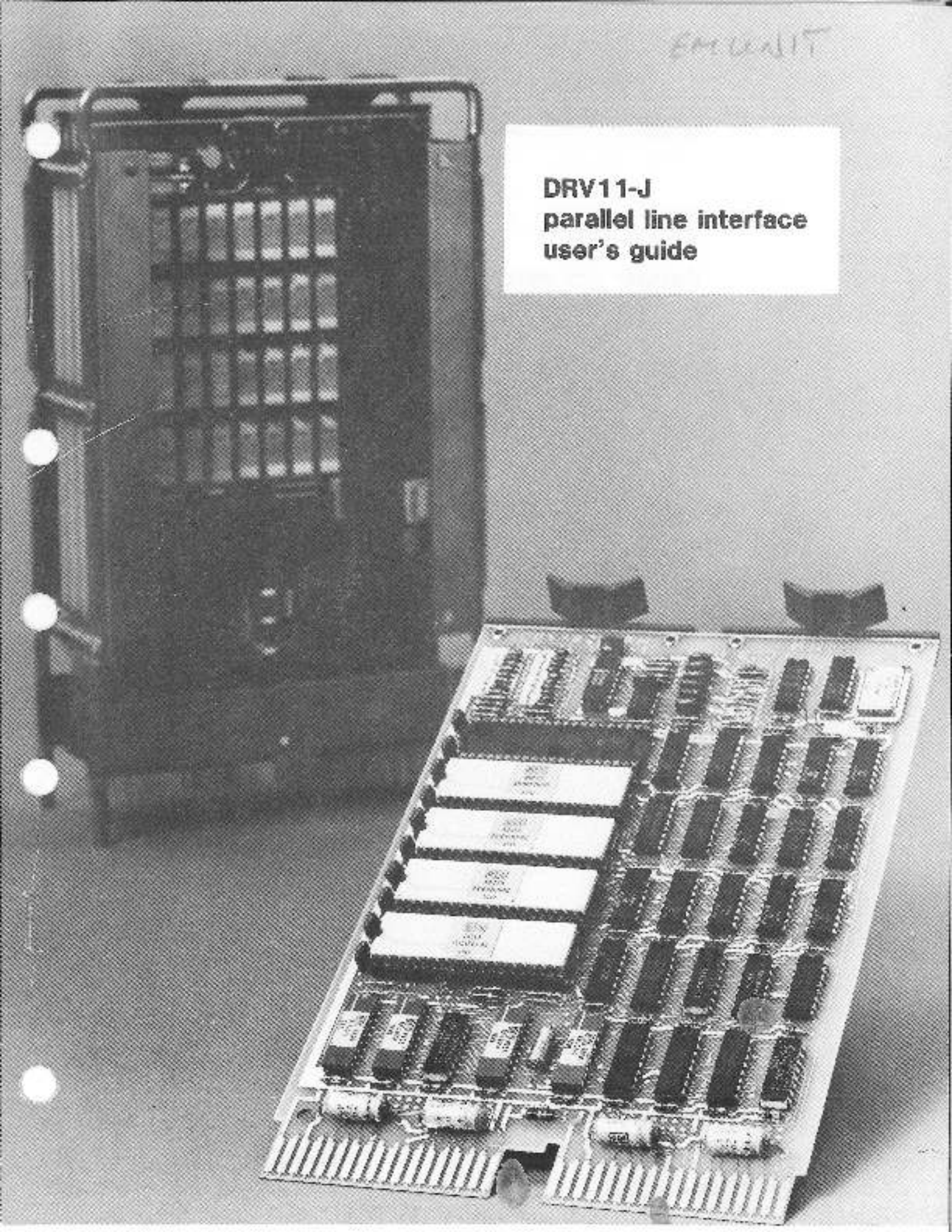


DRV11-J
parallel line interface
user's guide

DRV11-J
parallel line interface
user's guide

EK-DRV1J-UG-002

digital equipment corporation • marlboro, massachusetts

1st Edition, December, 1979
2nd Edition, November, 1980

Printed in U.S.A.

This document was set on DIGITAL's DECset-8000 computerized typesetting system.

CONTENTS

		Page
CHAPTER 1	**INTRODUCTION**	
1.1	GENERAL DESCRIPTION	1-1
1.2	FEATURES	1-1
1.3	DOCUMENTATION	1-2
1.4	DIAGNOSTIC SOFTWARE	1-2
1.5	SPECIFICATIONS	1-2
1.5.1	Physical Specifications	1-2
1.5.2	Electrical Specifications	1-2
1.5.3	Environmental Specifications	1-3
1.5.3.1	Operating and Storage Temperature Ranges	1-3
1.5.3.2	Relative Humidity	1-3
1.5.3.3	Airflow during Operation	1-3
1.5.3.4	Altitude	1-3
1.6	INSTALLATION	1-3
CHAPTER 2	**FUNCTIONAL DESCRIPTION**	
2.1	GENERAL DESCRIPTION	2-1
2.2	CONTROL/STATUS REGISTERS	2-1
2.3	DATA BUFFER REGISTERS	2-1
2.4	INTERRUPT CONTROL	2-1
2.4.1	Functional Description	2-7
2.4.2	Interrupt Controller Interface	2-7
2.4.3	Interrupt Controller Operating Description	2-9
2.4.4	Interrupt Control Reset	2-12
2.4.5	Interrupt Control Register Description	2-13
2.4.5.1	Status Register	2-13
2.4.5.2	Command Register	2-14
2.4.5.3	Mode Register	2-14
2.4.5.4	Interrupt Request Register (IRR)	2-14
2.4.5.5	Interrupt Service Register (ISR)	2-15
2.4.5.6	Interrupt Mask Register (IMR)	2-16
2.4.5.7	Auto-Clear Register (ACR)	2-16
2.4.5.8	Vector Address Memory	2-16
2.5	OPERATING OPTIONS	2-17
2.5.1	Interrupt Priority Mode Selection	2-17
2.5.2	Individual Vector or Common Vector Mode	2-19
2.5.3	Interrupt or Polled (Flag) Mode	2-19
2.5.4	Mode Register Bit 3	2-19
2.5.5	IRQ Polarity Option	2-19
2.5.6	Register Preselection Option	2-19
2.5.7	Master Mask Option	2-20
2.6	SYSTEM OPERATING SEQUENCE	2-20
2.7	COMMAND DESCRIPTIONS	2-21
2.7.1	Reset	2-21
2.7.2	Clear IRR and IMR	2-21
2.7.3	Clear Single IMR and IRR Bit	2-21
2.7.4	Clear IMR	2-21

CONTENTS (Cont)

		Page
2.7.5	Clear Single IMR Bit	2-22
2.7.6	Set IMR	2-22
2.7.7	Set Single IMR Bit	2-22
2.7.8	Clear IRR	2-22
2.7.9	Clear Single IRR Bit	2-22
2.7.10	Set IRR	2-22
2.7.11	Set Single IRR Bit	2-23
2.7.12	Clear Highest Priority ISR Bit	2-23
2.7.13	Clear ISR	2-23
2.7.14	Clear Single ISR Bit	2-23
2.7.15	Load Mode Bits M0 through M4	2-23
2.7.16	Control Mode Bits M5, M6 and M7	2-23
2.7.17	Preselect IMR for Writing	2-24
2.7.18	Preselect ACR for Writing	2-24
2.7.19	Preselect Vector Address Memory for Writing	2-24
2.7.20	Coding B2, B1, B0 Field Commands	2-25
CHAPTER 3	**CONFIGURATION**	
3.1	GENERAL DESCRIPTION	3-1
3.2	FACTORY CONFIGURATION	3-1
3.3	DEVICE ADDRESSES	3-1
3.4	DEVICE ADDRESS JUMPERS	3-3
3.5	INTERRUPT VECTOR ADDRESSES	3-5
CHAPTER 4	**INTERFACING**	
4.1	INTERFACE CONNECTORS	4-1
4.2	INPUT/OUTPUT SIGNAL FUNCTIONS	4-1
4.3	INPUT/OUTPUT SIGNAL ASSERTION LEVELS	4-1
4.4	INPUT/OUTPUT SIGNAL LOOPBACK CONNECTIONS	4-5
4.5	INTERFACE CABLE	4-5
4.6	INPUT/OUTPUT FUNCTION TIMING	4-5
4.7	INPUT DATA OPERATION	4-9
4.8	OUTPUT DATA OPERATION	4-9
4.9	INTERRUPT OPERATION	4-9
CHAPTER 5	**PROGRAMMING EXAMPLES**	
5.1	GENERAL DESCRIPTION	5-1
5.2	PROGRAMMED DATA TRANSFER WITHOUT HANDSHAKING	5-1
5.3	PROGRAMMED DATA TRANSFER WITH HANDSHAKING	5-1
5.4	INTERRUPT-DRIVEN TRANSFER	5-3
CHAPTER 6	**OPTIC ISOLATOR INTERFACE EXAMPLE**	

FIGURES

Figure No.	Title	Page
2-1	DRV11-J Block Diagram	2-2
2-2	CSRA Bit Assignments	2-3
2-3	CSRB Bit Assignments	2-4
2-4	CSRC Bit Assignments	2-5
2-5	CSRD Bit Assignments	2-6
2-6	Data Buffer Register Bit Assignments	2-7
2-7	Group 1 and Group 2 Interrupt Controller Interconnections	2-8
2-8	Intergroup Priority Resolution Timing	2-10
2-9	Interrupt Controller Block Diagram	2-11
2-10	CSRA and CSRC Status Registers' Bit Assignments	2-14
2-11	Mode Register Bit Assignments	2-15
2-12	DRV11-J Vector Address Format	2-17
2-13	Rotating Priority Mode	2-18
3-1	DRV11-J Jumper Locations	3-2
3-2	DRV11-J Device Address Format	3-5
4-1	DRV11-J I/O Connector Pin Locations	4-2
4-2	I/O Bus Interface, Simplified Schematic	4-4
4-3	DRV11-J I/O Function Timing	4-7
4-4	Input Data Transfer Sequence	4-10
4-5	Output Data Transfer Sequence	4-11
4-6	Interrupt Sequence	4-12
5-1	Example of a Programmed Data Transfer without Handshaking	5-2
5-2	Example of a Programmed Data Transfer with Handshaking	5-3
5-3	Example of an Interrupt-Driven Output Program	5-4
5-4	Example of an Interrupt-Driven Input Program	5-5
6-1	Example of an Optic Isolator Interface	6-2

TABLES

Table No.	Title	Page
1-1	DRV11-J Module Pin Assignment	1-4
2-1	CSRA Bit Functions and Descriptions	2-3
2-2	CSRB Bit Functions and Descriptions	2-4
2-3	CSRC Bit Functions and Descriptions	2-5
2-4	CSRD Bit Functions and Descriptions	2-6
2-5	Summary of Data Bus Transfers	2-12
2-6	Interrupt Control Register and Memory Summary	2-13
2-7	Fixed Priority Mode	2-17
2-8	Vector Address Memory Field Coding	2-25
2-9	Command Register B2, B1, B0 Field Coding	2-25

TABLES (Cont)

Table No.	Title	Page
2-10	DRV11-J Command Code Summary	2-26
3-1	DRV11-J Factory Jumper Configuration	3-3
3-2	DRV11-J Jumper Functions	3-4
3-3	DRV11-J Registers	3-4
4-1	I/O Connector Pin Assignments	4-2
4-2	I/O Signal Functions	4-5
4-3	DRV11-J Loopback Signal Connections	4-6
4-4	I/O Function Timing Tolerance	4-8

CHAPTER 1
INTRODUCTION

1.1 GENERAL DESCRIPTION

The DRV11-J is a double-height parallel line interface module designed for use in LSI-11 microcomputer systems. It contains four programmable ports designated A, B, C and D. Each port contains 16 I/O lines and is capable of transferring a 16-bit word between the LSI-11 bus and the user device(s). Data word transfers in or out of the DRV11-J are accomplished by the assertion of two control signals at each port of the DRV11-J and two control signals asserted by the user device to its respective port. These control signals must be asserted in a protocol sequence while observing timing constraints to ensure an orderly data transfer. The protocol sequence is described in Chapter 2.

The DRV11-J will also accept interrupt requests from up to 16 I/O lines to generate up to 16 individual vector addresses. This interrupt capability for real-time response makes it useful for sensor I/O applications. The DRV11-J may also be used as a general purpose interface to custom devices, or two DRV11-Js may be connected together as a link between two LSI-11 buses.

The DRV11-J contains two programmable mode registers that provide a number of operating modes to customize the module configuration for different system applications. The module may be programmed for use in vectored-interrupt-driven systems or software-polled systems. When used in vectored interrupt systems, the module may be programmed to operate in either a fixed priority or a rotating priority resolution mode. In addition, the module may be programmed to generate either a common vector address or individual vector addresses in response to user device(s) interrupt requests. Additional operating options available under program control include the selection of an active high or active low interrupt request polarity, preselection of internal registers, and the selection of a master mask bit to arm or disarm the interrupt capability of the DRV11-J. All of the operating modes and options are described in detail in Chapter 2.

The DRV11-J also contains two RAMs that are used to store programmed interrupt vector addresses. One 8-bit RAM location is used to store each interrupt vector address. One vector address may be programmed for each of the 16 interrupt request inputs.

1.2 FEATURES

The DRV11-J contains the following features:

- Four 3-state 16-bit parallel I/O ports
- User-assigned device addresses
- Acceptance of up to 16 external interrupt requests
- Programmable interrupt vector addresses
- Program-controlled input/output operations
- Programmable operating modes:

  Interrupt Controller Mode – Interrupt-driven
  Priority Modes – Fixed or Rotating
  Vector Address Selection – Individual or common vector

1.3 DOCUMENTATION

In addition to this user's guide, refer to the *Field Maintenance Print Set*, MP00866, for information on the DRV11-J module.

1.4 DIAGNOSTIC SOFTWARE

Diagnostic software is available for troubleshooting, fault isolation, and verification at both the module level and system level. Two diagnostics are required for testing at the module level and these must be run in sequence. A DECX11 module diagnostic is required to test the module at the system level. Turnaround cable BC05W-02 must be installed with a half twist to J1 and J2 when running the module- and system-level diagnostics. The diagnostic software is designated as follows:

- CVDRCAO Part 1
- CVDRDAO Part 2
- DECX11 Module CXDRJAO

1.5 SPECIFICATIONS

The following defines the physical, electrical and environmental specifications for the DRV11-J module.

1.5.1 Physical Specifications

Identification	M8049
Size	Double-height 22.8 cm × 13.2 cm (8.9 in × 5.2 in)

1.5.2 Electrical Specifications

Power	+5 Vdc ± 5% @ 1.8 A (maximum), 1.6 A (typical)
Bus Loads	ac 2 dc 1

I/O Signal Electrical Parameters:

Data Buffer 3-State Outputs V(OL) = 0.5 V @ I(OL) = 8 mA V(OL) = 0.4 V @ I(OL) = 4 mA V(OH) = 2.4 V @ I(OH) = −2.6 mA	Data Buffer Inputs I(IL) = −0.2 mA @ V(IL) = 0.4 V V(IH) = 20 µA @ V(IH) = 2.7 V
Protocol Signal 3-State Outputs V(OL) = 0.55 V @ I(OL) = 64 mA V(OH) = 2.4 V @ I(OH) = −15 mA	Protocol Signal Inputs Termination = 120 Ω I(IL) = −27 mA @ V(IL) = 0.5 V I(IH) = 80 µA @ V(IH) = 2.7 V

1.5.3 Environmental Specifications

The DRV11-J module may be operated or stored in the following environmental conditions.

1.5.3.1 Operating and Storage Temperature Ranges

Operating range: 5° to 60° C (41° to 140° F)

Storage range: –40° to 66° C (–40° to 150° F)

If the module is not within its operating temperature range, move it to an area within the range and allow it to stabilize for a minimum of five minutes before operating. Also, derate the maximum operating temperature by 1° C (1.8 ° F) for each 305 m (1000 ft) of altitude above 2440 m (8000 ft).

1.5.3.2 Relative Humidity

Storage: 10% to 90%, noncondensing

Operating: 10% to 90%, noncondensing

1.5.3.3 Airflow during Operation – Provide adequate airflow to limit the inlet-to-outlet temperature rise across the module to 5° C (9° F) when the inlet temperature is 60° C (140° F). For operation below 55° C (131° F), limit that rise to 10° C (18° F) maximum.

1.5.3.4 Altitude

Storage: The module will not be mechanically or electrically damaged at altitudes up to 15,240 m (50,000 ft), 90 mm mercury.

Operating: Up to 15,240 m (50,000 ft), 90 mm mercury. Note: Derate the maximum operating temperature by 1° C (1.8° F) for each 305 m (1000 ft) of altitude above 2440 m (8000 ft).

1.6 INSTALLATION

The DRV11-J is a bus request level 4 module and must be installed in an LSI-11 backplane dual-option slot following the rules for position-dependent interrupt priority configurations. In position-dependent configurations, peripheral devices with the highest priority must be installed closest to the processor and the remaining devices placed in the backplane in decreasing order of priority, with the lowest priority module farthest from the processor.

Before installing the module(s) in the backplane, check that the proper device address jumpers are installed. Three standard LSI-11 bus addresses are reserved for the DRV11-Js. If the application requires more than three DRV11-Js, the additional modules must be assigned addresses located in the user-reserved address space. Chapter 3 describes the address configuration procedure. The standard factory jumper configuration is described in Table 3-1, and Figure 3-2 shows the device address format.

CAUTION

DC power must not be applied to the backplane when installing or removing modules.

The DRV11-J's functionality must be proved after installation by performing an acceptance test. The acceptance test consists first of running the basic system diagnostics and then running the DRV11-J module-level diagnostics listed in Paragraph 1.4.

Module Pin Assignments

The DRV11-J module pin assignments are described in Table 1-1.

Table 1-1 DRV11-J Module Pin Assignment

Connector A			Connector B		
Side 1 Signal	Pin	Side 2 Signal	Side 1 Signal	Pin	Side 2 Signal
BIRQ 5 L	A	+5 V	NC	A	−5 V
BIRQ 6 L	B	NC	NC	B	NC
NC	C	GND	NC	C	GND
NC	D	NC	NC	D	NC
NC	E	BDOUT L	NC	E	BDAL 2 L
NC	F	BRPLY L	NC	F	BDAL 3 L
NC	H	BDIN L	NC	H	BDAL 4 L
NC	J	BSYNC L	NC	J	BDAL 5 L
NC	K	BWTBT L	NC	K	BDAL 6 L
NC	L	BIRQ 4	NC	L	BDAL 7 L
NC	M	BIAKI L	NC	M	BDAL 8 L
NC	N	BIAKO L	NC	N	BDAL 9 L
NC	P	BBS 7 L	BIRQ 7	P	BDAL 10 L
NC	R	BDMGI L	NC	R	BDAL 11 L
NC	S	BDMGO L	NC	S	BDAL 12 L
GND	T	BINIT L	GND	T	BDAL 13 L
NC	U	BDALO L	NC	U	BDAL 14 L
NC	V	BDALI L	NC	V	BDAL 15 L

NOTE:

1. Connector A, pin A, side 1 corresponds to bus pin AA1.
2. NC = no connection.

CHAPTER 2
FUNCTIONAL DESCRIPTION

2.1 GENERAL DESCRIPTION

The DRV11-J contains the logic necessary to provide communication between the LSI-11 bus and up to four user devices in 16 bit word lengths via four I/O ports. Four control lines associated with each of the four ports ensure orderly information transfers. Word transfers are executed by programmed I/O operations or interrupt-driven routines. Write data is output by the DRV11-J to the I/O bus through 3-state data latches, and read data is input through unlatched bus buffers. Figure 2-1 shows the main logic functions performed by the DRV11-J module.

