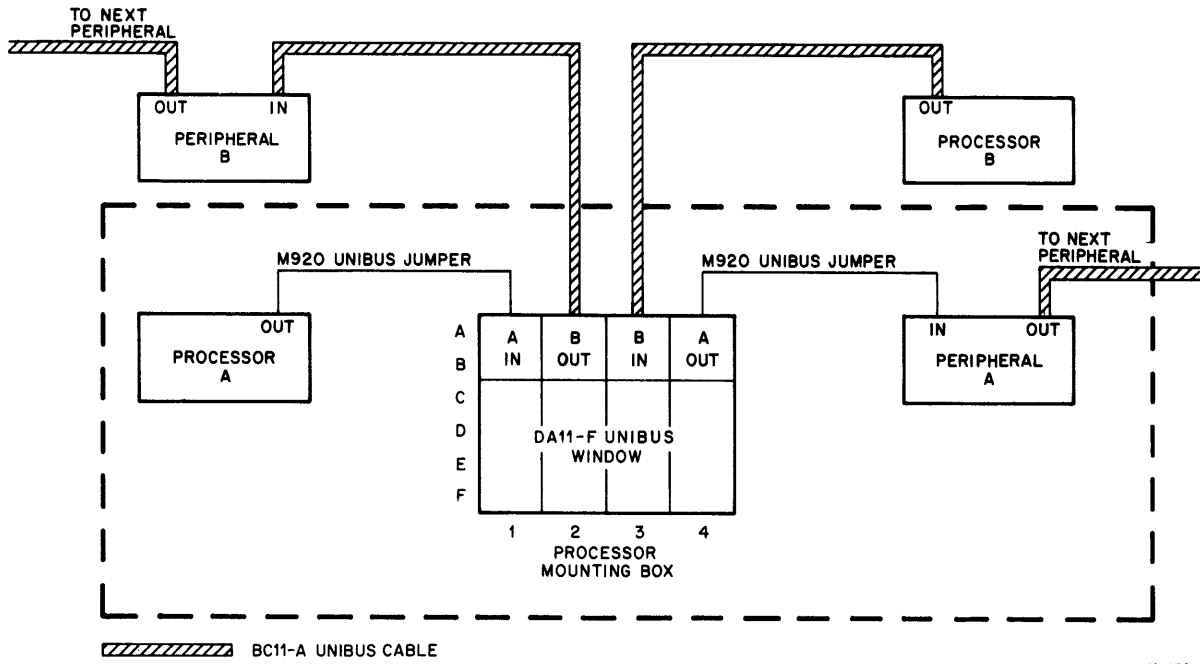
Figure 1-4 DA11-F Functional Layout

1.4 TYPICAL INSTALLATION

Figure 1-5 shows a typical DA11-F installation. Note that the A IN and A OUT connectors are located in the outside slots; this allows the use of M920 Unibus Jumper Modules to connect the DA11-F to an adjacent device. The DA11-F can be installed in either processor (or in an expansion box) but is shown in the A processor mounting box merely for illustrative purposes.

1.5 SPECIFICATION SUMMARY

Window Size:	1/2K, 1K, 2K, 4K, 8K, 16K, or 32K words. Field Adjustable.	
Window Starting Address:	For "N" K size, start on any even "N" K boundary throughout 128K bus address space, e.g., 8K window starts on any 8K boundary.	
Addressable Registers:	Seven normally assigned to user I/O area from 764000 ₈ to 764777 ₈ .	
	Control and Status	XXXX00
	Output Data Buffer	XXXX02
	Input Data Buffer	XXXX04
	Displacement Address	XXXX06
	Relocation Address	XXXX10
	Starting Address	XXXX12
	Vector Address	XXXX14
Interrupt Vector:	Assigned to floating vector field from 300 ₈ to 777 ₈ . DA11-F requires one vector that is allocated after all other options have been assigned vectors. (Can also be assigned to use vectors: 170, 174, 270, 274.)	
Priority Level:	BR7	
Bus Loading:	Places one unit bus load on each Unibus (A and B).	
Mechanical:	One system unit.	
Power:	+5Vdc, 5A maximum (drawn from mounting box power supply).	
Environment:	Temperature	10° to 50°C
	Relative Humidity	8% to 90%
Installation:	Can be installed in any system unit mounting box that accepts hex-height modules.	
Unibus Compatibility:	Can be used with any PDP-11 Family processor.	



11-1505

Figure 1-5 Typical DA11-F Window Installation

CHAPTER 2

THEORY OF OPERATION

2.1 GENERAL

This chapter describes the theory of operation of the DA11-F. Several terms used throughout this chapter are defined below:

Originator Bus – The bus connected to the master initiating a bus cycle. The master may be an NPR device or a processor.

Target Bus – The bus connected to the slave being addressed by the master. The slave may be any Unibus device but is usually memory.

Two-port Device – The DA11-F is a two-port device. A port may be thought of as the interface between a PDP-11 device and the external Unibus cable.

Window Starting Address – The starting address of the window on the originator port.

Window Field – This is the window size and can be adjusted to 1/2K, 1K, 2K, 4K, 8K, 16K, or 32K words.

Target Address – The address of the register in the slave device that is being accessed through the window.

DATO(B) – designates a DATO or DATOB operation.

DATI(P) – designates a DATI or DATIP operation.

Window Transaction – operations from originator to target in which the DA11-F synchronizes the data transfer cycles on the two Unibuses connected to the window. Also called “through-the-window” transaction.

The DA11-F is a two-port device capable of duplex operation, i.e., requests for data transfers can be handled simultaneously by both ports. In order to simplify the description, the theory of operation describes the DA11-F as a simplex device where a request for data transfer is initiated by the A bus only. The reader should be aware that a simultaneous request for data transfer can be made by the B bus.

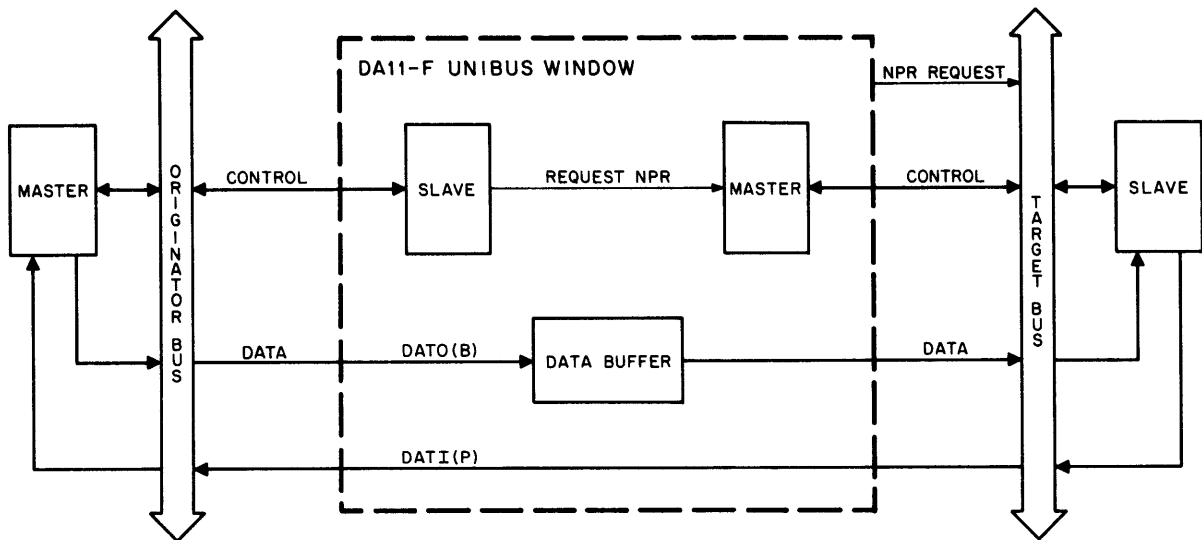
2.2 DA11-F BUS CYCLES

The DA11-F responds to all PDP-11 bus cycles for data transfers between Unibuses. When a device on the originator bus performs a bus cycle addressed to a location within its window field, the DA11-F gains control of the target bus with an NPR request and performs the same type of bus cycle but addressed to a location within the relocated window field on the target bus. In other words, a data transfer requires a bus cycle on each bus to complete the transaction.

A bus cycle operates on a master-slave “handshaking” sequence. As a result, the DA11-F appears as a *slave* to the master on the originator bus to accomplish the first bus cycle. In addition, the DA11-F appears as a *master* to the slave device on the target bus in order to accomplish the second bus cycle. This action is illustrated in Figure 2-1.

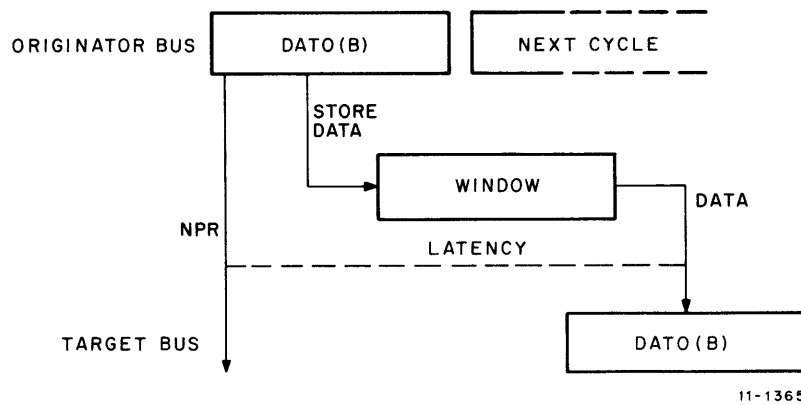
2.2.1 DATO, DATOB Cycles

If data is to be transferred from a master device on the originator bus to a slave on the target bus, a DATO(B) operation is performed. The first DATO(B) cycle causes the data on the originator bus to be transferred to a Data Buffer in the DA11-F. Since the data is stored in the DA11-F, the originator bus cycle (first DATO(B)) can be concluded. An NPR request is asserted on the target bus for the second DATO(B) cycle. When the DA11-F is granted control of the target bus, the second bus cycle causes the data to be transferred from the window Data Buffer to the target register (Figure 2-2).



11-1364

Figure 2-1 Unibus Window, Simplified Block Diagram



11-1365

Figure 2-2 DATO and DATOB Cycle

2.2.2 DATI, DATIP Cycles

If data is to be transferred from a slave device on the target bus to a master on the originator bus, a DATI(P) operation is required. In this type of operation, the originator bus cycle time is extended by the length of the NPR latency on the target bus. This is due to the fact that the data must be obtained from the slave and transferred to the master before the originator bus cycle can be terminated. The termination of the originator bus cycle is dependent on termination of the target bus cycle. This is illustrated for the DATI cycle in Figure 2-3 and for the DATIP in Figure 2-4. Data is gated directly through the window in a DATI or DATIP and is not stored in the Data Buffer as in the DATO and DATOB cycles.

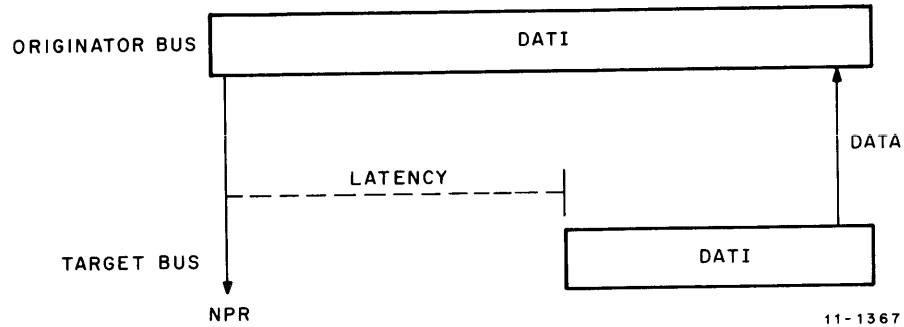


Figure 2-3 DATI Cycle

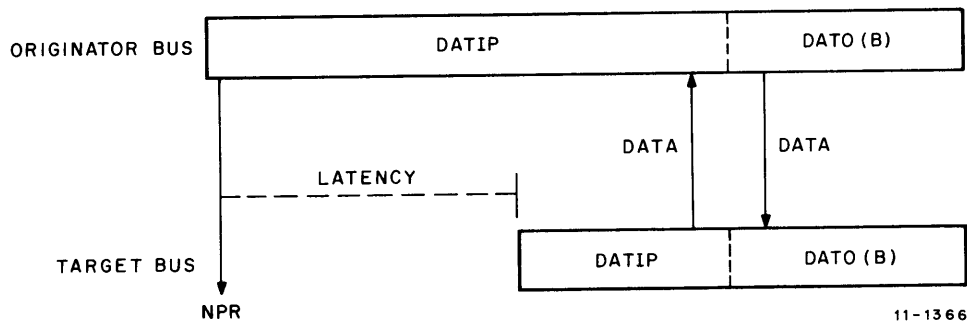


Figure 2-4 DATIP-DATO(B) Cycle (Read, Modify, Write)

2.2.3 Duplex Operation

Duplex operation occurs when requests occur on both buses simultaneously. If both ports receive DATO or DATOB cycles, the data paths are independent and the data transfer is made without interference. This is shown in Figure 2-5. Separate Data Buffers exist for both buses and separate control circuits are used.

If a DATO or DATOB cycle is requested on the A bus and a DATI or DATIP is requested on the B bus, the DATI or DATIP takes precedence and is completed before the DATO or DATOB can be completed (Figure 2-6). The sequence of operations begins with the DATO cycle where the data is transferred from the A bus to the Data Buffer in the window. This concludes the A bus portion of the DATO cycle. At this point the window gains control of the A bus and does a DATI(P) cycle. The data is directly transferred from the A bus to the B bus, thereby finishing the DATI(P) cycle on both the A bus and B bus. Then, the window gains control of the B bus and transfers the data from the window Data Buffer to the B bus to conclude the B bus portion of the DATO(B) cycle.

If DATI or DATIP cycles are requested simultaneously on both buses, neither port can gain control of its target bus. Therefore, both requests are aborted and both originating masters will time-out. This situation is illustrated in Figure 2-7.

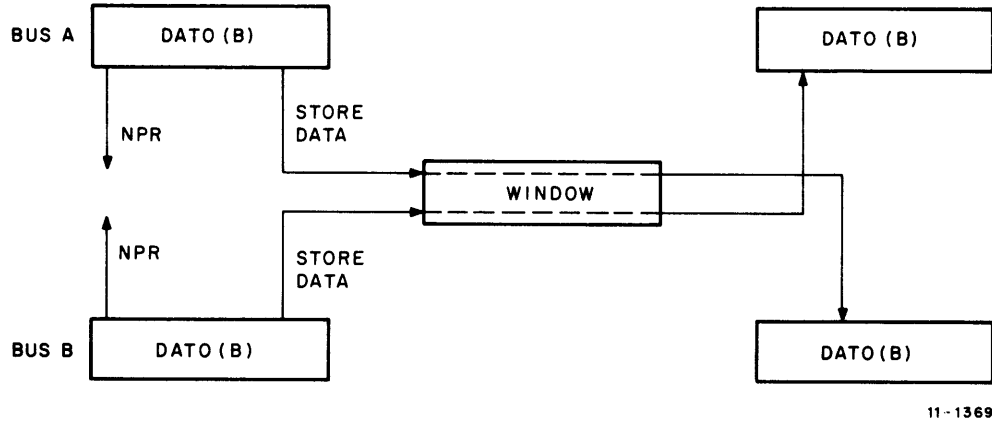


Figure 2-5 Duplex DATO(B)/DATO(B) Cycles

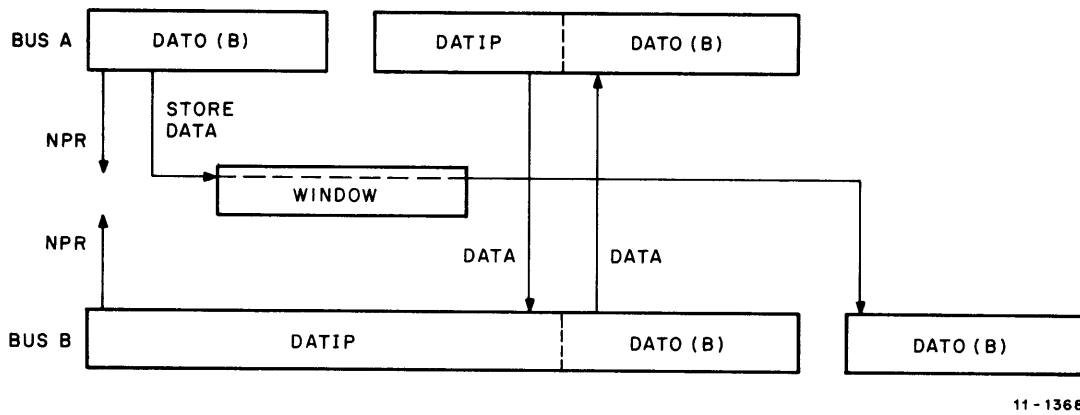


Figure 2-6 Duplex DATIP-DATO(B)/DATO(B) Cycles

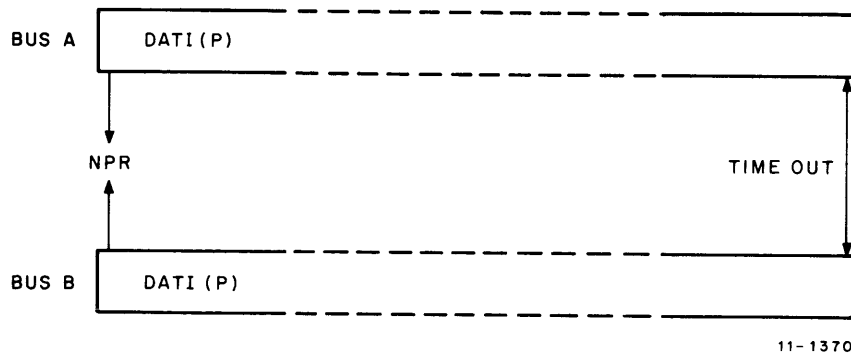


Figure 2-7 Duplex DATIP-DATO(B)/DATIP-DATO(B) Cycles

2.2.4 DATO, DATOB Cycle Detailed Description

Figure 2-8 is a detailed diagram of the DATO and DATOB cycles. Since timing between the Unibus and the DA11-F is asynchronous, precise timing diagrams are not shown. Instead, a functional flow diagram with associated timing relationships is provided. Time is shown vertically on the diagram. The master device initially gains control of the bus, asserts BBSY, and places the address, control, and data on the bus. The address is decoded by the DA11-F to determine if it has been addressed by the master. A short time later, the master asserts MSYN. If the window was selected by the master, the window performs the following three functions.

- a. Clocks the address, data, and control from the master into the following DA11-F registers.
 - Address to Displacement Address Register (DAR)
 - Data to Data Buffer (DB)
 - Control to Control Buffer (CBUF)
- b. Issues an NPR request to gain control of the target bus (if no error condition occurred).
- c. Returns SSYN to the master, indicating that the data from the master has been stored in the DA11-F. SSYN clears MSYN which, in turn, clears SSYN. BBSY is then cleared to complete the DATO cycle on the originator bus.

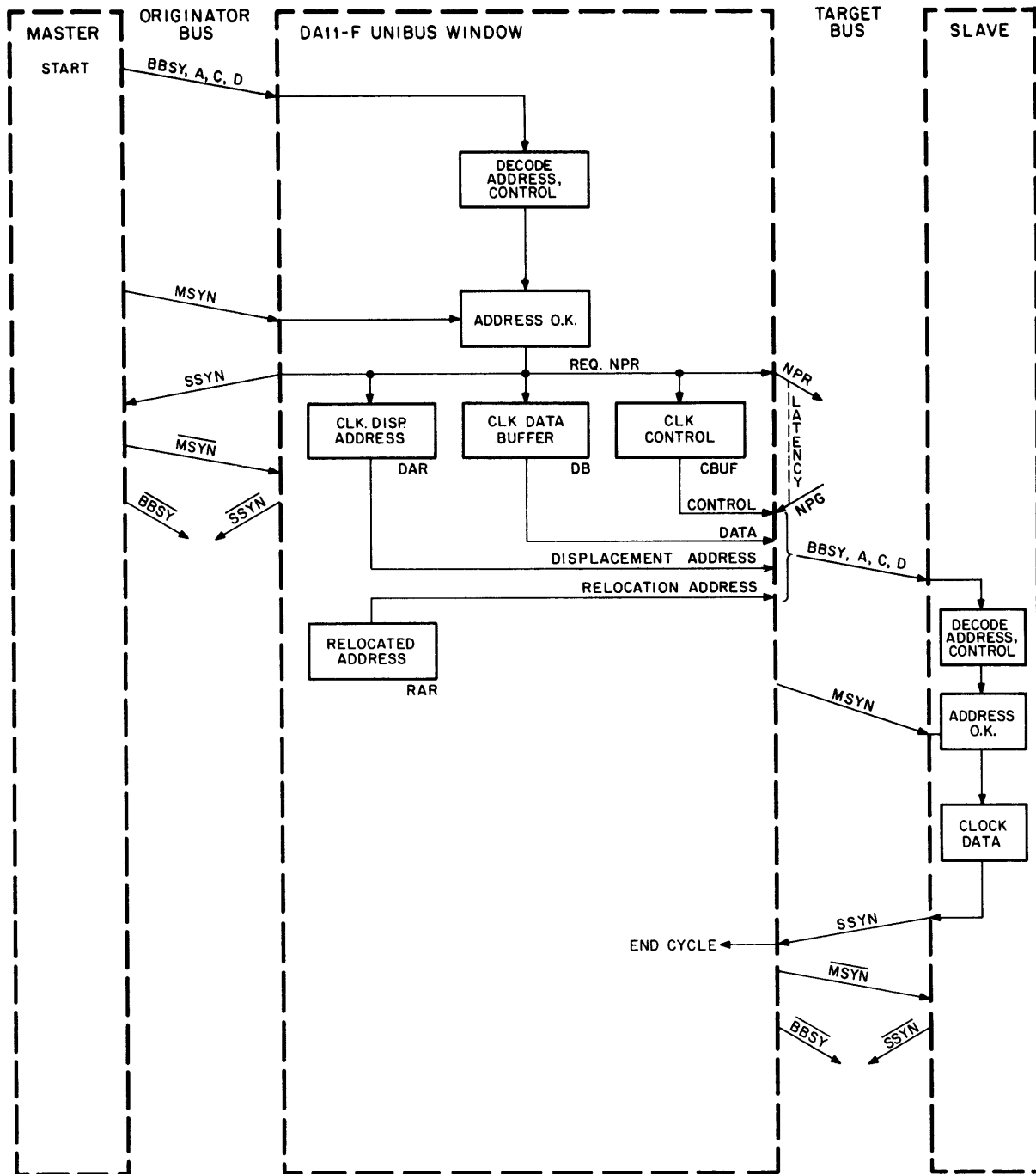
The latency time shown in Figure 2-8 is the time between issuance of the NPR request and the time that the DA11-F gains control of the target bus.

