CP1100/IEEE 488 INTERFACE 021-0230-00

TEKTRONIX®

TEKTRONIX®

CP1100/IEEE 488 INTERFACE 021-0230-00

INSTRUCTION MANUAL

Tektronix, Inc.
P.O. Box 500
Beaverton, Oregon 97077

Serial Number _____

First Printing MAR 1978 PRODUCED BY SPS DOCUMENTATION GROUP 070-2151-00

WARRANTY

Unless otherwise provided, Tektronix warrants that this instrument shall be free from defects in materials and workmanship under normal use and service for a period of one year from the date of installation. If this instrument fails during the warranty period under normal use and service because of a covered defect, Tektronix will repair or replace this instrument, provided that it is shipped freight prepaid to a designated service center. Tektronix will ship the repaired instrument or its replacement freight collect to the buyer. The risk of loss or damage and the duty to insure instruments in transit to or from Tektronix under this warranty shall be with the buyer.

LIMITATION OF WARRANTY

Any attempt by any person other than an authorized representative of Tektronix to repair this instrument or to make any modifications to the hardware or software, without the express prior written consent of Tektronix, will void all warranties. **Tektronix makes no other warranties, express or implied and specifically disclaims all warranties of merchantability or fitness for a particular purpose. Tektronix further disclaims any warranty that use of this instrument in conjunction with instruments supplied by buyer does not infringe any patent. Tektronix specifically excludes all special incidental and consequential damages for breach of any warranty. Tektronix' sole obligation for breach of any warranty shall be to repair or replace as set forth above.**

For fastest possible service, direct all requests for repairs and replacement parts to the Tektronix Field Office or representative in your area. Please include the instrument Type Number or Part Number and Serial Number with all requests for parts or service.

Specifications and price change privileges reserved.

Copyright © 1978 by Tektronix, Inc. Beaverton, Oregon. Printed in the United States of America. All rights reserved.

U.S.A. and foreign TEKTRONIX products covered by U.S. and foreign patents and/or patents pending.

TEKTRONIX and TEK are registered trademarks of Tektronix, Inc.

TABLE OF CONTENTS

SECTION 1 - INTRODUCTION **1-1**

Manual Organization 1-1
Interface Description (General) 1-1
IEEE 488 Interface Function Subsets 1-4
Related Information 1-5
Specifications 1-5

SECTION 2 - IEEE 488 BUS DESCRIPTION **2-1**

Introduction 2-1
Bus Signal Lines 2-1
The Data Bus 2-3
The Management Bus 2-3
The Transfer Bus (Handshake) 2-5
Bus Messages 2-5
Control Commands 2-7

SECTION 3 - INTERFACE REGISTERS AND INTERRUPT VECTORS **3-1**

Introduction 3-1
Programmable Registers 3-1
Register Addresses 3-3
Interrupt Vector Addresses 3-4
Register Descriptions 3-5
- Talker Data Buffer 3-6
- Listener Data Buffer 3-8
- Interrupt Control Register 3-10
- Bus Status Register 3-14
- Bus Control Register 3-17
- Interface Status Register 3-21
- Byte Counter Register 3-28
- Bus Address Register 3-29

Local Reset and Interface Clear 3-30

SECTION 4 - SOFTWARE PROTOCOLS **4-1**

Introduction 4-1
Programming Considerations 4-2
Addressing 4-2
Local Reset 4-2
The EOI Function 4-2
The Listen Enable Functions 4-3
Errors Detected and Reported 4-3
Implementation of the IEEE 488 Interface Functions 4-4
Controller Function 4-4
Usage of IFC and REN 4-5
Usage of ATN and TCS 4-5
Performing a Serial Poll 4-7
Performing a Parallel Poll 4-7
Passing Control 4-8
Interface Commands 4-8
Source Handshake Function 4-9
Acceptor Handshake Function 4-11
Talker Function 4-13
Listener Function 4-14
Service Request Function 4-17
Remote/Local Function 4-18
Parallel Poll Function 4-20
Device Clear Function 4-20
Device Trigger Function 4-21

OPERATION IN DMA MODE **4-22**

Introduction 4-22
Setting Up a DMA Transfer 4-23
Interface Status Register 4-23
Bus Status Register 4-27
Byte Counter Register 4-28
Bus Address Register 4-28

WARNING

The following servicing instructions are for use by qualified personnel only. To avoid personal injury, do not perform any servicing other than that contained in operating instructions unless you are qualified to do so.

SECTION 5 - SERVICING **5-1**

Strap Options 5-1
Register Addresses 5-1
Interrupt Vector Addresses 5-3
Operating Mode Jumpers (H1 and H2) 5-5
Calibration for DMA Transmission Rate (R402) 5-5
Cabling 5-5
Installation 5-7
General Maintenance 5-7
Static-Sensitive Parts 5-8

SECTION 6 - CIRCUIT DESCRIPTION **6-1**

Introduction (Block Diagram) 6-1
Schematic Diagrams 6-1
Bus Address and Control 6-2
Data/Selector Bank 6-5
Interrupt Control Bus 6-6
Interrupt and DMA State Machine 6-7
Introduction 6-7
Machine Operation 6-8
Source Handshake 6-15
Acceptor Handshake 6-18
Interface Registers 6-19
General 6-19
Attention Received 6-20
Attention Sent 6-20
Interface Clear 6-21

SECTION 7 - REPLACEABLE PARTS LIST 7-1

Electrical 7-1
Mechanical 7-7

SECTION 8 - DIAGRAMS

Block Diagram A
Block Diagram B
State Machine (DMA)

SCHEMATIC DIAGRAMS

UNIBUS Data/Selector Bank 1
BAR/CTR/ISR 2
BSR/BCR/ICR/Data Bus 3
Interrupt & DMA State Machine 4
UNIBUS Address 5
Interrupt Control Bus 6

COMPONENT LOCATORS

Located on reverse side of foldouts

CHANGE INFORMATION

@

SECTION 1

INTRODUCTION

Manual Organization

This manual contains two main divisions: programming and service.

When used with Tektronix software that includes a driver for this interface, the interface appears transparent to the operator and programmer. The first main division of this manual introduces the CP1100/IEEE 488 Interface and the IEEE 488 general purpose bus. Register addresses, register formats, interrupt handling, interrupt vector addresses, DMA transfers, and interface protocol are discussed to aid an assembly language programmer.

A warning page separates the second main division of this manual. Information following the warning page is intended for qualified service personnel only. The servicing instructions cover address and vector strap options, installation, cabling, and maintenance. Circuit description, parts lists, component locator, and schematic diagrams follow.

Interface Description (General)

The TEKTRONIX CP1100/IEEE 488 Interface is an assembly designed for use in TEKTRONIX CP1100-series controllers such as the TEKTRONIX CP1164. The interface includes the circuit card, a 2 meter cable (Tektronix Part Number 012-0809-00) to connect the interface to the IEEE 488 bus, and the instruction manual.

The interface can be used in other controllers that utilize the Digital Equipment Corporation UNIBUS, such as the PDP-11/04, PDP-11/05, PDP-11/10, PDP-11/34, PDP-11/35, PDP-11/40, and PDP-11/45; generally all of the PDP-11 models except the PDP-11/03 and other PDP-11 controllers based on the LSI-11 processor.

When this interface circuit card is properly installed, the controller can exchange data with and control instruments connected to a

general purpose bus (see Fig. 1-1) as specified by IEEE Standard 488-1975.

The CP1100/IEEE 488 Interface provides both direct memory access (DMA) and processor controlled data transfers (program) to or from the IEEE 488 bus. DMA transfer rates up to 400 K bytes/sec can be achieved by this interface. To reduce the load on the processor, the source and acceptor handshakes are implemented in hardware. Other DMA features of the interface include:

1. Capability to pack two bytes into a word. This is useful for acquiring data from digitizers which have more than eight bits per word, or for transferring command text strings.

2. Capability to swap the order in which bytes are sent or received when data is packed. This assures compatibility with both high byte first or low byte first 16-bit word transfers.

3. Capability to address memory space in either direction by incrementing or decrementing a Bus Address Register.

4. Operation in the burst (hog) or non burst (non hog) mode, allowing the user to make best use of DMA cycle stealing. For either mode, protection is built in to allow the processor to take control if the IEEE 488 bus should hang (or cycles too slowly).

Eight addressable registers and eight interrupt vectors are provided by the interface. The interface register addresses and the vector address range are user-set by selecting the desired jumper (strap) positions on the interface card. Strap selection, register addresses, register formats, and interrupt vectors are discussed in other sections of this manual.

With appropriate software, the controller can implement the full capabilities allowed by the interface functions specified by IEEE Standard 488-1975, except for responding to a parallel poll conducted by another controller on the bus. The CP1100/IEEE 488 Interface can perform a parallel or serial poll of instruments on the IEEE 488 bus when the CP1164 controller is acting as the controller-in-charge. See Fig. 1-1 for a system block diagram using this interface.

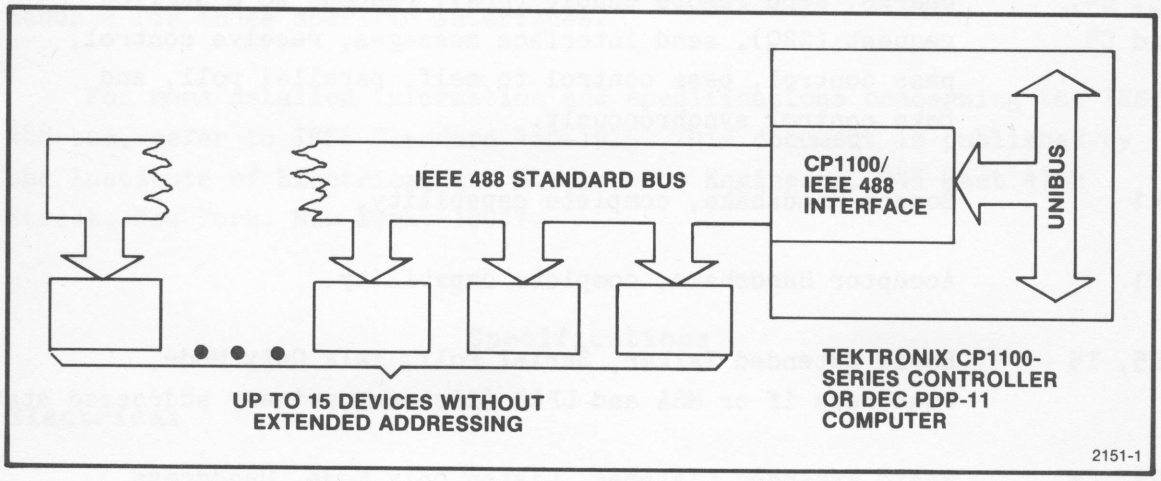


Fig. 1-1. Block diagram of an IEEE 488-compatible system using the CP1100/IEEE 488 Interface.

Up to 15 devices (including this interface) can be electrically connected to the IEEE 488 bus. The total bus length allowed by the standard is 20 meters. Normally, the devices are separated by cables no more than 2 meters in length. If this is not the case, a bus termination network must be provided for each 2 meters of cable length. If the required terminations are lumped, performance of the bus may be degraded.

IEEE Standard 488-1975 designates 31 primary address codes and 31 secondary address codes for talkers and listeners using the bus to communicate; this addressing scheme provides a total of 961 addresses.

IEEE 488 Interface Function Subsets

The IEEE 488 interface function subsets (defined by the standard) that can be implemented by a controller using this interface and the appropriate software are:

C1, C2, C3, C4, and C5	System controller, send interface clear (IFC) and take charge, send remote enable (REN), respond to a service request (SRQ), send interface messages, receive control, pass control, pass control to self, parallel poll, and take control synchronously.
SH1	Source Handshake, complete capability.
AH1	Acceptor Handshake, complete capability.
TE5, T5	Basic Extended Talker, Serial Poll, Talk Only Mode, Unaddress if or MSA and LPAS (listener primary addressed state).
LE3, L3	Basic Extended Listener, Listen only Mode, Unaddress if MSA and TPAS (Talker primary addressed state).
SR1	Service Request, complete capability.
RL1	Remote-Local, complete capability.
PP0	Parallel Poll, no capability (does not respond).
DC1	Device Clear, complete capability.
DT1	Device Trigger, complete capability.

My Listen Address (MLA), My Talk Address (MTA), and My Secondary address (MSA) are all established by software.

Related Information

Tektronix, Inc. provides manuals for the CP1100-series controllers. Digital Equipment Corporation publishes manuals and handbooks for its line of PDP-11 computers. Examples are the PDP-11 Processor Handbook and PDP-11 Peripherals Handbook. Information on device interfaces for TEKTRONIX IEEE 488-compatible instruments is available in Tektronix manuals for those specific interfaces.

For more detailed information and specifications concerning the IEEE 488 bus, refer to IEEE Standard 488-1975. This document is published by the Institute of Electrical and Electronics Engineers, 345 East 47th Street, New York, New York, 10017.

Specifications

Electrical

Power Requirement:

This interface requires 1.5 amperes at +5 Vdc (typical) from the controller mainframe.

Signal Levels:

All signal levels are TTL compatible. Refer to controller mainframe specifications for Unibus signal levels and IEEE Standard 488-1975 for signal levels on the IEEE 488 bus.

Environmental

The interface operates within the same environmental limits as the controller. Refer to specifications in the controller manual.

Mechanical

The interface is contained on a quad-height, extended length card and requires a full slot in the peripheral backplane of the controller (see Fig. 9.2, PDP-11 Peripherals Handbook).

SECTION 2

IEEE 488 BUS DESCRIPTION

Introduction

All instruments (devices) designed for use with the IEEE 488 bus must conform to IEEE Standard 488-1975 (ANSI MC1.1-1975), Standard Digital Interface for Programmable Instrumentation. IEEE Standard 488-1975 specifies the mechanical, electrical, and functional elements of the digital interface system.

All of the interface functions required to allow an instrument to control, talk, listen or to simply monitor the activity on the bus are contained within that instrument. The interconnecting bus is entirely passive. A typical interface system is illustrated in Fig. 2-1.

When a controller is connected to the bus, it can control the talk and listen functions when programmed to do so. Only one active talker is designated for any selected time interval. A talker or listener is not considered active until it is assigned to perform a talk or listen role by means of an address code sent by the controller. Systems can have more than one controller; when this is the case, the interface functions (digital circuitry) within the individual controllers can be programmed to allow each to take turns as controller-in-charge.

Bus Signal Lines

The signal lines are functionally divided into three component buses: an eight-line data bus, a three-line transfer bus, and a five-line management bus for a total of sixteen active signal lines (see Fig. 2-1).

The relationship between the binary logic states and the electrical state levels present on the signal lines is as follows:

Logical 1 corresponds to a low voltage level ($\leq$+0.8 V) and the signal line is asserted.

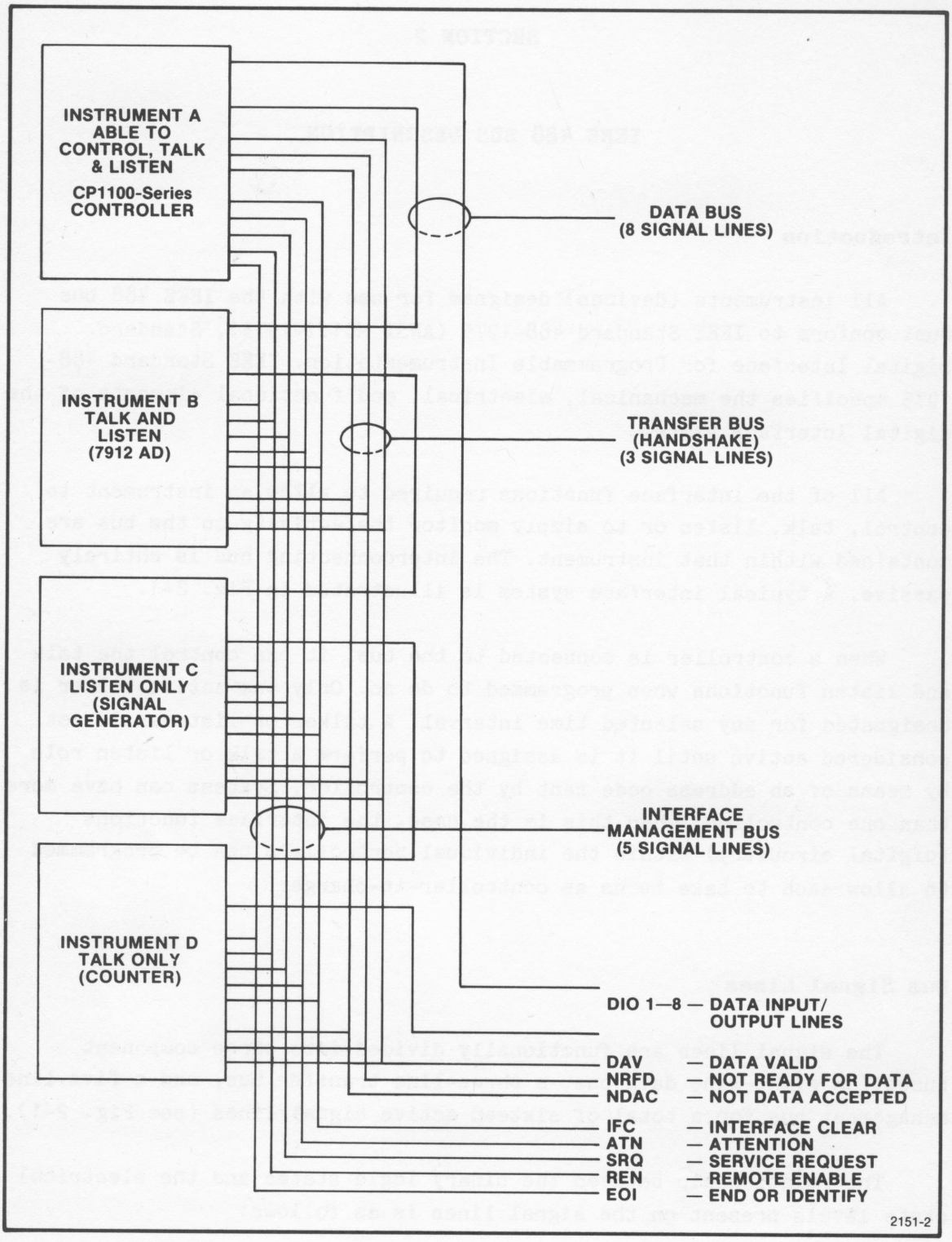


Fig. 2-1. A typical system showing the IEEE 488 bus organization.

Logical 0 corresponds to a high voltage level ($\geq$+2.0 V) and the signal line is not asserted.

The electrical states are based on standard TTL levels where the power source does not exceed +5.25 Vdc (referenced to logic ground).

The Data Bus

The data bus contains eight bidirectional signal lines, DIO1 through DIO8. One byte of information (eight bits) is transferred over the bus at a time. DIO1 represents the least significant bit in the byte and DIO8 represents the most significant bit in the byte. Data is transmitted in a byte-serial, bit-parallel format over the data bus.

Since the bus operates asynchronously, the transfer rate over the data bus is a function of the slowest instrument taking part in a data transfer at any one time. Data bytes can be formatted in ASCII (ISO 7-bit) code representation with or without parity, or they can be formatted in machine dependent binary code. The term "machine dependent binary code" refers to an internal binary format used by a device to store certain programs and data.

The Management Bus

The management bus is a group of five signal lines which are used to control data transfers over the data bus. The basic signal definitions for the management bus are as follows:

Interface Clear (IFC). The IFC signal line is asserted by the system controller to place all interface circuitry in a predetermined quiescent (power on) state.

The system controller is the only source for this signal. IEEE 488-1975 designates only three interface messages (universal commands) to be recognized during the time that IFC is asserted, device clear (DCL), local lockout (LLO), or parallel poll unconfigure (PPU).

Attention (ATN). The ATN signal line is asserted by a controller when peripheral instruments connected to the bus are being assigned as

talkers or listeners and for other interface control traffic. As long as the ATN signal line is asserted (ATN = 1), only instrument address codes and control messages are transferred over the data bus. When the ATN signal line becomes unasserted only those instruments designated as talkers and listeners can take part in the data transfer. The controller is the only source of the ATN signal.

Service Request (SRQ). Any peripheral instrument connected to the bus can request the attention of the controller by asserting the SRQ line.

The controller responds by asserting ATN and executing a serial poll to determine which instrument is requesting service. After the peripheral instrument requesting service is found (an instrument requesting service identifies itself by asserting its DIO7 line after being addressed) program control can be transferred to a service routine for that instrument. When the service routine is completed, program control returns to the main program. The SRQ signal line is released (unasserted) when the instrument requesting service is polled.

Remote Enable (REN). The REN signal line is asserted by the system controller whenever the interface system is operating under remote program control. The REN signal is used (in conjunction with other control messages) to cause an instrument on the bus to select between two alternate sources of programming data. A remote-local interface function is used to indicate to an instrument that either input information from the front-panel controls (local) or corresponding input information from the interface (remote) is to be used. Some instruments are designed so that specifically designated front-panel controls are enabled during a remote programming sequence.

End or Identify (EOI). The EOI signal can be used by a talker to indicate the end of a data transfer sequence. The talker asserts the EOI signal line as the last byte of data is transmitted. In this case, EOI is essentially a ninth data line and must observe the same setup times as the DIO lines. When the controller is programmed to listen, it assumes that a data byte received is the last byte in the transmission (if the EOI signal line has been asserted). When the controller is talking, it may assert the EOI signal line as the last byte is transferred. The EOI signal is also used in conjunction with the ATN signal if a parallel polling sequence is conducted by the controller. EOI is not used during

serial polls.

The Transfer Bus (Handshake)

A handshake sequence is executed between a talker and all designated listeners via the transfer bus each time a data byte is transferred over the data bus. The transfer bus signal lines are defined and operate as follows (see Fig. 2-2 for a basic timing relationship between these signals):

Not Ready for Data (NRFD). As long as the NRFD signal line is asserted, it indicates that one or more assigned listeners are not ready to receive the next data byte. When all of the assigned listeners for a particular data transfer have released NRFD, the NRFD signal line is unasserted. This tells the talker that it may place the next data byte on the data bus.

Data Valid (DAV). The DAV signal line is asserted by the talker after the talker places a valid data byte on the data bus. A low voltage level for the asserted DAV signal tells each assigned listener to accept the byte presently on the data bus. The talker is inhibited from asserting the DAV signal line as long as the NRFD signal line is asserted.

Not Data Accepted (NDAC). The NDAC signal line is asserted (held low true) by each assigned listener until the listener accepts the data byte currently on the data bus. When all assigned listeners have accepted the data byte, the NDAC signal line is released (unasserted). This tells the talker to remove the data byte from the data bus. The DAV signal line is unasserted when all assigned listeners have accepted the data byte.

Bus Messages

Messages on the bus are either interface messages or device dependent messages and are coded according to the ASCII (ISO 7-bit) code. Interface messages are used to manage the interface functions of the instruments; for example, they are sent with attention (ATN) asserted and designate a specific instrument to talk and others to listen. Device dependent messages are sent over the bus with ATN unasserted and are not

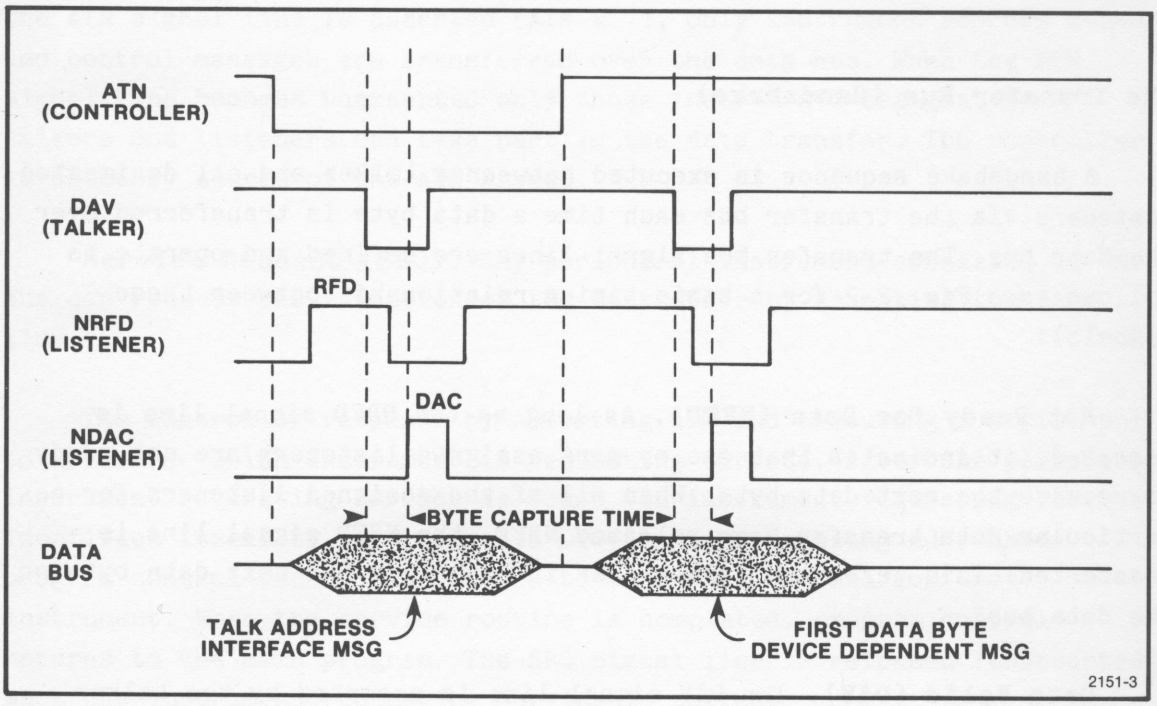


Fig. 2-2. A typical handshake timing sequence (idealized). Byte capture time is dependent on the slowest instrument involved in the handshake.

used by the instrument interfaces to change their state or configuration. Device dependent messages are passed on to the device functions of the instruments and can be considered to be data, such as a voltage reading by a digital multimeter or the functional setting of a front panel control.

Control Commands

Octal Code	Command	Meaning
024	DCL	Device Clear
010	GET	Group Execute Trigger
001	GTL	Go To Local
021	LLO	Local Lock Out
005	PPC	Parallel Poll Configure
16X	PPD	Parallel Poll Disable (see note)
14X	PPE	Parallel Poll Enable (see note)
025	PPU	Parallel Poll Unconfigure
004	SDC	Selected Device Clear
031	SPD	Serial Poll Disable
030	SPE	Serial Poll Enable
011	TCT	Take Control
077	UNL	UNListen
137	UNT	UNTalk

NOTE

The exact binary format for the PPE command is: 110SNNN where "NNN" is a number in the range 0 to 7 designating which data line to send the parallel poll response on, and "S" is the sense returned on data line NNN for a true response.

The exact binary format for the PPD command is 111DDDD, where D specifies a don't care bit that must be sent as a 0 but does not need to be decoded by the receiving device.

GET, GTL, PPC, and SDC, are sent to an instrument or group of instruments that have been addressed to listen. The TCT command is sent to a controller that has been addressed to talk.

DCL, LLO, SPE, SPD, and PPU are universal commands and affect all instruments on the bus.

The UNT command disables the talk mode of a previously selected talker, while the UNL command disables the listen mode of all previously selected listeners.

SECTION 3

INTERFACE REGISTERS AND INTERRUPT VECTORS

Introduction

This interface has eight addressable (programmable) registers and eight interrupt vectors that allow the programmer to control a system of instruments interfaced according to IEEE Standard 488-1975. Through the interface, the program can act as the system controller or it can pass control back and forth with another device that can act as controller-in-charge. The program can also act as a talker only or as a listener only (to monitor activity on the bus). When the program is acting as the controller in charge it can address devices, send universal and addressed commands, and transfer device-dependent messages. This interface can conduct a serial or parallel poll of instruments on the bus, but will not respond to a parallel poll conducted by another device. It can, if so programmed, respond to a serial poll.

Programmable Registers

Table 3-1 illustrates the data format and bus addresses (octal) for the eight interface registers. The Talker Data Buffer (TDB) and the Listener Data Buffer (LDB) are data ports to and from the IEEE 488 bus. The Interrupt Control Register (ICR) provides interrupt management bits. The Bus Status Register (BSR) and the Bus Control Register (BCR) provide a way to monitor and set the IEEE 488 bus control lines; these two registers also include interrupt clearing and interface mode control bits. The Interface Status Register (ISR) supplies information about direct memory access (DMA) operation. The Byte Counter Register (CTR) and the Bus Address Register (BAR) are used when DMA transfers are implemented between the controller bus and the IEEE 488 bus.

TALKER DATA BUFFER (TDB) 1AAA00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOI OR DO16	DO15	DO14	DO13	DO12	DO11	DO10	DO9	DI08	DI07	DI06	DI05	DI04	DI03	DI02	DI01

LISTENER DATA BUFFER (LDB) 1AAA02

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOI	ATN	REN	REN FALSE OCC					DI08	DI07	DI06	DI05	DI04	DI03	DI02	DI01

INTERRUPT CONTROL REGISTER (ICR) 1AAA04

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DONE	ATN	END OCC	REN FALSE OCC	IFC OCC				RDY	RDY INTR ENBL	DONE INTR ENBL	SRQ INTR ENBL	ATN INTR ENBL	END INTR ENBL	REN INTR ENBL	IFC INTR ENBL

BUS STATUS REGISTER (BSR) 1AAA06

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NO EOI DMA		END INTR CLR	REN INTR CLR	IFC INTR CLR	DMA INIT	LOCAL RESET		NRFD	NDAC	DAV	SRQ	ATN	EOI	REN	IFC

BUS CONTROL REGISTER (BCR) 1AAA10

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TCS	LIST W ATN	LIST W/O ATN	XBA 17	XBA 16				NRFD OUT	NDAC OUT	DAV OUT	SRQ OUT	ATN OUT	EOI OUT	REN OUT	IFC OUT

INTERFACE STATUS REGISTER (ISR) 1AAA12

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ERR	TIME ERR	WRITE ERR	DMA ERR	DMA ABORT				DMA DONE	DMA INTR ENBL	SWAP BYTE ORDER	HOG MODE	PACK MODE	DMA IN/ OUT	NO INCR/ DECR	GO

BYTE COUNTER REGISTER (CTR) 1AAA14

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

BUS ADDRESS REGISTER (BAR) 1AAA16

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

TABLE 3-1
CP1100/IEEE 488 Interface Registers

Register Addresses

The address format for the eight interface registers is illustrated in Fig. 3-1.

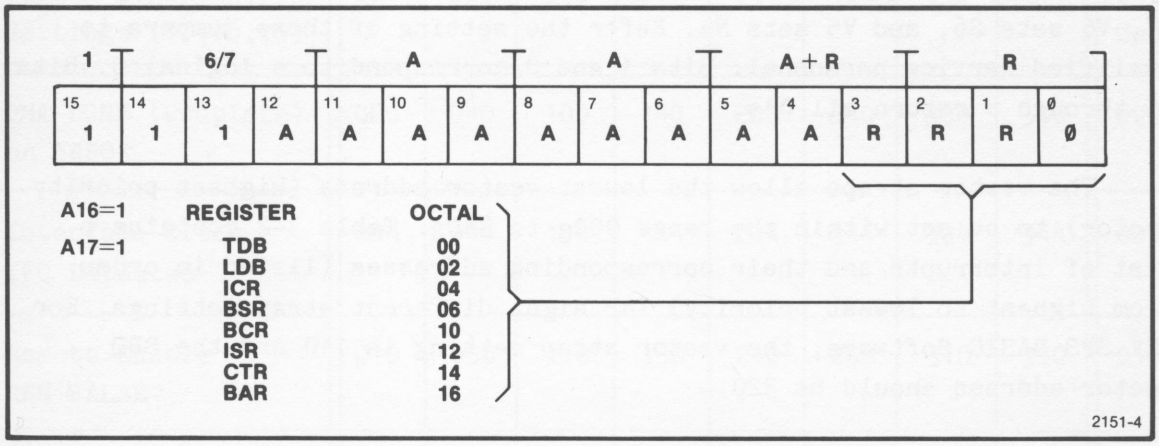


Fig. 3-1. Register address format.

Address bits 15 through 13 are hardwired for a logical 1. Bits 12 through 4 are set to a logical 1 or 0 by a group of nine jumpers as shown in the Servicing Section of this manual. Refer the setting of the jumpers to qualified service personnel. These nine jumpers allow the address of the lowest numbered register (1AAAOO-TDB) to be set withing the range 160000_8 through 177760_8. When bits 12 through 4 have been set, bits 3 through 0 are decoded to address (select) the desired register in the interface. Bit 0 is equivalent to logical 0; consequently, all registers have an even address.

For TEK SPS BASIC Software, the upper bits of the register address decoding logic are set for 164100_8. TEK SPS BASIC Software allows the use of more than one interface. The address straps of each interface can be set to create successively numbered addresses. For example:

Interface 0	164100_8 through 164116_8
Interface 1	164120_8 through 164136_8
Interface 2	164140_8 through 164156_8

Interrupt Vector Addresses

The CP1100/IEEE 488 interface has eight interrupts. The vector address format is illustrated in Fig. 3-2. A single group of three jumpers (V7, V6, and V5) set the S bits of the vector address. V7 sets S7, V6 sets S6, and V5 sets S5. Refer the setting of these jumpers to qualified service personnel. Bits 1 and 0 correspond to a logical 0. Bits 15 through 8 return all 0's.

The vector straps allow the lowest vector address (highest priority vector) to be set within the range 000_8 to 340_8. Table 3-2 contains a list of interrupts and their corresponding addresses (listed in order from highest to lowest priority) for eight different strap settings. For TEK SPS BASIC Software, the vector strap setting is 110 and the SRQ vector address should be 320.

Register address 1AAA00 can be read. When read, the address returned is the vector for the last interrupt granted.

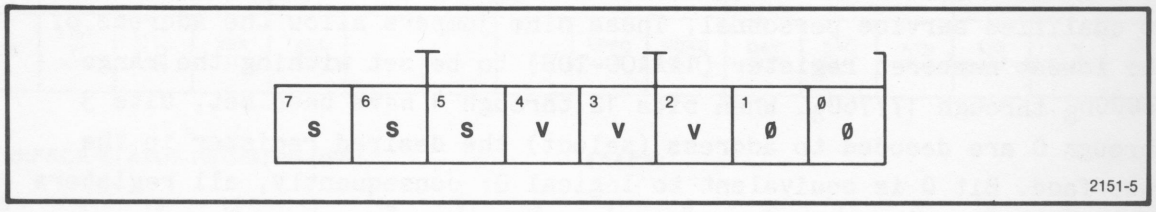


Fig. 3-2. Interrupt vector address format.

TABLE 3-2
Interrupt vectors and their corresponding addresses for different vector strap settings.

Vector Strap Settings (V7,V6,V5)	000	001	010	011	100	101	110	111	V-bit Code
DMA DONE (Complete) or ERROR	000	040	100	140	200	240	300	340	000
Interface Clear IFC Detected	004	044	104	144	204	244	304	344	001
Remote Enable REN FALSE	010	050	110	150	210	250	310	350	010
END of Message DAV and EOI Asserted	014	054	114	154	214	254	314	354	011
Service Request SRQ Asserted	020	060	120	160	220	260	320	360	100
Attention ATN Asserted	024	064	124	164	224	264	324	364	101
Data Ready (RDY) in LDB	030	070	130	170	230	270	330	370	110
DONE (Source Handshake Completed)	034	074	134	174	234	274	334	374	111 (NONE)

Register Descriptions

This part of the manual contains a functional description for the valid bits in each register to aid the programmer and service person. It describes the bit functions and their relationships to each other and discusses how and under what circumstances each bit is set or cleared (reset). In this section a RESET instruction refers to a LOCAL RESET function, a controller RESET, or whenever INIT is asserted on the controller's internal bus.

TALKER DATA BUFFER (TDB) **1AAA00**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOI OR DO16	DO15	DO14	DO13	DO12	DO11	DO10	DO9	DI08	DI07	DI06	DI05	DI04	DI03	DI02	DI01

Talker Data Buffer (TDB). The Talker Data Buffer is used to write to the IEEE 488 bus in any of the formats described below. The TDB is a write-only register. However, performing a read operation on register address 1AAA00 returns the vector address for the last interrupt granted. Reading register address 1AAA00 at system initialization time allows the programmer to determine how the vector address straps are set.

1. For a non-DMA transfer from the controller to a device on the IEEE 488 bus, data is written into this register. The low byte is latched and output under control of the interface source handshake logic. The MSB (DO16) is also latched; if set, the interface asserts the EOI message. Bits 14 through 8 are ignored (not latched or output on the bus).

2. For a DMA transfer to the IEEE 488 bus in the "unpacked" mode, the DMA IN/OUT and PACK MODE bits in the Interface Status Register must be cleared. The interface DMA control logic places a word in this register. The low byte is latched and output on the bus by the interface source handshake logic; the high byte is ignored.