All control/status and I/O data transfers take place over a bidirectional internal bus (TSD <15:00>) on the DRV11-J. The module contains four I/O buses, one for each port (A, B, C and D). Each port has an associated control/status register (CSRA, CSRB, CSRC or CSRD) that contains status information when read and command words when written. All ports have 16 bidirectional 3-state lines and perform controlled input/output operations. Note that port A is the only port that will perform bit interrupt functions in addition to input/output data transfers. The 16 external interrupt requests are functionally divided into two groups of eight lines, referred to as group 1 and group 2.

2.2 CONTROL/STATUS REGISTERS

The control/status registers (CSRA, CSRB, CSRC and CSRD) are read/write byte-addressable registers with bit assignments as shown in Figures 2-2, 2-3, 2-4 and 2-5. The function and description of the control/status register bits are described in Tables 2-1, 2-2, 2-3 and 2-4.

2.3 DATA BUFFER REGISTERS

The four data buffer registers (DBRA, DBRB, DBRC and DBRD) are 16-bit word-addressable registers. They are used as latched output data buffers when the DRV11-J is in output mode (write) and as unlatched bus buffers in input mode (read). The contents of the output data buffers may be examined while the DRV11-J is in an output mode by performing a read operation of the input data buffers. This ability to examine the output data buffers in the output mode provides software access to the internal conditions of the DRV11-J.

The latched output data buffer registers DRBA through DBRD are not cleared by BINIT. The bit assignment is the same for all registers and is shown in Figure 2-6.

2.4 INTERRUPT CONTROL

The DRV11-J is capable of monitoring 16 lines to generate 16 vectored interrupts. The interrupt control is performed by a DC003 interrupt logic chip and interrupt controller chips. A functional description of the signals required to initiate interrupts and the DRV11-J registers used for programming, reading and writing the internal registers of the interrupt controllers is given in Paragraph 2.4.1. An operating description of the interrupt controllers is given in Paragraph 2.4.2, and the internal registers of the interrupt controllers are described in Paragraph 2.5

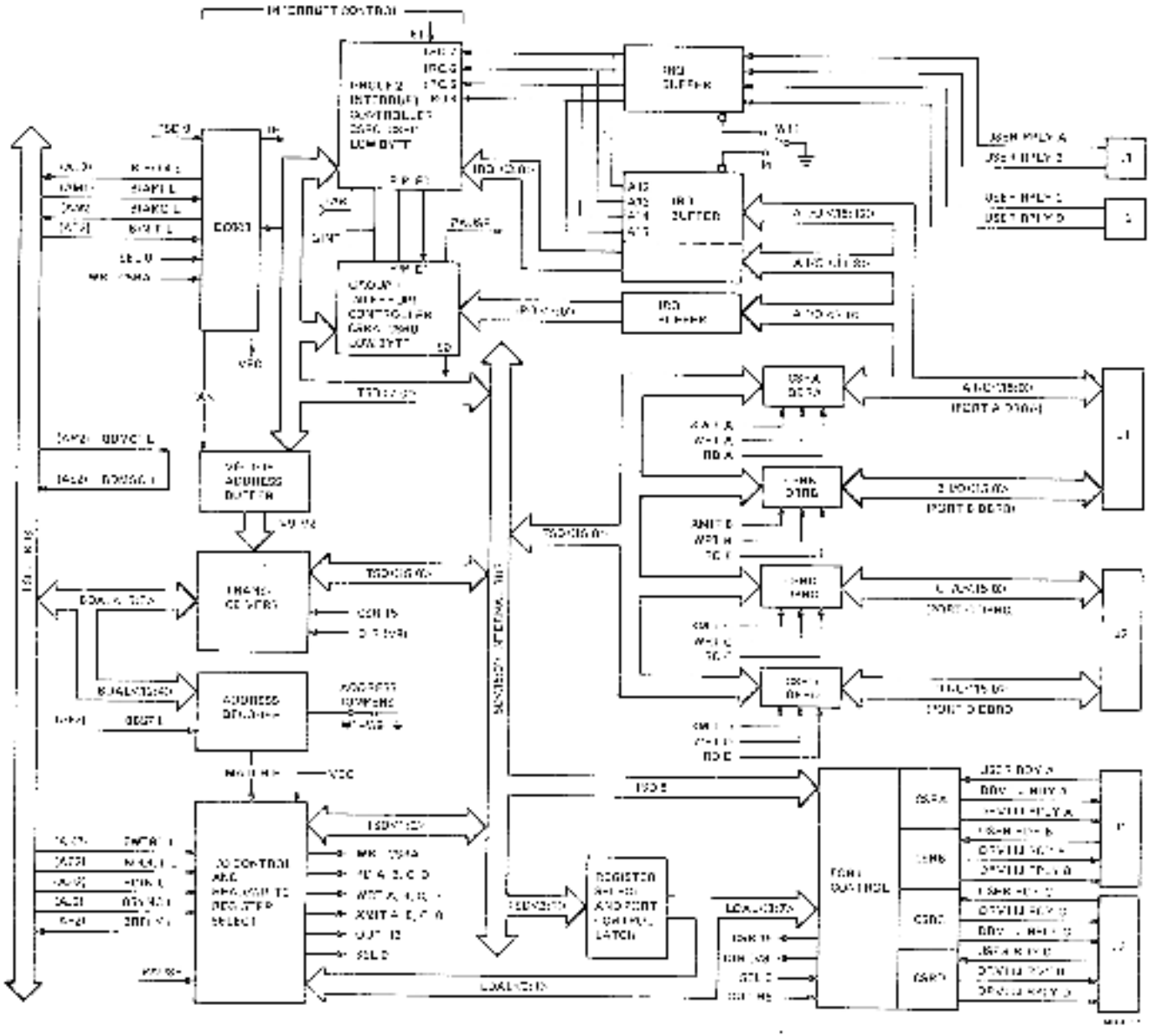
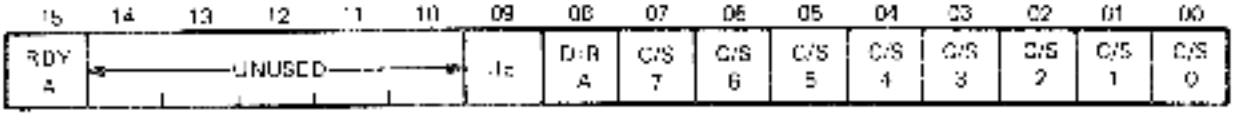


Figure 2-1 IDRV11-J Block Diagram



MR-4910

Figure 2-2 CSRA Bit Assignments

Table 2-1 CSRA Bit Functions and Descriptions

Bit	Name	Function	Description
07:00	C/S7–C/S0	Read/Write	These bits are used in conjunction with CSRB bits<07:00> to program interrupt control group 1. They contain status information when read and command words when written. Unaffected by BINIT. (See Paragraphs 2.4.5.1 and 2.4.5.2 for status and command definitions.)
08	DIR A	Read/Write	DIRECTION A. Used for controlling DBRA. This bit, in conjunction with the USER RDY signal, controls the direction of data transfer. When the DIR bit is cleared, the DRV11-J RDY output signal is asserted and the DRV11-J is the input device. When this bit is set and the USER RDY signal is asserted, the DRV11-J is the output device. The negation of either DIR or USER RDY causes the DRV11-J outputs to remain in their high-impedance state. Cleared by BINIT.
09	IE	Read/Write	INTERRUPT ENABLE. Enables the DRV11-J to generate processor interrupts when set. Used to enable both group 1 and group 2 interrupts. Cleared by BINIT.
14:10			Unused. Read as 0s.
15	RDY A	Read Only	USER READY A. Used for controlling DBRA. When read, this bit yields the state of the USER RDY signal. A 0 means negated and a 1 means asserted. This bit is used in conjunction with the DIR bit to enable DRV11-J output operations. The user device asserts this signal when it desires the DRV11-J to output data. Unaffected by BINIT.

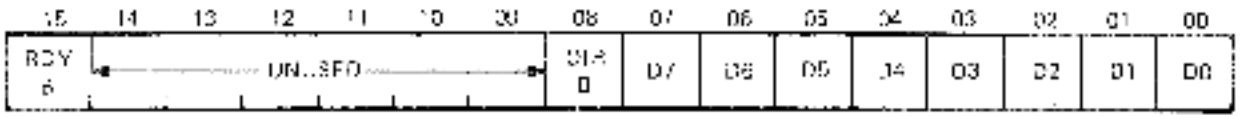


Figure 2-2 CSRB Bit Assignments

Table 2-2 CSRB Bit Functions and Descriptions

Bit	Name	Function	Description
07:00	D7 D0	Read/Write	These bits are used in conjunction with CSRA bits <07:00> to program interrupt control group 1. They contain information selected by the command word loaded through CSRA. The registers available are the IRR, ISR, ACR, IMR and the vector address memory. Unaffected by BINIT. (See Paragraphs 2.4.5.4 through 2.4.5.8 for a detailed description of the registers and their functions.)
08	DIR B	Read/Write	DIRECTION B. Used for controlling DBRB. This bit, in conjunction with the USER RDY signal, controls the direction of data transfer. When the DIR bit is cleared, the DRV11-J RDY output signal is asserted and the DRV11-J is the input device. When this bit is set and the USER RDY signal is asserted, the DRV11-J is the output device. The negation of either DIR or USER RDY causes the DRV11-J outputs to remain in their high-impedance state. Cleared by BINIT.
14:09			Unused. Read as 0s.
15	RDY B	Read Only	USER READY B. Used for controlling DBRB. When read, this bit yields the state of the USER RDY signal. A 0 means negated and a 1 means asserted. This bit is used in conjunction with the DIR bit to enable DRV11-J output operations. The user device asserts this signal when it desires the DRV11-J to output data. Unaffected by BINIT.

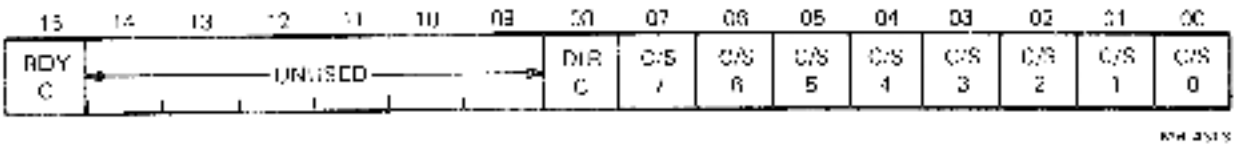


Figure 2-4 CSRC Bit Assignments

Table 2-3 CSRC Bit Functions and Descriptions

Bit	Name	Function	Description
07:00	C/S7–C/S0	Read/Write	These bits are used in conjunction with CSRD bits <07:00> to program interrupt control group 2. They contain status information when read and command words when written. Unaffected by BINIT. (See Paragraphs 2.4.5.1 and 2.4.5.2 for status and command definitions.)
08	DIR C	Read/Write	DIRECTION C. Used for controlling DBRC. This bit, in conjunction with the USER RDY signal, controls the direction of data transfer. When the DIR bit is cleared, the DRV11-J RDY output signal is asserted and the DRV11-J is the input device. When this bit is set and the USER RDY signal is asserted, the DRV11-J is the output device. The negation of either DIR or USER RDY causes the DRV11-J outputs to remain in their high-impedance state. Cleared by BINIT.
14:09			Unused. Read as 0s.
15	RDY C	Read Only	USER READY C. Used for controlling DBRC. When read, this bit yields the state of the USER RDY signal. A 0 means negated and a 1 means asserted. This bit is used in conjunction with the DIR bit to enable DRV11-J output operations. The user device asserts this signal when it desires the DRV11-J to output data. Unaffected by BINIT.

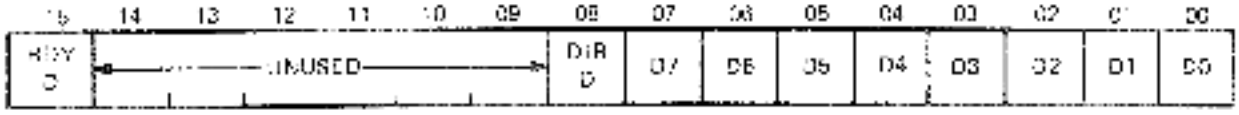


Figure 2-5 CSRD Bit Assignments

Table 2-4 CSRD Bit Functions and Descriptions

Bit	Name	Function	Description
07:00	D7-D0	Read/Write	These bits are used in conjunction with CSRC bits <07:00> to program interrupt control group 2. They contain information selected by the command word loaded through CSRC. The registers available are the IRR, ISR, ACR, IMR and the vector address memory. (See Paragraphs 2.4.5.4 through 2.4.5.8 for a detailed description of the registers and their functions.)
08	DIR D	Read/Write	DIRECTION D. Used for controlling DBRD. This bit, in conjunction with the USER RDY signal, controls the direction of data transfer. When the DIR bit is cleared, the DRV11-J RDY output signal is asserted and the DRV11-J is the input device. When this bit is set and the USER RDY signal is asserted, the DRV11-J is the output device. The negation of either DIR or USER RDY causes the DRV11-J outputs to remain in their high impedance state. Cleared by BINIT.
14:09			Unused. Read as 0s
15	RDY D	Read Only	USER READY D. Used for controlling DBRD. When read, this bit yields the state of the USER RDY signal. A 0 means negated and a 1 means asserted. This bit is used in conjunction with the DIR bit to enable DRV11-J output operations. The user device asserts this signal when it desires the DRV11-J to output data. Unaffected by BINIT.

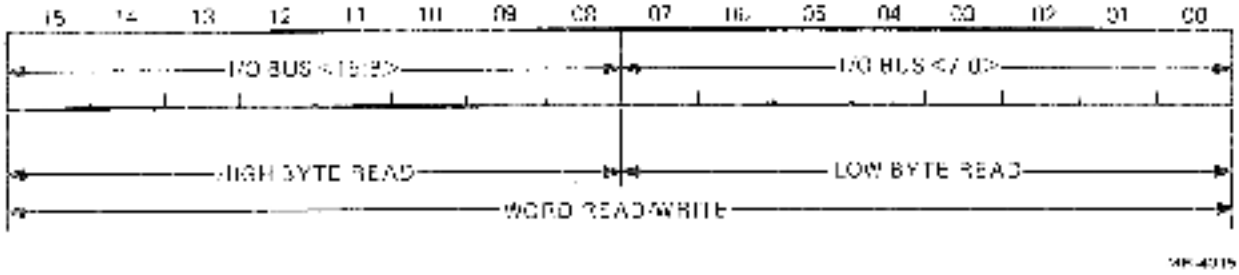


Figure 2-6 Data Buffer Register Bit Assignments

2.4.1 Functional Description

The interrupt control logic shown in Figure 2-1 consists primarily of a DC0DH interrupt logic chip and two interrupt controller chips. Five LSI-11 bus control signals (BIRQ4 L, BIAK1 L, BIAKO L, BDIN L and BINIT L) are used by the interrupt control logic for initialization, sending interrupt requests to the processor, receiving the interrupt acknowledge signal from the processor, and sending the vector address to the processor.

Each interrupt controller chip is responsible for monitoring a group of eight interrupt request inputs. Each group of eight interrupt requests is applied via IRQ buffers to an 8-bit interrupt request register (IRR) in the interrupt controller.

The two interrupt controllers (group 1 and group 2) are programmed independently. The group 1 interrupt controller is programmed through the low bytes of CSRA and CSRB while the group 2 interrupt controller is programmed through the low bytes of CSRC and CSRD. The only commonalities of the two groups are priority resolution and the interrupt enable (IE) CSRA bit 9. Both interrupt controllers must operate in the same mode, either interrupt or polled. Each interrupt controller contains an 8-bit interrupt mask register (IMR) that may be used to disable the processing of any undesired interrupt requests.

The group 1 interrupt controller has the higher priority and its enable output is connected to the enable input of group 2. Group 1 must be armed to accept interrupts with the master mask bit set in the mode register. When group 1 is armed, its enable output goes high, thus enabling group 2 interrupts. Therefore, whenever the interrupt mode is selected, group 1 must be armed, even if none of the group 1 interrupt requests are being used in order to pass the enable signal along to group 2.

Group 1 and group 2 may be programmed to respond to either an active high or an active low transition on the interrupt request lines. A bit in the interrupt request register (IRR) is set whenever the corresponding interrupt request line makes an inactive-to-active transition and meets the active pulse width requirements. Active pulse widths 270 ns or greater will set the corresponding IRR bit, while pulse widths 30 ns or less are ignored. Active pulse widths between 30 ns and 270 ns may or may not set the IRR bit.

2.4.2 Interrupt Controller Interface

The interconnections between the group 1 and group 2 interrupt controllers, their relation to the DRV11-J A I/O bus and the LSI-11 bus are shown in Figure 2-7. Latched data address line LDAL 3 L or H, along with the SEL 0 L signal, is used to select group 1 for subsequent reading/writing through the low byte of CSRA or CSRB, or group 2 for reading/writing through CSRC or CSRD. Intergroup priority management is controlled by the enable-in (EI), enable-out (EO) and the response-in-progress (RIP) signals. Note that the IAK 1, GINT, RIP, and PAUSE lines are respectively tied together. Group 1 is always enabled because its enable-in (EI) pin is floating high. The enable-out (EO) signal of group 1 is connected to the enable-in (EI) pin of group 2.

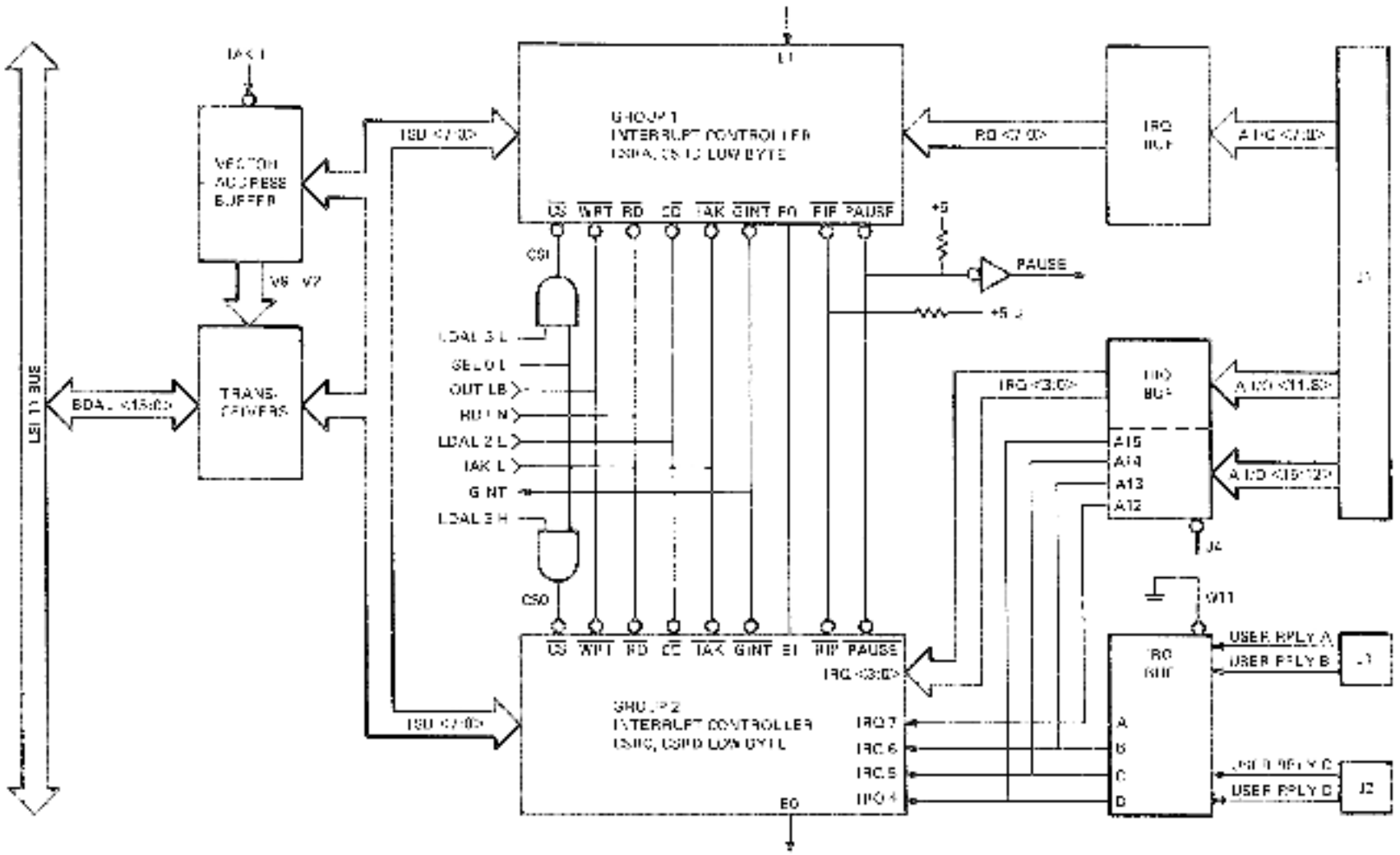


Figure 2-7 Group 1 and Group 2 Interrupt Controller Interconnections

Each interrupt controller group accepts eight IRQ inputs through the IRQ buffers. The timing relationship of the signals involved in intergroup priority resolution is shown in Figure 2-8. For purposes of this discussion, suppose that an active interrupt (IRQ 7) arrives at group 1. When IRQ 7 is applied to group 1, a group interrupt (GINT) will be generated if the request is not masked or the master mask bit has not disarmed the interrupt controller. The GINT signal will generate BIRQ 4 L, if the processor has enabled interrupts, by setting the interrupt enable bit. The processor will accept BIRQ 4 L after executing the current instruction, issue BIAK L, and disable its internal interrupt structure. When the processor returns the BIAK L signal, EO of group 1 goes low, PAUSE goes low to indicate that a data bus transfer operation is presently under way. The rising edge of PAUSE extends the IAK L pulse and is also ANDed with the RPLY signal of the I/O control logic to delay the assertion of BRPLY until the current data transfer is completed.