When the window gains control of the target bus, the DA11-F asserts BBSY and transfers address, data, and control information to the slave on the target bus.

NOTE

The address placed on the target bus (target address) is the summation of a displacement address and a relocation address and represents an address relocated from the originator address. This is described in detail in subsequent paragraphs in this chapter.

The slave device decodes the address to see if it is the device being addressed. The DA11-F then asserts MSYN. The slave that is being addressed clocks the data into its buffer and then returns SSYN, indicating that it has the data. SSYN clears MSYN and also generates an internal END CYCLE pulse, which clears the DA11-F bus cycle control logic. The clearing of MSYN clears SSYN and BBSY is then cleared to free the bus.



LEGEND
 A= ADDRESS
 C= CONTROL
 D= DATA

11-1371

Figure 2-8 Detailed DATO, DATOB Cycle

2.2.5 DATI, DATIP Cycle Detailed Description

Figure 2-9 shows a detailed diagram of the DATI and DATIP cycles. As in the DATO cycle, the master device initially gains control of the bus, asserts BBSY, and places the address and control information on the bus. Note, however, that no data is present in this case since the data is to be transferred from the slave to the master. The window decodes the address to determine if it is the device being selected by the master. A short time later, the master asserts MSYN. If the window was the device selected by the master, the window performs the following two functions:

- a. Clocks the address into the Displacement Address Register and the control information into the Control Buffer.
- b. Raises an NPR request on the target bus.

NOTE

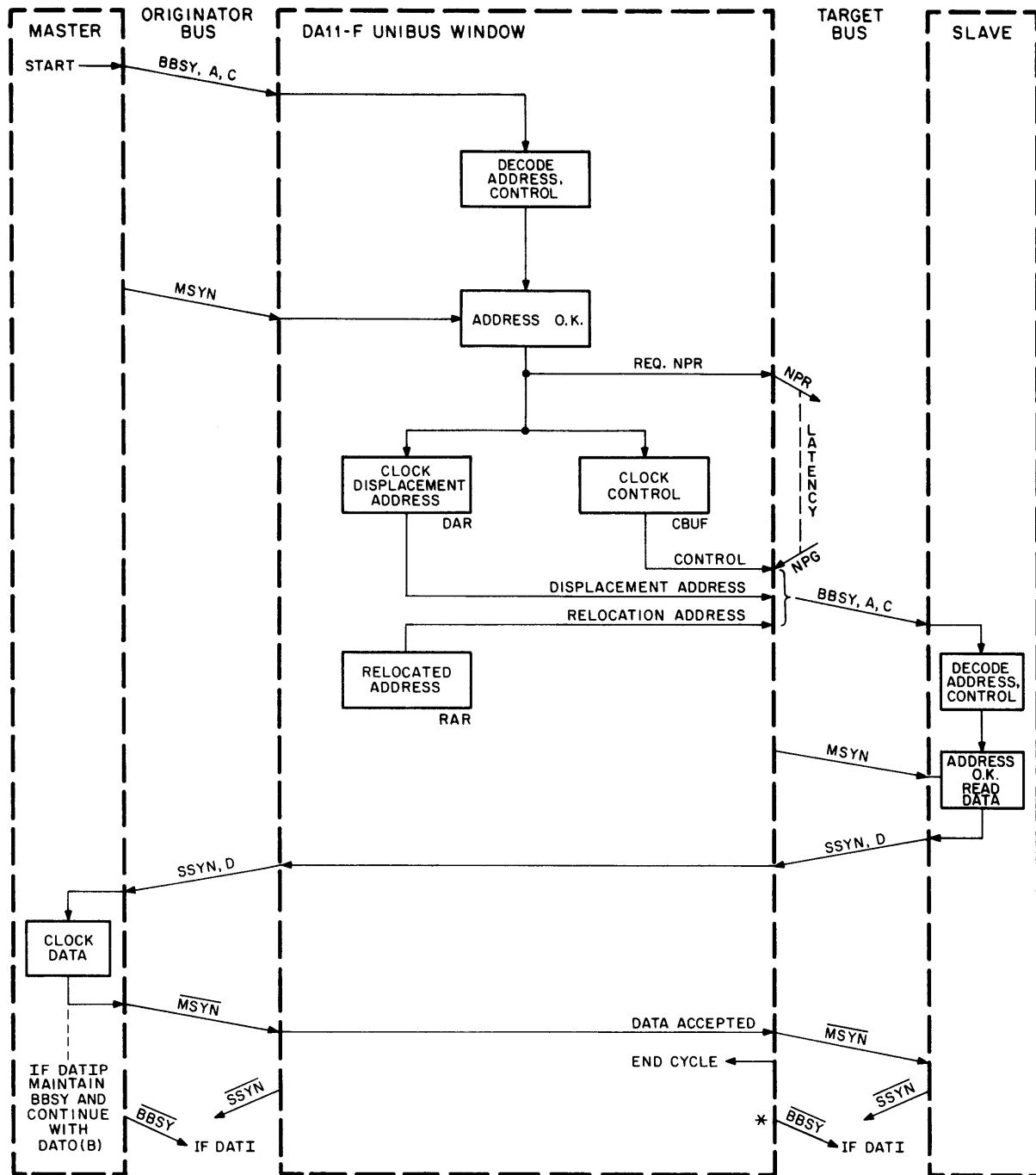
The originator bus DATI cycle is extended because the return of SSYN to the master is dependent on the completion of the target bus DATI cycle as shown. This is necessary because the data must be obtained from the slave on the target bus before it can be transferred to the master.

When the window gains control of the target bus as a result of the NPR request, it asserts BBSY and places the relocated target address and control information on the Unibus. This relocated target address is a summation of a displacement address and a relocation address and is described in subsequent paragraphs in this chapter. The slave being addressed by the window decodes the address and control information. When MSYN is asserted by the window, the slave reads the location and returns SSYN and the data to the window.

SSYN is transferred through the window to the originator bus. Simultaneously, the data is gated through the window and clocked into the master on the originator bus. The master clears MSYN, indicating that it has received the data, and after a short delay drops BBSY to release the bus (if the cycle was a DATI). The clearing of MSYN by the originator master causes the window to clear SSYN on the originator bus and also clears MSYN on the target bus. An internal signal, END CYCLE, clears the DA11-F bus cycle control logic.

If the cycle was a DATI, the window clears BBSY on the target bus. However, if the cycle was a DATIP, both the originating master and the window maintain BBSY asserted on both buses until the end of the DATO(B) cycle, which follows immediately.

Note the interlocking effect of the two DATI bus cycles, i.e., the originator bus DATI cycle cannot be terminated until SSYN from the target bus is received, and the target bus DATI cycle cannot be concluded until MSYN is cleared from the originator bus.



* If DATIP, maintain BBSY and wait for DATO(B) from originator.

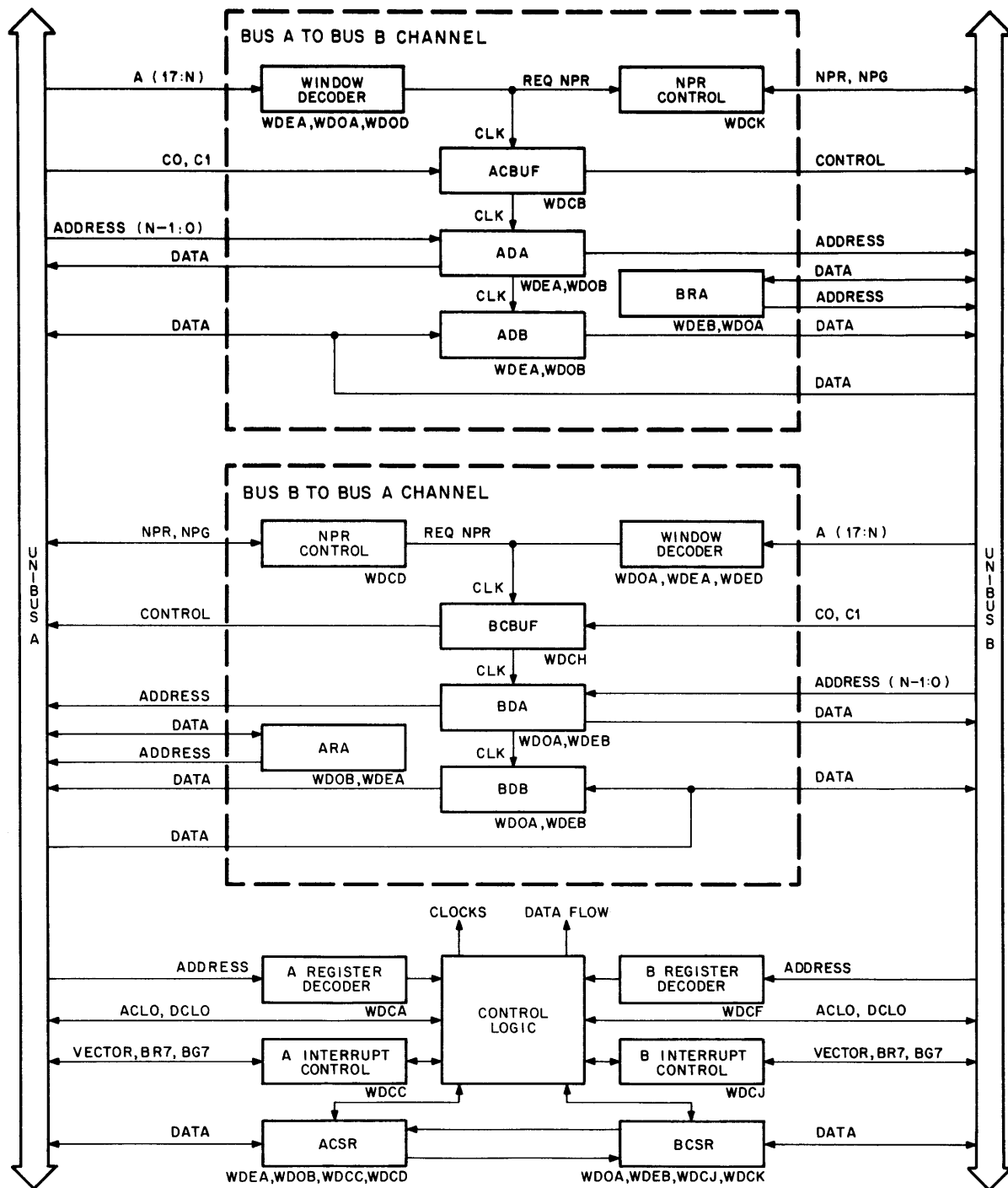
11-1372

Figure 2-9 Detailed DATI, DATIP Cycle

2.3 DA11-F BLOCK DIAGRAM DESCRIPTION

Figure 2-10 is a functional block diagram of the DA11-F, showing the two separate channels – bus A to bus B logic and bus B to bus A logic. The bottom portion of the diagram shows the control logic for each channel and some control logic that is common to both channels. Each of the bus A to bus B logic elements on the block diagram are listed below, together with a brief description of the function performed by that element. Similar functions are performed by the bus B to bus A logic elements except for a direction reversal.

- a. Window Decoder – used to decode high-order address bits (17:N) (Paragraph 2.4.3). Jumper plugs at the input to the window decoder determine the window size while jumpers in the window decoder itself determine the window starting address. The high-order bus address bits are compared to the jumpers in the window decoder and if a match occurs, the circuit following is enabled.
- b. NPR Control – This circuitry asserts an NPR request on the target bus if the high-order address bits on the originator bus compare with the internal jumpers in the window decoder. When the request is acknowledged, NPG (Non-Processor Grant) is returned.
- c. ACBUF – A buffer register to store control lines C0 and C1, which are used to determine the type of bus cycle to be performed.
- d. ADA – The A bus Displacement Address Register used to store low-order address bits (N–1:0) (Paragraph 2.4.3) from the originator bus. These bits are transferred to the target bus through the ADA. The contents of this register can be read back to the originator bus as data.
- e. BRA – The Relocation Address Register for the target bus. This register is loaded under program control and determines the upper portion (17–N) of the relocated target address.
- f. ADB – The Data Buffer used to store the data during a DATO(B) bus cycle. This buffer is not employed during DATI(P) operations. The contents of this buffer can be read back on the originator bus.
- g. A Register Decoder – The A register decoder consists of a decoding network to select one of the seven addressable DA11-F registers (refer to Chapter 3) during program transfers to the DA11-F.
- h. A Interrupt Control – The interrupt control logic enables the DA11-F to cause an interrupt when the window gains bus control via one of the bus request (BR) lines. The processor acknowledges the interrupt request with a bus grant signal, and the DA11-F then sends the processor the vector address to initiate the device service routine.
- i. Control Logic – This element contains the logic used to generate the various clock signals and data multiplexer select signals. This logic also is used to interrelate all the various control elements.
- j. A CSR – The A port Control and Status Register used to pass parameters and interrupt requests from the A port to the B port. The Control and Status Register is also used to define allowable window operations and to report error conditions.



LEGEND:
 ACBUF, BCBUF
 ADA, BDA
 ADB, BDB
 ARA, BRA
 ACSR, BCSR

Control Signal Buffers
 Displacement Address Register A, B
 Data Buffer Register A, B
 Relocation Address Register A, B
 Control and Status Register A, B

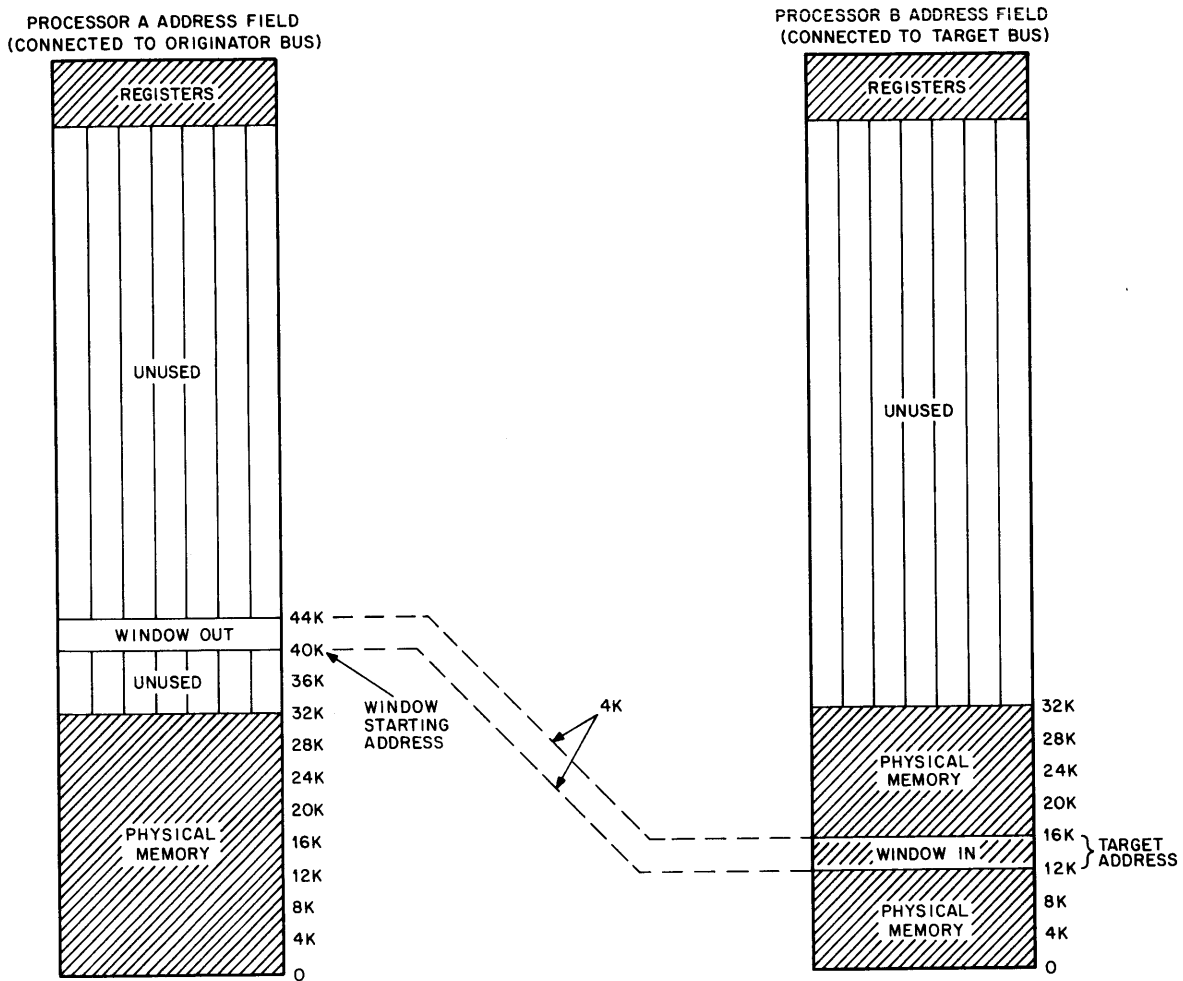
11-1361

Figure 2-10 DA11-F Functional Block Diagram

2.4 RELOCATED TARGET ADDRESS

The relocated target address represents the address in physical memory on the target bus. This address appears as virtual memory to the processor connected to the originator port. Figure 2-11 illustrates this idea and shows a 4K window with a starting address of 40K on the originator port. The 4K window is relocated to 12K to 16K in the B processor's physical memory. Thus, the 12K to 16K of physical memory in processor B appears as additional processor A memory. Note that the window has been relocated from 40K to 44K on the originator port to 12K to 16K on the target port.

From the foregoing description, it can be concluded that three parameters are variables: 1) the window size, 2) the window starting address, and 3) the relocated target address. These are described in the following paragraphs.



11-1374

Figure 2-11 Unibus Window Concept

2.4.1 Window Size

The window size or window field can be 1/2K, 1K, 2K, 4K, 8K, 16K, or 32K words. Window size is established by inserting a set of jumper plugs at the input to a hardwired window decoder. This is normally accomplished during installation of the DA11-F.

2.4.2 Window Starting Address

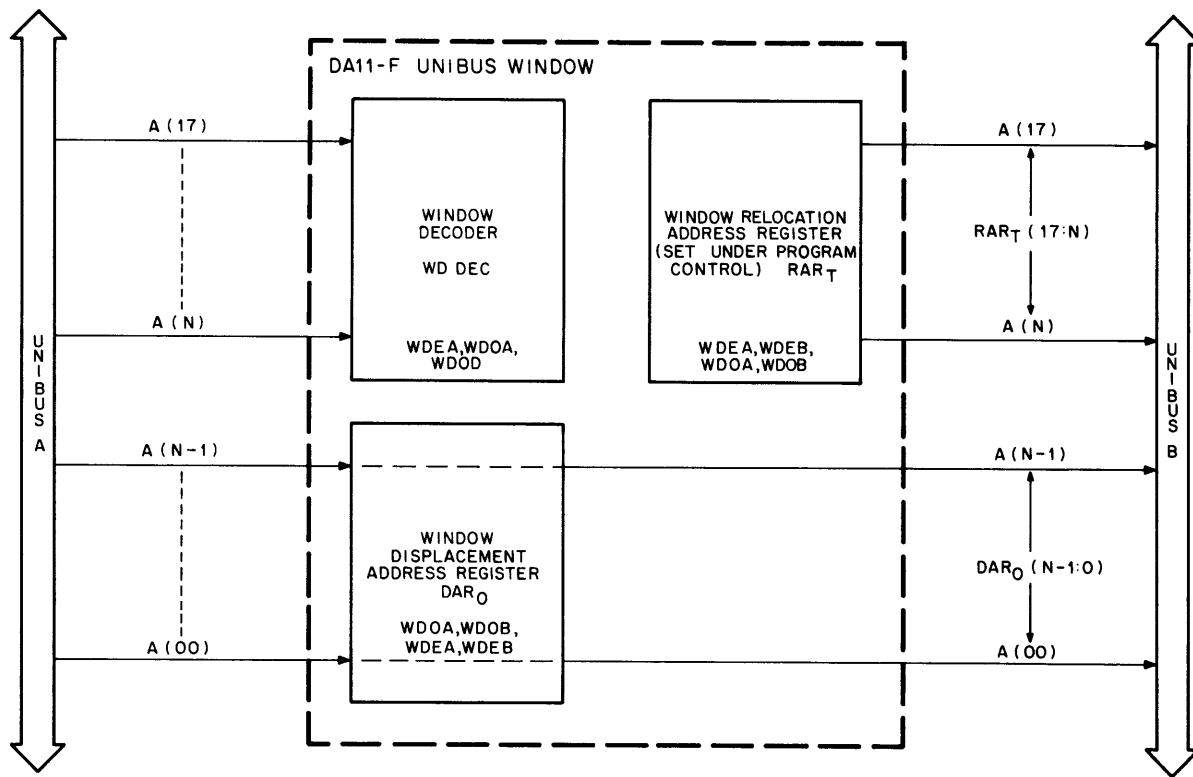
The window starting address (originator address) is determined by a set of window address select jumpers located in the window decoder (WDOD-A Bus; WDED-B Bus). The starting address is normally configured during installation of the DA11-F.

NOTE

Sheet designations of the logic prints are referenced, where applicable. For example, the A bus window decoder is on sheet WDOD as referenced above.

2.4.3 Target Address

The target address is the address of the slave device that the user desires to access. This address is derived from summation of a relocation address and a displacement address (Figure 2-12). The relocation address is program controllable and specifies bits (17:N) of the target address. N can be made equal to 10, 11, 12, 13, 14, 15, or 16 depending on the desired window size as shown below.



$RAR_T(17:N)$ = Window Relocation Address Register Target.
 $DAR_0(N-1:0)$ = Window Displacement Address Register Originator.
 Target Address = $RAR_T(17:N) + DAR_0(N-1:0)$.

11-1373

Figure 2-12 Address Calculation Block Diagram

Window Field	Window Decoder (Relocation Address) (17:N)	Window Displacement Address Register (N-1:00) (Displacement Address)
1/2K	17:10	09:00
1K	17:11	10:00
2K	17:12	11:00
4K	17:13	12:00
8K	17:14	13:00
16K	17:15	14:00
32K	17:16	15:00

For example, the window decoder decodes bits 17 through 11 (N=11) to establish a 1K window size. The remaining address bits (10 through 00) are used to form the displacement address. Address bits (17:N) are replaced on the target bus by the contents of the DA11-F Relocation Register. Displacement address bits (N-1:00) are not replaced, but are simply transferred to the target bus via the DA11-F Displacement Address Register.