3. For a DMA transfer to the IEEE 888 bus in the "packed" mode, the DMA IN/OUT bit must be cleared and the PACK MODE bit set. The interface DMA control logic places a word in this register. Either the high byte or the low byte is latched first, depending on the condition of the SWAP BYTE ORDER and NO INCR/DECR bits in the Interface Status Register (ISR), the setting of jumper H1, and the value of the least significant bit in the Bus Address Register (BAR). The byte that is latched is output on the bus by the interface source handshake logic. The

DMA control logic then performs another read cycle to latch and output the other byte. The user should not write to or read from register 1AAA00 while a DMA transfer is taking place.

The data written to the bus in any mode is removed from the bus before the source handshake cycle is completed if a RESET instruction is executed, if ATN is asserted by another controller on the bus, or if IFC is asserted. If an error condition (bits 15 through 11 in the ISR) occurs before the handshake is completed, toggling the NDAC OUT bit in the Bus Control Reigster (BCR) is the preferred method of clearing the handshake and removing data from the bus (see Section 4, Software Protocol). The DONE bit in the Interrupt Control Register (ICR) is set automatically when the handshake cycle is completed.

Bits 15 - 8: **DO16-DO9 (write only).**

Function: High byte data for the DMA "packed" mode. Bit 15 can be programmed (if desired) for the EOI function in the non-DMA mode only. When set for EOI, it causes the EOI message to be sent at the same time as the last data byte.

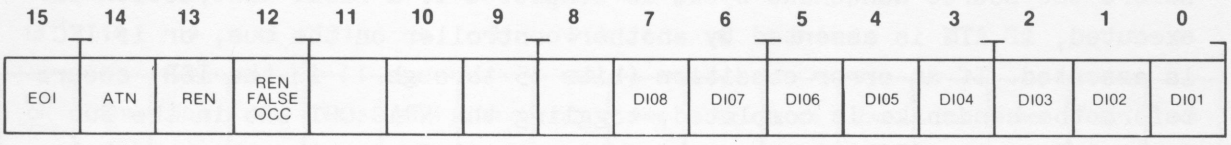
Set: By writing a 1 into the desired bit. These bits are cleared when outgoing data is accepted by all assigned listeners or otherwise removed from the bus (see above).

Bits 7 - 0: **DIO8-DIO1 (write only).**

Function: Low byte data for the DMA "packed" mode. Normal data output for the other modes.

Set: By writing a 1 into the desired bit. These bits are cleared when outgoing data is accepted by all the assigned listeners, by a RESET instruction, when IFC is asserted, or if another controller asserts ATN before the handshake is completed.

LISTENER DATA BUFFER (LDB) **1AAA02**



Listener Data Buffer (LDB). The Listener Data Buffer is a read only register and receives data sent by other devices on the IEEE 488 bus. This register is normally read in the non-DMA mode. However, when using the DMA control logic to send data on the IEEE 488 bus (DMA IN/OUT bit in the ISR must be cleared), the LDB can be used to read data as transmitted by this interface. The LISTEN control bits (bits 14 and 13 in the ISR) must be set if the acceptor handshake is to be controlled. Reading the LDB while the DMA control logic is sending allows for self-check procedures.

To accept data (listen to devices on the IEEE 488 bus) in a DMA mode, the interface acceptor handshake logic is enabled as soon as the DMA IN/OUT and GO bits in the ISR are set. For non-DMA listening, the LISTEN bits in the ISR control the enabling of the acceptor handshake function. The interface performs the acceptor handshake cycle and sets the RDY bit in the Interrupt Control Register (ICR) to indicate that valid data can be read from the LDB.

When the LDB is read, it returns the current value of the IEEE 488 data bus lines, the status of the EOI, ATN, and REN control lines, and the condition of the REN FALSE OCC bit. Reading the LDB also clears the RDY bit in the Interrupt Control Register.

Bits 15 - 12: **EOI, ATN, REN, REN FALSE OCC, respectively (read only).**

Function: EOI, ATN, and REN are read directly from the bus and indicate the asserted or unasserted states for these control lines. EOI asserted indicates the last data byte sent or received. ATN asserted indicates an interface message (control command) was sent by this interface or by another controller. REN asserted indicates that the system is under remote control. REN FALSE OCC, when read as a 1, indicates that all messages should be interpreted as data sent or received under local mode control (not remote). The REN FALSE OCC bit can be used for non-DMA listening programs if the REN INTR ENBL bit (1) in the ICR is not set. Bits 15 through 12 are ignored when this register is read by the DMA control logic. See Section 4 for more information concerning the REN FALSE OCC bit.

Bit 12: **REN FALSE OCC (read only).**

Set: By an internal flip-flop when the REN control line goes from its asserted to unasserted state.

Cleared: By a RESET instruction, or by writing a 1 into the REN INTR CLR bit (12) in the Bus Status Register (BSR). This bit is set and cleared at the same time as the REN FALSE OCC bit in the Interrupt Control Register.

Bits 7 - 0: **DIO8 - DIO1 (read only).**

Function: Read directly from the IEEE 488 data bus in any mode. When read, indicates the current values of the data lines; a 1 indicates that the corresponding bus line is asserted. A talker sending data to this interface does not normally remove the input data until the acceptor handshake cycle is completed.

INTERRUPT CONTROL REGISTER (ICR) **1AAA04**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DONE	ATN	END OCC	REN FALSE OCC	IFC OCC				RDY	RDY INTR ENBL	DONE INTR ENBL	SRQ INTR ENBL	ATN INTR ENBL	END INTR ENBL	REN INTR ENBL	IFC INTR ENBL

Interrupt Control Register (ICR). The Interrupt Control Register provides the interrupt management functions. Two bits are used to indicate completion of the source and acceptor handshake functions.

Bits in this register operate in pairs to generate interrupts related to conditions on the IEEE 488 bus control lines. Low order bits (0-6) enable the interrupts. High order bits (7, 11-15) indicate that interrupt conditions occurred. An interrupt occurs if an enable bit and its associated condition bit are both 1, regardless of which was set first. Once an interrupt occurs, another interrupt for the same condition cannot occur until one of the two bits is cleared and set to 1 again. If more than one interrupt condition is pending when the interrupt request was recognized, the interrupt with the lowest numbered vector address (highest priority interrupt) will be the one sent to the controller program. Refer to the Servicing Section of this manual for additional information concerning the interrupt vectors.

Bit 15: **DONE (read only).**

Function: Data byte transfer complete. Set true when powered up. After data is written into the TDB, the interface performs the source handshake cycle and sets this bit when data has been accepted by all of the assigned listeners (NDAC line is released). The next data byte can then be loaded into the TDB. The DONE interrupt is generated if this bit is set and its associated enabling bit has been set.

Set: By the NDAC bus line going from an asserted to an unasserted state at the end of a handshake cycle, by a RESET instruction, when IFC is asserted, or when another controller on the bus asserts ATN.

Cleared: By a correct write to the TDB. This bit will not clear if the DAV bus line is already asserted or remains asserted when NDAC is released.

Bit 14: ATN (read only).

Function: This bit is read directly from the bus and generates the ATN interrupt if its associated enabling bit has been set. A 1 indicates that the ATN line is asserted.

Bit 13: END OCC (read only).

Function: Essentially indicates the END of a device dependent message. Generates the END interrupt if its associated enabling bit has been set.

Set: By an internal latch when a talker on the IEEE 488 bus asserts the DAV and EOI lines simultaneously with the last data byte (ATN unasserted).

Cleared: By a RESET instruction or by writing a 1 into the END INTR CLR bit (13) in the BSR.

Bit 12: REN FALSE OCC (read only).

Function: For interrupt driven program. Detects when the IEEE 488 bus leaves the remote control mode and generates the Remote Enable (REN) Cleared interrupt vector if its associated enabling bit has been set.

Set: By an internal flip-flop when the REN bus line goes from its asserted to its unasserted state.

Cleared: By a RESET instruction or by writing a 1 into the REN INTR CLR bit (12) in the BSR.

Bit 11: **IFC OCC (read only).**

Function: Detects when the IFC bus line becomes asserted and generates the IFC interrupt if its associated enabling bit has been set.

Set: By an internal flip-flop when the IFC bus line goes from its unasserted state to its asserted state.

Cleared: By a RESET instruction or by writing a 1 into the IFC INTR CLR bit (11) in the BSR.

Bit 7: **RDY (read only).**

Function: When set at the end of an acceptor handshake cycle, this bit indicates that a valid data byte is available in the LDB. Interface must be in a listen mode (LIST W ATN or LIST W/O ATN). For a DMA listen mode, the DMA IN/OUT bit (2) in the ISR must be set. When this bit is set it generates the RDY interrupt if its enabling bit has been set. NOTE: Do not set the DONE or RDY interrupt enabling bits for a DMA transfer.

Set: By an internal flip-flop when DAV assumes its asserted state at the beginning of a handshake cycle. Data is normally placed on the bus by a talker before DAV is asserted.

Cleared: When the LDB is read, by a RESET instruction, or when the IFC line is asserted.

Bits 6-0: **INTR ENBL (Interrupt Enable) bits; RDY, DONE, SRQ, ATN, END, REN, and IFC, respectively (write or read).**

Function: When written, these bits are stored in a common octal flip-flop circuit with a common clear input. Each bit can be set (interrupt unmasked) or cleared (interrupt masked) by writing a 1 or 0 into the desired bit (1 equals set, 0 equals clear). The SRQ interrupt vector is generated if bit 4 is set and the SRQ bus line is asserted. The RDY INTR ENBL and the DONE INTR ENBL bits must not be set for a DMA transfer. All of these bits are cleared by a RESET instruction. The assertion of the IFC bus line does not clear these bits.

BUS STATUS REGISTER (BSR) **1AAA06**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NO EOI DMA		END INTR CLR	REN INTR CLR	IFC INTR CLR	DMA INIT	LOCAL RESET		NRFD	NDAC	DAV	SRQ	ATN	EOI	REN	IFC

Bus Status Register (BSR). The Bus Status Register essentially monitors the IEEE 488 bus management and handshake control lines. Bits included in this register are two DMA functions, three interrupt clearing bits, and a local reset function.

Bit 15: **NO EOI DMA (write or read).**

Function: This bit is provided for DMA transfers:
1) If bit 15 is set for a DMA transfer from the IEEE 488 bus to the controller memory or to another peripheral, the transfer is not terminated by the EOI message being received with a data byte. The DMA control logic continues to accept data until the Byte Counter Register (BCR) overflows or the DMA transfer is aborted.
2) If bit 15 is set for a DMA transfer from the controller memory or another peripheral to the IEEE 488 bus, the interface will not assert the EOI bus line at the same time as the last data byte. If this bit is clear, EOI will be asserted with the last data byte.
3) If the ATN OUT bit (3) in the BCR is set and a DMA transfer to the IEEE 488 bus is initialized, the assertion of the EOI bus line is automatically inhibited (bit 15 does not have to be set).

Set: By writing a 1 to the bit location.

Cleared: By writing a 0 into this bit or by a RESET instruction.

Bits 13 - 11: **END INTR CLR, REN INTR CLR, and IFC INTR CLR, respectively (gated write pulse only).**

Function: When written these bits are not stored or latched. When a 1 is written to the desired bit, it clears its corresponding interrupt condition bit (13, 12, or 11) in the ICR. Write pulse duration is about 50 nanoseconds.

Bit 10: **DMA INIT (gated write pulse only).**

Function: To be set before a DMA operation. This bit is not stored or latched. When a 1 is written to this bit it clears the ERR, TIME ERR, WRITE ERR, DMA ERR, DMA ABORT, DMA DONE, and GO bits (15, 14, 13, 12, 11, 7, and 0) in the ISR. Write pulse duration is about 50 nanoseconds.

Bit 9: **LOCAL RESET (gated write pulse only).**

Function: This bit is not stored or latched. When a 1 is written to this bit, it performs the same operation as occurs when a controller RESET instruction is executed. It is used to re-initialize this interface without affecting other interfaces (peripherals) within the system. Write pulse duration is about 50 nanoseconds. Asserting this bit affects the eight programmable (addressable) registers as follows (see Table 3-3):

TDB: Resets (clears) the EOI function associated with bit 15 and removes output data from the IEEE 488 bus.

LDB: Clears the REN FALSE OCC bit (12).

ICR: Sets the DONE bit (15). Clears the following bits: END bit (13), REN FALSE OCC bit (12), IFC OCC bit (11), RDY bit (7), and the INTR ENBL (interrupt enabling) bits (6 through 0).

BSR: Clears the NO EOI DMA bit (15).

BCR: Clears TCS (15), LIST W ATN (14), LIST W/O ATN (13), XBA17 (12), XBA16 (11), and resets the IEEE 488 control lines to their idle state.

ISR: Resets (clears) all the bits in this register.

CLR: Does not affect this register.

BAR: Does not affect this register.

Bits 7 - 0: **NRFD, NDAC, DAV, SRQ, ATN, EOI, REN, and IFC, respectively (read only).**

Function: Not Ready for Data, Not Data Accepted, Data Valid, Service Request, Attention, End or Identify, Remote Enable, and Interface Clear, respectively. These bits monitor the IEEE 488 control lines and are read directly from the bus. They are not stored or latched. When read, a 1 indicates that a control signal is received and that the corresponding bus line is asserted.

BUS CONTROL REGISTER (BCR) **1AAA10**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TCS	LIST W ATN	LIST W/O ATN	XBA 17	XBA 16				NRFD OUT	NDAC OUT	DAV OUT	SRQ OUT	ATN OUT	EOI OUT	REN OUT	IFC OUT

Bus Control Register (BCR). The Bus Control Register provides a way to set and monitor the IEEE 488 management and handshake control lines. This register also includes interface mode control bits.

Bit 15: **TCS (write or read).**

Function: Take control synchronously. This bit is used to take synchronous control of the bus in order to prevent the possible loss of data. Bit 15 sets up the interface hardware so that if a data transfer is taking place across the bus while ATN is unasserted, the NRFD line is asserted at the same time as DAV to inhibit the next handshake cycle; the ATN OUT bit (3) is then set automatically when DAV clears. As soon as the ATN OUT bit becomes set, it resets (clears) the TCS bit. It is possible for the ATN OUT bit to not be set within the expected time interval if the data transfer has slowed or stopped (talker has not yet received the DAC message).

Set: By writing a 1 into two internal flip-flops, with the second flip-flop clocked by the DAV signal.

Cleared: By writing a 0 into this bit, when the ATN OUT bit becomes set, by a RESET instruction, or when the IFC bus line is asserted.

Bits 14 and 13: LIST W ATN and LIST W/O ATN, respectively (write or read).

Function: Either, or both, of these bits enable the interface acceptor handshake logic for the non-DMA mode. See description of the LDB for information relative to enabling the acceptor handshake logic while this interface is outputting data (talking) on the bus during a DMA transfer. For the non-DMA mode, the logic is enabled under the following conditions:

1) When LIST W ATN is set and another controller on the bus asserts the ATN line.
2) If LIST W/O ATN is set and the ATN line is released (unasserted) by another controller on the bus.

The LIST W ATN bit (14) should be set to a 1 whenever another device on the bus is acting as controller-in-charge. The LIST W/O ATN bit (13) is automatically set if the LIST W ATN bit is set and another controller asserts the ATN line. This immediately sets the acceptor handshake function as soon as the ATN line is unasserted. If the program does not require the interface to accept data (it is not a listener) when ATN becomes unasserted, bit 13 should be cleared.

Set: Each, or both, of these bits are set by writing a 1 to internal flip-flops. LIST W/O ATN is automatically set by this interface if LIST W ATN is set and another controller asserts the ATN line.

Cleared: By writing a 0 into the desired bit. Bit 14 is cleared by a RESET instruction. Bit 13 is cleared either by a RESET instruction or the assertion of the IFC line.

Bits 12 and 11: XBA17 and XBA16, respectively (write or read).

Function: Extended address bits. Provided for DMA transfers in Memory Management systems.
XBA17 and XBA16 assert the extended address lines on the controller bus. For Memory Management systems set XBA17 and XBA16 according to the following guide:

00 for transfers in the area of 00-32 K.
01 for transfers in the area of 32-64 K.
10 for transfers in the area of 64-96 K.
11 for transfers in the area of 96-128 K.

Set: Either, or both, bits are set by writing a 1 to internal flip-flops.

Cleared: By writing a 0 to the desired bit. Both bits are cleared by a RESET instruction.

Bits 7 - 0: NRFD OUT, NDAC OUT, DAV OUT, SRQ OUT, ATN OUT, EOI OUT, REN OUT, IFC OUT, respectively (write or read).

Function: The following bits are latched by writing a 1 into the desired bit: bits 7, 6, 5, 4, 2, and 1. Bit 3 is set by writing a 1 to a flip-flop. Bit 0 (IFC OUT) is a triggered "one-shot" multivibrator; the IFC bus line is asserted for approximately 150 microseconds when this bit is loaded and then clears itself. The output of the latches drive the IEEE 488 bus in a corresponding manner; if the output of a latch is a 1, the corresponding bus line is asserted. Note that the ATN OUT bit (3) is also set automatically as described for the TCS bit (15). All of these bits, except REN OUT and IFC OUT, are cleared by either a RESET instruction or when IFC is asserted on the bus. REN OUT and IFC OUT can be cleared by RESET instruction. The status of the IEEE 488 bus lines associated with

these bits can also be read from the Bus Status Register (BSR). When the IFC bus line is asserted by this interface or another controller, the eight programmable registers are affected in the following manner (see Table 3-3):

TDB: Clears the EOI function associated with bit 15 and removes output data to the IEEE 488 bus.

LDB: Configured to receive new data.

ICR: Sets the DONE bit. Sets IFC OCC and generates an interrupt if IFC INTR ENBL is set. Clears the RDY bit.

BSR: A positive transition occurs at bit 0.

BCR: Clears TCS, LIST W/O ATN, NRFD OUT, NDAC OUT, DAV OUT, SRQ OUT, ATN OUT, and EOI OUT.

ISR: Sets DMA ABORT and clears the GO bit.

CTR: Does not affect this register.

BAR: Does not affect this register.

Note that the ATN OUT and SRQ OUT bits, when set, will generate interrupts through the ICR if their enabling bits have been set. The REN OUT bit generates a REN FALSE interrupt if it is cleared after being set and the enabling bit has been set. The status of the ATN, REN, and EOI bus lines can also be read from the LDB. In most cases, the LDB should not be read during a DMA transfer since it could cycle the acceptor handshake logic.

INTERFACE STATUS REGISTER (ISR) **1AAA12**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ERR	TIME ERR	WRITE ERR	DMA ERR	DMA ABORT				DMA DONE	DMA INTR ENBL	SWAP BYTE ORDER	HOG MODE	PACK MODE	DMA IN/ OUT	NO INCR/ DECR	GO

Interface Status Register (ISR). The primary function of the Interface Status Register allows the user to control and monitor the operation of this interface in the DMA modes. Other DMA primary control bits are found in the Bus Status Register. The Byte Counter Register and the Bus Address Register are also used for DMA operations.

Bit 15: **ERR (read only).**

Function: For DMA or non DMA modes. This bit is the inclusive-OR of the following four errors: 1) TIME ERR, 2) WRITE ERR, 3) DMA ERR, and 4) DMA ABORT. If any of the four error flags or the DMA DONE bit (7) becomes set, the GO bit is automatically cleared. If the DMA INTR ENBL bit is set and an error occurs during a DMA transfer, or the transfer is completed (DMA DONE set), the interface generates the highest priority interrupt and returns a vector address determined by the vector address straps. Bits 15 through 7 of this register are read by the program to determine the type of error or if the DMA transfer is complete.

Set: Automatically by this interface if bit 14, bit 13, bit 12, or bit 11 of this register become set.

Cleared: By writing a 1 into the DMA INIT bit (10) of the BSR, or by a RESET instruction.

Bit 14: **TIME ERR (read only).**

Function: For DMA or non DMA modes. When set, indicates that one of two possible timing errors has occurred: 1) Another controller asserting ATN while this interface is handshaking data. 2) An attempt to load the TDB before a previous byte was accepted by all listeners.

Set: By this interface if another controller asserts ATN during the time that DAV is asserted by this interface or, if the program tries to load the TDB before DAV assumes its unasserted state.

Cleared: By writing a 1 to the DMA INIT bit (10) in the BSR, or by a RESET instruction.

Bit 13: **WRITE ERR (read only).**

Function: For DMA or non DMA modes. When set, indicates that an attempt was made to load the TDB when no device on the IEEE 488 bus was listening.

Set: By this interface if an attempt is made to load the TDB while the NRFD and NDAC bus lines are in their unasserted states.

Cleared: By writing a 1 into the DMA INIT bit (10) of the BSR, or by a RESET instruction.

Bit 12: **DMA ERR (read only).**

Function: When set, indicates one of two possible error conditions occurred during a DMA transfer: 1) the Bus Address overflowed, or 2) no response from a designated slave device; for example, a non-existent memory location.

Set: By this interface if the BAR overflows or if the slave device does not reply when addressed.

Cleared: By writing a 1 into the DMA INIT bit (10) in the BSR or by a RESET instruction.

Bit 11: DMA ABORT (read only).

Function: When set, indicates one of four possible reasons for the DMA transfer to halt before completion: 1) IFC asserted, 2) ATN asserted by another controller, 3) A TIME ERR, WRITE ERR, or DMA ERR occurred, 4) The REN FALSE condition occurred with its enabling bit set. An error interrupt vector is generated if the DMA INTR ENBL bit (7) has been set and the DMA aborts; see ERR bit (15).

Set: By this interface during a DMA transfer under any of the following conditions; IFC asserted, ATN asserted by another controller, the ERR bit becomes set, or if REN goes from its asserted to unasserted state with the REN INTR ENBL bit (10) in the ICR set. The DMA DONE bit (7), if set, inhibits the setting of DMA ABORT.

Cleared: By writing a 1 into the DMA INIT bit (10) in the BSR, or by a RESET instruction.

Bit 7: DMA DONE (read only).

Function: When set, indicates the normal end of a DMA transfer when transmitting data in either direction across this interface.

Set: Automatically by this interface when the END bit (13) in the ICR becomes set and the NO EOI DMA bit has been cleared; if the NO EOI DMA bit has been set, the DMA DONE bit is set by Byte Counter Register overflow (all 1's to all 0's).

Cleared: By writing a 1 into the DMA INIT bit (10) in the BSR, or by a RESET instruction.

Bit 6: **DMA INTR ENBL (write or read).**

Function: When set, enables the interrupt condition for DMA DONE or for one of the error flags associated with bits 15 through 11 of this register.

Set: By writing a 1 to an internal flip-flop.

Cleared: By writing a 0 into this bit, or by a RESET instruction.

Bit 5: **SWAP BYTE ORDER (write or read).**

Function: This bit is used to control the manner in which memory is addressed when operating in the DMA "packed" mode (bit 3 of this register set). If this bit (5) is clear, memory is addressed in the standard fashion (16-bit memory word assumed):
1) Low order bytes first when incrementing the Bus Address Register.
2) High order bytes first when decrementing the Bus Address Register.

If the SWAP BYTE ORDER bit is set, memory is addressed in the following manner (16-bit memory word assumed):
1) High order bytes first when incrementing the Bus Address Register.
2) Low order bytes first when decrementing the Bus Address Register.

Refer to the Servicing Section of this manual for information on how to increment or decrement the BAR.

Set: By writing a 1 to an internal flip-flop.

Cleared: By writing a 0 into this bit, or by a RESET instruction.

Bit 4: **HOG MODE (write or read).**

Function: This bit defines how the controller bus is used during a DMA transfer; it can be set or cleared while the DMA is running. If this bit is clear, the interface gets control of the bus, transfers one byte, and then releases the bus before the next byte transfer. If this bit is set, the interface gets control of the bus and does not release it until the DMA transfer is completed. This interface will release the controller bus if a device on the IEEE 488 bus does not respond within 20 microseconds; this is not an "error" (an error flag is not set). The device was slow to respond and as soon as it does respond, the interface will proceed with the remainder of the DMA transfer.

Set: By writing a 1 to an internal flip-flop.

Cleared: By writing a 0 into this bit, or by a RESET instruction.

Bit 3: **PACK MODE (write or read).**

Function: This bit specifies how the data is stored in (or retrieved from) memory. If set, the 16-bit memory word is byte addressed. The order of byte transfer is dependent on the setting of the SWAP BYTE ORDER bit and whether the BAR is incremented or decremented. See discussion under SWAP BYTE ORDER bit and the BAR. In the PACK MODE, the BAR is incremented or decremented by one (1) after each byte transfer. The transfer may start at an even or odd address.
When this bit is clear, the interface is operating in an "unpacked" mode and the data is assumed to appear in the low order byte defined by the current value in the BAR. When transferring data from the IEEE 488 bus to memory in the "unpacked" mode, the high byte of a word will be ignored during the time that the low byte is written; the BAR is incremented or

decremented by two (2) after each byte transfer if the NO INCR/DECR bit is cleared.

Set: By writing a 1 to an internal flip-flop.

Cleared: By writing a 0 into this bit, or by a RESET instruction.

Bit 2: DMA IN/OUT (write or read).

Function: Specifies the direction of the DMA transfer. When set, the data is transferred from the IEEE 488 bus to the controller bus (DMA IN). When clear, the data is transferred from the controller bus to the IEEE 488 bus (DMA OUT).

Set: By writing a 1 to an internal flip-flop.

Cleared: By writing a 0 into this bit, or by a RESET instruction.

Bit 1: NO INCR/DECR (write or read).

Function: When set, indicates that the same address should be used for the entire DMA transfer; the BAR will not be incremented or decremented during the transfer.

Set: By writing a 1 to an internal flip-flop.

Cleared: By writing a 0 into this bit, or by a RESET instruction.

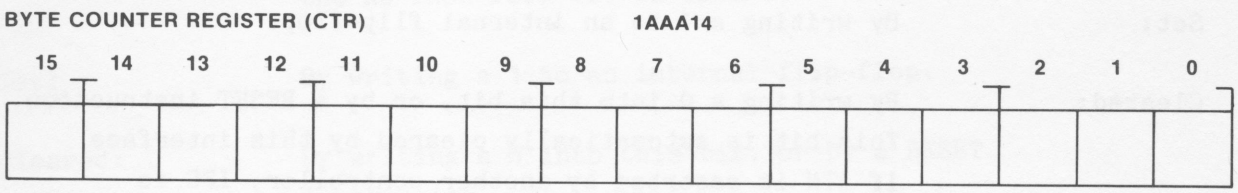
Bit 0: GO (write or read).

Function: When set, initiates a DMA controlled data transfer. This bit can be set simultaneously with bits 1 through 6 of this register, or set last in a sequence. Before setting the GO bit, the Bus Status Register (BSR)

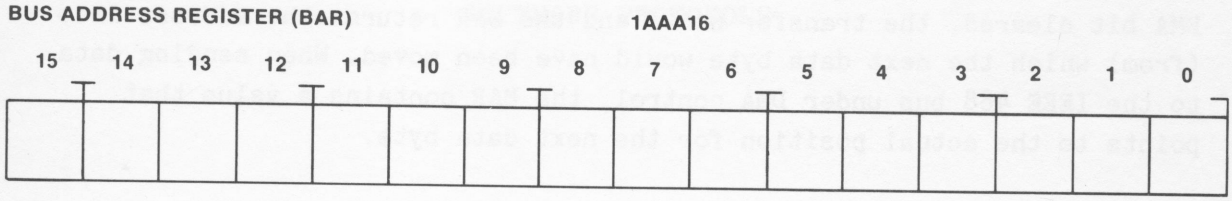
should be programmed properly for a DMA transfer.

Set: By writing a 1 to an internal flip-flop.

Cleared: By writing a 0 into this bit, or by a RESET instruction. This bit is automatically cleared by this interface if ATN is asserted by another controller, IFC is asserted, REN FALSE condition occurs with its enabling bit set, or if any of bits 15 through 7 of this register becomes set.



Byte Counter Register (CTR). The Byte Counter Register is a write or read register and used by the programmer for DMA operations. Initially, the CTR is loaded with the 2's complement of the number of bytes to be transferred. For each byte transferred, this register increments toward zero. When the Byte Counter Register overflows (all 1's to all 0's), the DMA DONE bit in the Interface Status Register is set and the DMA transfer is complete. If an EOI message is received and the NO EOI DMA bit is set, the transfer halts with the DMA DONE bit set. The Byte Counter Register contains the 2's complement of the initial byte count minus the number of bytes transferred. Each bit is set/cleared by writing a 1/0 into the desired bit. See NOTE under the description for the Bus Address Register.



Bus Address Register (BAR). The Bus Address Register is a write or read register and is used for DMA operations. This register (along with the extended address bits, XBA17 and XBA16 in the Bus Control Register) is used to specify the controller bus address of the location to (or from) which data is to be transfer during DMA cycles. The BAR is incremented or decremented according to how a strap (jumper) on this interface is set by the user. If the NO INCR/DECR bit in the Interface Status Register is set after the BAR is loaded, the same address can be used for the entire DMA transfer. If the PACK MODE bit in the ISR is set, the BAR is incremented (decremented) once for each byte transfer; if it is cleared, the BAR is incremented (decremented) twice for each byte transferred. If the BAR overflows during a DMA transfer, the DMA ERR bit in the ISR is automatically set. Since BAR overflow does not increment (decrement) the extended address bits, the maximum block of data that can be transferred in one operation is 64 K bytes.

The SWAP BYTE ORDER bit in the ISR does not affect the actual contents of the BAR; if the SWAP BYTE ORDER bit is set, the least significant bit in the BAR will appear complemented when read or when it is used as an address on the controller bus. Each bit of the BAR is set or cleared by writing a 1 (set) or 0 (clear) into the desired bit.

NOTE

If this interface is performing a DMA transfer from the controller bus to the IEEE 488 bus (a talk mode) and another controller performs a "take control synchronously" operation (TCS), the CTR and the BAR should be backed up one cycle before continuing.

If an EOI message is received during a DMA transfer with the NO EOI DMA bit cleared, the transfer halts and the BAR returns the address into (from) which the next data byte would have been moved. When sending data to the IEEE 488 bus under DMA control, the BAR contains a value that points to the actual position for the next data byte.

Local Reset and Interface Clear

Table 3-3 illustrates the bits in each register that is cleared (or set) by issuing a LOCAL RESET instruction or when IFC is asserted on the IEEE 488 bus.

TABLE 3-3
Local Reset/Interface Clear Functions.
Bits cleared by R (LOCAL RESET) and I (Interface Clear)

Bit	15 (EOI)	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
TDB	IR	-	-	-	-	-	-	-	IR	IR	IR	IR	IR	IR	IR	IR
LDB	-	-	-	R	-	-	-	-	-	-	-	-	-	-	-	-
ICR	(SET)	-	R	R	R	-	-	-	IR	R	R	R	R	R	R	R
BSR	R	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
BCR	IR	R	IR	R	R	-	-	-	IR	IR	IR	IR	IR	IR	R	IR
ISR	R	R	R	R	R	-	-	-	R	R	R	R	R	R	R	IR

NOTE: The DONE bit (15) in the Interrupt Control Register (ICR) is set by either LOCAL RESET (R) or Interface Clear (I). The DMA ABORT bit (11) in the Interface Status Register (ISR) will be set by Interface Clear if a DMA transfer has not been completed.

SECTION 4

SOFTWARE PROTOCOLS

Introduction

Programming protocol for the IEEE 488 interface is discussed in this section of the manual. The program may act as a system controller or control may be passed back and forth between this interface and another controller on the IEEE 488 bus.

The basic concept of the IEEE 488 system is described in Section 2 of this manual. For more specific details and state diagrams relative to the IEEE 488 interface functions, refer to IEEE Standard 488-1975.

This interface has eight addressable registers and eight interrupt vectors. The registers and interrupts are specifically discussed in Section 3 of this manual.

The IEEE 488 system usually requires that one device be designated as the system controller. It is possible, however, to have a system configured where more than one device has the controller function; when that is the case, only one device may be the controller-in-charge at any given time. To get control of the bus, any controller may request transfer of control from the current controller-in-charge. A system controller can take control from the current controller-in-charge by requesting that control be returned or by simply asserting the Interface Clear (IFC) message. Interface Clear can only be performed by a system controller and it causes the Talker and Listener functions, as well as the Controller functions in other devices, to go to their power on state. The other interface (IEEE 488) functions are not directly affected by the IFC message.

Programming Considerations

Addressing. This interface can be strapped to respond to addresses within the range of 160000_8 through 177776_8.

The extended address bits (XBA16 and XBA17) in the Bus Control Register are used when the controller has provision for Memory Management. In order to perform a DMA operation with this interface to, or from, another device in the peripheral space, the user must set XBA16 and XBA17 for data transfers to the desired memory area (see Section 3 - Bus Control Register).

The addressable registers on this interface do not support write byte operations. If the user performs a write byte operation on one of the interface registers, the byte will be loaded into both the high and low order bits of the register. Read byte operations are handled in the standard fashion.

The Talker Data Buffer is typically thought of as a write only register. However, register address 1AAA00 can be read and has the added feature that it returns the vector address for the last interrupt granted. This feature can be used by system software to determine how the interface interrupt vectors have been set (strapped).

Local Reset. The LOCAL RESET bit in the Bus Status Register provides a means of initializing the controller side of this interface. Writing a 1 into this bit performs the same function on this interface as occurs when a controller RESET (INIT) instruction is executed. The LOCAL RESET bit can be used to reinitialize this interface without affecting the other interfaces (peripherals) in the system. Refer to Table 3-3 (Section 3) for information concerning the bits in each register that are cleared by the LOCAL RESET and Interface Clear (IFC) functions.

The EOI function. The EOI line of the IEEE 488 bus is used to indicate the end of data transmission (or Identify for a Parallel Poll, covered later). The EOI line can be asserted by setting the EOI bit in the Bus Control Register; however, since it is normally sent only with the last data byte in a message, the EOI message may be sent by setting

bit 15 in the Talker Data Buffer when sending data in a non DMA mode. The programmer must take care when using the internal registers of the controller (R0-R5) that an automatic sign extension does not inadvertently set bit 15 and assert the EOI line. For DMA transfers, the EOI message is automatically sent along with the last byte of data unless the programmer sets the NO EOI DMA bit in the Bus Status Register before the DMA transfer is initiated. The EOI bus line can be monitored directly by reading the Bus Status Register; it is also available as the most significant bit (15) in the Listener Data Buffer.

The END condition (EOI and DAV asserted, ATN unasserted) is detected by this interface and indicated in the Interrupt Control Register as the END OCC bit. The END condition can also be monitored by enabling its interrupt. Once the interrupt is detected, either the END OCC bit or its interrupt enabling bit (2) must be reset and set again for another interrupt to occur. The END OCC bit is cleared by writing a 1 to the END INTR CLR bit in the Bus Status Register.

The Listen Enable Functions. The source and acceptor handshake functions for this interface are totally independent. If the LIST W/O ATN bit in the Bus Control Register is set and the user writes data to the Talker Data Buffer, the interface will listen to itself because the acceptor handshake logic has been enabled. This feature can be used by the programmer for system checkout, but the programer must take care not to inadvertently "hang" the IEEE 488 bus with the interface listening to itself.

The LIST W ATN bit in the Bus Control Register is implemented such that if it is set, the interface will automatically start listening when ATN is asserted by another controller, yet will not handshake data sent if the ATN OUT bit in the Bus Control Register is set. This feature allows this interface to respond properly to the system controller asserting the IFC and ATN lines, and allows this interface to send data as a controller-in-charge (ATN asserted) without listening to itself.

Errors Detected and Reported. This interface detects and reports four errors to the program, they are:

1. TIMING ERROR caused by:
    a) Writing to the Talker Data Buffer before the previous byte was accepted by all assigned listeners.

b) Another controller asserting ATN while this interface is sending data.

2. WRITE ERROR caused by:
   a) Writing to the Talker Data Buffer when no device on the IEEE 488 bus is listening.
3. DMA ERROR caused by:
   a) Overflow of the Bus Address Register.
   b) No response from an addressed "slave" device in a DMA transfer.
4. DMA ABORT caused by:
   a) Interface Clear (IFC) being issued during a DMA transfer.
   b) ATN being asserted by another controller during a DMA transfer.
   c) Remote Enable (REN) going from the asserted state to the unasserted state with the REN INTR ENBL bit set during a DMA transfer (a REN FALSE condition).
   d) A TIMING ERROR, WRITE ERROR, or DMA ERROR occurrence during a DMA transfer.

Implementation of the IEEE 488 Interface Functions

The IEEE 488 interface functions provide a wide variety of capabilities and options, making the implementation of all of them in hardware unrealistic. For this reason only the most generally used functions (Source and Acceptor Handshake functions) are implemented completely in hardware. The other functions must be implemented in software, though in most cases the interface has features which make the task easier.

Program examples are provided, in some cases, to illustrate and guide the user in implementing the interface functions. However, they may not be complete and are not guaranteed to execute correctly as written.