After the fall of BIAK L, group 1 and group 2 wait until a brief internal delay elapses and then examine EI. If EI is low, internal activity is suspended until EI goes high. If EI is high, the internal circuitry is checked to see if an unmasked interrupt request is pending. In this example, EI of group 1 is always high and EO stays low after the brief internal delay because of IRQ 7. The low EO signal of group 1 therefore disables group 2. The group 1 RIP signal is brought low, and PAUSE is brought high, causing the IAK signal to go high. When the IAK signal goes high, the vector address programmed for IRQ 7 is output through the vector address buffers and transceivers to the LSI-11 bus. Note that the PAUSE output automatically adjusts the position of its rising edge to accommodate the particular intergroup and intragroup priority resolution conditions that occur for each IAK cycle.

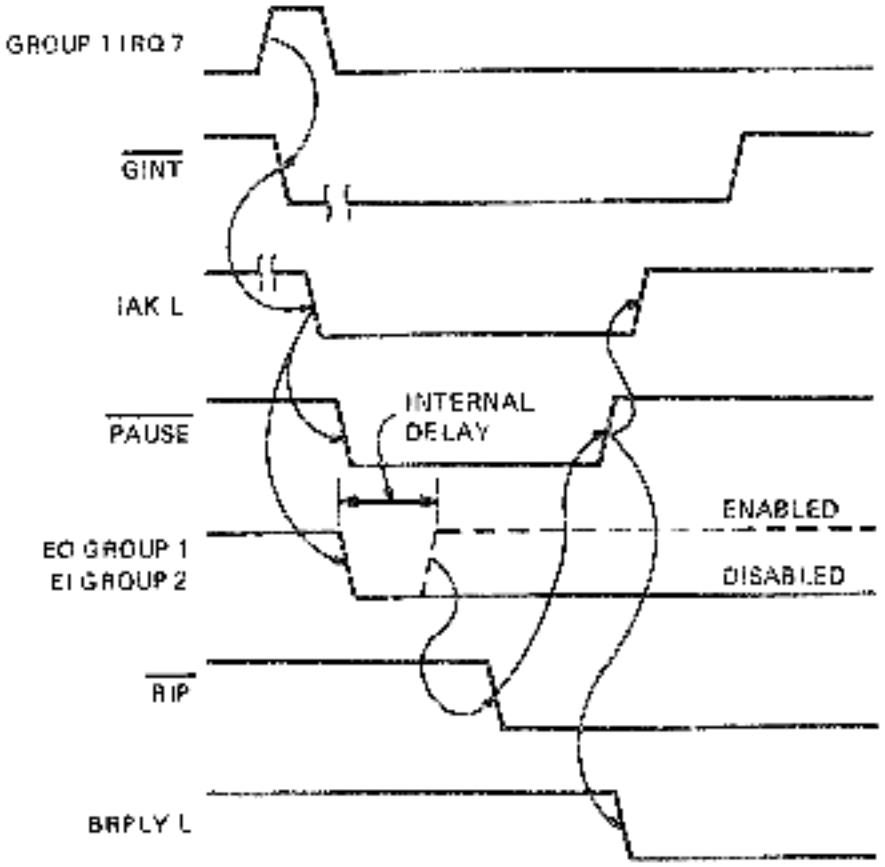
The RIP output serves two basic functions within the interrupt system. First, its falling edge informs the other interrupt controller that an interrupt request has been selected and PAUSE may therefore be released. Second, as long as RIP is low, only the interrupt controller that is causing RIP to go low is allowed to respond to IAK L inputs. RIP stays low until the vector address for the selected interrupt has been transferred. Suppose that a new interrupt request arrives at IRQ 0 of the group 2 interrupt controller during the time the vector address of group 1 is being transferred. Without the RIP signal there would be confusion when IRQ 0 arrives at the group 2 interrupt controller. The group 2 interrupt controller treats RIP as an input, and therefore, will not respond to IRQ 0 until RIP goes high.

2.4.3 Interrupt Controller Operating Description

The block diagram Figure 2-9 shows the registers, interface signals and basic information flow of an interrupt controller chip. The interrupt controller chips for group 1 and group 2 are identical and the following description applies to both. Interrupt requests (IRQ <7:0>) are captured and latched in the interrupt request register (IRR). Any requests not masked by the interrupt mask register (IMR) will cause a group interrupt (GINT) output to the processor if the interrupt controller is enabled, armed, and IE (CSRA) bit 9 is set. When the processor is ready to accept the interrupt, it issues an IAK L pulse that initiates two operations. First, the priority of pending interrupts is resolved, and second, the vector address associated with the highest priority interrupt is transferred from the vector address memory to the data bus (TSD <7:0>).

Other interrupt management functions are controlled by the auto clear register (ACR), the interrupt service register (ISR), and the mode register (MR). The command register is used by the processor to exercise control over the many functions provided by the DRV11-J, while the status register reports on the internal condition of the DRV11-J.

The interrupt controller is addressed by the processor as either a control port or a data port through use of the LDAL 2 bit. The control port provides direct access to the command register and the status register. The data port is used to communicate with all other internal locations.



NOTE:
EI OF GROUP 1 IS OPEN AND ALWAYS ENABLED.

Figure 2-8 Intergroup Priority Resolution Timing

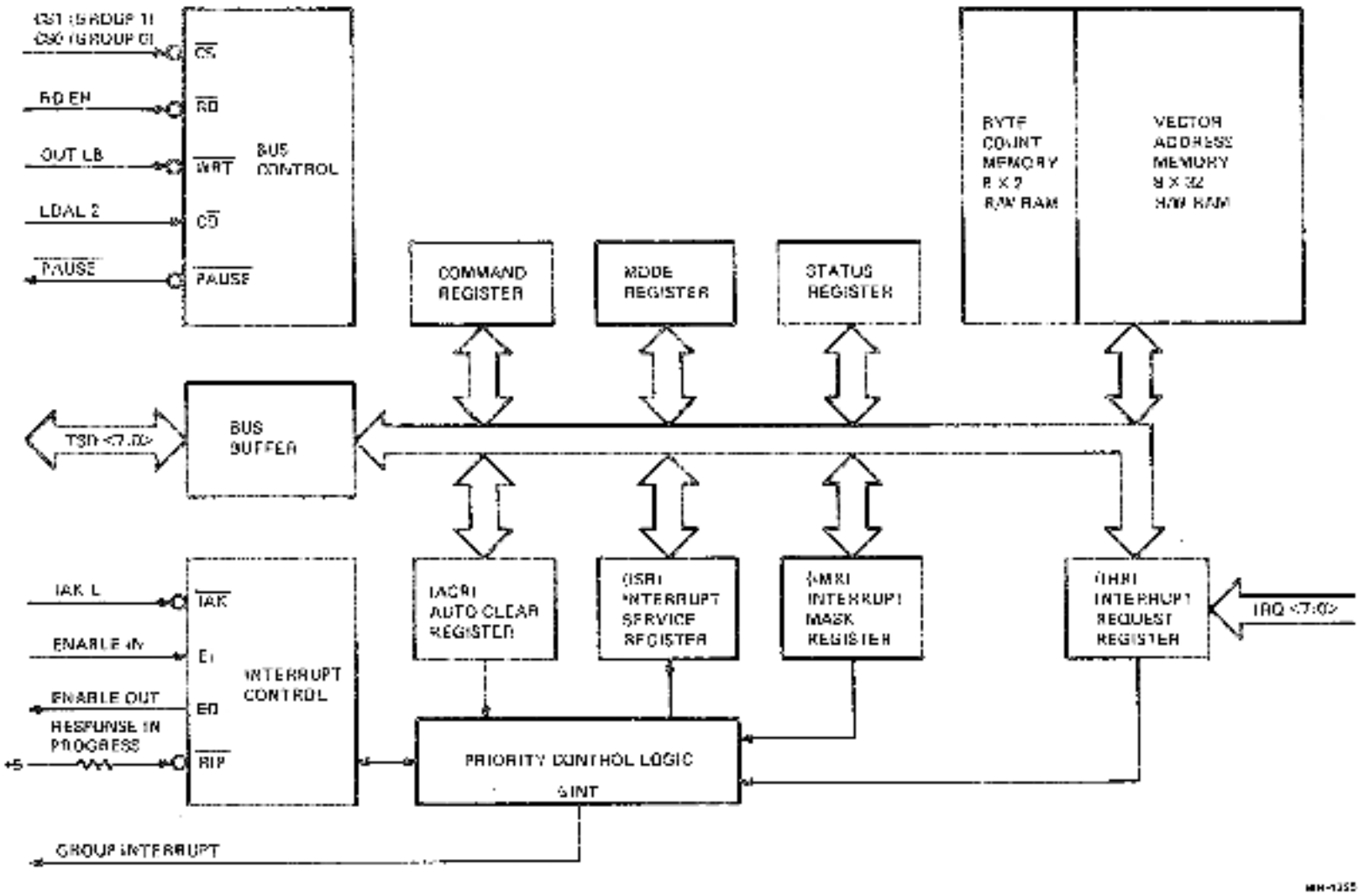


Figure 2-9 Interrupt Controller Block Diagram

Information is transferred through the interrupt controllers, the DRV11-J I/O bus, and the LSI-11 bus by the eight 3-state bidirectional data bus lines (TSD <7:0>). Control signal configurations for all information transfer operations are described in Table 2-5. The following conventions are assumed: RD EN and OUT LB are mutually exclusive; RD EN, OUT LB, and LDAL 2 have no meaning unless CS1 or CS0 is low; active IAK L pulses occur only when CS1 or CS0 is high.

Table 2-5 Summary of Data Bus Transfers

Control Input					TSD <7:0> Data Bus Operation
CS1 CS0	LDAL 2	RD EN	OUT LB	IAK L	
0	0	0	1	1	Transfer contents of preselected data register to data bus (read)
0	0	1	0	1	Transfer contents of data bus to preselected data register (write).
0	1	0	1	1	Transfer contents of status register to data bus (read)
0	1	1	0	1	Transfer contents of data bus to command register (write).
1	X	X	X	0	Transfer contents of selected vector address memory location to data bus (read).
1	X	X	X	1	No information transferred

NOTE: X = "don't care" condition; LDAL = 1 = control port, 0 = data port.

The status register is selected directly for reading by the LDAL control input. Other internal registers are read by preselecting the desired register with mode bits 5 and 6, and then executing a data read. The vector address memory can be read only with IAK L pulses.

The command register is selected directly for writing by the LDAL 2 control input. The mask and auto-clear registers are loaded following specific commands to that effect. To load each level (IRQ <7:0>) of the vector address memory, the vector address memory preselect command is issued to select the desired level. A data-write operation is then executed to load that level.

2.4.4 Interrupt Control Reset

The DRV11-J does not include an external hardware reset input for the interrupt control. The reset function is accomplished by software command, or automatically during power-up. The processor may initiate a reset at any time by writing all 0s into the command register of each interrupt controller. Power-up reset circuitry on each interrupt controller integrated circuit is internally triggered by the rising V_{CC} voltage (IC supply voltage, 5 V) to generate a brief reset pulse when the predetermined threshold is reached. The interrupt controllers are unaffected by a BINIT on the LSI-11 bus.

The vector address memory and byte count register contents are not affected by a software reset, but their contents are unpredictable after a power-up reset. Therefore, if the vector address memory and byte count register are to be used, they must be initialized by the processor after power-up.

The interrupt mask register is set to all 1s by either a software reset or a power-up reset, thus disabling recognition of interrupts by the DRV11-J. The status registers continue to reflect the internal condition of group 1 and group 2 and are not otherwise affected by a reset.

The mode registers are cleared to all 0s to provide the DRV11-J with a reasonable operating environment after a power-up or software reset. The mode registers after reset are assigned the following operating options.

Interrupt mode
Individual vectoring
Fixed interrupt priority
IRQ polarity active low
ISR preselected for reading
Interrupt controllers disarmed by master mask bit

2.4.5 Interrupt Control Register Description

The DRV11-J uses the control and operation registers, plus the vector address memories of the interrupt controllers, to perform and manage its many functions. Table 2-6 lists these elements and summarizes their size and number.

Table 2-6 Interrupt Control Register and Memory Summary

Description	Register Abbreviation	Bit Size Per Register	Quantity Per DRV11-J
Interrupt request register	IRR	8	2
Interrupt service register	ISR	8	2
Interrupt mask register	IMR	8	2
Auto-clear register	ACR	8	2
Status register	–	8	2
Mode register	–	8	2
Command register	–	8	2
Byte count	–	2	16
Vector address memory	–	8 × 32*	16

*Although each interrupt controller contains 32 vector address memory locations of 8 bits each, the DRV11-J uses only 8 of these memory locations.

2.4.5.1 Status Register – Each status register is eight bits wide and contains information describing the internal state of the DRV11-J. The status register is read directly by executing a read operation at CSRA for group 1 or CSRC for group 2. Figure 2-10 shows the status register bit assignments.

The high-order status bit S7 reflects the information state of the group interrupt (GINT) signal. Bit S7 remains valid when interrupts are disabled by the polled mode option, thus permitting the processor to check for interrupts by reading the status register.

Status bit S6 reflects the state of the enable-in (EI) input signal and indicates if group 2 is enabled or disabled. When S6 is high, group 2 can generate an interrupt request. When S6 is low, group 2 interrupt requests are disabled. Group 1 is always enabled.

Status bit S5 reflects the state of the priority mode option as specified by mode register bit M0. When S5 is high, rotating priority is selected. When S5 is low, fixed priority is selected.

Status bit S4 reflects the state of the interrupt mode option as specified by mode register bit 2. When S4 is high, the polled mode is selected and interrupt requests are disabled. When S4 is low, the interrupt mode is selected.

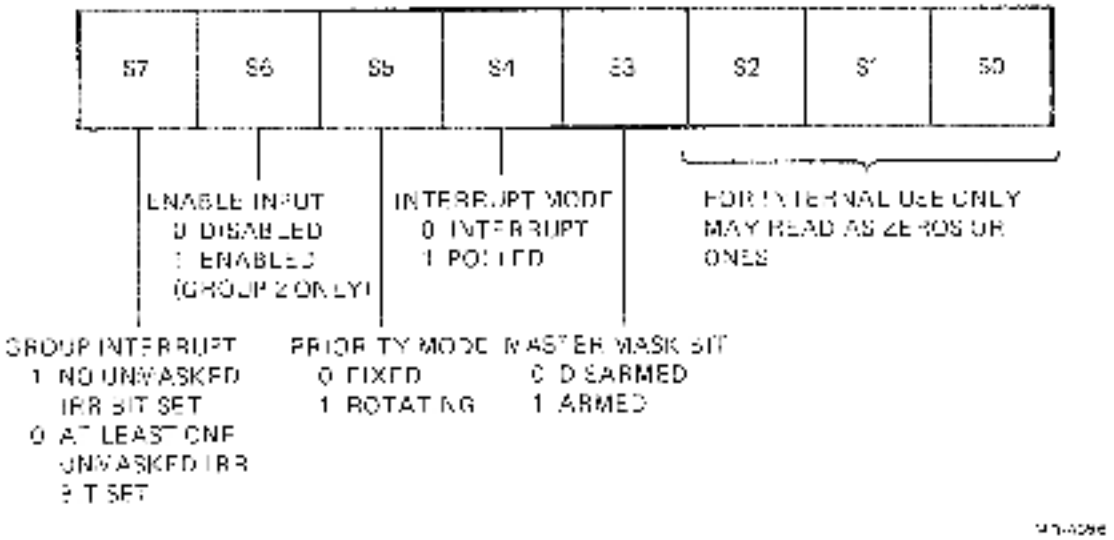


Figure 2-10 CSRA and CSRC Status Registers' Bit Assignments

Status bit S3 reflects the state of the master mask bit as specified by mode register bit M7. When S3 is low, the group is disarmed and IRR bits that are set will not generate interrupt requests. When S3 is high, the group is armed and interrupts can occur.

Status bits S2, S1 and S0 are for internal use by the DRV11-J. These bits may read as zeros or ones and should not be correlated with external events or operational states of the module.

2.4.5.2 Command Register – Each command register is eight bits wide and is used to store the most recently entered command. The register is loaded directly from the data bus by executing a write operation at CSRA for group 1 or CSRC for group 2. Depending on the specific command opcode that is entered, an immediate internal activity may be initiated, or CSRB and CSRD may be preconditioned for subsequent register transfers. The opcodes for each command operation are described in Paragraph 2.7. (The commands are summarized in Table 2-10.)

2.4.5.3 Mode Register – Each mode register is eight bits wide and is used to establish the operating modes and conditions for the many functional features of the DRV11-J. The mode register allows the processor to customize the interrupt system for a particular application. Figure 2-11 shows the mode register bit assignments. No single command or interface operation will load all bits of the mode register in parallel. The five low-order bits (M0 through M4) are loaded in parallel directly from the command register. Mode bits M5, M6 and M7 are controlled by separate commands. The mode register contents cannot be read out on the data bus. However, the conditions of mode bits M0, M2 and M7, which reflect the priority, interrupt and master mask bit modes, are available as part of the status register. The mode register is cleared by a software reset or a power-up reset.

2.4.5.4 Interrupt Request Register (IRR) – Each IRR is eight bits wide and is used to recognize and store active transitions on the eight interrupt request lines. A bit in the IRR is set whenever the corresponding IRQ input line makes an inactive-to-active transition and meets the minimum active

pulse width requirements. Also, the processor (under program control) may set the IRR bits by using two types of commands. This capability permits software-initiated interrupts and is a useful tool for system testing.

All IRR bits are cleared by a reset. Individual IRR bits are cleared automatically when their interrupts are acknowledged by the processor. Four types of commands, in addition to reset, allow the processor to clear IRR bits.