2.4.4 Target Address Calculation

The two equations below are used to calculate the target address. In the first equation, bus address bits (17:N) are compared with jumpered address bits (17:N) in the window decoder. If they agree bus address bits (N-1:00) are stored in Displacement Address Register DAR_0 . The second equation states that the target address is equal to the relocation address plus the displacement address.

- a. If $A_0(17:N) = WD\ DEC(17:N)$, then
 $A(N-1:00)$ to $DAR_0(N-1:00)$
- b. $A_T(17:00) = RAR_T(17:N) + DAR_0(N-1:00)$

To illustrate how the target address is calculated, an example is given. The example shows an originator bus address of 140646 and a desired target address of 520646. The window decoder detects bits 17 through 13, designating a 4K window. If originator bus address bits 17 through 13 match window decoder bits 17 through 13, the displacement address (bits 12 through 00) is stored and combined with the relocated address (bits 17 through 13) to yield 520646.

Example:

- A_0 (originator address) = 140646₈
- Window = 4K; therefore, N = 13 and window decoder decodes bits 17 through 13.
- Window Field = 140000₈ to 157777₈ (24K to 28K)
- DAR_0 (Displacement Address) = 0646₈ (bits 12 through 00)
- RAR_T (Relocation Address) = 52₈ (bits 17 through 13)
- A_T (Relocated Target Address) = 520646₈

BUS ADDRESS

	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Originator Address (140646)	0	0	1	1	0	0	0	0	0	1	1	0	1	0	0	1	1	0
WD Decoder (14)	0 0 1 1 0 Equals: 0 0 1 1 0					↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
Relocation Address (52)	1	0	1	0	1	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
Displacement Address (520646)	1	0	1	0	1	0	0	0	0	1	1	0	1	0	0	1	1	0

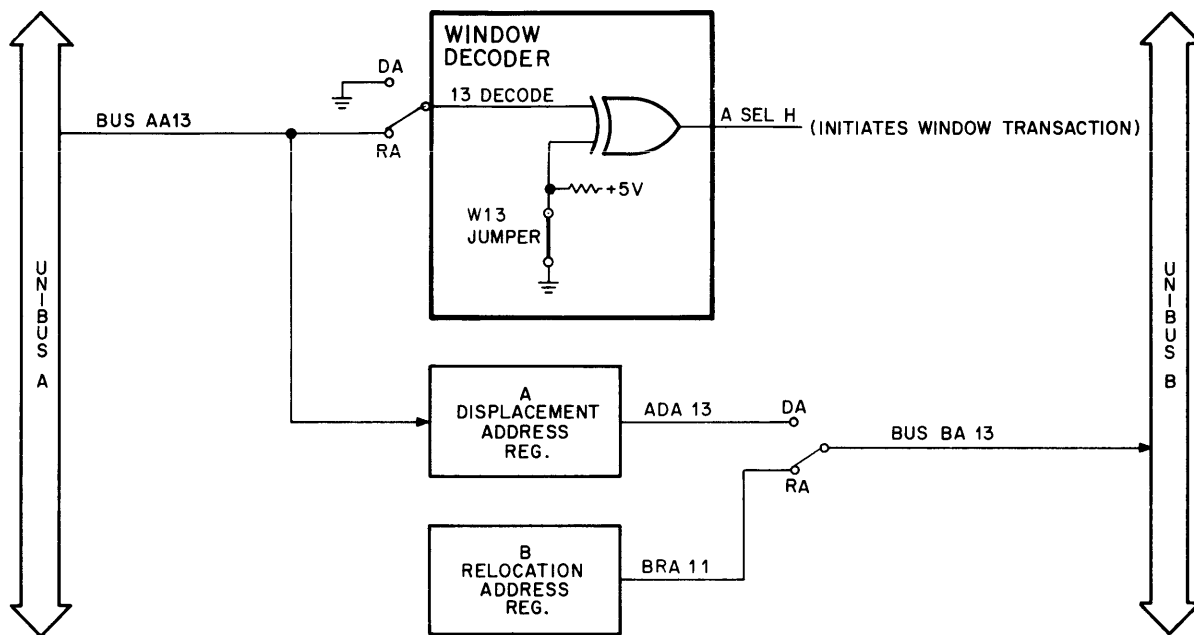
2.5 WINDOW FIELD JUMPER PLUGS

The size of the window is established by a group of jumper plugs (WDEA,WDOA) associated with address bits 15 through 10. The jumper plugs are inserted in sockets at the input to the hardwired window decoder. A DA (displacement address) or an RA (relocation address) plug can be inserted in the jumper plug receptacle associated with each of these six bits. The jumper plugs determine whether bits 15 through 10 are part of the displacement address or part of the relocation address. For example, N=13 for a 4K window as previously described. The relocation address is then designated by RA (17:13), and the displacement address is designated by DA (12:00). Since bits 15 through 13 are relocation bits, RA plugs are inserted in their plug receptacles, and since bits 12 through 10 are displacement address bits, DA plugs are inserted in the receptacles associated with these bits. Table 2-1 shows the type of plugs inserted to accommodate the various window sizes.

**Table 2-1
Jumper Plug Selection**

Window Size (Words)	Address Bits						Relocation Address	Displacement Address
	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10		
1/2K	RA	RA	RA	RA	RA	RA	(17:10)	(9:00)
1K	RA	RA	RA	RA	RA	DA	(17:11)	(10:00)
2K	RA	RA	RA	RA	DA	DA	(17:12)	(11:00)
4K	RA	RA	RA	DA	DA	DA	(17:13)	(12:00)
8K	RA	RA	DA	DA	DA	DA	(17:14)	(13:00)
16K	RA	DA	DA	DA	DA	DA	(17:15)	(14:00)
32K	DA	DA	DA	DA	DA	DA	(17:16)	(15:00)

A simplified block diagram of this action is shown in Figure 2-13, where bus A is the originator bus and bus B the target bus. The figure shows the complete path for address bit 13 in a window equal to 4K words. Bit 13 from the A bus is applied to the A window decoder. Therefore, bit 13 of the B Relocation Address Register is supplied to the target bus. This action is caused by the insertion of an RA plug in the bit 13 position. Bits 15 and 14 are routed through identical circuits. On the other hand, bits 12 through 10 are displacement address bits. These bits are first stored in the A Displacement Address Register and then transferred to the target bus during the bus cycle. This action is caused by inserting DA plugs in these bit positions.



NOTES:
 Bits (15:10) employ circuitry similar to that shown above. Bit 13 is shown for simplicity.
 Bits (09:00) are hardwired from the Displacement Address Register and are not decoded.
 Bits (17:16) are hardwired from the Relocation Address Register.

11-1504

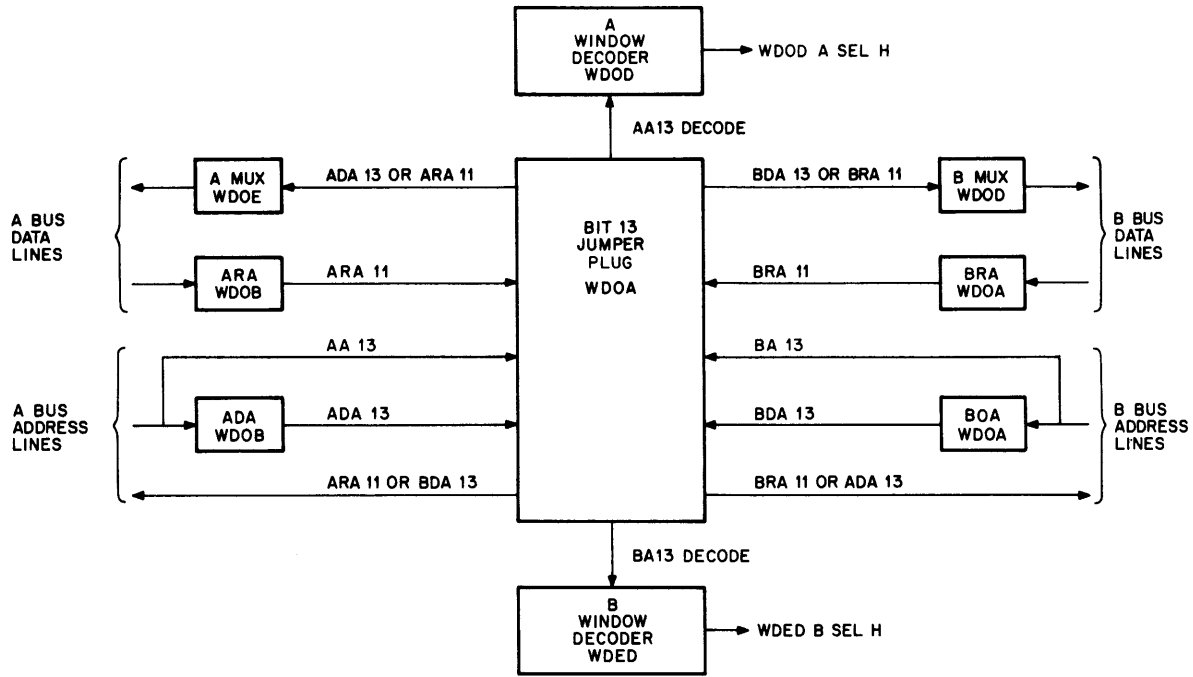
Figure 2-13 Jumper Plug Configuration

Figure 2-14 shows the jumper plugs on a block diagram level for one particular bit – namely, bit 13. The diagram shows DA bit 13 and RA bit 11 applied to the jumper plug, which selects one of the two bits. The jumper plug routes the bits to the appropriate data multiplexer, window decoder, or address line, depending on whether the bit is part of the displacement address or relocation address.

2.6 WINDOW BUS CONTROL

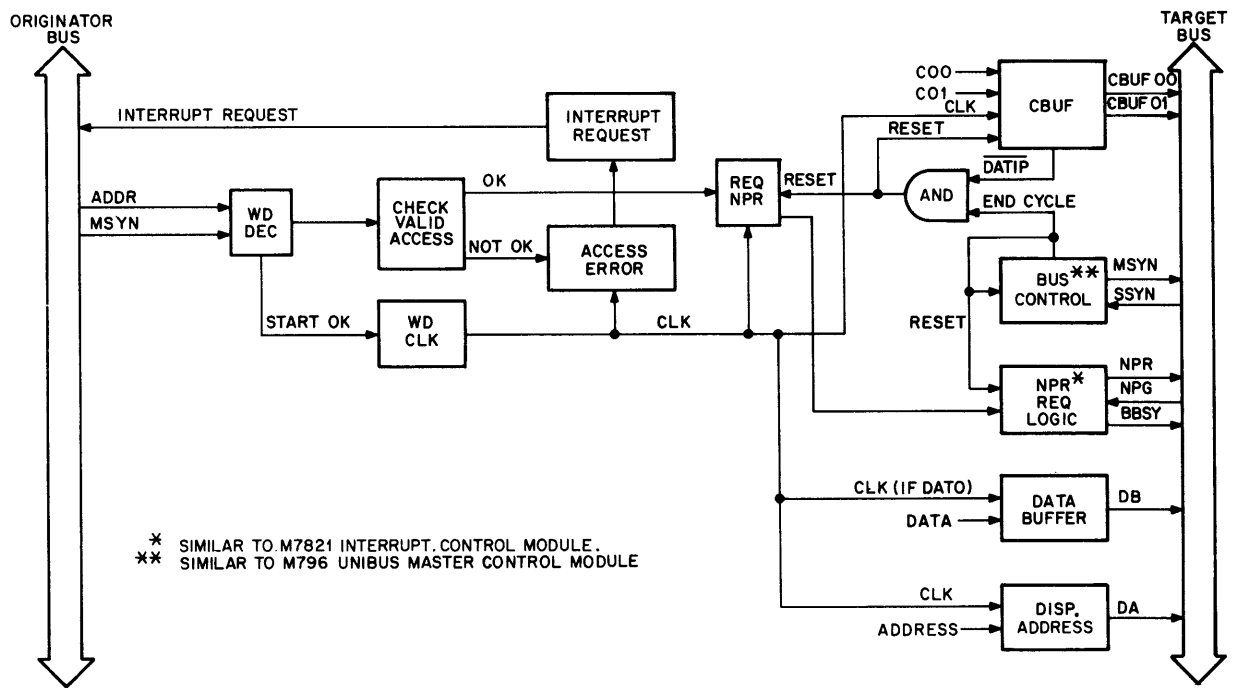
Figure 2-15 represents a simplified logic diagram of the window bus control circuits. The originator bus address and MSYN are applied to the window decoder, which then generates an internal clock pulse.

If the access is invalid (e.g., because the target bus transfer enable bit is not set), the ACCESS ERROR flip-flop is set and an interrupt request is generated on the originator bus (if its interrupt enable bit is set). The program is thus informed that an access error has occurred. The cause of the error can be evaluated by examining the contents of the CSR. (Refer to Paragraph 3.10 for detailed description of error conditions).



11-1507

Figure 2-14 Bit 13 Jumper Plug Concept



11-1508

Figure 2-15 Window Bus Control

If, on the other hand, the access is valid, the REQUEST NPR flip-flop is set and an NPR request is asserted on the target bus. When granted control of the bus, the bus control circuit asserts BBSY and initiates the type of bus cycle specified by the control lines.

Three other events occur at the time of the internal clock pulse from the window decoder. First, the originator bus control signals C00 and C01 are stored in the Control Bit Buffer Register (CBUF). Second, the displacement address (i.e., the low-order portion of the originator bus address) is stored in the Displacement Address Register (DAR). And third, if the cycle is a DATO(B), the originator's data is stored in the Data Buffer (DB).

Since Unibus protocol requires that a DATIP cycle must be followed by a DATO(B) cycle and also that the master must maintain BBSY asserted continuously throughout both cycles, the DATIP flip-flop (part of CBUF) is set by the internal clock pulse when that type of cycle is detected. The DATIP flip-flop inhibits the resetting of REQUEST NPR and BBSY until the end of the DATO(B) cycle. Thus the DA11-F maintains BBSY properly asserted on the target bus during a DATIP-DATO(B) sequence.

At the end of each bus cycle, the bus control circuit generates an END CYCLE pulse to reset itself. If the cycle is not a DATIP, this pulse also resets REQUEST NPR, BBSY, and CBUF.

The general description above is for both the A and B buses. The following chart defines the above-mentioned signals for both the A and B buses and designates the applicable drawings where the signals originate.

Signal Name	A Bus	B Bus
ACCESS ERROR	WDCC AACC ERR (1) H	WDCJ BACC ERR (1) H
Interrupt Request	WDCC ABUS REQUEST L	WDCJ BBUS REQUEST L
REQUEST NPR	WDCB AREQ NPR (1) H	WDCH BREQ NPR (1) H
Window Decoder Clock Pulse	WDCB ACLK H	WDCH BCLK H
C0, C1 Control Signals	BUS AC1 L, BUS AC0 L (WDCB)	BUS BC1 L, BUS BC0 L (WDCH)
DATIP	WDCB ADATIP (1) H	WDCH B DATIP (1) H
END CYCLE	WDCD AEND CYCLE (1) H	WDCK BEND CYCLE (1) H
BBSY	BUS ABSY L (WDCA)	BUS BBSY L (WDCF)
CBUF	WDCB ACBUF 01 (1) H	WDCH BCBUF 01 (1) H

2.7 DATA PATH BLOCK DIAGRAM DESCRIPTION

Figure 2-16 is a block diagram of the DA11-F data paths and shows how the data is routed through various registers during data transfer operations.

The key elements associated with routing the data are the data multiplexers. Each multiplexer has eight input lines and one common output. The multiplexer select signals (A MUX SEL, B MUX SEL) direct which input line is to be connected to the output line.

The data multiplexers are used in three modes – program addressable mode, window mode, and interrupt mode. In program addressable mode, a program instruction may read data from an addressable register. Each register is connected to a particular multiplexer input, which is selected by address bits 1, 2, and 3 (bit 0 being disregarded). The registers are connected to the multiplexer inputs in the same order in which the registers are addressed. The CSR is connected to input 0, Input Data Buffer to input 1, Output Data Buffer to input 2, displacement address to input 3, relocation address to input 4, the window starting address to input 5, and the interrupt vector address to input 6. During an interrupt cycle, input 6 is gated to the bus. This position is to place the interrupt vector address on the Unibus data lines. Input 7 is the position that is used to transfer the data directly from the target bus to the originator bus during DATI(P) operations. The data is gated through the window directly with no buffering.

In window mode operations during DATO(B) bus cycles, the data is transferred from the originator bus Data Buffer to input 2 of the target bus data multiplexer.

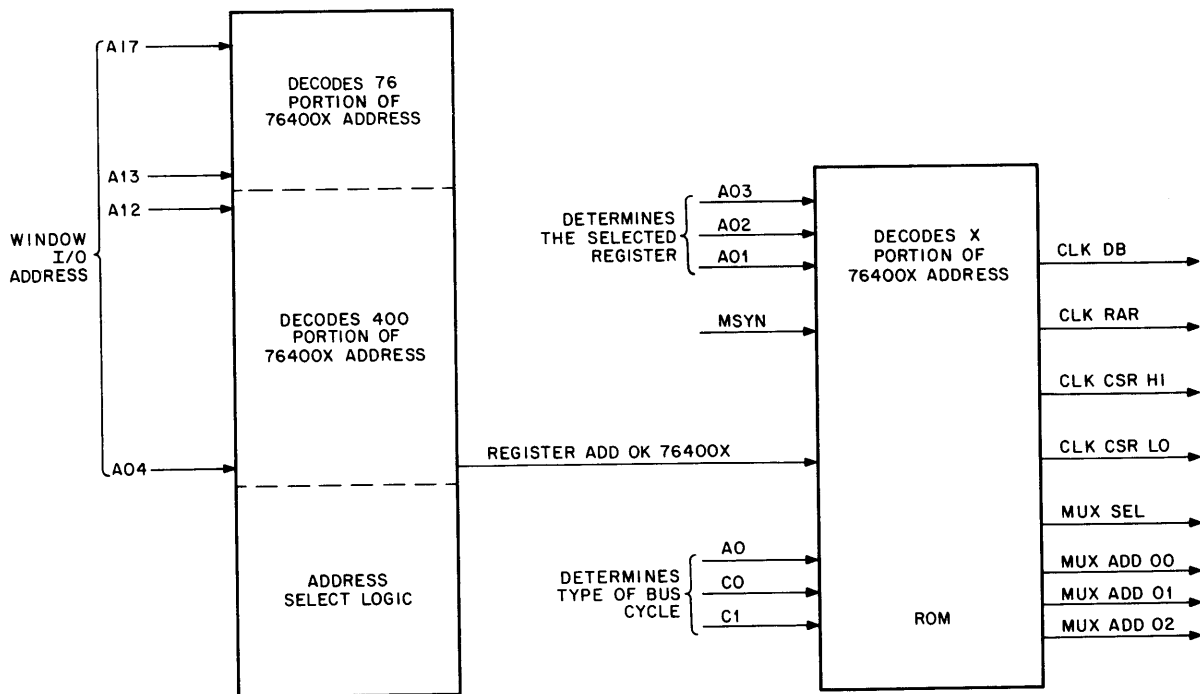
NOTE

The A Bus and B Bus multiplexers are divided into odd and even bits and are located on the sheets indicated below:

- A Bus multiplexer – odd bits WDOE
- A Bus multiplexer – even bits WDED
- B Bus multiplexer – odd bits WDOD
- B Bus multiplexer – even bits WDEE

2.8 ADDRESSABLE REGISTER SELECT LOGIC

Figure 2-17 shows a simplified diagram of the logic used to select the addressable window registers. Address bits 17 through 13 establish the high-order bits of the I/O address area. Address bits 12 through 4 select the particular block of locations used by the DA11-F. Each address bit is compared against +3V for a 1 or ground for a 0. The output of the address selector is applied to a ROM (read-only memory). The ROM decodes the register specified by address bits A03, A02, and A01 and also decodes the type of bus cycle specified. The outputs of the ROM are used either to clock the specified register that is being written into, or to gate the specified register to the bus data lines when the register is being read (MUX ADDR 0, 1, and 2).



Address illustrated 76400X

11-1375

NOTE:
Circuit is shown on sheet WDCA for A bus and sheet WDCF for B bus

Figure 2-17 Address Select, Simplified Diagram

Figure 2-18 shows the ROM mapping configuration. A total of 32 decimal locations is shown. The OCTAL LOC column is the octal equivalent of the decimal locations and represents the address where the binary data (in the BINARY DATA column) is stored. The first digit in the OCTAL LOC column specifies four groupings (each grouping being associated with eight ROM locations). These groupings represent the following commands.

Grouping	Command
0	CLOCK LOW BYTE
1	CLOCK FULL WORD
2	CLOCK HIGH BYTE
3	READ

The first three groups of ROM locations (0, 1, and 2) represent 24 of the 32_{10} addresses and are used during DATO operations to determine which of the seven addressable registers will get clock pulses. The fourth grouping (3_8) is used during DATI operations to generate MUX SEL signals, which select the addressable register that will be read. Address bit A0 and control lines C0 and C1 are combined to determine the type of bus cycle (DATO, DATI, DATIP) as shown below:

A0	C0	C1	
X	0	0	DATI
X	0	1	DATIP
X	1	0	DATO
0	1	1	DATOB CLK LO BYTE
1	1	1	DATOB CLK HI BYTE

X = don't care

A0, C0, C1 are decoded to yield one of the four command groupings just discussed. The second octal digit in the OCTAL LOC column sequences from 0 through 7 and represents the address of the selected register as shown below:

Register Address (Second Octal Digit)	Register
0	CSR
1	Input Data Buffer
2	Output Data Buffer
3	Displacement Address
4	Relocation Address
5	Starting Address
6	Vector Address
7	Direct bus-to-bus data path during DATIP

For example, if an instruction is executed that writes into the low byte of the CSR, ROM address 00_8 should be specified, and if an instruction is executed that writes into the upper and lower byte of the CSR, ROM address 10_8 should be specified.

