INSTALLATION AND OPERATION

990 CRU/TILINE Expansion

Part No. 2272075-9701 *A
March 1984

TEXAS INSTRUMENTS

LIST OF EFFECTIVE PAGES

INSERT LATEST CHANGED PAGES AND DISCARD SUPERSEDED PAGES

Note: The changes in the text are indicated by a change number at the bottom of the page and a vertical bar in the outer margin of the changed page. A change number at the bottom of the page but no change bar indicates either a deletion or a page layout change.

990 CRU/TILINE Expansion Installation and Operation (2272075-9701)

Original Issue . . . . . June 1983
Change 1 . . . . . March 1984

Total number of pages in this publication is 164 consisting of the following:

PAGE NO.	CHANGE NO.	PAGE NO	CHANGE NO.	PAGE NO.	CHANGE NO.
Cover	1	1-23 - 1-27	0	3-29 - 3-45	0
Effective Pages	1	1-28	1	3-46 - 3-47	1
iii/iv	1	1-29 - 1-35	0	3-48 - 3-54	0
v - vi	0	1-36 - 1-38	1	A-1 - A-8	0
vii - viii	1	2-1 - 2-9	0	A-9/A-10	1
ix/x	0	2-10	1	B-1 - B-6	1
1-1 - 1-2	0	2-11 - 2-19	0	C-1 - C-14	1
1-3	1	2-20	1	Index-1	0
1-4 - 1-8	0	3-1 - 3-24	0	Index-2 - Index-6	1
1-9	1	3-25/3-26	1	User's Response	1
1-10 - 1-17	0	3-27	0	Business Reply	1
1-18	1	3-28	1	Inside Cover	1
1-19/1-20	0	3-28A/3-28B	1	Cover	1
1-21/1-22	1				

The computers, as well as the programs that TI has created to use with them, are tools that can help people better manage the information used in their business; but tools—including TI computers—cannot replace sound judgment nor make the manager's business decisions.

Consequently, TI cannot warrant that its systems are suitable for any specific customer application. The manager must rely on judgment of what is best for his or her business.

Preface

The *990 CRU/TILINE Expansion — Installation and Operation* manual provides an introduction to the 990 input/output data buses and describes how to extend these buses into one or more additional chassis. This manual applies only to the low radiation (LR) expansions that meet VDE 0871 and FCC Part 15 limits for radio-frequency interference and electromagnetic interference (RFI/EMI).

Information in this manual is divided into three sections and one appendix:

Section

1 Introduction — Features, specifications, configurations, and physical descriptions.

2 Installation — Site requirements, installation, cabling, and initial power-up procedures.

3 Nonstandard and OEM Installation Data — Technical data for installations not part of a standard Texas Instruments system, or those including customer-designed devices.

Appendix

A Multiprocessor Interface — Technical description of the multiprocessor interrupt protocol.

B Configuring the Multiprocessor Interface — Technical description of multiprocessor interface configuration.

C Examples of TILINE Expansion Configurations.

The following manuals are related to CRU and TILINE expansion:

Title	Part Number
Model 990A13 Chassis, General Description	2308774-9701
Model 990A13 Chassis, Field Theory and Maintenance	2308775-9701
Model 990A13 Chassis, Depot Theory and Maintenance	2308776-9701

Contents

3 — Nonstandard and OEM Configuration Data

Appendixes

Index

Illustrations

Tables

1

Introduction to CRU/TILINE Expansion

1.1 GENERAL

This manual describes the hardware that expands a Business System computer from the 990A13 computer chassis into one or more additional chassis. Separate hardware extends the TILINE* high-speed parallel bus and the communications register unit (CRU) bus into an expansion chassis. An expansion chassis can include either one or both of these buses.

1.2 REASONS FOR EXPANSION

A Model 990A13 Chassis provides physical slot space, operating power, input/output (I/O) signal connections, and interrupt connections for up to 13 full-sized logic boards. With a full-sized logic board in slot 1, slots 2 through 13 can support a maximum of 24 half-sized logic boards. Most chassis include some combination of full-sized and half-sized logic boards.

Some computer systems outgrow a single computer chassis because of the number of circuit boards. All of the logic boards necessary to handle a large number of communications devices, video display terminals, and mass storage devices may not physically fit in a single chassis. Also, with special (non-TI) circuit boards, it is possible to exceed the power supply ratings of a single chassis power supply.

A TILINE or CRU expansion chassis kit provides a growth path to assure that power or physical slot space does not limit the size and capability of your computing system. These kits extend interrupt capability and the input/output buses to one or more additional chassis. An expansion chassis can include:

- TILINE and interrupts
- CRU and interrupts
- TILINE, CRU, and interrupts

This manual describes the hardware features, capabilities, and typical configurations for expansion. If you are planning to use standard Texas Instruments operating system software, you must check the software documentation to verify compatibility of the hardware and software. If you purchased your expansion kit as part of a Business System package, Texas Instruments takes care of hardware/software coordination.

* TILINE is a registered trademark of Texas Instruments Incorporated.

CRU expansion is limited to computing systems that operate with a single central processor unit (CPU). TILINE expansion allows independent central processors to share resources such as memory and mass storage devices. TILINE expansion hardware provides facilities for true multiprocessor operation of central processors in separate chassis. Multiprocessing requires special operating system software.

1.3 CRU INPUT/OUTPUT SYSTEM INTRODUCTION

All Business Systems and TMS 9900/99000-based central processors include the CRU bus. The CRU is a general-purpose serial input/output (I/O) bus. Every bit on the CRU bus has a unique address, up to a maximum of 4096 input bits and 4096 output bits. The CRU bus, which connects to all slots in the chassis, is the main control and I/O data path for low-speed and medium-speed devices including:

- Printers
- Video display terminals (excluding Model 931)
- Modems and other communications equipment

The central processor instructions that drive the CRU bus can address each CRU bit individually, or in fields of 1 to 16 bits. These CRU instructions can set, reset, or check any individual CRU bit, transfer CRU data bits into memory, or transfer data bits from memory to the CRU bus. The CRU instructions are:

- SBO — Set bit to logic one
- SBZ — Set bit to logic zero
- TB — Test bit
- STCR — Store communications register
- LDCR — Load communications register

All of the CRU instructions depend on bits 3 through 14 of a CRU base address stored in central processor workspace register 12 (WR12). A CRU instruction word includes a signed displacement that adds to CRU base address bits 3 through 14. The sum is the CRU address output on CRUBIT3 through CRUBIT15.

Section 3 describes the CRU instructions and CRU address development in greater detail. The important thing to remember is that the last bit of the CRU base address in WR12 is ignored. If the displacement is zero, a CRU base address of $02C0_{16}$ results in an address of 0160_{16} on address lines CRUBIT3 through CRUBIT15.

The CRU bus consists of 12 address lines (CRUBIT4 through CRUBIT15), one serial output line (CRUBITOUT), one serial input line (CRUBITIN), and an output clock line (STORECLK–). In a computer chassis, the central processor is the source of the CRU bus. Figure 1-1 is a simplified schematic of the CRU bus connections in a computer chassis.

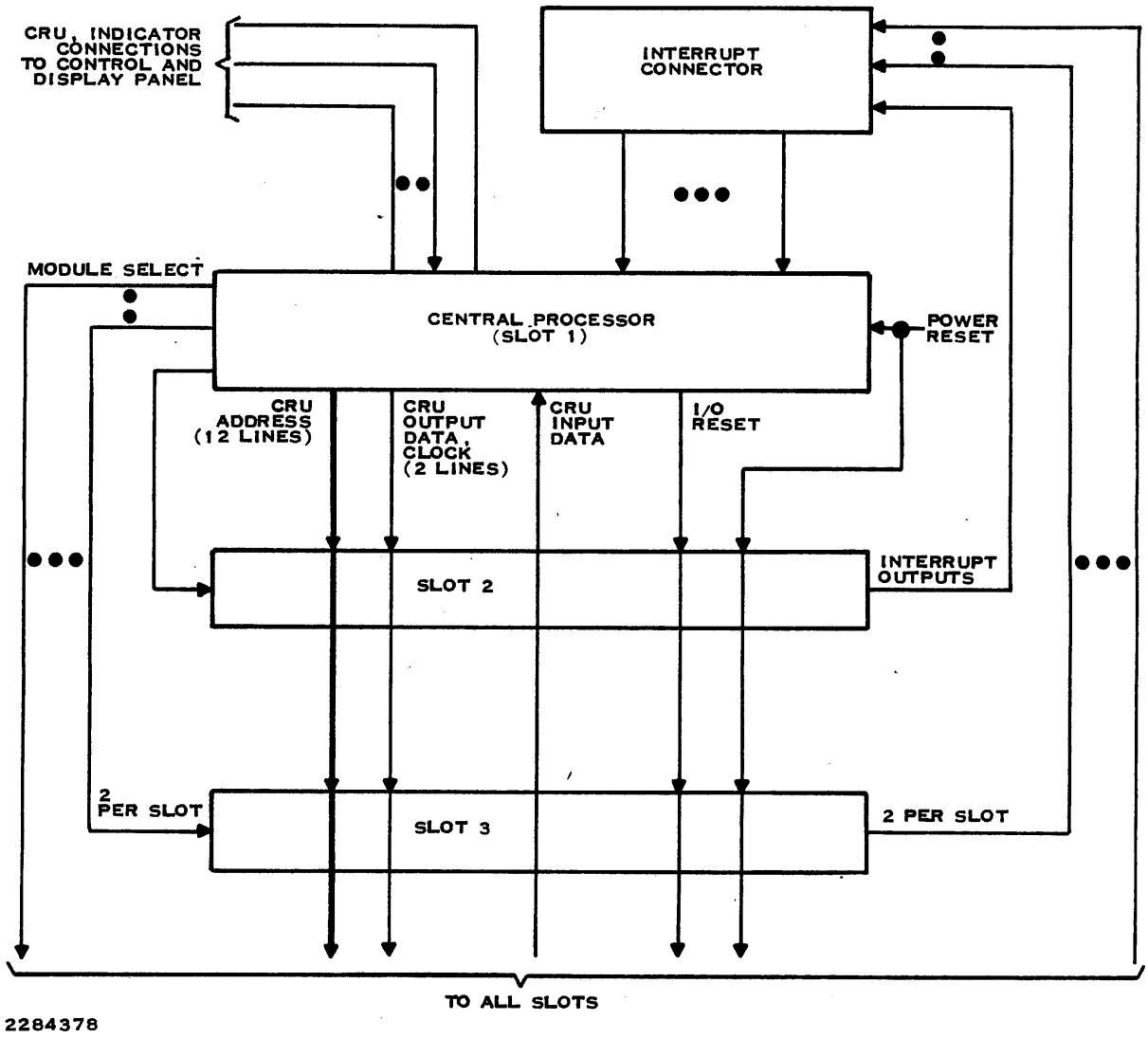


Figure 1-1. Simplified Schematic of CRU Connections in a Computer Chassis

Each CRU board has a block of bit addresses that it must recognize for read and write operations. To reduce the amount of decoding logic on each board, address decoding for most CRU boards is split between the central processor or CRU buffer in slot 1, and the CRU board. Module select signals from slot 1 assign a block of 16 CRU addresses to each half-slot connector in slots 2 and above.

The module select wiring has four practical effects:

- CRU boards can be built in half-slot sizes. With a center card guide, two half-sized boards can share a slot.
- Physical location in the chassis determines CRU addresses. A half-sized board decodes only the least significant four bits of the CRU address.
- Any CRU board that needs 16 to 32 bits ties up a full chassis slot, even if the board is physically half-sized. Such a board must go into the right (P2) half-slot, which has two module select inputs.
- Any board that needs more than 32 CRU bits must have full CRU address decoding on the board.

Figure 1-2 shows the CRU base address assigned to each half-slot in a 990A13 chassis. The addresses on the figure are as they appear in workspace register 12 of the central processor. Note that the lowest CRU base address (000) appears at the P2 side of the highest-numbered slot.

CRU expansion hardware and standard operating system software allow you to expand the CRU bus into a maximum of seven additional chassis. The main chassis that contains the central processor is always chassis 0. Expansion chassis numbers range from 1 to 7.

For a CRU expansion chassis, you must add (by hexadecimal addition) an offset to every CRU base address shown in Figure 1-2. These offsets assure that every half-slot of every chassis responds to a unique block of CRU addresses. The offsets are:

Expansion Chassis No.	CRU Offset
1	0400
2	0800
3	0C00
4	1000
5	1400
6	1800
7	1C00

As an example, slot 10, connector P1 in the first expansion chassis has a CRU base address of $0400_{16} + 00E0_{16} = 04E0_{16}$.

	P1 (CHASSIS FRONT)			P2 (CHASSIS REAR)		
SLOT NUMBER	FIXED CRU BASE ADDRESS	CIRCUIT BOARD	INTER-RUPT LEVEL	FIXED CRU BASE ADDRESS	CIRCUIT BOARD	INTER-RUPT LEVEL
1	N/A		N/A	N/A		N/A
2	02E0			02C0		
3	02A0			0280		
4	0260			0240		
5	0220			0200		
6	01E0			01C0		
7	01A0			0180		
8	0160			0140		
9	0120			0100		
10	00E0			00C0		
11	00A0			0080		
12	0060			0040		
13	0020			0000		

13-SLOT CHASSIS

2283011

Figure 1-2. CRU Base Address Assignments by Slot Location

1.4 CRU EXPANSION

A CRU expansion system requires:

- CRU Expander LR, part number 2230730-0001
- CRU Buffer LR, part number 2230725-0001
- Shielded, 40-pin, twisted-pair cable (5 meters), part number 2308633-0003
- Model 990A13 Expansion Chassis, part number 2309019-xxxx
- Type 13-1 Interrupt Board, part number 2310380-0001

These components replace the CRU expansion components phased out in 1983. The new low-radiation CRU expansion components can replace the older components without software changes.

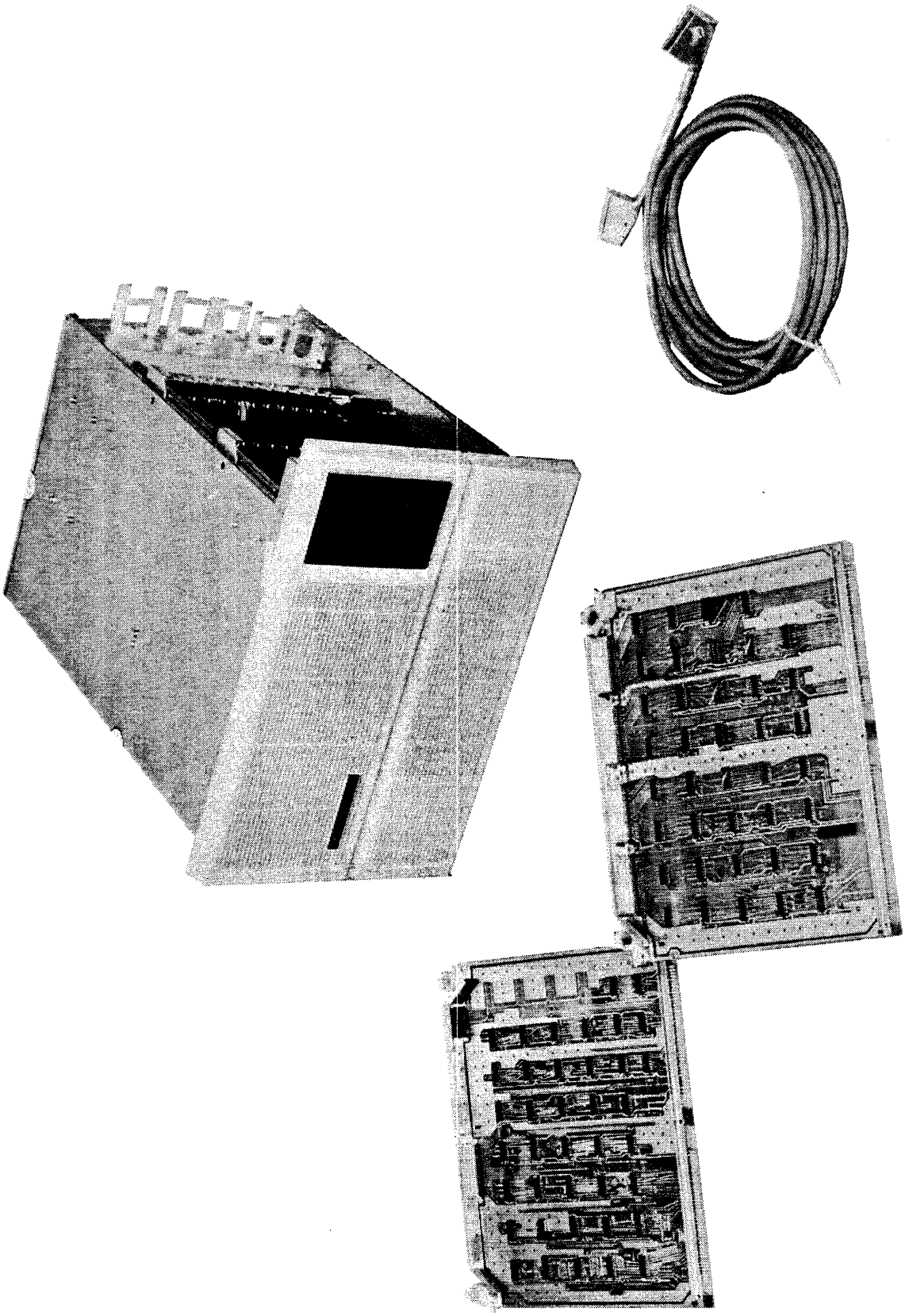
NOTE

Components of the new CRU expansion system cannot be intermixed with components of the older expansion system. The new low-radiation, self-grounding, shielded cable connectors do not mate with the older CRU expander and buffer boards. If you have an older expansion system and and wish to add an additional buffer, contact Texas Instruments for advice.

Table 1-1 lists the CRU expansion kits that are currently available, and Figure 1-3 shows the components of a typical expansion kit.

Table 1-1. CRU Expansion Kits

Part Number	Summary
2310919-0013	Master CRU expansion kit LR, including chassis with NEMA 5-15 power plug (100/120 Vac)
2310919-0014	Add-on CRU expansion kit LR (no expander), with same chassis as -0013
2310919-0015	OEM master CRU expansion kit LR, including OEM chassis with US NEMA 5-15 power plug
2310919-0016	OEM add-on CRU expansion kit LR (no expander), with same chassis as -0015
2310919-0017	Master CRU expansion kit LR, including chassis with CEE(7)VII power plug (220 Vac)
2310919-0018	Add-on CRU expansion kit LR (no expander), with same chassis as -0017
2310919-0019	OEM master CRU expansion kit LR, including OEM chassis with CEE(7)VII power plug (220 Vac)
2310919-0020	OEM add-on CRU expansion kit LR (no expander), with same chassis as -0019
2310919-0021	Master CRU expansion kit LR, including chassis with BSI 1363 power plug (240 Vac)
2310919-0022	Add-on CRU expansion kit LR (no expander), with same chassis as -0021
2310919-0023	OEM master CRU expansion kit LR, including chassis with BSI 1363 power plug (240 Vac)
2310919-0024	OEM add-on CRU expansion kit LR (no expander), with same chassis as -0023
2310919-0025	CRU expansion interface kit LR (no chassis), with expander, buffer, cable, and documentation



2284402

Figure 1-3. CRU Expansion Kit

A single full-sized CRU expander board in the main chassis extends the CRU bus to a maximum of four expansion chassis. Two expander boards extend the CRU bus to a maximum of seven chassis. Each of these chassis must have a CRU buffer in slot 1. The CRU buffer replaces the central processor as the source of the CRU bus. Figure 1-4 is a simplified schematic that shows the CRU connections in an expansion chassis.

A CRU buffer partially decodes the incoming CRU address and distributes the module select signals to individual half-slot connectors. Also, the buffer scans interrupts from the chassis interrupt jumper connector. Slot 1 is the only slot that has wiring to scan interrupts and distribute module select signals.

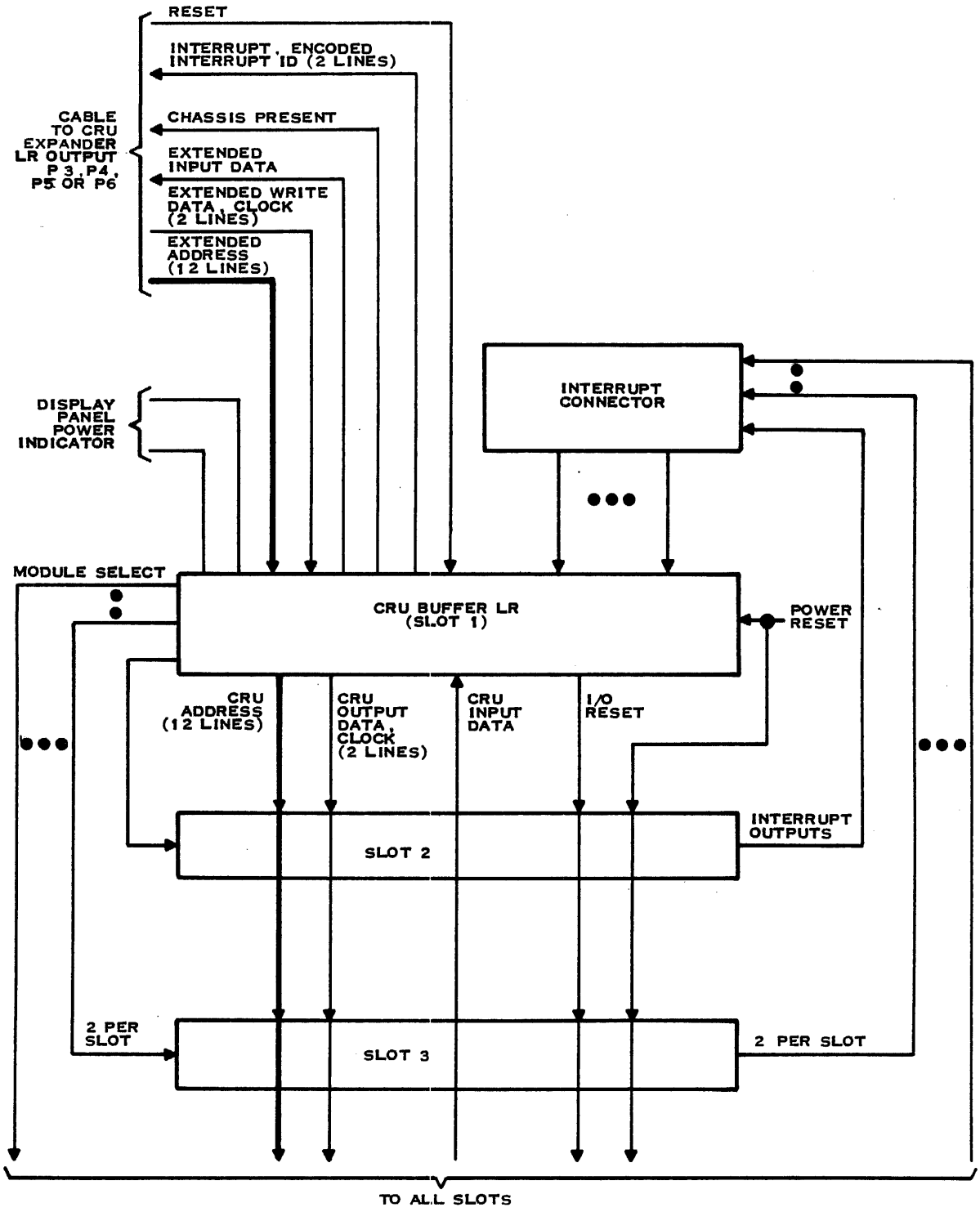
An expansion chassis has a display panel, part number 2309153-0001, instead of the control and display panel that appears on a computer chassis. The display panel has a single green POWER indicator. There are no overlays, controls, switches, or test connectors on the display panel. A plain overlay with the POWER indicator label is part of the upper trim panel. Otherwise, an expansion chassis is identical to a computer chassis. For additional information on the expansion chassis, refer to the *Model 990A13 Chassis, General Description* manual, part number 2308774-9701.