The IRR may be read onto the data bus by preselecting it in mode register bits M5 and M6 with a load mode register command, followed by a read of CRSB <7:0> for group 1 or CSRD <7:0> for group 2.

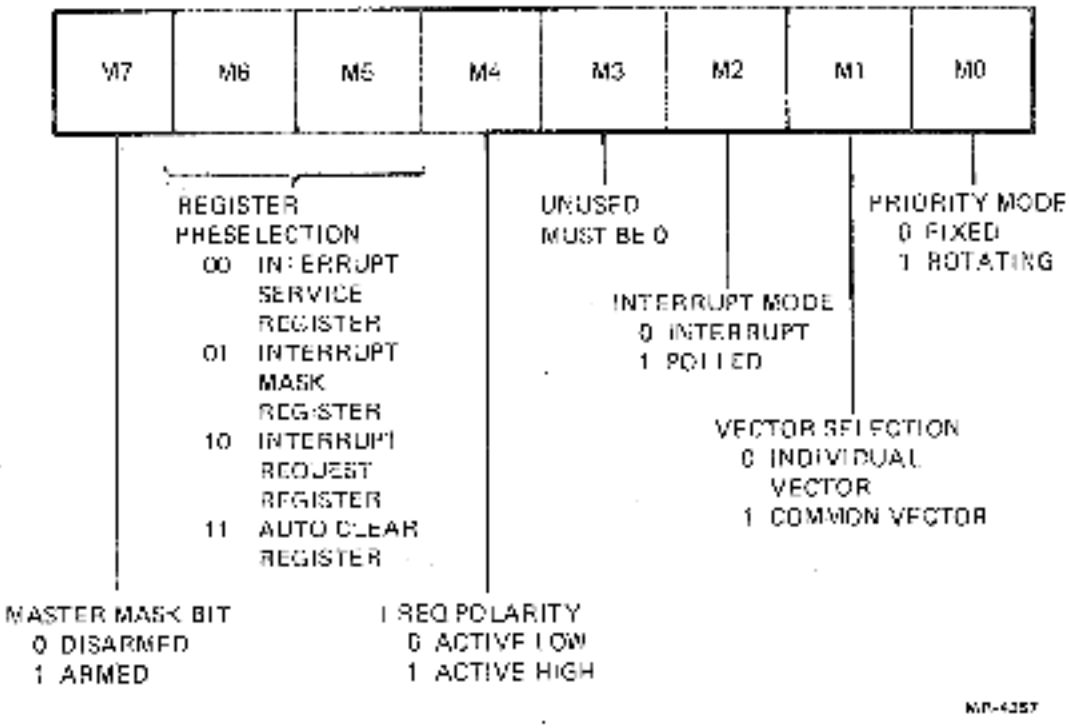


Figure 2-11 Mode Register Bit Assignments

2.4.5.5 Interrupt Service Register (ISR) – Each ISR is eight bits wide and is used to store the acknowledge status of individual interrupts. When the processor acknowledges an interrupt request, the DRV11-J selects the highest priority request that is pending, clears the associated IRR bit, and sets the associated ISR bit. When the ISR bit is programmed for automatic clearing, it is reset by the internal hardware before the end of the acknowledge sequence. When the ISR bit is not programmed for automatic clearing, it must be reset by command from the processor.

The DRV11-J uses the ISR internally to erect a "masking fence." When an ISR bit is set and fixed priority mode is selected, only requests of higher priority will cause a new group interrupt (GINT) output. Thus, requests from lower priority interrupts (and from new requests associated with the set ISR bit) will be "fenced out" and ignored until the ISR bit is cleared. In the rotating priority mode, all requests are fenced out by an ISR bit that is set and no new interrupts will be generated until the ISR bit is cleared. When automatic clearing is specified, no masking fence is erected since the ISR bit is cleared.

If an unmasked interrupt arrives from a device of higher priority than the current ISR, the processor will be interrupted if its interrupt input is enabled. When the new interrupt is acknowledged, the associated higher priority ISR bit is set and the fence moves up to the new priority level. When the new ISR bit is cleared, the fence will then fall back to the previous ISR level. The ISR may be read onto the data bus by preselecting it in mode register bits M5 and M6 with a load mode register command, followed by a read of CSRB <7:0> for group 1 or CSRD <7:0> for group 2.

2.4.5.6 Interrupt Mask Register (IMR) – Each IMR is eight bits wide and is used to enable/disable the processing of individual interrupts. Only unmasked IRR bits can generate an interrupt. The IMR does not otherwise affect the operation of the IRR. An IRR bit that is set while masked will cause an interrupt when its IMR bit is cleared.

All eight IMR bits for each group may be set, cleared, read or loaded in parallel by the processor. In addition, individual IMR bits may be set or cleared by command. This allows a control routine to enable or disable directly an individual interrupt without disturbing the other mask bits and without knowledge of their state or context.

The IMR polarity is active high for masking: a 0 enables the interrupt and a 1 disables it. The power-on reset and the software reset cause all IMR bits to be set, thus disabling all interrupt requests. The IMR may be read onto the data bus by preselecting it in mode register bits M5 and M6 with a load mode register command, followed by a read of CSRB <7:0> for group 1 or CSRD <7:0> for group 2.

2.4.5.7 Auto-Clear Register (ACR) – Each ACR is eight bits wide and specifies the automatic clearing option for each of the ISR bits. When an auto-clear bit is set, the corresponding ISR bit set in an interrupt acknowledge (IAK) cycle is cleared by the internal hardware before the end of the IAK sequence. When an auto-clear bit is not set, the corresponding ISR bit that has been set in an IAK cycle is cleared by a command from the processor.

When selected, the auto-clear option provides two related functional effects. First, it eliminates the need for the associated interrupt service routine to issue a command to clear the ISR bit. Second, it eliminates the masking fence that would otherwise have been erected, allowing lower priority interrupts to cause a new interrupt request.

The ACR is loaded in parallel from the data bus by issuing the ACR load preselect command, followed by a write into the data port. The ACR is read onto the data bus by preselecting it in mode register bits M5 and M6 with a load mode register command, followed by a read of CRSB <7:0> for group 1 or CSRD <7:0> for group 2.

2.4.5.8 Vector Address Memory – The vector addresses are programmed by the vector address memory preselect command, followed by a data-write operation to load the vector address required for each interrupt request level. The vector address memory preselect command is entered directly into the low byte <7:0> of CSRA for group 1 or the low byte <7:0> of CSRC for group 2. Preselect commands entered through CSRA select CSRB for subsequent loading of the vector addresses in group 1. Preselect commands entered through CSRC select CSRD to load the addresses for group 2.

Normally, one vector address is loaded after each preselection command. (Figure 2-12 shows the vector bit positions relative to the loaded byte.) This in turn causes one interrupt to occur for each valid transition on the corresponding IRQ input. Vector addresses are placed on the LSI-11 bus during IAK operation.

Loading the vector address into each new interrupt request level must be preceded by a new vector address memory preselect command. Therefore, 16 preselect commands, each followed by a data-write operation, are required to load 16 vector addresses into the vector address memory.

Note that while the DRV11-J only uses one vector address per interrupt, the interrupt controller chips are capable of handling four vector addresses per interrupt level. To ensure proper operation, the user must always use a byte count of one (BY0 = 0, BY1 = 0) and load only one data byte after each preselect command.

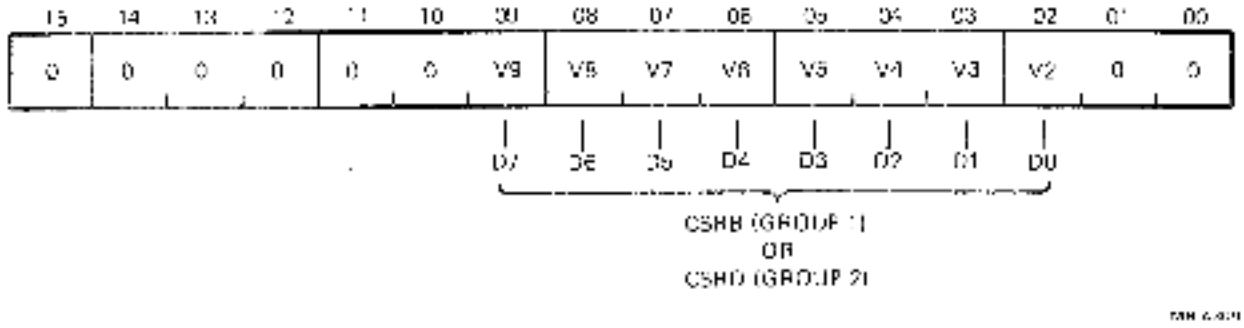


Figure 2-12 DRV11-J Vector Address Format

2.5 OPERATING OPTIONS

The mode register bits are program-controlled to establish the combination of interrupt operating options desired for a particular DRV11-J system application. Refer to Figure 2-11 for the mode register bit assignments. A detailed description of the various options available follows. The master mask bit will affect both group 1 and group 2; all other mode bits affect only their corresponding groups.

2.5.1 Interrupt Priority Mode Selection

Mode register bit M0 specifies either a fixed or rotating priority resolution mode for the DRV11-J. When M0 is low, fixed priority is selected and the eight IRQ inputs for both group 1 and group 2 are assigned a priority based on their physical location at the interface. Group 1 IRQ 0 has the highest priority and group 2 IRQ 7 has the lowest. Table 2-7 lists the priorities assigned to the A I/O <15:0> lines and the USER RPLY lines.

Table 2-7 Fixed Priority Mode

Group	Connection	Level	IRR, ISR, IMR, ACR Bit Assignments		Priority
1	A I/O 0	0	D0		Highest
1	A I/O 1	1	D1		
1	A I/O 2	2	D2		
1	A I/O 3	3	D3	CSRB	
1	A I/O 4	4	D4		
1	A I/O 5	5	D5		
1	A I/O 6	6	D6		
1	A I/O 7	7	D7		
2	A I/O 8	0	D0		
2	A I/O 9	1	D1		
2	A I/O 10	2	D2		
2	A I/O 11	3	D3	CSRD	
2	USER RPLY A	4*	D4		
2	USER RPLY B	5*	D5		
2	USER RPLY C	6*	D6		
2	USER RPLY D	7*	D7		Lowest

*Jumper W11 selects either USER RPLY (A:D) or A I/O <15:12> signals.

Interrupt acknowledge operations are initiated by the processor in response to a group interrupt (GINT) output by the interrupt controllers.

Interrupt priority is resolved after the processor initiates the interrupt acknowledge sequence. When the DRV11-J receives an IAK signal, the interrupt controllers perform priority arbitration to select the highest unmasked pending interrupt, and then output a vector address associated with the selected interrupt request. In the fixed priority mode, therefore, devices with a high priority may be serviced many times before a lower priority device is serviced once. In many systems, this is an appropriate method of servicing the interrupting devices. In those systems where this is not an appropriate method, the interrupt masking capability of the DRV11-J may be used to modify the effective priority structure. This may be accomplished by masking out recently serviced high priority devices, thus permitting recognition of lower priority inputs.

Alternatively, the rotating priority mode may be selected for use in systems where the eight interrupts of each group have similar priority and bandwidth requirements. Mode register bit M0 = 1 selects the rotating priority mode. As shown in Figure 2-13, the relative priorities remain the same as in the fixed mode. In the rotating priority mode, however, the lowest priority position in the circular chain is assigned by the hardware to the most recently serviced interrupt. Priority rotation occurs only within a given group and priority between group 1 and group 2 remains fixed, with group 1 having the higher priority.

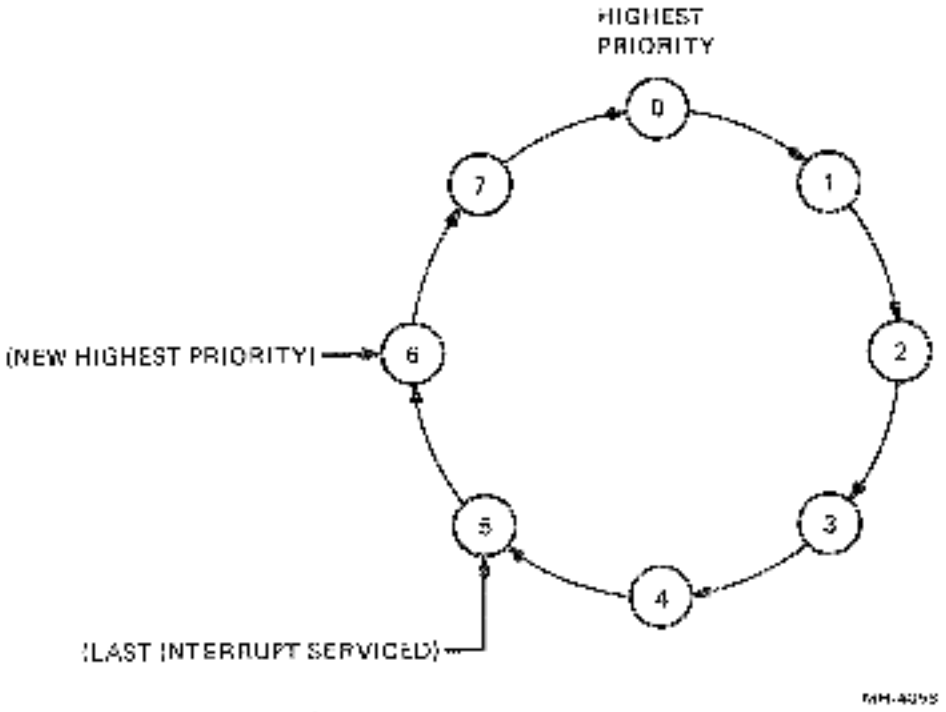


Figure 2-13 Rotating Priority Mode

The example shown in Figure 2-13 assumes IRQ 5 has been serviced and is assigned the lowest priority (7). IRQ 6 now occupies the new highest priority position, IRQ 7 next to the highest, etc. If two new interrupts simultaneously arrive at IRQ 1 and IRQ 4, IRQ 1 is selected and becomes the lowest priority. IRQ 4 will then be acknowledged unless an active input of IRQ 2 or IRQ 3 arrives in the meantime.

This rotating scheme prevents any one interrupt request from dominating the system. An interrupt request will not have to wait for more than seven more service cycles before being acknowledged. Priority is resolved when the ISR bit of the presently selected interrupt is cleared.

In the rotating priority mode, inputs other than the one currently serviced are fenced out and will not cause interrupts until the ISR bit is cleared. Thus, only one bit at a time is set in the ISR. Use care when selecting the rotating mode to keep from doing so again when more than one ISR is set.

2.5.2 Individual Vector or Common Vector Mode

Bit M1 of the mode register specifies the vectoring option. When M1 = 0, the individual vector mode is selected and each interrupt request line is associated with its own location in the vector address memory. Each location contains the vector address that was loaded by the program after system power-up.

When M1 = 1, the common vector mode is selected and all vector information is supplied from the vector address memory location associated with interrupt request line 0 (IRQ 0), regardless of which interrupt request line is acknowledged. The common vector mode is useful in systems where several similar devices share a common service routine and direct individual device identification is not important. This may be true because of the nature of the peripheral-system interaction or in the case of a transient system condition that uses the common vector temporarily to save the additional programming overhead required to load the vector address memory twice per group.

2.5.3 Interrupt or Polled (Flag) Mode

Bit 2 of the mode register allows the system to enable or disable interrupt requests. When M2 = 0, the interrupt mode is selected and interrupts are enabled. The interrupt mode may be considered the "normal" mode because it permits full use of the interrupt control and management capabilities of the DRV11-J.

When M2 = 1, the polled mode is selected, which forces the group interrupt (GINT) output of the interrupt controllers to the inactive state and thus prevents the DRV11-J from issuing a bus interrupt request (BIRQ 4 L). Since no bus interrupt requests are supplied, the processor cannot initiate the interrupt acknowledge sequence. Consequently, ISR bits are not set, masking fences are not erected, and IRR bits are not automatically cleared. Polled-mode operation requires the processor to read the status register to determine if requests are pending. Software routines must then be used to determine which input line requested the interrupt. IRR bits may be cleared by the software. The polled mode of operation effectively bypasses the hardware interrupt, vectoring, and fencing functions of the DRV11-J. What remains is the interrupt request latching and masking functions.

2.5.4 Mode Register Bit 3

Bit 3 of the mode register is not used and must be programmed to a 0.

2.5.5 IRQ Polarity Option

Bit 4 of the mode register specifies the polarity of interrupt request inputs to which the DRV11-J will respond. When M4 = 0, the interrupt request inputs are selected as active low and a negative-going transition is required to set the associated IRR bits. When M4 = 1, the interrupt request inputs are selected as active high and a positive-going transition is required to set the associated IRR bits. This polarity option may be used to simplify the design of the DRV11-J interface to the interrupting devices.

2.5.6 Register Preselection Option

Bits 5 and 6 of the mode register specify the internal data register contents that will be output by the DRV11-J during a read operation at the data port. These bits do not affect destinations for write operations. The four registers that may be read are the IRR, ISR, IMR and ACR. Preselect coding for each register is shown in Figure 2-11. The preselection remains in effect for all data transfers until the contents of M5 and M6 are changed.

The ability to examine these operating registers in conjunction with the status register contents provides important information regarding the current internal conditions of the DRV11-J. The processor's access to these registers permits dynamic operating flexibility and provides important diagnostic, testing, and debugging information.

2.5.7 Master Mask Option

Bit 7 of the mode register specifies the armed status of the DRV11-J by way of the master mask control bit. When M7 = 0, the group is disarmed as if all eight bits in the IMR had been set. IRQ inputs will be accepted and latched but will not be sent to the processor. When M7 = 1, the group is armed and any active unmasked interrupt inputs may cause interrupt requests to the processor.

The master mask option permits the system to disarm a group and prevent the processing of interrupts without disturbing the contents of the IMR. Thus, when the group is re-armed, the old IMR conditions are still valid and need not be reloaded. Note that a single command to the master mask bit of the highest priority interrupt group shuts down the entire interrupt system. This is the only mode bit that affects both groups.

2.6 SYSTEM OPERATING SEQUENCE

The management of interrupts by the DRV11-J requires interaction between the processor, the DRV11-J, and the user device. The operations performed by the system are described in the following typical sequence of events. The DRV11-J is initialized, enabled, and ready to run in the interrupt mode. The processor has enabled its internal interrupt structure to accept DRV11-J interrupt requests.

1. One (or more) of the IRQ inputs becomes active, indicating that service is desired.

2. The requests are captured and latched in the IRR asynchronously. The latching action of the IRR cannot be disabled and active requests will always be stored unless a previous request at the same IRR bit has not been cleared.

3. If the active IRR bit is masked by the corresponding bit in the IMR, no further action takes place. When the IRR bit is not masked, an interrupt request will be generated.

4. When the processor recognizes an interrupt request, it will complete the execution of its current instruction and then execute an interrupt acknowledge cycle.

5. When BIAKI L is received, the DRV11-J begins selection of the highest priority unmasked IRR bit. All interrupts that have become active before the falling edge of BIAK are considered. When selection is complete, the contents of the vector address memory location associated with the selected request are accessed.

6. The processor accepts the vector address on the LSI-11 bus and negates IAK.

7. In parallel with the transfer of the vector address, the DRV11-J automatically clears the selected IRR bit and sets the selected ISR bit. If the auto-clear function is not in force for the selected interrupt, the ISR bit will cause the erection of a masking fence, and interrupts will be disabled until a higher priority interrupt arrives or until the ISR bit is cleared. The interrupt service routine must clear the ISR bit near the end of the routine if the auto-clear function is not used.

8. If a higher priority request arrives while the current request is being serviced, and if the fixed priority mode is in effect, the DRV11-J will output another interrupt request (nested interrupt). The processor will recognize the interrupt signal only if it has enabled its internal interrupt logic. If this new request is acknowledged, the DRV11-J will clear the corresponding IRR bit and set the corresponding ISR bit.