Controller Function

If this interface is to act as controller of the IEEE 488 bus, the programmer must implement the Controller Functions in software. The Controller Function provides a device with the capability to send device addresses, universal commands, addressed commands, secondary addresses,

and secondary commands to other devices in the system. A controller function also provides the capability to conduct serial and parallel polls to determine which device requires service.

A controller function can exercise its capabilities, except for IFC and REN, only when it is asserting ATN on the bus. If more than one device on the bus has a Controller Function, then all but one of them must be in the Controller Idle State (CIDS) at any given time. The device containing the Controller Function which is not in CIDS is called the controller-in-charge.

The Controller Function in one of the devices connected to the bus, but not more than one, can exist in the System Controller Active State (CACS). It must remain in this state throughout the operation of the system. As the System Controller it possesses the capability to send the IFC and REN messages at any time, whether or not it is the controller-in-charge.

Usage of IFC and REN. If this interface is the System Controller, the programmer may assert IFC or REN by writing a 1 to the IFC OUT or REN OUT bits in the Bus Control Register. Interface Clear is implemented so that after approximately 150 microseconds it is automatically unasserted. The REN line remains asserted until the programmer clears the REN OUT bit, or issues a LOCAL RESET instruction, or a controller RESET (INIT) instruction is executed.

The programmer should be aware of the fact that if an IFC message is issued, the state of some bits in the interface registers are changed. A complete description of the bits affected by the IFC message is found in Table 3-3 (Section 3).

Usage of ATN and TCS. All data transmitted over the IEEE 488 system with the ATN line asserted is interpreted as system control information. This interface allows the user to assert the ATN line directly by setting the ATN OUT bit in the Bus Control Register. This, however, is an asynchronous operation with respect to the IEEE 488 bus and may cause loss of data if any data transfer operation is in progress. When this is the case, the programmer should set the TCS (take control synchronously) bit in the Bus Control Register. This procedure sets up the hardware on this interface so that if a data transfer (handshake) cycle is taking place, the NRFD bus line is asserted to stop the next cycle and then sets

the ATN OUT bit to assert the ATN line. Setting the ATN OUT bit clears the TCS bit; either bit may be monitored to determine when control is acquired. Another alternative is to use the ATN interrupt vector to determine when this interface has taken control.

Caution must be used when programming the TCS bit since it does not set the ATN OUT bit if the data transfer cycle has stopped for some reason. There are three conditions under which this may occur: 1) the talker has finished talking, 2) the talker is slow to send the next data byte, or 3) if a device on the bus is not functioning properly.

If it is known that all talkers in the system send the EOI message with the last byte of data they have to transfer the programmer can detect when the current talker is finished; just make sure that the END OCC bit in the Interrupt Control Register is clear before the transfer is started and it will become set if the EOI message is transmitted.

If the talker is not finished, there is a serious problem since it is possible for that talker to send a data byte at exactly the same time as the controller asserts ATN. This causes other devices to interpret the device dependent message as an interface command, placing them in an unknown state. To prevent this situation, the programmer should wait a "reasonable" amount of time for a byte transfer. If a handshake cycle does not occur during that time, the only recourse is to issue IFC (issuing IFC may cause a byte of device dependent data to be lost, depending on how the various devices react to it) or assert ATN asynchronously and set up the entire system again. If the latter course is followed, the programmer should assume that the state of each device in the system is unknown. To restart, a Device Clear (DCL) command should be issued in order to assure that all device functions are in a known state before proceeding with system configuration.

For example, suppose that the system consists of devices which are known to transmit (receive) data at a rate no slower than one handshake cycle in three seconds once the transfer is started. The programmer had set up a data transfer between two devices on the IEEE 488 bus and a third device asserted the SRQ line just after the transfer was initiated. To get control of the bus to perform a serial poll, the following procedure might be used:

```
        Set the ATN interrupt vector to address of L1:
        Enable the ATN interrupt.
        Set TCS.
    (Vector directly to L1 if interrupt occured)
        If the END OCC bit is set,
             Then assert ATN asyncronously
             and branch to L2:
        ELSE:
             Wait 3 seconds.
             Assert Interface Clear.
             Wait for IFC to become unasserted.
             Set ATN asynchronously.
             Set a flag that previous data transfer was
                  interrupted and should be retransmitted.
             Branch to L2:
    L1: Pop return address from stack
    L2: Perform Serial Poll
```

Performing a Serial Poll. To conduct a serial poll, the program should first issue an UNL (unlisten) command to prevent devices from listening to the status bytes sent to the controller during the poll. The program should then send a Serial Poll Enable (SPE) command and individually address each device as a talker. The program should then read one byte from each device (LIST W/O ATN must be set and the program must wait for the RDY bit in the Interrupt Control Register to become set). Bit 6 (= DIO 7 on the IEEE 488 bus) of the byte read determines whether that device requested service by asserting the SRQ bus line; the remaining bits of the byte read contain status information. When the status byte of all devices have been checked, the program should issue an UNT (untalk) command followed by the Serial Poll Disable (SPD) command. If a device dependent message was interrupted for a serial poll, the previous talkers and listeners may then be restored (readdressed) and the program should continue.

Performing a Parallel Poll. This interface allows the program software to conduct a parallel poll (check the status of several devices on the bus simultaneously). Before a parallel poll can be performed, the devices which have been selected to respond must first be configured. To do this, address each device to be included in the parallel poll as a listener, send a Parallel Poll Configure (PPC) command followed by a Parallel Poll Enable (PPE) command which includes a designation of which

data line the devices are to respond on and how to respond. For the PPE command, the exact binary format is 110SNNN, where "NNN" is an octal number equivalent to a decimal digit in the range 0 to 7 and designates the data line for the device to send the parallel poll response on. "S" is the sense required for a true response.

The parallel poll is activated when the program asserts both the ATN and EOI bus lines (sets bits 3 and 2 in the Bus Control Register). At this time, all devices which have been enabled for a parallel poll place their status code on the data lines and the program reads it directly through the Listener Data Buffer. There is no need to set the LISTEN bits or wait for the RDY bit to become set. When the parallel poll is completed, the program should clear the EOI OUT bit (and ATN OUT, if desired). The PPD and PPU commands are used when the system needs to be reconfigured.

Passing Control. As controller-in-charge, the program may relinquish control to other devices in the system capable of acting as a controller. To do this, address the desired device as a talker, followed by the Take Control (TCT) command. The other device becomes controller-in-charge when ATN is cleared.

Interface Commands. A list of currently defined control commands for the IEEE 488 bus system is found in Table 4-1.

Table 4-1.

List of IEEE 488 interface control commands.

Octal Code	Command	Meaning
024	DCL	Device Clear
010	GET	Group Execute Trigger
001	GTL	Go To Local
021	LLO	Local Lock Out
005	PPC	Parallel Poll Configure
160	PPD	Parallel Poll Disable
14N or 15N (N=0-7)	PPE	Parallel Poll Enable
025	PPU	Parallel Poll Unconfigure
004	SDC	Selected Device Clear

031	SPD	Serial Poll Disable
030	SPE	Serial Poll Enable
011	TCT	Take Control
077	UNL	UNListen
137	UNT	UNTalk

Source Handshake Function

The Source Handshake function provides a device with the capability to guarantee the proper transfer of data. An interlocked handshake sequence between the source and acceptor handshake functions guarantees asynchronous transfer of each data byte (see Section 2). The Source Handshake function utilizes the DAV, RFD, and DAC messages to transfer each byte across the IEEE 488 bus. The program needs to send data to other devices both for interface control information (ATN asserted) and for device dependent messages (ATN unasserted).

The Source Handshake function for this interface is implemented in hardware for the non DMA and the DMA modes. For the DMA talk mode, the programmer needs to set up the DMA transfer (address in memory where the data is stored and the number of bytes to transfer) and set the appropriate DMA control bits in the Interface Status Register, making sure that the DMA IN/OUT bit is clear. For more complete details about DMA data transfers, see the subsection titled "Operation in DMA Mode".

The non DMA Source Handshake functions is implemented through the Talker Data Buffer and the Interrupt Control Register. The byte to be transmitted is written into bits 0 through 7 of the Talker Data Buffer, with bit 15 set at the appropriate time if the EOI line is to be asserted with a data byte. To assert the EOI line independent of data transmission, use the EOI OUT bit in the Bus Control Register. The interface hardware performs the Source Handshake and sets the DONE bit in the Interrupt Control Register when the data has been accepted by all of the assigned listeners. When the DONE bit is set the next data byte can be loaded into the Talker Data Buffer. The DONE bit is cleared when the Talker Data Buffer is loaded and may be monitored by reading the Interrupt Control Register or by setting the DONE INTR ENBL bit. Any time both the DONE bit and its interrupt enable bit are set, a processor interrupt is generated to the DONE (talker) vector. Only one interrupt is generated and another will not occur until either the DONE bit or its

interrupt enabling bit are cleared and set again.

There are two error conditions detected by this interface that are associated with the Source Handshake function. a TIMING ERROR is detected if the program writes to the Talker Data Buffer when the DONE bit is not set or if another device on the IEEE 488 bus is asserting ATN when data is loaded into the Talker Data Buffer. A TIMING ERROR is not detected if the controller-in-charge asserts ATN after the Talker Data Buffer is loaded. It is the responsibility of the controller-in-charge to take control synchronously (tcs) and assure that the devices on the bus are ready to continue their data transmission after ATN is released (unasserted).

A WRITE ERROR is detected when an attempt is made to talk to a device on the IEEE 488 bus and no device is listening (neither the NRFD nor NDAC bus lines are asserted when the Talker Data Buffer is loaded).

The TIMING ERROR and the WRITE ERROR are indicated by the corresponding bits in the Interface Status Register being set. These error conditions may be monitored by reading the Interface Status Register and testing the ERR bit (set if any of the errors are detected) or by setting the interrupt enabling bit (6) in the Interface Status Register. The ERROR condition bits are cleared by writing a 1 to either the DMA INIT or LOCAL RESET bits in the Bus Status Register or when a controller RESET instruction is executed.

When the WRITE ERR bit is set in the Interface Status Register, the Talker handshake is hung waiting for a completed handshake cycle. The DMA INIT only clears the occurred bit, not the handshake. There are four ways in which the Talker handshake can be cleared: IFC asserted on the bus, receive the ATN message, set the LOCAL RESET bit, or toggle the NDAC OUT bit in the Bus Control Register. Toggling the NDAC OUT bit is probably the preferred way to clear the handshake because it does not have other side effects.

NOTE

Because the Talker Data Buffer is part of the data path from the controller internal put to the IEEE 488 bus, the program should not read from or write to the Talker Data Buffer while

a DMA transfer is taking place. The DMA transfer can be interrupted, however, so that data can be transmitted with the Talker Data Buffer in the non DMA mode as long the GO bit in the Interface Status Register is not set. The DMA transfer can be continued by setting the GO bit again. It is possible (for checkout purposes) to talk through the Talker Data Buffer and listen with DMA control logic. The programmer must take care never to have the RDY INTR ENBL nor the DONE INTR ENBL bits set during a DMA transfer. The reason for this is, that the handshake cycles will continue and by the time that the interrupt handler gets to read the byte it will be gone and a new byte will be in the Talker Data Buffer.

Acceptor Handshake Function

The Acceptor Handshake function provides a device with the capability to guarantee the proper reception of data. An interlocked handshake sequence between the source and acceptor handshake functions guarantees the asynchronous transfer of each message byte (see Section 2). The Acceptor Handshake function may delay either the initiation of, or the termination of, a data byte transfer until it is prepared to continue with the transfer process. The acceptor Handshake function utilizes the DAV, RFD, and DAC messages to perform each byte transfer.

The program needs to receive data sent by other devices on the IEEE 488 bus. The Listener Data Buffer is one of the registers on this interface used to implement this function; it is a read only register and returns the current value of the DIO1 - DIO8 bus lines (in bits 0 - 7), the ATN line, the EOI line, and the REN line when read.

In order to determine when valid data is present on the DIO lines the programmer must implement the Acceptor Handshake. Implementing the handshake cycle in software is a tedious and time consuming task. This interface implements the Acceptor Handshake in hardware.

To enable the Acceptor Handshake function for the DMA listen mode, make sure that the DMA IN/OUT bit in the Interface Status Register is set. The Acceptor Handshake function is enabled as soon as the DMA IN/OUT bit and the GO bit is set.

To enable the Acceptor Handshake function for the non DMA mode, set either the LIST W/O ATN or LIST W ATN bits in the Bus Control Register. The interface performs the Acceptor Handshake and sets the RDY bit in the Interrupt Control Register when valid data can be read from the Listener Data Buffer. The RDY bit may be monitored (similar to the DONE bit) by reading the Interrupt Control Register or by setting the RDY INTR ENBL bit, which generates a processor interrupt to the RDY (listen) vector when both are set. The RDY bit is cleared when the Listener Data Buffer is read.

When operating in the non DMA mode, the programmer may choose to perform a Not Ready For Data (NRFD) or Not Data Accepted (NDAC) hold-off of the current talker. To do this, the program asserts the NRFD or NDAC lines through the corresponding control-OUT bit in the Bus Control Register before reading the Listener Data Buffer. By setting the NRFD OUT bit, the programmer can halt the Acceptor Handshake function for this interface in the Acceptor Not Ready State (ANRS). In this case, when the Listener Data Buffer is read, the Acceptor Handshake transfers from the Accept Data State (ACDS) to the Acceptor Wait for New Cycle State (AWNS) and then to the ANRS state, where it remains until the NRFD OUT bit is cleared.

Setting the NDAC OUT bit when the RDY bit is set causes the Acceptor Handshake for this interface to halt in the Accept Data State; forcing the current talker to continue asserting the data and allowing the programmer to read the Listener Data Buffer more than once. This procedure allows several tests to be performed on the data without using one of the controller's internal registers for storage. The handshake remains in this state until the programmer clears the NDAC OUT bit.

NOTE

Because the Listener Data Buffer is part of the data path between the controller's internal bus and the IEEE 488 bus, the program should not read the Listener Data Buffer while a DMA transfer is

taking place. The Listener Data Buffer may safely be used, however, any time the GO bit is not set. It is possible (for checkout purposes) to listen through the Listener Data Buffer and talk with the DMA control logic. The programmer should take care never to have the RDY INTR ENBL nor the DONE INTR ENBL bit set during a DMA transfer (see note under Source Handshake Function).

Talker Function

The Talker function provides a device with the capability to send device dependent data over the IEEE 488 bus. This capability exists only when the Talker function in a device is addressed to talk (MTA). The Talker function must be implemented in software. When the Talker function is in the Talker Active State (TACS), the programmer may "talk" to other devices on the bus by writing bytes to the Talker Data Buffer or by enabling the proper DMA control bits with the DMA IN/OUT bit clear.

There are two alternative versions of the Talker function which the programmer may choose to implement. The normal Talker function uses a one byte address (MTA only). The Talker function with address extension uses a two byte address (MTA followed by a MSA). In all other respects, the capabilities of both versions are the same. Only one of the two alternative Talker functions need be implemented in a specific device. Talk and Listen address codes are listed in Table 4-2.

Only one device in the system can be in the Talker Active State at any given time. It commences talking when the ATN line is cleared and continues its talker status until: an Interface Clear (IFC) occurs, an UNTalk command is received from the controller-in-charge, another device is addressed as talker (OTA); or optionally, the same device is addressed as a listener (MLA).

When a controller with this interface installed is acting as the controller-in-charge and the program requires that it make itself a talker, it must first make sure that no other device is in the Talker Addressed State (TADS). This can be done by sending its own talk address on the bus. However, in systems with only one device capable of being a controller, sending the UNTalk command would probably be sufficient.

Listener Function

The Listener function provides a device with the capability to receive device dependent data transmitted over the IEEE 488 bus. This capability exists only when the Listener function is addressed to listen. There are two alternative versions of the Listener function; one with and one without address extension. The normal Listener function uses a one byte address (MLA only). The Listener function with address extension (Extended Listener) uses a two byte address (MLA followed by a MSA). In all other respects the capability of both versions are the same. Only one of the two alternative Listener functions need be implemented in a specific device. Listen and Talk address codes are listed in Table 4-2.

TABLE 4-2
List of listen, talk and secondary addresses.

Address Value	Listen (MLA)	Talk (MTA)	Secondary Address/Command (MSA)
0	040	100	140
1	041	101	141
2	042	102	142
3	043	103	143
4	044	104	144
5	045	105	145
6	046	106	146
7	047	107	147
8	050	110	150
9	051	111	151
10	052	112	152
11	053	113	153
12	054	114	154
13	055	115	155
14	056	116	156
15	057	117	157
16	060	120	160
17	061	121	161
18	062	122	162
19	063	123	163
20	064	124	164
21	065	125	165
22	066	126	166
23	067	127	167
24	070	130	170
25	071	131	171
26	072	132	172
27	073	133	173
28	074	134	174
29	075	135	175
30	076	136	176

Once a device has been properly addressed as a listener, it starts to listen to all device dependent data as soon as the ATN line is cleared and continues to listen until: an Interface Clear (IFC) occurs, an UNListen command is received; or optionally, the same device is addressed as a talker. More than one device may be programmed to listen to device dependent data (at any give time) from a talker.

The Listener function must be implemented in software. For the program to act as a non-controller, the programmer sets the LIST W ATN bit in the Bus Control Register to receive messages (commands) sent with ATN asserted. Then, to transfer from the Listener Idle State (LIDS) to the Listener Addressed State (LADS), the program sets the LIST W/O ATN bit (or the DMA IN/OUT and GO bits for DMA transfers).

Because the transfer from LIDS to LADS is performed in software, time must be allowed for the program to take the proper action upon receiving a message with ATN asserted. For this reason, if the LIST W ATN bit is set (the program for this interface is not the controller-in-charge) and the ATN line is asserted by the controller-in-charge, the LIST W/O ATN bit is automatically set by this interface. For example; if the controller-in-charge were to address this interface as a listener, it could possibly have released the ATN line and started sending device dependent data before the program was able to set the LIST W/O ATN bit. This interface sets the LIST W/O ATN bit every time it determines that the LIST W ATN bit is set and the controller-in-charge asserts the ATN line. The resulting software protocol is that the programmer must clear the LIST W/O ATN bit after the controller-in-charge releases the ATN line, or read the data bytes sent without ATN asserted (even if the data is not directed to this controller) in order to prevent the IEEE 488 bus from becoming locked in the handshake cycle for the first data byte after the controller-in-charge released ATN.

When the program for this interface is acting as controller-in-charge and there is a need to listen to messages from another device, the programmer sets the LIST W/O ATN bit (or the DMA IN/OUT and GO bits for a DMA transfer). The programmer should take care that the Listen Enable modes are cleared when the program needs to send data, because having either of them set implies that its interface performs the Acceptor Handshake function and the software will listen to the data it is sending.

When setting up another device as a talker and this interface as a listener, the programmer should make sure that the Acceptor Handshake function is enabled (either by setting the Listen Enable modes correctly or through the DMA control logic) before the ATN line is released. If this protocol is not followed some data may be lost because the talker is allowed to start sourcing data as soon as the ATN is released by the controller-in-charge.

Service Request Function

The Service Request function provides a device with the capability to asynchronously request service from the controller-in-charge. The function also synchronizes the values of the Service Request bit of the status byte present on the data lines during a serial poll, so that the SRQ message can be removed from the bus once this bit is received true by the controller-in-charge.

The Service Request function for this interface must be implemented in software. An SRQ OUT bit in the Bus Control Register is provided on this interface so that the SRQ line of the IEEE 488 bus may be asserted by the programmer to indicate to the controller-in-charge that service is desired.

The system controller may conduct a serial poll at any time (whether SRQ is asserted or not). As a device (non-controller) on the IEEE bus, the program for this interface should respond with a status byte loaded into the Talker Data Buffer. The program detects the serial poll by monitoring all messages sent by the system controller with ATN asserted. The program must watch for the Serial Poll Enable (SPE) command; if this command is received, the controller associated with this interface is in the Serial Poll Mode State (SPMS) of the Talker function. After the SPE command is received, the talk address for this interface may eventually be sent. If the talk address (MTA) is received before the SPD command, the transition should be made to the SPAS of the Talker function. Then, as soon as ATN is clear, the program should clear the SRQ OUT bit (if it was set) in the Bus Control Register to indicate that the request was serviced and send its status byte with bit 6 in the Talker Data Buffer set (same as bit 7 of the ASCII code) if it is requesting service. Bit 6 is set for a logical 0 if service was not requested. The controller-in-charge will eventually assert ATN (after the service routine is

completed) and send a Serial Poll Disable (SPD) command or an Other Talk Address (OTA) to complete to process.

If the device or this interface was not unlistened (received a UNListen command) by the controller-in-charge during the serial poll, it must handshake for all status bytes sent over the bus. The device may be aware that it should ignore them, but the IEEE 488 standard does not provide for this situation. Therefore, the controller-in-charge should unlisten all devices before performing a serial poll, and if necessary, restore (readdress) all listeners and talkers when the process is completed.

The procedure for conducting a serial poll as controller-in-charge of the IEEE 488 bus is provided in the discussion of the Controller Function. As controller-in-charge, the program for this interface detects a Service Request from another device on the bus if the SRQ INTR ENBL bit (4) in the Interrupt Control Register has been set by the programmer.

Remote/Local Function

The Remote/Local function provides a device with the capability to select between two sources of input information. The function indicates to the device that either input information from the front panel (LOCAL) or corresponding input information from the IEEE 488 bus (REMOTE) is to be used.

If the program for this interface is to act as the system controller, it must be able to assert the REN line on the bus. To do this, the programmer need only to set the REN OUT bit in the Bus Control Register. Only the system controller is permitted to assert the REN line, whether it is the controller-in-charge or not.

When the system controller asserts the REN line, all devices on the bus go to the REMOTE mode when they are addressed as a listener with their primary addresses, not before. In this case only, the primary address is sufficient to cause the device to go to the REMOTE mode; for example, several devices with different secondary but a common primary address, all go to the REMOTE mode when the primary address is received. The devices remain in the REMOTE mode until either the REN line is cleared, a front panel switch on the device is activiated to request the

LOCAL mode, or a Go To Local (GTL) command is received while the device is addressed as a listener. However, the controller can disable the local mode function of a device by sending a Local Lockout (LLO) command, which applies to all devices on the bus, addressed or not. An UNListen command (UNL) does not return a device to its LOCAL mode function.

If the program for this interface is not the system controller, the Remote/Local function must be implemented in software. To aid the programmer, the REN line can be read with the data in the high order byte of the Listener Data Buffer, making it easy to check if the REN line is asserted when My Listen Address (MLA) is received. The REN FALSE condition is also detected by this interface and may be monitored by setting the REN INTR ENBL bit (1) in the Interrupt Control Register, or by reading the Interrupt Control Register or the Listener Data Buffer and checking the REN FALSE OCC bit. The detection of REN FALSE must be handled with care by the programmer. According to the IEEE 488 standard, the REN FALSE interrupt must be recognized within 100 microseconds. If the REN FALSE interrupt is masked for over 100 microsconds (which may occur during DMA transfers), this consititues a violation of the specification which may be resolved if other protocol is observed. The interrupt prioritization handles this correctly by giving the REN FALSE interrupt higher priority than the RDY interrupt. If the programmer desires to monitor this function in a non-interrupt driven mode, the program must read the Listener Data Buffer and store the data bytes in an internal register (R0-R5), where the REN FALSE OCC bit can be checked to determine if the data is to be interpreted as received under REMOTE or LOCAL mode; this can only be done by reading the REN FALSE OCC bit and the data simultaneously. What must be assured is that any data bytes accepted greater than 100 microseconds after the REN line goes false (unasserted) are received with the knowledge that the controller with this interface is in a LOCAL, not REMOTE, mode.

When receiving data under DMA control, the DMA transfer is interrupted (DMA ABORT bit is automatically set and the GO bit is cleared) if the REN FALSE condition occurs and the REN INTR ENBL bit is set. This allows the programmer to choose whether to interrupt the DMA transfer and read the Bus Address Register to determine which data byte the REN FALSE condition occurred. The DMA transfer may be continued by clearing the REN FALSE OCC bit, performing a DMAINIT and setting the GO bit again.

Parallel Poll Function

The Parallel Poll function provides a device with the capability to present one bit of status to the controller-in-charge without being previously addressed to talk (as in a Serial Poll). The controller-in-charge can conduct a Parallel Poll at any time during a program run, it does not require a SRQ message.

This interface does not allow the program to respond to a Parallel Poll conducted by the controller-in-charge of the IEEE 488 bus, because it cannot assert the Parallel Poll response within 200 nanoseconds upon receiving the Identify command (EOI and ATN both asserted).

This interface does, however, have the capability to allow the programmer to assert the Identify command. The program initiates the Parallel Poll by asserting the EOI OUT and ATN OUT bits in the Bus Control Register. Information concerning how to perform a Parallel Poll as controller-in-charge can be found in the discussion of the Controller Function.

The response which the other devices on the bus make to the Parallel Poll conducted by another controller-in-charge can be obtained directly from the IEEE 488 data bus by reading the Listener Data Buffer (no need to wait for the RDY bit to become set).

Device Clear Function

The Device Clear function provides the device with the capability to be cleared (initialized) by a controller-in-charge, either individually or as part of a group of devices. The group can be either a subset or all of the addressed devices in one system.

The Device Clear function must be implemented in software. If the program for this interface is the controller-in-charge, the programmer may issue a Device Clear command by asserting ATN and sending either the Device Clear message (DCL in the universal command group), or the Selected Device Clear message (SDC in the addressed command group). If the DCL message is sent, all devices on the bus must clear or initialize their internal device functions. IF the SDC message is transmitted, only those devices which are in the Listener Addressed State (LADS) must

perform the clearing or intialization of their internal device functions.

If the program for this interface is not the controller-in-charge, the programmer must check all messages received with the ATN line asserted. If the message is DCL (or SDC and the controller with this interface is in LADS), the program must clear all of the internal device functions required. There is no restriction in the IEEE 488 standard concerning what state the DCL command causes the device to go to; it may, but does not have to be, the power-up state.

Device Trigger Function

The Device Trigger function provides the device with the capability to have its basic operation started by the controller-in-charge, either individually or as part of a group of devices. The group may be either a subset or all addressed devices in one system.

The Device Trigger function must be implemented in software. As a non-controller of the IEEE 488`bus, the program for this interface should check all messages received with ATN asserted. If the Group Execute Trigger (GET) is received and the controller with this interface is in the Listener Addressed State (LADS), it should begin its basic operation immediately. Once a device operation is started, the device must not respond to subsequent Device Trigger state transitions until the current operation is complete. Only after the completion of the operation can the device start the same basic operation in response to the next Device Trigger Active State (DTAS) condition.

To issue the Group Execute Trigger command as controller-in-charge, the programmer sets the ATN OUT bit, sends the Listen Addresses of devices which are to respond to the trigger, then sends the GET message.

OPERATION IN DMA MODE

Introduction

Before operating this interface for a DMA transfer, it is suggested that the software protocol preceding this subsection of the manual and Section 3 of this manual be studied, especially the Interface Status Register, the Bus Control Register, and the Bus Status Register.

The Listener Data Buffer and the Talker Data Buffer are part of the data path between the controller's internal bus and the IEEE 488 bus. The contents of these registers will change rapidly during a DMA transfer. The DMA control logic places a word in the Talker Data Buffer when the DMA IN/OUT bit is clear and accepts a word (low order bits only) from the Listener Data Buffer when the DMA IN/OUT bit is set.

The RDY and DONE bits in the Interrupt Control Register are also used by the DMA control logic during a DMA transfer. The programmer must take care to clear the RDY INTR ENBL and the DONE INTR ENBL bits before intializing a DMA transfer.

When using the DMA control logic to send data with the ATN OUT bit set, this interface automatically inhibits EOI from being asserted during the last data byte transmission, preventing an inadverent Identify command from being sent. The programmer is not required to set the NO EOI DMA bit when the program sends data with ATN asserted.

It is not possible to accept interface control messages (commands) sent by another controller using the DMA controlled handshake. If ATN is asserted by another controller, the DMA ABORT bit is automatically set and the GO bit is cleared. To accept data from another controller that is transmmitted with ATN asserted (which can happen during a DMA transfer), the program should watch for the ATN interrupt and then read each byte received by the Listener Data Buffer.

Setting Up a DMA Transfer

The following six steps may be used as a quick reference to follow when setting up a DMA transfer. There are a number of characteristics of this interface which should be considered when operating it in a DMA mode. These characteristics are summarized in a discussion of the DMA control bits found in the Interface Status Register and the Bus Status Register, as well as the use of the Byte Counter Register and the Bus Address Register.

1. Set the Byte Counter Register (CTR) to the 2's complement of the number of bytes to be transferred.
2. Set up the Bus Address Register (BAR).
3. Select mode for the NO EOI DMA bit and write a 1 to the DMA INIT bit, as well as other desired INTR CLR bits of the Bus Status Register (BSR).
4. Make sure that the appropriate INTR ENBL bits in the Interface Status Register (ISR) are set. Do not set the RDY INTR ENBL bit nor the DONE INTR ENBL bit.
5. Set the desired DMA control bits (1-6) in the ISR.
6. Set the GO bit.

(The last two steps may be done simultaneously.)

Interface Status Register

GO - This bit is used to initiate a DMA controlled data transfer. The GO bit may also be cleared under program control, which simply interrupts the transfer. The DMA transfer may be continued (if the device on the IEEE 488 bus with which it was communicating is enabled to do so) by setting the GO bit again.

NO INCR/DECR - The NO INCR/DECR bit specifies that the Bus Address Register should not be incremented (decremented if the the interface is so strapped) by the interface after each byte of the DMA transfer. It is most commonly used when transferring data directly to or from another peripheral on the controllers internal bus.

DMA IN/OUT - The DMA IN/OUT bit specifies the direction in which the DMA transfer is to take place. If this bit is set, the interface

transfers data from the IEEE 488 bus to the controllers internal bus. If it is clear, the data is transferred from the controllers internal bus to the IEEE 488 bus.

PACK MODE - This bit specifies how the data should be stored (or retrieved) from memory. If the PACK MODE bit is set, the memory is byte addressed so that the word has "packed" data in both the high and low order bytes. The BAR is incremented (decremented if so strapped) by one after each byte transferred and the programmer may start the transfer at an even or odd byte address.

When the PACK MODE bit is clear, the data is assumed to be unpacked (data appears in the low order byte of the word defined by the current value of the Bus Address Register). When transferring data from the IEEE 488 bus to memory in "unpacked" mode, the high byte of the word is cleared during the time the low byte is written. The Bus Address Register is incremented (decremented) by two after each transfer if the NO INCR/DECR bit is not set. The low order bit (A00) of the Bus Address Register is ignored by the slave device when the PACK MODE bit is not set. As a rule, the programmer should load the BAR with an even address in the unpacked mode.

HOG MODE - This bit defines how the controllers internal bus should be used during a DMA data transfer. When it is clear, the interface gets control of the internal bus, transfers one byte, and releases the bus so that the processor or some other DMA peripheral may share the bus cycles. With the HOG MODE bit set, the interface gets control of the bus and "hogs" it (doesn't release it until the DMA operation is completed), or the device on the IEEE 488 bus "times out" (doesn't respond within 20 microseconds). This "time out" is not considered an error; the device was slow to respond and as soon as it does respond, this interface automatically proceeds with the DMA transfer.

SWAP BYTE ORDER - This bit is used to control the manner in which memory is addressed when in packed mode. If it is clear, the memory is addressed in the standard fashion, incrementing (decrementing if the interface is so strapped) after each cycle, the address applied to the bus.

If this bit is set, the order in which the bytes within a word are addressed is swapped. Thus, if the interface is strapped to increment,

the high order byte of the word is addressed first. For decrementing with the SWAP BYTE ORDER bit set, the low order byte is addressed first.

This swapping of the bytes within word boundaries is particularly useful for 16-bit binary integer data transmissions which need to be stored in memory with one format and transmitted in another. For example; suppose memory is allocated such that for 16-bit integer arrays, the first array element is assigned a lower address than the last element and there is a need to transmit (receive) the array element by element with the high order byte first. To perform this operation:

* Set interface strap (H1) to increment the BAR.
* Set address of first element (word address) into the BAR.
* Byte count is assigned 2's complement of twice the number of array elements.
* Enable the DMA transfer with both the PACK MODE and SWAP BYTE ORDER bits set.

The byte swapping is accomplished by complementing the least significant bit of the address as the register is read (the actual value in the BAR is not affected). The programmer should be aware that if he writes an even address into the BAR when this bit is set and reads the address back immediately, the least significant bit will appear set. This is to indicate the actual address from (into) which the next byte of data is transferred.

The following examples illustrate the use of the SWAP BYTE ORDER bit when this interface is operating in the packed mode (PACK MODE bit set).

If the character string "ABCDEF" is sent over the IEEE 488 bus to this interface, the six bytes will be placed in memory as shown below for the four cases. It is assumed that in each case the Byte Counter Register is set to 177775 (octal).

Case 1. Increment with SWAP BYTE ORDER bit clear and the BAR = 147000 (octal).

```
147000  B A
147002  D C
147004  F E
```

Case 2. Increment with SWAP BYTE ORDER bit set and the BAR = 147000 (octal).

```
147000  A B
147002  C D
147004  E F
```

Case 3. Decrement with SWAP BYTE ORDER bit clear and the BAR = 147005 (octal).

```
147000  E F
147002  C D
147004  A B
```

Case 4. Decrement with SWAP BYTE ORDER bit set and the BAR = 147005 (octal).

```
147000  F E
147002  D C
147004  B A
```

DMA INTR ENBL - The interrupt enable bit is set to enable an interrupt on DMA DONE or one of the error conditions flagged in the ISR.

DMA DONE - This bit indicates that the DMA operation completed normally; the Byte Counter Register overflowed (all is to all 0's) or an END condition occurred when receiving data from the IEEE 488 bus. If the amount of data to be received is not known, the byte count may be set to the amount of space available in memory for the transfer. If the EOI message is received before the space if filled, the value of the Byte Count Register (originally loaded with the 2's complement of the number of bytes to be transmitted) added to the size of the block allocated (in bytes for packed mode) is the number of bytes actually transferred. The Bus Address Register contains the address into (from) which the next byte would be moved.

DMA ABORT - The DMA transfer may be interrupted by one of the following four conditions:

1. If Interface Clear (IFC) is asserted during a DMA transfer.

2. If ATN is asserted (by another device) during a DMA transfer.

3. If a REN FALSE condition occurred with the REN INTR ENBL bit set during a DMA transfer.

4. If a TIMING ERROR, WRITE ERROR or DMA ERROR occurred.

When the DMA transfer is interrupted, the GO bit is cleared and the DMA ABORT bit is set. This will set the ERROR bit and generate an interrupt (if the DMA was interrupted by one of the errors listed in #4 above, only one interrupt is generated since the ERROR bit was already set).

DMA ERROR - This bit is set if during a DMA transfer, the BAR overflows or this interface detects that the device (memory) on the controllers internal bus to (from) which it is transferring data did not respond. It is cleared by either the LOCAL RESET or DMA INIT being set.

Bus Status Register

LOCAL RESET - Writing a "1" to this bit performs the same operation to this interface board as occurs when a controller RESET instruction is executed. This may be used to re-initialize this interface without affecting the other interfaces (peripherals) in the system. See Table 3-3 (Section 3) for specification of which bits are changed by asserting LOCAL RESET.

DMA INIT - Writing a "1" to this bit clears the ERROR occurred bits, the DMA DONE bit and the GO bit of the ISR. The programmer should always set this bit before a DMA operation to make sure that the interrupt can occur. If both the DMA INTR ENBL and DMA DONE or ERROR are set, then one must be reset before another interrupt can be generated.

NO EOI DMA - This bit, if set, indicates that the END condition should be ignored during a DMA transfer. For a DMA transfer from the IEEE 488 bus to the controllers internal bus, this means that the DMA transfer is not terminated (DMA DONE bit set) by EOI being sent with a data byte. The DMA continues to accept data from the IEEE 488 bus until the Byte Count Register overflows or the DMA transfer is interrupted.

For DMA data transfers from the controllers internal bus to the IEEE 488 bus, this bit (when set) inhibits the interface from sending the EOI message along with the last byte of the transfer. This may be used to transfer several blocks of data stored in non-consecutive locations in memory as a single message.

If the ATN OUT bit of the BCR is set and a DMA transfer to the IEEE 488 bus is initialized, the EOI is automatically inhibited. (This bit does no need to be set when transferring data with ATN asserted.)

Byte Counter Register

The Byte Counter Register should be loaded with the two's complement of the number of bytes to be transferred. Each time a byte is transferred, the byte count is incremented. When the Byte Counter Register overflows (all ones to all zeros), the DMA transfer is terminated (DMA DONE bit set).