1.4.1 CRU Expander LR Physical Description

Figure 1-5 is a photograph of the CRU expander logic board. Notice the grounded shield-stiffener along the outer edge and the large ring-style ejector/injector tabs. These features identify the CRU expander as a shielded, low-radiation logic board. When a chassis is filled with shielded boards, the shield-stiffeners form a complete bulkhead that prevents electromagnetic interference (EMI) radiation.

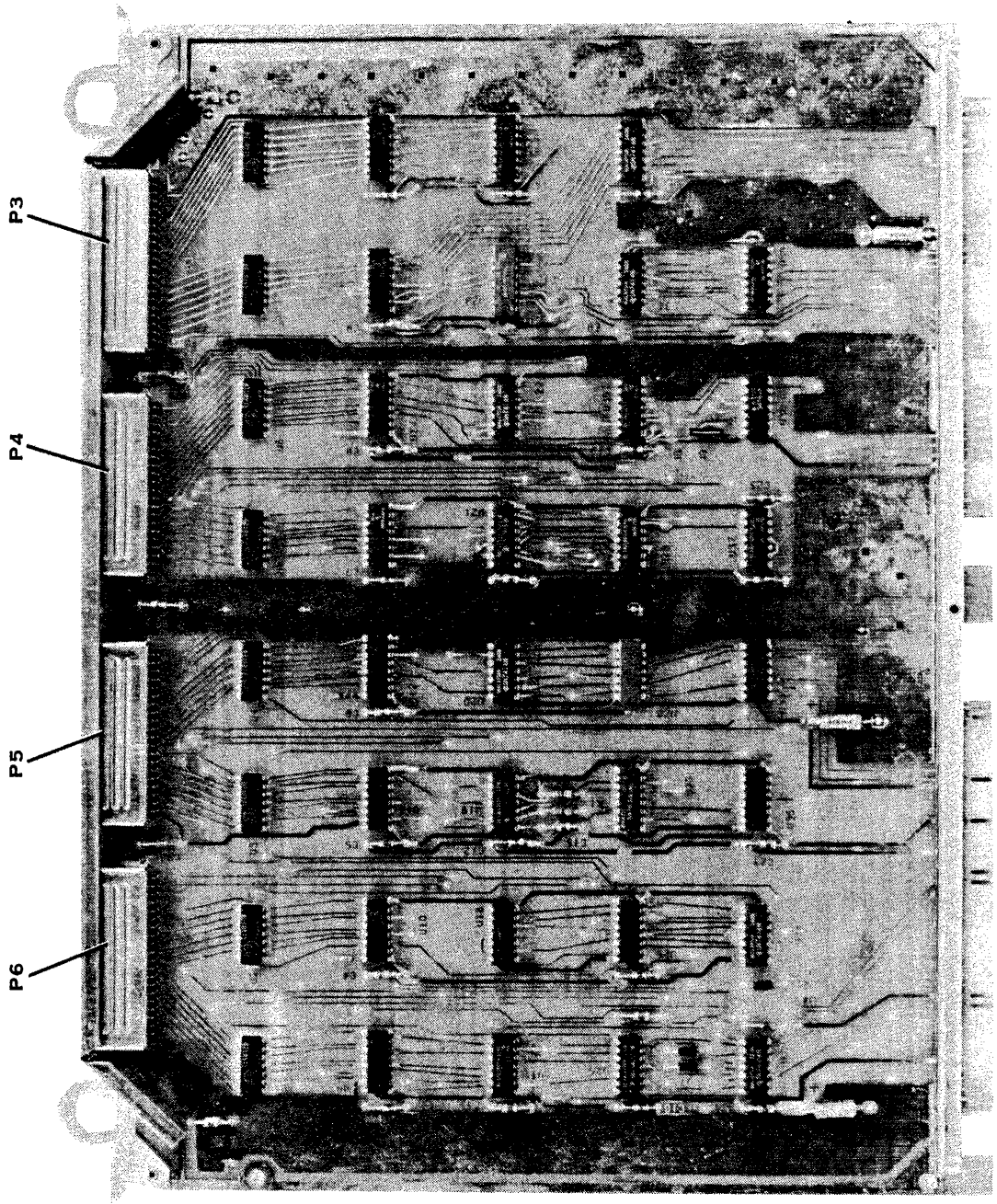
Four connectors, P3 through P6, penetrate the shield-stiffener. Additional shielding covers the connector bodies. The cable connectors that mate with P3 through P6 feature a metal shell that connects to chassis ground when the connector is inserted.

A single shielded, twisted-pair cable connects the CRU expander in the main chassis to a CRU buffer in the expansion chassis. The four connectors on the outer edge allow the CRU expander to serve four expansion chassis.



2284379

Figure 1-4. Simplified Schematic of CRU Connections in an Expansion Chassis



2284403

Figure 1-5. CRU Expander LR

1.4.2 CRU Expander LR Jumper Options

Table 1-2 lists the jumper options for a CRU expander. There are no option switches or address switches.

Table 1-2. CRU Expander LR Jumper Schedule

Jumpers	Function
J1 J2 X X	Select reset source TILINE power reset, TLPRES– TILINE I/O reset, TLIORES (standard jumper setting)
J3 J4 J5 X X X	Interrupt scanner enable Enable interrupt address decode jumpers J6–J9 (standard jumper setting) Module select A (MODSELA–) to interrupt scanner Module select B (MODSELB–) to interrupt scanner
J6 J7 J8 J9 X X X X X X X X	Interrupt address decode jumpers Chassis 1–4, CRU address F80[1] (workspace register 12 = 1F00) Chassis 5–7, CRU address F90[1] (workspace register 12 = 1F20) Chassis 9–12, CRU address FA0[2] (workspace register 12 = 1F40) Chassis 13–15, CRU address FB0[2] (workspace register 12 = 1F60)

Notes:

[1] Physical address on lines CRUBIT4–15 is one-half of the address in workspace register 12.

[2] Nonstandard connection not supported by software.

1.4.3 CRU Expander LR Specifications

Table 1-3 summarizes the specifications of the CRU expander.

Table 1-3. CRU Expander LR (2230730) Specifications

CRU Expansion Outputs	Four per expander
Data transfer type	Synchronous, serial bit-addressable data transfers, 1 to 16 bits per command
Maximum data transfer rate:	
Output	Four million bits per second
Input	Two million bits per second
Interrupt provisions	One normal interrupt with encoded interrupt vector One direct (nonstandard) interrupt
Reset features	Optional connection to reset all expansion chassis on main chassis power reset or I/O reset
Chassis space requirement	Any full slot except slot 1
Dc power requirements	+5.0 Vdc ±3% 0.9 A typical 1.4 A maximum (from chassis power supply)
Heat load	4.5 watts typical 7.2 watts maximum
Electromagnetic interference	Meets RFI/EMI limits of VDE 0871 (level A) and FCC docket 20780 (class A) when installed in a low-radiation chassis
Environmental limits:	
Temperature:	
Operating	0° to 65°C (32° to 149°F)
Nonoperating	−40 to 70°C (−40° to 158°F)
Humidity:	
Operating	0 to 95% noncondensing
Nonoperating	0 to 95% noncondensing

1.4.4 CRU Buffer LR Physical Description

Figure 1-6 is a photograph of the CRU buffer logic board. The buffer has two connectors on the shield-stiffener. P3, a 40-pin connector, mates with the cable from one of the four CRU expander connectors. The small 20-pin connector at the right side drives the display panel POWER indicator.

A single shielded, twisted-pair cable connects the CRU expander in the main chassis to a CRU buffer in the expansion chassis. The four connectors on the outer edge allow the CRU expander to serve four expansion chassis.

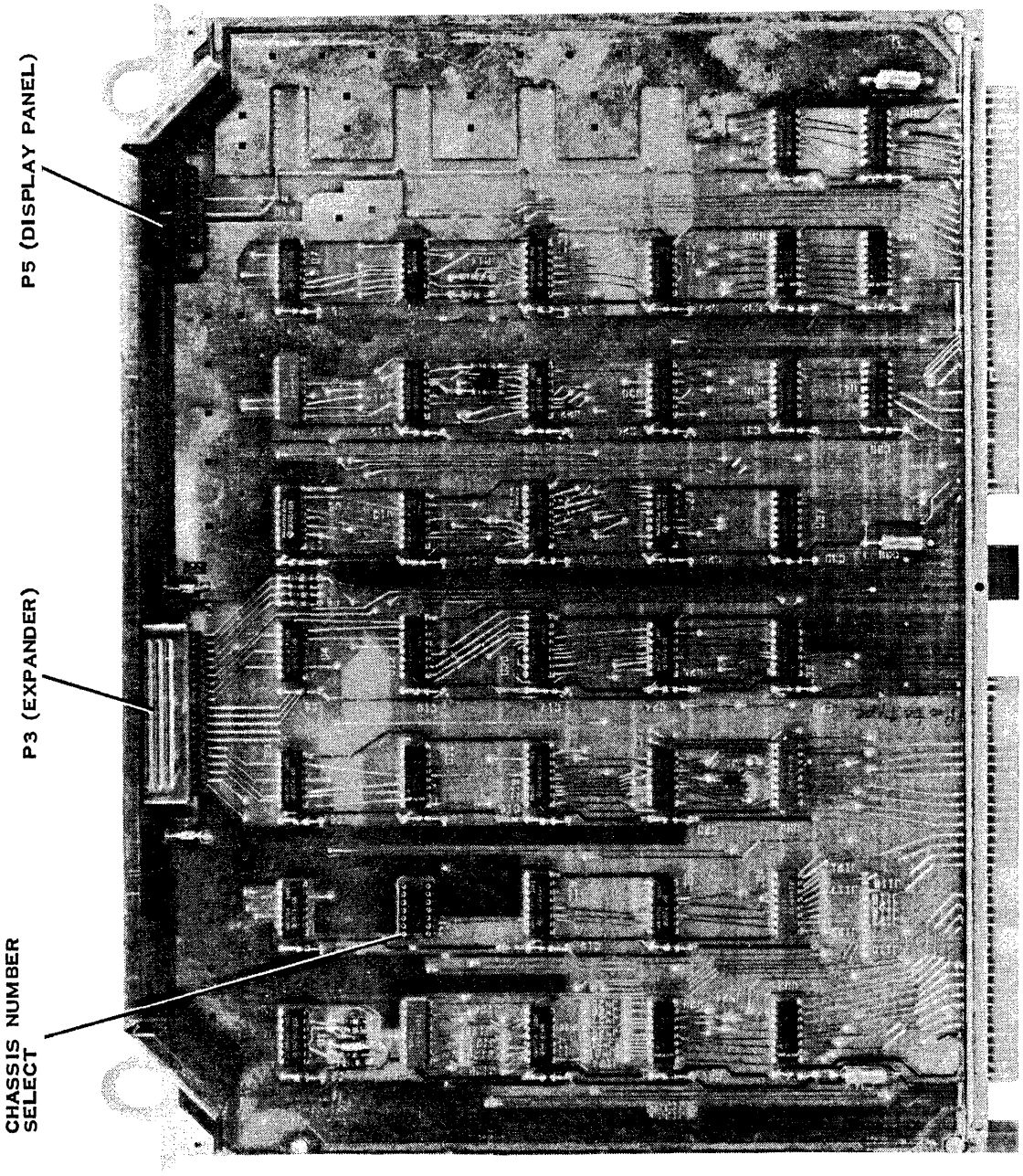


Figure 1-6. CRU Buffer LR

1.4.5 CRU Buffer LR Jumper Options

Table 1-4 lists the jumper options for a CRU buffer. There are no option switches.

Table 1-4. CRU Buffer LR Jumper Schedule

Jumpers	Function
J1 J2 JI10	Interrupt scanner control
	Scan interrupts (standard jumper setting)
X	Direct interrupt to expander
X	Enable interrupt level 0 on TLADR19- line*
JI16 through JI31	Accept interrupt levels 16 through 31*
U10-1 to U10-7	Chassis number select 1 through 7 (Removable Jumpers on IC socket XU10)

Note:

* Nonstandard connections are not supported by software.

1.4.6 CRU Buffer LR Specifications

Table 1-5 summarizes the specifications of the CRU buffer.

Table 1-5. CRU Buffer LR (2230725) Specifications

Data transfer type	Synchronous, serial bit-addressable data transfers, 1 to 16 bits per command
Maximum data transfer rate:	
From expander	Four million bits per second
To expander	Two million bits per second
Interrupt provisions	Interrupt priority encoder samples 16 interrupt level inputs (jumper options to 32 levels) One direct (nonstandard) interrupt
Reset features	Reset on command from expander and from local chassis power reset
Chassis space requirement	Slot 1 of expansion chassis (1 buffer maximum per chassis)
Dc power requirements	+ 5.0 Vdc ± 3% 0.9 A typical 1.4 A maximum (from chassis power supply)
Heat load	4.5 watts typical 7.2 watts maximum
Electromagnetic interference	Meets RFI/EMI limits of VDE 0871 (level A) and FCC docket 20780 (class A) when installed in a low-radiation chassis
Environmental limits:	
Temperature:	
Operating	0° to 65°C (32° to 149°F)
Nonoperating	– 40 to 70°C (– 40° to 158°F)
Humidity:	
Operating	0 to 95% noncondensing
Nonoperating	0 to 95% noncondensing

1.4.7 Cabling Configurations for CRU Expansion

A minimum CRU expansion adds 11 CRU slots, because the CRU expander fills a slot in the computer chassis and the CRU buffer fills slot 1 of the expansion chassis. Figure 1-7 shows the cabling for a minimum expansion. Notice that the chassis number select jumper on CRU buffer socket XU10 is in the chassis 1 position.

Figure 1-8 shows the cabling for four expansion chassis, the maximum allowable for a single CRU expander. By convention, CRU expander connector assignments are:

Connector	Chassis
P3	1 or 5
P4	2 or 6
P5	3 or 7
P6	4 (only)

Figure 1-9 shows the cabling for the largest standard expansion. This system adds a net of 82 available slots after subtracting the CRU expander and buffer slots.

NOTE

The second highest CRU base address, >1EC0 (which corresponds to the second highest CRU base address for expansion chassis number 7), cannot be used as a base address when the main chassis CPU is a 990/10A. The 990/10A defines the CRU address range from >1EC4 to >1ECE as a privileged area for use with external instructions only. This restriction does not affect the highest CRU base address, >1EE0, which is still available for use.

A value preceded by a right angle bracket (>) indicates a hexadecimal value.

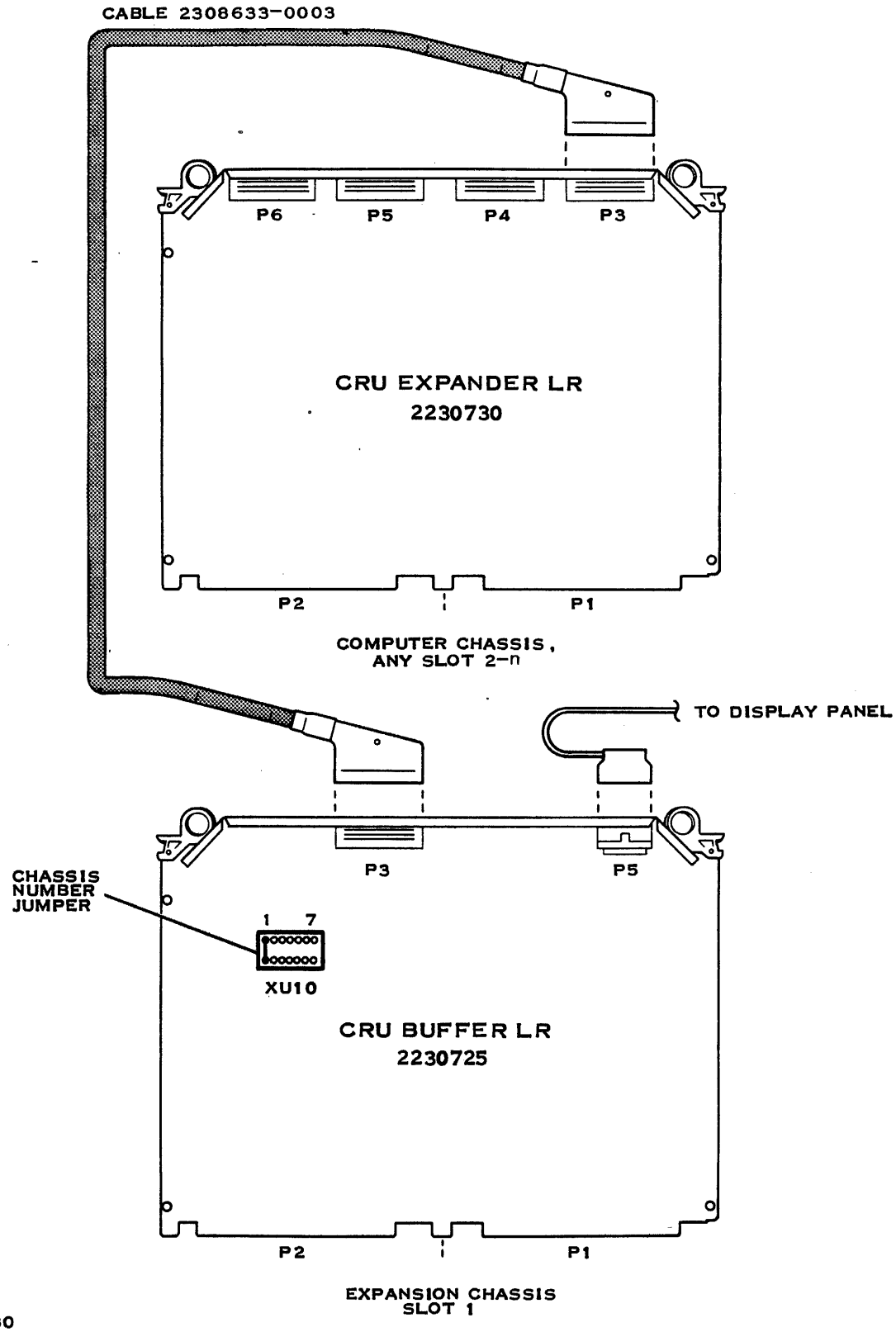


Figure 1-7. Cabling — Minimum CRU Expansion

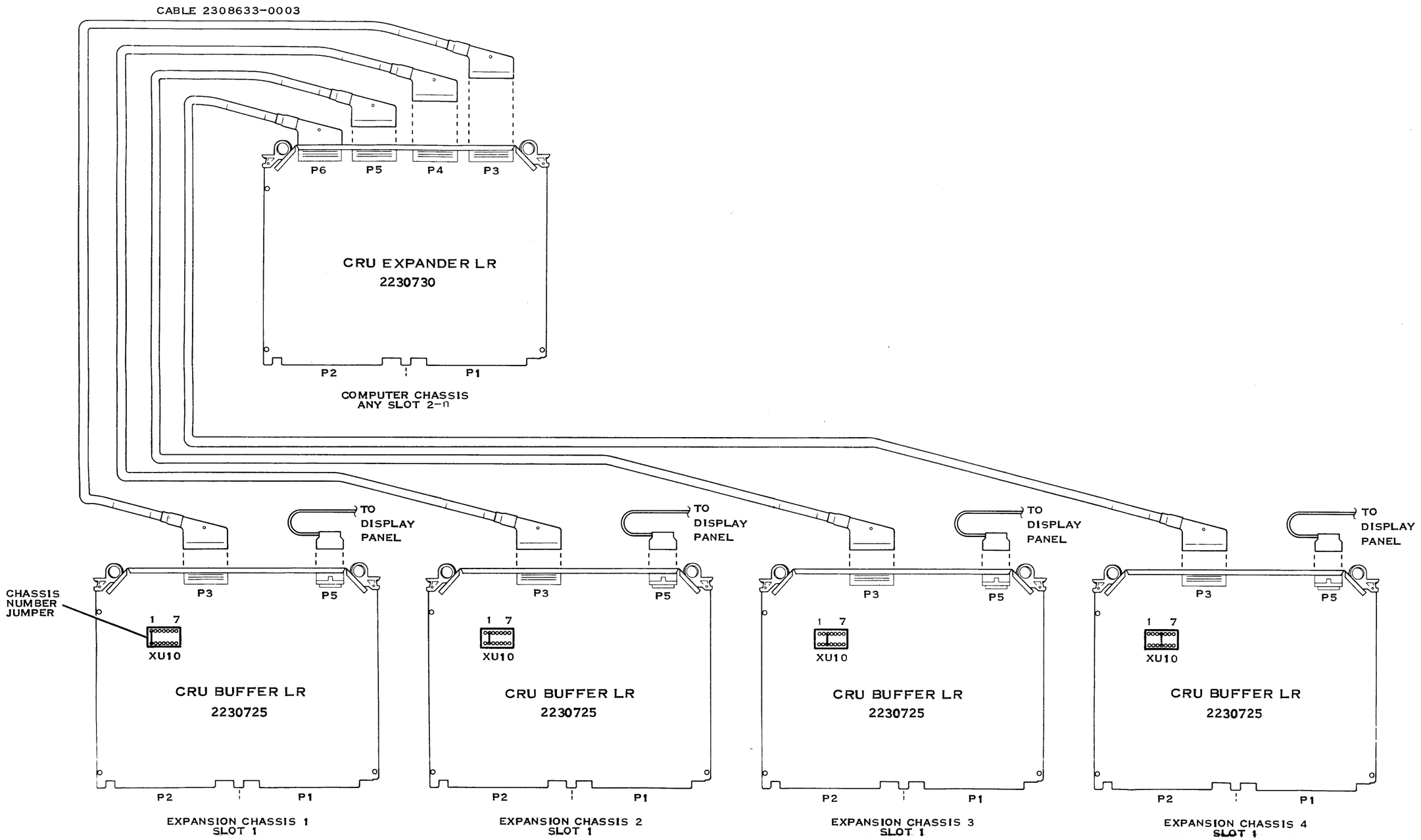
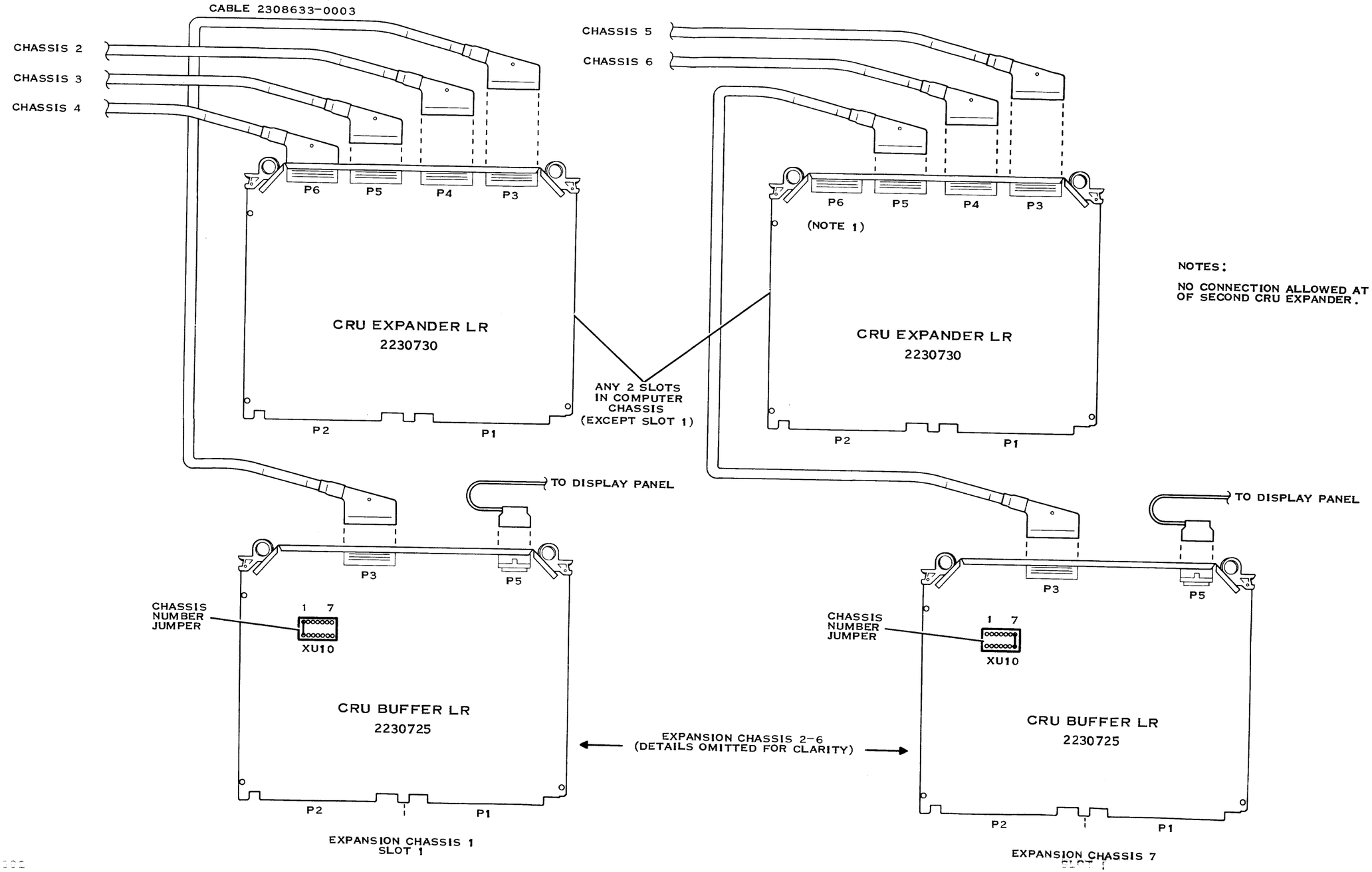


Figure 1-8. Cabling — Maximum CRU Expansion with Single CRU Expander



NOTES:

NO CONNECTION ALLOWED AT P6 OF SECOND CRU EXPANDER.