9. When the processor has completed all service associated with the interrupt, it will clear the remaining ISR bit (if the auto-clear capability is not used), enable its internal interrupt system (if it has not already done so), and return to the main program.

2.7 COMMAND DESCRIPTIONS

The DRV11-J command set allows the processor to customize the interrupt operating modes and options for a particular application. Commands are also used to initialize and update the vector address memory locations and to manipulate the internal controlling bits set during interrupt servicing. Commands are entered directly into the command register by writing into the low byte of CSRA for group 1 or CSRC for group 2. Preselection commands entered through CSRA select CSRB for subsequent group 1 register transfers. Preselection commands entered through CSRC select CSRD for subsequent group 2 register transfers. All the available commands are described below and are summarized in Table 2-10. An "X" in any bit position of the command code indicates a "don't care" condition. Any commands that alter the state of the IMR, IRR or master mask bit should be executed with the processor status word at a priority level equal to the DRV11-J to prevent undefined interrupts from occurring.

2.7.1 Reset

C7	C6	C5	C4	C3	C2	C1	C0
0	0	0	0	0	0	0	0

The reset command allows the processor to establish a known internal condition. The vector address memory and byte count registers are not affected by the software reset. The IMR is set to all 1s. The ISR, IRR, ACR and mode registers are cleared to all 0s.

2.7.2 Clear IRR and IMR

C7	C6	C5	C4	C3	C2	C1	C0
0	0	0	1	0	X	X	X

All bits in the IMR and IRR are cleared at the same time. Thus, all interrupts are enabled and the previous history of all IRQ transitions is forgotten. If the interrupt request was active when the command was entered, it becomes inactive.

2.7.3 Clear Single IMR and IRR Bit

C7	C6	C5	C4	C3	C2	C1	C0
0	0	0	1	1	B2	B1	B0

The same single bit is cleared in both the IMR and IRR. Other bits are not changed. If the specified IRR bit is generating an active interrupt output, the interrupt request may become inactive upon entry of the command. The bit position cleared is specified by the B2, B1, B0 field as shown in Table 2-9.

2.7.4 Clear IMR

C7	C6	C5	C4	C3	C2	C1	C0
0	0	1	0	0	X	X	X

All bits in the IMR are cleared. All IRR bits will therefore be unmasked and any IRR bits that were set will be able to cause an active interrupt request after the command is entered.

2.7.5 Clear Single IMR Bit

C7	C6	C5	C4	C3	C2	C1	C0
0	0	1	0	1	B2	B1	B0

A single bit in the IMR is cleared. Other bits are not changed. If the corresponding bit in the IRR is set, it will be unmasked and will be able to cause an active interrupt request after entry of the command. The IMR bit cleared is specified by the B2, B1, B0 field as shown in Table 2-9.

2.7.6 Set IMR

C7	C6	C5	C4	C3	C2	C1	C0
0	0	1	1	0	X	X	X

All bits in the IMR are set to 1. All IRR bits will therefore be masked and unable to generate an interrupt request. If the interrupt request is active, it will become inactive after the command is entered.

2.7.7 Set Single IMR Bit

C7	C6	C5	C4	C3	C2	C1	C0
0	0	1	1	0	B2	B1	B0

A single bit in the IMR is set. Other bits are not changed. If the corresponding bit in the IRR is active and generating an interrupt, the interrupt request will become inactive after the command is entered. The IMR bit set is specified by the B2, B1, B0 field as shown in Table 2-9.

2.7.8 Clear IRR

C7	C6	C5	C4	C3	C2	C1	C0
0	1	0	0	0	X	X	X

All bits in the IRR are cleared. The interrupt request will become inactive. New transitions on the IRQ inputs will be necessary to cause an interrupt.

2.7.9 Clear Single IRR Bit

C7	C6	C5	C4	C3	C2	C1	C0
0	1	0	0	1	B2	B1	B0

A single bit in the IRR is cleared; it will not cause an interrupt until it is set. The IRR bit cleared is specified by the B2, B1, B0 field as shown in Table 2-9.

2.7.10 Set IRR

C7	C6	C5	C4	C3	C2	C1	C0
0	1	0	1	0	X	X	X

All bits in the IRR are set to 1. Any that are unmasked will be able to cause an interrupt request. This command allows the processor to initiate eight interrupts in parallel.

2.7.11 Set Single IRR Bit

C7	C6	C5	C4	C3	C2	C1	C0
0	1	0	1	1	B2	B1	B0

A single bit in the IRR is set to 1; if unmasked, it will be able to generate an interrupt request. This command allows the processor to simulate with software the arrival of a hardware interrupt request. It also gives the software access to the hardware priority resolution, masking, and control features of the DRV11-J. The bit set is specified by the B2, B1, B0 field as shown in Table 2-9.

2.7.12 Clear Highest Priority ISR Bit

C7	C6	C5	C4	C3	C2	C1	C0
0	1	1	0	X	X	X	X

A single bit in the ISR is cleared. If only one bit was set, the set bit is cleared. If more than one bit was set, this command clears the bit with the highest priority. This command is useful in software contexts where the service routine does not know which device is being serviced. It should be used with caution since the highest priority ISR bit may not really be the bit intended. When using the auto-clear option on some interrupts, and/or when a subroutine nesting hierarchy is not priority-driven, the highest priority ISR bit may not correspond to the bit being serviced.

2.7.13 Clear ISR

C7	C6	C5	C4	C3	C2	C1	C0
0	1	1	1	0	X	X	X

All bits in the ISR are cleared. Mask fencing is eliminated.

2.7.14 Clear Single ISR Bit

C7	C6	C5	C4	C3	C2	C1	C0
0	1	1	1	1	B2	B1	B0

A single bit in the ISR is cleared. If the bit was already cleared, no effective operation takes place. The bit cleared is specified by the B2, B1, B0 field as shown in Table 2-9. This command is most useful to service routines in managing the ISR without the help of the auto-clear option.

2.7.15 Load Mode Bits M0 through M4

C7	C6	C5	C4	C3	C2	C1	C0
1	0	0	M4	0	M2	M1	M0

The five low-order bits of the command register are transferred into the five low-order bits of the mode register. This command controls all of the mode options except the master mask and the register preselection.

2.7.16 Control Mode Bits M5, M6 and M7

C7	C6	C5	C4	C3	C2	C1	C0
1	0	1	0	M6	M5	M1	M0

The M6, M5 field in the command is loaded into the M6, M5 locations in the mode register. This field controls the register preselection bits in the mode register. The N1, N0 field in the command controls mode bit M7 (master mask) and is decoded as follows.

N1	N0	
0	0	No change to M7
0	1	Set M7
1	0	Clear M7
1	1	(Illegal)

Thus, this command may be considered as three distinct commands, depending on the coding of N1 and N0.

1. Load M5, M6 only
2. Load M5, M6, and set M7
3. Load M5, M6, and clear M7

The command summary in Table 2-10 lists these three versions.

2.7.17 Preselect IMR for Writing

C7	C6	C5	C4	C3	C2	C1	C0
1	0	1	1	X	X	X	X

The IMR is targeted for loading from the data bus when the next write operation occurs at the data port. All subsequent data-write operations will also load the IMR until a different command is entered. Read operations may be successfully inserted between the entry of this command and the subsequent writing of data into the IMR. The mode register is not affected by this command.

2.7.18 Preselect ACR for Writing

C7	C6	C5	C4	C3	C2	C1	C0
1	1	0	0	X	X	X	X

The ACR is targeted for loading from the data bus when the next write operation occurs at the data port. All subsequent data-write operations will also load the ACR until a different command is entered. Read operations may be successfully inserted between the entry of this command and the subsequent writing of data into the ACR. The mode register is not affected by this command.

2.7.19 Preselect Vector Address Memory for Writing

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	BY1	BY0	L2	L1	L0

One level in the vector address memory is targeted for loading from the data bus by subsequent data-write operations. The byte count register for that level is loaded from the BY1, BY0 field in the command. The L2, L1, L0 field specifies which of the eight request levels is being selected. Table 2-8 describes the byte count register field and IRQ-level field coding. This command should be followed by one data-write operation at CSRB group 1 or CSRD group 2 to load the desired vector address. See Figure 2-12 for vector address bit assignments. The byte count should be 1 since the LSI-11 requires only one vector for an interrupt.

Table 2-8 Vector Address Memory Field Coding

Byte Count Register		
BY1	BY0	Count
0	0	1
0	1	2
1	0	3
1	1	4

IRQ Level			
L2	L1	L0	Level
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

The byte count value does not control the number of bytes written into the vector address memory. However, it does control the number of bytes read from the memory by IAK sequences and the number of interrupts generated by the selected request level. Vector address locations are output by the DRV11-J in the same order they were entered. The number of addresses written must equal the byte count; otherwise, erroneous data may remain in the memory and cause an invalid address to be output as a vector.

2.7.20 Coding B2, B1, B0 Field Commands

Table 2-9 describes the coding of the B2, B1, B0 field of the command register that is used to set or clear a specified bit in the IRR, IMR or ISR. Refer to Table 2-10 for a summary of the B2, B1, B0 field coding.

Table 2-9 Command Register B2, B1, B0 Field Coding

Command Register Field			Bit Specified
B2	B1	B0	
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Table 2-10 DRV11-J Command Code Summary

Command Code								Command Description
7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	Reset.
0	0	0	1	0	X	X	X	Clear all IRR and IMR bits.
0	0	0	1	1	B2	B1	B0	Clear the IRR and IMR bits specified by the B2, B1, B0 field.
0	0	1	0	0	X	X	X	Clear all IMR bits.
0	0	1	0	1	B2	B1	B0	Clear the IMR bit specified by the B2, B1, B0 field.
0	0	1	1	0	X	X	X	Set all IMR bits.
0	0	1	1	1	B2	B1	B0	Set the IMR bit specified by the B2, B1, B0 field
0	1	0	0	0	X	X	X	Clear all IRR bits.
0	1	0	0	1	B2	B1	B0	Clear the IRR bit specified by the B2, B1, B0 field.
0	1	0	1	0	X	X	X	Set all IRR bits.
0	1	0	1	1	B2	B1	B0	Set the IRR bit specified by the B2, B1, B0 field.
0	1	1	0	X	X	X	X	Clear the highest priority ISR bit.
0	1	1	1	0	X	X	X	Clear all ISR bits.
0	1	1	1	1	B2	B1	B0	Clear the ISR bit specified by the B2, B1, B0 field.
1	0	0	M4	M3	M2	M1	M0	Load mode register bits 00:04 with the specified pattern.
1	0	1	0	M6	M5	0	0	Load mode register bits 5 and 6 with the specified pattern
1	0	1	0	M6	M5	0	1	Load mode register bits 5 and 6 and set mode bit 7.
1	0	1	0	M6	M5	1	0	Load mode register bits 5 and 6 and clear mode bit 7
1	0	1	1	X	X	X	X	Preselect the IMR for subsequent writing through CSRB or CSRD.
1	1	0	0	X	X	X	X	Preselect the ACR for subsequent writing through CSRB or CSRD.
1	1	1	BY1	BY0	L2	L1	L0	Load BY1 and BY0 into the byte count register and preselect the vector address memory request level specified by the L2, L1, L0 field for subsequent writing through CSRB or CSRD.

NOTE: X = "don't care" condition

CHAPTER 3
CONFIGURATION

3.1 GENERAL DESCRIPTION

This chapter describes how users may configure the DRV11-J module to function with their systems. Eleven wire-wrap jumpers or jumper clips may be installed or removed in various combinations to select the desired configuration. Nine of the jumpers (W1 through W9) are used to select the device starting address. Jumper W10 is reserved for future use. Jumper W11 is used to select the combination of high-byte port A signals used to generate the interrupt requests. The location of these jumpers is shown in Figure 3-1.

3.2 FACTORY CONFIGURATION

Users may reconfigure any of the jumpers (except W10) so that the module will function in their particular systems. The factory configuration as shipped is described in Table 3-1 to help users determine the jumper changes required. Table 3-2 lists the jumpers and describes their functions.

3.3 DEVICE ADDRESSES

The DRV11-J contains eight device registers that can be individually addressed by the computer program. The eight device registers are divided into four control/status registers (CSRA, CSRB, CSRC and CSRD) and four data buffer registers (DRBA, DRBB, DRBC and DRBD). Each of the I/O ports (A, B, C and D) is accessed by a control/status register and a data buffer register associated with that port. Table 3-3 lists the eight addressable device registers.

The DRV11-J jumper arrangement provides the capability to configure any address from 760000_8 to 777600_8. But the only addresses that may be used must fall within the block of addresses that are assigned to the area of the address map reserved for users. This area is the range of addresses from 764000_8 to 767776_8.

Three standard device addresses have been assigned for use with DRV11-Js: 764160, 764140 and 764120. The module is configured at the factory for an address of 764160. If two additional modules are used in a system, the second DRV11-J would normally be configured for 764140 and the third for 764120.

If the system application requires more than three DRV11-Js, addresses for the additional modules must be selected from the user-reserved area of the address map and assigned in descending order in a modulus of 20 (octal). When selecting addresses other than the three standard addresses, refer to the current issue of the *Microcomputer Interfaces Handbook* to avoid possible I/O device address conflicts.

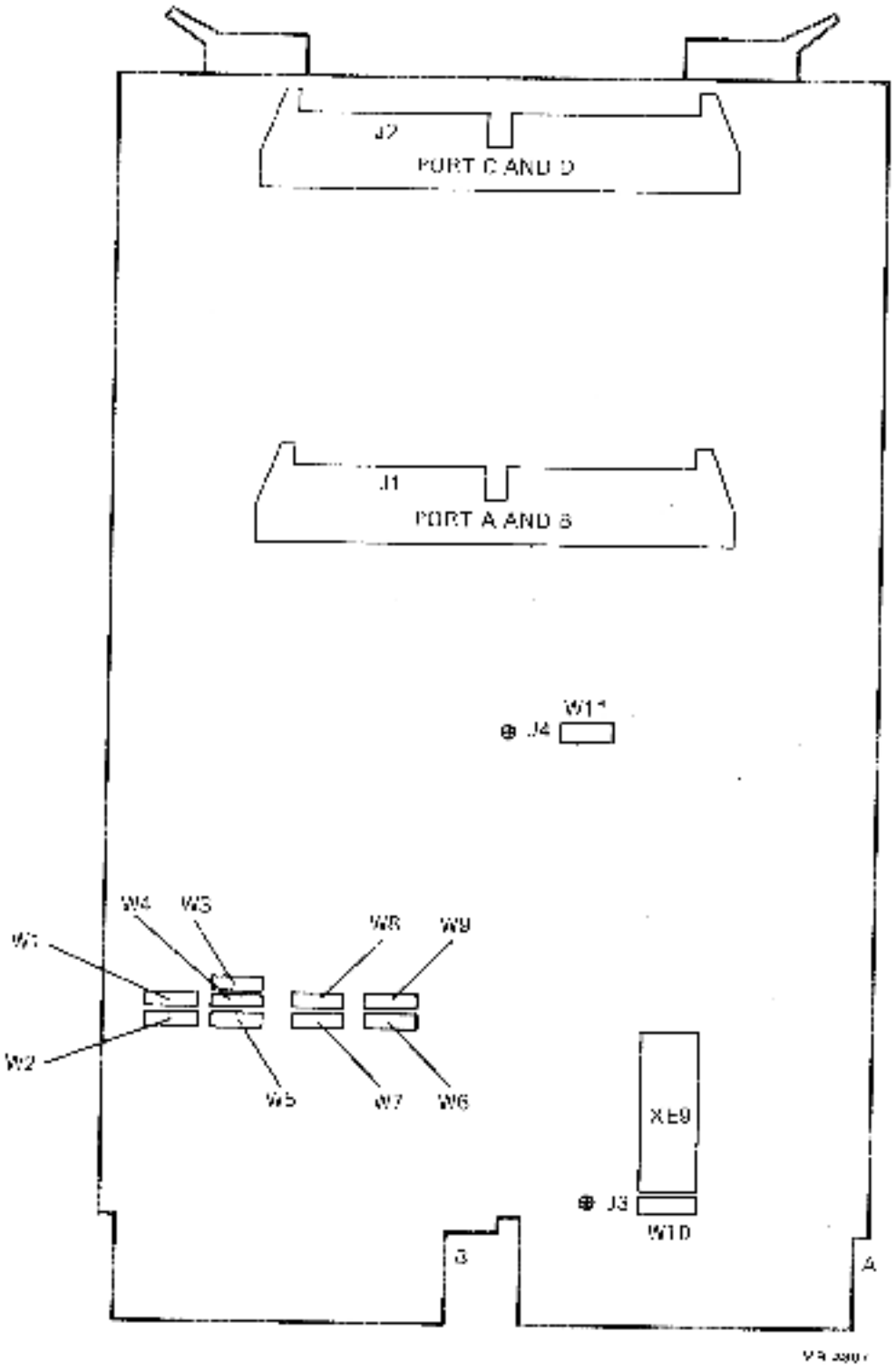


Figure 3-1 DRV11-J Jumper Locations

Table 3-1 DRV11-J Factory Jumper Configuration

Jumper	Jumper State*	Function Implemented
W1	R	This arrangement of jumpers W1 through W9 assigns the device address 764160$_8$ to the first of eight addressable bus registers. With a starting address of 764160$_8$, the remaining bus registers are automatically assigned the following contiguous addresses.
W2	I	
W3	R	CSRA 764160 DBRA 764162
W4	R	CSRB 764164 DBRB 764166
W5	R	CSRC 764170 DBRC 764172
W6	R	CSRD 764174 DBRD 764176
W7	I	
W8	I	
W9	I	
W10	I	Reserved for future use
W11	I	DRV11-J monitors group 2 vectored interrupts using port A I/O bits <11:08> and USER REPLY (A through D) signals (default configuration).

* R = removed = 0.
I = installed = 1.

Table 3-2 DRV11-J Jumper Functions

Jumper	Function	Description
W1	A12	Jumpers W1 through W9 correspond to address bits A12 through A4, respectively. The jumpers implement an octal base device address in the 760000 through 777760 range. A jumper installed connects the address bit to ground and permits a match with the corresponding BDAL L = 1 (low) bit. Removing a jumper permits an address bit match with the corresponding BDAL L = 0 (high) bit.
W2	A11	
W3	A10	
W4	A9	
W5	A8	
W6	A7	
W7	A6	
W8	A5	
W9	A4	
W10		Reserved for future use.
W11	Group 2 vectored interrupts	W11 selects the port A high-byte (group 2) signals to be used for generating vectored interrupts. When W11 is installed (factory configuration), port A I/O bits <11:08> and the USER REPLY (A through D) signals are used for generating vectored interrupts. Connecting W11 to J4 permits the port A I/O bits <15:08> to cause interrupts and disables the USER REPLY (A through D) interrupt inputs.

Table 3-3 DRV11-J Registers

Mnemonic	Description	Address (octal)*
CSRA	Control status register A	7XXXX0
DBRA	Data buffer register A	7XXXX2
CSRB	Control status register B	7XXXX4
DBRB	Data buffer register B	7XXXX6
CSRC	Control status register C	7XXXX10
DBRC	Data buffer register C	7XXXX12
CSRD	Control status register D	7XXXX14
DBRD	Data buffer register D	7XXXX16

*XXXX is jumper-selectable between 6000_8 and 7770_8 to configure the module for a group of addresses in a modulus of 20 (octal); factory set to 6416_8 (CSRA = 764160_8; DBRD = 764176_8).