Bus Address Register

The Bus Address Register should be loaded with the address into (from) which the data is to be transferred. Whether it is incremented or decremented is controlled by setting a strap on the interface board. The NO INCR/DECR bit in the ISR can be set to indicate that the same address should be used for the entire DMA transfer. The PACK MODE bit controls how many times the Bus Address Register is incremented (decremented) for each byte transferred. If it is set, the BAR is incremented (decremented) once for each transfer, while if clear, the BAR is incremented (decremented) twice for each byte transfer, addressing one byte in each word. The SWAP BYTE ORDER bit does not affect the contents of the BAR, only whether the least significant bit is complemented when it is read or used as an address on the controllers internal bus. Care should be taken in setting up the BAR if the SWAP BYTE ORDER bit is set, making sure that the address loaded is the first byte of the block to be transferred. See the examples in the discussion of SWAP BYTE ORDER bit, case 2 and case 4.

WARNING

THE FOLLOWING SERVICING INSTRUCTIONS ARE FOR USE BY QUALIFIED PERSONNEL ONLY. TO AVOID PERSONAL INJURY, DO NOT PERFORM ANY SERVICING OTHER THAN THAT CONTAINED IN OPERATING INSTRUCTIONS UNLESS YOU ARE QUALIFIED TO DO SO.

SECTION 5

SERVICING

Strap Options

Strap options on the CP1100/IEEE 488 Interface card allow the selection of register addresses, interrupt vector addresses, incrementing or decrementing the Bus Address Register, and inhibiting or enabling the non-processor grant (NPG H) signal.

Register Addresses

The address format for the eight programmable registers is illustrated in Fig. 5-1.

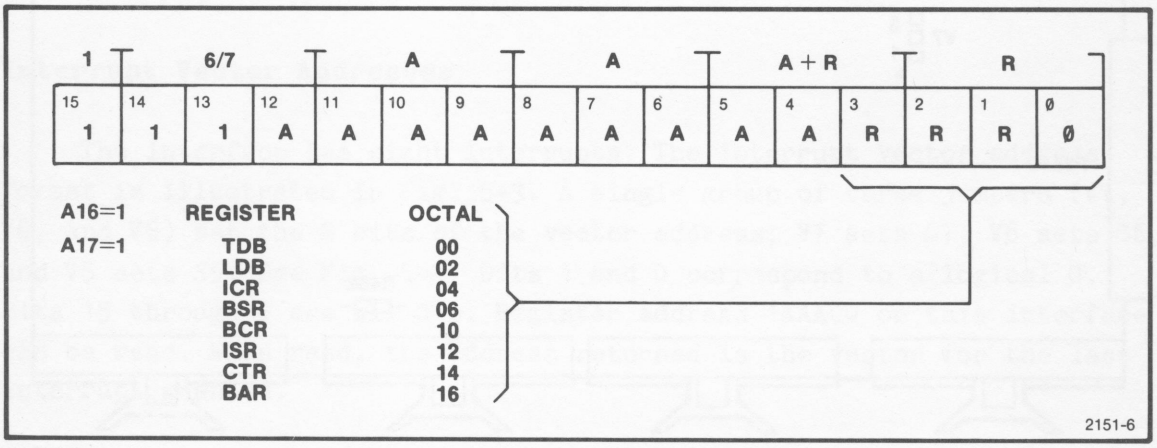


Fig. 5-1. Register address format.

Address bits 15 through 13 are hardwired for a logical 1. Bits 12 through 4 are set by a single group of nine jumpers (straps) as shown in Fig. 5-2. These nine jumpers allow the value of the lowest numbered register (1AAA00 - Talker Data Buffer) to be set within the range 160000_8 through 177760_8. Bit 12 is set by jumper A12, bit 11 by A11, bit 10 by A10, etc. As a visual aid, a caret is etched on the circuit board to designate the strap position for a logical 1.

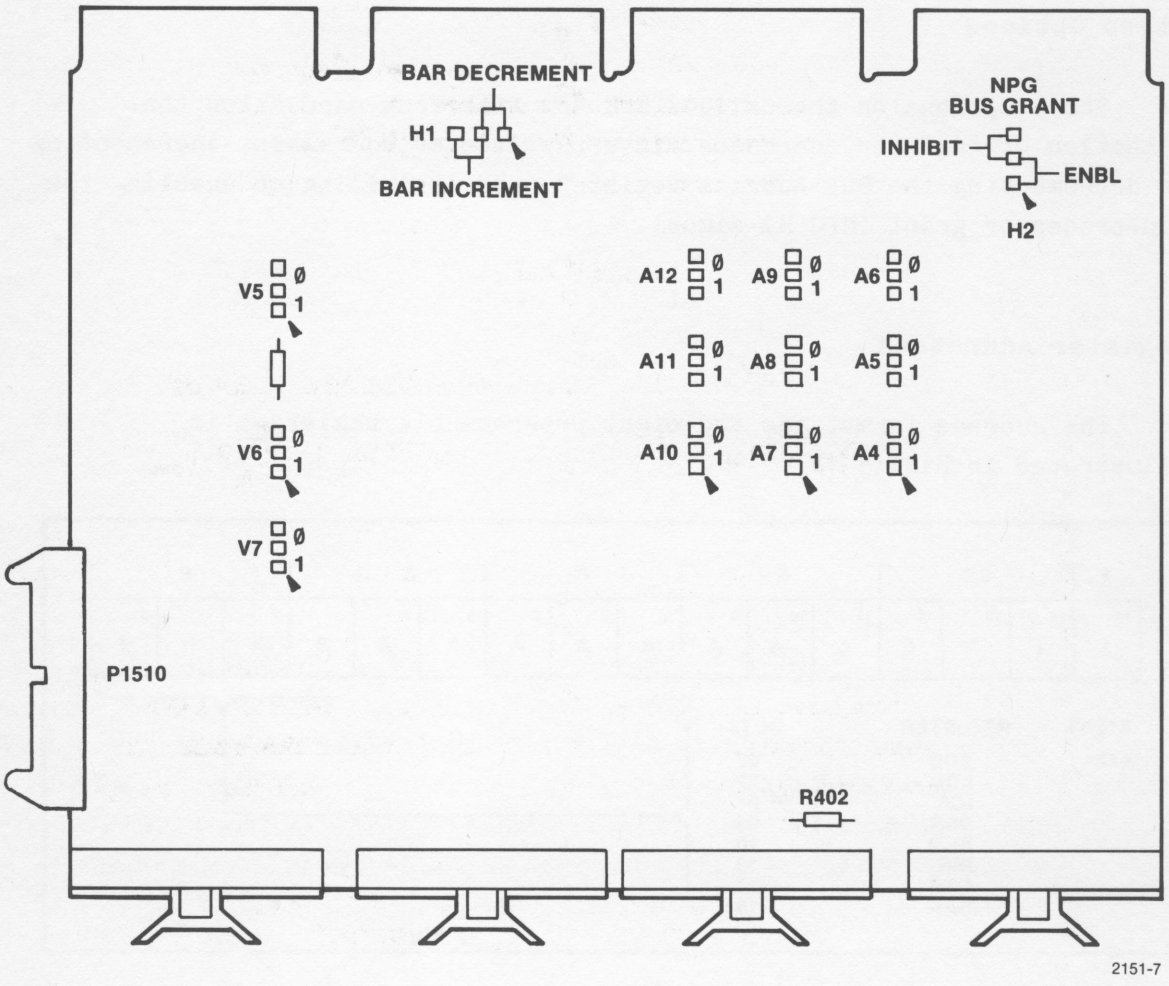


Fig. 5-2. Jumper selection on the CP1100/IEEE 488 Interface.

When bits 12 through 4 have been set, bits 3 through 0 are decoded to address (select) the desired register on this interface. Bit 0 is equivalent to a logical 0. As with any peripheral register connected to the CP1100 bus, the programmable registers on this interface must be addressed with the extended address bits (A17, A16) set.

For TEK SPS BASIC Software, the upper bits of the address decoding logic are set for 164100_8; A11 and A6 set to a logical 1, the remaining seven address jumpers set to a logical 0. TEK SPS BASIC Software allows the use of more than one CP1100/IEEE 488 Interface. The address straps of each interface can be set to create successively numbered addresses. For example:

Interface 0	164100_8 through 164116_8
Interface 1	164120_8 through 164136_8
Interface 2	164140_8 through 164156_8

Interrupt Vector Addresses

The interface has eight interrupts. The interrupt vector address format is illustrated in Fig. 5-3. A single group of three jumpers (V7, V6, and V5) set the S bits of the vector address; V7 sets S7, V6 sets S6, and V5 sets S5. See Fig. 5-2. Bits 1 and 0 correspond to a logical 0. Bits 15 through 8 are all 0's. Register address 1AAA00 on this interface can be read. When read, the address returned is the vector for the last interrupt granted.

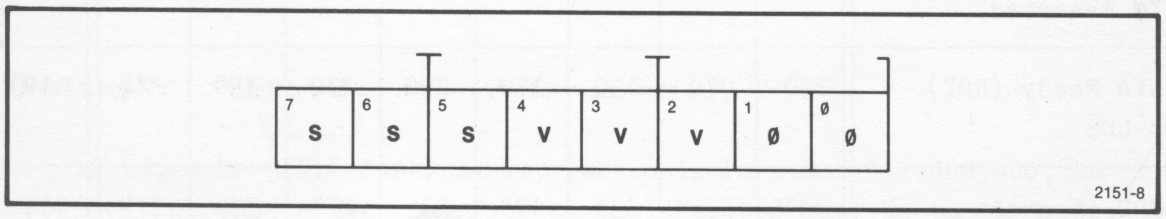


Fig. 5-3. Interrupt vector address format.

The vector address straps allow the lowest vector address (highest priority vector) to be set within the range 000_8 to 340_8. Table 5-1 contains a list of interrupt vectors and their corresponding addresses (listed in order from highest to lowest priority) for eight different strap settings. For TEK SPS BASIC Software, set V7 and V6 for a logical 1, V5 for a logical 0; vector strap setting is 110 and the SRQ vector address should be 320.

TABLE 5-1
Interrupt vectors and their corresponding addresses for different vector strap settings.

Vector Strap Settings (V7,V6,V5)	000	001	010	011	100	101	110	111	V-bit Code
DMA DONE (Complete) or ERROR	000	040	100	140	200	240	300	340	000
Interface Clear IFC Detected	004	044	104	144	204	244	304	344	001
Remote Enable REN FALSE	010	050	110	150	210	250	310	350	010
END of Message DAV and EOI Asserted	014	054	114	154	214	254	314	354	011
Service Request SRQ Asserted	020	060	120	160	220	260	320	360	100
Attention ATN Asserted	024	064	124	164	224	264	324	364	101
Data Ready (RDY) in LDB	030	070	130	170	230	270	330	370	110
DONE (Source Handshake Completed)	034	074	134	174	234	274	334	374	111 (NONE)

Operating Mode Jumpers (H1 and H2)

Jumper H1 is set to its indicated position to increment or decrement the Bus Address Register during DMA transfers (see Fig. 5-2).

Jumper H2 allows the user to inhibit or enable the NPG H daisy-chain path on the UNIBUS. If this interface is not the last device in the NPG H daisy-chain path, the associated wire wrap section between CA1 and CB1 on the UNIBUS backplane must be cut and jumper H2 set to the ENBL (enable) position. If this interface is the last device in the daisy-chain path, the associated wire wrap section does not have to be cut, but jumper H2 must be set to the INHIBIT (disable) position. If this interface is permanently removed from its original system unit slot and the wire wrap was cut in the daisy-chain path, reconnect the associated wire wrap connection between CA1 and CB1.

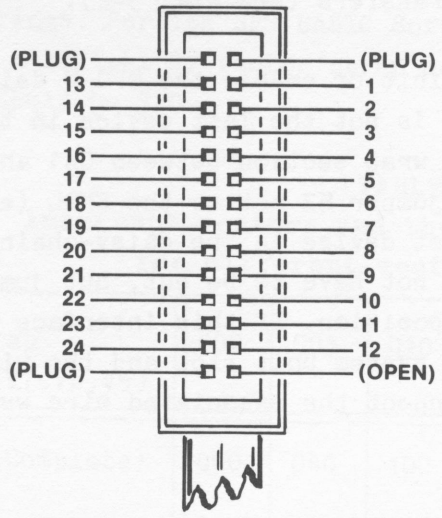
Calibration for DMA Transmission Rate (R402)

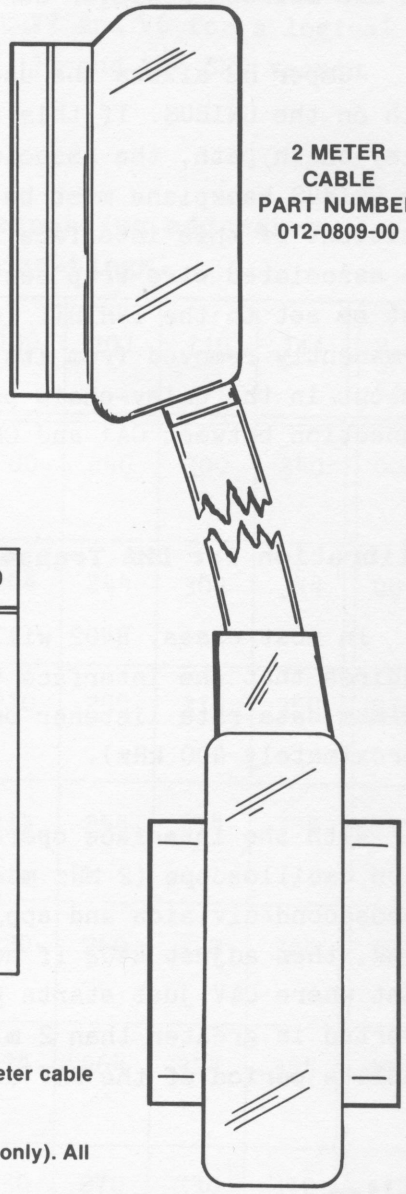
In most cases, R402 will not need adjustment. The adjustment of R402 requires that the interface be operated in the DMA hog mode talking to a maximum data rate listener on the bus (data transfer rate of approximately 400 kHz).

With the interface operating under the above stated conditions, set up an oscilloscope (2 MHz minimum bandwidth) for a sweep rate of 0.5 microsecond/division and apply the DAV bus signal to the vertical input. Check, then adjust R402 if necessary, so that the time interval from the point where DAV just starts going positive to the point where DAV becomes asserted is greater than 2 microseconds (about 2.3 microseconds). The complete period of the DAV signal must be less than 2.5 microseconds.

Cabling

A two-meter (6.6 feet) cable, Tektronix Part No. 012-0809-00, is provided to connect the interface to the IEEE 488 bus. Up to 15 devices can be daisy-chained on the bus. Maximum allowable cable length is 20 meters. Fig. 5-4 shows the interconnecting pin assignments between P1510, the 2 meter cable, and the IEEE 488 bus.





P1510	2 METER CABLE				P1510
—	(PLUG)	X	X	(PLUG)	—
B	D105	13	1	D101	A
D	D106	14	2	D102	C
F	D107	15	3	D103	E
J	D108	16	4	D104	H
L	REN	17	5	E01	K
N	GND(6)	18	6	DAV	M
R	GND(7)	19	7	NRFD	P
T	GND(8)	20	8	NDAC	S
V	GND(9)	21	9	IFC	U
X	GND(10)	22	10	SRQ	W
Z	GND(11)	23	11	ATN	Y
BB	GND,LOGIC	24	12	SHIELD	AA
—	(PLUG)	X	0	(OPEN)	—

NOTE 1: P1510 is on the circuit card. The pin numbering for 2 meter cable matches the IEEE standard connector.

NOTE 2: P1510 only—+5V dc on pins UU and VV (for test fixtures only). All other unused pins go to ground.

2151-9

Fig. 5-4. Interconnecting pin assignments for P1510, 2 meter cable, and the IEEE 488 bus.

Installation

Before installation, read and perform the instructions for Operating Mode Jumper H2. Check or set the required addresses for the programmable registers and the interrupt vectors. Check for the correct operating position for jumper H1.

Turn off the controller power. Slide the controller out of the rack. Remove the top cover. Route the bus cable into the controller from the rear and connect it to the connector on the interface card. Insert the card into the previously selected slot in the controller backplane. Secure the cable at the back of the controller with the strain relief bar. Replace the controller cover and slide the controller back into the rack.

IEEE Standard 488-1975 states that a system must be operated with at least one more than half the devices with power turned on. Note that powering up a device while the system is running may cause faulty operation.

General Maintenance

Follow maintenance procedures for the controller in which this interface is installed. The interface should be cleaned by first loosening dust and other foreign matter with a small paint brush. Then, use a small vacuum cleaner or dry low-pressure air to remove the loosened material. Persistent dirt and most stains can be removed with a soft cloth or cotton swab dampened in a water and mild detergent solution. Isopropyl alcohol is recommended only when cleaning solder flux or oxide deposits.

CAUTION

Avoid the use of chemical cleaning agents that will damage the plastic materials on this interface. In particular, avoid chemicals that contain benzene, toluene, xylene, acetone, or similar solvents. Do not use immersion techniques to clean the interface.

Troubleshooting any apparent malfunctions of the interface requires an understanding of circuit operation for this interface and the controller. Study the block diagram, register descriptions, and circuit descriptions. For information on how circuit operation can be programmed, see Section 4. For general information concerning the operation of the controllers internal bus and the input/output operations for the controller, refer to the CP1164 Service manual or the PDP-11 Peripherals Handbook (published by Digital Equipment Corporation).

Static Sensitive Parts

This instrument contains electrical components which are susceptible to damage from static charges. Observing the following precautions will aid in avoiding the breakdown of these parts. When returning static sensitive parts to TEKTRONIX, INC. package them in antistatic or conductive material.

CAUTION

1. Minimize the handling of static-sensitive parts.

2. Transport and store static-sensitive parts in their original containers, on a metal rail, or on conductive foam.

3. Discharge the static charge on yourself by physical contact with your work station ground before handling these devices.

4. Keep the leads shorted together whenever possible.

5. Pick up the part by the body, never by the leads.

6. Do not subject the part to sliding movements over any suface.

7. Avoid handling parts in areas having a floor covering that contributes to the generation of a static charge.

8. Use a soldering iron that has a connection to earth ground.

9. Use a special suction type desoldering tool, Silverstat Soldapullt® (Tektronix Part Number 003-0795-00), or a wick type desoldering device (Tektronix Part Number 006-1356-00).

10. Be careful when using a multimeter (VOM) for measurements on static sensitive devices. Certain devices cannot sustain more than 10 mA of test current. Check your VOM for maximum output current capability on all resistance ranges.

SECTION 6

CIRCUIT DESCRIPTION

Introduction (Block Diagram)

A two page block diagram precedes the schematic diagrams. It can be used to identify the circuit components associated with the valid bits in each of the eight registers and the interrupt vectors, along with the logic gates required for the register interrelationships. The block diagram, when studied along with the information found in Section 3 (Interface Registers and Interrupt Vectors), serves as an aid in understanding the over-all operation of the interface. Note that the block diagram is drawn to aid in understanding the operation of the interface registers; it does not exactly match the schematic diagrams.

Schematic Diagrams

The schematic diagrams are to be studied in accordance with the following conventions:

Interface signals whose names are shown with a bar above them are true when at a TTL low voltage level (less than 0.8V). A circuit component whose input or output terminal is true low is shown with a small circle at the pin connection. When the inputs are true, the output asserts the true level implied by the signal name; high if no bar, low if a bar appears above the name. Not all inputs respond to a voltage level. Clock inputs respond to a dynamic transition and follow the convention that a positive transition is shown with no circle, a negative transition is shown with a small circle.

Some TTL high voltage levels are set by a resistor network at the output or input of a component. These high voltage levels can measure between +2.4V and +5.0V.

The signal names for the CP1100 bus and the functions they perform can be found in the CP1164 Controller Service manual. They can also be found in the PDP-11 Peripheral Handbook, published by Digital Equipment

Corporation. Section 2 of this manual can be studied to identify the functions performed by the signal names on the IEEE 488 bus. Refer to Sections 3 and 4 for information relative to programming this interface.

Bus Address and Control

Refer to schematic diagram 5 for this discussion. The 18 address lines, A00-A17, on the CP1100 bus carry 18 address bits from the bus master during a data transfer operation. These bits, when received and decoded by the interface, specify one of the eight programmable registers. When addressed properly, the interface responds as a slave device for the data transfer. All addresses received are even numbered.

When the interface receives permission to become bus master, it asserts BBSY. As bus master, it can use its Bus Address Register to select a location on the internal bus and send data to or receive data from that addressed slave location. The address lines, control lines, and data lines are all asserted at the same time. The bus master always asserts BBSY before placing address/data on the bus.

Note that address line A00 and control line C0 are ignored when received by the interface (pin 13 of U660 and pin 10 of U360 are not connected). This means that write-byte operations are not supported. If a write-byte operation is performed by the user, the addressed (selected) register on the interface will typically be loaded with the data byte in both the low-order and high-order byte positions of the register.

This interface responds to addresses 764000_8 through 777760_8 to allow systems with Memory Management option. When addressing a register on this interface, address bits A17, A16, A15, A14, and A13 must be set for logical 1s. If the Memory Management option is not used, an octal address 164000_8 through 177760_8 will be automatically interpreted by the processor as 764000_8 - 777760_8 if address bits A15, A14, and A13 are 1s.

Address bits A15-A13 are applied to U550 (pins 6, 4, and 2) where they are compared with the high voltage level (a logical 1) hardwired to pins 5, 3, and 1. Address bits A12-A04 are applied to U450 and U550 (hex comparator/latches) where they are compared with the settings of register address straps A12-A4. A strap is set to a high voltage level for a logical 1 and a low voltage for a logical 0. The comparators have open-

collector outputs connected in a wired-OR configuration. A high state is the result of a correct address comparison. With the extended address bits (A17 and A16) set, three inputs to U740B are set to a high state.

After a short delay to allow the bus lines to settle and address decoding to be performed, the bus master (processor in this case) asserts MSYN. MSYN is a strobe, or gating signal, for the address and control lines; it is always cleared before address and control are changed. The output of U740B goes low to indicate that this interface has been selected for a data transfer.

When the output of U740B goes low it triggers and sets U330A (a one-shot) for about 75 nanoseconds. U330A then resets itself. The action of U330A (pin 7 going low for about 75 nanoseconds) enables pin 5 of U750, a 3-line to 8-line decoder, and sets U510D. For a data transfer to this interface, the bus master has asserted the C1 control line, enabling pin 6 of U750 through U640B. U750 then decodes the binary levels on the data selector bank lines (SEL3, SEL2, and SEL1) to load one of seven addressed registers determined by address bits A03, A02, and A01 (A00 will be ignored).

When U330A resets itself, the output of U000C goes high to assert SSYN; this tells the bus master that data has been accepted. The bus master then negates MSYN, data is removed from the bus, and this interface negates SSYN by resetting U510D through U350B.

The Listener Data Buffer is not selected by U750. Data contained in this register is always transferred from the IEEE 488 bus to the CP1100 bus and is a DATA OUT operation with respect to this interface. The Listener Data Buffer is read from the Data/Selector Bank (schematic diagram 1) when the binary (octal) code for SEL3, SEL2, and SEL1 is set for 001. Address bits A03 and A02 are ANDed together and inverted at the output of U540C to enable pin 4 of U740A (A01 enables pin 5). Control line C1 is not asserted for a Listener Data Buffer read operation and both inputs to U540B go low. The high level output from U540B completely enables U740A to clear the RDY bit in the Interrupt Control Register (schematic diagram 3) and simultaneously enables the CP1100 data bus transceivers (schematic diagram 1) through U540D. When the bus master has read the Listener Data Buffer it releases MSYN. This clears U330A and resets U510D to negate SSYN. The negation of SSYN informs all bus devices that the slave has concluded the transfer. For a DATA OUT operation the

negation of SSYN indicates that the negation of MSYN has been received.

For DMA transfers the DMA State Machine (discussed in a separate subsection) asserts BBSY and SACK. This interface then becomes the bus master for data-in or data-out operations. For data output to memory or another device on the bus, the data bus transceivers are enabled through U540D. The address bus transceivers are enabled by the ADDR line from the DMA State Machine being set low true. Data and address transceivers are enabled simultaneously.

When the DMA State Machine is reading the Listener Data Buffer for a DATA OUT operation, both inputs to U340C are high true and both inputs to U640B are high true (the high output of U340C asserts control line C1 and sets pin 5 of U640B high). U750 is disabled by the low output from U640B. With the ADDR line high true at the input to U530B, the information contained in the Listener Data Buffer is placed on the CP1100 data bus from the Data/Selector Bank (the LDB code for SEL3, SEL2, and SEL1 is 001, schematic diagram 1).

If the DMA State Machine is operating in the packed mode (PACK MODE bit set) when listening to the IEEE 488 bus, pin 9 of U340D is set high. This causes control line C0 to be asserted with control line C1. Both control lines asserted tell the slave device that a data byte is on the CP1100 bus. The BYTE control line from U340D is held low when the data is not packed (a complete 16-bit data word is placed on the bus). For DMA transfers from the IEEE 488 bus to the CP1100 bus in the packed mode, the high level on the BYTE control line causes the low-order byte from the LDB (D00-D07) to be placed in both the low-order and high-order byte positions on the CP1100 bus. The slave device then loads the high-order or low-order byte, depending on whether the least significant bit of the address was a 1 or a 0.

When the DMA State Machine is transferring data from the CP1100 bus to the IEEE 488 bus, the BYTE control line is held low by U340C being disabled at pin 9. Disabling U340C also asserts control line C1 and sets pin 5 of U640B and U540B both low. This partially enables U750 with a register select code of 001 on SEL3, SEL2, and SEL1. The machine then loads the Talker Data Buffer by asserting pin 4 of U340B, which sets pin 14 of U750 low. For non DMA transfers, pin 15 of U750B is set low with a register select code of 000. U540A ensures that, for either mode, the Talker Data Buffer is not loaded during the time that DAV is asserted on

the IEEE 488 bus. The TDB can only be loaded when DAV is not asserted.

Note that this interface, when so programmed, can assert the extended address bits through address transceiver U760. The function of these bits, XBA17 and XBA16, are discussed under Bus Control Register in Section 3.

Data/Selector Bank

Refer to schematic diagrams 1 and 5 for this discussion. The data transceivers (U1260, U1360, U1460, and U1560) are enabled with a high true or low true level from U540D. They are enabled with a high level on pins 7 and 9 when data is to be transferred from the CP1100 bus to this interface (a write operation) and enabled with a low level on pins 7 and 9 when data is to be transferred from the interface to the CP1100 bus (a read operation, or interface DATA OUT).

Data selectors/multiplexers are used to select valid data from the registers during a read operation. The data select code (octal) on selector lines SEL3, SEL2, and SEL1 is as follows: Interrupt Vector Data - 000, LDB - 001, ICR - 010, BSR - 011, BCR - 100, ISR - 101, CTR - 110, BAR - 111.

Valid bits 0 through 7 from the registers are multiplexed by data selectors U1250, U1240, U1230, U1220, U1350, U1340, U1330, and U1320, respectively. Valid bits 11 through 15 from the registers are multiplexed by data selectors U1430, U1150, U1050, U950, and U850, respectively. Bit 8 from six registers is blank, while bits 9, 10, 11, and 12 in the BSR are write only. U1440 selects only bits 8, 9, and 10 from the CTR and the BAR. U430C operates as a strobe gate for U1440 and has a low-true output only when the BAR or CTR has been selected for a read operation. Pin 1 of U1440 is encoded low true to select the CTR bits and encoded high true to select the BAR bits. Data on the input selected is transferred through the data selectors without inversion. Inversion occurs in both directions through the data bus transceivers.

When the DMA State Machine is outputting data to the CP1100 bus in the packed mode, the BYTE control line (U1450 and U1550, pin 1) is high true, which selects the low-byte data inputs to U1450 and U1550; data line D15 will then correspond to data line D07 and data line D08

corresponds to data line D00.

The ADDR line is held low true to enable bits 12 through 15, except during DMA transfers to the CP1100 bus. The DMA State Machine sets the ADDR line high when reading the Listener Data Buffer. This causes bits 12 through 15 of the LDB to return all 0s when read.

Interrupt Control Bus

This interface has eight interrupts. The basic logic functions required for the generation of an interrupt request are shown on schematic diagram 6. The eight interrupt request sources originate from the appropriate interface registers on schematic diagrams 2 and 3. A typical interrupt request for the END condition is described below. The remaining seven interrupts operate in a similar manner.

The generation of the END interrupt requires that the END OCC bit in the ICR be set along with the condition that the programmer has set the END INTR ENBL bit. The END OCC bit becomes set when the DAV and EOI lines on the IEEE 488 bus are both asserted (with ATN unasserted). The END INTR request (a positive transition at pin 11 of U920B) clocks pin 6 of U920B to a high state. When the END INTR request signal reaches its high state it enables U900C, producing an END request signal on pin 8 of U1100 and asserting the INTR REQ signal through U1110 to the interrupt request flip-flop, U040A (Schematic diagram 4). Note that if the source of the interrupt request is terminated, the output of the nand gates will go false (high), causing INTR REQ to go false. The octal D-type flip-flop operates as a synchronizer.

The events following the generation of the INTR REQ signal at the output of U1110 is discussed in more detail under a separate subsection, Interrupt and DMA State Machine. For this discussion, if the DMA State Machine has not been granted bus mastership before it reaches state 4, the END interrupt request signal (low true) at pin 8 of U1100 is clocked and latched into U1100 and sets pin 1 of U1200 low true. The DMA State Machine also generates the VECTOR signal which enables U1300.

U1200 operates as an 8-line to 3-line priority encoder. If an interrupt is generated and then clocked by the DMA State Machine, an octal code is set on the output of U1200 (110 for END). The octal code is

applied to the input of U1300 (a 3-line to 8-line decoder) and sets the V-bit code for VEC4, VEC3, and VEC2. VEC7, VEC6, and VEC5 are set by the user. VEC1 and VEC0 are logical 0s. The vector address for the last interrupt granted can be read by the processor at any time.

For the END interrupt, the octal code of 110 at the input to U1300 asserts the Y3 output and presets (initializes) the END flip-flop to a 1. With the flip-flop set, another interrupt will not be generated unless the source of the interrupt is cleared and set again.

Interrupt and DMA State Machine

Introduction. Refer to schematic diagram 4 and the state diagram (foldout) for the DMA State Machine for this discussion.

The direct memory access (DMA) logic functions for the DMA State Machine conduct data transfers in both directions across this interface, placing data directly in memory (peripherals) or in devices on the IEEE 488 bus. If the interface is operating in the non-hog mode, it will share bus cycles with the processor or other peripherals on the bus. When operating in the hog mode, the interface will relinquish control of the CP1100 bus during DMA transfers if an addressed device on either bus fails to respond within 20 microseconds. Memory addressing, timing, and control of signal generation/response are provided by the DMA State Machine; the processor need only to set up the registers properly before the start of a data transfer.

The interface requests control (mastership) of the CP1100 bus for a data transfer by asserting NPR. The priority arbitration logic responds by asserting NPG. NPG is high true when asserted on the bus. The interface then responds by asserting SACK to acknowledge the grant. As soon as NPG and BBSY are negated on the bus, the interface assumes bus mastership by asserting BBSY.

Schematic diagram 4 also contains the logic functions to handle the interrupt requests generated by this interface. The logic functions to request bus mastership for interrupts and to receive the grant are not part of the DMA State Machine. The request and grant can occur during a DMA transfer, but they are not processed until the completion of a machine cycle.

Machine Operation. The DMA State Machine has 16 distinct states, 0 through 15. These states are firmware controlled by data stored in U130 (PROM) and latched in U140. The data ouput from these devices automatically sets up the next desired state address for the machine. U130 and U140 are programmed to cause the machine to either wait in an addressed state for a condition to be cleared, branch to the desired state, or advance to the next state.

U150 and U250 along with gates U110C and U110D operate as decision-making circuits. If the output of U150 or U250 is high, the machine either waits or branches, depending on which logic gate, U110D or U110C, is enabled by the data output from U140. Data on pin 12 of U140 is set high to enable the wait function (U110D) and set low to enable the branch function (U110C).

U210A and U210B operate as the machine's internal clock. Its output (pin 5) is high between states and then goes low to enable the state decoder devices, U120 and U220. These devices are selected by address/data bit SA4 (0 = U120, 1 = U220). The clock output remains low while the machine is waiting. After the required events have occurred, it goes high to latch the next address/data output from U140 and U130. The state decoder devices are disabled when the clock output is high (between states).

The same four address lines that drive the state decoders also drive the two selectors, U150 and U250. These devices (combined) serve to select one of 16 inputs, depending on the machines current state. The selected input is then gated to become the condition for a possible branch, or alternatively, to cause the internal state clock to pause until the wait condition becomes unasserted.

The 16 distinct states are identified (set) with the low level outputs from U120 and U220. U120 sets states 0 through 7, while U220 sets states 8 through 15. Pin 15 of U120 is set for state 0 and pin 7 is set for state 7. Pin 15 of U220 is set for state 8 and pin 7 is set for state 15.

For this discussion it is assumed that the programmer has set up the desired DMA program and the machine has just been addressed for state 0. The terms "talking" and "listening" mean that this interface is either outputting data (talking) or accepting data (listening) on the IEEE 488

bus. The DMA signal line (input to U010D and U800C) will be high when listening and low when talking.

The data output from U140 is cleared through U440B and U240B when a register on the interface is addressed (selected) or an INIT (reset) is performed. U140 is also cleared (reset) through U240A if an interrupt is generated before state 9 is reached. U130, pin 7, enables the reset function (U240A) only during states 5, 6, 7, and 8. The State Machine starts out in state 0 due to INIT. It then branches to state 6 and normally idles there. When an interrupt has been granted, it resets to state 0.

State 0 contains a branch instruction and U110C is enabled at pin 9. When the machine enters state 0, pin 15 of U120 goes low. This sets U510C through U200B, resets U020B through U200A, and resets U510B through U110B. This operation sets the VECTOR signal high (false), enables U340A, clears MSYN on the bus, and clears NPR through U160D.

If an interrupt has occurred, pin 14 of U040A is set high and BR5 has been asserted on the bus. Q054 is on, U060D is disabled, and BG5 is held low before the bus grant, level 5, is received. When BG5 is received from the priority logic circuits, the negative transition clocks U040A to a set state and turns off Q054. The input logic levels to U060D have now reversed and the bus grant signal (BG5) is not passed down the daisy-chain path on the bus. If an interrupt did not occur, U040A would not be set and both inputs to U060D would be high when BG5 was received; this allows BG5 OUT to be asserted high true and be sent to the next device on the bus. If U040A becomes set when BG5 is received, it will be cleared through U030C when BG5 is negated.

Also, if an interrupt did not occur, U020C (SACK OUT) remains clear and pins 4 and 3 of U150 are high. This establishes the branch condition and the machine will branch to state 6 (discussed later). Logic inversion does not occur in U150 or U250.

Assuming an interrupt did occur, pin 13 of U040 goes low when the bus grant is received and sets U020C. SACK OUT is asserted low true at this time. Pins 4 and 3 of U150 are set low (no branch), so the machine advances to state 1 when the internal clock goes high and latches the address/data for state 1.

In state 1, no new operations are performed because pin 14 of U120 is not connected. However, the machine checks the interrupt status again. If SACK OUT is not asserted, the machine returns to state 0, checks again, and branches directly to state 6. If SACK OUT is asserted, an interrupt request has been made from one of the eight sources on this interface and the bus grant has been received; the machine will then advance to state 2.

State 2 contains a wait instruction. When state 2 is entered, pin 13 of U120 goes low to prevent U1100 (schematic diagram 6) from latching new data for another interrupt until the current interrupt has been serviced. New interrupt data can only be latched in U1100 when the machine leaves state 2. Since the machine has been instructed to wait, the output of the internal state clock remains low while a check is made on the BBSY signal (U150, pin 2). If the bus is busy, pin 2 of U150 will be high and the machine waits with the output of U110D and pin 9 of U210B (clock) high. When the current bus master releases BBSY, pin 9 of U210B goes low and the address/data for state 3 is latched in U140.

State 3 is entered when the clock output goes low and pin 12 of U120 is set low true. The machine resets U510C and the VECTOR signal goes low true for the indicated interrupt. U340A is disabled and BR5 is negated (set low); this delays the next interrupt request from this interface. U1100 (schematic diagram 6) has now latched the data for the interrupt request (when leaving state 2) and U1300 has been enabled to preset (reintialize) the flip-flop that generated the request (see Interrupt Control Bus). The vector address has also been placed on the data bus for a processor read operation (the output of U640A is set high when its inputs are unequal).

U020A was also set when state 3 was entered, causing BBSY to be asserted (if not already asserted by the current bus master). Since the machine has been instructed to wait in state 3, it will not advance to state 4 until the current bus master clears BBSY and the slave negates SSYN. When SSYN is released (cleared) on the bus, pin 1 of U150 is set low; the machine then advances to state 4. This interface is now the bus master.