DECIMAL LOC	OCTAL LOC	BINARY DATA	OCTAL DATA
0	00	11101111	357
1	01	11111111	377
2	02	11111111	377
3	03	11111111	377
4	04	11111111	377
5	05	11111111	377
6	06	11111111	377
7	07	11111111	377
8	10	11001111	317
9	11	01111111	177
10	12	11111111	377
11	13	11111111	377
12	14	10111111	277
13	15	11111111	377
14	16	11111111	377
15	17	11111111	377
16	20	11011111	337
17	21	11111111	377
18	22	11111111	377
19	23	11111111	377
20	24	11111111	377
21	25	11111111	377
22	26	11111111	377
23	27	11111111	377
24	30	11110111	367
25	31	11110110	366
26	32	11110101	365
27	33	11110100	364
28	34	11110011	363
29	35	11110010	362
30	36	11110001	361
31	37	11110000	360

Figure 2-18 ROM Address Map

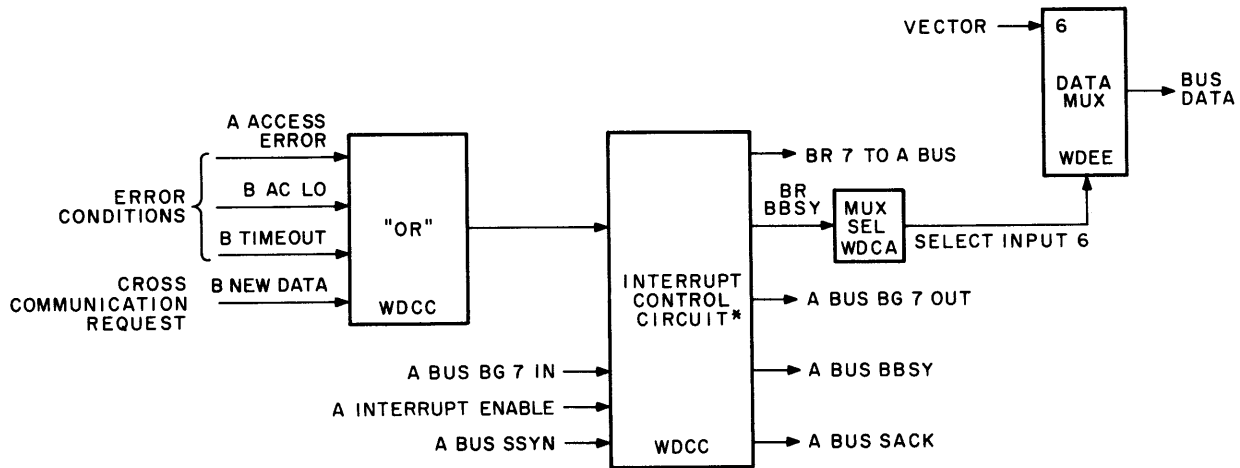
2.9 INTERRUPT CONTROL LOGIC

Figure 2-19 is a simplified block diagram of the interrupt control logic that is contained on the bus control modules. The DA11-F can be set to interrupt on either an error condition or a cross-communication request. The interrupt is raised on BR7, which is the highest priority level. The error conditions that can cause an interrupt are access errors (i.e., the originating master was unable to get through the window) or a power-fail sequence on the target bus. The cross-communications interrupt is used to pass parameters between computers. If the interrupt enable (IE) bit in the CSR is set and an interrupt request is asserted (either ERROR or NEW DATA), the interrupt control logic asserts BUS REQUEST. This signal is routed through the priority jumper plug on the control module to the BR7 signal line. When the bus priority arbitration logic in the processor recognizes the request, it issues the corresponding bus grant signal BG7. The interrupt control acknowledges receipt of BG7 by asserting SACK. After the previous bus master completes its operations and releases the bus, the interrupt control clears SACK and asserts BBSY and INTR to notify the processor that an interrupt operation is in progress. At the same time, the output of the BR BBSY flip-flop forces the data multiplexer select logic MUX SEL to gate the vector address from input 6 of the multiplexer

to the bus data lines. After the processor has read the interrupt vector address, it returns SSYN to the interrupt control. The interrupt control then clears INTR, BBSY, and the data lines and resets itself in preparation for the next interrupt request.

The chart below defines the signals described above for both the A bus and B bus and also references the applicable logic print.

Signal Name	A Bus	B Bus
BR7	BUS ABR7 (WDCC)	BUS BBR7 L (WDCJ)
ERROR	WDCC AACC ERROR (1) H	WDCJ BACC ERROR (1) H
NEW DATA	WDCC ANEW DATA (1) H	WDCJ BNEW DATA (1) H
BUS REQUEST	WDCC ABUS REQUEST L	WDCJ BBUS REQUEST L
BG7	BUS BG7 IN H (WDCC)	BUS BG7 IN H (WDCJ)
SACK	BUS ASACK L (WDCC)	BUS BSACK L (WDCJ)
BBSY	BUS ABSY L (WDCC)	BUS BBSY L (WDCJ)
INTR	BUS AINTR L (WDCC)	BUS BINTR L (WDCJ)
BR BBSY Flip-Flop	WDCC BR ABSY (1) H	WDCJ BR BBSY (1) H
MUX SEL	WDCA MUX SEL 2, 1, 0, L	WDCF MUX SEL 2, 1, 0, L
SSYN	WDCA ASSYN L	WDCF BSSYN L



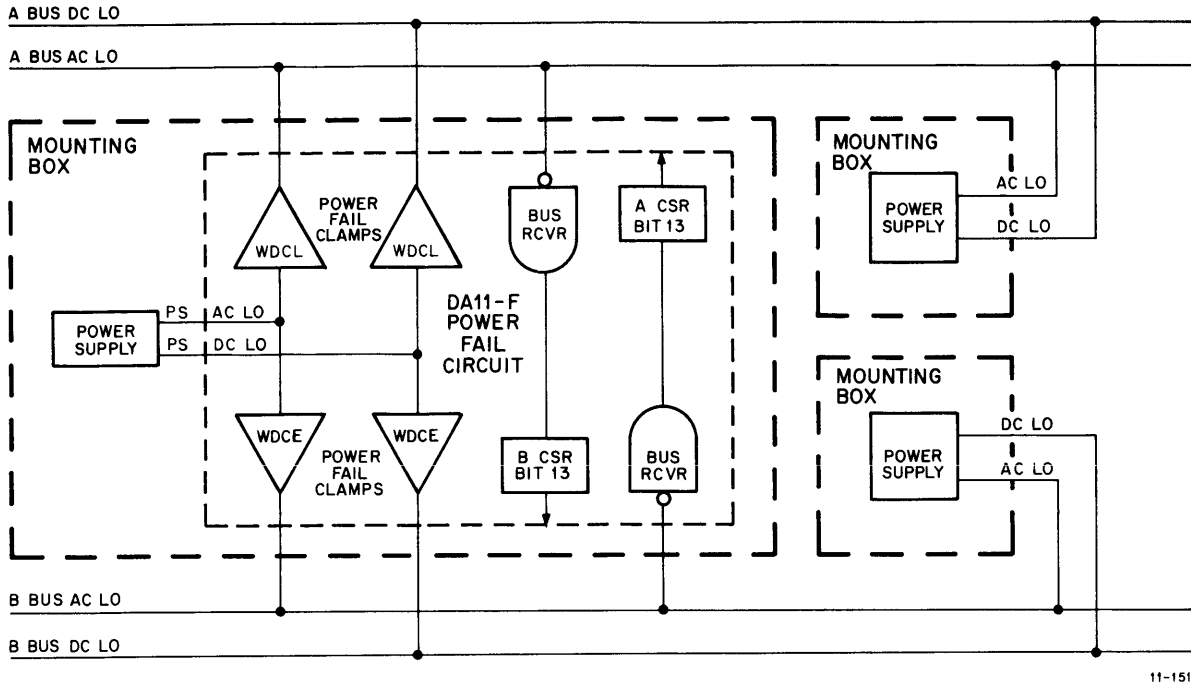
* THE INTERRUPT CONTROL CIRCUIT IS SIMILAR TO THE M7821 INTERRUPT CONTROL MODULE

11-1510

Figure 2-19 Interrupt Control for a Port

2.10 POWER FAIL

Figure 2-20 represents a simplified diagram of the power-fail circuitry in the DA11-F. Two power-fail situations are handled by the DA11-F: first, a powerdown by the DA11-F's mounting box power supply, and second, a powerdown by a power supply in some external mounting box.



11-1511

Figure 2-20 Power Fail Simplified Block Diagram

If powerdown occurs in the supply that furnishes dc power to the window, the power-fail clamp circuits assert AC LO and DC LO on both buses, thereby initiating power shutdown sequences in both computers.

If powerdown in some external power supply occurs, that device asserts AC LO on the bus. The bus receiver connected to this bus senses the powerdown and reflects it in bit 13 of the CSR on the opposite bus. For example, if a power supply connected to bus A should power down, the bus receiver senses this and reflects it in bit 13 of the CSR connected to the B bus. An interrupt would be generated on the B bus if its interrupt enable (IE) were set. Furthermore, if the B bus processor tried to access the window while AC LO is asserted on bus A, an access error would occur since bus A is in the power-fail mode. This condition could be verified through examination of bit 13 of the CSR on the B bus.

The chart below defines the signals described above for both the A bus and B bus and also references the applicable logic print.

Signal Name	A Bus	B Bus
PS AC LO L	WDCE PS AC LO L	WDCL PS AC LO L
PS DC LO L	WDCE PS DC LO L	WDCL PS DC LO L
BUS AC LO L	BUS AAC LO L (WDCE)	BUS BAC LO L (WDCL)
BUS DC LO L	BUS ADC LO L (WDCE)	BUS BDC LO L (WDCL)
CSR 13	WDCE BAC HI L	WDCL AAC HI L

CHAPTER 3 PROGRAMMING INFORMATION

3.1 GENERAL

Each port of the DA11-F contains seven addressable registers that are under Read/Write control of their respective buses. The registers are assigned addresses in I/O address space as shown in Table 3-1. Note that all addresses in this chapter are in octal notation.

**Table 3-1
Register Addresses**

Register	Mnemonic		Program Operation	Address
	A Port	B Port		
Control and Status	ACSR	BCSR	Read/Write	XXXX00
Output Data Buffer	ADB	BDB	Read/Write	XXXX02
Input Data Buffer	BDB	ADB	Read only	XXXX04
Displacement Address	ADA	BDA	Read only	XXXX06
Relocation Address	ARA	BRA	Read/Write	XXXX10
Starting Address	ASA	BSA	Read only	XXXX12
Vector Address	AVA	BVA	Read only	XXXX14

The Control and Status Register can be assigned to any address in the I/O register page (760000 to 777777) on an even 20_8 boundary. The addresses of the other registers are assigned sequentially after the CSR. For example, if the CSR is assigned to 764000, the addressable registers are assigned the following addresses:

Control and Status	764000
Output Data Buffer	764002
Input Data Buffer	764004
Displacement Address	764006
Relocation Address	764010
Starting Address	764012
Vector Address	764014

These registers are *normally* assigned to the user address field from 764000 to 764776.

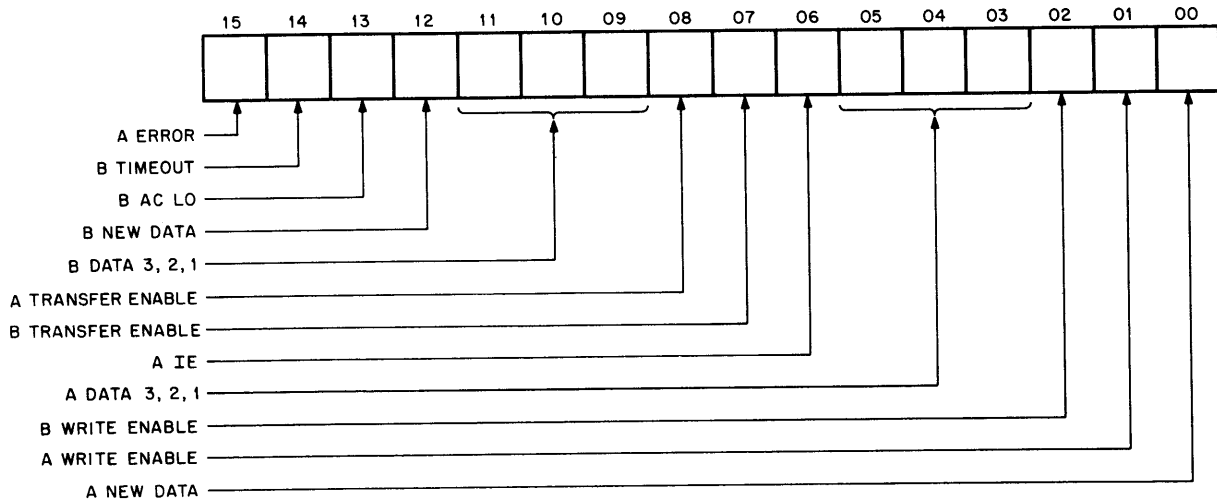
The DA11-F is normally to the user address field as follows:

	A Port	B Port
First Unit	764000	764020
Second Unit	764020	764040
Third Unit	764040	764060
Fourth Unit	764060	764000

Each of the addressable I/O registers is described in detail in the following paragraphs.

3.2 CONTROL AND STATUS REGISTER (CSR)

Each port contains a 16-bit Control and Status Register (CSR). The CSRs are interconnected to provide a means of transferring parameters and interrupt requests from one processor to another. Figure 3-1 illustrates the bit format of the CSR. Table 3-2 defines the interaction between each of the CSRs. For example, bits 11, 10, and 9 of the ACSR are used to read the contents of bits 5, 4, and 3 of the BCSR. The description in this chapter covers the ACSR; however, by reversing the A and B prefixes of each bit, the description is also applicable to the BCSR. Each bit is described in the following paragraphs. The program can read the contents of all bits but can only write into the bits so designated.



NOTE:
 Bits are designated for A CSR. Reverse A and B prefixes to find equivalent B CSR designations.

11-1512

Figure 3-1 ACSR Bit Format

**Table 3-2
Control and Status Registers WCSR 764XXX**

Bit	ACSR	BCSR	Program Action
15*	A Error	B Error	Read/Write 0**
14	B Time Out	A Time Out	Read Only
13	B ACLO	A ACLO	Read Only
12*	B New Data	A New Data	Read/Write 0**
11	B Data 3	A Data 3	Read Only
10	B Data 2	A Data 2	Read Only
9	B Data 1	A Data 1	Read Only
8	A Transfer Enable	B Transfer Enable	Read/Write
7	B Transfer Enable	A Transfer Enable	Read Only
6	A IE	B IE	Read/Write
5	A Data 3	B Data 3	Read/Write
4	A Data 2	B Data 2	Read/Write
3	A Data 1	B Data 1	Read/Write
2	B Write Enable	A Write Enable	Read/Write
1	A Write Enable	B Write Enable	Read/Write
0	A New Data	B New Data	Read/Write 1***

*If bit 15 or bit 12 is set and the interrupt enable bit (bit 6) is set, an interrupt will occur.

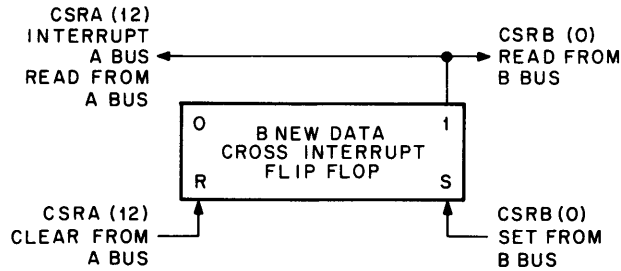
**The program can read the contents of all bits but can only write a 0 into these bit positions.

***The program can read the contents of all bits but can only write a 1 into this bit position.

All bits are cleared by INIT.

Bit	Function
15	<p>Error. Set on this port if an originating device on this bus cannot make a legal access to the opposite bus. The following conditions cause this bit to set:</p> <ol style="list-style-type: none"> 1. Originator attempts a window transaction and the Transfer Enable of the opposite bus is not set. To clear, write a 0 into bit 15. 2. Originator attempts a DATO(B) or a DATIP-DATO(B) cycle and the Write Enable bit of the opposite bus is not set. To clear, write a 0 into bit 15. 3. SSYN Time Out occurs while the window is master of the opposite bus. (Also indicated by bit 14.) To clear, write a 0 into bit 15. 4. ACLO is asserted on the opposite bus. (Also indicated by bit 13.) To clear, restore power to opposite bus. 5. Originators on both buses attempted DATI(P) cycles simultaneously. To clear, write a 0 into bit 15. When the error bit is set, further window transactions originating on this bus are inhibited. An attempt to use the window will result in SSYN Time Out. If the Interrupt Enable bit (bit 6) is set when the Error bit sets, an interrupt will occur.

Bit	Function
14	Time Out. Set on this port if SSYN Time Out occurs while the window is master of the opposite bus. When set, sets Error (bit 15). Clear by writing 0 into bit 15.
13	ACLO. Set on this port if ACLO is asserted on the opposite bus. When set, sets Error (bit 15). Clear by restoring power to opposite bus.
12	New Data. Read out of contents of CSR (0) on the opposite port. Used by the opposite bus processor to indicate that it has loaded new information into its CSR or its Data Buffer (DB). The cross-interrupt bit (New Data) is read/set as bit 0 on one port and read/clear as bit 12 on the other port. If Interrupt Enable (bit 6) is set, then setting New Data on the opposite port causes an interrupt on this bus. For example, if bit 0 (B New Data) is set on the B port, a 1 appears at bit 12 of the A port. If Interrupt Enable (bit 6) on the A port is set, an interrupt request will be generated on the A bus. When the A processor has serviced the interrupt, it clears B New Data by writing a 0 into bit 12 of its own CSR. This action clears bit 0 of the B port CSR, thereby indicating that the message has been received. The cross interrupt logic for B New Data is illustrated in Figure 3-2.
(11:9)	Data (3:1). Read out of contents of CSR (5:3) on the opposite bus. Used to pass parameters between programs. Defined by program.
8	Transfer Enable. Set by program on this bus to indicate that this port is ready to be used as a target for transactions originating on the opposite bus. Setting Transfer Enable implies that the Relocation Address Register on this port is loaded with the correct relocation factor and that CSR1 Write Enable is in the correct state.
7	Transfer Enable. Read out of contents of CSR(8) on the opposite bus. When set, indicates that an originator on this bus may perform a window transaction to the opposite bus. If clear, an attempt to originate a window transaction from this bus will set Error on this port.
6	IE – Interrupt Enable. When set by program on this bus, allows interrupts to occur when CSR (15 or 12) of this port are set.
(5:3)	Data (3:1). Loaded by program on this bus to pass parameters to program on opposite bus. Defined by program.
2	Write Enable. Read out of contents of CSR(1) on the opposite bus. When set, indicates that an originator on this bus may do a DATO(B) cycle to a location on the opposite bus. If clear, an attempt to originate a DATO(B) cycle from this bus will set Error on this port.
1	Write Enable. Set by program on this bus to indicate that the program on the opposite bus may do a DATO(B) cycle to a location on this bus.
0	New Data. Set by program on this bus to indicate that either the CSR or the Data Buffer (DB) on this port is loaded with new information. Cleared by writing 0 into CSR(12) on the opposite port.



11-1513

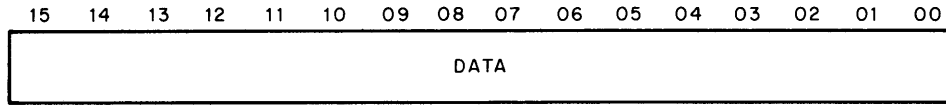
Figure 3-2 Cross Interrupt Logic

3.3 OUTPUT DATA BUFFER (DB)

The Output Data Buffer operates (Figure 3-3) in two modes: either as temporary data storage during window transactions or as a cross communications register during interprocessor program transfers.

If Transfer Enable is set and this port is on the originator bus, the Data Buffer temporarily stores the data during DATO(B) cycles. In this mode, the Data Buffer is read only by the program.

If Transfer Enable is clear, the Data Buffer (DB) can be used by the program to pass information to the opposite processor. In this mode, the Data Buffer is Read/Write from this bus and read only from the opposite bus. The Data Buffer is cleared by INIT.



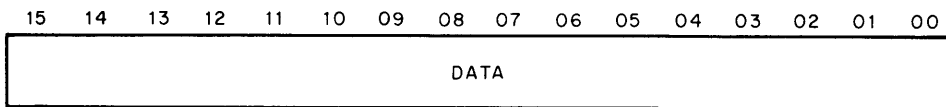
11-1514

Figure 3-3 Output Data Buffer Bit Format

3.4 INPUT DATA BUFFER (DB)

The Input Data Buffer (Figure 3-4) is a read-only access to the contents of the Data Buffer on the opposite bus.

When Transfer Enable is clear, the Input Data Buffer can be used by the program to receive information from the opposite processor.



11-1515

Figure 3-4 Input Data Buffer Bit Format

3.5 DISPLACEMENT ADDRESS REGISTER (DAR)

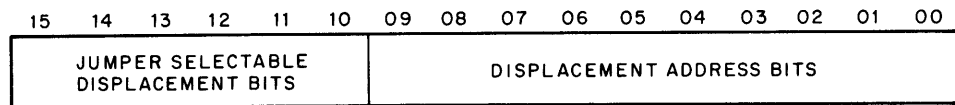
The Displacement Address Register (Figure 3-5) serves as temporary storage for the window displacement address when this port is on the originator bus.

When the originator window address decoder detects a match in bits (17:N), the low-order address bits (N-1:0) are stored in the DA Register and then used in conjunction with the Relocation Address Register on the opposite bus to form the relocated target address.

The window decoder and the Displacement Address Register vary in length according to the size of the window field:

Window Field (Words)	Window Decoder	DAR
1/2K	17:10	09:00
1K	17:11	10:00
2K	17:12	11:00
4K	17:13	12:00
8K	17:14	13:00
16K	17:15	14:00
32K	17:16	15:00

If an access error occurs, further transactions from originating devices on this bus are inhibited and the DAR then contains the address of the illegal access. The recovery program can inspect the DAR to determine the relative address that failed. The DAR is written into automatically by originating window transactions. The program can read the contents of the DAR but cannot write into it. The DAR is cleared by INIT.



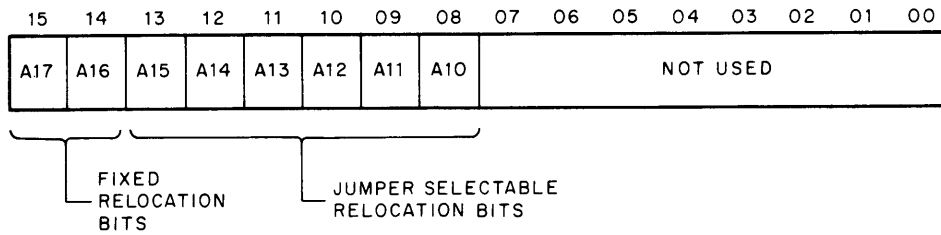
11- 1516

Figure 3-5 Displacement Address Register Bit Format

3.6 RELOCATION ADDRESS REGISTER (RAR)

The Relocation Address Register (Figure 3-6) specifies the high-order bits (17:N) of the relocated target address.

The RAR is loaded under program control and specifies bits (17:N) of the relocated target address when this port is master of the target bus. The RAR is used in conjunction with the Displacement Address Register on the opposite bus to form the complete target address.