Figure 1-9. Cabling — Maximum Standard CRU Expansion

2272075-9701

1.5 TILINE INTRODUCTION

All central processors in the Business Systems 600 and Business Systems 800 family include Texas Instruments' patented TILINE. The TILINE is an asynchronous, high-speed, parallel data transfer bus with associated address and control lines. A 16-bit word is the smallest unit of data on the TILINE. The TILINE links the central processor, memory, and high-speed peripheral device controllers such as disk controllers and tape controllers. Some communications and terminal controllers have data rates that exceed CRU capabilities and connect to the TILINE. Data transfer rates up to 50 million bits-per-second are possible on the TILINE. Delays due to TILINE couplers decrease this rate for chassis-to-chassis transfers.

The TILINE is incorporated directly into the addressable memory space of the central processor. This means that a TILINE address appears to the central processsor like any memory location. Unlike the CRU, there are no special instructions to operate the TILINE. Any instruction that transfers data into or out of memory can operate the TILINE.

Each 16-bit TILINE data word is accompanied by a 20-bit word address, for a total address range of over a million words. To take full advantage of this address space, the central processor must have memory mapping capability. All processors in the Business Systems 600/800 family include mapping capability.

1.5.1 Master/Slave Relationships

Another major difference between the CRU and the TILINE is that TILINE control is distributed throughout a chassis, not centralized at the central processor. Two types of devices connect to the TILINE: TILINE masters that initiate data transfers, and TILINE slaves that accept or supply data in response to some TILINE master. Read or write data transfers always occur between a master and a slave.

The central processor is one example of a master, and a cache memory board is one example of a slave. Disk controllers, tape controllers, and TILINE couplers act as slaves or as masters, depending on the operation. Each slave device recognizes a specific range of addresses, set by switches or jumpers. Every slave monitors the TILINE address lines, but remains inactive until it decodes an address in its range. Slave addresses must be unique, or multiple slaves will attempt to respond to the common addresses.

Master devices operate independently of each other, and may request access to the TILINE bus at any time. Access is first-come, first-served, unless the requests occur at the same time. Slot position determines priority for simultaneous requests. A scheduling scheme allows a master to reserve the next TILINE access during the current operation. This overlapping reduces the overhead time to transfer data between masters. When a master gains access to the bus, it must place a 20-bit address on the bus and exchange hand-shaking control signals for each data word it transfers to or from a slave.

Most masters transfer one word per access request, but the TILINE design allows multiple-word transfers.

1.5.2 TILINE Peripheral Control Space

The TILINE peripheral control space (TPCS) is a block of addresses that are reserved for TILINE peripheral device controllers. This range includes 1024 byte addresses, extending from $F800_{16}$ to $FBFE_{16}$. In terms of the 20-bit word addresses that appear on the TILINE, the address range is $FFC00_{16}$ to $FFDFF_{16}$.

To convert a TPCS byte address to a TILINE word address, precede the byte address by 1F (for example, F800 to 1FF800), convert to binary, delete the least significant bit, and reconvert to hexadecimal form.

Each TILINE peripheral controller, including the TILINE coupler, is assigned a block of addresses in the TPCS. The central processor uses these addresses for the control and status words that set up and monitor the coupler or controller operations. A TILINE coupler requires five addresses for control and status words W0 through W4.

1.6 TILINE EXPANSION

TILINE expansion can be as simple as an extension of the TILINE bus to an additional 12 slots, or as complicated as a true multiprocessor link. The simplest expansion requires a TILINE coupler in each chassis and a set of three cables.

TILINE coupler functions:

- Provide line drivers and receivers for data, address, and control signals
- Provide a switch-selected TILINE address range (window) for data transfers to the other chassis
- Add a switch-selected bias to addresses within the window — the bias shifts addresses in the window up or down to match the addresses in the other chassis
- Check parity on data and address transfers between chassis
- Provide one maskable and one nonmaskable direct interrupt between chassis, with switch-selected direction of flow
- Provide a true multiprocessing interrupt system under TPCS control
- Resolve timing conflicts between access requests that originate in separate chassis
- Extend the software interlock instruction (ABS) to the other chassis
- Provide remote reset control and power status
- Provide full address switch, bias switch, and option switch setting checks by software
- Provide extensive function control and status checks by software

TILINE expansion requires the following hardware:

- TILINE Coupler LR, part number 2230735 (two minimum)
- Data cable (40-pin, 5-meter, twisted-pair, shielded cable), part number 2308633-0003
- Address cable (identical to data cable)
- Control (vector) cable, part number 2230740-0001
- Model 990A13 Expansion Chassis, part number 2309019-xxxx
- Programmable Interrupt Kit, part number 2309085-0001

These components replace the TILINE coupler and cables phased out of production in 1983. The new low-radiation coupler has no CRU connections. Instead, the new coupler has a block of five TPCS words dedicated to status, control, and multiprocessor interface operations. These TPCS words provide additional capability, and are not required for ordinary interchassis TILINE read or write operations.

NOTE

Components of the new TILINE expansion kit cannot be directly connected to components of the older expansion system. The cable connectors and logic board connectors are not compatible.

It is possible for old and new couplers to reside in the same chassis, connected by the backplane. For example, a computer chassis could have two TILINE expansion chassis, one served by a pair of older couplers and the other served by a pair of new (LR) couplers. Do not add older couplers into a computing system that is certified to meet the FCC and VDE EMI requirements.

Table 1-6 lists the TILINE expansion kits that are currently available.

Table 1-6. TILINE Expansion Kits

Part Number	Summary
2310920-0007	Master TILINE expansion kit LR, including chassis with NEMA 5-15 power plug (100/120 Vac)
2310920-0008	OEM master TILINE expansion kit LR, including OEM chassis with US NEMA 5-15 power plug
2310920-0009	Master TILINE expansion kit LR, including chassis with CEE(7)VII power plug (220 Vac)
2310920-0010	OEM master TILINE expansion kit LR, including OEM chassis with CEE(7)VII power plug (220 Vac)
2310920-0011	Master TILINE expansion kit LR, including chassis with BSI 1363 power plug (240 Vac)
2310920-0012	OEM master TILINE expansion kit LR, including chassis with BSI 1363 power plug (240 Vac)
2310920-0013	TILINE expansion interface kit LR (no chassis), with couplers, cables, and documentation

1.6.1 TILINE Coupler LR Physical Description

Figure 1-10 is a photograph of the TILINE coupler logic board. Notice the grounded shield-stiffener along the outer edge and the large ring-style ejector/injector tabs. These features identify the TILINE coupler as a shielded, low-radiation logic board.

Four connectors, P3 through P6, penetrate the shield-stiffener. Connectors P3, P4, and P5 mate with the address, data, and control cables. P6, the small connector, drives the POWER indicator on the display panel of a chassis that does not include a central processor or a CRU buffer.

Notice that there are six sets of miniature switches on the TILINE coupler board. These switches are:

- Option switches 1 through 7
- Option switches 8 through 12
- TILINE address upper limit switches
- TILINE address lower limit switches
- TILINE address bias switches
- Coupler TPCS base address select

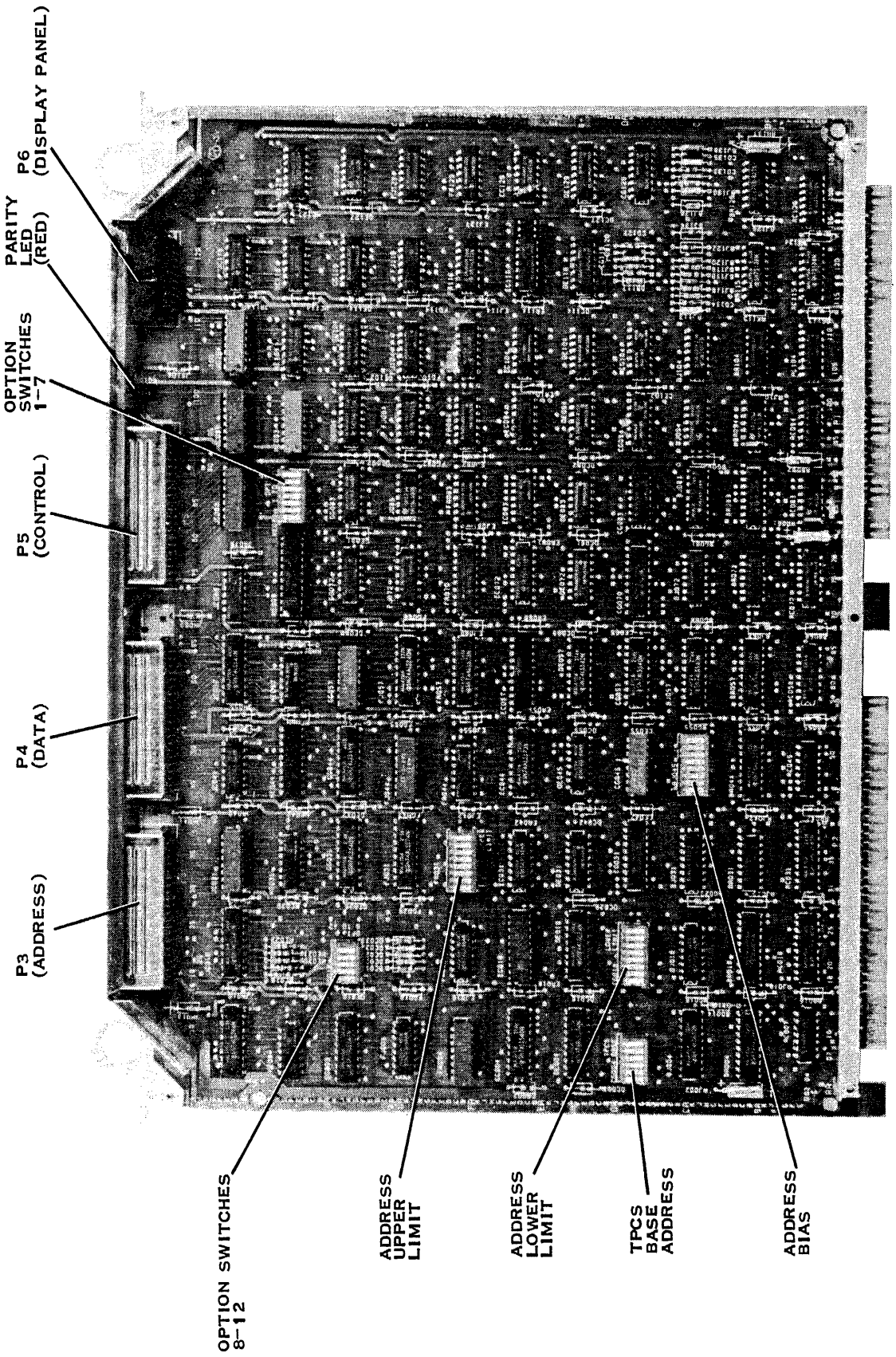


Figure 1-10. TILINE Coupler LR

1.6.2 TILINE Coupler LR Option Switches

Table 1-7 lists the switch-selectable options for a TILINE coupler. Section 3 describes these options in greater detail.

Table 1-7. TILINE Coupler LR Option Switches

Option Switch	Function When On (Closed)
OSW1	Nonmaskable interchassis interrupt option 1. Enables interrupt from backplane (P2-66) to remote coupler.
OSW2	Nonmaskable interchassis interrupt option 2. Enables interrupt from remote coupler to local backplane. Option switches 1 and 2 must not be closed in the same coupler.
OSW3	TILINE I/O reset option 1. Enables local TILINE I/O reset signal to send a reset to remote coupler.
OSW4	TILINE I/O reset option 2. Enables reset (from other coupler) to generate TILINE I/O reset in the local chassis. Option switches 3 and 4 must not be closed in the same coupler. Do not close option switch 4 if there is a CPU in the same chassis.
OSW5	Interrupt mask defeat option. Bypasses the mask for the maskable interchassis interrupt. When option switch 5 is closed, the maskable interrupt is always enabled and bit 3 of TPCS word 2 (read) is always 1. Option switch 5 must be open to use the multiprocessor interface.
OSW6	Remote chassis write disable. Prevents the local chassis from writing to the remote coupler.
OSW7	Maskable interchassis interrupt option. Enables the TPCS-maskable interrupt from the backplane (P1-66) to the remote coupler. Never set option switch 7 on both couplers. Option switch 7 must be open to use the multiprocessor interface.
OSW8	TPCS block 0 enable to remote coupler. Cannot be mapped by TPCS control. Disables mapping to block 0 of OSW9, OSW10, and OSW11.
OSW9	TPCS block 1 enable to remote coupler. Can be mapped to block 0 of remote TPCS by local TPCS control.
OSW10	TPCS block 2 enable to remote coupler. Can be mapped to block 0 of remote TPCS by local TPCS control.
OSW11	TPCS block 3 enable to remote coupler. Can be mapped to block 0 of remote TPCS by local TPCS control.
OSW12	TPCS block 4 enable to remote coupler. Enable complete TPCS to remote coupler. Supersedes block 0 through 4 selection by option switches 8 through 11. Cannot be mapped by TPCS control.

Option switches 8 through 12 control TPCS address transfers to the remote chassis and are independent of the address upper limit, lower limit, and bias switches. These switches allow a central processor in the local chassis to issue commands or check the status of TILINE controllers in the remote chassis.

The TILINE coupler divides the TPCS addresses into block 0 through block 4, where block 4 is the entire TPCS address range. Blocks 0 through 3 each consist of eight 16-word groups. One block could drive 16 disk or tape controllers in the remote chassis. .

Table 1-8 lists the option switch 8 through 12 settings and the corresponding TPCS address blocks.

Table 1-8. TPCS Address Blocks for Transfer to Remote Chassis

Option Switch	TPCS Block	TPCS Addresses to Remote Chassis
OSW8	0	F800 — F81E F880 — F89E F900 — F91E F980 — F99E FA00 — FA1E FA80 — FA9E FB00 — FB1E FB80 — FB9E
OSW9	1	F820 — F83E F8A0 — F8BE F920 — F93E F9A0 — F9BE FA20 — FA3E FAA0 — FABE FB20 — FB3E FBA0 — FBBE
OSW10	2	F840 — F85E F8C0 — F8DE F940 — F95E F9C0 — F9DE FA40 — FA5E FAC0 — FADE FB40 — FB5E FBC0 — FBDE
OSW11	3	F860 — F87E F8E0 — F8FE F960 — F97E F9E0 — F9FE FA60 — FA7E FAE0 — FAFE FB60 — FB7E FBE0 — FBFE
OSW12	4	F800 — FBFE (Entire TPCS)

Control bits in the local TPCS word allow the local coupler to bias TPCS block 1, 2, or 3 to block 0 of the remote chassis TPCS. Table 1-9 shows the mapping for each of the TPCS blocks.

Table 1-9. TPCS Block Mapping

TPCS Bias Enable (W0, Bits 2, 3)			
Bit 2	**Bit 3**	**Local TPCS**	**Remote TPCS**
0	0	Direct block transfer, no mapping	
0	1	F820 to F83E	F800 to F81E
		F8A0 to F8BE	F880 to F89E
		F920 to F93E	F900 to F91E
		F9A0 to F9BE	F980 to F99E
		FA20 to FA3E	FA00 to FA1E
		FAA0 to FABE	FA80 to FA9E
		FB20 to FB3E	FB00 to FB1E
		FBA0 to FBBE	FB80 to FB9E
1	0	F840 to F85E	F800 to F81E
		F8C0 to F8DE	F880 to F89E
		F940 to F95E	F900 to F91E
		F9C0 to F9DE	F980 to F99E
		FA40 to FA5E	FA00 to FA1E
		FAC0 to FADE	FA80 to FA9E
		FB40 to FB5E	FB00 to FB1E
		FBC0 to FBDE	FB80 to FB9E
1	1	F860 to F87E	F800 to F81E
		F8E0 to F8FE	F880 to F89E
		F960 to F97E	F900 to F91E
		F9E0 to F9FE	F980 to F99E
		FA60 to FA7E	FA00 to FA1E
		FAE0 to FAFE	FA80 to FA9E
		FB60 to FB7E	FB00 to FB1E
		FBE0 to FBFE	FB80 to FB9E

Note:

If OSW8 or OSW12 is closed, the mapping feature is disabled regardless of the TPCS bias enable bits.

1.6.3 TILINE Address Limit and Bias Switches

Each TILINE coupler has two sets of switches that determine which addresses are passed on to the remote chassis. A third switch set adds a positive or negative bias to the passed addresses. Each set includes eight individual switches for an 8-bit binary address word. The minimum switch increment is eight kilobytes (four kilowords).

The lower limit switches set the lowest address that the coupler recognizes for transfer to the remote chassis. The upper limit switches set the highest address for transfer to the remote chassis. These switch settings have no effect on the TPCS unless the upper limit switch setting is 2048 kilobytes ($1FFFE_{16}$ bytes or $FFFFF_{16}$ words).

The bias adjusts the passed addresses to the desired range of addresses in the remote chassis. The following example shows the effect of the address limit and bias switches.

Chassis A:	**Chassis B:**
Processor A	Processor B
Memory 1 (0 to 256K)	Memory 1 (0 to 256K)
Memory 2 (256K to 512K)	Memory 2 (256K to 512K)
Coupler A	Coupler B

Note:

K equals 1024 bytes

Assume that you want the processor in chassis A to share the second memory board in chassis B. The problem is how to convert a range of memory addresses in chassis A to the correct range of addresses in chassis B. Coupler A does the range selection and biasing before transmitting the address to coupler B.

First, set the lower limit switch on coupler A to 512 kilobytes, which is the first address above the chassis A memory range. Set the upper limit switch to 768 kilobytes, which is the lower limit plus the size of the shared memory. The bias is negative 256 kilobytes. This bias moves chassis A addresses 512 through 768 kilobytes down to 256 through 512 kilobytes for transfer to coupler B. Drivers in coupler B put the address on the chassis B backplane, and the memory responds to the TILINE read or write command.

NOTE

If you set the TILINE address upper limit switch to the upper limit setting ($FFFFF_{16}$ words), there is a TPCS address conflict. TPCS addresses now have two paths to the remote chassis. This condition may cause erratic operation that appears like a hardware failure.

1.6.4 TILINE Coupler LR TPCS Base Address Switches

Each TILINE coupler has a group of five dedicated control and status words in the TPCS. These words occupy five consecutive word addresses from the TPCS base address to TPCS base address + 4 word addresses.

The TPCS base address is switch-selected in eight-word increments between TILINE addresses FFC00 and FFDF8 (central processor byte addresses F800 to FBF0).

1.6.5 TILINE Coupler LR Specifications

Table 1-10 summarizes the specifications for the TILINE Coupler LR.

Table 1-10. TILINE Coupler LR (2230735) Specifications

Data transfer type	Asynchronous, parallel data transfers initiated by TILINE masters located anywhere on the bus
Interrupt provisions	Two interchassis interrupts (one maskable, one nonmaskable) with switch options for direction of flow
	Multiprocessor interrupts under software control (requires multiprocessor software)
	Parity error interrupt
Reset features	Optional connection to reset all expansion chassis on main chassis power reset or I/O reset
Chassis space requirement	Any full slot
Dc power requirements	+ 5.0 Vdc ± 3% 3.0 A maximum
	− 5.0 Vdc ± 3% 0.03 A maximum (from chassis power supply)
Heat load	15.6 watts maximum
Electromagnetic interference	Meets RFI/EMI limits of VDE 0871 (level A) and FCC docket 20780 (class A) when installed in a low-radiation chassis
Environmental limits:	
Temperature:	
Operating	0° to 65 °C (32° to 149 °F, derated 2 °C for each 2500 ft elevation)
Nonoperating	− 40 to 70 °C (− 40° to 158 °F)
Humidity:	
Operating	0 to 95% noncondensing
Nonoperating	0 to 95% noncondensing

1.7 TILINE COUPLER LR CABLING CONFIGURATIONS

Figure 1-11 is a cabling diagram that shows how to connect two chassis by TILINE expansion. Notice that the data and address cables have the same part number. It is possible to accidentally cross-connect the data cable to the address connector and the address cable to the data connector. A crossed cable bit in one of the TPCS status words informs the software of the error.

The control (vector) cable is keyed so that it cannot be cross-connected. Note the vector arrowhead direction on the control cable. In cases of simultaneous coupler requests for the TILINE, the vector direction determines which coupler gets access. Priority goes down in the direction of the arrowhead. In Figure 1-11, chassis B has priority.

TILINE couplers can be connected into many different network configurations. There are three fundamental restrictions:

- It must not be possible for the TILINE of one chassis to address itself through the TILINE coupler, even after biasing through other couplers. This could happen in a circular (ring) network if the address limits and biases are not selected carefully.
- A TILINE coupler, cables, and chassis backplane all add delays in the signal path. Delays can become excessive if data must pass through multiple couplers and chassis.
- A chassis containing two or more couplers cannot have more than one coupler control vector pointing into the chassis. Conflicting vectors defeat the purpose of the vector scheme and may lead to TILINE lockup.