When state 4 is entered, the interface asserts the INTR signal on the bus through U350C and U160B. It also negates SACK OUT by resetting U020C through U320C and U230A. INTR is asserted under the authority of

BG5 to inform the processor that an interrupt is to be performed and that the vector address is on the data lines. When the arbitrator receives the INTR signal it ceases to issue bus grants (BGs) until the processor authorizes it to do so. However, Non Processor Grants (NPGs) may be granted during this time.

The machine was instructed to wait in state 4 until the processor asserts SSYN to signify that it has accepted the interrupt vector address. Pin 15 of U150 is held high until the processor accepts the data and then goes low. The clock output then goes high to disable U120 and negate the INTR command, the processor then negates SSYN (when the machine leaves state 4). The machine then enters state 5.

When state 5 is entered another check on the interrupt status is performed. The address/data output from U130 and U140 contains machine reset enabling information (U130, pin 7) which sets a low level on pin 2 of U240A. If SACK OUT is asserted it means that U340A has been enabled by the machine setting U510C, BR5 was asserted due to an interrupt request, and that BG5 was received, setting U020C. If a bus grant was received when state 5 was entered, the machine resets to state 0 to perform another interrupt routine. The machine also removed the first vector address from the data bus by setting the output of U510C high (false) and cleared BBSY by resetting U020A. If a bus grant for another interrupt is not received, the machine advances to state 6. The State Machine will normally be at state 6 if it is not called on to do anything. When an interrupt generated by this interface has been processed it steps from state 5 to state 6 and then waits. If another interrupt request is granted (BG5 is received) it will reset to state 0 and proceed to process the interrupt. If called on to perform a DMA cycle it will step from state 6 to state 7, and so on.

State 6 has a wait instruction, one operation, and a possible machine reset. BBSY is asserted in state 10 and the 20 microsecond one-shot, U330B, cleared (turned off) in state 12 if the interface is operating in the non-hog mode (HOG MODE bit not set and pin 10 of U310C is low). If U330B is off, pin 12 of U310D is set low (enabled) and BBSY is cleared through U200B, U310D, and U000B when state 6 is entered. If the machine is operating in the hog mode (HOG MODE bit set), BBSY is not cleared.

U240A is also enabled with machine reset data in state 6 to check again for another bus grant due to an interrupt request. The machine will reset to state 0 if SACK OUT is asserted low at pin 3 of U240A.

The State Machine will step from state 6 to state 7 under one of two conditions; if talking on the IEEE 488 bus it waits for the GO and DONE bits to be set, if listening to the IEEE 488 bus it waits for the GO and RDY bits to be set. The DONE bit indicates that the Talker Data Buffer can be loaded with a data byte, while the RDY bit indicates that the Listener Data Buffer contains a valid data byte and can be read. The output of U060A remains high (a wait condition) until the GO bit is set. When the wait conditions are cleared, the machine advances to state 7.

State 7 contains no new operations (pin 7 of U120 is not connected), except for a possible machine reset function through U240A and U240B if SACK OUT is asserted and a branch instruction if the machine is operating in the hog mode. The machine senses a high level on pin 12 of U150 if BBSY has not been cleared in state 6. If pin 12 is high, the interface is still the bus master from a previous byte transfer; it will then branch to state 10. If pin 12 is low, the machine advances to state 8 in order to obtain bus mastership.

State 8 contains a wait instruction. When state 8 is entered pin 15 of U220 is asserted to set U510B, which asserts NPR low true through U160D and sets pin 7 of U040B (NPG flip-flop) high. Pin 8 of U040B and pin 4 of U250 are high before U040B receives the NPG IN signal. The machine waits until NPG is received.

If another device on the controller's bus does not have SACK asserted, the priority arbitrator will issue the Non-Processor Grant (NPG) when it senses that SACK has not been asserted for at least 75 nanoseconds. Pin 5 of U040B goes low when NPG is received through U050A, causing U040B to change state. Pin 4 of U250 then goes low and machine advances to state 9. U040B is cleared through U030B when NPG is negated.

U260C operates as an inverter or as an inhibiting gate for the NPG OUT signal. The setting for jumper H2 depends on where this interface is positioned in the NPG daisy-chain path. If this interface is not the last device in the path, the associated wire wrap section between CA1 and CB1 on the UNIBUS backplane must be cut and jumper H2 set to cause U260C to operate as an inverter (enabled position). If the interface is the last

device in the daisy-chain path, the associated wire wrap section (between CA1 and CB1) does not have to be cut, but jumper H2 must be set to cause U260C to operate as an inhibitor. On the schematic, jumper H2 is set to inhibit the NPG OUT signal.

The machine is addressed for state 9 when NPG is received. State 9 contains another wait instruction. When state 9 is entered, pin 14 of U220 is asserted, setting U020C and asserting SACK through U160A.

The output of U530A (pin 3 of U250) will be low only if both inputs are low, it will be high if either input is high. Either, or both, inputs(s) to U530A, when high, establish a wait condition.

When the arbitrator received the assertion of SACK it negated NPG and set pin 2 of U530A low. If the interface is operating in the non-hog mode, BBSY may be asserted by another device at this time; if so, pin 1 of U530A will be high. Even though the negation of NPG was received, the machine must wait until BBSY is negated to release the bus. As soon as NPG and BBSY are both negated, pin 3 of U250 goes low, allowing the machine to advance to state 10. Note that if the interface is operating in the hog mode, it detected that BBSY was already asserted by this interface in state 7. It then skipped states 8 and 9.

When the machine is addressed for state 10, pin 15 of U140 is latched high to set the indicated logic levels on the ADDR lines (see Bus Address and Control). If the machine is talking, the DMA IN line (pin 2 of U610A) will be high, causing the output of U640A to be set low for data byte transfers to the IEEE 488 bus. When listening, U610A is enabled to set unequal logic levels at the input to U640A, setting the output of U640A high for data transfers to the controller's internal bus (memory).

State 10 is entered when the clock output goes low. U220 is enabled (pin 13 is asserted), U020A is set to assert BBSY, the 20 microsecond one-shot (U330B) is turned on (if it is off), and NPR is negated by resetting U510B through U110B. Pin 7 of U040B is also set low to keep the NPG flip-flop in a cleared state for NPR-NPG arbitration sequences between other devices on the controller's internal bus. The machine has no wait or branch condition for state 10. State 11 is addressed when the clock output goes high.

State 11 contains the last wait instruction during a machine cycle. When state 11 is entered, pin 12 of U220, pin 13 of U530D, pin 10 of U320C, and pin 6 of U020B are all asserted low true. SACK is negated by resetting U020C through U320C and U230A, MSYN is asserting by setting U020B, and U530D is enabled at pin 13.

The 20 microsecond one-shot, U330B, was set in state 10. If the interface is not programmed for hog mode (HOG MODE bit clear), U330B is deliberately cleared in state 12. At state 6, BBSY will be released because U330B was not on. If the interface is bus master with the HOG MODE bit set, BBSY will not be released in state 6 unless the 20 microsecond one-shot times out. U330B will time out if the interface does not receive the assertion of SSYN in state 11 or if a device on the IEEE 488 bus does not respond within 20 microseconds. An error flag is not set when a device on the IEEE 488 bus does not respond within 20 microseconds. If the GO bit gets cleared because the DMA DONE bit gets set in state 12 or gets cleared due to an error flag being set, the 20 microsecond one-shot will time out at state 6 and release BBSY.

The output of U240C (pin 1 of U220) is high (a wait condition) only when both inputs are low. The addressed slave on the controller's bus must respond and set pin 8 of U240C high before U330B times out and sets pin 9 high. If U330B times out with U530D enabled at pin 13 and the slave device has not yet asserted SSYN, the NO MEM line is asserted low to set the DMA ERR bit in the Interface Status Register. The machine will then complete its current cycle, and branch to state 6 with the GO bit cleared. State 12 is addressed when the clock output goes high.

When U220 is enabled for state 12, the following inputs are asserted low true: pin 12 of U200D, pin 1 of U200A, pin 9 of U310C, pin 4 of U310B, and pin 4 of U340B (schematic diagram 5).

MSYN is cleared by resetting U020B through U200A, the RDY bit (U1010B, schematic diagram 3) is cleared (if listening) by asserting the INPUT line through U200D, the Talker Data Buffer is loaded (if talking), U310B is enabled, and U330B is deliberately cleared if the interface is operating in the non-hog mode (HOG MODE bit not set).

With U310B enabled, the DMA DONE bit will be set if the NO EOI DMA bit is not set and the END of message was received (END OCC bit set) or the Byte Counter Register overflowed. These conditions are sensed through

U200C and U240D. Pin 2 of U200C will be high if the NO EOI DMA bit is not set. If both inputs to U240D are low, the machine does not set the DMA DONE bit. The END interrupt may be ignored by the program.

If the machine was talking when state 12 was entered, the interface performs the Source Handshake; if listening, the interface performs the Acceptor Handshake. These functions are discussed under separate headings in this section.

When the internal state clock goes high to latch the address for state 13, pin 15 of U140 is latched at a low level. The ADDR lines then switch logic levels, removing the contents of the BAR from the controller's internal bus and setting the output of U640A to a low level (also removing the data).

State 13 has only one operation with no branch or wait instruction. Pin 12 of U640D goes low when state 13 is entered and the Bus Address Register is clocked through U440C. The BAR is incremented (or decremented) by one in state 13. The machine then advances to state 14.

State 14 contains a branch instruction. When state 14 is entered, the Byte Counter Register is incremented by one through U230F (if it has not overflowed). If the PACK MODE bit was set by the program, pin 13 of U250 will be high; if so, the machine branches out of state 14 and enters state 6. If operating in the unpacked mode (PACK MODE bit clear) it is addressed for state 15.

In state 15 the Bus Address Register is incremented (or decremented) a second time if the interface is operating in the unpacked mode. After the BAR is clocked through U640D and U440C, the machine branches out of state 15 and enters state 6, ready for another byte transfer or to make an interrupt request.

Source Handshake

Refer to schematic diagrams 3 and 2 for this discussion. Unless otherwise stated, all references to a bus in this section refer to the IEEE 488 bus.

The bus transceivers are arranged in three sections: U1510 for the handshake bus (plus EOI), U1520 for the management bus, U1530 and U1410 for the data bus. U1530 and U1410 are enabled when the DONE bit is not asserted (cleared). U1510 and U1520 are always enabled so that the individual handshake and management lines can be controlled internally. Signal inversion occurs in either direction in the transceivers. Each transceiver provides four open-collector drivers (with internal termination resistors) and four receivers. Each line on the bus is low-true when asserted.

The DONE bit is normally asserted. When the interface becomes a talker it can send the first data byte immediately because the DONE bit is set (pin 15 of U810A is high). When the Talker Data Buffer is loaded, the DONE bit is cleared and the data transceivers are enabled, causing data to be asserted on the bus. A short time later (greater than 500 nanoseconds), the listeners assert NRFD and release NDAC when the data byte is accepted. When NDAC is unasserted by all addressed listeners, pins 1 and 14 of U810A go low and pin 15 of U810A goes high. This operation causes the interface to release DAV and set the DONE bit to indicate that another data byte can be sent. The DONE bit, when set, causes an interrupt request if the DONE INTR ENBL bit (pin 6 of U1130) has been set. This bit should not be set during a DMA transfer (see DMA State Machine). When a data byte is to be transmitted, the low-order byte or high-order byte latched in U1420 and U1540 is designated by the output level from U630C. Pin 10 of U1420 and U1540 is high for the low-order byte and low for the high-order byte.

If there is no listener on the bus or if an addressed listener is in an invalid state, the NRFD and NDAC bus lines will be high (both low at the input to U700A, schematic 2). If an attempt is then made to place data on the bus (load the Talker Data Buffer), pin 11 of U500D goes low and sets the WRITE ERR flip-flop, U720A. This causes an interrupt request if the DMA INTR ENBL bit (pin 15 of U1210) has been set. This interrupt can occur for DMA or non-DMA transfers, depending on the program.

If the interface attempts to assert DAV while another controller on the bus has the ATN line already asserted, or asserts ATN during DAV, TIME ERR flip-flop U520B becomes set, causing an interrupt request (if the DMA INTR ENBL bit has been set). This error is sensed through U500B.

If the ERROR bit becomes set for any of the high level input conditions for U730B and the DMA State Machine has not completed its total byte transfer, the DMA transfer will be interrupted (pin 3 of U1120A goes high) and the GO bit flip-flop, U710A, will be cleared through U530C, U730A, U600C and U610C.

Assuming that the Talker Data Buffer is being loaded and that no error conditions have been (or will be) sensed, pin 9 of U410A (schematic 3) goes low. U410A, a one-shot, is triggered in either mode (DMA or non-DMA) and delays its output by a time interval greater than 500 nanoseconds before clocking U100B, whose output goes low at pin 13 of U610D. If all of the assigned listeners are ready for data (RFD), the NRFD bus line (high) has already set pin 12 of U610D low; U510A then becomes set and DAV is asserted through U1500A and U1410.

If another controller asserts ATN through U600B and U810A (pin 2) while this interface has DAV asserted, U510A is reset to clear DAV. When all assigned listeners have accepted the data byte, pin 1 of U810A goes low, setting the DONE bit and clearing DAV by resetting U510A. The DONE bit is cleared through U1400 and C800 by a negative pulse after the TDB is loaded.

The interface sends data (talks) with ATN asserted or unasserted and can send EOI as a ninth data bit. EOI may also be transmitted with the last data byte under DMA control; this feature can be disabled by setting the NO EOI DMA bit (U820B). EOI can also be sent asynchronously by setting the EOI OUT bit (U1310, pin 5) in the Bus Control Register. This is normally done with the ATN OUT bit (U710B) also set in order to perform a parallel poll.

EOI appears high at pin 4 of U1510 to be asserted on the bus. U1500B, pin 5, receives asynchronous EOI and pin 4 of U1500B receives the EOI signal that is gated by U700C so that it can be asserted when data is asserted (DONE bit not set).

U100A is loaded whenever the Talker Data Buffer is loaded so that EOI can be an extension of the transmitted byte. For the non-DMA mode, U320B and U630C are disabled at pin 3 and 10 (BBSY not asserted) and U010B is enabled at pin 5. Pin 8 of U010C and pin 2 of U100A will not follow whatever signal appears at pin 4 of U010B. This is data bit D15 of the data/address bus. When the Talker Data Buffer is loaded, the value of

D15 gets loaded into U100A.

During a DMA operation, U010B is disabled at pin 5, U320B and U630C are enabled at pins 3 and 10 (BBSY asserted). When the DMA State Machine loads the Talker Data Buffer in state 12, U100A is set when the CTR overflows if ATN is not asserted on the bus and if the NO EOI DMA bit is cleared at pin 1 of U310A. If the NO EOI DMA bit has been set, EOI will not be asserted on the bus.

Acceptor Handshake

If the interface is programmed to listen, the RDY bit flip-flop, U1010B, becomes set when the interface senses that DAV has been asserted (through U700D) by another device (a talker) on the bus. The RDY bit remains asserted until it is cleared by reading the Listener Data Buffer. After the Listener Data Buffer has been read, this interface releases NDAC to notify the talker to remove the data byte from the bus. When the talker releases DAV, NDAC is reset low true and NRFD is cleared (set false) on the bus to signal ready for data. The RDY bit, when set, causes an interrupt request if the RDY INTR ENBL bit (U1130, pin 5) has been set. The RDY INTR ENBL bit should not be set during a DMA transfer (see DMA State Machine).

The Acceptor Handshake function for the DMA mode is enabled when the GO bit and the DMA IN/OUT bit are set high at pins 1 and 2 of U430A. For the non-DMA mode, the Acceptor Handshake function is enabled by the low output level at pin 8 of U420C. For either mode, the output of U030A will be high when the interface is accepting data from the bus (listening).

Before the talker asserts DAV, the output of U1500D is low and the output of U1500C is high. The RDY bit has been cleared. The LST output from U030A (high) has enabled U000A, U000D, and U700D. DAV has not yet been received (asserted); therefore, the clock input to U1010B (RDY) is at a low level and U820A has been preset (pin 5 high). Also, before DAV is asserted, U020D has been set and pin 10 of U1500C is high (pin 9 is low).

As soon as the talker asserts DAV, it is gated through U700D to set the RDY bit (U1010B) and gated through U010A (with inversion) to clear the ready for data (RFD) message (NRFD is asserted low true on the bus).

When the Listener Data Buffer has been read, pin 3 of U820A goes high to clock U820A. Pin 5 of U820A then goes low to clear the RDY bit. This also clears U020D and the output of U1500C goes low to release (unassert) NDAC on the bus. The data byte has now been read and accepted by the interface.

When the talker releases DAV on the bus, NRFD OUT (pin 11 of U1500D) is reset to a low level and U020D is set again (pin 13 goes high again), ensuring that NDAC is asserted low true on the bus.

During a DMA listen operation, the same conditions are established for NRFD and NDAC before the talker asserts DAV. Before the DMA State Machine reaches state 12, the talker has already asserted data and DAV, setting the RDY bit and clearing the RFD message (NRFD has been asserted). All that remains is to read the data, clear the RDY bit and assert NDAC when the talker releases DAV. The DMA State Machine reads the Listener Data Buffer during states 10 and 11 before entering state 12.

When state 12 is entered, pin 1 of U820A goes low and clears the RDY bit. This presets U820A and sets pin 14 of U020D high. When DAV is released, pin 15 of U020D goes low to set pin 13 to a high level, causing NDAC to be asserted low-true on the bus. The interface is ready to accept another data byte.

The NRFD and NDAC bus lines may also be asserted or cleared asynchronously by setting or clearing pins 15 and 12 of U1310 at the appropriate times during the handshake cycle. These lines correspond to data/address bits D7 and D6, respectively, in the Bus Control Register.

Interface Registers

General. Refer to schematic diagrams 2 and 3 for this discussion. The Talker Data Buffer and Listener Data Buffer have been discussed under separate headings; Source Handshake and Acceptor Handshake. Also, some of the bits in the Interface Status Register, Bus Control Register, and Interrupt Control Register have been discussed for the DMA State Machine, Source Handshake, and Acceptor Handshake. The interface register layouts as portrayed on schematic diagrams 2 and 3 are identified at the top of the associated schematic with the interconnect wiring fairly straightforward. Each bit is easily identified with the proper signal

name and the logic levels required to activate the desired functions are as indicated. Addressing and loading each register has been previously discussed under Address Decoding.

A complete understanding of each bit in each register can be obtained by reading Section 3 of this manual and relating each bit to the components associated with that bit on the block diagram. To understand how the registers may be programmed, read Section 4 of this manual while studying schematic diagrams 2 and 3. Keep in mind that the block diagram does not exactly match the schematics; the schematic wiring has precedence over the block diagram.

Attention Received. When ATN is asserted on the bus by another controller (not this interface) pin 10 of U1520 goes high, while pin 11 of U1520 and pin 12 of U300B remain low. ATN going high at pin 11 of U300B clocks a low level to pin 9 and a high level to pin 8. Pin 3 of U800B is also set high. If the LIST W ATN bit (U830, pin 10) has been set, the output of U430D goes low and presets the LIST W/O ATN flip-flop, U620A. Since the ATN REC signal at pin 5 of U610 went high when ATN was asserted, the output of U610 goes high and the Acceptor Handshake is enabled by the LIST W ATN bit through U430D and U420C. The ATN REC message also went low at the output of U1400E to disable U320B at pin 4 (previously discussed). The ATN REC message also set pin 10 of U1120 high and this causes an interrupt requst if the ATN INTR ENBL bit (U1130, pin 12) has been set.

When the controller-in-charge releases ATN for device-dependent messages, pin 4 of U800B goes low and presets pin 9 of U300B high and pin 8 low. Pin 5 of U610 goes low and pin 13 of U430D goes high. If the LIST W ATN bit has been cleared (not set), U430D is disabled and the Acceptor Handshake is enabled immediately by the LIST W/O ATN bit through U610 and U420C.

Attention Sent. This interface, when so programmed, can take control of the bus asynchronously or synchronously. When control is taken asynchronously, the program sets the ATN OUT flip-flop, U710B (in the Bus Control Register), without setting the TCS bit. If control is taken synchronously, the TCS flip-flop, U620B, is set by the program.

The TCS flip-flop, if set, places a high level on pin 2 of U300A. When DAV is asserted for device-dependent data from the current talker

(ATN unasserted), pin 10 of U500C is set low and pin 3 of U300A goes high; clocking the high level on pin 2 through U1500D to assert the NRFD message (Not Ready for Data). This inhibits the next source handshake from the talker. When the talker releases DAV, pin 10 of U500C goes low and presets the ATN OUT flip-flop, U710B. As soon as ATN OUT becomes set, U300A and U620B are cleared through U600A. This operation asserts the ATN bus line and resets the NRFD line to a high level (ready for data). The next data byte (an address or command) can then be sent by this interface.

Interface Clear. As system controller, the interface may assert the IFC bus line through U1510 when the program loads data/address bit D0 in the Bus Control Register. If D0 is a 1, U410B is triggered when the Bus Control Reigster is loaded and its output goes high for about 150 microseconds, then automatically resets to a low level, A RESET instruction on pin 4 clears this bit if it is locked high. Refer to Table 3-3, Section 3 for information concerning the bits in each register that are cleared when IFC is asserted or a RESET instruction is executed.

REPLACEABLE ELECTRICAL PARTS

PARTS ORDERING INFORMATION

Replacement parts are available from or through your local Tektronix, Inc. Field Office or representative.

Changes to Tektronix instruments are sometimes made to accommodate improved components as they become available, and to give you the benefit of the latest circuit improvements developed in our engineering department. It is therefore important, when ordering parts, to include the following information in your order: Part number, instrument type or number, serial number, and modification number if applicable.

If a part you have ordered has been replaced with a new or improved part, your local Tektronix, Inc. Field Office or representative will contact you concerning any change in part number.

Change information, if any, is located at the rear of this manual.

SPECIAL NOTES AND SYMBOLS

X000	Part first added at this serial number
00X	Part removed after this serial number

ITEM NAME

In the Parts List, an Item Name is separated from the description by a colon (:). Because of space limitations, an Item Name may sometimes appear as incomplete. For further Item Name identification, the U.S. Federal Cataloging Handbook H6-1 can be utilized where possible.

ABBREVIATIONS

ACTR	ACTUATOR	PLSTC	PLASTIC
ASSY	ASSEMBLY	QTZ	QUARTZ
CAP	CAPACITOR	RECP	RECEPTACLE
CER	CERAMIC	RES	RESISTOR
CKT	CIRCUIT	RF	RADIO FREQUENCY
COMP	COMPOSITION	SEL	SELECTED
CONN	CONNECTOR	SEMICOND	SEMICONDUCTOR
ELCTLT	ELECTROLYTIC	SENS	SENSITIVE
ELEC	ELECTRICAL	VAR	VARIABLE
INCAND	INCANDESCENT	WW	WIREWOUND
LED	LIGHT EMITTING DIODE	XFMR	TRANSFORMER
NONWIR	NON WIREWOUND	XTAL	CRYSTAL

CROSS INDEX—MFR. CODE NUMBER TO MANUFACTURER

Mfr. Code	Manufacturer	Address	City, State, Zip
01121	ALLEN-BRADLEY COMPANY	1201 2ND STREET SOUTH	MILWAUKEE, WI 53204
01295	TEXAS INSTRUMENTS, INC., SEMICONDUCTOR GROUP	P O BOX 5012, 13500 N CENTRAL EXPRESSWAY	DALLAS, TX 75222
04222	AVX CERAMICS, DIVISION OF AVX CORP.	P O BOX 867, 19TH AVE. SOUTH	MURTLE BEACH, SC 29577
04713	MOTOROLA, INC., SEMICONDUCTOR PROD. DIV.	5005 E MCDOWELL RD,PO BOX 20923	PHOENIX, AZ 85036
07263	FAIRCHILD SEMICONDUCTOR, A DIV. OF FAIRCHILD CAMERA AND INSTRUMENT CORP.	464 ELLIS STREET	MOUNTAIN VIEW, CA 94042
07910	TELEDYNE SEMICONDUCTOR	12515 CHADRON AVE.	HAWTHORNE, CA 90250
27014	NATIONAL SEMICONDUCTOR CORP.	2900 SEMICONDUCTOR DR.	SANTA CLARA, CA 95051
32997	BOURNS, INC., TRIMPOT PRODUCTS DIV.	1200 COLUMBIA AVE.	RIVERSIDE, CA 92507
56289	SPRAGUE ELECTRIC CO.		NORTH ADAMS, MA 01247
72982	ERIE TECHNOLOGICAL PRODUCTS, INC.	644 W. 12TH ST.	ERIE, PA 16512
80009	TEKTRONIX, INC.	P O BOX 500	BEAVERTON, OR 97077
91637	DALE ELECTRONICS, INC.	P. O. BOX 609	COLUMBUS, NE 68601

Ckt No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
A1	670-5387-00			CKT BOARD ASSY:11/IEEE 488	80009	670-5387-00
C012	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C022	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C062	281-0623-00			CAP.,FXD,CER DI:650PF,5%,500V	04222	7001-1362
C064	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C102	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C122	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C152	281-0623-00			CAP.,FXD,CER DI:650PF,5%,500V	04222	7001-1362
C162	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C202	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C212	283-0175-00			CAP.,FXD,CER DI:10PF,5%,200V	72982	8101-200C0G0100J
C232	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C302	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C330	283-0175-00			CAP.,FXD,CER DI:10PF,5%,200V	72982	8101-200C0G0100J
C334	283-0028-00			CAP.,FXD,CER DI:0.0022UF,20%,50V	56289	19C606
C332	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C362	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C402	283-0177-00			CAP.,FXD,CER DI:1UF,+80-20%,25V	72982	8131N039 E 105Z
C412	283-0111-00			CAP.,FXD,CER DI:0.1UF,20%,50V	72982	8121-N088Z5U104M
C414	283-0330-00			CAP.,FXD,CER DI:100PF,5%,50V	72982	8111N068C0G0101J
C422	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C462	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C502	283-0010-00			CAP.,FXD,CER DI:0.05UF.+100-20%,50V	56289	273C20
C522	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C562	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C602	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C622	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C662	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C702	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C724	283-0330-00			CAP.,FXD,CER DI:100PF,5%,50V	72982	8111N068C0G0101J
C742	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C762	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C800	281-0523-00			CAP.,FXD,CER DI:100PF,+/-20PF,500V	72982	301-000U2M0101M
C802	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C812	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C862	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C902	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C922	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C962	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1032	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1062	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1102	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1122	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1162	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1202	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1204	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1242	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1262	290-0106-00			CAP.,FXD,ELCTLT:10UF,+75-10%,15V	56289	30D106G015BA9
C1342	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1362	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1404	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20

Ckt No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
C1432	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1462	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1502	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1542	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
C1562	283-0010-00			CAP.,FXD,CER DI:0.05UF,+100-20%,50V	56289	273C20
CR612	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	07910	1N4152
CR614	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	07910	1N4152
CR712	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	07910	1N4152
CR714	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	07910	1N4152
CR822	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	07910	1N4152
CR824	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	07910	1N4152
CR922	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	07910	1N4152
CR924	152-0141-02			SEMICOND DEVICE:SILICON,30V,150MA	07910	1N4152
Q52	151-0190-00			TRANSISTOR:SILICON,NPN	80009	151-0190-00
Q54	151-0190-00			TRANSISTOR:SILICON,NPN	80009	151-0190-00
R032	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
R034	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
R042	315-0181-00			RES.,FXD,CMPSN:180 OHM,5%,0.25W	01121	CB1815
R044	315-0181-00			RES.,FXD,CMPSN:180 OHM,5%,0.25W	01121	CB1815
R062	315-0391-00			RES.,FXD,CMPSN:390 OHM,5%,0.25W	01121	CB3915
R064	315-0391-00			RES.,FXD,CMPSN:390 OHM,5%,0.25W	01121	CB3915
R102	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
R152	315-0181-00			RES.,FXD,CMPSN:180 OHM,5%,0.25W	01121	CB1815
R162	315-0181-00			RES.,FXD,CMPSN:180 OHM,5%,0.25W	01121	CB1815
R164	315-0181-00			RES.,FXD,CMPSN:180 OHM,5%,0.25W	01121	CB1815
R202	315-0103-00			RES.,FXD,CMPSN:10K OHM,5%,0.25W	01121	CB1035
R212	315-0392-00			RES.,FXD,CMPSN:3.9K OHM,5%,0.25W	01121	CB3925
R262	315-0181-00			RES.,FXD,CMPSN:180 OHM,5%,0.25W	01121	CB1815
R264	315-0222-00			RES.,FXD,CMPSN:2.2K OHM,5%,0.25W	01121	CB2225
R322	315-0472-00			RES.,FXD,CMPSN:4.7K OHM,5%,0.25W	01121	CB4725
R402	311-1283-00			RES.,VAR,NONWIR:10K OHM,10%,0.50W	32997	3329W-L58-103
R412	315-0622-00			RES.,FXD,CMPSN:6.2K OHM,5%,0.25W	01121	CB6225
R414	315-0183-00			RES.,FXD,CMPSN:18K OHM,5%,0.25W	01121	CB1835
R432	315-0393-00			RES.,FXD,CMPSN:39K OHM,5%,0.25W	01121	CB3935
R552	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
R612	321-0193-00			RES.,FXD,FILM:1K OHM,1%,0.125W	91637	MFF1816G10000F
R722	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
R724	315-0222-00			RES.,FXD,CMPSN:2.2K OHM,5%,0.25W	01121	CB2225
R752	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
R962	315-0222-00			RES.,FXD,CMPSN:2.2K OHM,5%,0.25W	01121	CB2225
R1002	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
R1332	315-0102-00			RES.,FXD,CMPSN:1K OHM,5%,0.25W	01121	CB1025
U000	156-0480-00			MICROCIRCUIT,DI:QUAD 2-INPUT AND GATE	80009	156-0480-00
U010	156-0382-00			MICROCIRCUIT,DI:QUAD 2-INPUT NAND GATE	01295	SN74LS00N
U020	156-0804-00			MICROCIRCUIT,DI:QUADRUPLE S-R LATCH	07263	74LS279PC
U030	156-0382-00			MICROCIRCUIT,DI:QUAD 2-INPUT NAND GATE	01295	SN74LS00N
U040	156-0387-00			MICROCIRCUIT,DI:DUAL J-K NEG EDGE TRIG	01295	SN74LS73N
U050	156-0455-00			MICROCIRCUIT,DI:HEX.BUS VEC	27014	DM8837N
U060	156-0382-00			MICROCIRCUIT,DI:QUAD 2-INPUT NAND GATE	01295	SN74LS00N
U100	156-0388-00			MICROCIRCUIT,DI:DUAL D-TYPE FLIP-FLOP	01295	SN74LS74N
U110	156-0480-00			MICROCIRCUIT,DI:QUAD 2-INPUT AND GATE	80009	156-0480-00
U120	156-0469-00			MICROCIRCUIT,DI:3-LINE TO 8-LINE DECODER	01295	SN74LS138N

Ckt No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
U130	156-0785-16			MICROCIRCUIT,DI:256 BIT PROM,PROGRAMMED	80009	156-0785-16
U140	156-0222-00			MICROCIRCUIT,DI:HEX.LATCH	01295	SN74174N
U150	156-0470-00			MICROCIRCUIT,DI:8-INPUT DATA SELECTOR	27014	DM74LS251N
U160	156-0145-00			MICROCIRCUIT,DI:QUAD 2-INPUT POS NAND BFR	01295	SN7438N
U200	156-0480-00			MICROCIRCUIT,DI:QUAD 2-INPUT AND GATE	80009	156-0480-00
U210	156-0172-00			MICROCIRCUIT,DI:DUAL RETRIG MONOSTABLE MV	80009	156-0172-00
U220	156-0469-00			MICROCIRCUIT,DI:3-LINE TO 8-LINE DECODER	01295	SN74LS138N
U230	156-0385-00			MICROCIRCUIT,DI:HEX.INVERTER	01295	SN74LS04N
U240	156-0383-00			MICROCIRCUIT,DI:QUAD 2-INPUT NOR GATE	01295	SN74LS02N
U250	156-0470-00			MICROCIRCUIT,DI:8-INPUT DATA SELECTOR	27014	DM74LS251N
U260	156-0145-00			MICROCIRCUIT,DI:QUAD 2-INPUT POS NAND BFR	01295	SN7438N
U300	156-0388-00			MICROCIRCUIT,DI:DUAL D-TYPE FLIP-FLOP	01295	SN74LS74N
U310	156-0171-00			MICROCIRCUIT,DI:QUAD 2-INPUT OR GATE	01295	SN7432N
U320	156-0386-00			MICROCIRCUIT,DI:TRIPLE 3-INPUT NAND GATE	01295	SN74LS10N
U330	156-0405-00			MICROCIRCUIT,DI:DUAL RETRIG MONOSTABLE MV	07263	9602PC
U340	156-0480-00			MICROCIRCUIT,DI:QUAD 2-INPUT AND GATE	80009	156-0480-00
U350	156-0385-00			MICROCIRCUIT,DI:HEX.INVERTER	01295	SN74LS04N
U360	156-0653-00			MICROCIRCUIT,DI:QUAD UNIFIED BUS XCVR	27014	DS8838N
U400	156-0385-00			MICROCIRCUIT,DI:HEX.INVERTER	01295	SN74LS04N
U410	156-0172-00			MICROCIRCUIT,DI:DUAL RETRIG MONOSTABLE MV	80009	156-0172-00
U420	156-0480-00			MICROCIRCUIT,DI:QUAD 2-INPUT AND GATE	80009	156-0480-00
U430	156-0382-00			MICROCIRCUIT,DI:QUAD 2-INPUT NAND GATE	01295	SN74LS00N
U440	156-0382-00			MICROCIRCUIT,DI:QUAD 2-INPUT NAND GATE	01295	SN74LS00N
U450	156-0539-00			MICROCIRCUIT,DI:BUS COMPTR	27014	DM8136N
U460	156-0653-00			MICROCIRCUIT,DI:QUAD UNIFIED BUS XCVR	27014	DS8838N
U500	156-0382-00			MICROCIRCUIT,DI:QUAD 2-INPUT NAND GATE	01295	SN74LS00N
U510	156-0804-00			MICROCIRCUIT,DI:QUADRUPLE S-R LATCH	07263	74LS279PC
U520	156-0731-00			MICROCIRCUIT,DI:DUAL J-K FF W/PRESET & CLR	80009	156-0731-00
U530	156-0171-00			MICROCIRCUIT,DI:QUAD 2-INPUT OR GATE	01295	SN7432N
U540	156-0383-00			MICROCIRCUIT,DI:QUAD 2-INPUT NOR GATE	01295	SN74LS02N
U550	156-0539-00			MICROCIRCUIT,DI:BUS COMPTR	27014	DM8136N
U560	156-0653-00			MICROCIRCUIT,DI:QUAD UNIFIED BUS XCVR	27014	DS8838N
U600	156-0480-00			MICROCIRCUIT,DI:QUAD 2-INPUT AND GATE	80009	156-0480-00
U610	156-0479-00			MICROCIRCUIT,DI:QUAD 2-INPUT OR GATE	27014	DM74LS32N
U620	156-0388-00			MICROCIRCUIT,DI:DUAL D-TYPE FLIP-FLOP	01295	SN74LS74N
U630	156-0480-00			MICROCIRCUIT,DI:QUAD 2-INPUT AND GATE	80009	156-0480-00
U640	156-0381-00			MICROCIRCUIT,DI:QUAD 2-INPUT EXCL OR GATES	01295	SN74LS86N
U650	156-0653-00			MICROCIRCUIT,DI:QUAD UNIFIED BUS XCVR	27014	DS8838N
U660	156-0653-00			MICROCIRCUIT,DI:QUAD UNIFIED BUS XCVR	27014	DS8838N
U700	156-0383-00			MICROCIRCUIT,DI:QUAD 2-INPUT NOR GATE	01295	SN74LS02N
U710	156-0388-00			MICROCIRCUIT,DI:DUAL D-TYPE FLIP-FLOP	01295	SN74LS74N
U720	156-0804-00			MICROCIRCUIT,DI:QUADRUPLE S-R LATCH	07263	74LS279PC
U730	156-0165-00			MICROCIRCUIT,DI:DUAL 4-INPUT POS NOR GATE	01295	SN7425N
U740	156-0464-00			MICROCIRCUIT,DI:DUAL 4-INPUT NAND GATE	01295	SN74LS20N
U750	156-0469-00			MICROCIRCUIT,DI:3-LINE TO 8-LINE DECODER	01295	SN74LS138N
U760	156-0653-00			MICROCIRCUIT,DI:QUAD UNIFIED BUS XCVR	27014	DS8838N
U800	156-0479-00			MICROCIRCUIT,DI:QUAD 2-INPUT OR GATE	27014	DM74LS32N
U810	156-0731-00			MICROCIRCUIT,DI:DUAL J-K FF W/PRESET & CLR	80009	156-0731-00
U820	156-0388-00			MICROCIRCUIT,DI:DUAL D-TYPE FLIP-FLOP	01295	SN74LS74N
U830	156-0392-00			MICROCIRCUIT,DI:QUAD LATCH	01295	SN74LS175N
U840	156-0422-00			MICROCIRCUIT,DI:UP/DOWN SYNC BINARY COUNTER	80009	156-0422-00
U850	156-0075-00			MICROCIRCUIT,DI:SGL 8-BIT DATA SEL MUX	80009	156-0075-00
U860	156-0422-00			MICROCIRCUIT,DI:UP/DOWN SYNC BINARY COUNTER	80009	156-0422-00