11-1517

Figure 3-6 Relocation Address Register Bit Format

The length of the RAR is jumper selectable and is set to the same length as the window decoder on the opposite bus. The window address field may be set to 1/2K, 1K, 2K, 4K, 8K, 16K, or 32K words. The window field and the RAR are related as follows:

Window Field (Words)	Window Decoder	RAR
1/2K	17:10	15:8
1K	17:11	15:9
2K	17:12	15:10
4K	17:13	15:11
8K	17:14	15:12
16K	17:15	15:13
32K	17:16	15:14

Note that the data bits in the RAR are shifted two places from their corresponding address bits. That is:

RAR	Specifies	Target Bus Address
15		17
14		16
13		15
12		14
11		13
10		12
9		11
8		10

The RAR is Read/Write for bits (17:N). Bits (N-1:0) are always read as 0. Writing has no effect on (N-1:0). The RAR is cleared by INIT.

3.7 STARTING ADDRESS REGISTER (SAR)

The Starting Address Register (SAR) is a read-only register containing the address bits of the window decoder. These bits are used to determine the window field starting address (Figure 3-7).

At installation, the window field starting address is established by cutting a set of jumpers on the data paths modules. The pattern cut in the jumpers is a binary representation of the address of the lower boundary of the window field on the originator port. The SAR provides the program with the capability to read the jumper pattern and verify that it is set to the correct address. Its main use is in maintenance diagnostics.

Note that SAR (7:0) are not used and are always read as 0s.

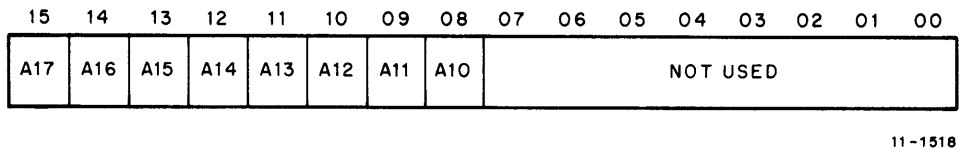


Figure 3-7 Starting Address Register Bit Format

3.8 VECTOR ADDRESS REGISTER (VAR)

The Vector Address Register (VAR) is a read-only register containing the vector address. This address is placed on the Unibus by the DA11-F during an interrupt cycle (Figure 3-8).

At installation, the vector address is established by cutting a set of jumpers on the data paths modules. The pattern cut in the jumpers is a binary representation of the vector address used by the DA11-F to interrupt the processor. The VAR provides the program with the capability to read the jumper pattern and verify that it is set to the correct address. Its main use is in maintenance diagnostics.

Note that VAR (15:10) are not used and are always read as 0s. Also, VAR (1:0) are always read as 0s since vector addresses occur only on even 4_8 boundaries.

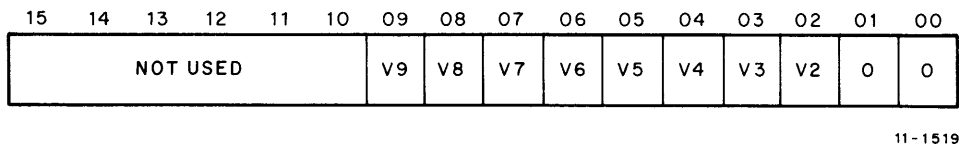


Figure 3-8 Vector Address Register Bit Format

3.9 PROGRAMMING EXAMPLES

This section describes two programs using the DA11-F Unibus window. The first program illustrates how the window is opened into the A bus, and the second illustrates a block transfer through the window.

3.9.1 Open-The-Window Program

The programming example below (Figure 3-9) illustrates a method to open the window into the A bus. By using this program, the lowest portion of core memory on the A bus can be deposited and/or examined from the operator's console on the B processor. Depressing the START switch will clear the A port Relocation Address Register, thereby setting the starting address on the target bus (A bus) at location 0.

```

; PROGRAM TO OPEN THE WINDOW FROM THE A SIDE,
; THE PROGRAM ENABLES THE BOTTOM OF CORE ON THE A SIDE
; TO BE EXAMINED OR DEPOSITED FROM THE B SIDE CONSOLE,
; WHEN THE START KEY IS DEPRESSED THE RELOCATION
; ADDRESS IS SET TO 0 RELOCATING THE WINDOW
; TO THE BOTTOM OF CORE

000006          R6=      X6
164000          WCSRA=   164000          ;ADDRESS OF CONTROL STATUS REGISTER A

000004          .=4
003000          3000          ;POINTER IN CASE WINDOW DOES NOT RESPOND AT 164000
;BUS ERROR TRAP NEW PROCESSOR STATUS

000006          0
001000          .=1000

001000  012706  000600      MOV      #600,R6          ;SET UP STACK IN CASE OF BUS ERROR
001004  012737  000402  164000  MOV      #402,0#WCSRA      ;SET WRITE ENABLE AND TRANSFER ENABLE ON THE A SIDE
001012  000001          WAIT          ;STOP HERE WITH COMPUTER A RUNNING,
;NOW COMPUTER B CAN ACCESS MEMORY THROUGH
;THE WINDOW.

003000          .=3000
003000          000000          HALT          ;ERROR THE WINDOW DID NOT RESPOND
;CSRA MUST BE AT SOME OTHER ADDRESS

000001          .END

```

Figure 3-9 Open-The-Window Program

3.9.2 Block Transfer Program

This program (Figure 3-10) illustrates a method whereby a block of data containing 400₈ words is transferred through the window and checked. A similar program can be run in the other computer at the same time; however, in order to accomplish this the address of the second group of addressable registers must be changed. This can be accomplished by changing the address of the CSR.

This example program is set up to run with a 4K window size. The target address is located at the lowest portion of core (00) since the Relocation Address Register is cleared when the START switch is depressed. The originator field is assumed to be from 4K to 8K. Register R1 contains the window starting address and register R0 contains the 2's complement of the number of words to be transferred. (If this program is run in a processor whose window field is at some range other than 4K to 8K, the number that is loaded into R1 must be modified so that R1 corresponds to the new starting address.) Register R5 is used to keep track of the number of bus errors that occur throughout the program.

3.10 ERROR REPORTING PROCEDURES

The window channel is similar to a memory location from the point of view of the master device on the originator bus. That is, the master addresses the window field as if it were memory and expects it to respond in the same way that actual memory does. In fact, the actual memory location is on the target bus and the window is simply the intermediary. As long as the channel is used correctly, the window will give the proper responses, as described in Paragraph 2.2. However, if an illegal access is made, the window's response will be a function of the particular type of bus cycle used during that access.

```

,ARS
,TITLE WINDOW BLOCK
R0# X0
R1# X1
R2# X2
R5# X5
R6# X6
WCSRA# 164000 ADDRESS OF THE STATUS REGISTER A
FIRST# 15200 FIRST TIME SWITCH

R00004 R00004 ,#4
R15000 15000

R00006 R00000 *

003000 ,#3000
003000 012737 000000 15200 START: MOV #0,##FIRST
003006 012735 177777 MOV #*1,R5
003012 012706 000000 BLOCK: MOV #000,R6
003016 012700 177400 MOV #177400,R0
003022 012701 021000 MOV #21000,R1
003026 012702 001000 MOV #1000,R2
003032 005767 012142 TST FIRST
003036 001004 BNE TRY
003040 005267 012134 INC FIRST
003044 000167 011730 JMP OPEN

003050 011211 TRY: MOV (R2),(R1)
003052 022221 CMP (R2)+,(R1)+
003054 001004 BNE ERR
003056 005200 INC R0
003060 001403 REQ REST
003062 000167 177762 JMP TRY
003066 000000 ERR: HALT
003070 000167 177716 REST: JMP BLOCK

R15000 ,# 15000

R15000 012737 000402 164000 OPEN: MOV #402,##WCSRA
R15006 022737 000606 164000 TEN: CMP #606,##WCSRA
R15014 001374 BNE TEN
R15016 005205 INC R5
R15020 012706 000600 MOV #600,R6
R15024 000167 166020 JMP TRY
000001 ,END

```

Figure 3-10 Block Transfer Program

When a slave location on the Unibus is addressed by a master device, it responds by asserting SSSYN to indicate that it has correctly performed the operation specified by the bus control signals C1 and C0. During a DATO(B) cycle, SSSYN means that the slave has received the data word from the master. Likewise, during a DATI(P) cycle, SSSYN means that the slave has asserted the correct data word on the bus. If the window detects an illegal access (e.g., if the DA11-F is prevented from completing the cycle on the target bus), the window field becomes in effect non-existent memory. The window's response is based on the following rules:

1. Access errors will set ERROR CSR 15, indicating a failure has occurred.
2. If the cycle is a DATO(B), the DA11-F itself can return SSSYN since the window Data Buffer Register is used as a temporary storage location for the data.
3. If the cycle is a DATI(P), SSSYN is not returned by the DA11-F since it is never able to retrieve the correct data from the target location.
4. If ERROR has been set previously, the DA11-F will not respond in any way to a subsequent window access. This ensures that the contents of the Displacement Address Register is not modified.

The exact error reporting procedures are listed in Table 3-3.

**Table 3-3
Error Reporting Procedures After Illegal Window Channel Access**

Conditions Causing Access Errors	Error Reporting Procedure					
	DATI		DATIP		DATO(B)	
	CSR	Return SSYN	CSR	Return SSYN	CSR	Return SSYN
1. ERROR CSR 15 set previously. (May have been set by Access Error, Timeout on target bus, or ACLO on target bus.)	No Change	No	No Change	No	No Change	No
2. Window channel not open (Transfer Enable CSR bit 7=0)	Set CSR 15	No	Set CSR 15	No	Set CSR 15	Yes
3. Write operations disabled (Write Enable CSR bit 2=0)	N/A	N/A	Set CSR 15	No	Set CSR 15	Yes
4. Timeout occurs on target bus during this bus cycle. (DA11-F addressed a non-existent location on the target bus.)	Set CSR 15, CSR 14	No	Set CSR 15, CSR 14	No	Set CSR 15, CSR 14	Yes
5. Originating masters on both buses attempt DATI(P) cycles through the window simultaneously.	Set CSR 15	No	Set CSR 15	No	N/A	N/A

In the situations when the DA11-F does not return SSYN, the originating master will time out as if it had addressed non-existent memory. If the master is the processor, the timeout will cause a trap to location 4. However, if the master is an NPR device, the device will set its Non-Existent Memory flag (and interrupt if so enabled). In all cases, the DA11-F Error flag is set and the window will interrupt if enabled.

The length of time that is required for the DA11-F to gain control of the target bus is a function of the configuration of the system and any other NPR activity on that bus. The window should generally be located as the first device on the NPR priority level in order to minimize the delay of “through-the-window” transactions. Even in an optimum configuration, the originating master might not receive SSYN for as much as five microseconds during a DATI(P) cycle. Therefore, the time-out timers of all devices that use the window channel should be set to exceed the worst case latency of valid window operations. This will prevent false Non-Existent Memory errors. A minimum of twenty microseconds is recommended in dual processor systems. Multiple processor installations that operate windows in cascade may require longer settings of their time-out timers.

CHAPTER 4 MAINTENANCE

4.1 GENERAL

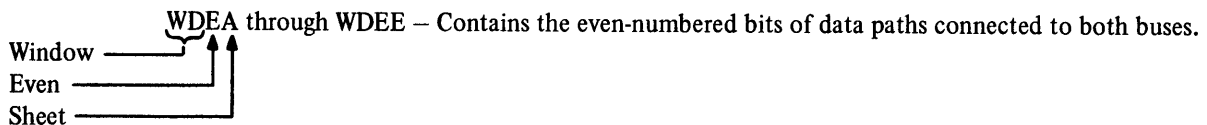
This chapter contains installation and maintenance information concerning the DA11-F. The installation information is described along with an example of a typical customer configuration which might be ordered.

This chapter presents the user with the necessary maintenance information to understand normal system operation. The user can utilize this information when analyzing trouble symptoms to determine necessary corrective action. It is beyond the scope of this manual to present detailed troubleshooting information.

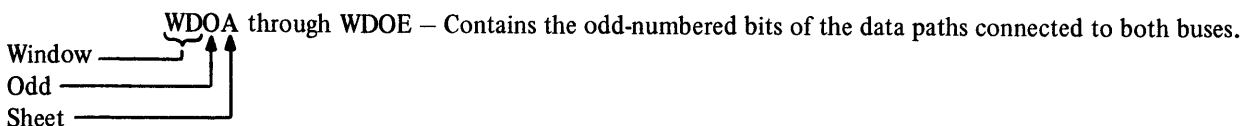
4.2 ENGINEERING DRAWINGS

A complete set of engineering drawings and circuit schematics is provided in a supplementary manual entitled *DA11-F Unibus Window Engineering Drawings*. The general logic symbols used on these drawings are described in *DEC Logic Handbook, 1971*. Specific symbols and conventions are also included in the *PDP-11 Conventions Manual*, DEC-11-HR6B-D and in certain PDP-11 system manuals.

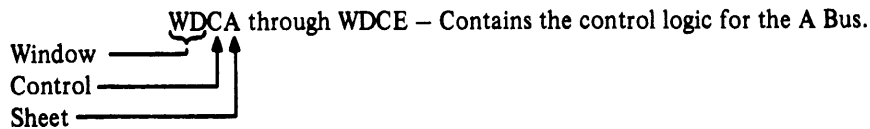
The following paragraph lists the logic prints associated with the DA11-F.



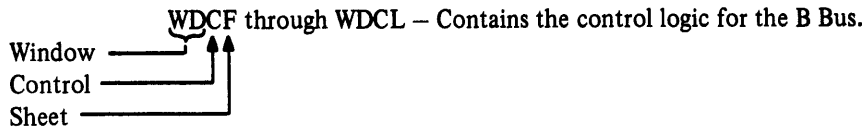
- WDEA – Window Even Registers – A Bus
- WDEB – Window Even Registers – B Bus
- WDEC – Window Even Receivers and Transmitters – A Bus and B Bus
- WDED – Window Even Multiplexers – A Bus
- WDEE – Window Even Multiplexers – B Bus



- WDOA – Window Odd Registers – B Bus
- WDOB – Window Odd Registers – A Bus
- WDOC – Window Odd Receivers and Transmitters – A Bus and B Bus
- WDOD – Window Odd Multiplexers – B Bus
- WDOE – Window Odd Multiplexers – A Bus



- WDCA – Register Control – A Bus
- WDCB – Access Control – A Bus
- WDCC – Interrupt Control – A Bus
- WDCD – NPR Control – A Bus
- WDCE – Power Fail Control – B Bus



- WDCF – Register Control – B Bus
- WDCH – Access Control – B Bus
- WDCJ – Interrupt Control – B Bus
- WDCK – NPR Control – B Bus
- WDCL – Power Fail Control – A Bus

The following paragraphs describe the signal nomenclature conventions used on the drawing set.

Signal names in the DA11-F print set are in the following basic form:

SOURCE	SIGNAL NAME	POLARITY
--------	-------------	----------

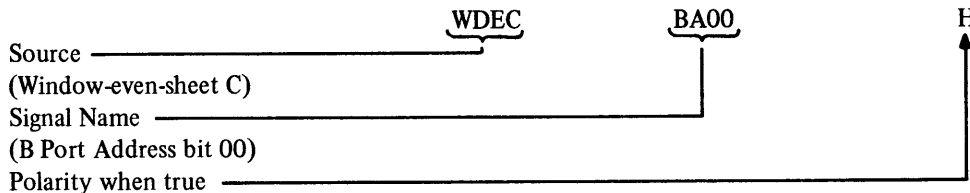
SOURCE indicates the drawing number of the print set where the signal originates. The drawing designation of a print is located in the lower right-hand corner of the print title block (WDOE, WDCC, WDEA, etc.).

SIGNAL NAME is the proper name of the signal. The names used on the print set are also used in this manual for correlation between the two.

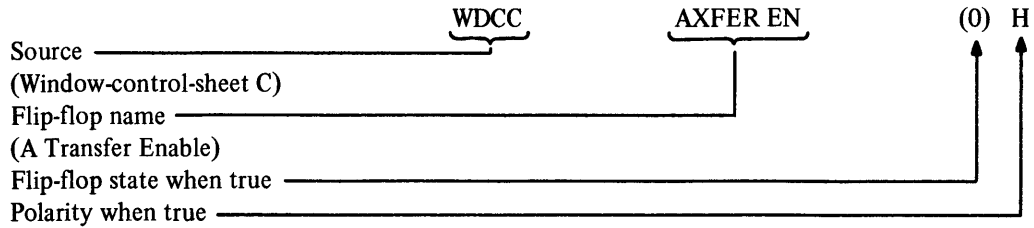
POLARITY is either H or L to indicate the voltage level of the signal: H means +3V; L means ground.

Three examples of the signal conventions are shown below. Note in example 3 that the source of bus signals is designated as BUS.

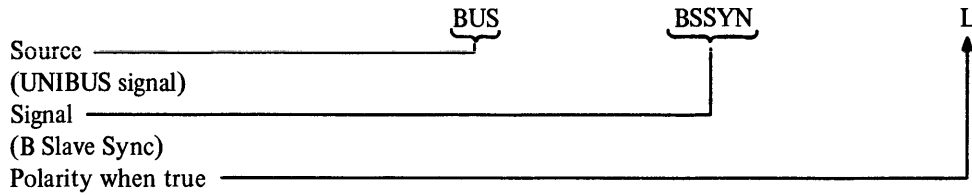
Example 1



Example 2



Example 3



4.3 INSTALLATION

The first step in designing a DA11-F Bus Window configuration for a particular multiple-processor system is to match the parameters of the bus window to the selected systems. The window parameters are determined by the customer and are specified when the option is ordered. The bus window configuration is then implemented in accordance with the information contained in the following paragraphs.

4.3.1 Customer Specifications

The customer should first reproduce and fill out a DA11-F Configuration Chart (Figure 4-1). The chart provides all the necessary information to configure both ports of a DA11-F and, when completed, should be forwarded to DEC with the option purchase order. If a system contains more than one DA11-F Bus Window, a separate chart should be filled out for each one. The following steps outline the procedure for completing the DA11-F Configuration Chart. Figure 4-2 is an example of a completed DA11-F Configuration Chart.

Step	Procedure
1	Circle the window field size selected on line A. Note that the size of the window field is the same on both ports of the DA11-F.
2	Indicate on the Bus Address Maps (line B) the lower and upper boundaries of memory and the size of the window field on each Unibus. The window field must not overlap the I/O Register field (124K to 128K) and must not overlap the vector addresses starting at location 0.

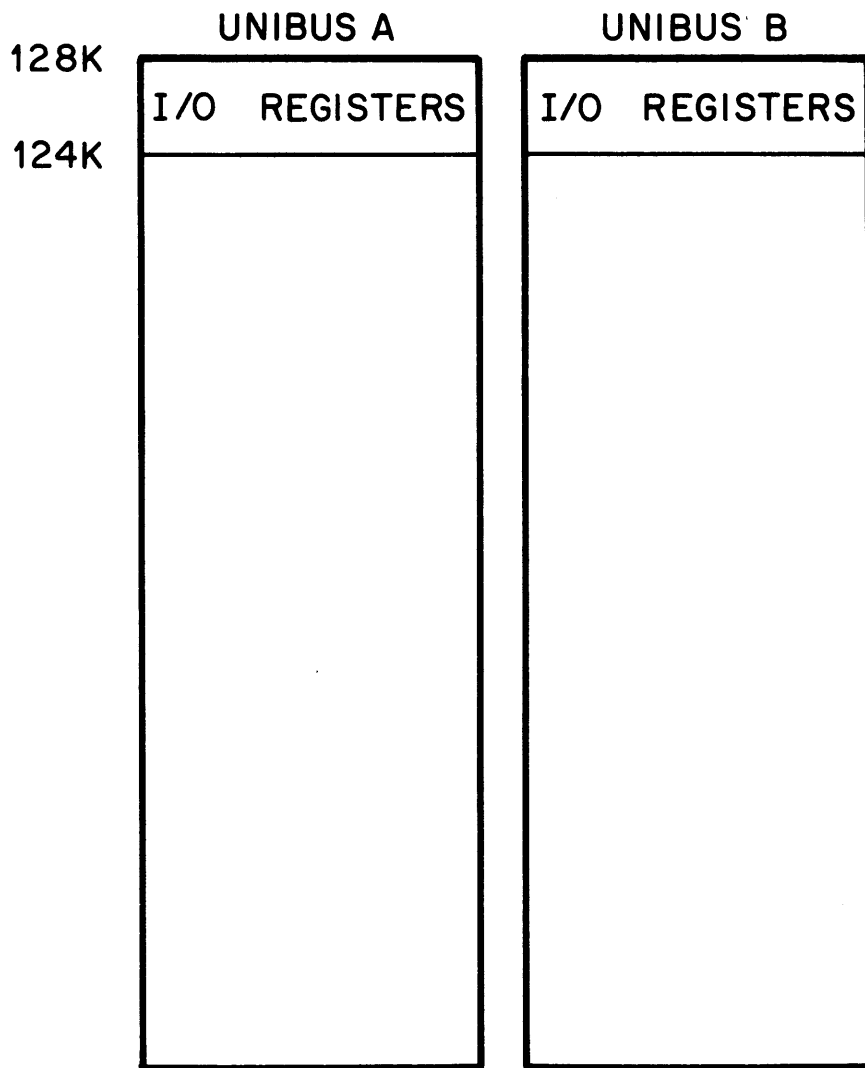
NOTE

It is recommended that the window field be located immediately above all memory on the Unibus. This configuration produces continuous addresses throughout both the physical memory and the window field and does not create any unused memory locations between the physical memory addresses when the window is closed.