3.4 DEVICE ADDRESS JUMPERS

Nine address jumpers (W1 through W9) are installed or removed to establish a base device register address. Figure 3-2 shows the format of a DRV11-J device address. Note that address bits A13 through A15 are neither configured nor decoded by the module. These bits are decoded by the bus master module as the bank 7 select (BBS 7 L) bus signal. Address bit 0 is used by the program to select a high-byte or low-byte operation. Address bits 1 through 3 are used to select one of the eight device registers in the addressed module.

3.5 INTERRUPT VECTOR ADDRESSES

The DRV11-J may be programmed to operate in systems that are either interrupt-driven or software-polled. If the DRV11-J is used in an interrupt-driven system, the interrupt vector addresses must be programmed into a RAM (vector address memory) contained in the two interrupt controller chips E2 and E10.

A total of 16 vector addresses may be stored in the vector address memory. Although the vector address bits (07:00) (see Figure 2-12) provide the capability to program addresses in the 0000 through 1774 (octal) range, the vector addresses actually assigned must conform to the floating vector conventions established for the LSI-11 bus. The floating vector convention used for communications devices (and other devices that interface with the PDP-11 series of products) assigns vectors in order, starting at 300 and ending at 776 (octal). To avoid device conflicts, refer to the current issue of the *Microcomputer Interfaces Handbook* when assigning vector addresses.

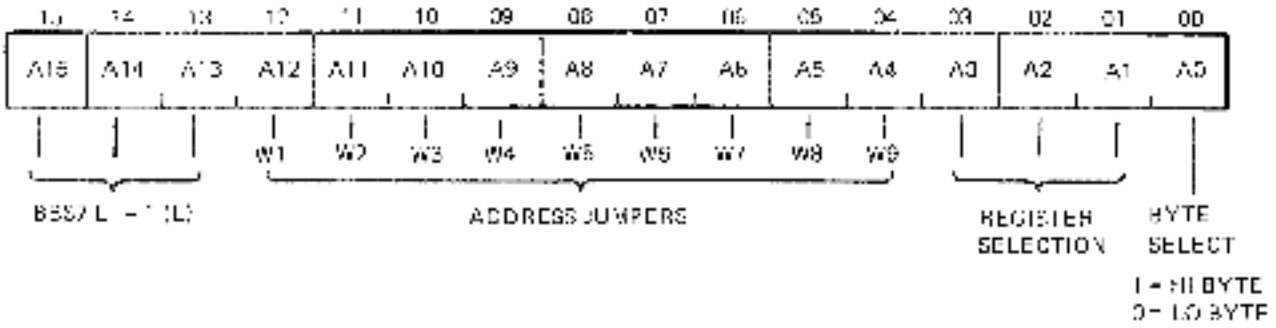


Figure 3-2 DRV11-J Device Address Format

CHAPTER 4
INTERFACING

4.1 INTERFACE CONNECTORS

Two board mounted 50-pin male connectors (J1 and J2) interface the DRV11-J to the user device. Connector J1 is used to interface the port A and port B signals, while J2 is used for the port C and port D signals. Figure 4-1 shows the location of the J2 connector and the pin numbering scheme. The numbering of pins on connector J1 (not shown) is similar to that on J2. The interface signal names and their respective connector pins are described in Table 4-1.

4.2 INPUT/OUTPUT SIGNAL FUNCTIONS

Programmed input/output data transfers between the DRV11-J and the user device may be accomplished by the assertion of four control signals associated with each DRV11-J port. The control signals must be asserted in a handshaking sequence (protocol) to synchronize the DRV11-J and the user device, thus ensuring that no data is lost. The simplified schematic Figure 4-2 shows the relation of the I/O signals to the internal interface logic. Table 4-2 lists the I/O signals and describes their functions.

4.3 INPUT/OUTPUT SIGNAL ASSERTION LEVELS

The DRV11-J I/O signal assertion levels at the J1 and J2 connectors are defined separately for the I/O bus signals, protocol signals and USER RPLY signals. All I/O bus signals (I/O <15:0>) are defined as being asserted (1) high (+3 V) and negated (0) low (ground). Write data is output by the DRV11-J to the I/O bus through latched drivers, while input data is received through unlatched Schmitt trigger buffers.

CAUTION

In order for group 1 IRR bits <7:0> and group 2 IRR bits <3:0> to be valid, the A I/O <11:0> lines must be connected to the user device and have active signals present. If the A I/O <11:0> lines are open, the IRR bits must be masked in the corresponding IMR register.

The protocol signals (DRV11-J RDY, DRV11-J RPLY and USER RDY) are defined as being asserted (1) low (ground) and negated (0) high (+3 V). Active transition of the USER RPLY signals is defined by setting mode register bit M4 in the group 2 interrupt controller when the W11 jumper is installed.

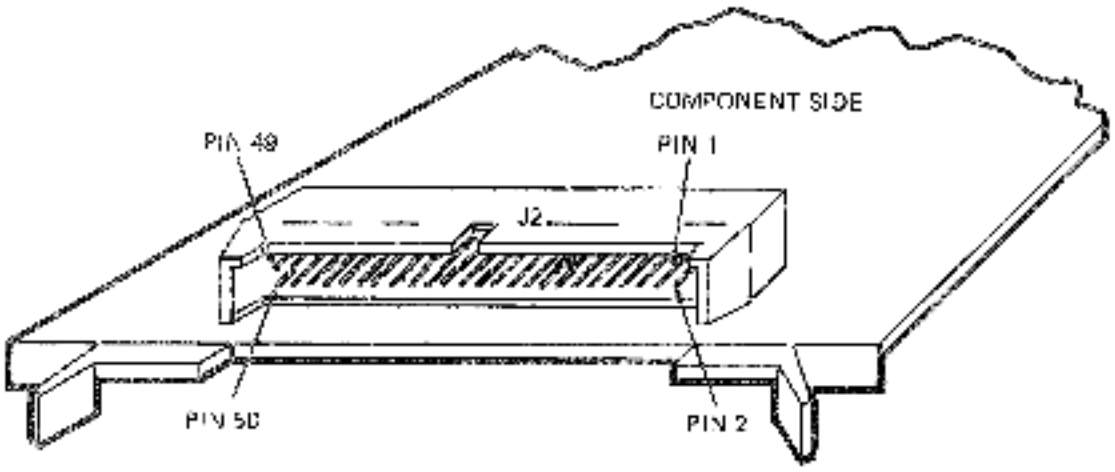


Figure 4-1 DRV11-J I/O Connector Pin Locations

Table 4-1 I/O Connector Pin Assignments

Signal Name	J1 Connector Pin	Signal Name	J2 Connector Pin
DRV11J RDY A	J1-29	DRV11J RDY D	J2-29
DRV11J RPLY A	J1-33	DRV11J RPLY D	J2-33
USER RDY A	J1-31	USER RDY D	J2-31
USER RPLY A	J1-27	USER RPLY D	J2-27
A I/O 15	J1-45	D I/O 15	J2-45
A I/O 14	J1-46	D I/O 14	J2-46
A I/O 13	J1-43	D I/O 13	J2-43
A I/O 12	J1-49	D I/O 12	J2-49
A I/O 11	J1-48	D I/O 11	J2-48
A I/O 10	J1-44	D I/O 10	J2-44
A I/O 9	J1-50	D I/O 9	J2-50
A I/O 8	J1-47	D I/O 8	J2-47
A I/O 7	J1-41	D I/O 7	J2-41
A I/O 6	J1-36	D I/O 6	J2-36
A I/O 5	J1-42	D I/O 5	J2-42
A I/O 4	J1-35	D I/O 4	J2-35
A I/O 3	J1-40	D I/O 3	J2-40
A I/O 2	J1-38	D I/O 2	J2-38
A I/O 1	J1-39	D I/O 1	J2-39
A I/O 0	J1-37	D I/O 0	J2-37
GND	J1-26	GND	J2-26
GND	J1-28	GND	J2-28
GND	J1-30	GND	J2-30
GND	J1-32	GND	J2-32
GND	J1-34	GND	J2-34

Table 4-1 I/O Connector Pin Assignments (Cont)

Signal Name	J1 Connector Pin	Signal Name	J2 Connector Pin
DRVIIJ RDY B	J1-20	DRVIIJ RDY C	J2-20
DRVIIJ RPLY B	J1-24	DRVIIJ RPLY C	J2-24
USER RDY B	J1-22	USER RDY C	J2-22
USER RPLY B	J1-18	USER RPLY C	J2-18
B I/O 15	J1-6	C I/O 15	J2-6
B I/O 14	J1-5	C I/O 14	J2-5
B I/O 13	J1-8	C I/O 13	J2-8
B I/O 12	J1-2	C I/O 12	J2-2
B I/O 11	J1-3	C I/O 11	J2-3
B I/O 10	J1-7	C I/O 10	J2-7
B I/O 9	J1-1	C I/O 9	J2-1
B I/O 8	J1-4	C I/O 8	J2-4
B I/O 7	J1-10	C I/O 7	J2-10
B I/O 6	J1-15	C I/O 6	J2-15
B I/O 5	J1-9	C I/O 5	J2-9
B I/O 4	J1-16	C I/O 4	J2-16
B I/O 3	J1-11	C I/O 3	J2-11
B I/O 2	J1-13	C I/O 2	J2-13
B I/O 1	J1-12	C I/O 1	J2-12
B I/O 0	J1-14	C I/O 0	J2-14
GND	J1-17	GND	J2-17
GND	J1-19	GND	J2-19
GND	J1-21	GND	J2-21
GND	J1-23	GND	J2-23
GND	J1-25	GND	J2-25

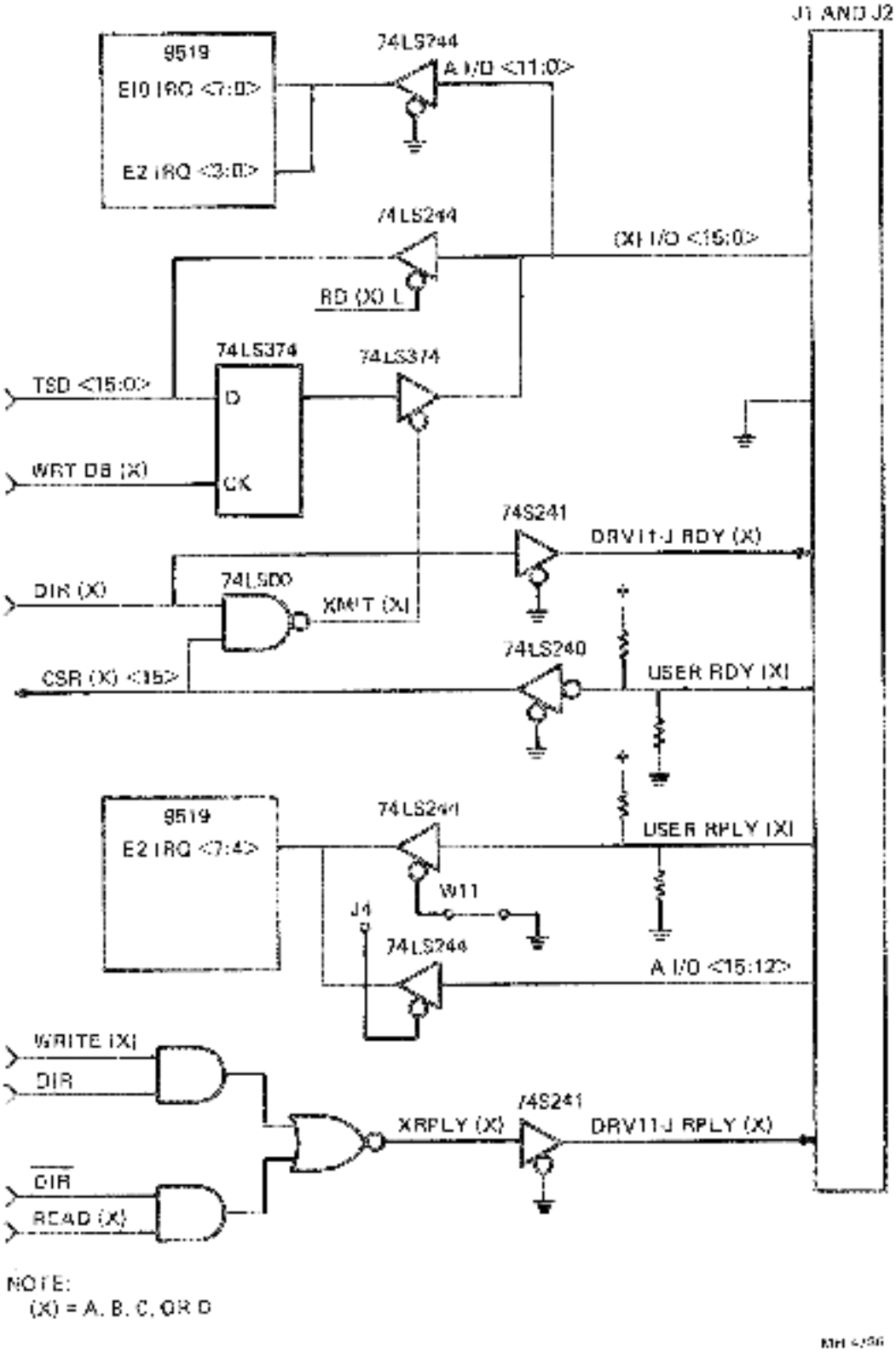


Figure 4-2 I/O Bus Interface, Simplified Schematic

Table 4-2 I/O Signal Functions

Signal Name*	Function
DRV11J RDY (X)	The DRV11-J asserts this signal during an input operation (read) to inform the user device that it is ready to accept data. The signal is asserted when the corresponding DRV11J DIR bit is cleared.
DRV11J RPLY (X)	This pulse is generated by the DRV11-J to notify the user device that data has been accepted (read) or that data is available (write). When the DRV11-J is the input device (read), the pulse is generated by reading the corresponding data buffer with the associated DIR bit *cleared*. When the DRV11-J is the output device (write), the pulse is generated by writing the corresponding data buffer with the associated DIR bit *set*.
(X) I/O <15:0>	These are the 16 3-state I/O bus inputs and outputs.
USER RDY (X)	The user device asserts this signal during a DRV11-J output operation to inform the DRV11-J that it desires data. This signal, in conjunction with the associated DIR bit, enables the DRV11-J 3-state outputs. It appears as bit 15 in CSR (X) and must be asserted by the user device to enable the DRV11-J to output data.
USER RPLY (X)	This signal is asserted by the user device to inform the DRV11-J that data is available or that data has been accepted. When the DRV11-J is the input device, the signal is asserted to indicate that data is available. When the DRV11-J is the output device, the signal is asserted when the user device accepts the data. This signal will generate an interrupt request if WIE is installed.

*(X) = A, B, C or D.

4.4 INPUT/OUTPUT SIGNAL LOOPBACK CONNECTIONS

The DRV11-J signal pin assignments are arranged to permit loopback operation when a BC05W-XX cable is installed with a half twist connecting J1-1 to J2-50. Cable BC05W-XX must be installed to run the CVDRCA, CVDRDA and DECX11 module diagnostics. With the cable installed in this manner, the proper connections are made to loopback the DRV11-J protocol signals. Communication with this type of connection is made between ports A and C and between ports B and D. This arrangement also permits interconnecting two DRV11-Js by the same method, with communication between either J1 and J1 or J1 and J2. Table 4-3 describes the loopback signal connections between ports A and C and between ports B and D.

4.5 INTERFACE CABLE

The BC05W-XX cable may be used to connect the DRV11-J to user devices or to link two UNI-11 buses together through two DRV11-Js. The BC05W-XX is a flat shielded cable with 50-pin connectors at both ends, and is available in 0.6 m (2 ft), 3.0 m (10 ft) and 7.6 m (25 ft) lengths. The cable length (XX) is specified in feet. For example, a 2-foot BC05W cable is ordered as BC05W-02.

The maximum cable length of 25 feet is specified for the distance between two DRV11-Js or from a DRV11-J to a user device with an ac load equivalent to the DRV11-J. The maximum cable length may have to be shortened if the ac load of the user device is greater than the ac load of the DRV11-J.

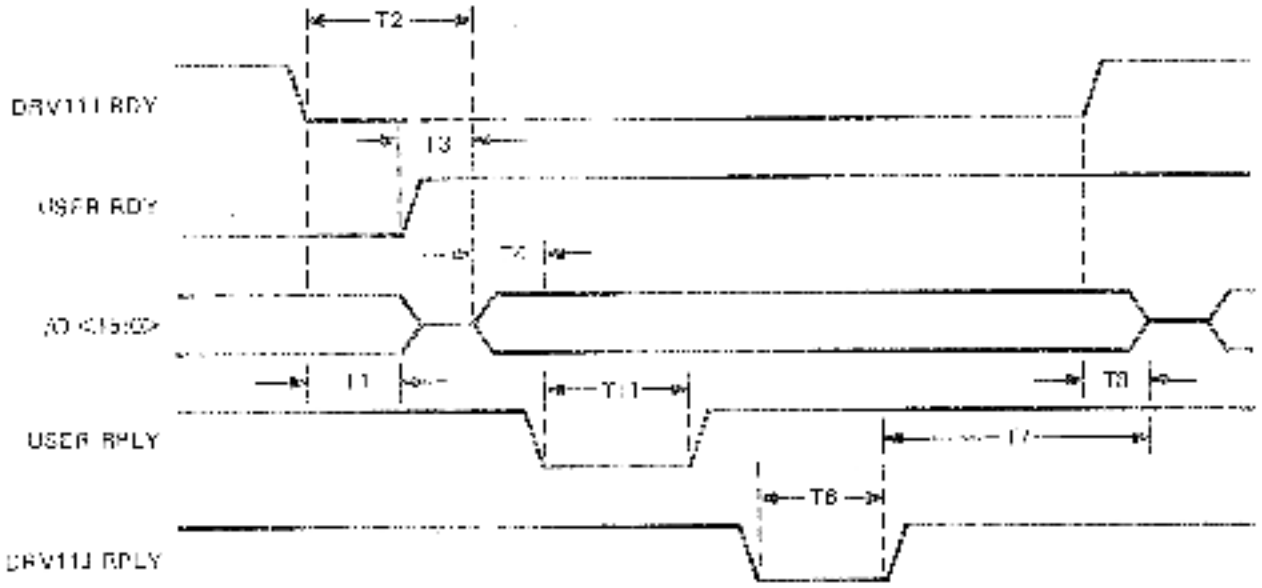
4.6 INPUT/OUTPUT FUNCTION TIMING

The time relationships between the DRV11-J signals and the user device signals required to perform input/output data transfers are shown in Figure 4-3. The timing tolerances between the various signals are described in Table 4-4.