Ckt No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Name & Description	Mfr Code	Mfr Part Number
U900	156-0382-00			MICROCIRCUIT,DI:QUAD 2-INPUT NAND GATE	01295	SN74LS00N
U910	156-0388-00			MICROCIRCUIT,DI:DUAL D-TYPE FLIP-FLOP	01295	SN74LS74N
U920	156-0388-00			MICROCIRCUIT,DI:DUAL D-TYPE FLIP-FLOP	01295	SN74LS74N
U930	156-0388-00			MICROCIRCUIT,DI:DUAL D-TYPE FLIP-FLOP	01295	SN74LS74N
U940	156-0422-00			MICROCIRCUIT,DI:UP/DOWN SYNC BINARY COUNTER	80009	156-0422-00
U950	156-0075-00			MICROCIRCUIT,DI:SGL 8-BIT DATA SEL MUX	80009	156-0075-00
U960	156-0422-00			MICROCIRCUIT,DI:UP/DOWN SYNC BINARY COUNTER	80009	156-0422-00
U1000	156-0382-00			MICROCIRCUIT,DI:QUAD 2-INPUT NAND GATE	01295	SN74LS00N
U1010	156-0388-00			MICROCIRCUIT,DI:DUAL D-TYPE FLIP-FLOP	01295	SN74LS74N
U1020	156-0388-00			MICROCIRCUIT,DI:DUAL D-TYPE FLIP-FLOP	01295	SN74LS74N
U1030	156-0480-00			MICROCIRCUIT,DI:QUAD 2-INPUT AND GATE	80009	156-0480-00
U1040	156-0422-00			MICROCIRCUIT,DI:UP/DOWN SYNC BINARY COUNTER	80009	156-0422-00
U1050	156-0075-00			MICROCIRCUIT,DI:SGL 8-BIT DATA SEL MUX	80009	156-0075-00
U1060	156-0422-00			MICROCIRCUIT,DI:UP/DOWN SYNC BINARY COUNTER	80009	156-0422-00
U1060	156-0422-00			MICROCIRCUIT,DI:UP/DOWN SYNC BINARY COUNTER	80009	156-0422-00
U1100	156-0865-00			MICROCIRCUIT,DI:OCTAL D TYPE FF W/CLEAR	80009	156-0865-00
U1110	156-0035-00			MICROCIRCUIT,DI:SGL 8-INPUT POS NAND GATE	80009	156-0035-00
U1120	156-0480-00			MICROCIRCUIT,DI:QUAD 2-INPUT AND GATE	80009	156-0480-00
U1130	156-0865-00			MICROCIRCUIT,DI:OCTAL D TYPE FF W/CLEAR	80009	156-0865-00
U1140	156-0422-00			MICROCIRCUIT,DI:UP/DOWN SYNC BINARY COUNTER	80009	156-0422-00
U1150	156-0075-00			MICROCIRCUIT,DI:SGL 8-BIT DATA SEL MUX	80009	156-0075-00
U1160	156-0422-00			MICROCIRCUIT,DI:UP/DOWN SYNC BINARY COUNTER	80009	156-0422-00
U1200	156-0219-00			MICROCIRCUIT,DI:8-INPUT PRIORITY DCDR	07263	9318DC
U1210	156-0391-00			MICROCIRCUIT,DI:HEX LATCH WITH CLEAR	01295	SN74LS174N
U1220	156-0470-00			MICROCIRCUIT,DI:8-INPUT DATA SELECTOR	27014	DM74LS251N
U1230	156-0470-00			MICROCIRCUIT,DI:8-INPUT DATA SELECTOR	27014	DM74LS251N
U1240	156-0470-00			MICROCIRCUIT,DI:8-INPUT DATA SELECTOR	27014	DM74LS251N
U1250	156-0470-00			MICROCIRCUIT,DI:8-INPUT DATA SELECTOR	27014	DM74LS251N
U1260	156-0653-00			MICROCIRCUIT,DI:QUAD UNIFIED BUS XCVR	27014	DS8838N
U1300	156-0469-00			MICROCIRCUIT,DI:3-LINE TO 8-LINE DECODER	01295	SN74LS138N
U1310	156-0391-00			MICROCIRCUIT,DI:HEX LATCH WITH CLEAR	01295	SN74LS174N
U1320	156-0470-00			MICROCIRCUIT,DI:8-INPUT DATA SELECTOR	27014	DM74LS251N
U1330	156-0470-00			MICROCIRCUIT,DI:8-INPUT DATA SELECTOR	27014	DM74LS251N
U1340	156-0470-00			MICROCIRCUIT,DI:8-INPUT DATA SELECTOR	27014	DM74LS251N
U1350	156-0470-00			MICROCIRCUIT,DI:8-INPUT DATA SELECTOR	27014	DM74LS251N
U1360	156-0653-00			MICROCIRCUIT,DI:QUAD UNIFIED BUS XCVR	27014	DS8838N
U1400	156-0058-00			MICROCIRCUIT,DI:HEX.INVERTER	01295	SN7404N
U1410	156-0600-00			MICROCIRCUIT,DI:QUAD BUS XCVR	04713	MC3441P
U1420	156-0373-00			MICROCIRCUIT,DI:QUAD 2-INPUT,MUX W/STORE	80009	156-0373-00
U1430	156-0470-00			MICROCIRCUIT,DI:8-INPUT DATA SELECTOR	27014	DM74LS251N
U1440	156-0530-00			MICROCIRCUIT,DI:SEL/MULTIPLEXE,16 PIN DIP	80009	156-0530-00
U1450	156-0530-00			MICROCIRCUIT,DI:SEL/MULTIPLEXE,16 PIN DIP	80009	156-0530-00
U1460	156-0653-00			MICROCIRCUIT,DI:QUAD UNIFIED BUS XCVR	27014	DS8838N
U1500	156-0479-00			MICROCIRCUIT,DI:QUAD 2-INPUT OR GATE	27014	DM74LS32N
U1510	156-0600-00			MICROCIRCUIT,DI:QUAD BUS XCVR	04713	MC3441P
U1520	156-0600-00			MICROCIRCUIT,DI:QUAD BUS XCVR	04713	MC3441P
U1530	156-0600-00			MICROCIRCUIT,DI:QUAD BUS XCVR	04713	MC3441P
U1540	156-0373-00			MICROCIRCUIT,DI:QUAD 2-INPUT,MUX W/STORE	80009	156-0373-00
U1550	156-0530-00			MICROCIRCUIT,DI:SEL/MULTIPLEXE,16 PIN DIP	80009	156-0530-00
U1560	156-0653-00			MICROCIRCUIT,DI:QUAD UNIFIED BUS XCVR	27014	DS8838N

REPLACEABLE MECHANICAL PARTS

PARTS ORDERING INFORMATION

Replacement parts are available from or through your local Tektronix, Inc. Field Office or representative.

Changes to Tektronix instruments are sometimes made to accommodate improved components as they become available, and to give you the benefit of the latest circuit improvements developed in our engineering department. It is therefore important, when ordering parts, to include the following information in your order: Part number, instrument type or number, serial number, and modification number if applicable.

If a part you have ordered has been replaced with a new or improved part, your local Tektronix, Inc. Field Office or representative will contact you concerning any change in part number.

Change information, if any, is located at the rear of this manual.

SPECIAL NOTES AND SYMBOLS

X000	Part first added at this serial number
00X	Part removed after this serial number

FIGURE AND INDEX NUMBERS

Items in this section are referenced by figure and index numbers to the illustrations.

INDENTATION SYSTEM

This mechanical parts list is indented to indicate item relationships. Following is an example of the indentation system used in the description column.

1 2 3 4 5 Name & Description

Assembly and/or Component
Attaching parts for Assembly and/or Component
- - - * - - -
Detail Part of Assembly and/or Component
Attaching parts for Detail Part
- - - * - - -
Parts of Detail Part
Attaching parts for Parts of Detail Part
- - - * - - -

Attaching Parts always appear in the same indentation as the item it mounts, while the detail parts are indented to the right. Indented items are part of, and included with, the next higher indentation. The separation symbol - - - * - - - indicates the end of attaching parts.

Attaching parts must be purchased separately, unless otherwise specified.

ITEM NAME

In the Parts List, an Item Name is separated from the description by a colon (:). Because of space limitations, an Item Name may sometimes appear as incomplete. For further Item Name identification, the U.S. Federal Cataloging Handbook H6-1 can be utilized where possible.

ABBREVIATIONS

"	INCH
#	NUMBER SIZE
ACTR	ACTUATOR
ADPTR	ADAPTER
ALIGN	ALIGNMENT
AL	ALUMINUM
ASSEM	ASSEMBLED
ASSY	ASSEMBLY
ATTEN	ATTENUATOR
AWG	AMERICAN WIRE GAGE
BD	BOARD
BRKT	BRACKET
BRS	BRASS
BRZ	BRONZE
BSHG	BUSHING
CAB	CABINET
CAP	CAPACITOR
CER	CERAMIC
CHAS	CHASSIS
CKT	CIRCUIT
COMP	COMPOSITION
CONN	CONNECTOR
COV	COVER
CPLG	COUPLING
CRT	CATHODE RAY TUBE
DEG	DEGREE
DWR	DRAWER
ELCTRN	ELECTRON
ELEC	ELECTRICAL
ELCTLT	ELECTROLYTIC
ELEM	ELEMENT
EPL	ELECTRICAL PARTS LIST
EQPT	EQUIPMENT
EXT	EXTERNAL
FIL	FILLISTER HEAD
FLEX	FLEXIBLE
FLH	FLAT HEAD
FLTR	FILTER
FR	FRAME or FRONT
FSTNR	FASTENER
FT	FOOT
FXD	FIXED
GSKT	GASKET
HDL	HANDLE
HEX	HEXAGON
HEX HD	HEXAGONAL HEAD
HEX SOC	HEXAGONAL SOCKET
HLCPS	HELICAL COMPRESSION
HLEXT	HELICAL EXTENSION
HV	HIGH VOLTAGE
IC	INTEGRATED CIRCUIT
ID	INSIDE DIAMETER
IDENT	IDENTIFICATION
IMPLR	IMPELLER
IN	INCH
INCAND	INCANDESCENT
INSUL	INSULATOR
INTL	INTERNAL
LPHLDR	LAMPHOLDER
MACH	MACHINE
MECH	MECHANICAL
MTG	MOUNTING
NIP	NIPPLE
NON WIRE	NOT WIRE WOUND
OBD	ORDER BY DESCRIPTION
OD	OUTSIDE DIAMETER
OVH	OVAL HEAD
PH BRZ	PHOSPHOR BRONZE
PL	PLAIN or PLATE
PLSTC	PLASTIC
PN	PART NUMBER
PNH	PAN HEAD
PWR	POWER
RCPT	RECEPTACLE
RES	RESISTOR
RGD	RIGID
RLF	RELIEF
RTNR	RETAINER
SCH	SOCKET HEAD
SCOPE	OSCILLOSCOPE
SCR	SCREW
SE	SINGLE END
SECT	SECTION
SEMICOND	SEMICONDUCTOR
SHLD	SHIELD
SHLDR	SHOULDERED
SKT	SOCKET
SL	SLIDE
SLFLKG	SELF-LOCKING
SLVG	SLEEVING
SPR	SPRING
SQ	SQUARE
SST	STAINLESS STEEL
STL	STEEL
SW	SWITCH
T	TUBE
TERM	TERMINAL
THD	THREAD
THK	THICK
TNSN	TENSION
TPG	TAPPING
TRH	TRUSS HEAD
V	VOLTAGE
VAR	VARIABLE
W/	WITH
WSHR	WASHER
XFMR	TRANSFORMER
XSTR	TRANSISTOR

CROSS INDEX—MFR. CODE NUMBER TO MANUFACTURER

Mfr. Code	Manufacturer	Address	City, State, Zip
00779	AMP, INC.	P O BOX 3608	HARRISBURG, PA 17105
01295	TEXAS INSTRUMENTS, INC., SEMICONDUCTOR GROUP	P O BOX 5012, 13500 N CENTRAL EXPRESSWAY	DALLAS, TX 75222
15476	DIGITAL EQUIPMENT CORP.	146 MAIN ST.	MAYNARD, MA 01754
22526	BERG ELECTRONICS, INC.	YOUK EXPRESSWAY	NEW CUMBERLAND, PA 17070
73803	TEXAS INSTRUMENTS, INC., METALLURGICAL MATERIALS DIV.	34 FOREST STREET	ATTLEBORO, MA 02703
75037	MINNESOTA MINING AND MFG. CO., ELECTRO PRODUCTS DIVISION	3M CENTER	ST. PAUL, MN 55101
80009	TEKTRONIX, INC.	P O BOX 500	BEAVERTON, OR 97077
82647	TEXAS INSTRUMENTS, INC., CONTROL PRODUCTS DIV.	34 FOREST ST.	ATTLEBORO, MA 02703

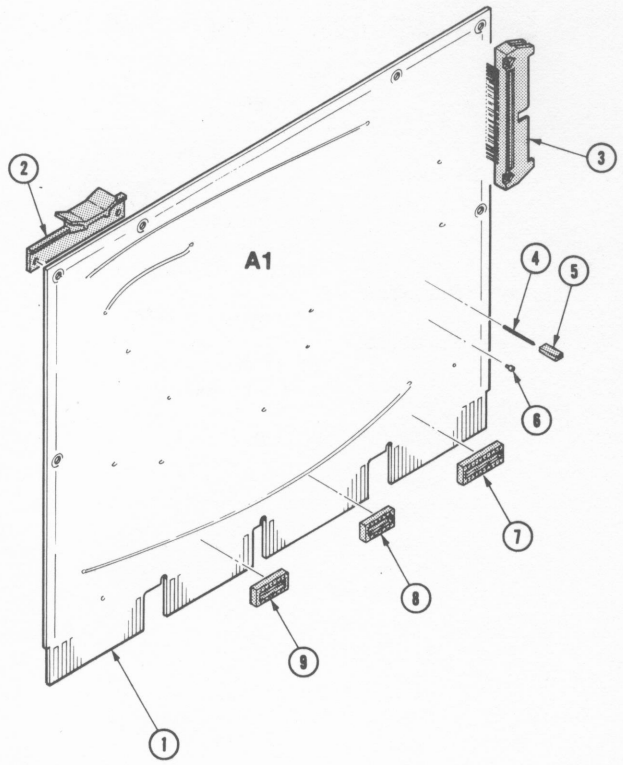


Fig. & Index No.	Tektronix Part No.	Serial/Model No. Eff	Dscont	Qty	1 2 3 4 5	Name & Description	Mfr Code	Mfr Part Number
-1	----- -----			1		CKT BOARD ASSY:PDP 11/IEEE 488(SEE A1 EPL)		
-2	367-0183-00			2		. PULL,CKT CARD:	15476	23930-00
-3	131-1648-00			1		. CONN,RCPT,ELEC:CKT BOARD,40 CONTACT	75037	3432-3005
-4	131-0608-00			42		. CONTACT,ELEC:0.365 L X 0.25 PH BRZ GOLD PL	22526	47357
-5	131-0993-00			14		. LINK,TERM.CONNE:2 WIRE BLACK	00779	530153-2
-6	136-0252-04			6		. SOCKET,PIN TERM:0.188 INCH LONG	22526	75060
-7	136-0634-00			2		. SOCKET,PLUG-IN:20 LEAD DIP,CKT BD MTG	73803	C952002
-8	136-0269-02			47		. SOCKET,PLUG-IN:14 CONTACT,LOW CLEARANCE	01295	C951401
-9	136-0260-02			63		. SOCKET,PLUG-IN:16 CONTACT,LOW CLEARANCE	82647	C951601
						STANDARD ACCESSORIES		
	070-2151-00			1		MANUAL,TECH:INSTRUCTION	80009	070-2151-00

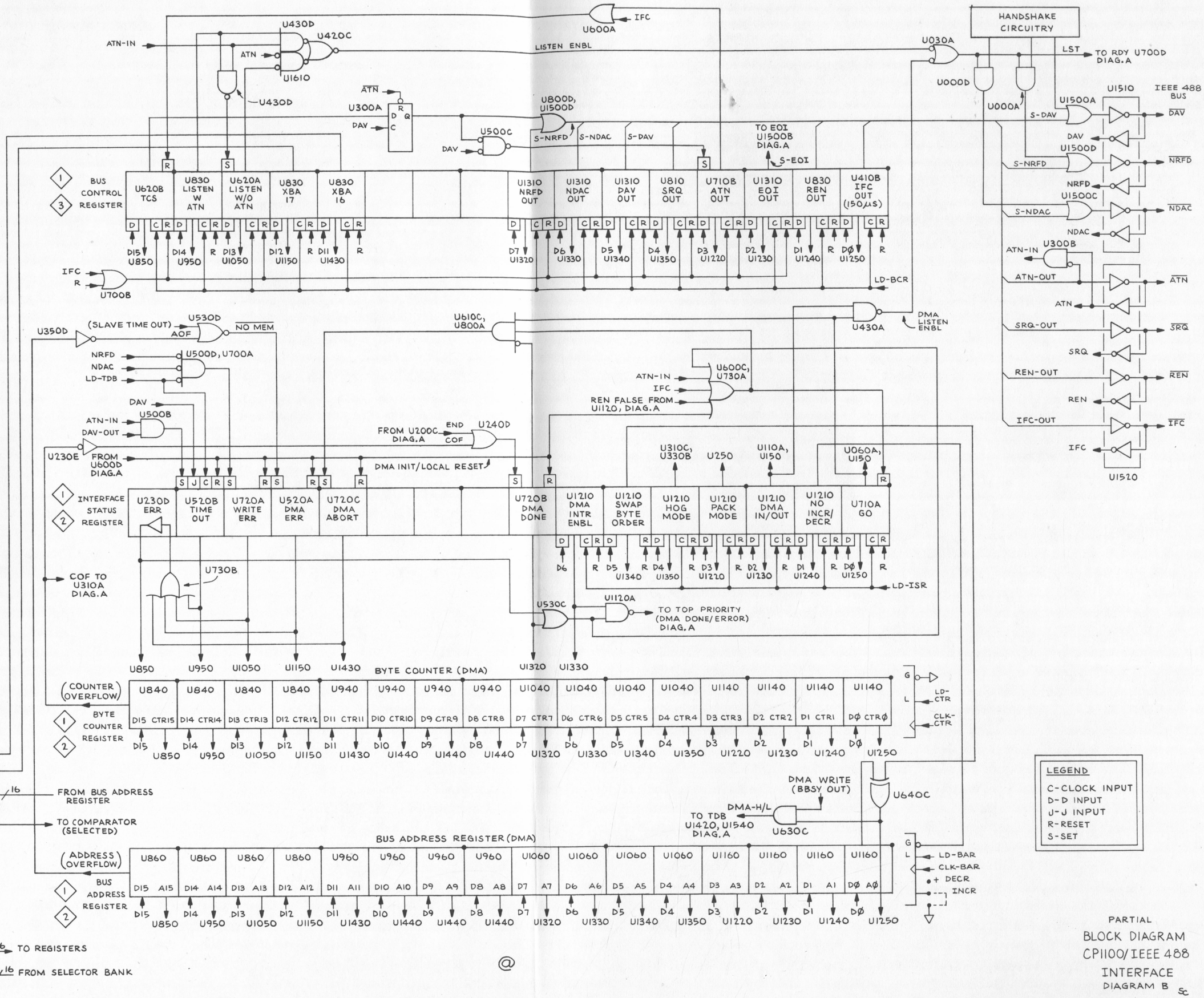
HIGH ORDER BYTE (DMA)
LOW ORDER BYTE (DMA)
SELECTOR
U1420, U1540
DMA-H/L
DONE
U700C
U1500B
U1510
S-EOI FROM U1310 DIAG. B
EOI
$\overline{EOI}$
LD-TDB
TALKER DATA BUFFER
U100A EOI
D15 D14 D13 D12 D11 D10 D09 D08 D07 D06 D05 D04 D03 D02 D01 D00
U1560 U1460 U1360 U1260
LATCH
U1410, U1530
IEEE 488 DATA BUS
U010C
U010B
U310A, U320B
COF FROM DIAG. B
DMA DATA (READ) FROM U140 PIN 15
EOI ATN REN
IMPLIED AND FUNCTIONS
LISTENER DATA BUFFER
DMA DATA (WRITE)
$\overline{ATN}$
EOI ATN REN REN FALSE OCC
DI08 DI07 DI06 DI05 DI04 DI03 DI02 DI01
U850 U950 U1050 U1150
U200D, U820A
U1320 U1330 U1340 U1350 U1220 U1230 U1240 U1250
READ LDB
IFC
DAV
LSI
FROM U030D (HANDSHAKE CIRCUITRY DIAG. B)
U700D
+V
LD-ICR
D6 D5 D4 D3 D2 D1 D0
U810B
REN
IFC (U1520)
LD-TDB
NDAC
ATN-IN
IFC
U600B
ATN
DAV
EOI
ATN (U1520)
U320
INTERRUPT CONTROL REGISTER
U810A DONE
ATN
U720D END OCC
U810B REN FALSE OCC
U1020A IFC OCC
U1010B RDY
U1130 RDY INTR ENBL
U1130 DONE INTR ENBL
U1130 SRQ INTR ENBL
U1130 ATN INTR ENBL
U1130 END INTR ENBL
U1130 REN INTR ENBL
U1130 IFC INTR ENBL
R (NORMALLY SET)
U1120A DUPLICATED ON DIAG. B
TOP PRIORITY
FROM U1210 DIAG. B
U530C DIAG. B
U1120A
DMA DONE/ERROR
U1030A
IFC
U1120B
REN FALSE
REN FALSE DIAG. B
U1120D
END
U1030C
SRQ
U1120C
ATN
U1030D
RDY
U1030B
DONE OR (NONE)
BOTTOM PRIORITY
TO U1110 (IRQ)
SRQ FROM U1520 DIAG. B
END
U200C
TO U240D DIAG. B
U850 U950 U1050 U1150 U1430
TO RESET DIAG. B
U420B U420A U420D U600D U700B
INIT
U1320 U1330 U1340 U1350 U1220 U1230 U1240 U1250
U1520
U440A U440D U430C U500A U260A
U1510 U1510 U1510 U1520 U1510 U1520 U1520
BUS STATUS REGISTER
U820B NO EOI DMA
NRFD NDAC DAV SRQ ATN EOI REN IFC
DI15 U850
DI3 DI2 DI1 DI0 D9
END INTR CLR
REN INTR CLR
IFC INTR CLR
DMA INIT
LOCAL RESET
LD-BSR
STRAPS
VECTOR
INTERRUPT VECTOR
S S S V V V Ø Ø
U1100, U1200, U1300
INTR VECTOR

V-BIT CODE

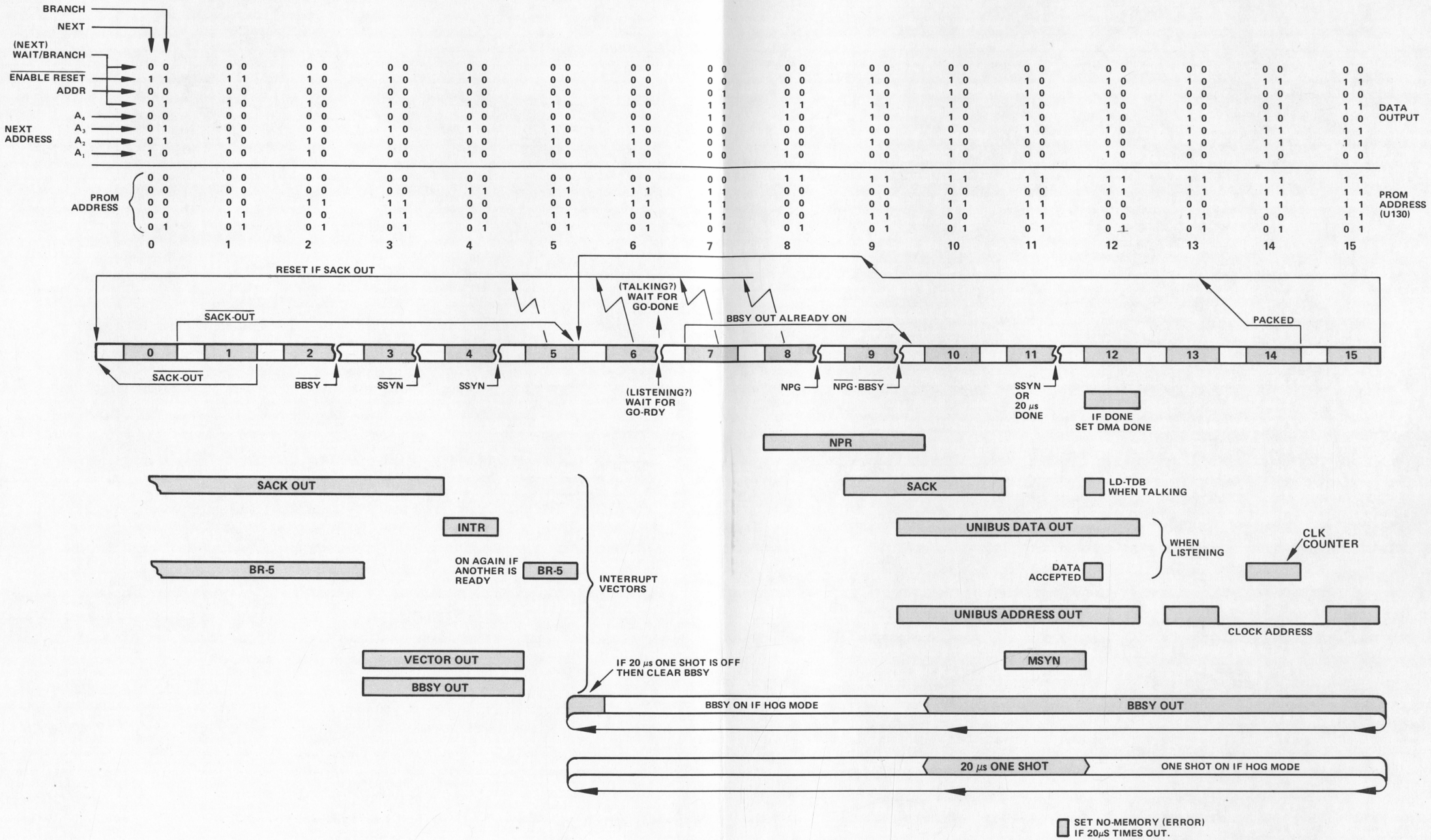
DMA DONE/ERROR	000
IFC	001
REN FALSE	010
END	011
SRQ	100
ATN	101
RDY	110
DONE (NONE)	111

LEGEND
C-CLOCK INPUT
D-D INPUT
J-J INPUT
R-RESET
S-SET

PARTIAL
BLOCK DIAGRAM
CP1100/IEEE 488
INTERFACE
DIAGRAM A Sc



PARTIAL
BLOCK DIAGRAM
CP1100/IEEE 488
INTERFACE
DIAGRAM B



INTERRUPT & DMA STATE MACHINE
CP1100/IEEE 488 INTERFACE

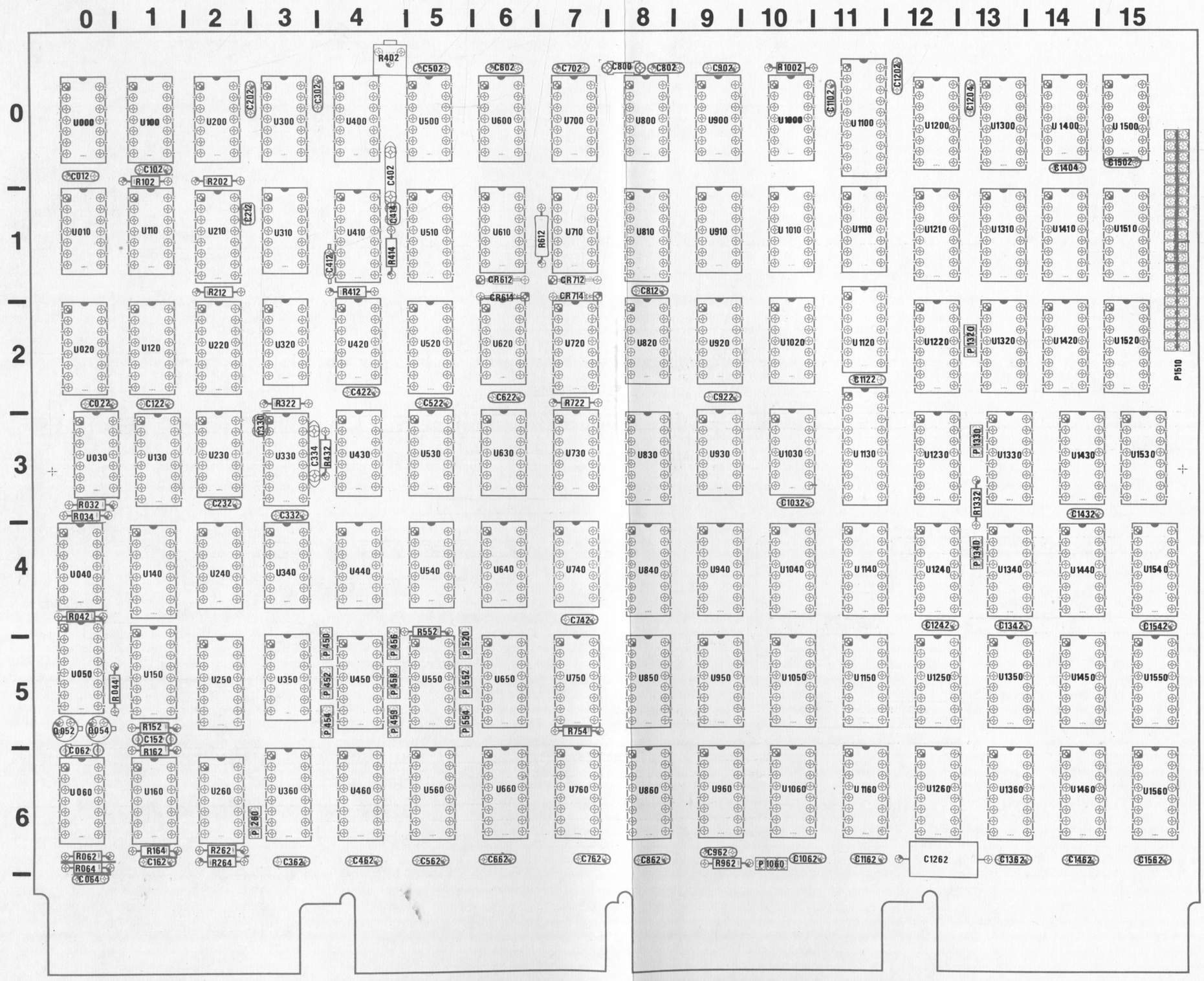
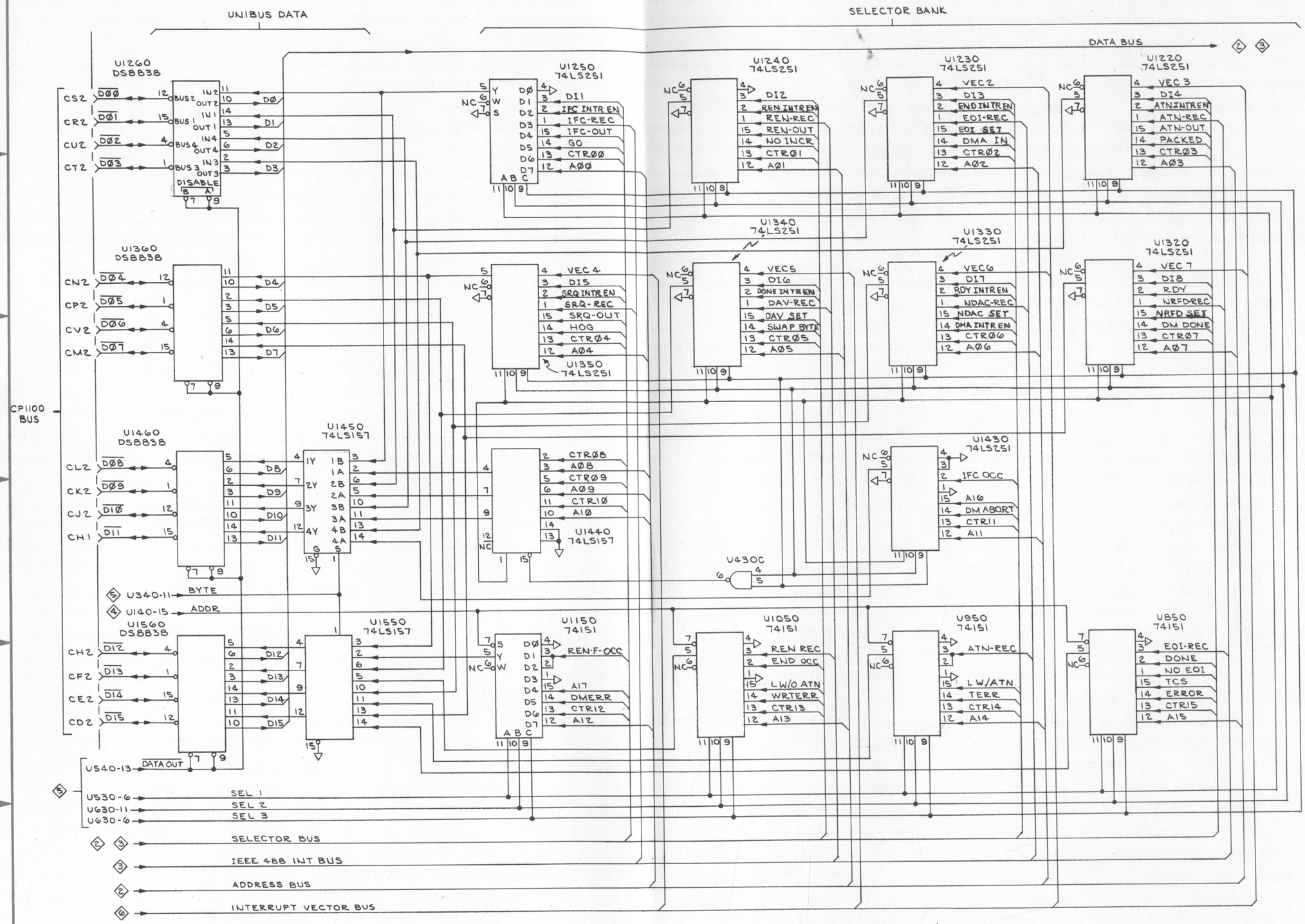


Fig. 8-1. Interface parts locator diagram.