A. WINDOW FIELD SIZE (words) – circle one

512 1K 2K 4K 8K 16K 32K

B. BUS ADDRESS MAPS (fill in lower and upper boundaries of memory and window fields)



11-1546

C. WINDOW START ADDRESS

D. DA11-F CSR ADDRESS

E. DA11-F VECTOR ADDRESS

F. TYPE OF PROCESSOR

PDP-11/

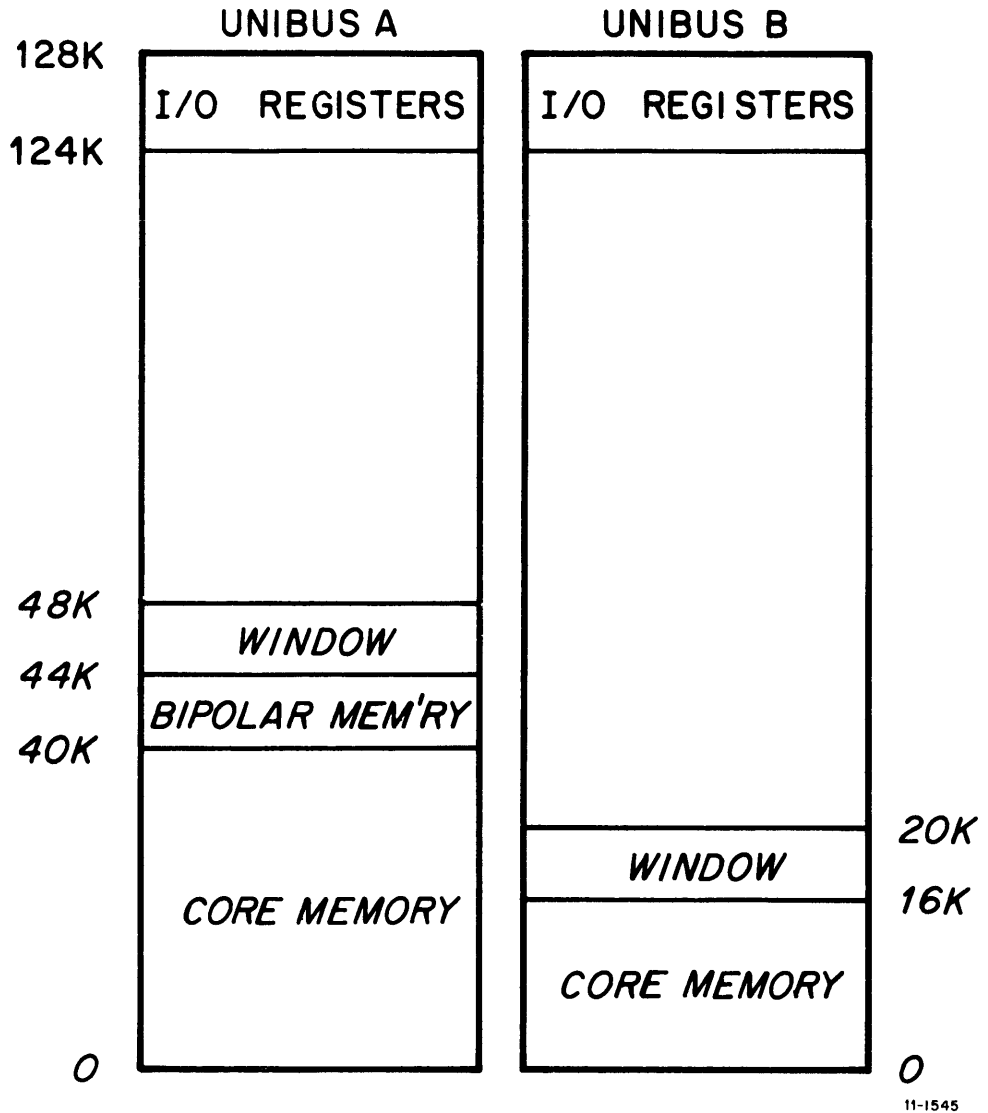
PDP-11/

Figure 4-1 DA11-F Configuration Chart

A. WINDOW FIELD SIZE (words) – circle one

512 1K 2K **4K** 8K 16K 32K

B. BUS ADDRESS MAPS (fill in lower and upper boundaries of memory and window fields)



C. WINDOW START ADDRESS	<u>44K</u>	<u>16K</u>
D. DA11-F CSR ADDRESS	<u>764000</u>	<u>764020</u>
E. DA11-F VECTOR ADDRESS	<u>320</u>	<u>374</u>
F. TYPE OF PROCESSOR	<u>PDP-11/45</u>	<u>PDP-11/10</u>

11-1545

Figure 4-2 DA11-F Configuration Chart Example

- | Step | Procedure |
|------|---|
| 3 | On line C, enter the window starting address for each bus. This address corresponds to the lowest boundary of the window field as specified in Step 2. |
| 4 | On line D, enter the address of the Control and Status Register. This address determines the location of all DA11-F addressable registers (Paragraph 3.1). The CSR addresses are selected as follows: |

DA11-F Unit No.	Unibus A	Unibus B
1	764000	764020
2	764020	764040
3	764040	764060
4	764060	764000

- | | |
|---|---|
| 5 | On line E, enter the DA11-F vector address for each port. Select the vector from the Floating Vector Field (location 300 and above), depending on the requirements of the rest of the system. The DA11-F vector address is assigned after all other device vectors have been assigned. |
| 6 | Enter the type of PDP-11 processor that is connected to each port of the DA11-F. Figure 4-2 shows an example of a completed DA11-F Configuration Chart. This example should be kept in mind as it is used in subsequent paragraphs to illustrate how a selected configuration is implemented. |

4.3.2 Implementing Customer Specifications

The DA11-F consists of four quad-height modules: two M7284 Bus Control and two M7283 Data Path Modules. Each module contains jumper wires in specified patterns so as to determine the addresses to which the various address decoders will respond. The information on the DA11-F Configuration Chart is implemented by setting the binary equivalent of the addresses in the jumper wires. All modules initially contain all jumper wires. A jumper wire that has not been deleted represents a binary 0, and a jumper wire that has been deleted (cut) represents a binary 1.

NOTE

As manufactured, the M7283 Data Path Modules are physically identical and the M7284 Bus Control Modules are physically identical. However, once the addresses are established by cutting the jumper wires, the modules are no longer interchangeable and must be inserted in their corresponding system unit slots.

Each module must first be marked with a tag that clearly indicates its function.

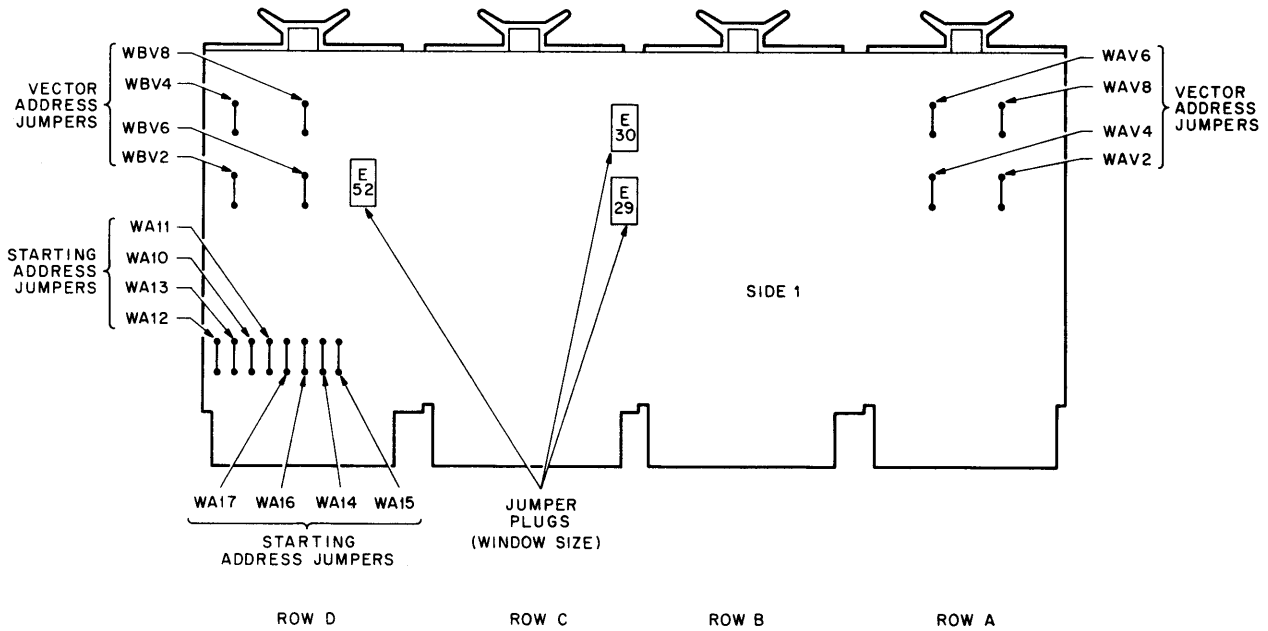
Module	Function	Tag	System Unit Slot
First M7284	A Bus Control	A	1
Second M7284	B Bus Control	B	4
First M7283	Odd Data Paths	Odd	2
Second M7283	Even Data Paths	Even	3

The following paragraphs describe the various jumper configurations and refer to the modules by their functional or tag designations rather than by the module type.

4.3.3 Starting Address Jumpers

As previously mentioned, the starting address is the address of the lower boundary of the window field. The starting address jumpers for the A bus are contained on the odd data path module and the starting address jumpers for the B bus are contained on the even data path module. Figure 4-3 shows the approximate location of the starting address jumpers for both data path modules. To cut the correct pattern into the jumpers, convert the address on line C of the DA11-F configuration chart to a binary number and then cut the jumpers corresponding to binary 1s in the address. Figure 4-4 is the Window Starting Address Chart which provides a convenient means for determining the starting address. The following steps outline the procedure to complete the form.

- | Step | Procedure |
|------|---|
| 1 | Convert the A bus starting address from line C in the DA11-F Configuration Chart to an octal number using Table 4-1. This table lists the three most significant digits of the octal addresses of all locations on 512 word boundaries. The three least significant octal digits are always zero. |
| 2 | Convert each octal digit to its equivalent binary number. |
| 3 | Cut the designated jumpers on the odd data paths modules corresponding to 1s in the binary address. |
| 4 | Repeat Steps 1, 2, and 3 for the B bus starting address and cut jumpers corresponding to 1s in the binary address on the even data paths module. |



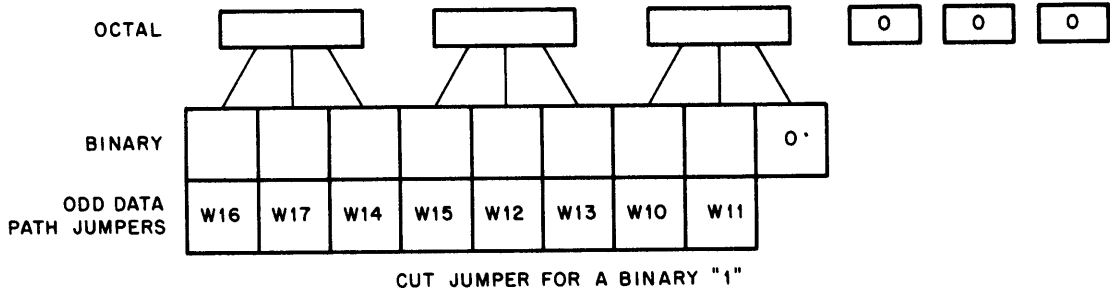
11-1529

Figure 4-3 Data Path Module Jumper Locations

Figure 4-5 shows the completed Window Starting Address Chart for the example shown in Figure 4-2 (with a BUS A starting decimal address of $44K_{10}$ and a BUS B starting decimal address of $16K_{10}$).

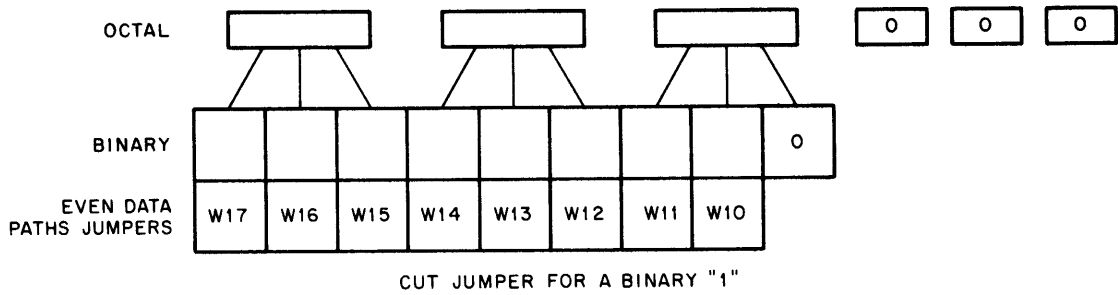
UNIBUS A

- A. Starting address (decimal) from line C of configuration chart. _____
- B. Starting address (octal) from Table 4-1. _____ 000
- C. Convert octal starting address to binary.



UNIBUS B

- A. Starting address (decimal) from line C of configuration chart. _____
- B. Starting address (octal) from Table 4-1. _____ 000
- C. Convert octal starting address to binary.

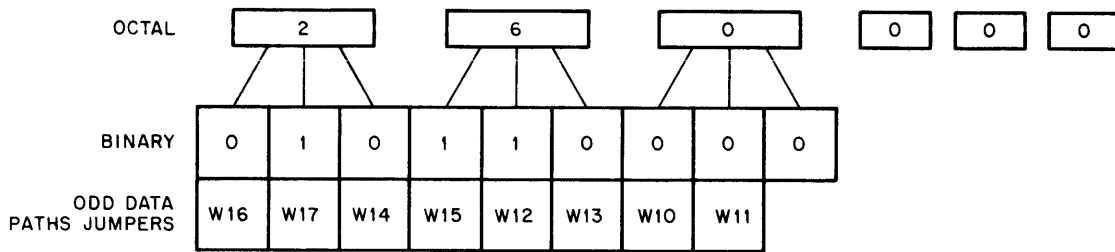


11-1531

Figure 4-4 Window Starting Address Chart

UNIBUS A

- A. Starting address (decimal) from line C of configuration chart. 44K
- B. Starting address (octal) from Table 4-1. 26000
- C. Convert octal starting address to binary.

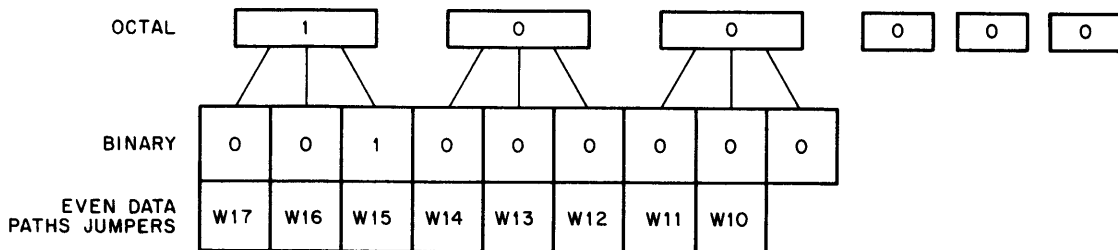


CUT JUMPER FOR A BINARY "1"

NOTE:
Cut jumpers W17, W15 and W12.

UNIBUS B

- A. Starting address (decimal) from line C of configuration chart. 16K
- B. Starting address (octal) from Table 4-1. 10000
- C. Convert octal starting address to binary.



CUT JUMPER FOR A BINARY "1"

NOTE:
Cut jumper W15.

11-1532

Figure 4-5 Example of Completed Window Starting Address Chart

Table 4-1
Decimal-To-Octal Address Conversion Chart

Decimal Address (In Kilo-Words)	0.0	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5.0	5.5	6.0	6.5	7.0	7.5
0	000	002	004	006	010	012	014	016	020	022	024	026	030	032	034	036
8	040	042	044	046	050	052	054	056	060	062	064	066	070	072	074	076
16	100	102	104	106	110	112	114	116	120	122	124	126	130	132	134	136
24	140	142	144	146	150	152	154	156	160	162	164	166	170	172	174	176
32	200	202	204	206	210	212	214	216	220	222	224	226	230	232	234	236
40	240	242	244	246	250	252	254	256	260	262	264	266	270	272	274	276
48	300	302	304	306	310	312	314	316	320	322	324	326	330	332	334	336
56	340	342	344	346	350	352	354	356	360	362	364	366	370	372	374	376
64	400	402	404	406	410	412	414	416	420	422	424	426	430	432	434	436
72	440	442	444	446	450	452	454	456	460	462	464	466	470	472	474	476
80	500	502	504	506	510	512	514	516	520	522	524	526	530	532	534	536
88	540	542	544	546	550	552	554	556	560	562	564	566	570	572	574	576
96	600	602	604	606	610	612	614	616	620	622	624	626	630	632	634	636
104	640	642	644	646	650	652	654	656	660	662	664	666	670	672	674	676
112	700	702	704	706	710	712	714	716	720	722	724	726	730	732	734	736
120	740	742	744	746	750	752	754	756	760	762	764	766	770	772	774	776

The vertical and horizontal coordinates of the chart list decimal addresses in rounded kilo-words (1K = 1024). The body of the chart lists the three most significant digits of the octal UNIBUS ADDRESSES. The three least significant digits are 000.

Example = Convert 44K to octal

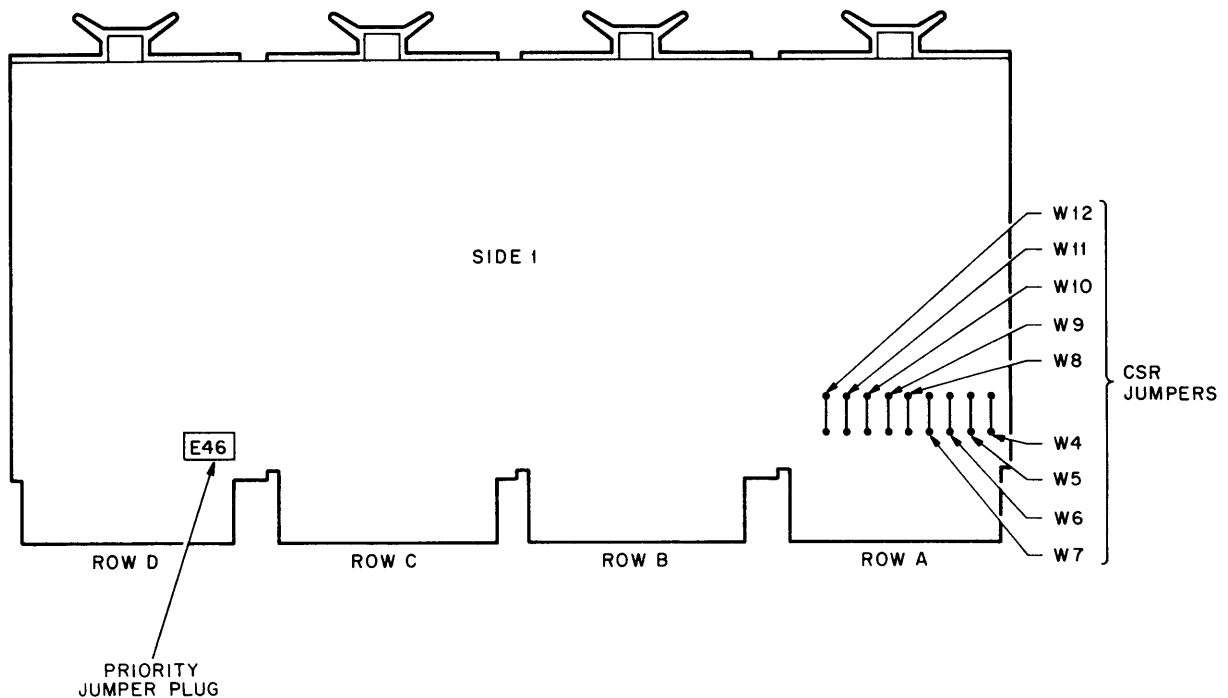
$$44K = 40K + 4K = 260000_8.$$

4.3.4 Control and Status Register Address Jumpers

The Control and Status Register contains the lowest address of the seven DA11-F addressable registers. The CSR address for each port is selected by jumpers on the corresponding bus control modules (see Figure 4-6 for approximate location of jumpers); the other six registers follow in sequence and are selected by the decoder circuits. The DA11-F addressable registers may be located anywhere in the I/O Register page (124 to 128K) but are normally assigned to the locations listed in Paragraph 4.3.1. Address bits A (12:04) are jumper selectable; A (17:13) are hardwired as 1s, and A (03:00) are internally decoded. To determine the CSR address jumpers to cut, fill out the CSR Address Chart shown in Figure 4-7 as follows:

Step	Procedure
1	Enter the A BUS CSR Address from line D of the DA11-F Configuration Chart.
2	Convert each octal digit to its equivalent binary number.
3	Cut the designated jumpers on the A bus control module corresponding to binary 1s in the binary address.
4	Repeat Steps 1, 2, and 3 for the CSR address of the B bus control module and cut jumpers corresponding to binary 1s on the B bus control module.

Figure 4-8 shows a completed CSR Address Chart using the data provided in the example in Figure 4-2.



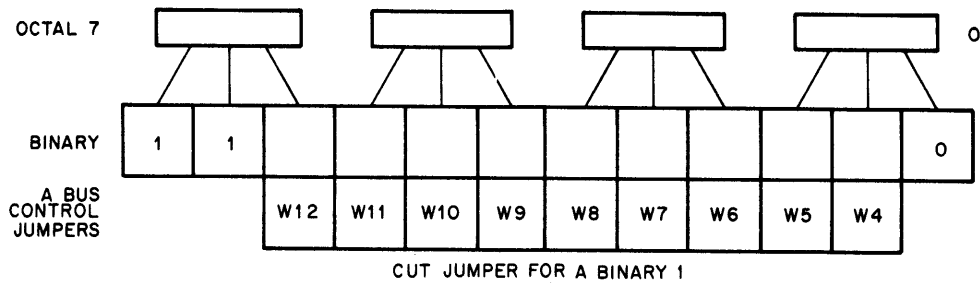
11-1530

Figure 4-6 Control Module Jumper Locations

UNIBUS A

A. CSR Address (octal) from line D of configuration chart. _____

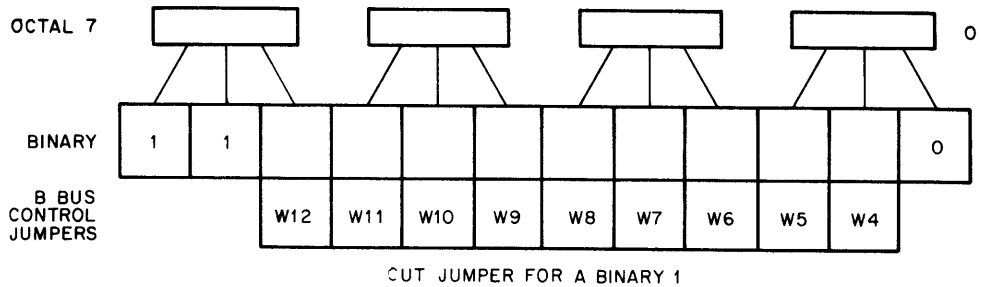
B. Convert octal to binary.