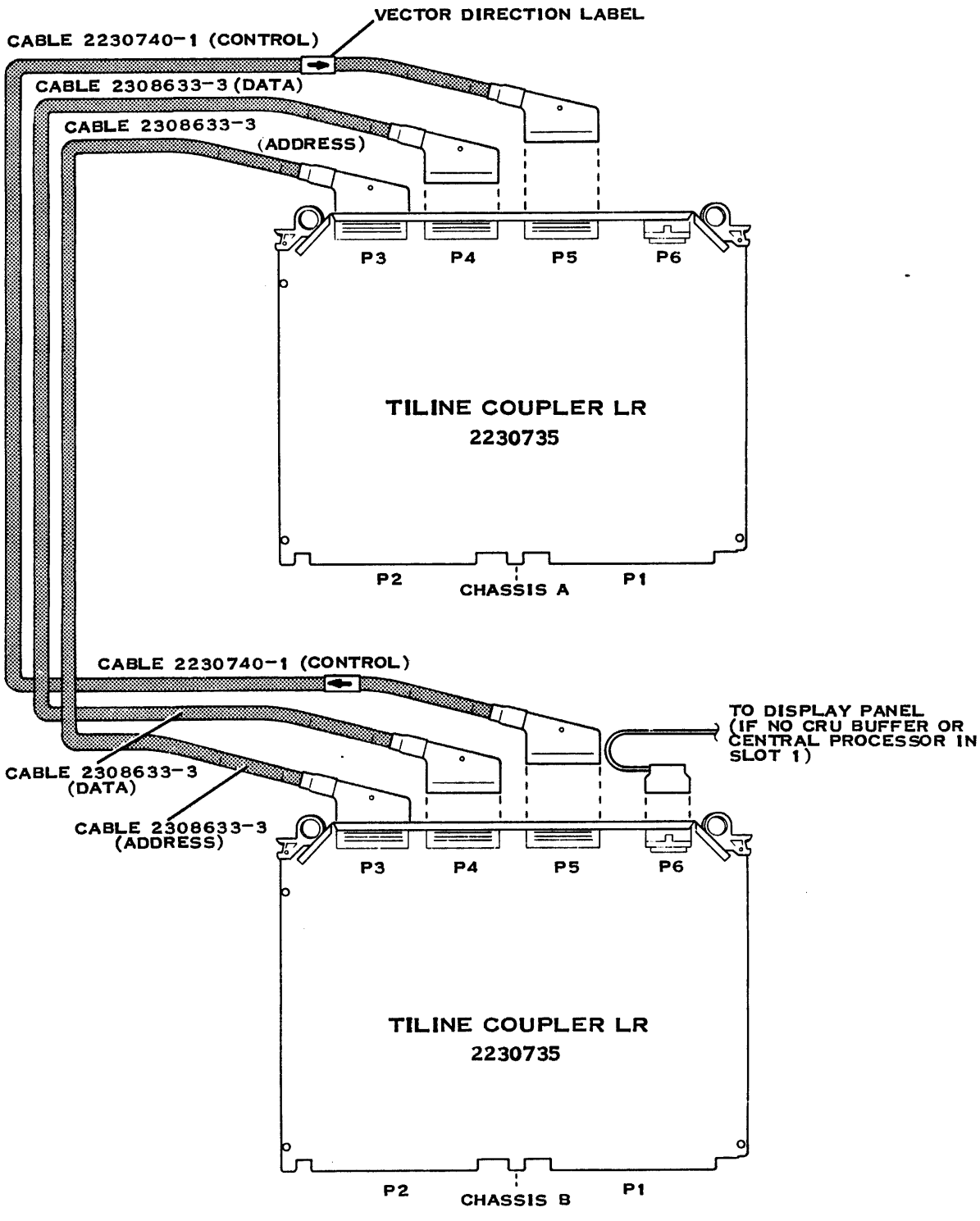


Figure 1-11. TILINE Coupler LR Cabling

2

Installation

2.1 GENERAL

This section provides the information that you need to install and connect the CRU expansion and TILINE expansion hardware. Expansion chassis details are available in the chassis manuals, and are not repeated here.

2.2 SELECTING A SLOT FOR THE CRU EXPANDER LR

If you purchased your CRU expander as part of a Business Systems package, the slot selection is already done. Skip ahead to the slot installation description.

The preferred slot for a CRU expander in a 990A13 chassis is slot 11, with interrupt level 7 on P2-66. A CRU expander cannot share an interrupt level with another CRU expander, other CRU device, or TILINE device. Each CRU expander must have its own non-shared interrupt level.

The type 13-1 interrupt board has interrupt level 7 on 11P2-66 and on 5P1-66. Whatever board goes in slot 5 should not generate interrupts on P1-66. Your alternatives are to:

- Select a different slot for the CRU expander.
- Install a programmable interrupt board that does not require the CRU expander to share an interrupt level.

NOTE

Preferred slot assignments may change with the introduction of new peripheral devices or operating system revisions.

If your CRU expander has all of the standard jumper selections, skip ahead to the jumper verification description.

2.2.1 CRU Addresses Dedicated to the CRU Expander

A CRU expander passes unmodified CRU addresses to as many as four CRU buffers. These addresses are totally independent of the slot location of the CRU expander board. Two CRU base addresses are dedicated to reading the interrupt vector word. These addresses are:

CRU Base Address (Workspace Register 12)	CRU Base Address (CRUBIT3-15)	Interrupt Vector
1F00	F80	Chassis 1-4
1F20	F90	Chassis 5-7

Jumper options allow you to choose CRU expander decoding of these two base addresses or to control decoding with a module select input.

With the standard jumpers, onboard logic decodes the CRU base address for the interrupt vector word. This makes the addressing independent of CRU expander slot location.

If jumper J4 or J5 is installed, the interrupt vector word is no longer independent of slot location. TI standard operating system software does not support the jumper J4 or J5 option.

2.2.2 CRU Expander Interrupts

With standard jumpers, the CRU expander interrupt is on P2-66. This interrupt should not share an interrupt level with any other device, especially another CRU expander or a TILINE device.

There is a nonstandard jumper option that allows an interrupt output on P2-65. The 990A13 backplane does not have any connection to P2-65. To use this option, you have to custom-wire the CRU expander (P2-65 to P1-66) or add wires to the backplane. TI operating system software does not support this option.

2.3 PREPARING A SLOT FOR THE CRU EXPANDER LR

No slot preparation is necessary for the CRU expander.

2.4 VERIFYING THE CRU EXPANDER LR JUMPER OPTIONS

Figure 2-1 shows the standard jumper options for a CRU expander board. J8 and J9 are the only jumpers that you can change without either cutting wire or soldering eyelets. Movable jumper J8 selects the standard CRU base address for the chassis 1-4 interrupt vector. J9 selects the standard CRU base address for the chassis 5-7 interrupt vector.

Verify the jumper configuration on your board by reference to Figure 2-1.

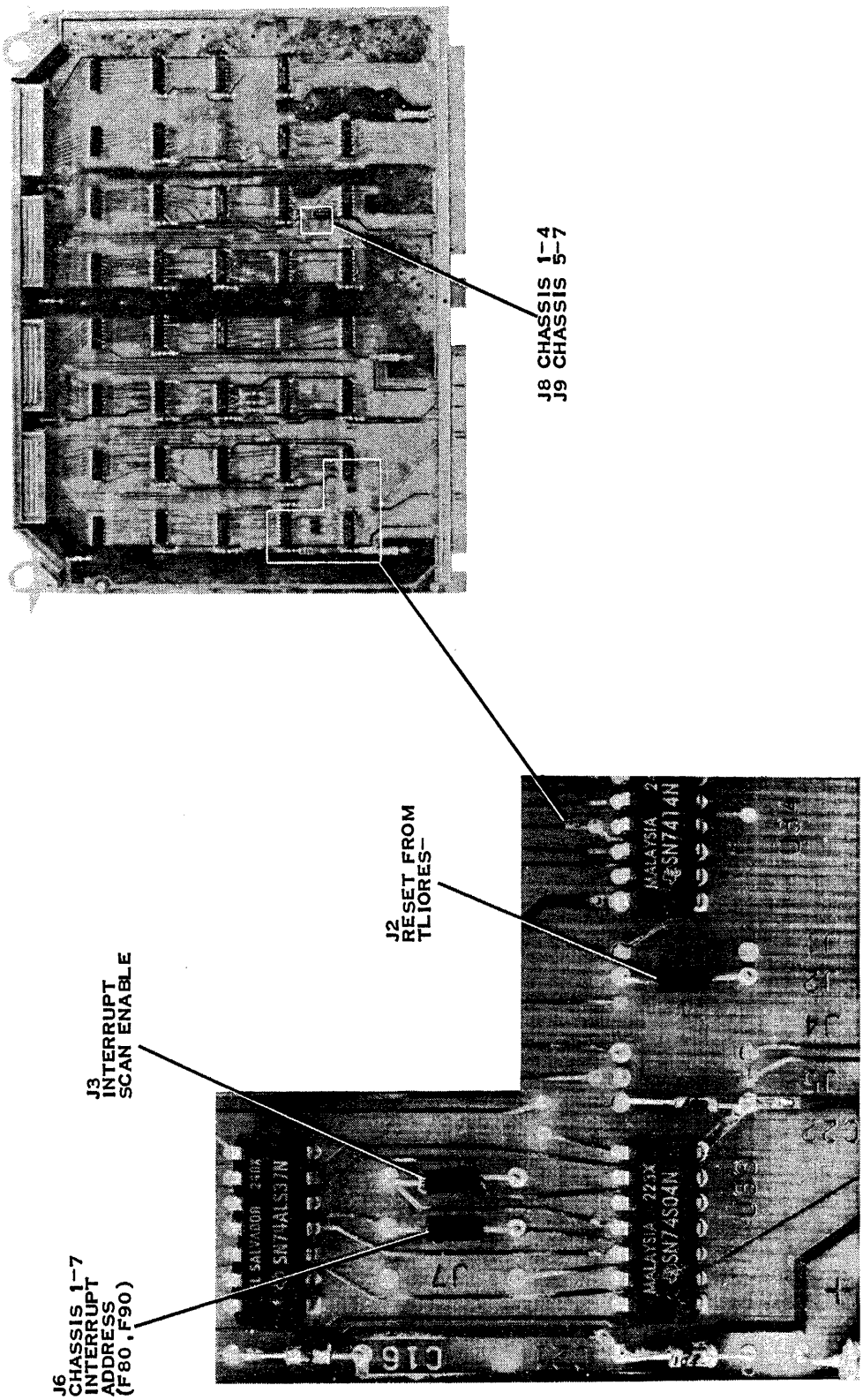


Figure 2-1. CRU Expander LR Standard Jumper Settings

2284406

2.5 INSTALLING AND CABLING THE CRU EXPANDER LR

Section 2 of the *Model 990A13 Chassis, General Description* manual describes how to install shielded logic boards in a 990A13 chassis. Install the CRU expander by following the installation procedure for shielded logic boards.

Refer to the CRU expansion cabling drawings in Section 1. If your system has only one expansion chassis, select connector P3 for the output cable. Table 2-1 lists the connector assignments for systems with more than one expansion chassis.

Table 2-1. CRU Expander LR Connector Assignments to Expansion Chassis

Expander	Connector	Expansion Chassis
1	P3	1
1	P4	2
1	P5	3
1	P6	4
2	P3	5
2	P4	6
2	P5	7
2	P6	N/A

NOTE

Do not connect to P6 of a CRU expander board at workspace register 12 address $1F20_{16}$ (absolute address $F90_{16}$).

Because the connectors are close together, it is easier to install cables in the higher-numbered connectors first. By working right-to-left (from P6 to P3), you always have clear access to the connectors.

Install each cable as described in Section 2 of *Model 990A13 Chassis, General Description.*

2.6 CRU BUFFER LR SLOT SELECTION

Both the interrupt and the CRU module select wiring require that you install the CRU buffer in slot 1 of the expansion chassis. No other slot selection will work.

2.7 CRU BUFFER LR SLOT PREPARATION

If you purchased your CRU expansion system as part of a complete Business Systems package, the interrupt wiring is done for you. Verify that the interrupt board is properly installed in the interrupt connector above slot 1. Skip ahead to the buffer installation and cabling description.

Interrupt levels from the interrupt connector are hardwired to chassis slot 1. Connections between slot 2 through 13 interrupt outputs and the CRU buffer interrupt level inputs are made by the plug-in interrupt board.

2.8 VERIFYING THE CRU BUFFER LR JUMPER OPTIONS

Figure 2-2 shows the standard jumper options for a CRU buffer board. A chassis number select jumper on socket XU10 is the only jumper you can change without either cutting wire or soldering eyelets. Chassis are numbered 1 through 7, from left to right on the socket. There must be a chassis number select jumper in socket XU10.

Verify the jumper configuration on your board by reference to Figure 2-2.

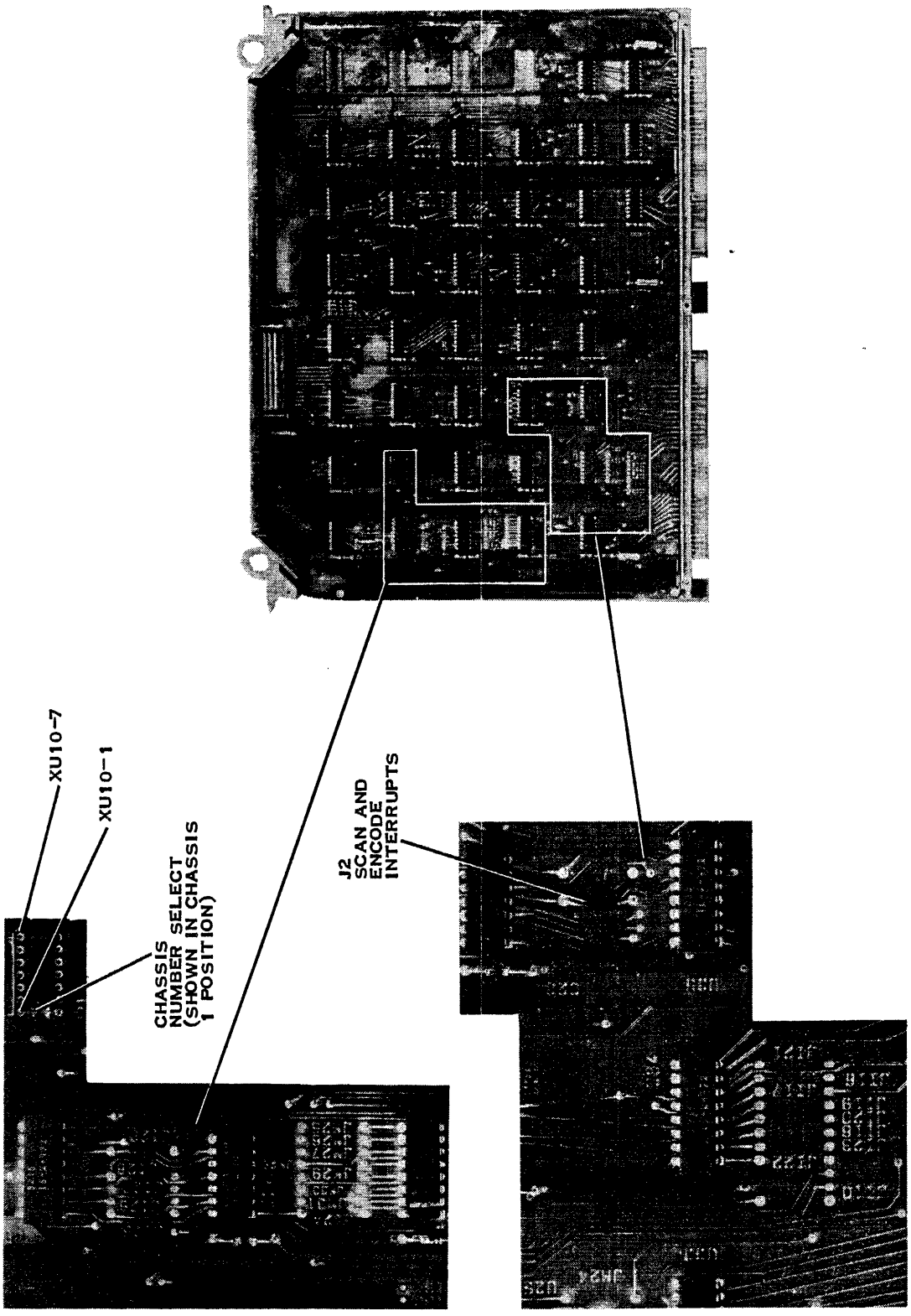


Figure 2-2. CRU Buffer LR Standard Jumper Settings

2.9 INSTALLING AND CABLING THE CRU BUFFER LR

Section 2 of the *Model 990A13 Chassis, General Description* manual describes how to install shielded logic boards in a 990A13 chassis. Install the CRU buffer by following the installation procedure for shielded logic boards.

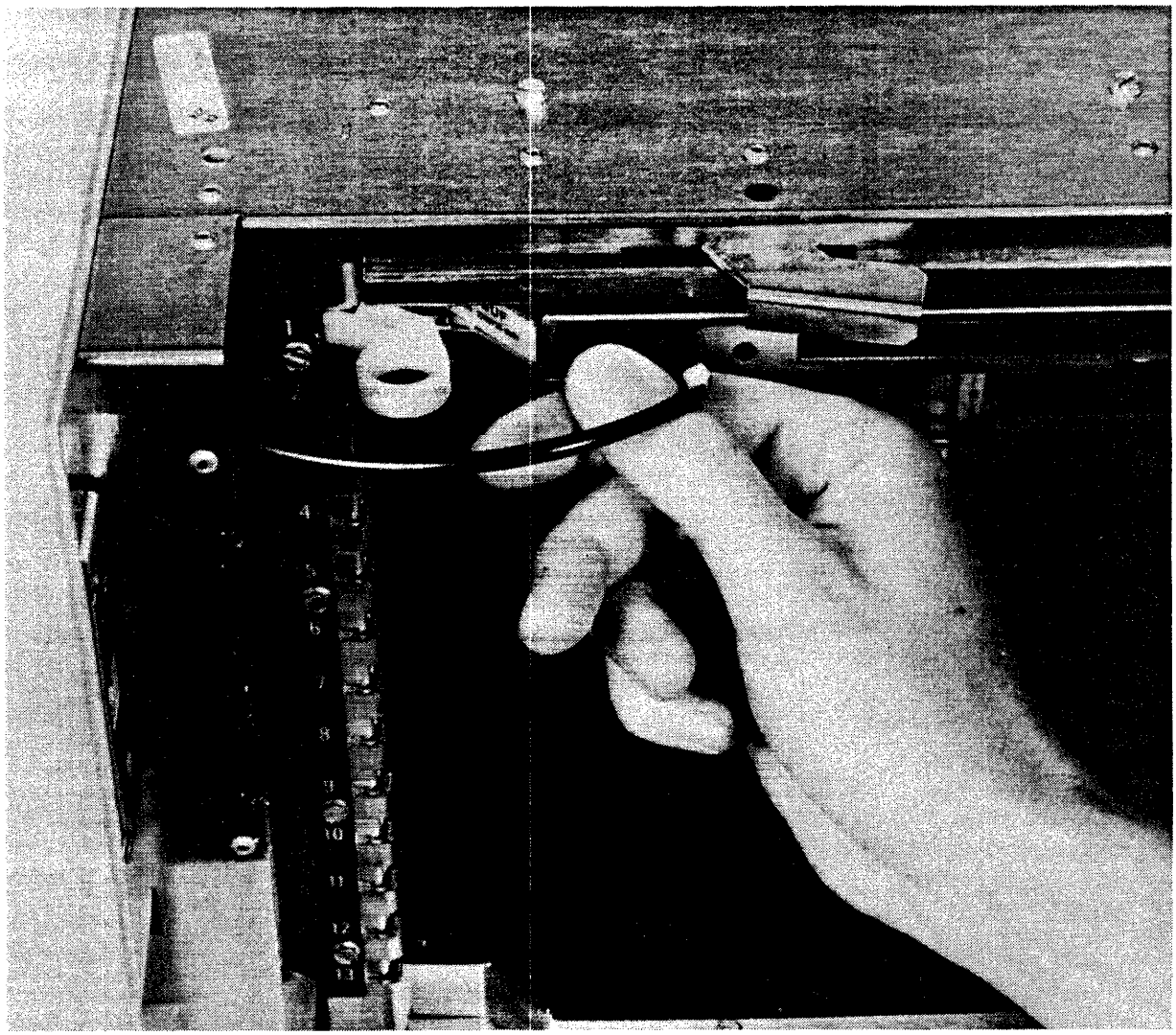
Refer to the CRU expansion cabling drawings in Section 1. If you have more than one expansion chassis, make sure that you have the correct cable from the CRU expander. Cable connector assignments appear in Table 2-1.

NOTE

The chassis number select jumper position and the CRU expander output connector *must* agree as shown in Table 2-1.

Section 2 of the *Model 990A13 Chassis, General Description* manual describes how to cable a shielded logic board. Connect the 2308633-3 expansion cable to CRU buffer connector P3 by following the procedure in the general description manual.

Follow the instructions for connecting the display panel, and connect the display panel cable to CRU buffer board connector P5. Figure 2-3 shows how to connect the display panel cable.



2284408

Figure 2-3. Connecting the Display Panel

2.10 SELECTING A SLOT FOR THE TILINE COUPLER LR

If you purchased your TILINE coupler as part of a Business Systems package, the slot selection is already done. Skip ahead to the slot installation description.

The preferred slot for a TILINE coupler in a 990A13 computer chassis is slot 6. In a TILINE expansion chassis, the preferred location is slot 1. If the chassis also contains a CRU buffer, the coupler preferred location becomes slot 2. The CRU buffer, if any, must be in slot 1. There are no currently-defined preferred slot locations in multiprocessor systems.

2.11 PREPARING A SLOT FOR THE TILINE COUPLER LR

You must remove the TILINE access-granted (TLAG–) jumper associated with the TILINE coupler slot. Refer to Section 2 of *Model 990A13 Chassis, General Description* for the procedure.

2.12 VERIFYING THE TILINE COUPLER LR SWITCH SETTINGS

There are no jumper options on the TILINE coupler board. All of the options are switch-selectable, as are the TPCS base address, upper and lower address limits, and the address bias.

Section 3 of this manual describes all of the option switch settings.

Figure 2-4 is a switch chart that you can use to verify the switch settings. Make copies of the blank chart when you initially choose the options and addresses. Enter the switch settings on the copies and save them for future use.

The TILINE coupler words in the TPCS allow a software verification of all switch settings on the TILINE coupler board.