P/O 670-5387-00				
COMPONENT NUMBER	SCHEMATIC LOCATION		BOARD LOCATION	
	COL	ROW	COL	ROW
U430C	E	4	4	3
U850	H	5	8	5
U950	F	5	9	5
U1050	E	5	10	5
U1150	D	5	11	5
U1220	H	1	12	2
U1230	F	1	12	3
U1240	E	1	12	4
U1250	D	1	12	5
U1260	B	1	12	6
U1320	H	2	13	2
U1330	F	2	13	3
U1340	E	2	13	4
U1350	D	3	13	5
U1360	B	2	13	6
U1430	F	3	14	3
U1440	D	4	14	4
U1450	C	4	14	5
U1460	B	4	14	6
U1550	C	5	15	5
U1560	B	5	15	6

COMPONENT CROSS REFERENCE TABLE

SCHEMATIC 1


A B C D E F G H
1 2 3 4 5 6
UNIBUS DATA
SELECTOR BANK
DATA BUS
U1260 DS8838
U1250 74LS251
U1240 74LS251
U1230 74LS251
U1220 74LS251
U1340 74LS251
U1330 74LS251
U1320 74LS251
U1350 74LS251
U1360 DS8838
U1460 DS8838
U1450 74LS157
U1440 74LS157
U1430 74LS251
U430C
U1560 DS8838
U1550 74LS157
U1150 74151
U1050 74151
U950 74151
U850 74151
CP1100 BUS
U340-11 BYTE
U140-15 ADDR
U540-13 DATA OUT
U530-6 SEL 1
U630-11 SEL 2
U630-6 SEL 3
SELECTOR BUS
IEEE 488 INT BUS
ADDRESS BUS
INTERRUPT VECTOR BUS
UNIBUS DATA/SELECTOR BANK
CP1100/IEEE 488 INTERFACE
P/O 670-5387-00

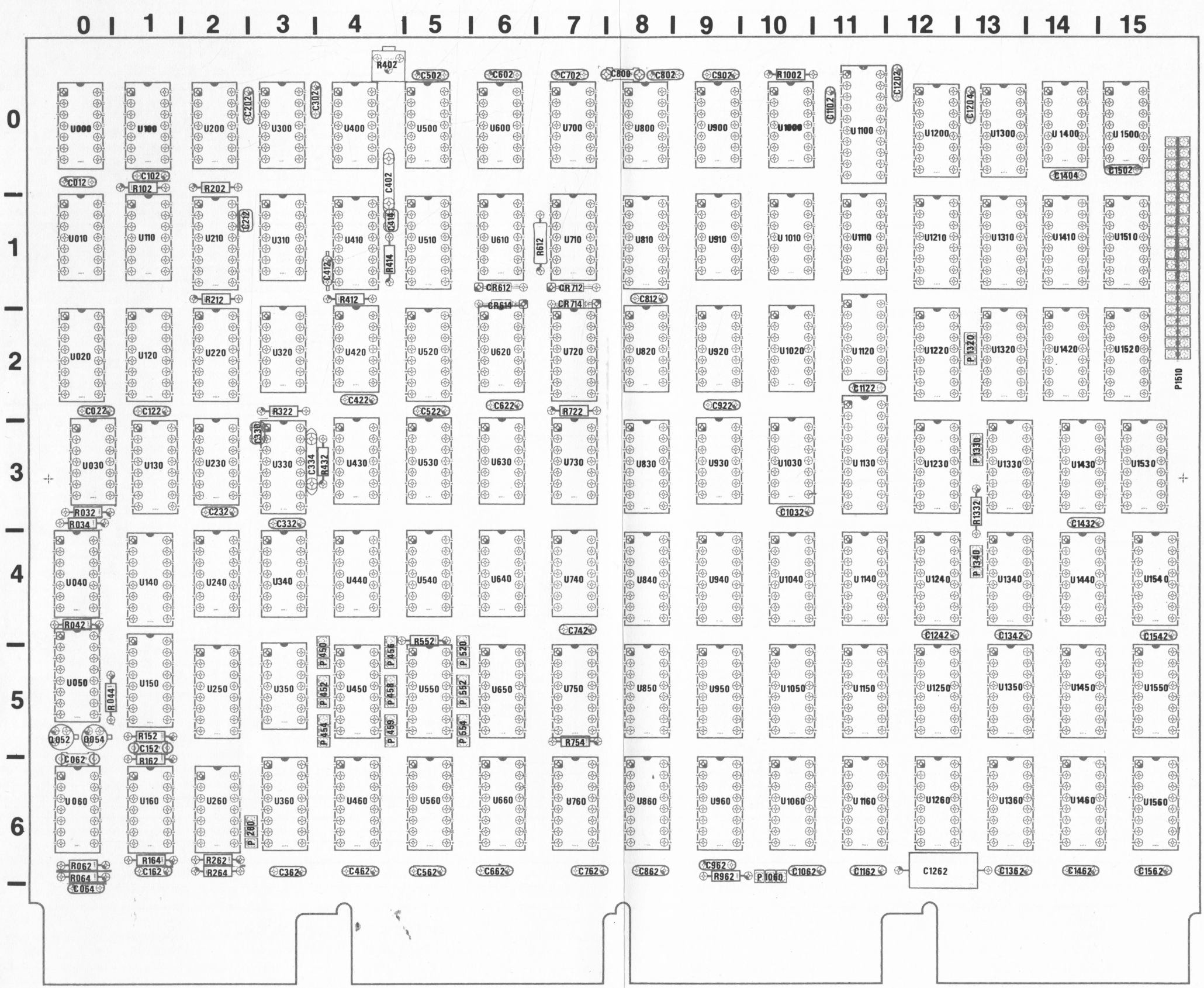


Fig. 8-2. Interface parts locator diagram.

P/O 670-5387-00

COMPONENT NUMBER	SCHEMATIC LOCATION		BOARD LOCATION	
	COL	ROW	COL	ROW
P1060	B	2	10	6
R962	B	2	9	6
U230D	G	6	2	3
U230E	D	5	2	3
U350D	B	5	3	5
U400F	G	2	4	0
U500A	F	6	5	0
U500B	D	5	5	0
U500D	D	6	5	0
U520A	F	4	5	2
U520B	F	5	5	2
U530C	G	3	5	3
U600C	F	2	6	0
U600D	F	6	6	0
U610C	G	2	7	1
U640C	C	2	6	4
U700A	D	6	7	0
U710A	F	2	7	1
U720A	F	5	7	2
U720B	F	4	7	2
U720C	F	4	7	2
U730A	E	2	7	3
U730B	G	6	7	3
U800A	G	4	8	0
U840	D	4	8	4
U860	B	4	8	6
U940	D	4	9	4
U960	B	4	9	6
U1040	D	3	10	4
U1060	B	3	10	6
U1120A	G	3	11	2
U1140	D	2	11	4
U1160	B	2	11	6
U1210	F	3	12	1
U1400C	F	3	14	0

COMPONENT CROSS REFERENCE TABLE

BUS ADDRESS REGISTER (BAR) | COUNTER (CTR) | INTERFACE STATUS REGISTER (ISR)

3 U260-3 & 11 R
U400-10 CLR
U1120-6 REN-INT

NO INCR
SWAP BYTE
U640C
CLK-ADDR
U1160 74LS191
U1140 74LS191
U1060 74LS191
U1040 74LS191
U960 74LS191
U940 74LS191
U860 74LS191
U840 74LS191
+5V
R962 2.2K
P1060
HI
DN/UP LD
RIPPLE CLOCK
D0 D1 D2 D3 → A00 A01 A02 A03 / CTR00 CTR01 CTR02 CTR03
D4 D5 D6 D7 → A04 A05 A06 A07 / CTR04 CTR05 CTR06 CTR07
D8 D9 D10 D11 → A08 A09 A10 A11 / CTR08 CTR09 CTR10 CTR11
D12 D13 D14 D15 → A12 A13 A14 A15 / CTR12 CTR13 CTR14 CTR15

* JUMPER H1 IS SET SO ADDRESS COUNTS UP

4 U230-12 CLK-CTR

5 U750-7 LD-CTR
U750-9 LD-BAR

U350D A.O.F. (ADDRESS OVERFLOW)
U230E C.O.F.

1 DATA BUS
3 U1500-3 DAV-OUT
5 U750-10 LD-ISR
4 U310-6 DMA-DONE
U530-11 NO-MEM
U440-8 CLK-ADDR
5 U540-1 LD-TDB
3 IEEE 488 INT BUS
5 U630-3 TDB
U350-10 LD-BSR

U500B
ATN-IN
U700A
NRFD-REC
NDAC-REC
DAV-REC
U500D

U730A 7425
+V3
CLR
REN-INT
ATN-IN
LD-ISR
U710A 74LS74
U600C
U610C
U400F
U1210 74LS174
U1400C
LD ISR
U1120A
U530C
U720B
U800A 74LS32
U720C
U520A 74LS76
U720A 74LS279
U520B
U730B 7425
U600D
U500A 74LS00

ADDRESS BUS → 1 5
SELECTOR BUS → 1
A00 → U630-9 3
GO → U430-2 3
GO → U060-2 4
DMA IN → U610-2 4
DMA IN → U430-1 3
DMA IN → U010-12 U800-10 4
DMA IN → U340-9 5
NO INCR
DMA IN
PACKED
HOG
SWAP BYTE
DMA INTREN
DMA-INT → U1000-2 U1010-3 6
HOG → U310-10 4
PACKED → U250-13 4 U340-13 5
DM DONE
DM ABORT
NO-MEM
A.O.F.
CLK-ADDR
DMERR
NC
NO LISTENER 2,3
WRTERR
DAV-REC
TDB
TERR
DMA INIT
ERROR → U230D
DI0 LD-BSR
(COUNTER OVERFLOW) C.O.F. → U320-5 3 U240-11 4

@

BAR/CTR/ISR
CP1100/IEEE 488 INTERFACE
P/O 670-5387-00
2
3/9/78

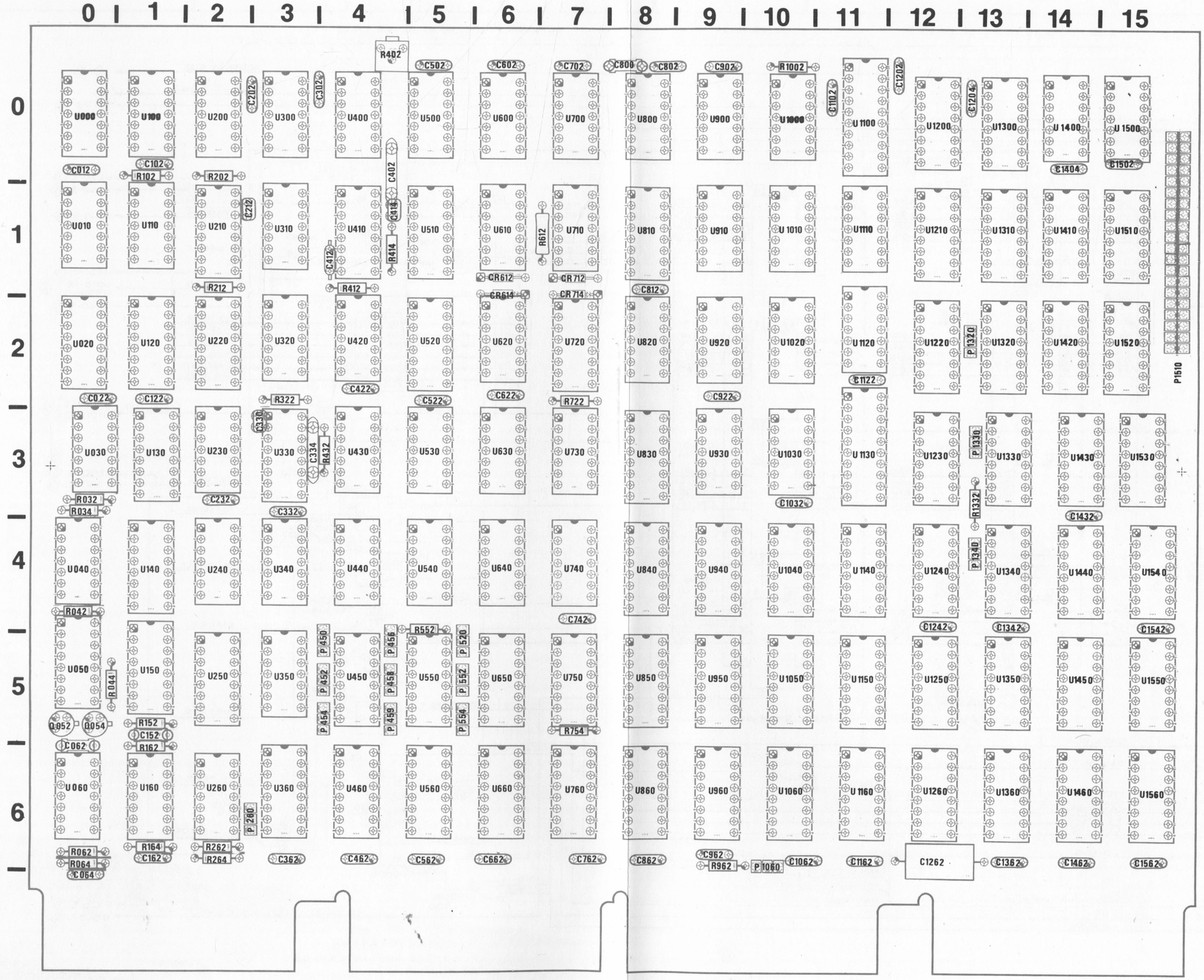


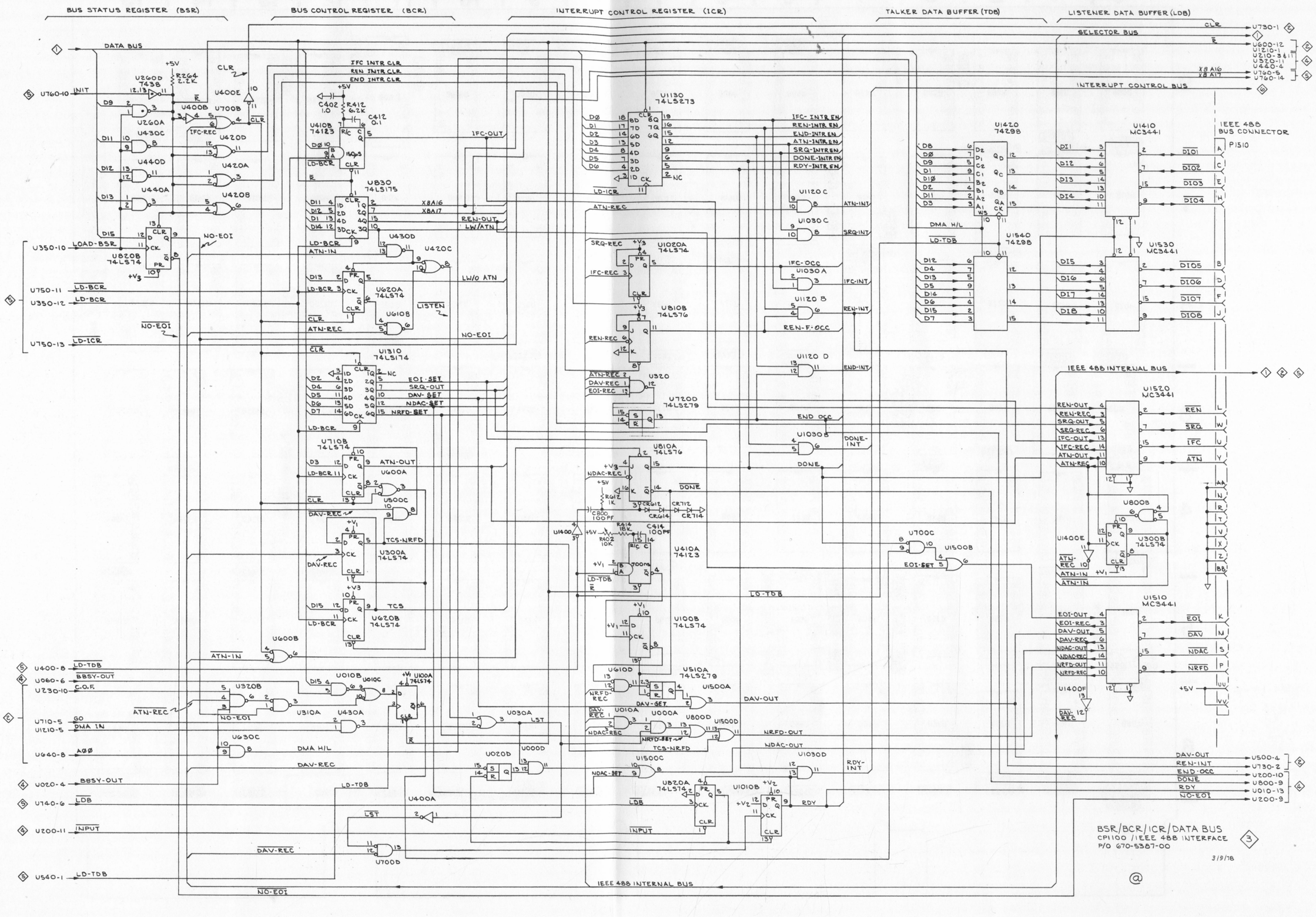
Fig. 8-3. Interface parts locator diagram.

P/O 670-5387-00

COMPONENT NUMBER	SCHEMATIC LOCATION COL	SCHEMATIC LOCATION ROW	BOARD LOCATION COL	BOARD LOCATION ROW
C402	C	1	4	0
C412	D	2	4	1
C414	F	5	4	1
C800	F	5	8	0
CR612	F	5	6	1
CR614	F	5	6	1
CR712	F	5	7	1
CR714	G	5	7	1
P1510	K	3	15	2
R264	B	1	2	6
R402	F	5	4	0
R412	D	1	4	1
R414	F	6	4	1
R612	F	5	7	1
U000A	F	7	0	0
U000D	E	7	0	0
U010A	F	7	0	1
U010B	C	6	0	1
U010C	D	7	0	1
U020D	E	7	0	2
U030A	E	7	0	3
U100A	D	6	1	0
U100B	F	6	1	0
U260A	B	1	2	6
U260D	B	1	2	6
U300A	D	5	3	0
U300B	K	5	3	0
U310A	C	7	3	1
U320A	F	4	3	2
U320B	C	7	3	2
U400A	D	8	4	0
U400B	B	2	4	0
U400E	C	1	4	0
U410A	F	5	4	1
U410B	C	2	4	1
U420A	B	2	4	2
U420B	B	2	4	2
U420C	D	3	4	2
U420D	B	2	4	2
U430A	D	7	4	3
U430C	B	2	4	3
U430D	D	3	4	3
U440A	B	2	4	4
U440D	B	2	4	4

COMPONENT NUMBER	SCHEMATIC LOCATION COL	SCHEMATIC LOCATION ROW	BOARD LOCATION COL	BOARD LOCATION ROW
U500C	D	5	5	0
U510A	F	6	5	1
U600A	D	5	6	0
U600B	C	6	6	0
U610B	D	3	6	1
U610D	F	6	6	1
U620A	D	3	6	2
U620B	D	6	6	2
U630C	C	7	6	3
U700B	B	2	7	0
U700C	I	5	7	0
U700D	D	8	7	0
U710B	D	5	7	1
U720D	F	4	7	2
U800B	K	5	8	0
U800D	G	7	8	0
U810A	F	5	8	1
U810B	F	3	8	1
U820A	G	7	8	2
U820B	B	3	8	2
U830	D	2	8	3
U1010B	G	7	10	1
U1020A	F	3	10	2
U1030A	H	3	10	3
U1030B	H	4	10	3
U1030C	H	2	10	3
U1030D	H	7	10	3
U1120B	H	3	11	2
U1120C	H	2	11	2
U1120D	H	4	11	2
U1130	F	2	11	3
U1310	D	4	13	1
U1400B	F	5	14	0
U1400E	J	5	14	0
U1400F	J	7	14	0
U1410	K	2	14	1
U1420	I	2	14	2
U1500A	G	7	15	0
U1500B	I	5	15	0
U1500C	F	7	15	0
U1500D	G	7	15	0
U1510	K	6	15	1
U1520	K	4	15	2
U1530	K	3	15	3
U1540	I	3	15	4

COMPONENT CROSS REFERENCE TABLE


BUS STATUS REGISTER (BSR)
BUS CONTROL REGISTER (BCR)
INTERRUPT CONTROL REGISTER (ICR)
TALKER DATA BUFFER (TDB)
LISTENER DATA BUFFER (LDB)
DATA BUS
INTERRUPT CONTROL BUS
SELECTOR BUS
IEEE 488 BUS CONNECTOR
P1510
IEEE 488 INTERNAL BUS
IFC INTR CLR
REN INTR CLR
END INTR CLR
U1130 74LS273
U1420 74298
U1410 MC3441
U1530 MC3441
U1540 74298
U1520 MC3441
U1510 MC3441
U820B 74LS174
U830 74LS175
U1310 74LS174
U710B 74LS174
U300A 74LS174
U620B 74LS174
U620A 74LS174
U100B 74LS74
U810A 74LS76
U810B 74LS76
U820A 74LS74
U1010B
U800B
U300B 74LS74
U1400E
U1400F
U1500B
U1500C
U1500D
U1030A
U1030B
U1030C
U1030D
U1120 B
U1120 C
U1120 D
U700C
U700D
U510A 74LS279
U720D 74LS279
U1020A 74LS74
U410A 74123
U410B 74123
U400A
U400B
U400E
U420A
U420B
U420C
U420D
U430A
U430B
U430C
U430D
U440A
U440B
U440C
U440D
U500A
U500C
U600A
U600B
U610B
U620C
U630A
U630C
U630D
U010A
U010B
U010C
U000A
U000D
U020A
U020D
U320
U310A
U320B
U730A
U730B
U730C
U730D
U810C
U820C
U820D
U1010A
U1020B
U1020C
U1020D
U1120A
U1120C
U1120D
U1130A
U1130B
U1130C
U1130D
U1130E
U1130F
U1130G
U1130H
U1130J
U1130K
U1130L
U1130M
U1130N
U1130P
U1130R
U1130S
U1130T
U1130U
U1130V
U1130W
U1130X
U1130Y
U1130Z
U1130AA
U1130BB
U1130CC
U1130DD
U1130EE
U1130FF
U1130GG
U1130HH
U1130JJ
U1130KK
U1130LL
U1130MM
U1130NN
U1130PP
U1130RR
U1130SS
U1130TT
U1130UU
U1130VV
U1130WW
U1130XX
U1130YY
U1130ZZ
U1130AAA
U1130BBB
U1130CCC
U1130DDD
U1130EEE
U1130FFF
U1130GGG
U1130HHH
U1130JJJ
U1130KKK
U1130LLL
U1130MMM
U1130NNN
U1130PPP
U1130RRR
U1130SSS
U1130TTT
U1130UUU
U1130VVV
U1130WWW
U1130XXX
U1130YYY
U1130ZZZ
U1130AAAA
U1130BBBB
U1130CCCC
U1130DDDD
U1130EEEE
U1130FFFF
U1130GGGG
U1130HHHH
U1130JJJJ
U1130KKKK
U1130LLLL
U1130MMMM
U1130NNNN
U1130PPPP
U1130RRRR
U1130SSSS
U1130TTTT
U1130UUUU
U1130VVVV
U1130WWWW
U1130XXXX
U1130YYYY
U1130ZZZZ
U1130AAAAA
U1130BBBBB
U1130CCCCC
U1130DDDDD
U1130EEEEE
U1130FFFFF
U1130GGGGG
U1130HHHHH
U1130JJJJJ
U1130KKKKK
U1130LLLLL
U1130MMMMM
U1130NNNNN
U1130PPPPP
U1130RRRRR
U1130SSSSS
U1130TTTTT
U1130UUUUU
U1130VVVVV
U1130WWWWW
U1130XXXXX
U1130YYYYY
U1130ZZZZZ
U1130AAAAAA
U1130BBBBBB
U1130CCCCCC
U1130DDDDDD
U1130EEEEEE
U1130FFFFFF
U1130GGGGGG
U1130HHHHHH
U1130JJJJJJ
U1130KKKKKK
U1130LLLLLL
U1130MMMMMM
U1130NNNNNN
U1130PPPPPP
U1130RRRRRR
U1130SSSSSS
U1130TTTTTT
U1130UUUUUU
U1130VVVVVV
U1130WWWWWW
U1130XXXXXX
U1130YYYYYY
U1130ZZZZZZ
U1130AAAAAAA
U1130BBBBBBB
U1130CCCCCCC
U1130DDDDDDD
U1130EEEEEEE
U1130FFFFFFF
U1130GGGGGGG
U1130HHHHHHH
U1130JJJJJJJ
U1130KKKKKKK
U1130LLLLLLL
U1130MMMMMMM
U1130NNNNNNN
U1130PPPPPPP
U1130RRRRRRR
U1130SSSSSSS
U1130TTTTTTT
U1130UUUUUUU
U1130VVVVVVV
U1130WWWWWWW
U1130XXXXXXX
U1130YYYYYYY
U1130ZZZZZZZ
U1130AAAAAAAA
U1130BBBBBBBB
U1130CCCCCCCC
U1130DDDDDDDD
U1130EEEEEEEE
U1130FFFFFFFF
U1130GGGGGGGG
U1130HHHHHHHH
U1130JJJJJJJJ
U1130KKKKKKKK
U1130LLLLLLLL
U1130MMMMMMMM
U1130NNNNNNNN
U1130PPPPPPPP
U1130RRRRRRRR
U1130SSSSSSSS
U1130TTTTTTTT
U1130UUUUUUUU
U1130VVVVVVVV
U1130WWWWWWWW
U1130XXXXXXXX
U1130YYYYYYYY
U1130ZZZZZZZZ
U1130AAAAAAAAA
U1130BBBBBBBBB
U1130CCCCCCCCC
U1130DDDDDDDDD
U1130EEEEEEEEE
U1130FFFFFFFFF
U1130GGGGGGGGG
U1130HHHHHHHHH
U1130JJJJJJJJJ
U1130KKKKKKKKK
U1130LLLLLLLLL
U1130MMMMMMMMM
U1130NNNNNNNNN
U1130PPPPPPPPP
U1130RRRRRRRRR
U1130SSSSSSSSS
U1130TTTTTTTTT
U1130UUUUUUUUU
U1130VVVVVVVVV
U1130WWWWWWWWW
U1130XXXXXXXXX
U1130YYYYYYYYY
U1130ZZZZZZZZZ
U1130AAAAAAAAAA
U1130BBBBBBBBBB
U1130CCCCCCCCCC
U1130DDDDDDDDDD
U1130EEEEEEEEEE
U1130FFFFFFFFFF
U1130GGGGGGGGGG
U1130HHHHHHHHHH
U1130JJJJJJJJJJ
U1130KKKKKKKKKK
U1130LLLLLLLLLL
U1130MMMMMMMMMM
U1130NNNNNNNNNN
U1130PPPPPPPPPP
U1130RRRRRRRRRR
U1130SSSSSSSSSS
U1130TTTTTTTTTT
U1130UUUUUUUUUU
U1130VVVVVVVVVV
U1130WWWWWWWWWW
U1130XXXXXXXXXX
U1130YYYYYYYYYY
U1130ZZZZZZZZZZ
U1130AAAAAAAAAAA
U1130BBBBBBBBBBB
U1130CCCCCCCCCCC
U1130DDDDDDDDDDD
U1130EEEEEEEEEEE
U1130FFFFFFFFFFF
U1130GGGGGGGGGGG
U1130HHHHHHHHHHH
U1130JJJJJJJJJJJ
U1130KKKKKKKKKKK
U1130LLLLLLLLLLL
U1130MMMMMMMMMMM
U1130NNNNNNNNNNN
U1130PPPPPPPPPPP
U1130RRRRRRRRRRR
U1130SSSSSSSSSSS
U1130TTTTTTTTTTT
U1130UUUUUUUUUUU
U1130VVVVVVVVVVV
U1130WWWWWWWWWWW
U1130XXXXXXXXXXX
U1130YYYYYYYYYYY
U1130ZZZZZZZZZZZ
U1130AAAAAAAAAAAA
U1130BBBBBBBBBBBB
U1130CCCCCCCCCCCC
U1130DDDDDDDDDDDD
U1130EEEEEEEEEEEE
U1130FFFFFFFFFFFF
U1130GGGGGGGGGGGG
U1130HHHHHHHHHHHH
U1130JJJJJJJJJJJJ
U1130KKKKKKKKKKKK
U1130LLLLLLLLLLLL
U1130MMMMMMMMMMMM
U1130NNNNNNNNNNNN
U1130PPPPPPPPPPPP
U1130RRRRRRRRRRRR
U1130SSSSSSSSSSSS
U1130TTTTTTTTTTTT
U1130UUUUUUUUUUUU
U1130VVVVVVVVVVVV
U1130WWWWWWWWWWWW
U1130XXXXXXXXXXXX
U1130YYYYYYYYYYYY
U1130ZZZZZZZZZZZZ
U1130AAAAAAAAAAAAA
U1130BBBBBBBBBBBBB
U1130CCCCCCCCCCCCC
U1130DDDDDDDDDDDDD
U1130EEEEEEEEEEEEE
U1130FFFFFFFFFFFFF
U1130GGGGGGGGGGGGG
U1130HHHHHHHHHHHHH
U1130JJJJJJJJJJJJJ
U1130KKKKKKKKKKKKK
U1130LLLLLLLLLLLLL
U1130MMMMMMMMMMMMM
U1130NNNNNNNNNNNNN
U1130PPPPPPPPPPPPP
U1130RRRRRRRRRRRRR
U1130SSSSSSSSSSSSS
U1130TTTTTTTTTTTTT
U1130UUUUUUUUUUUUU
U1130VVVVVVVVVVVVV
U1130WWWWWWWWWWWWW
U1130XXXXXXXXXXXXX
U1130YYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
U1130CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
U1130DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
U1130EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE
U1130FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
U1130GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
U1130HHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHH
U1130JJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJJ
U1130KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK
U1130LLLLLLLLLLLLLLLLLLLLLLLLLLLLLLLL
U1130MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
U1130NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
U1130PPPPPPPPPPPPPPPPPPPPPPPPPPPPPPPP
U1130RRRRRRRRRRRRRRRRRRRRRRRRRRRRRRRR
U1130SSSSSSSSSSSSSSSSSSSSSSSSSSSSSSSS
U1130TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT
U1130UUUUUUUUUUUUUUUUUUUUUUUUUUUUUUUU
U1130VVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVV
U1130WWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
U1130XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
U1130YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
U1130ZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZZ
U1130AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
U1130BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB

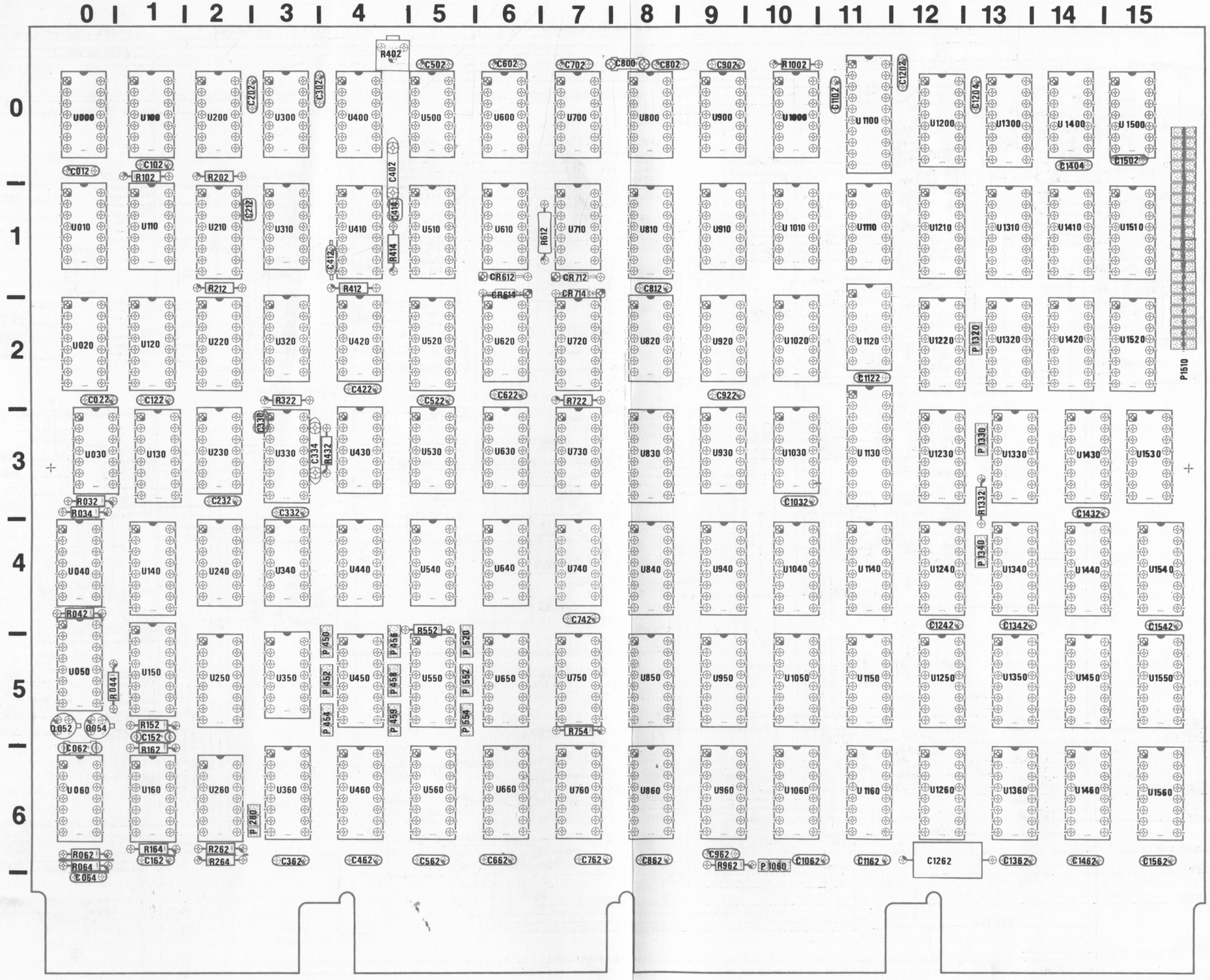
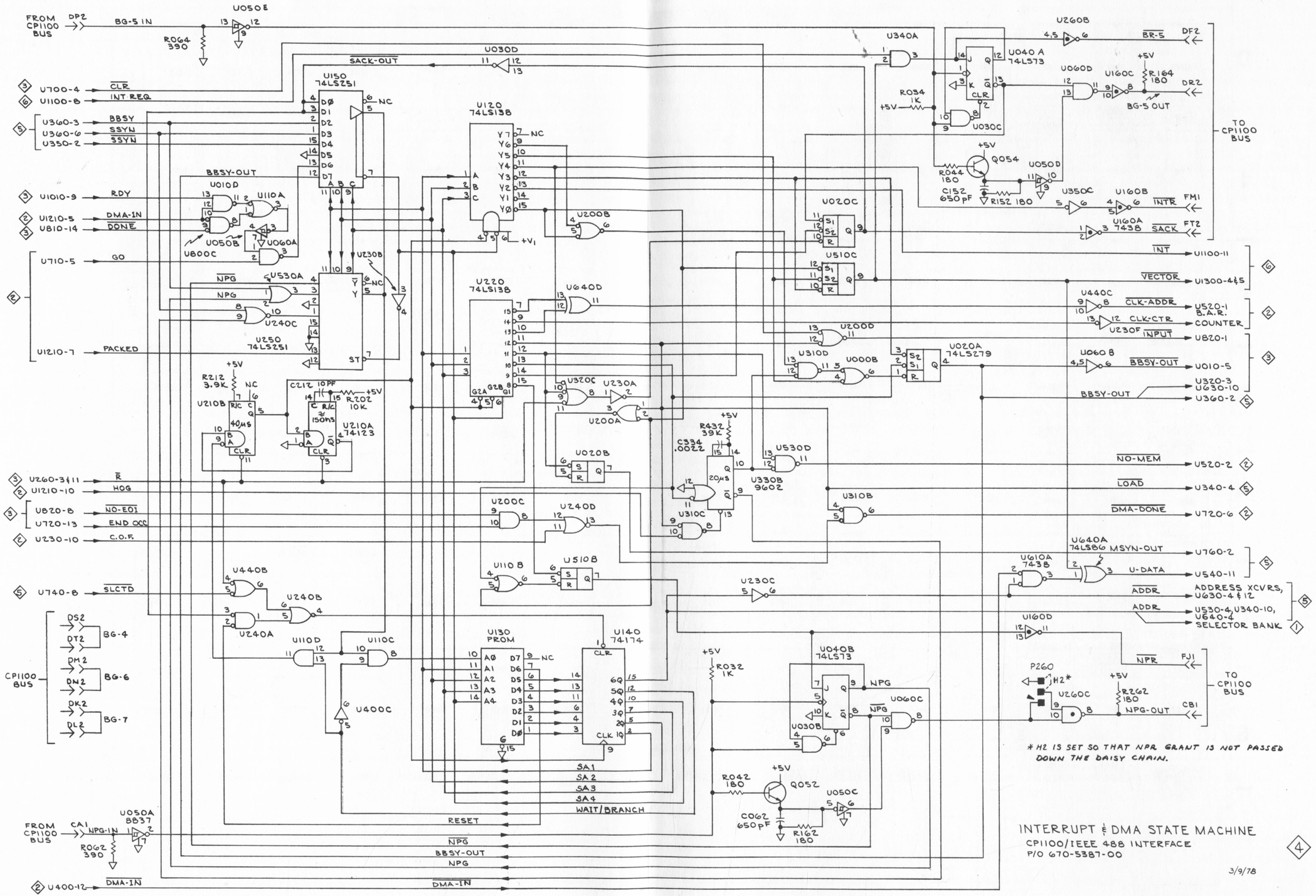


Fig. 8-4. Interface parts locator diagram.