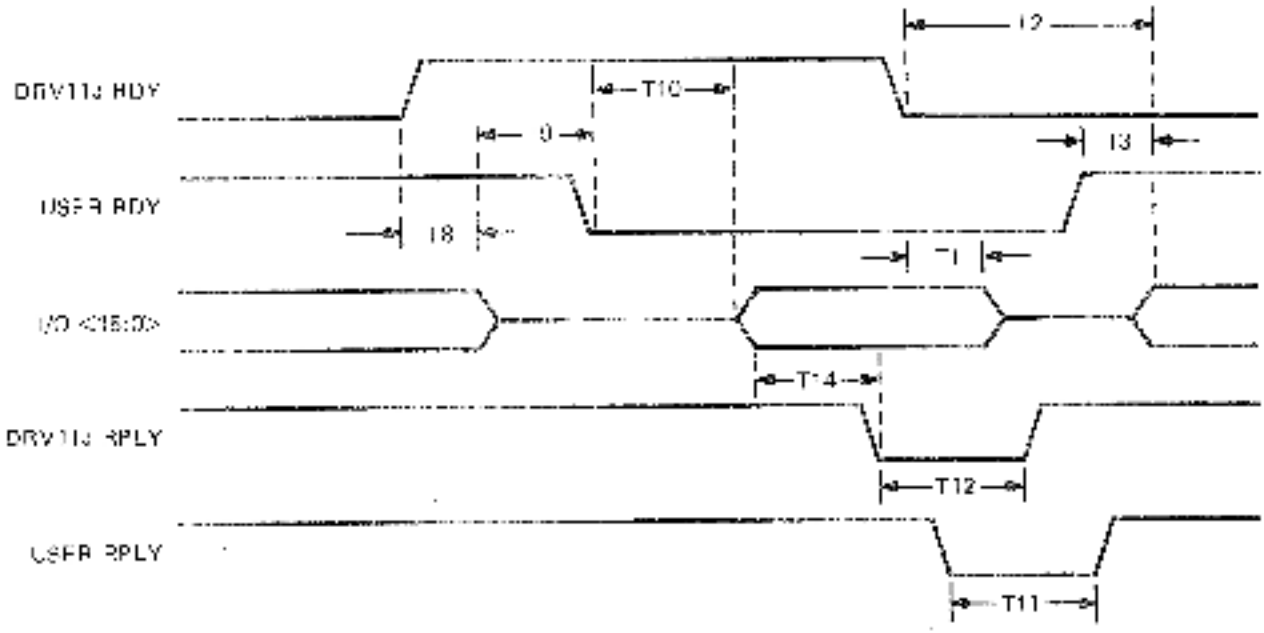
Table 4-3 DRV11-J Loopback Signal Connections

J1 Pin No.		Signal		Signal	J2 Pin No.
	1	B I/O 0	↔	D I/O 0	50
	2	B I/O 12	↔	D I/O 12	49
	3	B I/O 11	↔	D I/O 11	48
	4	B I/O 8	↔	D I/O 8	47
	5	B I/O 14	↔	D I/O 14	46
	6	B I/O 15	↔	D I/O 15	45
	7	B I/O 10	↔	D I/O 10	44
	8	B I/O 13	↔	D I/O 13	43
Port B	9	B I/O 5	↔	D I/O 5	42 Port D
	10	B I/O 7	↔	D I/O 7	41
	11	B I/O 3	↔	D I/O 3	40
	12	B I/O 1	↔	D I/O 1	39
	13	B I/O 2	↔	D I/O 2	38
	14	B I/O 0	↔	D I/O 0	37
	15	B I/O 6	↔	D I/O 6	36
	16	B I/O 4	↔	D I/O 4	35
	18	USER RPLY B	→	DRV11J RPLY D	33
	20	DRV11J RDY B	→	USER RDY D	31
	22	USER RDY B	←	DRV11J RDY D	29
	24	DRV11J RPLY B	→	USER RPLY D	27
	37	USER RPLY A	←	DRV11J RPLY C	24
	39	DRV11J RDY A	→	USER RDY C	22
	31	USER RDY A	←	DRV11J RDY C	20
	33	DRV11J RPLY A	←	USER RPLY C	18
	35	A I/O 4	↔	C I/O 4	16
	36	A I/O 6	↔	C I/O 6	15
	37	A I/O 0	↔	C I/O 0	14
	38	A I/O 2	↔	C I/O 2	13
Port A	39	A I/O 1	↔	C I/O 1	12 Port C
	40	A I/O 3	↔	C I/O 3	11
	41	A I/O 7	↔	C I/O 7	10
	42	A I/O 5	↔	C I/O 5	9
	43	A I/O 13	↔	C I/O 13	8
	44	A I/O 10	↔	C I/O 10	7
	45	A I/O 15	↔	C I/O 15	6
	46	A I/O 14	↔	C I/O 14	5
	47	A I/O 8	↔	C I/O 8	4
	48	A I/O 11	↔	C I/O 11	3
	49	A I/O 12	↔	C I/O 12	2
	50	A I/O 9	↔	C I/O 9	1

NOTE: Connector pins 17, 19, 21, 23, 25, 26, 28, 30, 32 and 34 on J1 and J2 are grounds



DRV11J INPUT TIMING



DRV11J OUTPUT TIMING

NOTE

REFER TO TABLE 4-4 I/O FUNCTION TIMING TOLERANCE FOR DESCRIPTION OF T1 THROUGH T14.

Figure 4-3 DRV11-J I/O Function Timing

Table 4-4 I/O Function Timing Tolerance

Name*	Description	Tolerance**	
		Min.	Max.
T1	DRVTJ RDY to DRVTJ 3-state outputs disabled	0	50
T2	DRVTJ RDY to user device 3-state outputs enabled	50	–
T3	USER RDY to user device 3-state output enable	0	–
T4	User device 3-state data setup to USER RPLY	0	–
T5	DRVTJ RPLY pulse width (input mode)	410	2030
T7	User device 3-state data hold time after DRVTJ RPLY	0	–
T8	DRVTJ RDY to user device 3-state output disable	0	
T9	User 3-state outputs disabled to USER RDY assertion	0	
T10	USER RDY to DRVTJ 3-state outputs enabled	0	
T11	USER RPLY pulse width	270	–
T12	DRVTJ RPLY pulse width (output)	410	2030
T13	USER RPLY hold time after DRVTJ RPLY	0	–
T14	DRVTJ 3-state data to DRVTJ RPLY assertion	300	

*Refer to Figure 4-3 for illustration of T1 through T14.
**Tolerances are in nanoseconds

4.7 INPUT DATA OPERATION

An input data operation is a transferral of a 16-bit data word from a user device to any DRV11-J input port. Three control signals and the I/O <15:0> bus lines are used to perform an input data transfer. The transfer sequence is initiated when the DRV11-J asserts DRV11J RDY to inform the user device that data may be placed on the I/O bus associated with the RDY signal. The user device then places the data on the bus and asserts USER RPLY to inform the DRV11-J that data is available. The DRV11-J reads the input data buffer and then asserts DRV11J RPLY to notify the user device that the data has been accepted.

The sequence of operations performed by the DRV11-J and the user device during an input data transfer is shown in Figure 4-4. (The timing between the control signals is shown in Figure 4-3.)

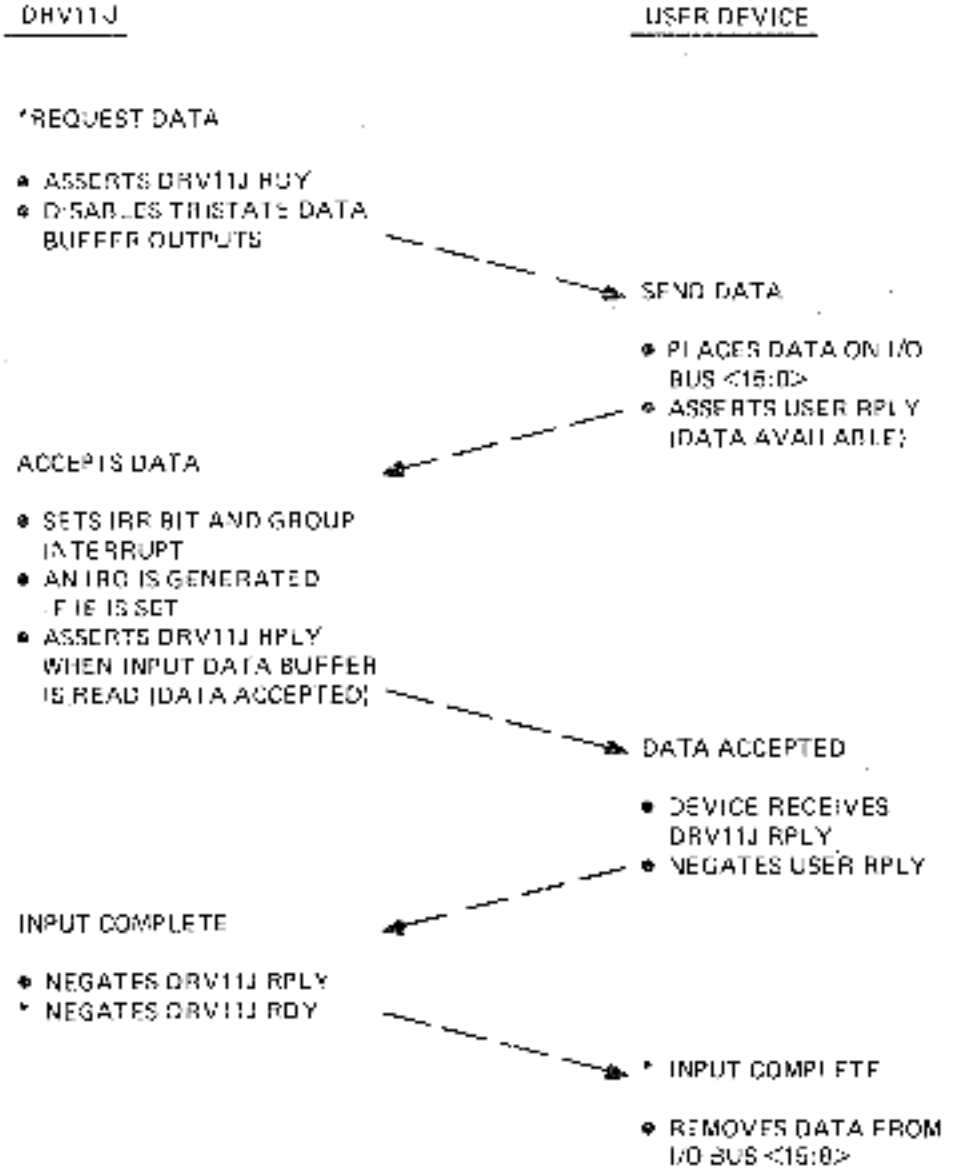
4.8 OUTPUT DATA OPERATION

An output data operation is a transferral of a 16-bit data word from any DRV11-J port to a user device. Three control signals and the I/O <15:0> bus lines are used to perform an output data transfer. The output data transfer is initiated when the user device asserts USER RDY to inform the DRV11-J to send data. The DRV11-J outputs the data on the I/O <15:0> bus lines and asserts DRV11J RPLY to inform the user device that data is available. The user device accepts the data and then asserts USER RPLY to notify the DRV11-J that the data has been accepted.

The sequence of operations performed by the user device and the DRV11-J during an output data transfer is shown in Figure 4-5. (The timing between the control signals is shown in Figure 4-3.)

4.9 INTERRUPT OPERATION

The user device can input up to 16 individual interrupt requests to the DRV11-J, either through the port A I/O <15:0> lines or through the 4 USER RPLY signals and the A I/O <11:0> lines. The DRV11-J cannot process interrupt requests until its interrupt control logic is enabled by the processor. The processor enables the DRV11-J by setting the interrupt enable bit 9 of CSRA. The sequence of DRV11-J and user device signals during an interrupt operation is shown in Figure 4-6.



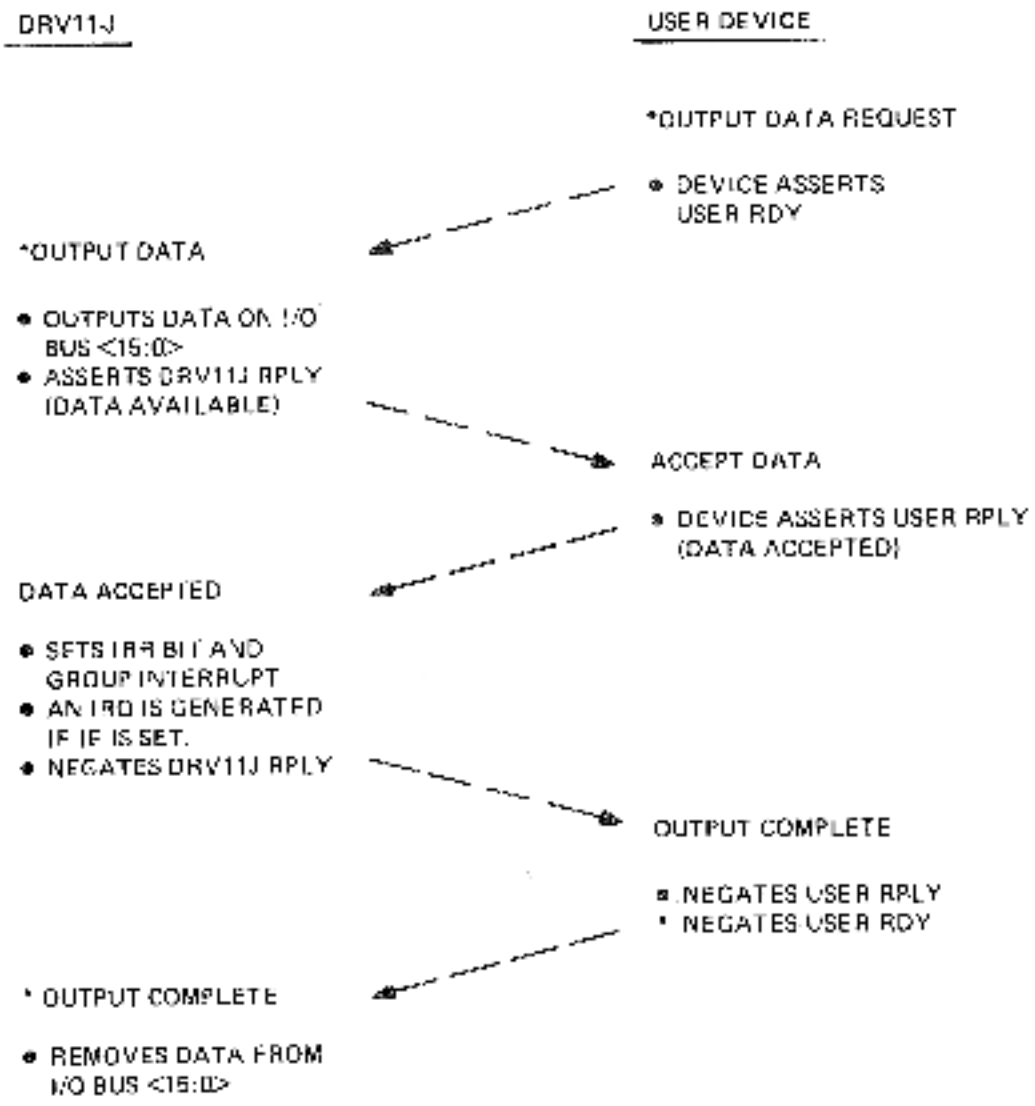
NOTES

IF THE USER DEVICE IS INCAPABLE OF EXECUTING THE INPUT FUNCTION PROTOCOL, DATA TRANSFER IS DEPENDENT UPON PERIODIC READING OF THE INPUT BUFFER, WITH THE DRV11-J IN AN INPUT MODE. (DIR BIT CLEARED)

* THESE STEPS ARE ONLY REQUIRED WHEN I/O MODES ARE SWITCHED FROM INPUT TO OUTPUT OR OUTPUT TO INPUT. IF MODES ARE NOT SWITCHED, THE USER DEVICE SENDS DATA AND THE DRV11-J ACCEPTS THE DATA TO COMPLETE THE DATA TRANSFER.

MR-4351

Figure 4-4 Input Data Transfer Sequence



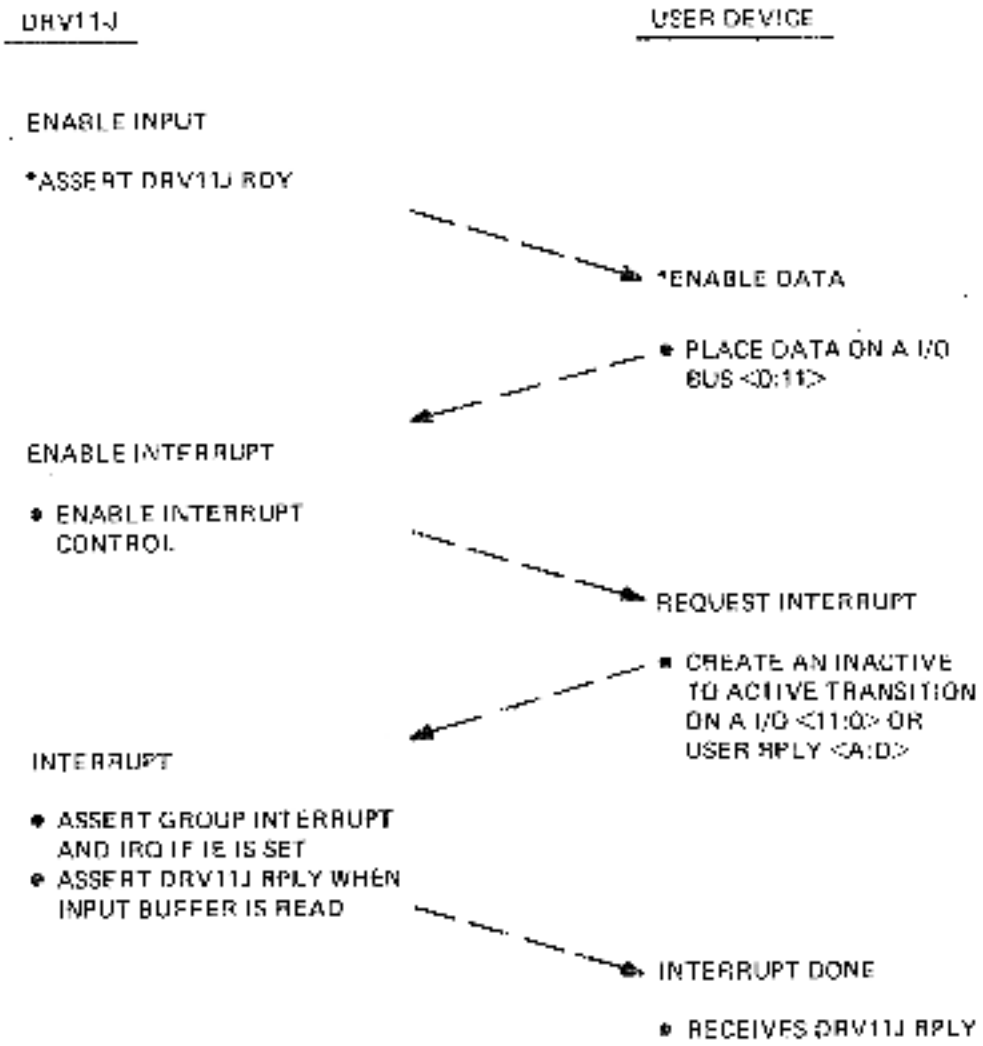
NOTES

IF THE USER DEVICE IS INCAPABLE OF PERFORMING THE OUTPUT FUNCTION PROTOCOL, THEN DATA TRANSFERS ARE DEPENDENT ON PERIODICALLY WRITING THE OUTPUT DATA BUFFER WHILE THE USER RDY SIGNAL IS HELD ASSERTED (GND) WITH THE DRV11-J IN AN OUTPUT MODE. (DIR BIT SET)

* THESE STEPS ARE ONLY REQUIRED IF MODES ARE SWITCHED BETWEEN INPUT AND OUTPUT OR OUTPUT AND INPUT. IF MODES ARE NOT SWITCHED, THE DRV11-J SENDS THE DATA AND THE USER DEVICE ACCEPTS THE DATA TO COMPLETE THE DATA TRANSFER.

MR-4362

Figure 4-5 Output Data Transfer Sequence



NOTE
* THESE STEPS ARE NOT REQUIRED IF MODES ARE NOT CHANGING FROM OUTPUT TO INPUT.

MR-4256

Figure 4-6 Interrupt Sequence

CHAPTER 5
PROGRAMMING EXAMPLES

5.1 GENERAL DESCRIPTION

The DRV11-J may be used in systems where the data is transferred to or from the user device under program control, or in those using interrupt-driven service routines. Programmed data transfers may be performed with or without the protocol control signals (handshaking), depending on the system's complexity. The simplest of system applications may not require the handshaking signals, whereas more complicated system applications require handshaking signals to synchronize the processor with the user device. The following three programming examples illustrate how the DRV11-J may be programmed to operate in program-controlled data transfer systems without handshaking and with, and in interrupt-driven systems.