COUPLER ID _____

CHASSIS _____

SLOT NO. _____

INTERRUPT LEVELS(S) _____ P1-66

_____ P2-66

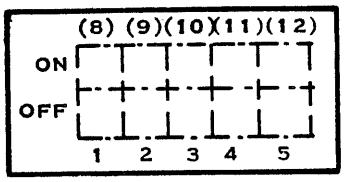
MULTIPROCESSOR AUX/HOST _____

VECTOR DIRECTION:

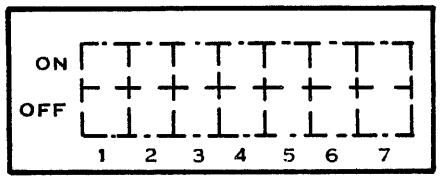
INCOMING _____

OUTGOING _____

OPTION SWITCHES 8-12



OPTION SWITCHES 1-7

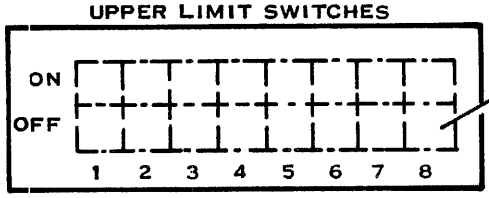


(SEE NOTE 1)

NOTES:

1. DO NOT USE ANY OF THESE OPTION COMBINATIONS:
   1 AND 2
   3 AND 4
   5 AND 7
2. UPPER LIMIT TILINE ADDRESS = SWITCH X 2000_{16} +$1FFE_{16}$ BYTES
3. LOWER LIMIT TILINE ADDRESS = SWITCH X 2000_{16} BYTES
4. BIAS SETTING IS 8 MOST SIGNIFICANT BITS OF SUM:
   TWO'S COMPLEMENT OF LOWER LIMIT ADDRESS + REMOTE STARTING ADDRESS
   OMIT LSB FROM ALL BIAS CALCULATIONS

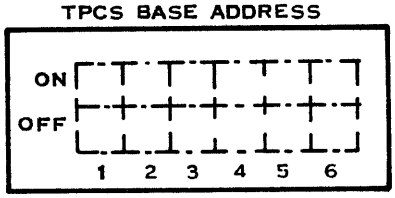
UPPER LIMIT SWITCHES



LSB = 8 KILO-BYTES (4 KILO-WORDS)

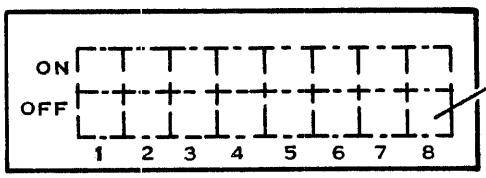
OFF = LOGIC 1 (SEE NOTE 1)

TPCS BASE ADDRESS



OFF LOGIC 1

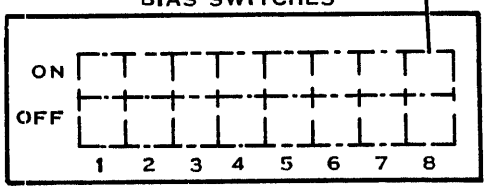
LOWER LIMIT SWITCHES



LSB = 8 KILO-BYTES (4 KILO-WORDS)

OFF = LOGIC 1 (SEE NOTE 3)

BIAS SWITCHES



OFF = LOGIC 1

OTHER NOTES.

_____

_____

_____

_____

_____

2284384

Figure 2-4. TILINE Coupler LR Switch Chart

2.12.1 TPCS Base Address Switch

Five words in the TPCS are dedicated to TILINE coupler status and control words. These five words are distinct from the block of TILINE addresses that a coupler passes to another chassis. A six-section switch determines the lowest (base) address of the group of TPCS words.

Table 2-2 shows typical settings for the TPCS base address switches. The table includes the central processor byte address and the corresponding 20-bit TILINE word address for each setting.

NOTE

Any switch section that is off corresponds to a logic 1.

Table 2-2. Typical TPCS Base Address Switch Settings

TPCS Base Address Switch 1 2 3 4 5 6	Byte Address	20-Bit TILINE Address
0 0 0 0 0 0	F800	FFC00
0 0 0 0 0 1	F810	FFC08
0 0 0 0 1 0	F820	FFC10
0 0 0 0 1 1	F830	FFC18
0 0 0 1 0 0	F840	FFC20
0 0 0 1 0 1	F850	FFC28
0 0 0 1 1 0	F860	FFC30
0 0 0 1 1 1	F870	FFC38
0 0 1 0 0 0	F880	FFC40
0 0 1 0 0 1	F890	FFC48
0 0 1 0 1 0	F8A0	FFC50
0 0 1 0 1 1	F8B0	FFC58
0 0 1 1 0 0	F8C0	FFC60
0 0 1 1 0 1	F8D0	FFC68
0 0 1 1 1 0	F8E0	FFC70
0 0 1 1 1 1	F8F0	FFC78
0 1 0 0 0 0	F900	FFC80
0 1 0 0 0 1	F910	FFC88
0 1 0 0 1 0	F920	FFC90
0 1 0 0 1 1	F930	FFC98
0 1 0 1 0 0	F940	FFCA0
0 1 0 1 0 1	F950	FFCA8
0 1 0 1 1 0	F960	FFCB0
0 1 0 1 1 1	F970	FFCB8
0 1 1 0 0 0	F980	FFCC0
0 1 1 0 0 1	F990	FFCC8
0 1 1 0 1 0	F9A0	FFCD0
0 1 1 0 1 1	F9B0	FFCD8
0 1 1 1 0 0	F9C0	FFCE0
0 1 1 1 0 1	F9D0	FFCE8
0 1 1 1 1 0	F9E0	FFCF0
0 1 1 1 1 1	F9F0	FFCF8

Table 2-2. Typical TPCS Base Address Switch Settings (Continued)

TPCS Base Address Switch 1 2 3 4 5 6	Byte Address	20-Bit TILINE Address
1 0 0 0 0 0	FA00	FFD00
1 0 0 0 0 1	FA10	FFD08
1 0 0 0 1 0	FA20	FFD10
1 0 0 0 1 1	FA30	FFD18
1 0 0 1 0 0	FA40	FFD20
1 0 0 1 0 1	FA50	FFD28
1 0 0 1 1 0	FA60	FFD30
1 0 0 1 1 1	FA70	FFD38
1 0 1 0 0 0	FA80	FFD40
1 0 1 0 0 1	FA90	FFD48
1 0 1 0 1 0	FAA0	FFD50
1 0 1 0 1 1	FAB0	FFD58
1 0 1 1 0 0	FAC0	FFD60
1 0 1 1 0 1	FAD0	FFD68
1 0 1 1 1 0	FAE0	FFD70
1 0 1 1 1 1	FAF0	FFD78
1 1 0 0 0 0	FB00	FFD80
1 1 0 0 0 1	FB10	FFD88
1 1 0 0 1 0	FB20	FFD90
1 1 0 0 1 1	FB30	FFD98
1 1 0 1 0 0	FB40	FFDA0
1 1 0 1 0 1	FB50	FFDA8
1 1 0 1 1 0	FB60	FFDB0
1 1 0 1 1 1	FB70	FFDB8
1 1 1 0 0 0	FB80	FFDC0
1 1 1 0 0 1	FB90	FFDC8
1 1 1 0 1 0	FBA0	FFDD0
1 1 1 0 1 1	FBB0	FFDD8
1 1 1 1 0 0	FBC0	FFDE0
1 1 1 1 0 1	FBD0	FFDE8
1 1 1 1 1 0	FBE0	FFDF0
1 1 1 1 1 1	FBF0	FFDF8

Notes:

1 = off

0 = on

2.12.2 Lower Limit and Upper Limit Address Switch Settings

Settings of the lower limit and upper limit address switches select a range of addresses that the TILINE coupler passes to a remote coupler. Addresses outside this range affect only the local chassis. TPCS addresses are an exception to this rule, as they are controlled by option switch selections and TPCS bias bits in TPCS word 0.

Table 2-3 shows some examples of lower limit addresses and the switch settings for those limits. Table 2-4 shows examples of upper limit addresses and switch settings.

NOTE

Set the lower limit and upper limit switches to pass only the minimum number of addresses that must go to the remote chassis. If you set the address range too wide, the TILINE coupler may generate unneeded TILINE access requests in the remote chassis. The result is unnecessary delay of TILINE operations.

Table 2-3. TILINE Coupler LR Lower Limit Address Switch Settings

Lower Limit Switches 1 2 3 4 5 6 7 8	Byte Address Hexadecimal	Kilobytes
0 0 0 0 0 0 0 0	000000	0
0 0 0 0 0 0 0 1	002000	8
0 0 0 0 0 0 1 0	004000	16
0 0 0 0 0 0 1 1	006000	24
0 0 0 0 0 1 0 0	008000	32
0 0 0 0 0 1 0 1	00A000	40
0 0 0 0 0 1 1 0	00C000	48
0 0 0 0 0 1 1 1	00E000	56
0 0 0 0 1 0 0 0	010000	64
0 0 0 0 1 0 0 1	012000	72
0 0 0 0 1 0 1 0	014000	80
0 0 0 0 1 0 1 1	016000	88
0 0 0 0 1 1 0 0	018000	96
0 0 0 0 1 1 0 1	01A000	104
0 0 0 0 1 1 1 0	01C000	112
0 0 0 0 1 1 1 1	01E000	120
0 0 0 1 0 0 0 0	020000	128
0 0 1 0 0 0 0 0	040000	256
0 0 1 1 0 0 0 0	060000	384
0 1 0 0 0 0 0 0	080000	512
0 1 0 1 0 0 0 0	0A0000	640
0 1 1 0 0 0 0 0	0C0000	768
0 1 1 1 0 0 0 0	0E0000	896
1 0 0 0 0 0 0 0	100000	1024
1 0 0 1 0 0 0 0	120000	1152
1 0 1 0 0 0 0 0	140000	1280
1 0 1 1 0 0 0 0	160000	1408
1 1 0 0 0 0 0 0	180000	1536
1 1 0 1 0 0 0 0	1A0000	1664
1 1 1 0 0 0 0 0	1C0000	1792
1 1 1 1 0 0 0 0	1E0000	1920
1 1 1 1 1 1 1 1	1FE000	2040

Notes:

1 = off

0 = on

Starting address = switch setting (hex) × 2000 (hex)

Table 2-4. TILINE Coupler LR Upper Limit Address Switch Settings

Upper Limit Switches 1 2 3 4 5 6 7 8	Byte Address Hexadecimal	Kilobytes
0 0 0 0 0 0 0 0	001FFE	8
0 0 0 0 0 0 0 1	003FFE	16
0 0 0 0 0 0 1 0	005FFE	24
0 0 0 0 0 0 1 1	007FFE	32
0 0 0 0 0 1 0 0	009FFE	40
0 0 0 0 0 1 0 1	00BFFE	48
0 0 0 0 0 1 1 0	00DFFE	56
0 0 0 0 0 1 1 1	00FFFE	64
0 0 0 0 1 0 0 0	011FFE	72
0 0 0 0 1 0 0 1	013FFE	80
0 0 0 0 1 0 1 0	015FFE	88
0 0 0 0 1 0 1 1	017FFE	96
0 0 0 0 1 1 0 0	019FFE	104
0 0 0 0 1 1 0 1	01BFFE	112
0 0 0 0 1 1 1 0	01DFFE	120
0 0 0 0 1 1 1 1	01FFFE	128
0 0 0 1 0 0 0 0	021FFE	136
0 0 0 1 0 1 1 0	02FFFE	192
0 0 0 1 1 1 1 1	03FFFE	256
0 0 1 0 1 1 1 1	05FFFE	384
0 0 1 1 1 1 1 1	07FFFE	512
0 1 0 0 1 1 1 1	09FFFE	640
0 1 0 1 1 1 1 1	0BFFFE	768
0 1 1 0 1 1 1 1	0DFFFE	896
0 1 1 1 1 1 1 1	0FFFFE	1024
1 0 0 0 1 1 1 1	11FFFE	1152
1 0 0 1 1 1 1 1	13FFFE	1280
1 0 1 0 1 1 1 1	15FFFE	1408
1 0 1 1 1 1 1 1	17FFFE	1536
1 1 0 0 1 1 1 1	19FFFE	1664
1 1 0 1 1 1 1 1	1BFFFE	1792
1 1 1 0 1 1 1 1	1DFFFE	1920
1 1 1 1 1 1 1 1	1FFFFE	2048

Notes:

1 = off

0 = on

Ending address = (Switch setting (hex) × 2000 (hex)) + 1FFE

2.12.3 Setting the Address Bias Switches

Upper and lower limit address switch settings select a range of local chassis addresses. The coupler adds a bias to addresses in this range before sending them to the remote chassis. The effect of the bias is to map the address range in the local chassis to remote chassis addresses. This bias may be either positive or negative.

To determine the bias switch settings, subtract the lower limit address in the local chassis from the desired remote starting address. The most significant eight bits of the result represent the desired bias switch settings.

Follow these detailed steps to determine the bias switch settings:

1. Take the two's complement of the local starting address as follows:

   a. Write the binary representation of the address. This address must have 21 bits, to correspond to the full 2048-kilobyte TILINE address space. Do not omit leading zeros.

   b. Change all zeros to ones, and all ones to zeros (one's complement).

   c. Truncate the LSB (byte selector bit).

   d. Add one to the remaining 20-bit number, which converts the number to 20-bit two's complement.

2. Write the remote chassis starting address as a 21-bit binary number and then truncate the byte selector bit (LSB). Do not omit leading zeros.

3. Add the 20-bit two's complement of the local address to the 20-bit remote chassis starting address. Adding the two's complement is the same as subtraction.

4. The most significant eight bits of the sum represent the bias switch settings. Remember that logic 1 is off.

The following fully-worked examples show you how to determine bias switch settings.

Local Chassis
Starting Address: 08000

Remote Chassis
Starting Address: 10000

1. Two's complement form of local starting address:

   a. Local starting address in binary:

      0 0000 1000 0000 0000 0000 (byte address)

   b. One's complement:

      1 1111 0111 1111 1111 1111

   c. Truncate the LSB:

      1 1111 0111 1111 1111 111

   d. Add 1 to the truncated complement:

```
  1 1111 0111 1111 1111 111
                       + 1
-------------------------
  1 1111 1000 0000 0000 000
```

2. Write the remote address as a 21-bit binary number and truncate the LSB:

   0 0001 0000 0000 0000 000

3. Add two's complement of local address to the remote address:

```
    0 0001 0000 0000 0000 000
  + 1 1111 1000 0000 0000 000
-------------------------
    0 0000 1000 0000 0000 000
```

4. Use the eight most significant bits of the result to represent the switch settings (1 = off, 0 = on).

SW 1	SW 2	SW 3	SW 4	SW 5	SW 6	SW 7	SW 8
ON	ON	ON	ON	ON	OFF	ON	ON

Example 2-1. Bias Switch Settings for a Positive Bias

The following example shows how to add a negative bias with the bias switches.

Local Chassis
Starting Address: 16000

Remote Chassis
Starting Address: 04000

1. Two's complement of local starting address:

   a. Local starting address in 21-bit binary form:

      0 0001 0110 0000 0000 0000

   b. One's complement:

      1 1110 1001 1111 1111 1111

   c. Truncate LSB:

      1 1110 1001 1111 1111 111

   d. Add one:

```
  1 1110 1001 1111 1111 111
                        +1
  ------------------------
  1 1110 1010 0000 0000 000  (2's complement)
```

2. Write the remote address as a 21-bit binary number and truncate the LSB:

   0 0000 0100 0000 0000 000

3. Add two's complement of local address to the remote address:

```
  0 0000 0100 0000 0000 000
+ 1 1110 1010 0000 0000 000
  ------------------------
  1 1110 1110 0000 0000 000
```

4. Switch settings:

SW 1	SW 2	SW 3	SW 4	SW 5	SW 6	SW 7	SW 8
OFF	OFF	OFF	OFF	ON	OFF	OFF	OFF

Example 2-2. Bias Switch Settings for a Negative Bias

Table 2-5 lists typical TILINE coupler bias switch settings.

Table 2-5. Typical TILINE Coupler LR Bias Switch Settings

Bias Switch 1	2	3	4	5	6	7	8	Bytes	Bias	Kilobytes
0	0	0	0	0	0	0	0	000000	0	
0	0	0	0	0	0	0	1	002000	+8	(−2040)
0	0	0	0	0	0	1	0	004000	+16	(−2032)
0	0	0	0	0	0	1	1	006000	+24	(−2024)
0	0	0	0	0	1	0	0	008000	+32	(−2016)
0	0	0	0	0	1	0	1	00A000	+40	(−2008)
0	0	0	0	0	1	1	0	00C000	+48	(−2000)
0	0	0	0	0	1	1	1	00E000	+56	(−1992)
0	0	0	0	1	0	0	0	010000	+64	(−1984)
0	0	0	0	1	0	0	1	012000	+72	(−1976)
0	0	0	0	1	0	1	0	014000	+80	(−1968)
0	0	0	0	1	0	1	1	016000	+88	(−1960)
0	0	0	0	1	1	0	0	018000	+96	(−1952)
0	0	0	0	1	1	0	1	01A000	+104	(−1944)
0	0	0	0	1	1	1	0	01C000	+112	(−1936)
0	0	0	0	1	1	1	1	01E000	+120	(−1928)
0	0	0	1	0	0	0	0	020000	+128	(−1920)
0	0	0	1	1	0	0	0	030000	+192	(−1856)
0	0	1	0	0	0	0	0	040000	+256	(−1792)
0	0	1	1	0	0	0	0	060000	+384	(−1664)
0	1	0	0	0	0	0	0	080000	+512	(−1536)
0	1	0	1	0	0	0	0	0A0000	+640	(−1408)
0	1	1	0	0	0	0	0	0C0000	+768	(−1280)
0	1	1	1	0	0	0	0	0E0000	+896	(−1152)
1	0	0	0	0	0	0	0	100000	+1024	(−1024)
1	0	0	1	0	0	0	0	120000	+1152	(−896)
1	0	1	0	0	0	0	0	140000	+1280	(−768)
1	0	1	1	0	0	0	0	160000	+1408	(−640)
1	1	0	0	0	0	0	0	180000	+1536	(−512)
1	1	0	1	0	0	0	0	1A0000	+1664	(−384)
1	1	1	0	0	0	0	0	1C0000	+1792	(−256)
1	1	1	1	0	0	0	0	1E0000	+1920	(−128)
1	1	1	1	1	0	0	0	1F0000	+1984	(−64)
1	1	1	1	1	1	0	0	1F8000	+2016	(−32)
1	1	1	1	1	1	1	0	1FC000	+2032	(−16)
1	1	1	1	1	1	1	1	1FE000	+2040	(−8)

Notes:

1 = off

0 = on

2.13 INSTALLING AND CABLING THE TILINE COUPLER LR

Section 2 of the *Model 990A13 Chassis, General Description* manual describes how to install shielded logic boards in a 990A13 chassis. Install the TILINE coupler LR by following the installation procedure for shielded logic boards.

Refer to the TILINE coupler cabling drawing in Section 1. Remember that the direction of the control cable vector is important in any system that has TILINE masters in an expansion chassis.

Because the connectors are close together, installing cables in the lower-numbered connectors first is easier. By working from left to right (from P3 to P6), you always have clear access to the connectors.

If the chassis has no central processor or CRU buffer board, connect the display panel cable to connector P6 as shown in Figure 2-3.

NOTE

Do not mix up the data and address cables. Connectors on these cables are physically identical.

2.14 POWER-UP AND OPERATION

Neither the CRU nor the TILINE expansion system imposes any special power-up or power-down procedures. An expansion chassis has no operating controls on the display panel. An ON-1/OFF-0 switch at the rear of the chassis controls ac power. A green POWER indicator on the display panel lights when ac power is on and the chassis is not reset (TILINE power reset inactive).

There are no operating procedures associated with either the TILINE or the CRU expansion system. Parity indicators on the TILINE couplers are not normally visible to an operator. Parity indicators on both couplers light (red) when either coupler detects a parity error on an interchassis transfer. To verify expansion system operation, run the TLCPLR diagnostic for TILINE expansion systems, and the CRUEXP diagnostic for CRU expansion systems.

NOTE

Do not attempt to load the TILINE coupler diagnostic through a disk system that resides in the expansion chassis. A fault in either coupler can prevent the diagnostic from loading, and the diagnostic cannot provide any insight into the failure.

3

Nonstandard and OEM Configuration Data

3.1 CRU INSTRUCTIONS AND ADDRESSING

CRU operations divide naturally into two categories: those involving a single-bit transfer, and those involving input or output of several data or status bits. The addressing schemes for single-bit and multiple-bit operations are slightly different.

3.1.1 CRU Addressing for Single-Bit Operations

There are three single-bit CRU instructions: test bit (TB), set bit to one (SBO), and set bit to zero (SBZ). For the two output operations, the central processor places a CRU bit address on CRUBIT4 through CRUBIT15, and places the data bit on the CRUBITOUT line. A pulse on the CRU clock line (STORECLK–) loads the data bit into the CRU device.

The TB instruction is an input instruction that transfers the addressed CRU bit from the CRUBITIN line to the status register. There is no CRU clock pulse for a CRU input read operation.

Addresses for single-bit operations are based on a CRU base address in workspace register 12 (R12) and a displacement contained in bits 8 through 15 of the instruction. The displacement allows two's-complement addressing from base address minus 128 bits to base address plus 127 bits.

Figure 3-1 shows the steps involved in address development. Bit 15 of the workspace register is ignored. Bits 3 through 14 are added to the signed displacement from the instruction. The sum is the CRU address for one bit. The address goes out to the backplane on the CRUBIT4–CRUBIT15 lines. Only one CRU device can respond to that address.

Notice that the CRU address on CRUBIT4 through CRUBIT15 is divided into three fields. The first field, CRUBIT4 through CRUBIT6, selects one of the eight possible chassis. Each CRU buffer has decoding logic and a chassis identification jumper to determine whether to further decode the incoming address.

The next field, CRUBIT7 through CRUBIT11, is the module select field. A 5-bit field allows module select outputs 0 through 31. In practice, a 13-slot chassis has wiring for module select outputs 0 through 23. Each module is a block of 16 CRU bits.

The last field, CRUBIT13 through CRUBIT15, selects the individual CRU bit within the 16-bit module. CRUBIT12 through CRUBIT15 are wired to each half-slot in the chassis. The only device that decodes the bit select field is the one that also has an active module select from a CRU buffer or central processor.

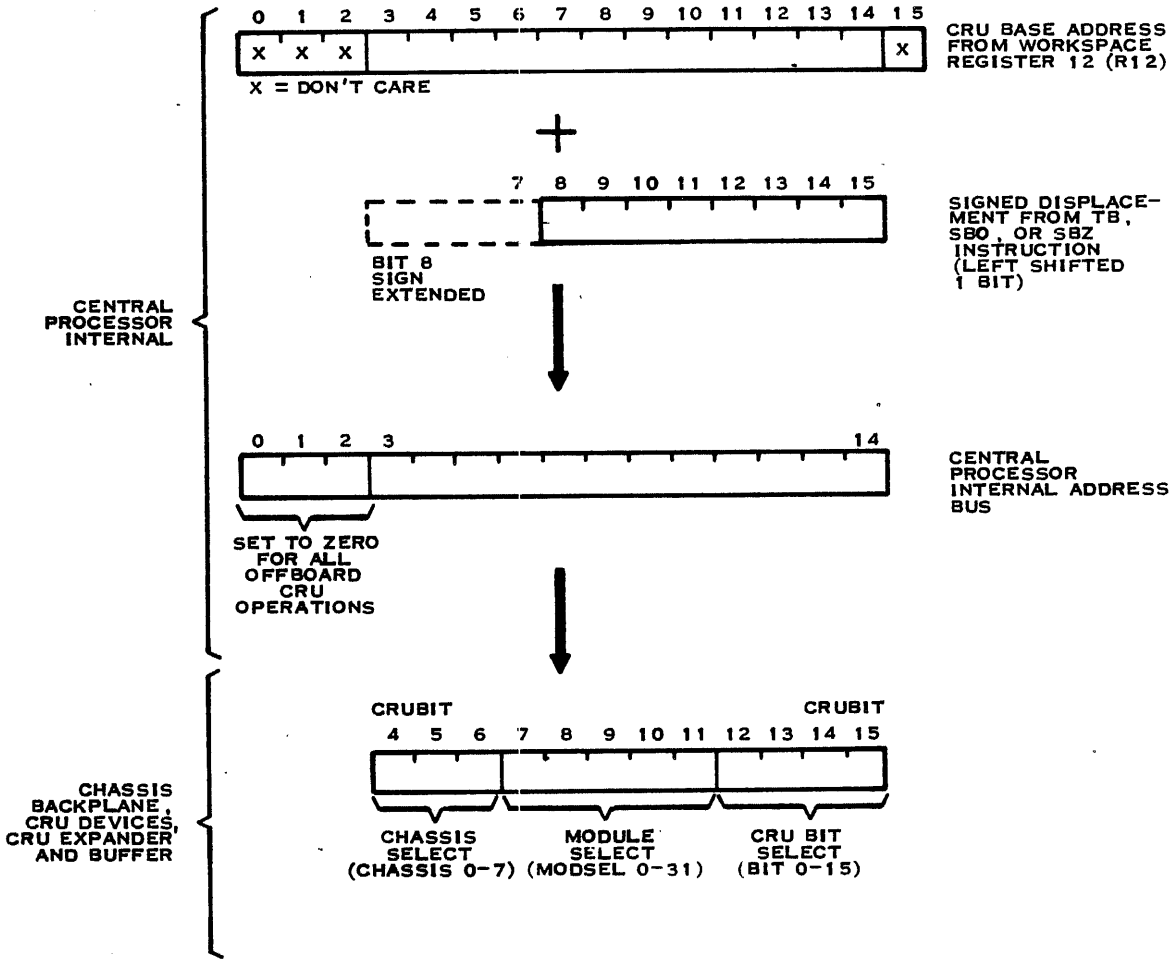


Figure 3-1. CRU Address Development and Format — Single-Bit Instructions

3.1.2 CRU Addressing for Multiple-Bit Operations

A central processor performs two multiple-bit CRU instructions: load communications register (LDCR) and store communications register (STCR). LDCR and STCR can read or write any number of bits from 1 to 16 with a single instruction.