UNIBUS B

A. CSR address (octal) from line D of configuration chart. _____

B. Convert octal to binary.



11-1541

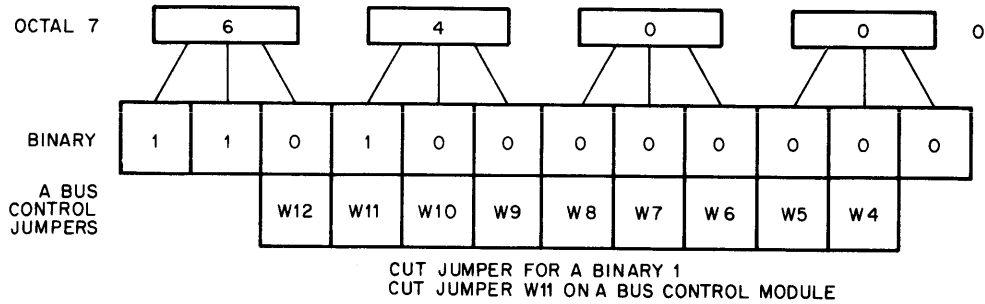
Figure 4-7 CSR Address Chart

UNIBUS A

A. CSR address (octal) from line D of configuration chart.

764000

B. Convert octal to binary.

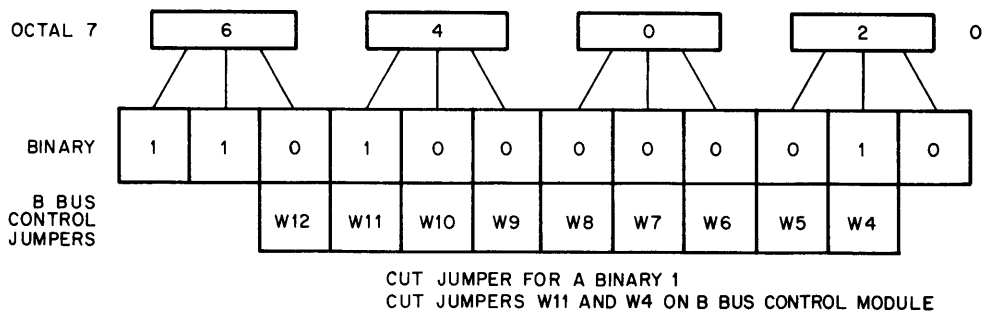


UNIBUS B

A. CSR address (octal) from line D of configuration chart.

764024

B. Convert octal to binary.

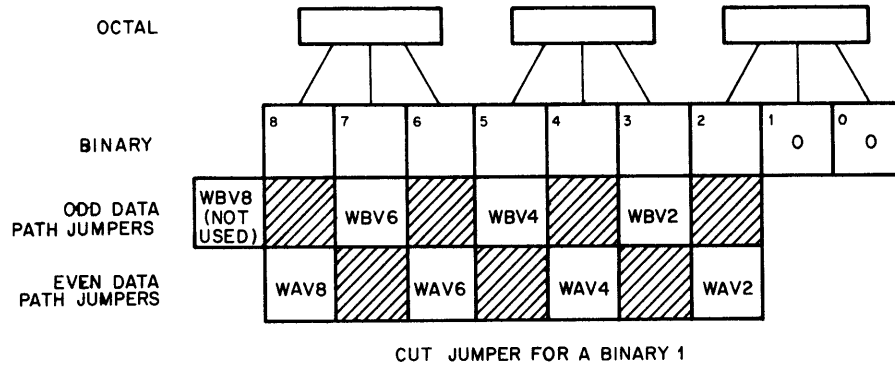


11-1542

Figure 4-8 Example of Completed CSR Address Chart

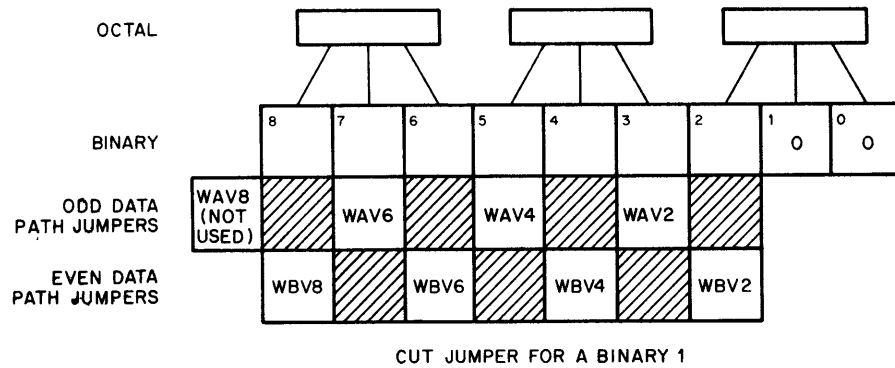
UNIBUS A

- A. Vector address from line E of configuration chart. _____
- B. Convert octal to binary.



UNIBUS B

- A. Vector address from line E of configuration chart. _____
- B. Convert octal to binary.



11-1543

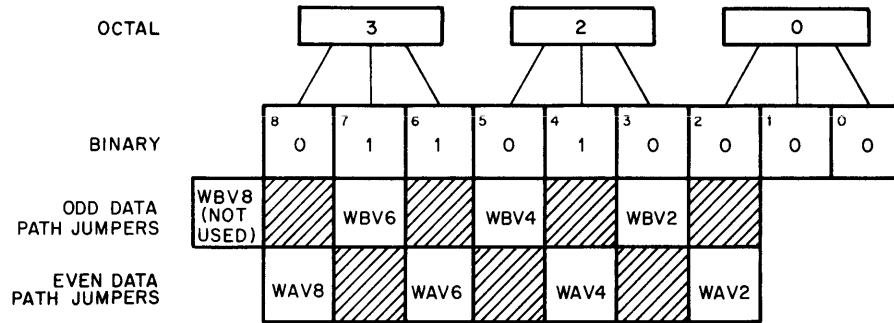
Figure 4-9 Vector Address Chart

UNIBUS A

A. Vector address from line E of configuration chart.

320

B. Convert octal to binary.



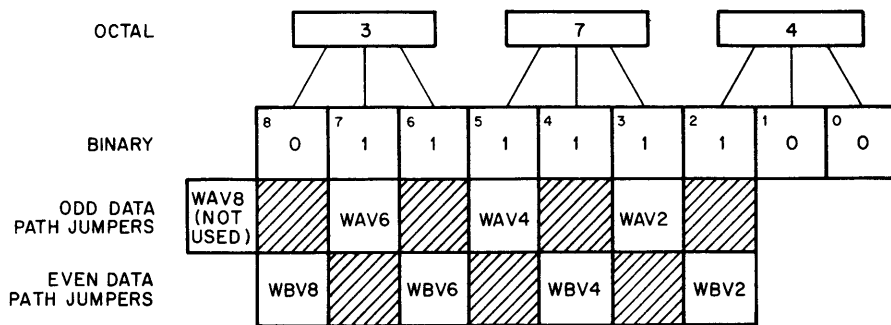
CUT JUMPER FOR A BINARY 1
 CUT JUMPER WBV6 ON THE ODD DATA PATH MODULE
 CUT JUMPERS WAV6 AND WAV4 ON THE EVEN DATA PATH MODULE

UNIBUS B

A. Vector address from line E of configuration chart.

374

B. Convert octal to binary.



CUT JUMPER FOR A BINARY 1
 CUT JUMPERS WAV6, WAV4 AND WAV2 ON THE ODD DATA PATH MODULE
 CUT JUMPERS WBV6, WBV4 AND WBV2 ON THE EVEN DATA PATH MODULE

11-1544

Figure 4-10 Example of Completed Vector Address Chart

4.3.5 Vector Address Jumpers

The vector address is the address of the interrupt service routine pointer used by the DA11-F. The vector address jumpers are located on the data path modules (Figure 4-3). Because the data path modules are symmetrical, the jumpers for a particular port are located partly on the odd data path module and partly on the even data path module. Vector address bits V (08:02) are jumper selectable while vector address bits V (01:00) are permanently wired as 0s. To determine the vector address, reproduce the Vector Address Chart (Figure 4-9) and proceed as follows:

Step	Procedure
1	Enter the Unibus A vector address from line E of the DA11-F Configuration Chart (Figure 4-2).
2	Convert the vector address to its binary equivalent.
3	Cut jumpers on the appropriate module if the binary equivalent of the address is a 1 on that module.
4	Repeat the above procedure for the Unibus B vector address by completing the lower portion of the Vector Address Chart.

Figure 4-10 shows the vector address jumper configuration for both Unibus A and Unibus B based on the example vector addresses provided in Figure 4-2.

4.3.6 Priority Jumper Plugs

The interrupt priority (BR level) of the DA11-F on each bus is determined by the type of priority jumper plug inserted in location E46 of the appropriate bus control module (Figure 4-6). Normally, the DA11-F is assigned to BR7 (highest interrupt level) due to the importance of bus window interrupt service to system performance. The DA11-F is equipped with BR7 priority jumper plugs on both ports when manufactured. To change a BR level in the field, the BR7 jumper plug is removed and replaced with a jumper plug corresponding to the desired interrupt level. Field Service personnel from DEC should be consulted before modifying BR levels. To ensure proper system operation, a priority jumper plug must be installed in each port of the DA11-F.

4.3.7 Window Size Jumper Plugs

The size of the window field is determined by a set of jumper plugs that are inserted in receptacles in the data path modules (Figure 4-3). Each of these plugs determines whether a particular bit is used as part of the displacement address or as part of the relocation address. A total of six jumper plugs must be installed – three on the odd data path module and three on the even data path module. The plugs are identified by the following DEC part numbers printed on them:

DA (Displacement Address) Plug	P/N 54-10212
RA (Relocation Address) Plug	P/N 54-10214

To configure the desired window size, install the plug pattern denoted in Table 4-2 based on the window size selected on line A of the DA11-F Configuration Chart. For example, on line A of Figure 4-2, a 4K window is specified. Referring to Table 4-2, address bits A(15:13) should have DA plugs installed in slots E30 and E52 on the even data path module and slot E30 on the odd data path module. RA plugs should be installed in slots E29 and E52 on the odd data path module and E29 on the even data path module.

**Table 4-2
Window Size Chart**

Window Size	Type of Plug Installed in Address Bit					
	A15	A14	A13	A12	A11	A10
1/2K	RA	RA	RA	RA	RA	RA
1K	RA	RA	RA	RA	RA	DA
2K	RA	RA	RA	RA	DA	DA
4K	RA	RA	RA	DA	DA	DA
8K	RA	RA	DA	DA	DA	DA
16K	RA	DA	DA	DA	DA	DA
32K	DA	DA	DA	DA	DA	DA
Location:	E29 Odd	E29 Even	E52 Odd	E52 Even	E30 Odd	E30 Even

4.3.8 Mechanical Installation

Since the DA11-F must connect between two Unibuses, it must be installed in a PDP-11 chassis that can accept hex-height modules such as the BA11-F Expander Box.

The DA11-F system unit frame is oriented with its bus cable slots in line with those of the other options in the chassis. The system unit is attached to the chassis by means of a thumbscrew at either end of the system unit.

The modules are inserted firmly in the system unit slots according to the functions of the modules specified by the tags which were attached to each module. Refer to Figure 1-3 for the module locations.

4.3.9 Electrical Installation

The electrical installation of the Unibus window consists of connecting both Unibuses to the DA11-F and also connecting the power cable from the system unit to the mounting box distribution panel.

4.3.9.1 Unibus Connection – Unibus A is connected to the outermost bus cable slots. BUS A IN is inserted in slot AB01 and BUS B OUT is inserted into slot AB04. If other options are installed in adjacent system unit positions in the chassis, the M920 Unibus Jumper Module can be used to make the bus connections.

Unibus B is connected to the inner bus cable slots. The BC11-A cable from the companion computer is inserted in slot AB03 (BUS B IN). If required, Unibus B can be extended from slot AB02 (BUS B OUT) via another BC11-A cable.

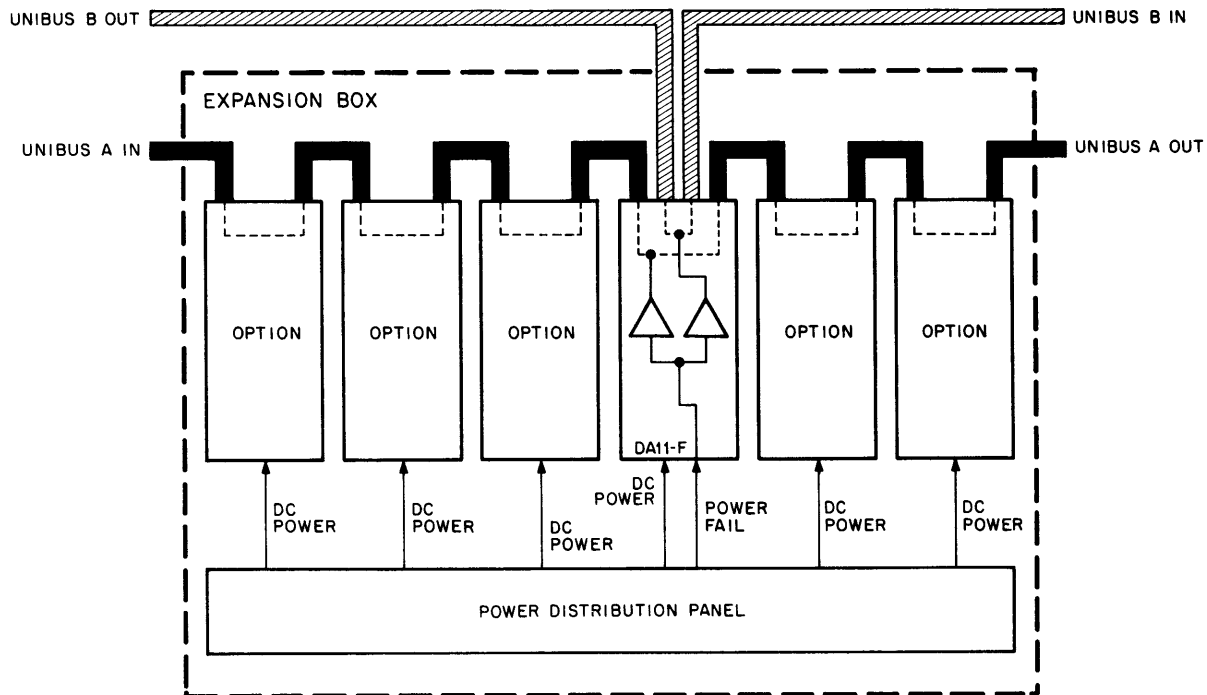
If either Unibus ends at the DA11-F, that bus must be terminated by inserting an M930 Unibus Terminator Module in the corresponding BUS OUT slot.

4.3.9.2 Power Cable Connection – The +5 Vdc power is distributed to the modules through the printed circuit backplane that is attached to the system unit assembly. The power cable attaches to the backplane by means of quick-disconnect tabs (refer to the cable drawings in the engineering drawing set for connections). Two types of cables are furnished with the DA11-F:

1. If the DA11-F is to be installed in an expansion box, use power cable Part No. 7009099-0-0. The rectangular MATE-N-LOK connector is inserted in the power distribution panel located above the system unit spaces.
2. If the DA11-F is to be installed in a PDP-11/45 central processor box, use power cable Part No. 7009162-0-0. The flat MATE-N-LOK connector is inserted in the power distribution panel located on the rear panel of the chassis.

Because the DA11-F is connected to two Unibuses, a special power fail signal interconnection method is required. The power fail signals (ACLO and DCLO) originate in the power supply of the mounting box in which the DA11-F is installed. If that power supply shuts down, both processors must be notified that a system power failure is about to occur. In order to assert the signals on both buses, the power fail signals are connected from the power supply to the DA11-F via the power cable. The M7284 Bus Control Modules contain driver circuits that are used to assert ACLO and DCLO on both Unibuses as described in Paragraph 2.10.

Figure 4-11 illustrates the correct method of connecting the DA11-F and other options to the power distribution panel in a typical PDP-11 Expansion Box. This cabling scheme furnishes dc power to all options in the box but connects the Power Fail signals (ACLO and DCLO) only to the DA11-F. The A and B power fail drivers in the DA11-F then assert the signals on both Unibuses. Since the power cables normally supplied with the other options already contain connections for the power fail signals, it is necessary to break these connections when installing a DA11-F. The exact method of breaking the connections depends on the type of option and the type of expansion box. In some cases, the power fail signals can be deleted by simply cutting jumpers on the distribution panel or the cable connector module. In other situations, it may be necessary to physically remove the ACLO and DCLO wires from the options power cable to break the connections.



NOTE:
The figure shows typical power cable connections
for a DA11-F when installed in a PDP-11 expansion box.

11-1540

Figure 4-11 Typical Power Cable Connections for a DA11-F

In summary, to install a DA11-F in a PDP-11 Mounting Box:

1. Connect dc power to all options in that box.
2. Connect ACLO and DCLO only to the DA11-F. Break power fail signal connections to all other devices in the box.

4.4 PREVENTIVE MAINTENANCE

There are no special preventive maintenance tasks associated with the DA11-F; however, the following points should be kept in mind when working on this device:

1. Verify that all modules have been configured correctly in accordance with the procedures described in the Installation Section.
2. Ensure that all modules and bus cables have been seated firmly in the system unit.
3. Inspect backplane wiring for broken wires or damaged pins. Repair or replace as required.
4. Ensure that the power cable is firmly attached to the correct tabs on the system unit backplane and that the MATE-N-LOK connector is firmly seated in the power distribution panel on the chassis.
5. Remove power from both Unibuses by de-energizing both computers before removing or installing modules or bus cables.
6. Ensure that both Unibuses are terminated at both ends before operating the system.
7. Verify that the logic power is +5 Vdc $\pm 0.25V$ at the system unit backplane.

There are no adjustments, timing margin, or voltage margin tests for the DA11-F.

4.5 CORRECTIVE MAINTENANCE

Corrective maintenance on the DA11-F is normally performed by running the DA11-F diagnostic programs to verify the units operation. If errors are reported by the diagnostics, standard troubleshooting procedures should be used to isolate the problem and correct the malfunction. A thorough understanding of Unibus theory and the DA11-F theory of operation is essential for locating malfunctions and initiating corrective action.

The following test equipment is recommended for troubleshooting:

Equipment	Type
Oscilloscope	Tektronix 453 or equivalent
Volt/ohmmeter	Triplet or equivalent
Two double height X 8.5 inch module extender boards	DEC W900
16-pin IC clip	AP Incorporated AP923700

4.6 DIAGNOSTIC PROGRAMS

Two diagnostic programs are furnished with the DA11-F:

Bus Window Static Test	MD-11-DZDAA
Bus Window Exerciser	MD-11-DZDAB

Complete instructions on operating and using these programs are included in the program listings.

4.6.1 Bus Window Static Test

The Static test program is a complete functional test of all the programmable features and window channel data paths through the DA11-F. The static test can be run in three different modes: One Bus Test, Two Bus Test, and Power Fail.

The One Bus Test provides a convenient means of verifying the programmable features of the DA11-F using only one central processor. The standard configuration procedure implements different addresses for the A port and B port CSR (Control and Status Register). Therefore, if both ports are connected to the same processor, each can be operated independently of the other. A special bus connection scheme is required to use the One Bus Test:

1. Remove all Unibus cables or jumper modules from slots AB04 (BUS A OUT), AB03 (BUS B IN), and AB02 (BUS B OUT).
2. Insert the M9200 Unibus Jumper Module (furnished with the accessory kit) between slots AB04 and AB03. This connects BUS A OUT to BUS B IN.
3. Insert an M930 Unibus Termination Module in slot AB02 (BUS B OUT).

The One Bus Test verifies all the program control operations in both ports and also checks the cross-communications and cross-interrupt facilities between the ports. No window transactions are performed in this test.

The Two Bus Test verifies all programmable operations on the DA11-F and all window channel transactions in both directions. This test requires the standard dual processor bus cable connection. After the Static Test program is loaded and started in both computers, the two processors synchronize and proceed through all the tests in step with each other. One processor acts as master and initiates a dialogue with the companion computer. After each pass of the program, the processors reverse roles and repeat the test sequence using the opposite computer as the master.

The Power Fail Test checks all possible combinations of power fail sequences (Paragraph 2.10) and verifies that the DA11-F transmits the appropriate signals to both computers.

4.6.2 Bus Window Exerciser

The exerciser program verifies that each processor can execute code through the window. The program also operates any mass storage devices (RF11, RK11, RP11, RC11, TC11, or TM11) that are attached to either computer and verifies that each can transfer data through the window to the memory on the opposite bus. By dynamically changing the value loaded in the Relocation Address Registers, the exerciser tests window operations throughout all the memory connected to each Unibus.

4.7 PARTS LIST

The DA11-F Unibus window option consists of the following items:

1. One DA11-F System Unit
2. Two M7283 Window Data Path Modules
3. Two M7284 Window Bus Control Modules

In addition, the following items are also supplied for use in configuring or maintaining the option.

1. Six DA Jumper Plugs, PN 54-10212
2. Six RA Jumper Plugs, PN 54-10214

NOTE

**Six of the twelve jumper plugs are installed on the modules.
The other six are supplied with the option in case of field
changes to the configuration.**

3. One M9200 Unibus Jumper Module
4. One Power Cable, P/N 7009099-0-0, for use in an Expander Box
One Power Cable, P/N 7009162-0-0, for use in the PDP-11/45 CPU Box
5. Two diagnostic programs:

Static Test MD-11-DZDAA	Listing and Paper Tape
Exerciser MD-11-DZDAB	Listing and Paper Tape

4.8 IC SPARE PARTS LIST

The following is a listing of the recommended IC spares which should be available.

IC Type	Description	DEC Part No./Drawing No.	Recommended Spares
380	Quad 2-input NOR	19-09485	3
7400	Quad 2-input NAND	19-05575	2
7402	Quad 2-input NOR	19-09004	2
7404	Hex Inverter	19-9686	2
7408	Quad 2-input AND	19-10155	2
7410	Triple 3-input NAND	19-05576	2
7416	Hex Inverter Driver	19-09928	2
7430	Single 8-input NAND	19-05578	2
7440	Double 4-input NOR	19-05579	2
7474	Dual D-Type Flip-Flops	19-05547	2
74123	Dual Retriggerable One-shot	19-10436	2
74151	1 of 8 MUX	19-09936	4
74153	Dual 1 of 4 MUX	19-09937	2
74174	Hex D-Type Flip-Flops	19-10652	2
74175	Quad D-Type Flip-Flops	19-10651	2
74H53	Expandable AND OR Inverter	19-09062	2
74H55	2 Wide AND OR Inverter	19-09063	2
74H74	Dual D-Type Flip-Flops	19-09667	2
8242	Quad 2-input X NOR	19-09712	2
8881	Quad 2-input NAND	19-09075	4
7488	Read-Only Memory	23-006A1-04	2

APPENDIX A IC DESCRIPTION

74H74 D-TYPE EDGE-TRIGGERED FLIP-FLOPS

The 74H74 consists of two D-type edge-triggered flip-flops. Each flip-flop has individual clear and preset inputs and complementary Q and \bar{Q} outputs. Information at input D is transferred to the Q output on the positive-going edge of the clock pulse.