5.2 PROGRAMMED DATA TRANSFER WITHOUT HANDSHAKING

In the simplest system applications, input and output data transfers may be performed under program control by reading and writing the data buffer registers (DBRA, DBRB, DBRC and DBRD). Data can be transferred on a bit-by bit basis, the method used when the DRV11-J is connected to a simple user device that does not generate or interpret handshaking signals. For example, 1 port could monitor 16 independent switches. If, in actual operation, input to the DRV11-J is allowed to change while the software is reading the buffer, erroneous data may be read. In such a case, the software can "debounce" the line by reading the line until it gives reproducible results. The routines shown in Figure 5-1 illustrate the software interface to the DRV11-J. The first routine initializes the DRV11-J for operation. The second returns the status of 1 of 32 independent input lines that are connected to the A and B I/O pins.

5.3 PROGRAMMED DATA TRANSFER WITH HANDSHAKING

In more complicated system applications, handshaking (DRV11-J polled mode) must be used between the DRV11-J and the user device to indicate the availability of data and to synchronize the sender and receiver so that data is not lost. For example, when the DRV11-J sends a 16-bit command to a user device, it must wait until the command is executed before it can send another. Another example is where the user device assembles 16 signals and then informs the DRV11-J that data is available. In the programming example in Figure 5-2, the first routine initializes the DRV11-J for operation. An input routine reads data from the port A I/O lines after detecting the USER RPLY signal with the group interrupt bit in CSRA. The output routine waits for the USER RDY signal from the user device (available in CSRB) before it outputs data on the port B I/O lines.

```
                                ; Routine to initialize the DRV11-J for input without handshaking.
                                ;       Uses ports A and B for 32 lines of input.
                                ;
000000                          INITDX::
000000  005037  164160                  CLR     @#CSRA          ; initialize for input
000004  005037  164164                  CLR     @#CSRB          ; initialize for input
000010  000207                          RTS     PC              ; return to caller

                                ; Routine to check the status of one of the 32 input lines.
                                ;       The routine expects a line number from 0 to 31. in R0.
                                ;       It returns the state of the line (0 or 1) in R0.
                                ;
000012                          RDLINE::
000012  010046                          MOV     R0, -(SP)       ; save the line number
000014  042700  177757                  BIC     #177757, R0     ; clear all but port flag
000020  040016                          BIC     R0, (SP)        ; clear all but "line in port" bits
000022  006200                          ASR     R0              ; form offset from DBRA
000024  006200                          ASR     R0
000026  016000  164162                  MOV     DBRA(R0), R0    ; load the appropriate buffer register
000032  005316                  1$:     DEC     (SP)            ; shift the bits
000034  100402                          BMI     10$
000036  006200                          ASR     R0              ; in R0
000040  000774                          BR      1$              ; until the right one is in bit 0

000042  042700  177776          10$:    BIC     #177776, R0     ; remove the other bits
000046  005726                          TST     (SP)+           ; pop the saved line number
000050  000207                          RTS     PC              ; and return to the caller


        000001                          .END

SYMBOL TABLE

BASE  = 164160      CSRC  = 164170      DBRA  = 164162      DBRC  = 164172      INITDX  000000RG
CSRA  = 164160      CSRD  = 164174      DBRB  = 164166      DBRD  = 164176      RDLINE  000012RG
CSRB  = 164164
```

Figure 5-1 Example of a Programmed Data Transfer without Handshaking

```
                                        ; Initialize the DRV11-J for programmed I/O with handshaking.
                                        ; Set up to read from port A, write to port B.
                                        ;
  000000                                INITDR::
  000000  010046                                MOV     R0, -(SP)       ; save R0
  000002  005037  164160                        CLR     @#CSRA          ; clear CSR, reset group 1 interrupt control
  000006  012737  [illegible]  164164           MOV     [illegible]     ; set CSRB bit for output
  000014  012700  164170                        MOV     #CSRC, R0       ; R0 points to CSRC
  000020  105010                                CLRB    (R0)            ; reset group 2 interrupt control
  000022  112710  000054                        MOVB    #54, (R0)       ; clear group 2 IRR bit 4 (user reply A)
  000026  112710  000057                        MOVB    #55, (R0)       ; clear group 2 IRR bit 5 (user reply B)
  000032  112710  000204                        MOVB    #204, (R0)      ; set group 2 polled mode
  000036  012600                                MOV     (SP)+, R0       ; restore saved R0
  000040  000207                                RTS     PC              ; return to caller

                                        ; Routine to wait for data available on port A and return the data to the
                                        ; caller in R0.
                                        ;
  000042                                RDPORT::
  000042  105737  164170                10$:    TSTB    @#CSRC          ; wait for group 2 "interrupt"
  000046  100375                                BPL     10$
  000050  112737  000114  164170                MOVB    #114, @#CSRC    ; clear group 2 IRR bit 4
                                                                        ; so we can detect user reply A again
  000056  013700  164162                        MOV     @#DBRA, R0      ; get DBRA
  000062  000207                                RTS     PC              ; and return to caller

                                        ; Routine to send data passed by the caller in R0 to port B and wait for
                                        ; it to be accepted by the user device.
                                        ;
  000064                                WTPORT::
  000064  010037  164166                        MOV     R0, @#DBRB      ; put data into DBRB
  000070  105737  164170                10$:    TSTB    @#CSRC          ; wait for group 2 event (data accepted)
  000074  100375                                BPL     10$
  000076  112737  000115  164170                MOVB    #115, @#CSRC    ; clear group 2 IRR bit 5
  000104  000207                                RTS     PC              ; return to the caller

          000001                                .END

SAMPLE  [illegible]

CSRA  = 164160      CSRC  = 164170      DBRB  = 164166      USER  = [illegible]      RDPORT  000042RG
DBRA  = 164162      CSRB  = [illegible] DBRC  [illegible]   INITDR  000000RG         WTPORT  000064RG
CSRB  = 164164      DBRA  = [illegible]
```

Figure 5-2 Example of a Programmed Data Transfer with Handshaking

5.4 INTERRUPT-DRIVEN TRANSFER

In systems where the number of devices and/or the complexity of service increases, the DRV11-J may be used to enhance processor throughput and response time by eliminating the need for a polling program. In such applications, the DRV11-J can be initialized to interrupt the processor when the user device has accepted data (output) or when it has data available (input). The following two programs output data from port A (see Figure 5-3) and input data from port C (see Figure 5-4) under interrupt control. The program in Figure 5-3 initializes the DRV11-J to interrupt on USER RPLY A (output) and the program in Figure 5-4 initializes the DRV11-J to interrupt on USER RPLY C (input). The DRV11-J vector address memory is loaded with the vector address and the appropriate group 2 interrupt line enabled. The output program will then force an interrupt to occur by setting the group 2 port A IRR bit 4. This starts the interrupt service routine, which runs in parallel with the main program. The programs perform unrelated functions while input/output is proceeding asynchronously. The programs then wait for a done flag, which is set by the interrupt service routines to indicate that the input/output transfer is completed.

```
                                  ; Program to send 256 words of data to DRV11-J port A under interrupt
                                  ; control.
                                  ;
                                  ; Set up DRV11-J vector at an unused location (location 470)
                                  ;
 000000                                   .ASECT
                                  ;
          000470                          . = 470
                                  ;
 000470   001110                          WTINT             ; interrupt is vectored to location WTINT
 000472   000340                          340               ; at priority level 7 (interrupts disabled)
                                  ;
 000000                                   .PSECT
                                  ;
 000000                           OBUF:                     ; this is the 256 word output buffer
          000000                          CONT = 0          ; fill buffer with ascending numbers for test
          000400                          .REPT   256.
                                          .WORD   CONT
                                          CONT = CONT+1
                                          .ENDR
                                  ;
 001000                           OPNTR:  .BLKW             ; next word pointer
 001002                           OFLAG:  .BLKW             ; output done flag
                                  ;
 001004                           START:
 001004   012737  001400  164160          MOV     #1400, @#CSRA  ; set port A for output, enable interrupts,
                                                                 ; and reset group 1 interrupt controller
 001012   012700  164170                  MOV     #CSRC, R0      ; R0 points to CSRC
 001016   105010                          CLRB    (R0)           ; reset group 2 interrupt controller
 001020   112710  000344                  MOVB    #344, (R0)     ; preselect vector address memory (line 4)
 001024   112737  000100  164174          MOVB    #100, @#CSRD   ; load vector of 400
 001032   112737  000241  164160          MOVB    #241, @#CSRA   ; arm group 1 with master mask bit
                                                                 ; this will enable group 2
 001040   112710  000241                  MOVB    #241, (R0)     ; arm group 2 with master mask bit
 001044   112710  000300                  MOVB    #300, (R0)     ; preselect ACR for writing
 001050   112737  000020  164174          MOVB    #20, @#CSRD    ; set ACR to clear line 4 (user reply A)
 001056   012767  000000' 177714          MOV     #OBUF, OPNTR   ; initialize output pointer
 001064   005067  177712                  CLR     OFLAG          ; and done flag
 001070   112710  000034                  MOVB    #34, (R0)      ; clear group 2 IRR bit 4 (user reply A)
 001074   112710  000134                  MOVB    #134, (R0)     ; set group 2 IRR bit 4 (cause an interrupt
                                                                 ; to get things started)
                                  ;
                                  ; Now, the CPU can be used for other things while the data is being sent.
                                  ;
                                  ; Some time later...
                                  ;
 001100   005767  177676          10$:    TST     OFLAG          ; wait for output complete
 001104   001775                          BEQ     10$
 001106   000000                          HALT
                                  ;
                                  ; The following is the interrupt service routine.
                                  ;
 001110                           WTINT:
 001110   112737  000114  164170          MOVB    #114, @#CSRC   ; clear group 2 IRR bit 4 (REPLY A)
 001116   026727  177656  001000'         CMP     OPNTR, #OBUF+512.  ; sent all words in buffer?
 001124   001407                          BEQ     10$            ; if so, we're done
 001126   017737  177646  164162          MOV     @OPNTR, @#DBRA ; if not send out next word
 001134   062767  000002  177636          ADD     #2, OPNTR      ; point to following word
 001142   000002                          RTI                    ; return from interrupt
                                  ;
 001144   005267  177632          10$:    INC     OFLAG          ; signal output complete
 001150   000002                  RTI                            ; return from interrupt
                                  ;
                                  ;
          001004'                         .END    START


SYMBOL TABLE

BASE  = 164160      CSRB  = 164164      DBRA  = 164162      DBRD    164176      OPNTR   001000R
CONT  = 000400      CSRC    164170      DBRB  = 164166      OBUF    000000R     START   001004R
CSRA  = 164160      CSRD    164174      DBRC    164172      OFLAG   001002R     WTINT   001110R
```

Figure 5-3   Example of an Interrupt-Driven Output Program

```
                                ; Program to read 256 words of data from DRV11-J port C under interrupt
                                ; control.

                                ; Set up DRV11-J vector at an unused location (location 400).
                                ;
000000                                  .ASECT

        000400                          .=400

000400  001104'                         RCINT           ; Interrupt is vectored to location RCINT
000402  000340                          340             ; at priority level 7 (interrupts disabled)

000000                                  .PSECT

000000                          IBUF:   .BLKW   256.    ; 256 word input buffer
001000                          IPNTR:  .BLKW           ; next empty word pointer
001002                          IFLAG:  .BLKW           ; input done flag

001004                          START:
001004  012737  001000  164160          MOV     #1000,@#CSRA    ; reset group 1 interrupt controller,
                                                                ; reset = DRV11-J interrupts
001012  012700  164170                  MOV     #CSRC,R0        ; R0 points to CSRC
001016  005010                          CLR     (R0)            ; set port C for input, reset group 2
                                                                ; interrupt controller
001020  112710  000244                  MOVB    #244,(R0)       ; preset vector address memory (line 6)
001024  112737  000100  164174          MOVB    #100,@#CSRD     ; load vector of 400
001032  112737  000241  164162          MOVB    #241,@#CSRB     ; arm group 1 with master mask bit
                                                                ; (this will enable group 2)
001040  112710  000241                  MOVB    #241,(R0)       ; arm group 2 with master mask bit
001044  112710  000300                  MOVB    #300,(R0)       ; preselect ACR for writing
001050  112737  000100  164174          MOVB    #100,@#CSRD     ; set ACR to clear line 6 (user reply C)
001056  012767  000000' 177714          MOV     #IBUF,IPNTR     ; initialize input pointer
001064  005067  177712                  CLR     IFLAG           ; and done flag
001070  112710  000056                  MOVB    #56,(R0)        ; clear group 2 IRR bit 6 (user reply C)
                                                                ; interrupts will now be generated on data
                                                                ; received from port C
                                ;
                                ; Now, the CPU can be used for other things while the data is being received.
                                ;
                                ; Some time later...
                                ;
001074  005767  177702          10$:    TST     IFLAG           ; wait for input finished
001100  001775                          BEQ     10$
001102  000000                          HALT

                                ; The following is the interrupt service routine.
                                ;
001104  112737  000116  164170  RCINT:  MOVB    #116,@#CSRC     ; clear group 2 IRR bit 6 (REPLY C)
001112  013777  164172  177660          MOV     @#DBRC,@IPNTR   ; get the word just received
001120  062767  000002  177652          ADD     #2,IPNTR        ; bump buffer pointer
001126  022767  177646  001000'         CMP     IPNTR,#IBUF+512.        ; buffer full ?
001134  001005                          BNE     10$             ; branch if buffer not full
001136  112737  000036  164170          MOVB    #36,@#CSRC      ; We're done.  Set group 2 IMR bit 6 (disable
                                                                ; interrupt)
001144  005267  177632                  INC     IFLAG           ; signal input complete
001150  000002                  10$:    RTI                     ; return from interrupt.


        001004'                         .END    START
```

SYMBOL TABLE

BRSB	= 164160	CSRC	= 164170	DBRB	= 164166	IBUF	000000R	RCINT	001104R
CSRA	= 164160	CSRD	= 164174	DBRC	= 164172	IFLAG	001002R	START	001004R
CSRB	= 164164	DBRA	= 164162	DBRD	= 164176	IPNTR	001000R		

Figure 5-4 Example of an Interrupt-Driven Input Program

CHAPTER 6
OPTIC ISOLATOR INTERFACE EXAMPLE

GENERAL DESCRIPTION

The DRV11-J can be used for industrial machine control, process control, monitoring applications, etc. When the module is used in industrial applications, the computer system must often operate in a hostile electrical environment. The DRV11-J may have to control or monitor components such as lamps, motors, relays and switches, all of which generate electrical noise. In such an environment, interfacing the DRV11-J to the user device(s) through optically coupled isolators may be necessary. Optic isolators are used to isolate electrically and/or convert signal levels between the user device(s) and the DRV11-J latched output drivers in output mode, or the unlatched Schmitt trigger buffers in input mode. The simplified schematic Figure 6-1 shows how the optic isolators may be connected to the DRV11-J for data input and output transfers. The choice of an appropriate optic isolator or optic isolator module depends upon the requirements of the specific application.

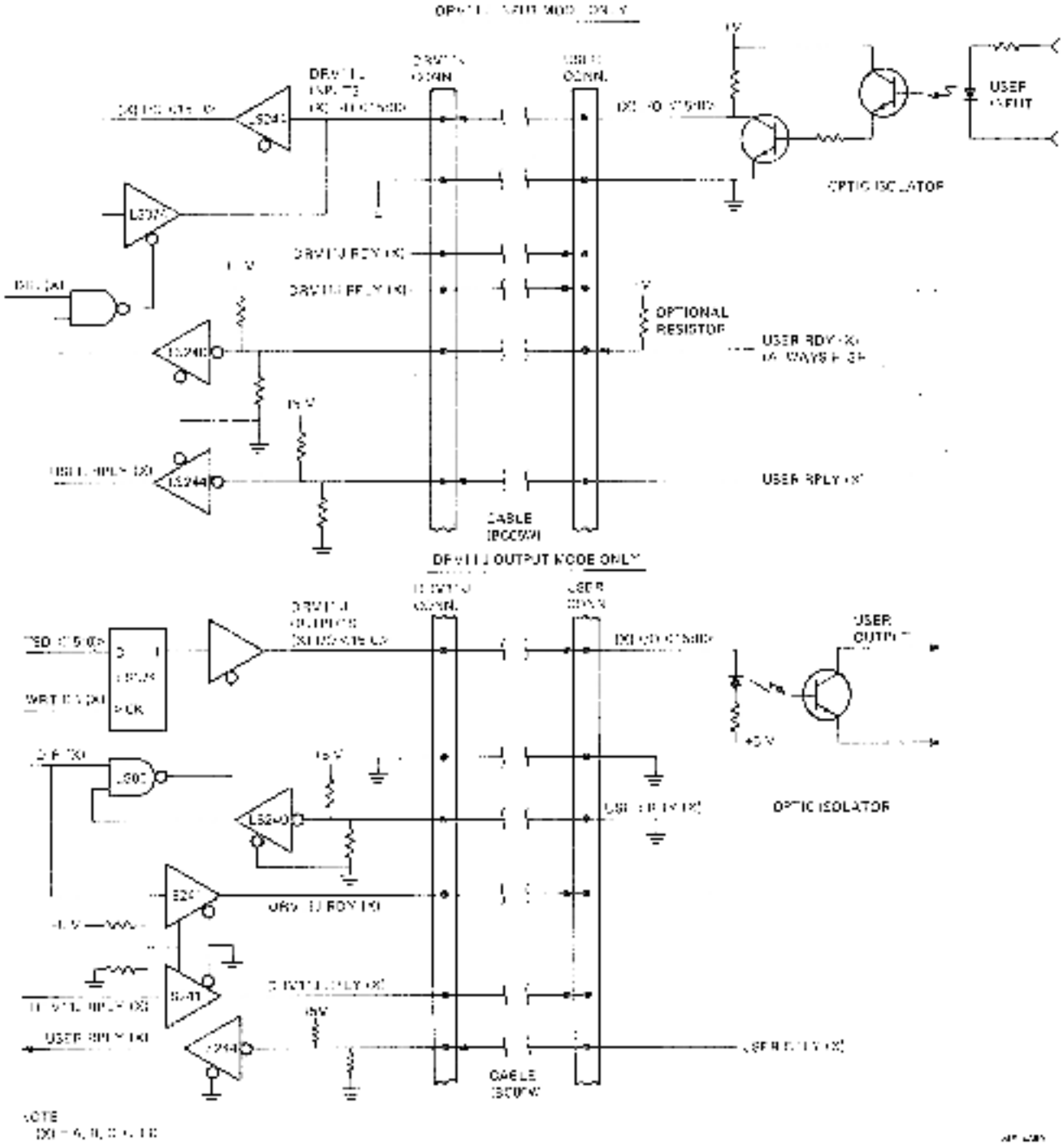


Figure 6-1 Example of an Optic Isolator Interface

DRV11-J Parallel Line Interface
User's Guide

Reader's Comments

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 or errors have you found in the manual? _____

_____

_____

_____

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

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

_____

_____

_____

☐ Please send me the current copy of the *Technical Documentation Catalog*, which contains information on the remainder of DIGITAL's technical documentation.

Name _____ Street _____

Title _____ City _____

Company _____ State/Country _____

Department _____ Zip _____

Additional copies of this document are available from:

Digital Equipment Corporation
Accessories and Supplies Group
Cotton Road
Nashua, NH. 03060

Attention: *Documentation Products*
Telephone: 1-800-258-1710

Order No. LK-DRV11-UG-002

Fold Here

Do Not Tear — Fold Here and Staple

digital

No Postage
Necessary
if Mailed in the
United States

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 33 MAYNARD, MA

POSTAGE WILL BE PAID BY ADDRESSEE

Digital Equipment Corporation
Educational Services Development and Publishing
200 Forest Street (MR1-2/T17)
Marlboro, MA 01752