An LDCR instruction includes a memory address for the source data bits, a displacement (from the base address in workspace register 12), and a bit count. This instruction fetches a word from memory and shifts the word to the right while incrementing the CRU address. STORECLK- pulses load each data bit into a register at the destination.

If the LDCR operation involves eight or more bits, the bits come from the right-justified field in the addressed byte of the memory word. This is the upper byte (0 through 7) for even memory addresses and the lower byte (8 through 15) for odd addresses. If the LDCR involves more than eight bits, these bits come from the right-justified field within the whole memory word.

NOTE

A bit count of 0 in an LDCR or STCR command transfers 16 data bits.

Refer to Figure 3-2, which summarizes an LDCR operation. After each bit goes out on the CRUBITOUT line, the CRU address increments for the next bit. This addressing scheme results in an order reversal of bits; bit 15 (or bit 7) of the memory word becomes the lowest-addressed CRU bit in the CRU, and bit 0 becomes the highest-addressed CRU bit.

An STCR instruction, Figure 3-3, transfers data from the CRU to memory. An STCR instruction includes a memory destination address, a displacement from the CRU base address, and a bit count. For 8 bits or less, the incoming CRU data is right-justified in the addressed memory byte, with the leading bits set to 0. For 9 to 16 bits, the incoming data is right-justified in the full word, with the leading bits set to 0. Upon completion, the first bit from the CRU is in the least significant bit of the memory word or byte.

3.2 CRU EXPANSION SYSTEM INTERCONNECTIONS

Figure 3-4 shows the signal connections between the central processor, CRU expander, CRU buffers, and expansion chassis. Table 3-1 summarizes the CRU-related signals in the chassis backplane, and Table 3-2 summarizes the signals between the expander and buffer. Notice that the CRU expander has four sets of identical signals at connectors P3-P6. The suffix on the signal name (A, B, C, or D) identifies connector P3, P4, P5, or P6 respectively.

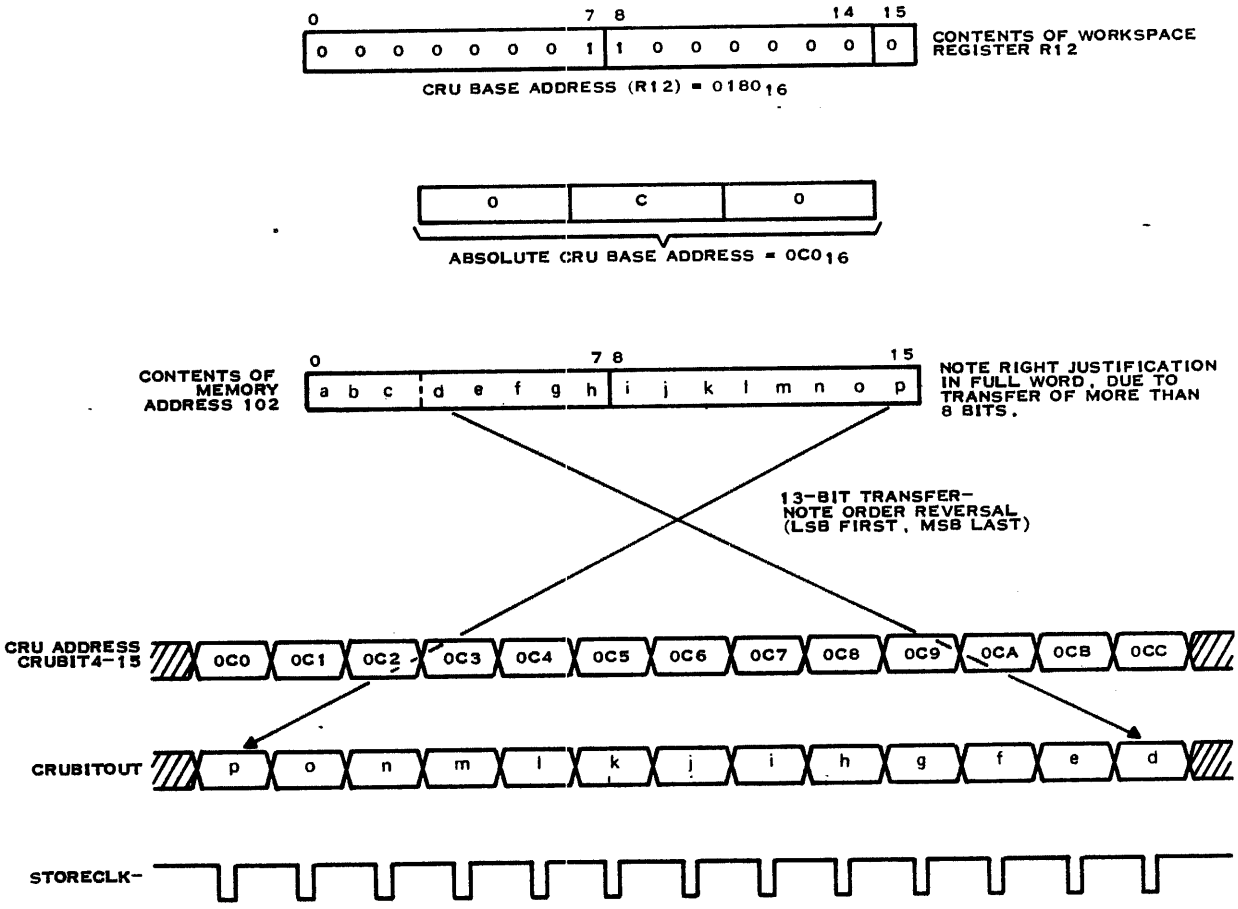


Figure 3-2. Multiple-Bit CRU Output Operation — LDCR

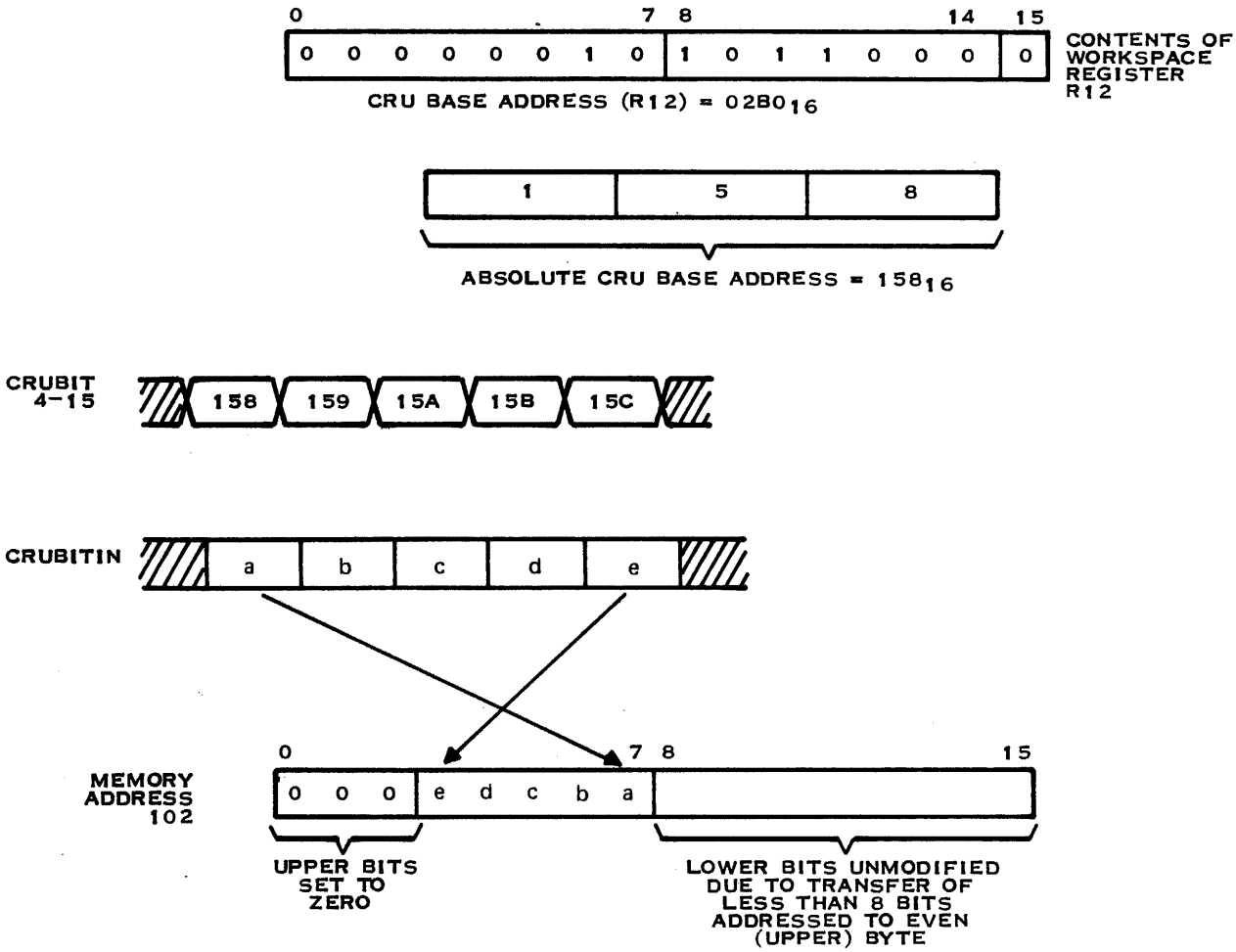
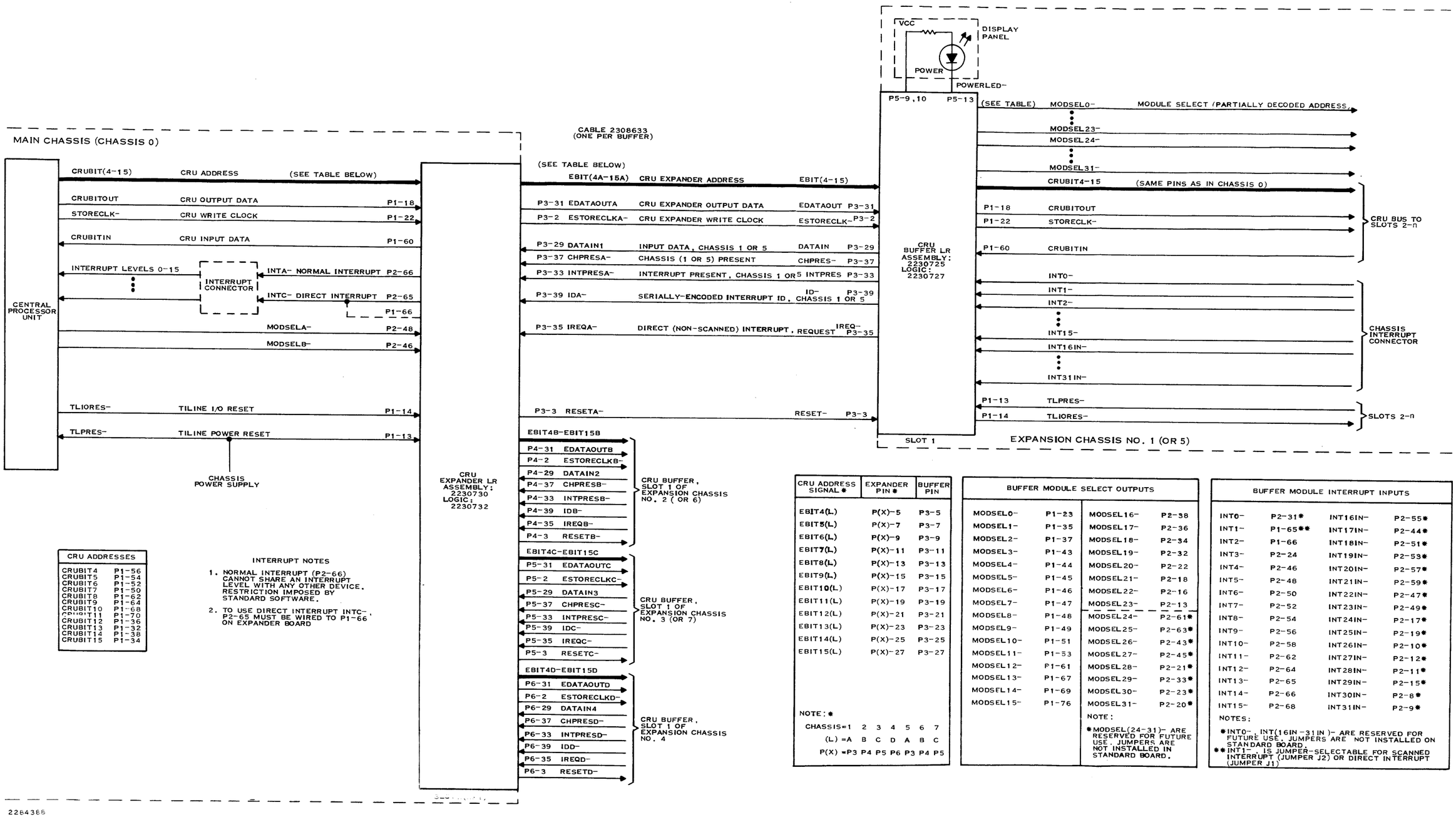


Figure 3-3. Multiple-Bit CRU Input Operation — STCR



CRU ADDRESSES	
CRUBIT4	P1-56
CRUBIT5	P1-54
CRUBIT6	P1-52
CRUBIT7	P1-50
CRUBIT8	P1-62
CRUBIT9	P1-64
CRUBIT10	P1-68
CRUBIT11	P1-70
CRUBIT12	P1-36
CRUBIT13	P1-32
CRUBIT14	P1-38
CRUBIT15	P1-34

INTERRUPT NOTES

1. NORMAL INTERRUPT (P2-66) CANNOT SHARE AN INTERRUPT LEVEL WITH ANY OTHER DEVICE. RESTRICTION IMPOSED BY STANDARD SOFTWARE.
2. TO USE DIRECT INTERRUPT INTC-, P2-65 MUST BE WIRED TO P1-66 ON EXPANDER BOARD

CRU ADDRESS SIGNAL*	EXPANDER PIN*	BUFFER PIN
EBIT4(L)	P(X)-5	P3-5
EBIT5(L)	P(X)-7	P3-7
EBIT6(L)	P(X)-9	P3-9
EBIT7(L)	P(X)-11	P3-11
EBIT8(L)	P(X)-13	P3-13
EBIT9(L)	P(X)-15	P3-15
EBIT10(L)	P(X)-17	P3-17
EBIT11(L)	P(X)-19	P3-19
EBIT12(L)	P(X)-21	P3-21
EBIT13(L)	P(X)-23	P3-23
EBIT14(L)	P(X)-25	P3-25
EBIT15(L)	P(X)-27	P3-27

NOTE: *

CHASSIS=1	2	3	4	5	6	7
(L) =A	B	C	D	A	B	C
P(X) =P3	P4	P5	P6	P3	P4	P5

BUFFER MODULE SELECT OUTPUTS

MODSEL0-	P1-23	MODSEL16-	P2-38
MODSEL1-	P1-35	MODSEL17-	P2-36
MODSEL2-	P1-37	MODSEL18-	P2-34
MODSEL3-	P1-43	MODSEL19-	P2-32
MODSEL4-	P1-44	MODSEL20-	P2-22
MODSEL5-	P1-45	MODSEL21-	P2-18
MODSEL6-	P1-46	MODSEL22-	P2-16
MODSEL7-	P1-47	MODSEL23-	P2-13
MODSEL8-	P1-48	MODSEL24-	P2-61*
MODSEL9-	P1-49	MODSEL25-	P2-63*
MODSEL10-	P1-51	MODSEL26-	P2-43*
MODSEL11-	P1-53	MODSEL27-	P2-45*
MODSEL12-	P1-61	MODSEL28-	P2-21*
MODSEL13-	P1-67	MODSEL29-	P2-33*
MODSEL14-	P1-69	MODSEL30-	P2-23*
MODSEL15-	P1-76	MODSEL31-	P2-20*

NOTE:
*MODSEL(24-31)- ARE RESERVED FOR FUTURE USE. JUMPERS ARE NOT INSTALLED IN STANDARD BOARD.

BUFFER MODULE INTERRUPT INPUTS

INT0-	P2-31*	INT16IN-	P2-55*
INT1-	P1-65**	INT17IN-	P2-44*
INT2-	P1-66	INT18IN-	P2-51*
INT3-	P2-24	INT19IN-	P2-53*
INT4-	P2-46	INT20IN-	P2-57*
INT5-	P2-48	INT21IN-	P2-59*
INT6-	P2-50	INT22IN-	P2-47*
INT7-	P2-52	INT23IN-	P2-49*
INT8-	P2-54	INT24IN-	P2-17*
INT9-	P2-56	INT25IN-	P2-19*
INT10-	P2-58	INT26IN-	P2-10*
INT11-	P2-62	INT27IN-	P2-12*
INT12-	P2-64	INT28IN-	P2-11*
INT13-	P2-65	INT29IN-	P2-15*
INT14-	P2-66	INT30IN-	P2-8*
INT15-	P2-68	INT31IN-	P2-9*

NOTES:
*INT0-, INT(16IN-31IN)- ARE RESERVED FOR FUTURE USE. JUMPERS ARE NOT INSTALLED ON STANDARD BOARD.
**INT1- IS JUMPER-SELECTABLE FOR SCANNED INTERRUPT (JUMPER J2) OR DIRECT INTERRUPT (JUMPER J1)

2264388

Figure 3-4. CRU Expansion System Signal Connections

Table 3-1. CRU Interface Signals in the Chassis Backplane

Signature	Slot 1 Pin Number	Slot 2-13 Pin Number	Function
CRUBIT4	P1-56	P1-56	Address bits generated by the processor to select a particular chassis (bits 4-6), a 16-bit module within that chassis (bits 7-11), and a particular bit from that module (bits 12-15).
CRUBIT5	P1-54	P1-54	
CRUBIT6	P1-52	P1-52	
CRUBIT7	P1-50	P1-50	
CRUBIT8	P1-62	P1-62	
CRUBIT9	P1-64	P1-64	
CRUBIT10	P1-68	P1-68	
CRUBIT11	P1-70	P1-70	
CRUBIT12	P1-36	P1-36, P2-36	
CRUBIT13	P1-32	P1-32, P2-32	
CRUBIT14	P1-38	P1-38, P2-38	
CRUBIT15	P1-34	P1-34, P2-34	
CRUBITOUT	P1-18	P1-18, P2-18	Serial data line for transfer of data from the processor to the addressed CRU bit(s). This line is active only when STORECLK- goes low.
CRUBITIN	P1-60	P1-60, P2-60	Serial data line for transfer of data from the addressed CRU bit(s) to the processor. This line must be driven by an open-collector gate and only when the module is selected.
STORECLK-	P1-22	P1-22, P2-22	An active-when-low pulse that indicates to the selected CRU module that the operation is a write (SBO, SBZ, or LDCR) operation. This pulse transfers the data from the CRUBITOUT line into a holding flip-flop that is the CRU bit.
MODSEL0-	P1-23	Slot 13,P2-48	Module select signals generated by the processor from address bits 6-10 (CRUBITS 7-11) for use within the main or an expansion chassis. Note that P1 in each slot of the backpanel receives one module select signal, whereas P2 receives two module select signals. This configuration permits P2 to use 32 bits of the CRU. Note that pin 48 of successive P2 connectors in the chassis slots is connected to even-numbered module select signals and, at the CRU circuit board level, carries a MODSELA- signature. Pin 46 of successive P2 connectors in the chassis slots is connected to P1,
MODSEL1-	P1-35	Slot 13, P1-48 and P2-46	
MODSEL2-	P1-37	P2-48	
MODSEL3-	P1-43	Slot 12, P1-48 and P2-46	
MODSEL4-	P1-44	Slot 11, P2-48	
MODSEL5-	P1-45	Slot 11, P1-48 and P2-46	
MODSEL6-	P1-46	Slot 10, P2-48	
MODSEL7-	P1-47	Slot 10, P1-48 and P2-46	

Table 3-1. CRU Interface Signals in the Chassis Backplane (Continued)

Signature	Slot 1 Pin Number	Slot 2-13 Pin Number	Function
MODSEL8- MODSEL9-	P1-48 P1-49	Slot 9, P2-48 Slot 9, P1-48 and P2-46	pin 48 of that slot, then to an odd-numbered module select signal, and carries a signature of MODSELB-. Pin P1-48 is not used when a full-sized CRU circuit board is installed in a chassis slot.
MODSEL10- MODSEL11-	P1-51 P1-53	Slot 8, P2-48 Slot 8, P1-48 and P2-46	
MODSEL12- MODSEL13-	P1-61 P1-67	Slot 7, P2-48 Slot 7, P1-48 and P2-46	
MODSEL14- MODSEL15-	P1-69 P1-71	Slot 6, P2-48 Slot 6, P1-48 and P2-46	
MODSEL16- MODSEL17-	P2-38 P2-36	Slot 5, P2-48 Slot 5, P1-48 and P2-46	
MODSEL18- MODSEL19-	P2-34 P2-32	Slot 4, P2-48 Slot 4, P1-48 and P2-46	
MODSEL20- MODSEL21-	P2-22 P2-18	Slot 3, P2-48 Slot 3, P1-48 and P2-46	
MODSEL22- MODSEL23-	P2-16 P2-13	Slot 2, P2-48 Slot 2, P1-48 and P2-46	
MODSEL24- MODSEL25-	P2-61 P2-63		
MODSEL26- MODSEL27-	P2-43 P2-45		
MODSEL28- MODSEL29-	P2-21 P2-33		
MODSEL30- MODSEL31-	P2-23 P2-20		

Table 3-2. CRU Expander/Buffer LR Interface Signals

Signature	P3-P6 Pin Number	Function
CHPRESA-	27	Chassis present. Low to indicate that the expansion chassis is connected and has power on, and that the TILINE power reset (TLPRES-) is not active.
DATAIN1	29	Data in from buffer. Serial CRU input data from the expansion chassis backplane.
EBIT4A	5	Expander CRU address output bits 4-15. Equivalent to CRUBIT 4-15 from the main chassis.
EBIT5A	7	
EBIT6A	9	
EBIT7A	11	
EBIT8A	13	
EBIT9A	15	
EBIT10A	17	
EBIT11A	19	
EBIT12A	21	
EBIT13A	23	
EBIT14A	25	
EBIT15A	27	
EDATAOUTA	31	CRU expander output data. Equivalent to CRUBITOUT from the main chassis.
ESTORECLKA-	2	CRU expander store clock. Equivalent to STORECLK- from the main chassis.
IDA-	39	Expansion chassis interrupt identification. Serially-encoded interrupt ID from the buffer. Identifies the active interrupt level in the expansion chassis. The data on this line is part of the serial interrupt vector that the central processor reads from the CRU expander.
INTPRESA-	33	Interrupt present in expansion chassis. Interrupt input from the CRU buffer. A scanner within the buffer identifies the active interrupt level. See IDA-.
IREQA-	35	Direct (non-scanned) interrupt present in the expansion chassis. Bypasses the interrupt scanner and the serial interrupt vector. The central processor can process this interrupt faster than scanned interrupts, but there can be only one direct interrupt level in each expansion chassis.