TRUTH TABLE (Each Flip-Flop)

t_n	t_{n+1}	
INPUT D	OUTPUT Q	OUTPUT \bar{Q}
L	L	H
H	H	L

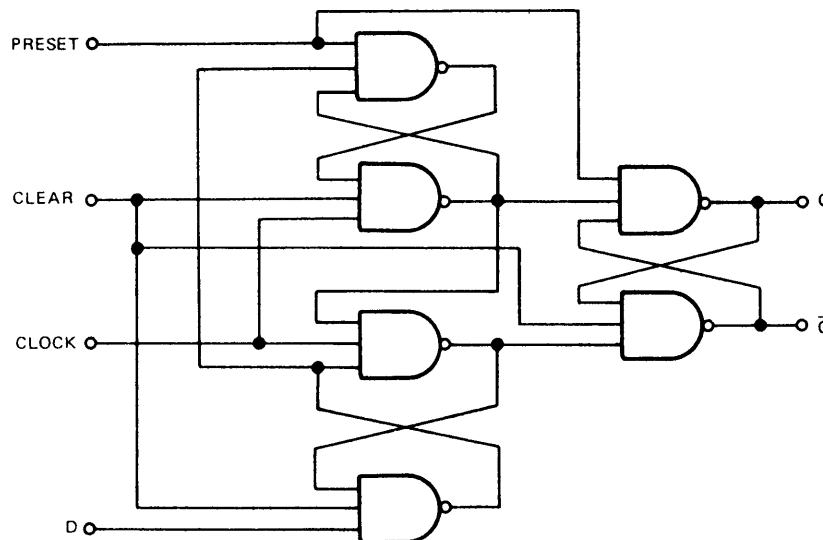
H = high level, L = low level

NOTES: A. t_n = bit time before clock pulse.
B. t_{n+1} = bit time after clock pulse.

Signal/Pin Designation

Signal Name	Circuit #1	Circuit #2
D	2	12
CLOCK	3	11
CLEAR	1	13
PRESET	4	10
Q	5	9
\bar{Q}	6	8

functional block diagram (each flip-flop)



7488 READ-ONLY MEMORY

The 7488 Read-Only Memory is custom programmed and consists of 256 bits organized as 32 8-bit words. Each 32-word array is addressed in straight 5-bit binary code. The memory includes an over-riding memory enable input, which when high will inhibit all 32 address gates and cause the outputs to remain high.

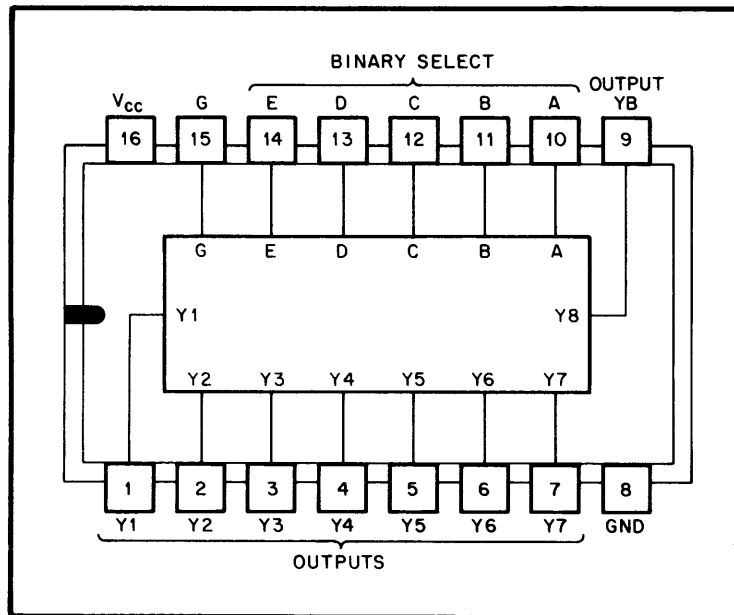
The read-only memory is preprogrammed by DEC and after it has been programmed is assigned DEC part number 23-006A1-04. The pin connection diagram, schematic, and ROM program pattern is shown below. Binary select inputs A, B, C, D, E shown on the pin connection diagram are the array address inputs. Input A is the least significant bit of the address and input E is the most significant bit.

For example, the binary select configuration below corresponds to the octal location of 25.

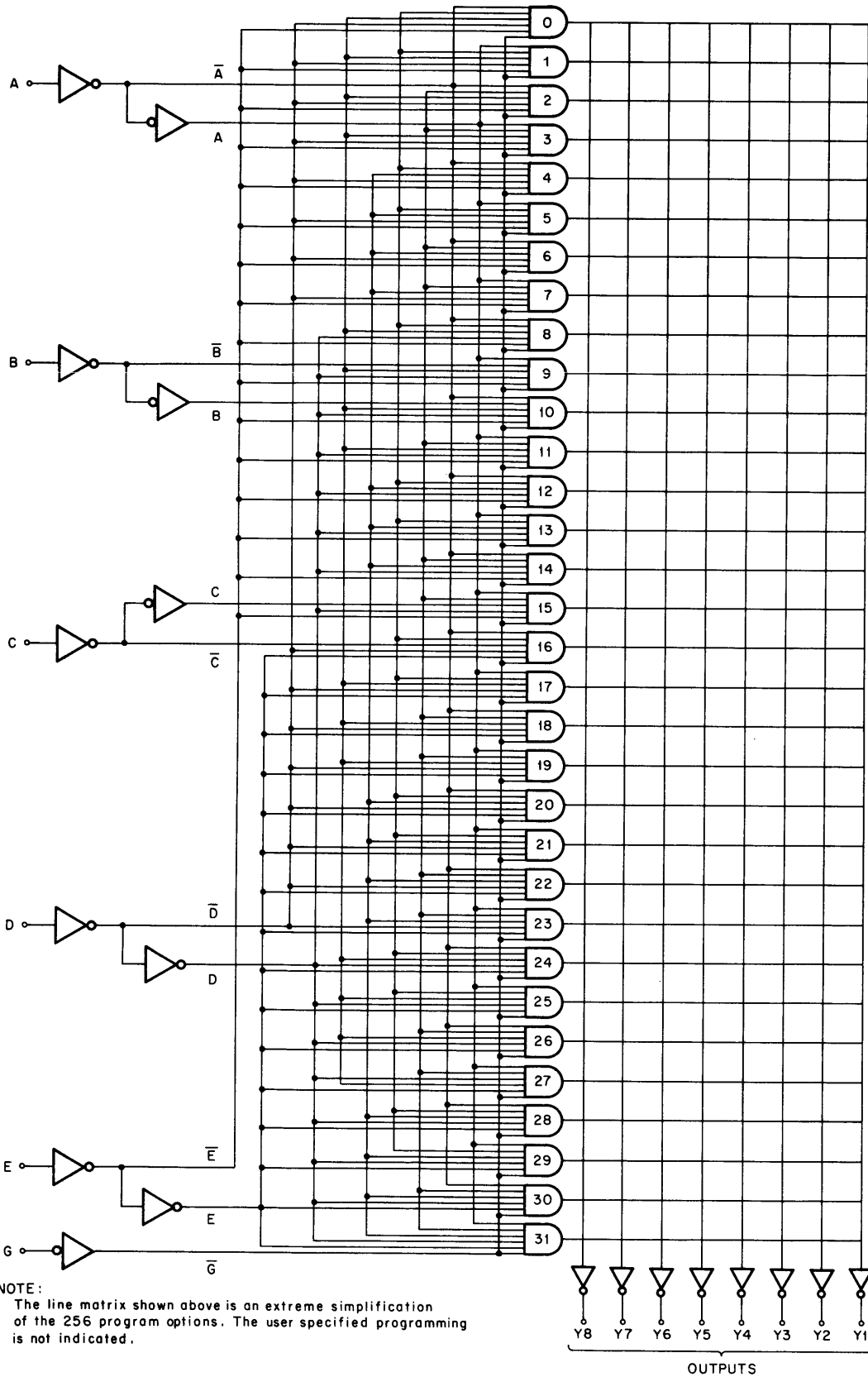
SELECT INPUTS	E D		C B A
Binary Address	1 0		1 0 1 ₂
Octal Address	2		5 ₈

Definitions:

Signal To	Binary Value	Signal Level
BINARY SELECT INPUT(E:A)	1	H
	0	L
ENABLE G	Enable	L
	Disable	H
Output(Y8: Y1)	1	L
	0	H



11-1115



PART NUMB: 23-006A1-04

DECIMAL LOC	OCTAL LOC	BINARY DATA	OCTAL DATA
0	00	11101111	357
1	01	11111111	377
2	02	11111111	377
3	03	11111111	377
4	04	11111111	377
5	05	11111111	377
6	06	11111111	377
7	07	11111111	377
8	10	11001111	317
9	11	01111111	177
10	12	11111111	377
11	13	11111111	377
12	14	10111111	277
13	15	11111111	377
14	16	11111111	377
15	17	11111111	377
16	20	11011111	337
17	21	11111111	377
18	22	11111111	377
19	23	11111111	377
20	24	11111111	377
21	25	11111111	377
22	26	11111111	377
23	27	11111111	377
24	30	11110111	367
25	31	11110110	366
26	32	11110101	365
27	33	11110100	364
28	34	11110011	363
29	35	11110010	362
30	36	11110001	361
31	37	11110000	360

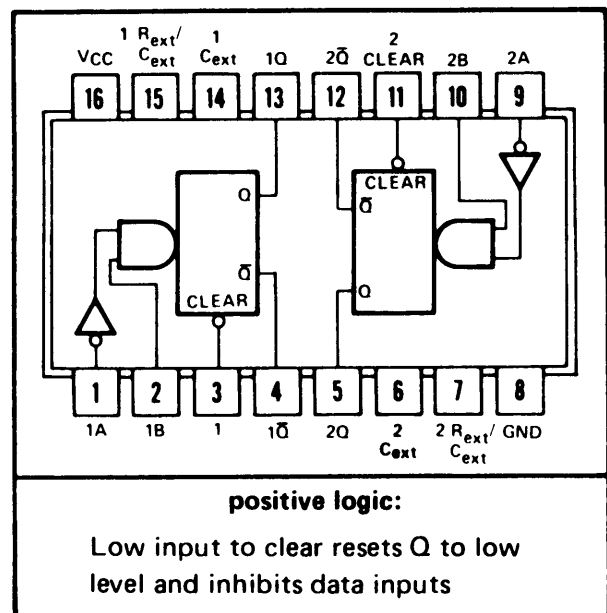
74123 RETRIGGERABLE MONOSTABLE MULTIVIBRATOR

The 74123 multivibrator features dc triggering from gated low-level active (A) and high-level active (B) inputs. The circuit also features over-riding direct clear input and complementary outputs. The duration of the output pulse may be extended by triggering the input before the output is terminated. The over-riding clear input permits any output pulse to be terminated at some predetermined time.

Truth Table
(See Note A)

INPUTS		OUTPUTS	
A	B	Q	\bar{Q}
H	X	L	H
X	L	L	H
L	↑	⎓	⎓
↓	H	⎓	⎓

J or N Dual-In-Line or
W Flat Package (Top View)
(See Note B)



† Pin assignments for these circuits are the same for all packages.

- A. H = high level (steady state), L = low level (steady state), ↑ = transition from low to high level, ↓ = transition from high to low level, ⎓ = one high-level pulse, ⎓ = one low-level pulse, X = irrelevant (any input, including transitions).
- B. An external timing capacitor may be connected between C_{ext} and R_{ext}/C_{ext} (positive).

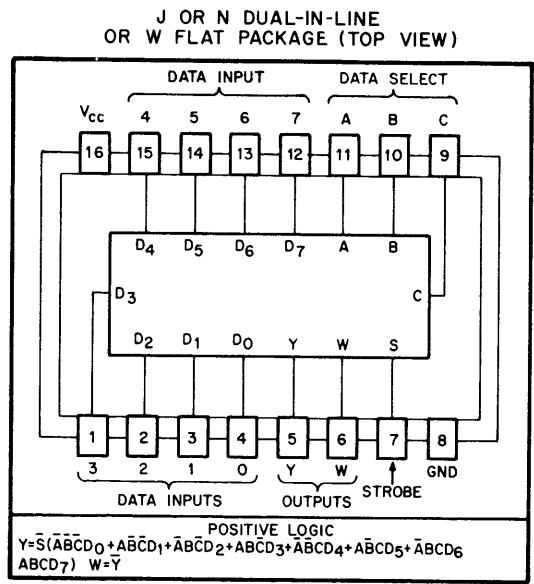
74151 8-LINE TO 1-LINE MULTIPLEXER

The 74151 selects one of eight data sources for multiplexing the output onto one line. The circuit can be used for parallel-to-serial conversion or can be used as a five-variable function operator.

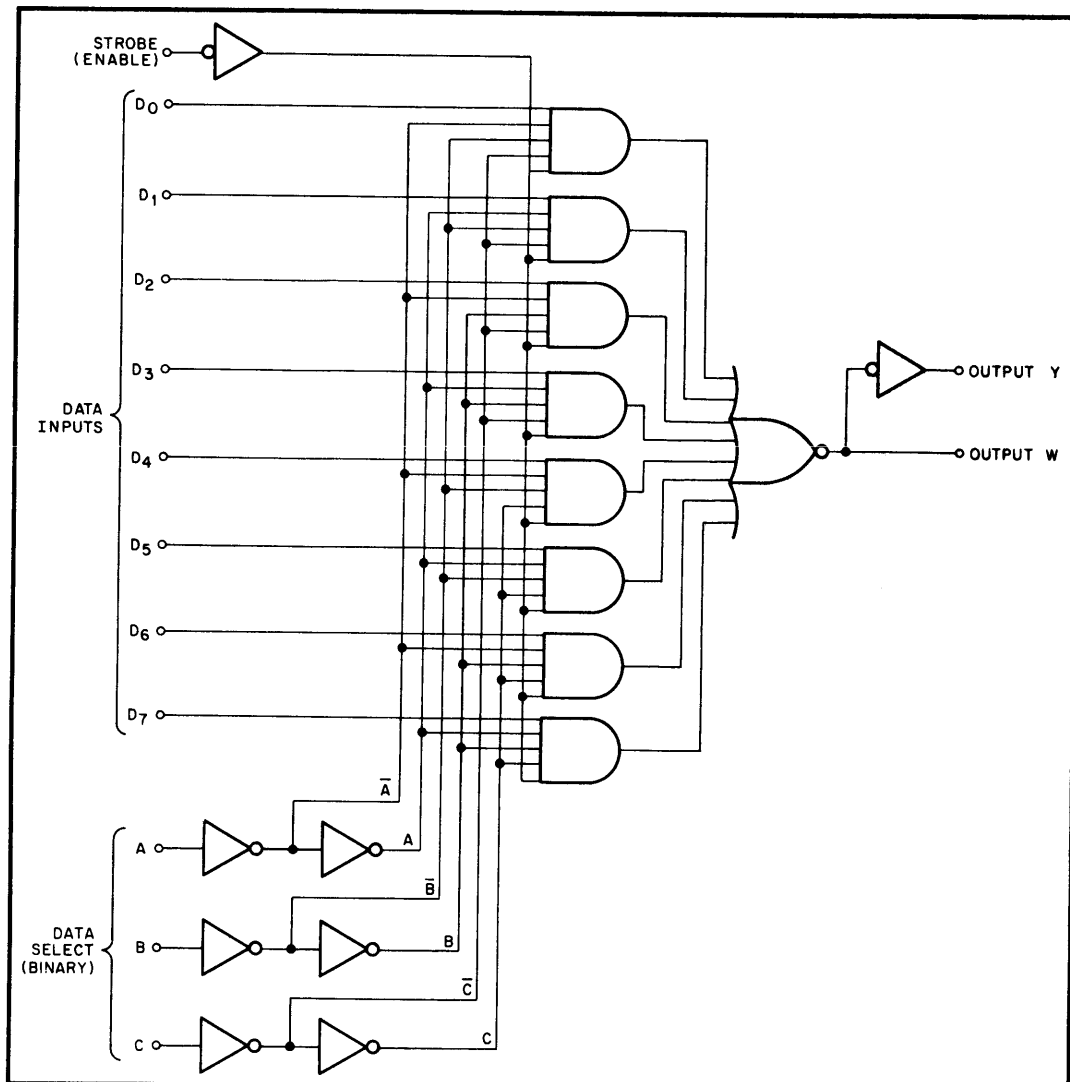
TRUTH TABLE

Inputs												Outputs	
C	B	A	Strobe	D ₀	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	Y	W
x	x	x	1	x	x	x	x	x	x	x	x	0	1
0	0	0	0	0	x	x	x	x	x	x	x	0	1
0	0	0	0	1	x	x	x	x	x	x	x	1	0
0	0	1	0	x	0	x	x	x	x	x	x	0	1
0	0	1	0	x	1	x	x	x	x	x	x	1	0
0	1	0	0	x	x	0	x	x	x	x	x	0	1
0	1	0	0	x	x	1	x	x	x	x	x	1	0
0	1	1	0	x	x	x	0	x	x	x	x	0	1
0	1	1	0	x	x	x	1	x	x	x	x	1	0
1	0	0	0	x	x	x	x	0	x	x	x	0	1
1	0	0	0	x	x	x	x	1	x	x	x	1	0
1	0	1	0	x	x	x	x	x	0	x	x	0	1
1	0	1	0	x	x	x	x	x	1	x	x	1	0
1	1	0	0	x	x	x	x	x	x	0	x	0	1
1	1	0	0	x	x	x	x	x	x	1	x	1	0
1	1	1	0	x	x	x	x	x	x	x	0	0	1
1	1	1	0	x	x	x	x	x	x	x	1	1	0

When used to indicate an input, x = irrelevant.



II-0634



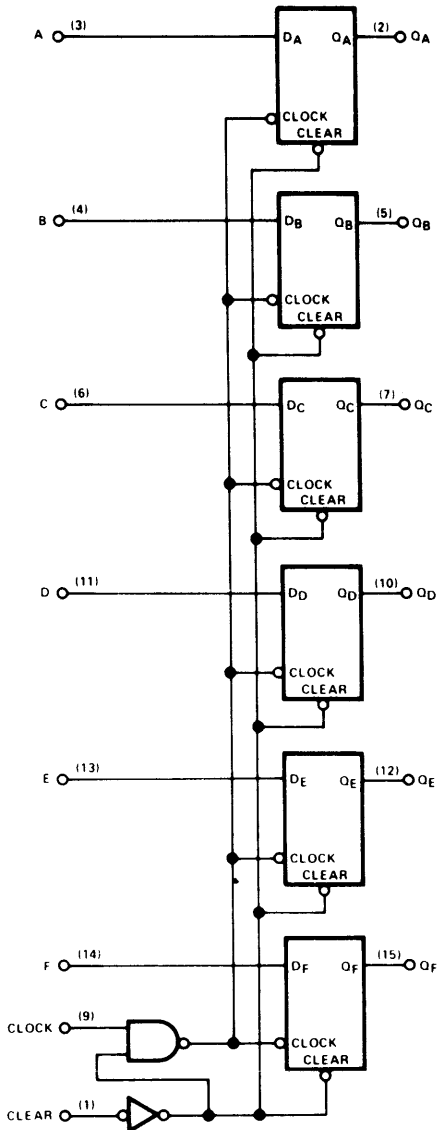
11-0635

74174/74S174 HEX D-TYPE FLIP-FLOPS

The 74S174 contains six flip-flops with single outputs. The flip-flops contain direct clear inputs and buffered clock inputs.

INPUT	OUTPUTS	
t_n	t_{n+1}	
D	Q	Q
H	H	L
L	L	H

t_n = Bit time before clock pulse.
 t_{n+1} = Bit time after clock pulse.



Pin (16) = V_{CC} , Pin (8) = GND

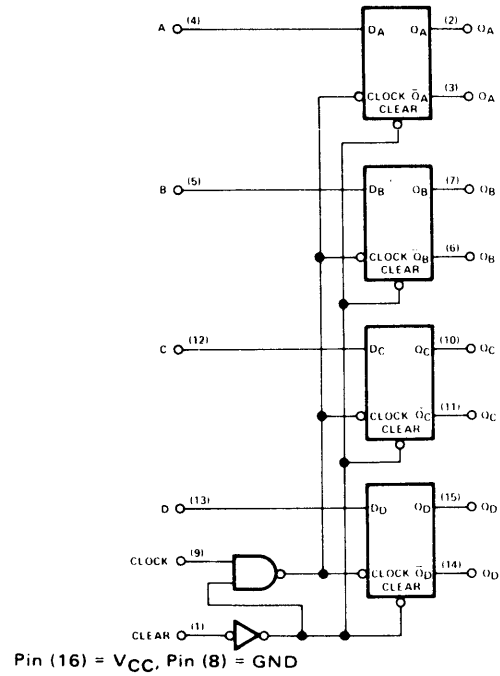
74175 QUAD D-TYPE FLIP-FLOPS

The 74175 contains four D-type flip-flops with dual outputs. Each flip-flop has direct clear and buffered clock inputs.

INPUT t_n	OUTPUTS t_{n+1}	
D	Q	\bar{Q}
H	H	L
L	L	H

t_n = Bit time before clock pulse.

t_{n+1} = Bit time after clock pulse.



**DA11-F UNIBUS WINDOW MAINTENANCE MANUAL
ERRATA SHEET**

The following changes are to be inserted in the above manual as shown below:

Page 4-7 – On Figure 4-3, interchange WA15 and WA14.

Page 4-16 – Last two sentences should read:

Referring to Table 4-2, address bits A(15:13) should have DA plugs installed in slots E30 and E52 on the even data path module and slot E30 on the odd data path module. RA plugs should be installed in slots E29 and E52 on the odd data path module and E29 on the even data path module.

Page 4-17 – Delete Table 4-2 and insert the Table below in its place.

**Table 4-2
Window Size Chart**

Window Size	Type of Plug Installed in Address Bit					
	A15	A14	A13	A12	A11	A10
1/2K	RA	RA	RA	RA	RA	RA
1K	RA	RA	RA	RA	RA	DA
2K	RA	RA	RA	RA	DA	DA
4K	RA	RA	RA	DA	DA	DA
8K	RA	RA	DA	DA	DA	DA
16K	RA	DA	DA	DA	DA	DA
32K	DA	DA	DA	DA	DA	DA
Location:	E29 Odd	E29 Even	E52 Odd	E52 Even	E30 Odd	E30 Even

READER'S COMMENTS

**DA11-F UNIBUS WINDOW
MAINTENANCE MANUAL
DEC-11-HDAFA-D**

Your comments and suggestions will help us in our continuous effort to improve the quality and usefulness of our publications.

What is your general reaction to this manual? In your judgment is it complete, accurate, well organized, well written, etc.? Is it easy to use? _____

What features are most useful? _____

What faults do you find with the manual? _____

Does this manual satisfy the need you think it was intended to satisfy? _____

Does it satisfy *your* needs? _____ Why? _____

Would you please indicate any factual errors you have found. _____

Please describe your position. _____

Name _____ Organization _____

Street _____ Department _____

City _____ State _____ Zip or Country _____

digital

**DIGITAL EQUIPMENT CORPORATION
MAYNARD, MASSACHUSETTS 01754**