P/O 670-5387-00

COMPONENT NUMBER	SCHEMATIC LOCATION COL	SCHEMATIC LOCATION ROW	BOARD LOCATION COL	BOARD LOCATION ROW
C062	F	6	0	6
C152	G	2	1	5
C212	C	3	2	1
C334	E	4	3	3
P260	H	5	2	6
Q052	F	6	0	5
Q054	G	2	0	5
R032	E	5	0	3
R034	G	1	0	3
R042	F	6	0	4
R044	G	2	0	5
R062	A	6	1	6
R064	B	1	0	6
R152	G	2	1	5
R162	F	6	1	6
R164	H	1	1	6
R202	C	3	2	0
R212	B	3	2	1
R262	H	5	2	6
R432	E	3	4	2
U000B	F	3	0	0
U010D	B	2	0	1
U020A	G	3	0	2
U020B	E	4	0	2
U020C	F	2	0	2
U030B	F	6	0	3
U030C	G	1	0	3
U030D	D	1	0	3
U040A	G	1	0	4
U040B	F	5	0	4
U050A	A	6	0	5
U050B	B	2	0	5
U050C	F	6	0	5
U050D	H	1	0	5
U050E	B	1	0	5
U060A	B	2	0	6
U060B	H	3	0	6
U060C	G	5	0	6
U060D	H	1	0	6
U110A	B	2	1	1
U110B	D	5	1	1
U110C	B	5	1	1
U110D	B	5	1	1
U120	D	1	1	2
U130	D	5	1	3
U140	E	5	1	4
U150	C	1	1	5
U160A	H	2	1	6
U160B	H	2	1	6
U160C	H	1	1	6
U160D	H	5	1	6
U200A	E	3	2	0
U200B	E	2	2	0
U200C	D	4	2	0
U200D	F	3	2	0
U210A	C	3	2	1
U210B	B	3	2	1
U220	D	3	2	2
U230A	E	3	2	3
U230B	C	3	2	3
U230C	F	5	2	3
U230F	H	3	2	3
U240A	B	5	2	4
U240B	C	5	2	4
U240C	B	3	2	4
U240D	D	4	2	4
U250	C	3	2	5
U260B	H	1	2	6
U260C	H	5	2	6
U310B	F	4	3	1
U310C	E	4	3	1
U310D	F	3	3	1
U320C	D	3	3	2
U340A	G	1	3	4
U350C	H	2	3	5
U400C	C	5	4	0
U440B	B	5	4	4
U440C	H	3	4	4
U510B	D	4	5	1
U510C	F	2	5	1
U530A	B	3	5	3
U530D	F	4	5	3
U610A	H	4	6	1
U640A	H	4	6	4
U640D	D	3	6	4
U800C	B	2	8	0

COMPONENT CROSS REFERENCE TABLE



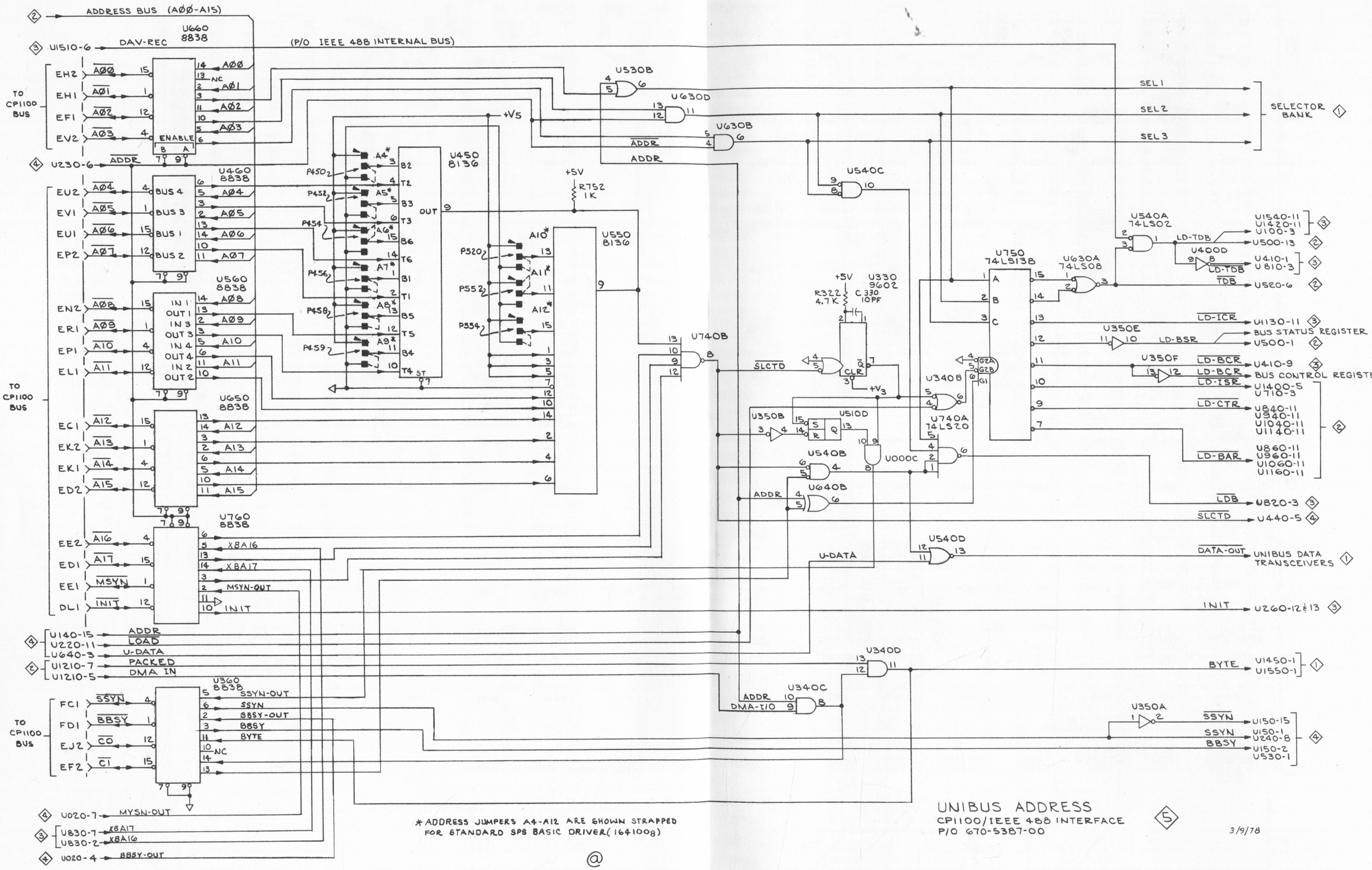
INTERRUPT & DMA STATE MACHINE
CP1100/IEEE 488 INTERFACE
P/O 670-5387-00

3/9/78

Fig. 8-5. Interface parts locator diagram.

P/O 670-5387-00				
COMPONENT NUMBER	SCHEMATIC LOCATION		BOARD LOCATION	
	COL	ROW	COL	ROW
C330	F	3	3	3
P450	C	2	4	5
P452	C	2	4	5
P454	C	2	4	5
P456	C	2	4	5
P458	C	3	4	5
P459	C	3	4	5
P520	D	2	5	5
P552	D	2	5	5
P554	D	3	5	5
R322	F	3	3	2
R752	D	2	7	5
U000C	G	4	0	0
U330	F	3	3	3
U340B	G	3	3	4
U340C	F	5	3	4
U340D	F	5	3	4
U350A	H	5	3	5
U350B	F	3	3	5
U350E	H	3	3	5
U350F	H	3	3	5
U360	B	5	3	6
U400D	I	2	4	0
U450	C	2	4	5
U460	B	2	4	6
U510D	F	3	5	1
U530B	E	1	5	3
U540A	H	2	5	4
U540B	F	4	5	4
U540C	F	2	5	4
U540D	G	4	5	4
U550	D	2	5	5
U560	B	3	5	6
U630A	H	2	6	3
U630B	E	1	6	3
U630D	E	1	6	3
U640B	F	4	6	4
U650	B	3	6	5
U660	B	1	6	6
U740A	G	4	7	4
U740B	E	3	7	4
U750	G	2	7	5
U760	B	4	7	6

COMPONENT CROSS REFERENCE TABLE



*ADDRESS JUMPERS A4-A12 ARE SHOWN STRAPPED FOR STANDARD SPS BASIC DRIVER(1641008)

@

UNIBUS ADDRESS

CP1100/IEEE 488 INTERFACE
P/O 670-5387-00

5

3/9/78

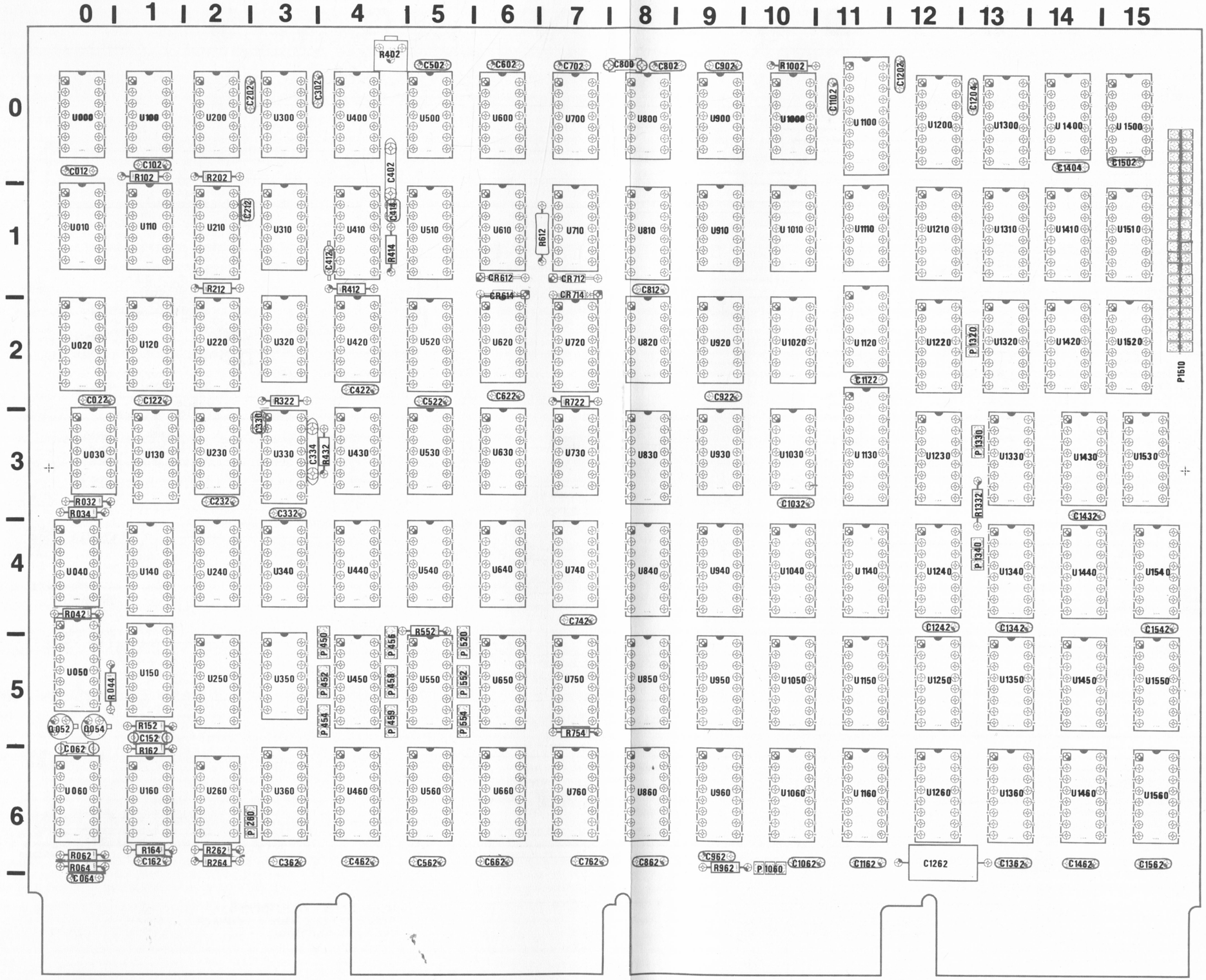


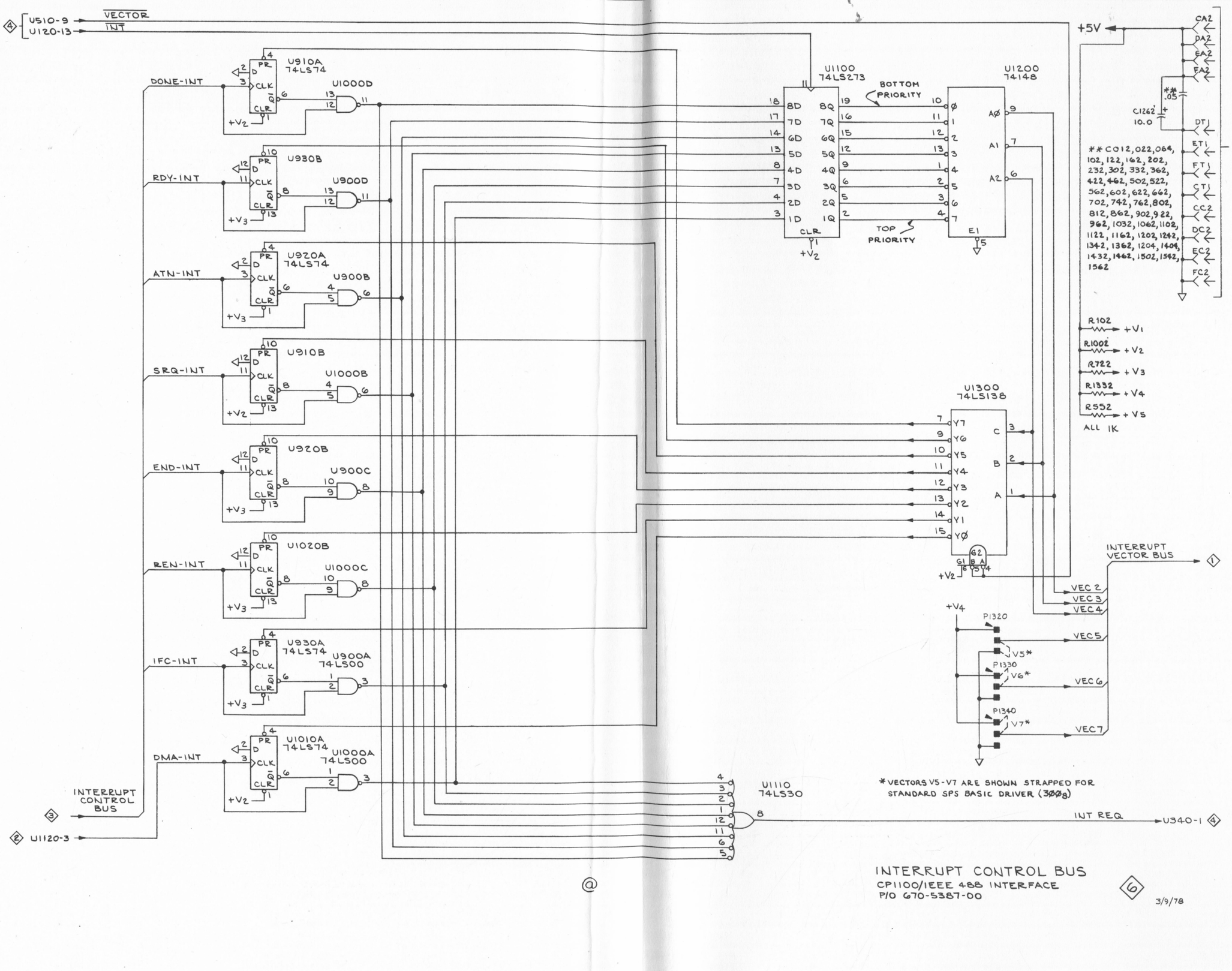
Fig. 8-6. Interface parts locator diagram.

P/O 670-5387-00

COMPONENT NUMBER	SCHEMATIC LOCATION COL	SCHEMATIC LOCATION ROW	BOARD LOCATION COL	BOARD LOCATION ROW
C012	H	2	0	0
C022	H	2	0	2
C064	H	2	0	6
C102	H	2	1	0
C122	H	2	1	2
C162	H	2	1	6
C202	H	2	2	0
C232	H	2	2	3
C302	H	2	3	0
C332	H	2	3	3
C362	H	2	3	6
C422	H	2	4	2
C462	H	2	4	6
C502	H	2	5	0
C522	H	2	5	2
C562	H	2	5	6
C602	H	2	6	0
C622	H	2	6	2
C662	H	2	6	6
C702	H	2	7	0
C742	H	2	7	4
C762	H	2	7	6
C802	H	2	8	0
C812	H	2	8	1
C862	H	2	8	6
C902	H	2	9	0
C922	H	2	9	2
C962	H	2	9	6
C1032	H	2	10	3
C1062	H	2	10	6
C1102	H	2	11	0
C1122	H	2	11	2
C1162	H	2	11	6
C1202	H	2	12	0
C1204	H	2	13	0
C1242	H	2	12	4
C1262	H	1	12	6
C1342	H	2	13	4
C1362	H	2	13	6
C1404	H	2	14	0
C1432	H	2	14	3
C1462	H	2	14	6
C1502	H	2	15	0
C1542	H	2	15	4
C1562	H	2	15	6

COMPONENT NUMBER	SCHEMATIC LOCATION COL	SCHEMATIC LOCATION ROW	BOARD LOCATION COL	BOARD LOCATION ROW
P1320	G	5	13	2
P1330	G	5	13	3
P1340	G	5	13	4
R102	H	3	1	0
R552	H	3	5	4
R722	H	3	7	2
R1002	H	3	10	0
R1332	H	3	13	3
U900A	C	5	9	0
U900B	C	2	9	0
U900C	C	4	9	0
U900D	C	2	9	0
U910A	B	1	9	1
U910B	B	3	9	1
U920A	B	2	9	2
U920B	B	4	9	2
U930A	B	5	9	3
U930B	B	2	9	3
U1000A	C	6	10	0
U1000B	C	3	10	0
U1000C	C	4	10	0
U1000D	C	1	11	0
U1010A	B	5	10	1
U1020B	B	4	10	2
U1100	F	1	11	0
U1110	F	6	11	1
U1200	G	1	12	0
U1300	G	3	13	0

COMPONENT CROSS REFERENCE TABLE



INTERRUPT CONTROL BUS
CP1100/IEEE 488 INTERFACE
P/O 670-5387-00

3/9/78

YOUR COMMENTS COUNT

The Manual Writers at Tektronix, Inc. are interested in what you think about this manual, how you use it, and changes you might like to see in future manuals. Any queries regarding this manual will be answered personally.

What did you find that was:

interesting? _____

_____

frustrating? _____

_____

helpful? _____

_____

confusing? _____

_____

Is there anything you would like to see added to or deleted from this manual? _____

_____

_____

_____

What is your major application area for this product? _____

_____

_____

_____

Have you found any interesting applications, operating hints, or software routines which you would like to share with us? _____

_____

_____

_____

_____

_____

_____

_____

_____

_____

* * * * *

Name: _____ Position: _____

Company: _____ Department: _____

Street: _____

City: _____ State: _____ Zip: _____

Fold on dotted lines and tape.
Postage will be paid by Tektronix, Inc. if mailed in U.S.A.

FIRST CLASS
PERMIT NO. 61
BEAVERTON, OREGON

BUSINESS REPLY MAIL

No postage necessary if mailed in the United States

Postage will be paid by

TEKTRONIX, INC.
P.O. Box 500
Beaverton, Oregon 97005

ATTN: SPS Manuals 94-384

MANUAL CHANGE INFORMATION

At Tektronix, we continually strive to keep up with latest electronic developments by adding circuit and component improvements to our instruments as soon as they are developed and tested.

Sometimes, due to printing and shipping requirements, we can't get these changes immediately into printed manuals. Hence, your manual may contain new change information on following pages.

A single change may affect several sections. Since the change information sheets are carried in the manual until all changes are permanently entered, some duplication may occur. If no such change pages appear following this page, your manual is correct as printed.

SERVICE NOTE

Because of the universal parts procurement problem, some electrical parts in your instrument may be different from those described in the Replaceable Electrical Parts List. The parts used will in no way alter or compromise the performance or reliability of this instrument. They are installed when necessary to ensure prompt delivery to the customer. Order replacement parts from the Replaceable Electrical Parts List.

CALIBRATION TEST EQUIPMENT REPLACEMENT

Calibration Test Equipment Chart

This chart compares TM 500 product performance to that of older Tektronix equipment. Only those characteristics where significant specification differences occur, are listed. In some cases the new instrument may not be a total functional replacement. Additional support instrumentation may be needed or a change in calibration procedure may be necessary.

Comparison of Main Characteristics

DM 501 replaces 7D13		
PG 501 replaces 107	PG 501 - Risetime less than 3.5 ns into 50 Ω.	107 - Risetime less than 3.0 ns into 50 Ω.
108	PG 501 - 5 V output pulse; 3.5 ns Risetime.	108 - 10 V output pulse; 1 ns Risetime.
111	PG 501 - Risetime less than 3.5 ns; 8 ns Pretrigger pulse delay.	111 - Risetime 0.5 ns; 30 to 250 ns Pretrigger Pulse delay.
114	PG 501 - ±5 V output.	114 - ±10 V output. Short proof output.
115	PG 501 - Does not have Paired, Burst, Gated, or Delayed pulse mode; ±5 V dc Offset. Has ±5 V output.	115 - Paired, Burst, Gated, and Delayed pulse mode; ±10 V output. Short-proof output.
PG 502 replaces 107		
108	PG 502 - 5 V output	108 - 10 V output.
111	PG 502 - Risetime less than 1 ns; 10 ns Pretrigger pulse delay.	111 - Risetime 0.5 ns; 30 to 250 ns Pretrigger pulse delay.
114	PG 502 - ±5 V output	114 - ±10 V output. Short proof output.
115	PG 502 - Does not have Paired, Burst, Gated, Delayed & Undelayed pulse mode; Has ±5 V output.	115 - Paired, Burst, Gated, Delayed & Undelayed pulse mode; ±10 V output. Short-proof output.
2101	PG 502 - Does not have Paired or Delayed pulse. Has ±5 V output.	2101 - Paired and Delayed pulse; 10 V output.
PG 506 replaces 106	PG 506 - Positive-going trigger output signal at least 1 V; High Amplitude output, 60 V.	106 - Positive and Negative-going trigger output signal, 50 ns and 1 V; High Amplitude output, 100 V.
067-0502-01	PG 506 - Does not have chopped feature.	0502-01 - Comparator output can be alternately chopped to a reference voltage.
SG 503 replaces 190, 190A, 190B	SG 503 - Amplitude range 5 mV to 5.5 V p-p.	190B - Amplitude range 40 mV to 10 V p-p.
191	SG 503 - Frequency range 250 kHz to 250 MHz.	191 - Frequency range 350 kHz to 100 MHz.
067-0532-01	SG 503 - Frequency range 250 kHz to 250 MHz.	0532-01 - Frequency range 65 MHz to 500 MHz.
TG 501 replaces 180, 180A	TG 501 - Marker outputs, 5 sec to 1 ns. Sinewave available at 5, 2, and 1 ns. Trigger output - slaved to marker output from 5 sec through 100 ns. One time-mark can be generated at a time.	180A - Marker outputs, 5 sec to 1 μs. Sinewave available at 20, 10, and 2 ns. Trigger pulses 1, 10, 100 Hz; 1, 10, and 100 kHz. Multiple time-marks can be generated simultaneously.
181	TG 501 - Marker outputs, 5 sec to 1 ns. Sinewave available at 5, 2, and 1 ns.	181 - Marker outputs, 1, 10, 100, 1000, and 10,000 μs, plus 10 ns sinewave.
184	TG 501 - Marker outputs, 5 sec to 1 ns. Sinewave available at 5, 2, and 1 ns. Trigger output - slaved to marker output from 5 sec through 100 ns. One time-mark can be generated at a time.	184 - Marker outputs, 5 sec to 2 ns. Sinewave available at 50, 20, 10, 5, and 2 ns. Separate trigger pulses of 1 and .1 sec; 10, 1, and .1 ms; 10 and 1 μs. Marker amplifier provides positive or negative time marks of 25 V min. Marker intervals of 1 and .1 sec; 10, 1, and .1 ms; 10 and 1 μs.
2901	TG 501 - Marker outputs, 5 sec to 1 ns. Sinewave available at 5, 2, and 1 ns. Trigger output - slaved to marker output from 5 sec through 100 ns. One time-mark can be generated at a time.	2901 - Marker outputs, 5 sec to 0.1 μs. Sinewave available to 50, 10, and 5 ns. Separate trigger pulses, from 5 sec to 0.1 μs. Multiple time-marks can be generated simultaneously.

NOTE: All TM 500 generator outputs are short-proof. All TM 500 plug-in instruments require TM 500-Series Power Module.

REV. A, OCT 1975

TEKTRONIX®
committed to technical excellence

MANUAL CHANGE INFORMATION

PRODUCT 021-0230-00
070-2151-00

CHANGE REFERENCE C1/1078

DATE 10-9-78

CHANGE:	DESCRIPTION

CP1100/IEEE 488 INTERFACE

EFF SN B010100 (Pilot Changes #1, #5 & #7)

ELECTRICAL PARTS LIST AND SCHEMATIC CHANGES

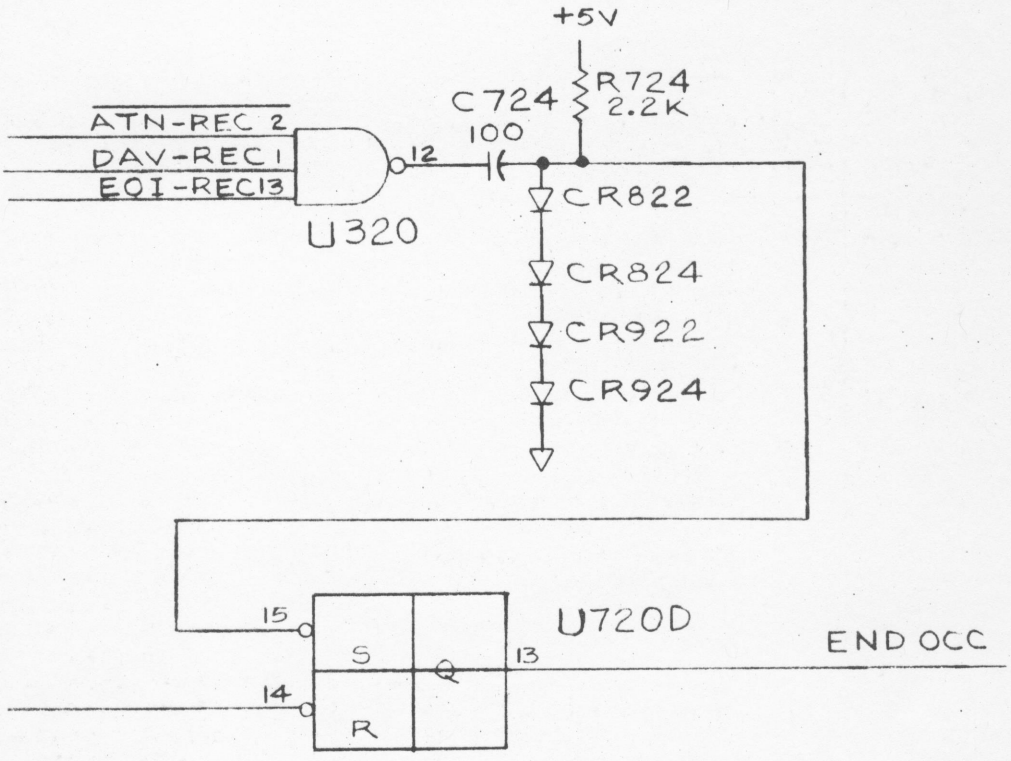
CHANGE TO:

C412	285-1050-00	CAP.,FXD,PLASTIC:0.1UF,1%,200V
R412	321-0269-00	RES.,FXD,FILM:6.19K OHM,1%,0.125W
R402	311-1285-00	RES.,VAR,NONWIR:25K OHM,10%,0.50W

The above parts are shown on diagram 3 BSR/BCR/ICR/DATA BUS.

DIAGRAM ◇3◇ BSR/BCR/ICR/DATA BUS - Partial

Add C724, R724, CR822, CR824, CR922 and CR924 as shown below.



TEKTRONIX®
committed to technical excellence

MANUAL CHANGE INFORMATION

PRODUCT 021-0230-00
070-2151-00

CHANGE REFERENCE M34569

DATE 8-15-78

CHANGE:	DESCRIPTION

EFF 901-03-A

ELECTRICAL PARTS LIST AND SCHEMATIC CHANGE

CHANGE TO:

R412 311-1268-00 RES.,VAR,NONWIR:10K OHM,10%,0.50W

The above component is shown on diagram 3 BSR/BCR/ICR/DATA BUS.

Tektronix®
COMMITTED TO EXCELLENCE

MANUAL CHANGE INFORMATION

Change Reference: M34959 REV. Date: 2-15-79

Product: CP1100/IEEE 488 INTERFACE 021-0230-00 070-2151-00

CHANGE	DESCRIPTION

EFF:ID# 903-05-B

SERVICING CHANGES

Page 5-5

CHANGE the title of the DMA transmission rate calibration to:

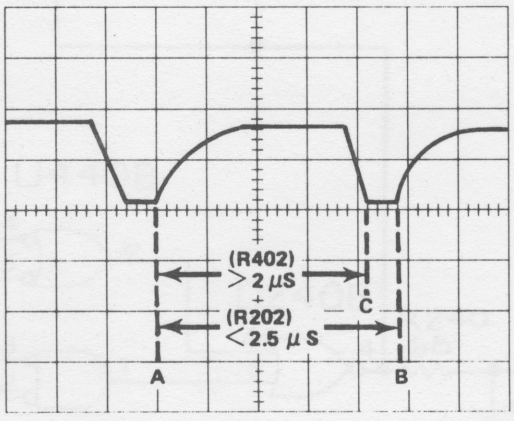
Calibration for DMA Transmission Rate (R402, ID# 903-04-A and Below)

ADD the following procedure and waveform between the Calibration and Cabling information:

Calibration for DMA Transmission Rate (R202-R402, ID# 903-05-B and Up)

The adjustment of R202 and R402 requires that the interface be operated in the DMA hog mode talking to a maximum data rate listener on the bus (data transfer rate of approximately 400 kHz).

With the interface operating under the above stated conditions, set up an oscilloscope (2 MHz minimum bandwidth) for a sweep rate of 0.5 microsecond/division and apply the DAV bus signal to the vertical input. Adjust R202 for just less than 2.5 µs between points A and B (see waveform). Adjust R402 for >2 µs between A and C. Check the time between A and B for <2.5 µs. If necessary, re-adjust R202 for <2.5 µs between A and B.



CHANGE	DESCRIPTION

ELECTRICAL PARTS LIST AND SCHEMATIC CHANGES

CHANGE TO:

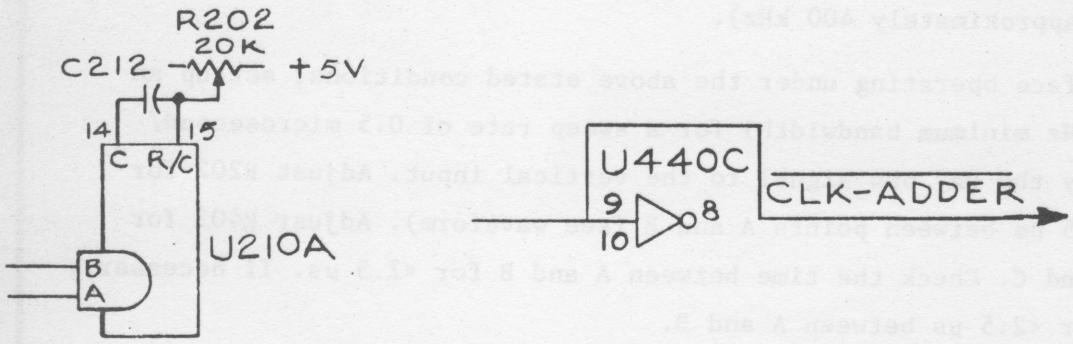
A1	670-5387-01	CKT BOARD ASSY:11/IEEE 488
R202	311-0644-01	RES.,VAR,NONWIR:20K OHM,0.50W
U300	156-0331-01	MICROCIRCUIT,DI:DUAL D-TYPE FLIP-FLOP,74S74

Diagram 2 BAR/CTR/ISR CLK-ADDR input line notation from U440-8 to U440-9,10

U300 is shown on diagram 3 BSR/BCR/ICR/DATA BUS

DIAGRAM 4

INTERRUPT & DMA STATE MACHINE - Partial



TEKTRONIX®
committed to technical excellence

MANUAL CHANGE INFORMATION

PRODUCT CP1100/IEEE 488
INTERFACE 021-0230-00
CHANGE REFERENCE M35671
DATE 12-6-78

CHANGE:	DESCRIPTION

070-2151-00 EFF ID# 907-01-C

ELECTRICAL PARTS LIST AND SCHEMATIC CHANGES

CHANGE TO:

A1	670-5387-02	CKT BOARD ASSY:11/IEEE 488
U240	156-0043-01	MICROCIRCUIT,DI:QUAD 2-INPUT NOR GATE,7402

ADD:

C140	283-0197-00	CAP.,FXD,CER DI:470PF,5%,50V
C240	283-0197-00	CAP.,FXD,CER DI:470PF,5%,50V
R140	317-0680-00	RES.,FXD,CMPSN:68 OHM,5%,0.125W
R240	317-0680-00	RES.,FXD,CMPSN:68 OHM,5%,0.125W

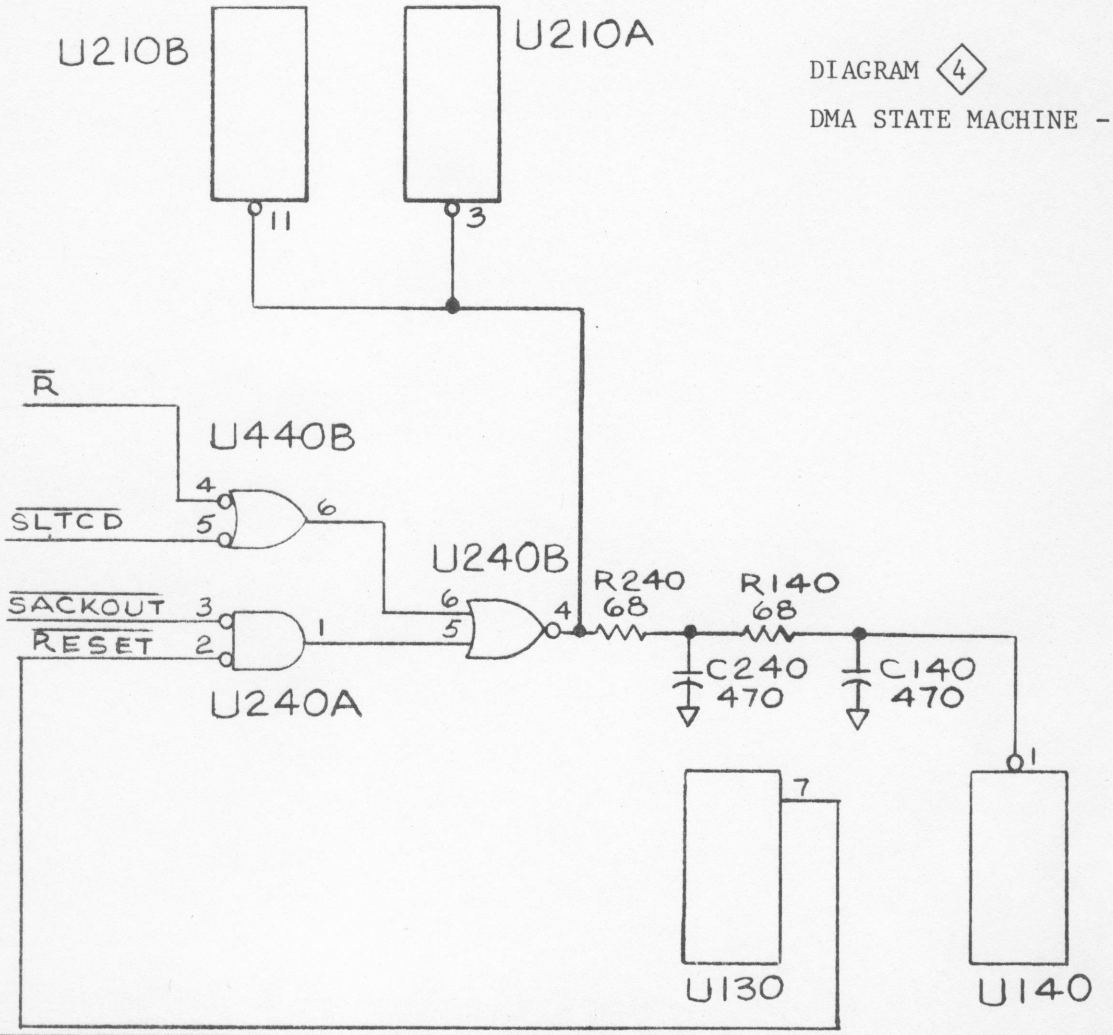



DIAGRAM 4
DMA STATE MACHINE - Partial