Table 3-2. CRU Expander/Buffer LR Interface Signals (Continued)

Signature	P3-P6 Pin Number	Function
RESETA-	3	Reset expansion chassis. Causes the CRU buffer to issue a TILINE I/O reset (TLIORES-) in the expansion chassis. A jumper option on the CRU expander selects either TILINE I/O reset or TILINE power reset as the source for RESETA-.

Note:

The suffix letter on each signature identifies the CRU expander I/O connector as follows:

A — P3

B — P4

C — P5

D — P6

3.3 CRU EXPANDER AND CRU BUFFER BLOCK DIAGRAM

Figure 3-5 is a block diagram of the CRU expansion system. This diagram shows the interrupt handling logic, signal line drivers/receivers, and the jumper options.

3.3.1 CRU Interrupt Handling

Each half-slot position in an expansion chassis has the potential to generate an interrupt. All of the individual interrupt lines go to the interrupt connector above slot 1. A plug-in interrupt board assigns interrupts to the interrupt level inputs of slot 1.

A standard CRU buffer board can handle interrupt levels 2 through 15 with no jumper modifications. Nonstandard jumpers allow expansion to 32 interrupt levels.

A free-running clock (approximately 3.2 megahertz) drives a scan counter. The scan counter and interrupt scanner multiplexers scan the inputs looking for an active interrupt. An active interrupt latches up an interrupt present flip-flop and stops the scan counter.

A latched interrupt present signal goes to the CRU expander. In a similar manner, the expander scans all four possible expansion chassis interrupts. The active interrupt is again latched and the expander interrupts the central processor.

The central processor can tell which expander supplied the interrupt, but needs more information to identify the source. The CRU expansion interrupt vectors, Figure 3-6, provide the additional information.

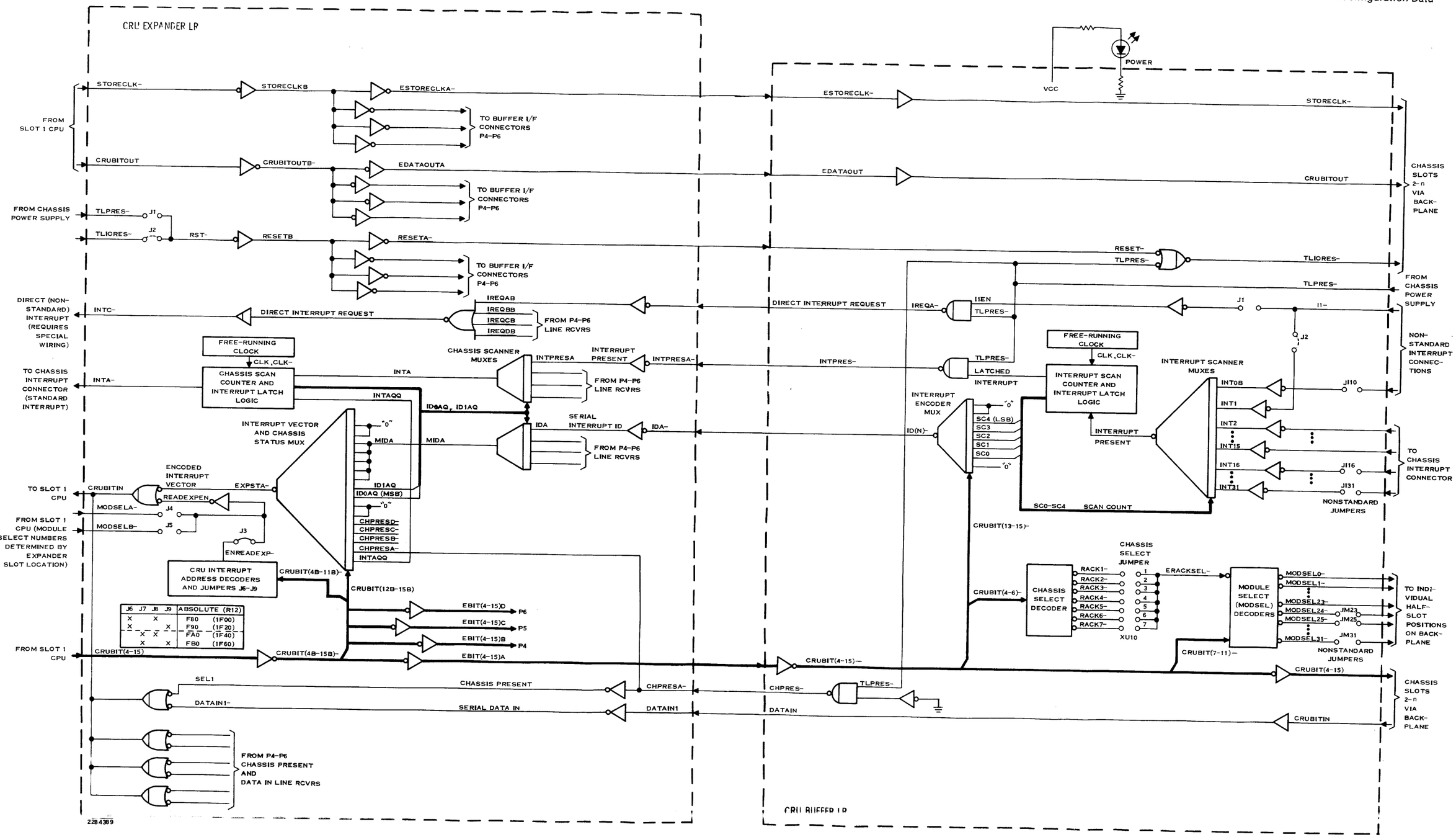


Figure 3-5. CRU Expander and CRU Buffer Block Diagram

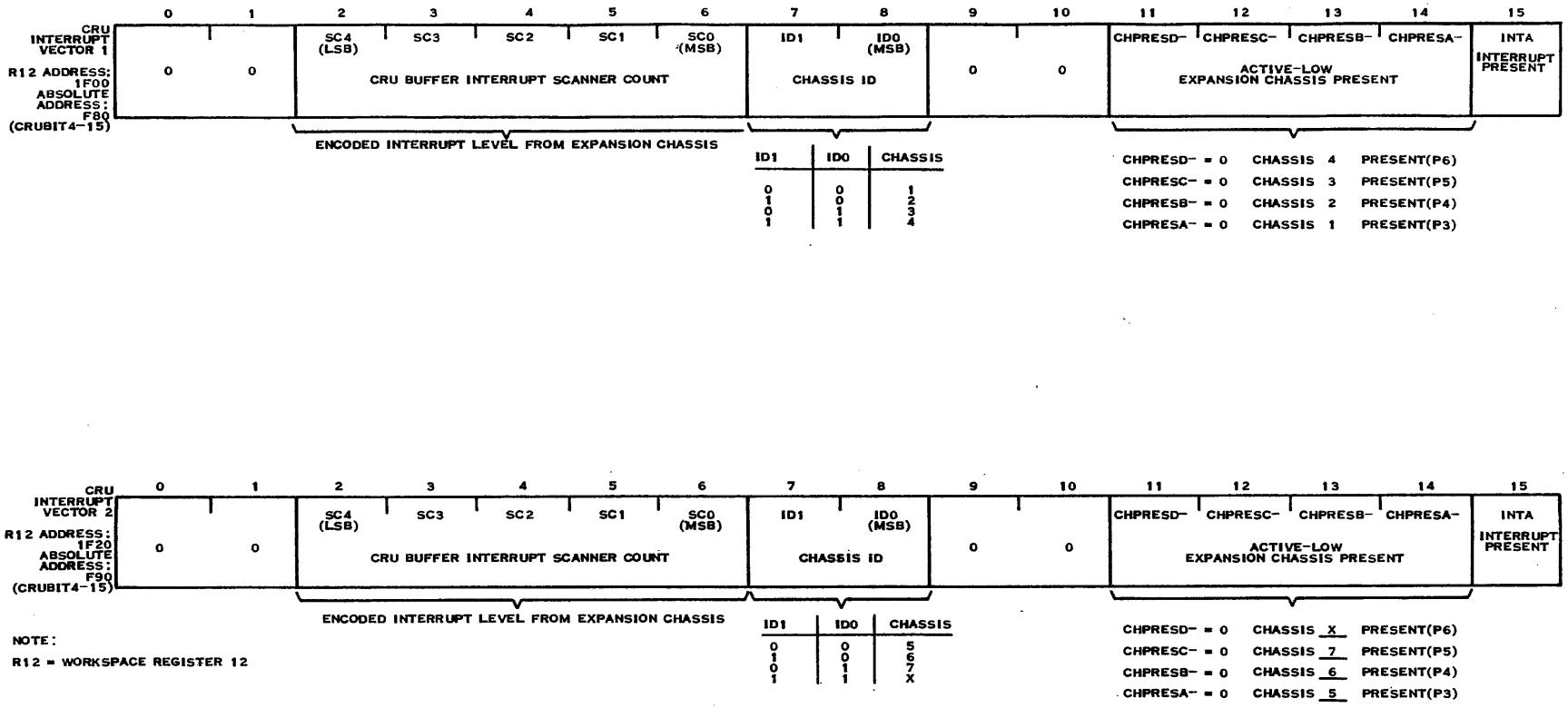


Figure 3-6. CRU Expansion Interrupt Vectors

A CRU expansion vector is a 16-bit CRU word located at a reserved CRU base address. The standard base addresses in workspace register 12 are $1F00_{16}$ for chassis 1 through 4 and $1F20_{16}$ for chassis 5 through 7. A chart on the CRU expander block diagram shows the jumper assignments for each interrupt vector base address.

The central processor can read an entire interrupt vector with an STCR instruction, or check an individual status bit with a TB instruction.

Assume that the central processor reads the entire vector. The CRU buffer and the expander work together to supply the interrupt vector word. The first two bits are hardwired zeros. The next five bits are the scan count from the CRU buffer. They identify the active interrupt level in the expansion chassis.

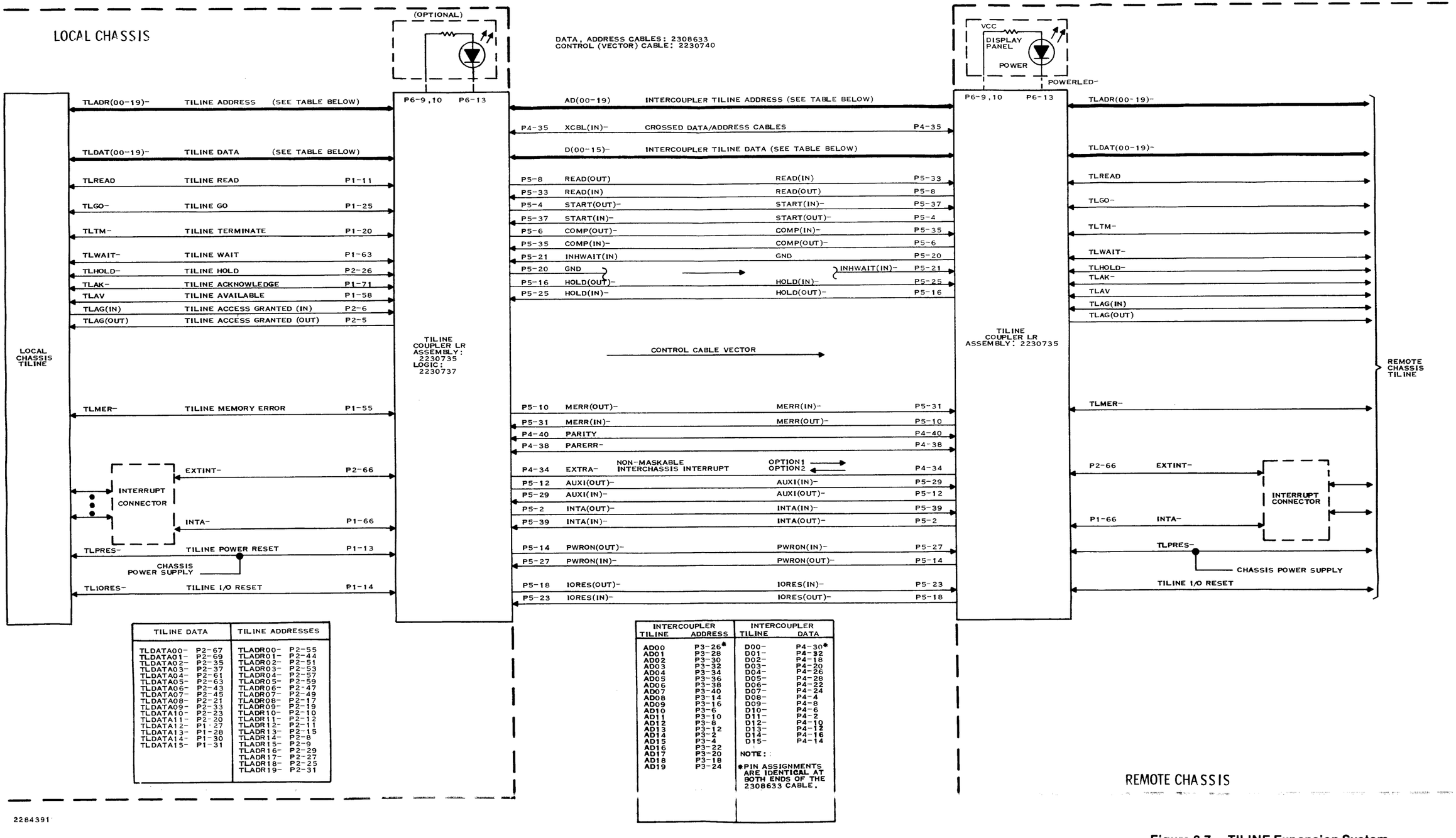
The next two bits are the chassis identifier from the chassis scan counter, followed by two more hardwired zeros. Bits 11 through 14 are the active-low chassis present bits. Bit 15 is the interrupt present bit.

After checking the expansion interrupt vector, the central processor must clear the interrupt by a CRU output to the device that originated the interrupt. Buffer and expander interrupt logic clear out after the original interrupt goes inactive.

A jumper option (J1) on the buffer board allows you to bypass the entire interrupt scanning system. However, the direct interrupts are limited to one level per expansion chassis.

3.4 TILINE EXPANSION SYSTEM INTERCONNECTIONS

Figure 3-7 shows the signal connections for a set of TILINE couplers. Table 3-3 summarizes TILINE signals in the chassis backplane, and Table 3-4 summarizes the signals between the couplers.



TILINE DATA		TILINE ADDRESSES	
TLDATA00-	P2-67	TLADR00-	P2-55
TLDATA01-	P2-69	TLADR01-	P2-44
TLDATA02-	P2-35	TLADR02-	P2-51
TLDATA03-	P2-37	TLADR03-	P2-53
TLDATA04-	P2-61	TLADR04-	P2-57
TLDATA05-	P2-63	TLADR05-	P2-59
TLDATA06-	P2-43	TLADR06-	P2-47
TLDATA07-	P2-45	TLADR07-	P2-49
TLDATA08-	P2-21	TLADR08-	P2-17
TLDATA09-	P2-33	TLADR09-	P2-19
TLDATA10-	P2-23	TLADR10-	P2-10
TLDATA11-	P2-20	TLADR11-	P2-12
TLDATA12-	P1-27	TLADR12-	P2-11
TLDATA13-	P1-28	TLADR13-	P2-15
TLDATA14-	P1-30	TLADR14-	P2-8
TLDATA15-	P1-31	TLADR15-	P2-9
		TLADR16-	P2-29
		TLADR17-	P2-27
		TLADR18-	P2-25
		TLADR19-	P2-31

INTERCOUPLER TILINE ADDRESS		INTERCOUPLER TILINE DATA	
AD00	P3-26*	D00-	P4-30*
AD01	P3-28	D01-	P4-32
AD02	P3-30	D02-	P4-18
AD03	P3-32	D03-	P4-20
AD04	P3-34	D04-	P4-26
AD05	P3-36	D05-	P4-28
AD06	P3-38	D06-	P4-22
AD07	P3-40	D07-	P4-24
AD08	P3-14	D08-	P4-4
AD09	P3-16	D09-	P4-8
AD10	P3-6	D10-	P4-6
AD11	P3-10	D11-	P4-2
AD12	P3-8	D12-	P4-10
AD13	P3-12	D13-	P4-12
AD14	P3-2	D14-	P4-16
AD15	P3-4	D15-	P4-14
AD16	P3-22		
AD17	P3-20	NOTE:	
AD18	P3-18	*PIN ASSIGNMENTS ARE IDENTICAL AT BOTH ENDS OF THE 2308633 CABLE.	
AD19	P3-24		

2284391

Figure 3-7. TILINE Expansion System Signal Connections

Table 3-3. TILINE Signals in the Chassis Backplane

Signature	Pin No.	Definition
TLGO-	P1-25	TILINE Go — Initiates all data transfers when transition from high (3.0 V) to low (1.0V) occurs.
TLREAD	P1-11	TILINE Read — When high (3.0 V), designates a read from SLAVE operation; when low (1.0 V), designates a write to SLAVE operation.
TLADR00-	P2-55	TILINE Address — Defines the location of data during a fetch or store operation. When high (≥2.0 V), the corresponding address bit is a zero; when low (≤0.8 V), the corresponding address bit is a one.
01-	P2-44	
02-	P2-51	
03-	P2-53	
04-	P2-57	
05-	P2-59	
06-	P2-47	
07-	P2-49	
08-	P2-17	
09-	P2-19	
10-	P2-10	
11-	P2-12	
12-	P2-11	
13-	P2-15	
14-	P2-8	
15-	P2-9	
16-	P2-29	
17-	P2-27	
18-	P2-25	
TLADR19-	P2-31	
TLDAT00-	P2-67	TILINE Data — Bidirectional data lines that, when high (≥2.0 V), represent zero data bits and, when low (≤0.8 V), represent one data bit.
01-	P2-69	
02-	P2-35	
03-	P2-37	
04-	P2-61	
05-	P2-63	
06-	P2-43	
07-	P2-45	
08-	P2-21	
09-	P2-33	
10-	P2-23	
11-	P2-20	
12-	P1-27	
13-	P1-28	
14-	P1-30	
TLDAT15-	P1-31	
TLTM-	P1-20	TILINE Terminate — When low (1.0 V), indicates that the SLAVE device has completed the requested operation.
TLMER-	P1-55	TILINE Memory Error — When low (≤0.8 V), indicates that a non-recoverable error has occurred during a memory read operation.

Table 3-3. TILINE Signals in the Chassis Backplane (Continued)

Signature	Pin No.	Definition
TLAG(in)	P2-6	TILINE Access Granted — When high (≥2.0 V), this signal indicates that no higher priority device has requested use of the TILINE. When low (≤0.8 V), this signal prevents the receiving device from gaining access to the TILINE bus.
TLAG(out)	P2-5	TILINE Access Granted — When high (≥2.0 V), this signal indicates that neither the sending device nor any higher priority device is requesting use of the TILINE. When low (≤0.8 V), this signal indicates that either the sending device or some higher priority device is requesting use of the TILINE bus and prevents all lower priority devices from gaining access to the bus.
TLAK-	P1-71	TILINE Acknowledge — When high (3.0 V), this signal indicates that no TILINE device has been recognized as the next device to use the TILINE. When low (1.0 V), this signal indicates that some TILINE device has requested access, has been recognized, and is waiting for the bus to become available.
TLAV	P1-58	TILINE Available — When high (3.0 V), this signal indicates that no TILINE device is using the bus. When low (1.0 V), this signal indicates that the TILINE bus is busy.
TLWAIT-	P1-63	TILINE Wait — A normally high (3.0 V) signal that, when low (1.0 V), temporarily suspends all TILINE master devices from using the TILINE bus. This signal is generated by a bus coupler to allow it to use the bus as the highest priority user.
TLIORES-	P1-14 P2-14	TILINE I/O Reset — A normally high (≥2.0 V) signal that, when low (≤0.8 V), halts and resets all TILINE I/O devices. This signal is a 100- to 500-nanosecond pulse generated by the RESET switch on the control console or by the execution of a Reset (RSET) instruction in the central processor.
TLPRES-	P1-13 P2-13	TILINE Power Reset — A normally high (≥2.0 V) signal that goes low (≤0.8 V) to reset all TILINE devices and inhibit critical lines to external equipment. The signal is generated by the power supply at least 10 microseconds before dc voltages begin to fail during power-down, and until dc voltages are stable during power-up.
TLPFWP-	P1-16 P2-16	TILINE Power Failure Warning Pulse — A two-millisecond pulse preceding TLPRES-. When low (≤0.8 V), this signal indicates that a power-down sequence is in progress, allowing the central processor to perform a power-failure interrupt subroutine.
TLCACHEN	P1-73	TILINE Cache Enable — The cache memory in the cache controller is enabled when this signal is high.

Table 3-4. TILINE Intercoupler Signals

Signature	Pin No.	Definition
AD00	P3-26	Intercoupler address bus, bits 00-19. Consists of the TILINE address from the source chassis backplane, plus the 8-bit coupler bias switch setting, added to the most significant 8 bit positions (00-07) for memory addresses between the upper and lower address limit. For TPCS addresses, consists of addresses in blocks as determined by option switches 8-12. May also be mapped to TPCS block 0 of the destination chassis. Memory address bias switch settings do not affect TPCS addresses.
01	28	
02	30	
03	32	
04	34	
05	36	
06	38	
07	40	
08	14	
09	16	
10	6	
11	10	
12	8	
13	12	
14	2	
15	4	
16	22	
17	20	
18	18	
AD19	P3-24	
AUXI(IN)- AUXI(OUT)-	P5-29 P5-12	Multiprocessor interrupt (attention or acknowledge) to auxiliary processor (AUXI(OUT)) or from the auxiliary processor (AUXI(IN)). Enabled and initiated by TPCS control in the host coupler.
COMP(IN)- COMP(OUT)-	P5-6 P5-35	Operation complete from the remote chassis. Informs the local coupler that the requested TILINE cycle finished (TLTM-) in the remote chassis. The local coupler responds with a TILINE terminate in the local chassis. On a read cycle, the local coupler also latches the incoming data from D(00-15)-.
D00-	P4-30	Intercoupler TILINE data.
01-	32	
02-	18	
03-	20	
04-	26	
05-	28	
06-	22	
07-	24	
08-	4	
09-	8	
10-	6	
11-	2	
12-	10	
13-	12	
14-	16	
D15-	P4-14	
EXTRA-	P4-34	Nonmaskable interchassis interrupt.

Table 3-4. TILINE Intercoupler Signals (Continued)

Signature	Pin No.	Definition
HOLD(IN)- HOLD(OUT)-	P5-25 P5-16	Hold TILINE access between read and write cycle. Generated during software interlock instructions.
INHWAIT(IN)	P5-21	Inhibit TILINE wait cycles. Vector direction of control cable determines whether waits are inhibited or not.
INTA(IN)- INTA(OUT)-	P5-39 P5-2	Maskable interchassis interrupt.
IORES(IN)- IORES(OUT)-	P5-23 P5-18	Interchassis I/O reset.
MERR(IN)- MERR(OUT)-	P5-31 P5-10	Memory error.
PARERR-	P4-38	Latched parity error status.
PARITY-	P4-40	Remote coupler parity bit. Provided for data/address parity comparisons between local and remote coupler.
PWRON(IN)- PWRON(OUT)-	P5-27 P5-14	Chassis power on. Inverted form of chassis TLPRES-.
READ(IN) READ(OUT)	P5-33 P5-8	TILINE coupler read/write command to remote coupler. Latched version of TLREAD from chassis backplane. OSW6, if closed, holds READ(OUT) high.
START(IN)- START(OUT)-	P5-37 P5-4	Start remote TILINE cycle. Command to the remote coupler to start a TILINE read or write cycle in the remote chassis.
XCBL(IN)-	P4-35	Crossed cables indicator. Low if the data and address cables are cross-connected.

Notes:

1. Remote chassis and local chassis are relative terms; local refers to the chassis and coupler that initiate an operation with a START- command, while remote refers to the chassis and coupler that respond and return the COMP command.

2. Host and auxiliary are terms set by software choice and controlled via the TPCS.

3.5 COUPLER OPERATION AND TIMING

The following paragraphs provide basic and detailed descriptions of the TILINE coupler operating cycle.

3.5.1 Summary of Coupler Operating Cycle

Assume that a central processor or disk controller has to perform a read or write operation to memory in the remote chassis. The processor or controller acts as a TILINE master to gain access to the local TILINE bus. The local TILINE coupler, acting as a slave, recognizes that the address is in range and starts a remote coupler cycle.

The remote coupler, acting as a TILINE master, gains access to the TILINE in the remote chassis and performs the read or write operation. The remote coupler notifies the local coupler that the remote TILINE cycle is complete. The local coupler, remote coupler, and local master go through a series of disengagement steps to complete the cycle.

3.5.2 Detailed TILINE Coupler Operating Cycle

A timing diagram, Figure 3-8, and the following steps describe the coupler operating cycle in detail:

1. The local TILINE master gains access to the TILINE utilizing the bus acquisition signals. Once the master has the TILINE, it gates the address (and data, if it is a write cycle) on the TILINE and issues a TILINE go (TLGO-) signal.

2. The local coupler delays TLGO- until it has had sufficient time to decode the address lines. If the address is within the allowable limits, the coupler issues an internal address OK signal.

3. The delayed TLGO- and the address OK signals set the coupler START latch. The START latch gates biased address signals AD00 through AD19 to the remote coupler. If this is a write cycle, the coupler also gates data lines D00- through D15- to the remote coupler.

4. The local coupler issues a START(OUT)- signal to the remote coupler.

5. START(OUT)- appears as START(IN)- at the remote coupler. START(IN)- sets the device access request latch to initiate a TILINE master cycle.

6. The remote coupler acts as a master to gain access to the remote chassis TILINE:

   a. The remote coupler withdraws TILINE access granted by forcing it low. This signal is daisy chained, and the effect of pulling it low is to signal lower priority devices that this master wants the TILINE.

   b. The remote coupler waits for at least 200 nanoseconds after forcing TLAG(OUT) low. The delay allows the daisy-chained signal to propagate to the lowest priority device.

c. After the TLAG delay, the coupler examines the TLAK- signal. If TLAK- is high, the coupler will issue TLAK-. If TLAK- is low, the coupler waits for it to go high. The coupler cannot advance through the access cycle until it receives the high TILINE acknowledge signal.

d. After issuing the TLAK- signal, the coupler waits for the TLAV signal to go high. When the TLAV signal goes high, the coupler sets it low and releases TLAG- and TLAK-. The coupler has access to the bus and gates the address lines (from the remote coupler) onto the bus.

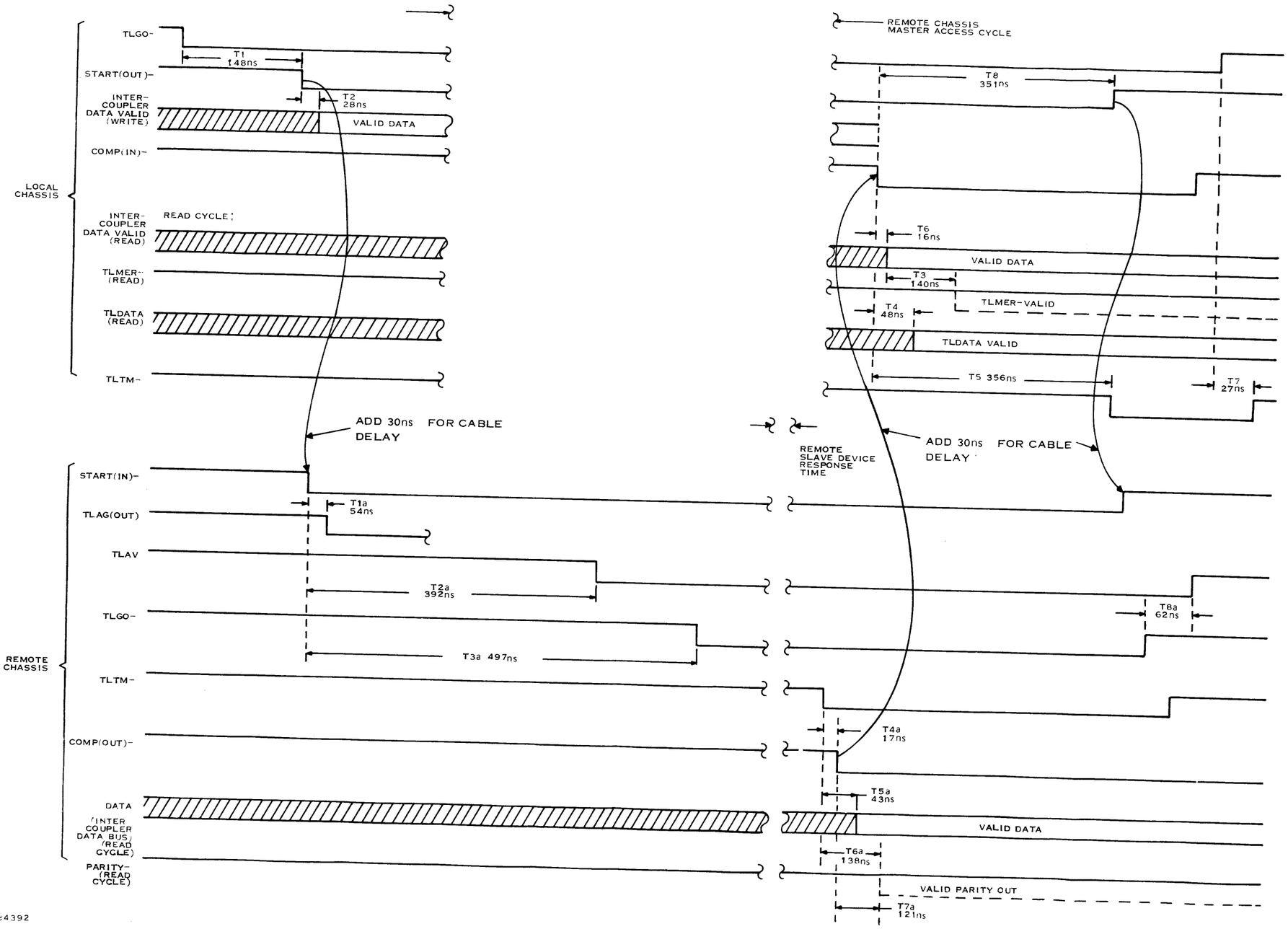
e. The coupler waits for both TLGO- AND TLTM- to go high, then issues the TLGO- and, for a write cycle, gates the incoming data onto the TLDAT00- through TLDAT15- lines. The coupler waits for a TILINE terminate (TLTM-) signal from the slave device.

7. When TLTM- is received, the remote coupler issues a COMP(OUT-) signal to the local coupler.

8. The COMP(OUT)- signal is received as COMP(IN)- at the local chassis. COMP(IN)- clears the START(OUT)- signal and issues TLTM- on the local TILINE. COMP(IN)- also causes the local chassis to latch the data lines from the remote chassis for a read cycle.

9. The release of START(OUT)- causes the remote coupler to release the TLGO- signal in the remote chassis.

10. The master that initiated the entire process releases TLGO- in the local chassis. In response, the local coupler completes its slave cycle by releasing TLTM-. If the current cycle is a read cycle, the local coupler also disables its data line drivers on release of TLGO-.



NOTE:
ALL TIMES ARE TYPICAL MEASURED VALUES.

Figure 3-8. TILINE Coupler Timing — Local and Remote Chassis

3.5.3 Exceptions to the TILINE Coupler Operating Cycle

TILINE wait, hold, and timeout conditions alter the normal operating cycle.

3.5.3.1 TILINE Wait. In multiprocessor systems there is a very high probability that two central processors will attempt a data transfer across the coupler link simultaneously. The couplers recognize a preferred direction to resolve these conflicts. The coupler that has master responsibility in the preferred direction issues a TLWAIT- signal. This is a signal to all other TILINE masters to relinquish control of the bus immediately pending a high priority data transfer from the coupler. The coupler goes ahead and performs the data transfer.

When the coupler releases TLWAIT-, the master that had to release the bus can continue. This action causes no state change in master devices and, with the exception of its TILINE drivers, the master device does not realize that TLWAIT- has been asserted. Figure 3-9 summarizes the timing of a wait operation.

3.5.3.2 TILINE Hold. Certain instructions are intended to be used as software interlocks. These instructions read a word during one cycle and write it back in the next cycle. In multiple-processor systems it would be possible for one processor to modify a memory location between the read and write cycles of another processor's interlock instructions.

In order to prevent this, the central processor issues a TLHOLD- signal on the backplane. TLHOLD- is used and propagated by the TILINE couplers in order to maintain the effectiveness of software interlock instructions. TLHOLD- prevents couplers from releasing access to the TILINE during software interlock instructions.

3.5.3.3 TILINE Time-Out. The TILINE coupler contains an on-board TILINE time-out circuit in the master control logic. If TLTM- has not been received in 32 microseconds (typical), a TILINE time-out unlocks the system. The coupler TILINE time-out is a fail-safe circuit in case the master that initiated the TILINE cycle fails to detect an error and to issue a TILINE time-out. All masters other than the TILINE coupler are specified to have a TILINE time-out occur after 10 microseconds.

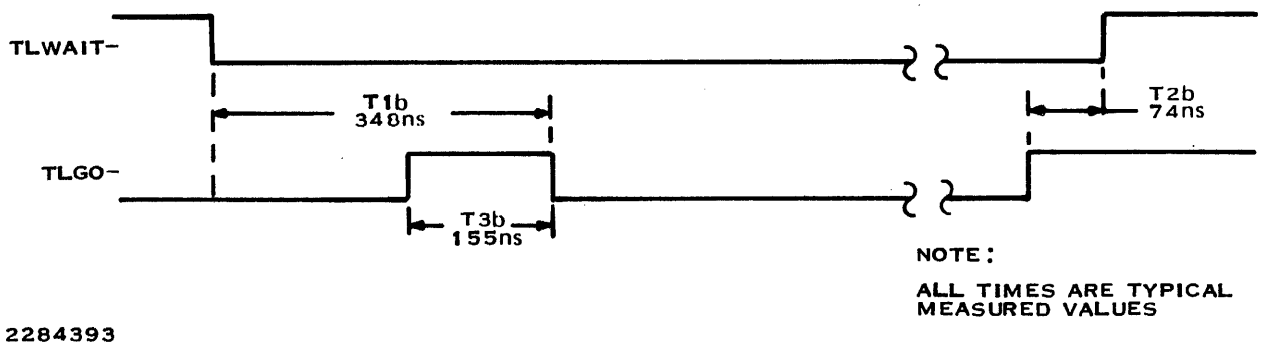


Figure 3-9. TILINE Coupler Wait Timing

3.6 TILINE COUPLER CHASSIS RESET PROVISIONS

Figure 3-10 summarizes the TILINE coupler provisions for local and remote I/O reset. A local I/O reset can be accomplished in one of two ways:

- Hardwired I/O Reset — A TLIORES- signal or a TLPRES- signal in the local chassis always causes a local coupler IORES- signal to be issued.
- Software I/O Reset — A bit in the TPCS which, when written to, issues an I/O reset in the local chassis.

A remote I/O reset can also be accomplished in one of two ways:

- Hardwired Remote I/O Reset — A switch (OSW3) is provided on the coupler which enables a local chassis I/O reset to the remote coupler. Closing a switch in the remote chassis (OSW4) is also required to allow the inter-coupler reset onto the TILINE. OSW3 and OSW4 should *never* both be closed on any one coupler.

  Do not close OSW4 in a chassis that contains a central processor. The processor will not be aware of the incoming reset and may react unpredictably.

WARNING

In a nonmultiprocessing configuration, the Hardwired Remote I/O Reset is often enabled across the TILINE couplers from the main chassis to the expansion chassis. When the main chassis's power is cycled, the I/O reset generated is active for several milliseconds during power-up and for more than 10 microseconds during power-down. If a TILINE Peripheral Bus Interface (TPBI) controller board is installed in the expansion chassis, the long I/O reset can cause its self-test to fail. The TPBI's self-test LED remains lit to indicate that a failure has occurred. This error is normal and expected under these circumstances. Power must be recycled at the expansion chassis to clear the failure LED. To prevent this error, you must power up the main chassis first, then power up the expansion chassis. If the main chassis is powered down for any reason, the expansion chassis must also be powered down.

- Software Remote I/O Reset — A bit is provided in the TPCS which, when written to, issues an I/O reset to the remote chassis only.

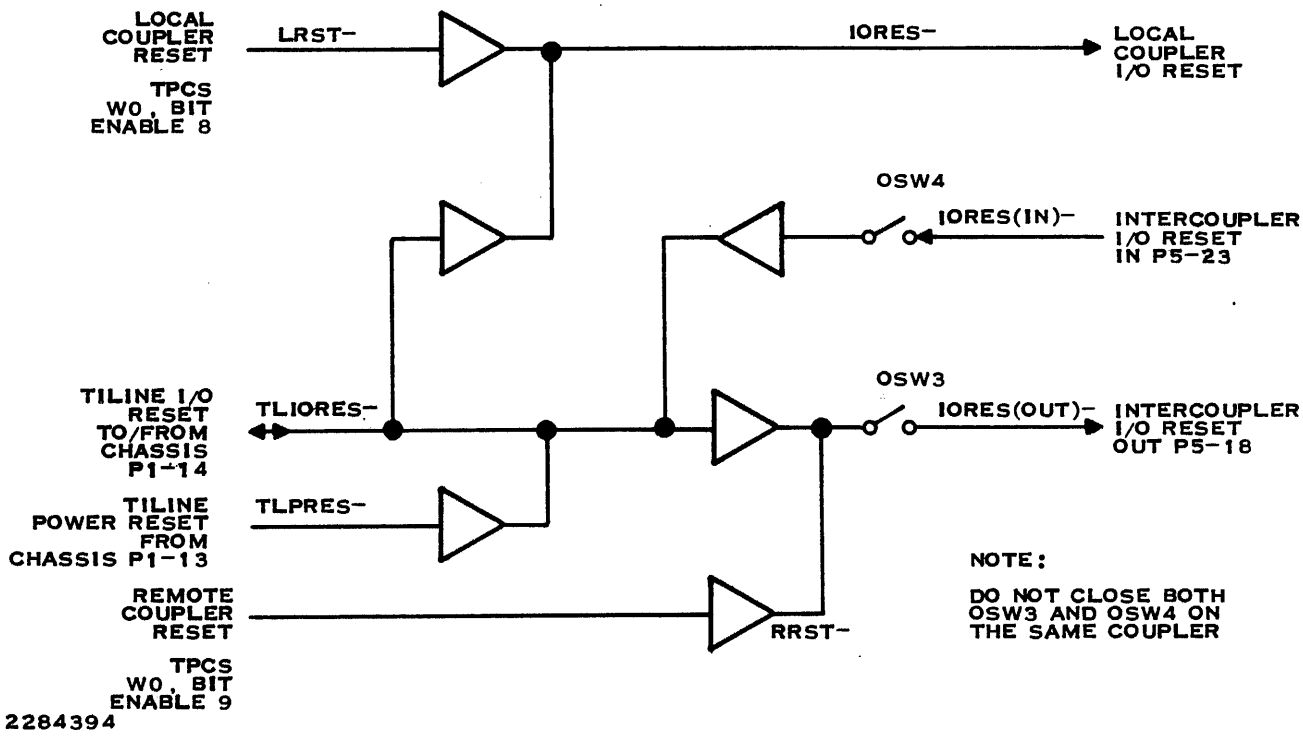


Figure 3-10. Coupler Reset Options

3.7 TILINE COUPLER INTERRUPTS

The TILINE coupler includes options for two direct interchassis interrupts, an internal parity error interrupt, and two multiprocessor interrupts. Each coupler has all of these capabilities because it may serve as either end of an interchassis link. Many of these option switch settings are mutually exclusive. Of course, the operating system software must be aware of the selected options and must be capable of dealing with the resulting interrupts.

Figure 3-11 is a functional block diagram of the interrupt logic. You will find this diagram useful to clarify the interrupt schemes and to identify illogical option switch settings.

3.7.1 Interchassis Interrupts

There are two switch-selectable direct interchassis interrupts on the TILINE coupler. One of these interrupts can be masked through a bit in one of the coupler TPCS command words. This is the maskable interchassis interrupt. The central processor has a choice whether or not to allow this interrupt at any time.

The other direct interchassis interrupt cannot be enabled or disabled by a central processor. If the interrupt input occurs, and the option switch is closed, the interrupt goes to the central processor. This is the nonmaskable interchassis interrupt.

The interchassis interrupts are compatible with each other, for a maximum of two interchassis interrupts. If the expansion chassis must supply more than two interrupts, consider adding a CRU expansion. The CRU buffer/expander combination can prioritize and encode a large number of interrupts. CRU interrupt decoding is necessarily slower than a direct interrupt.

NOTE

Do not attempt to use CI403/CI404 four-channel communications controllers with CRU buffer interrupt encoding. Instead, connect all the CI403/CI404 interrupts on one level and then connect them to either TILINE coupler interrupt input P1-66 or P2-66.

3.7.1.1 Nonmaskable Interchassis Interrupt. Option switch 1 enables the nonmaskable interchassis interrupt from the backplane (P2-66) out to the remote coupler. The EXTRA- interrupt (P4-34) goes to the remote coupler via the data cable.

Option switch 2 enables interrupts into the local coupler from the remote coupler. This interchassis interrupt comes in as EXTRA- on P4-34 and goes to the backplane on connector P2-66. It is not maskable. TILINE power reset prevents the interrupt unless chassis power is up and stable.

NOTE

To prevent having the interrupt latch itself (which could lock up the system), *never* close option switches 1 and 2 on the same coupler.

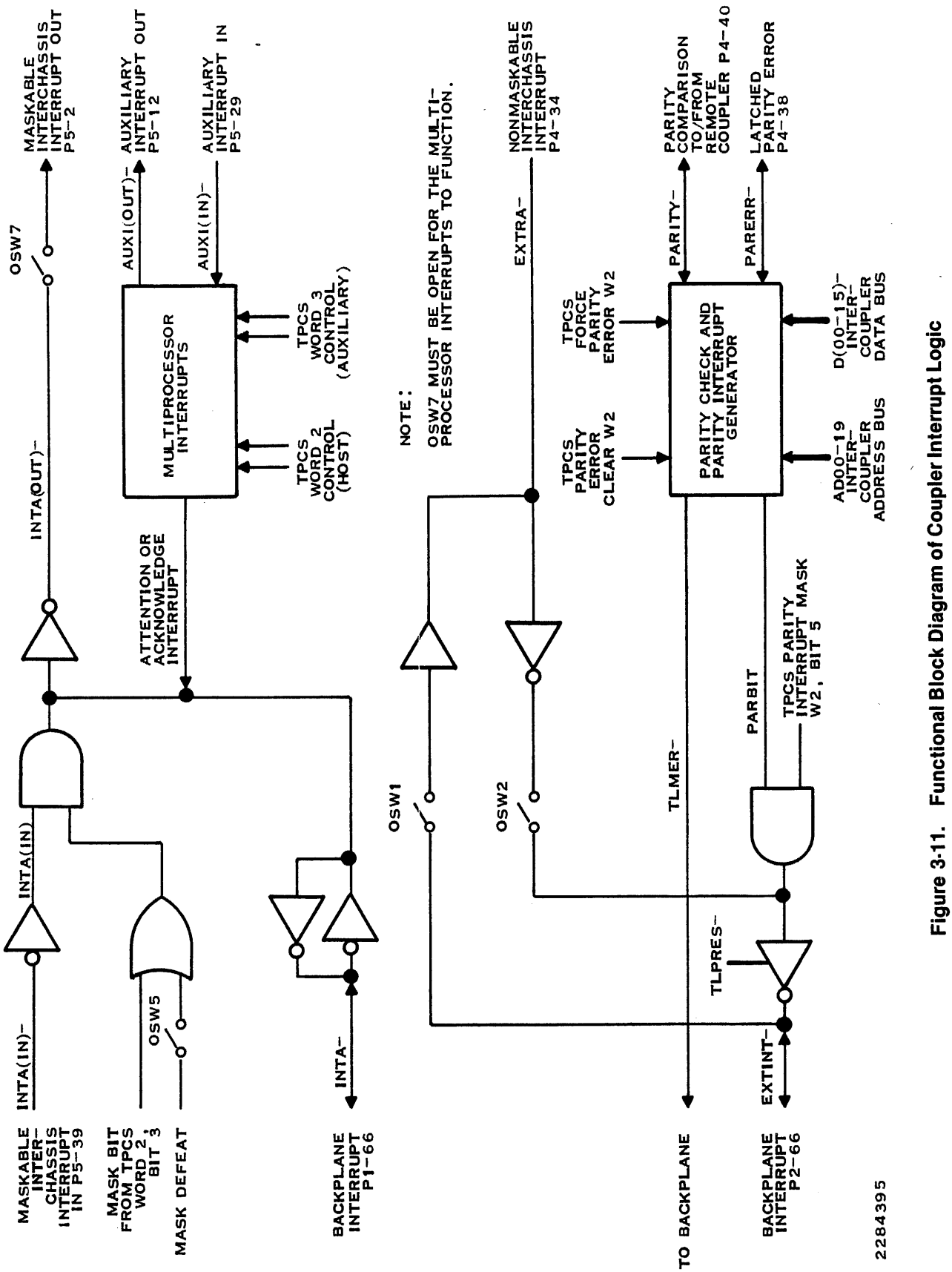


Figure 3-11. Functional Block Diagram of Coupler Interrupt Logic

3.7.1.2 Maskable Interchassis Interrupt. Refer to Figure 3-12, which shows the signal flow for a maskable interchassis interrupt. Option switch 7 connects the maskable interchassis interrupt from the backplane out to the remote coupler. A mask bit in the TPCS enables or disables this interrupt coming in from the remote coupler. The mask bit is bit 3 of word 2, which is the same mask bit used for the attention to host interrupt.

NOTE

Because the interrupt mask bit has differing uses for multiprocessor and maskable direct interrupts, these types of interrupts are incompatible. Do not attempt to use multiprocessor interrupts if option switch 7 (OSW7) is closed.

Option switch 5 is a mask defeat. When option switch 5 is closed, the maskable interrupt is always enabled. Also, bit 3 of TPCS word 2 always appears as a logic 1 on a TPCS read. Option switch 5 must be open to use the multiprocessor interface.

An interrupt pending status bit is available in word 0 of the TPCS. This bit is not affected by the interrupt mask. The maskable interchassis interrupt appears on connector P1-66.

NOTE

To prevent having the interrupt latch itself (which could lock up the system), *never* close option switch 7 on both couplers at the same time.

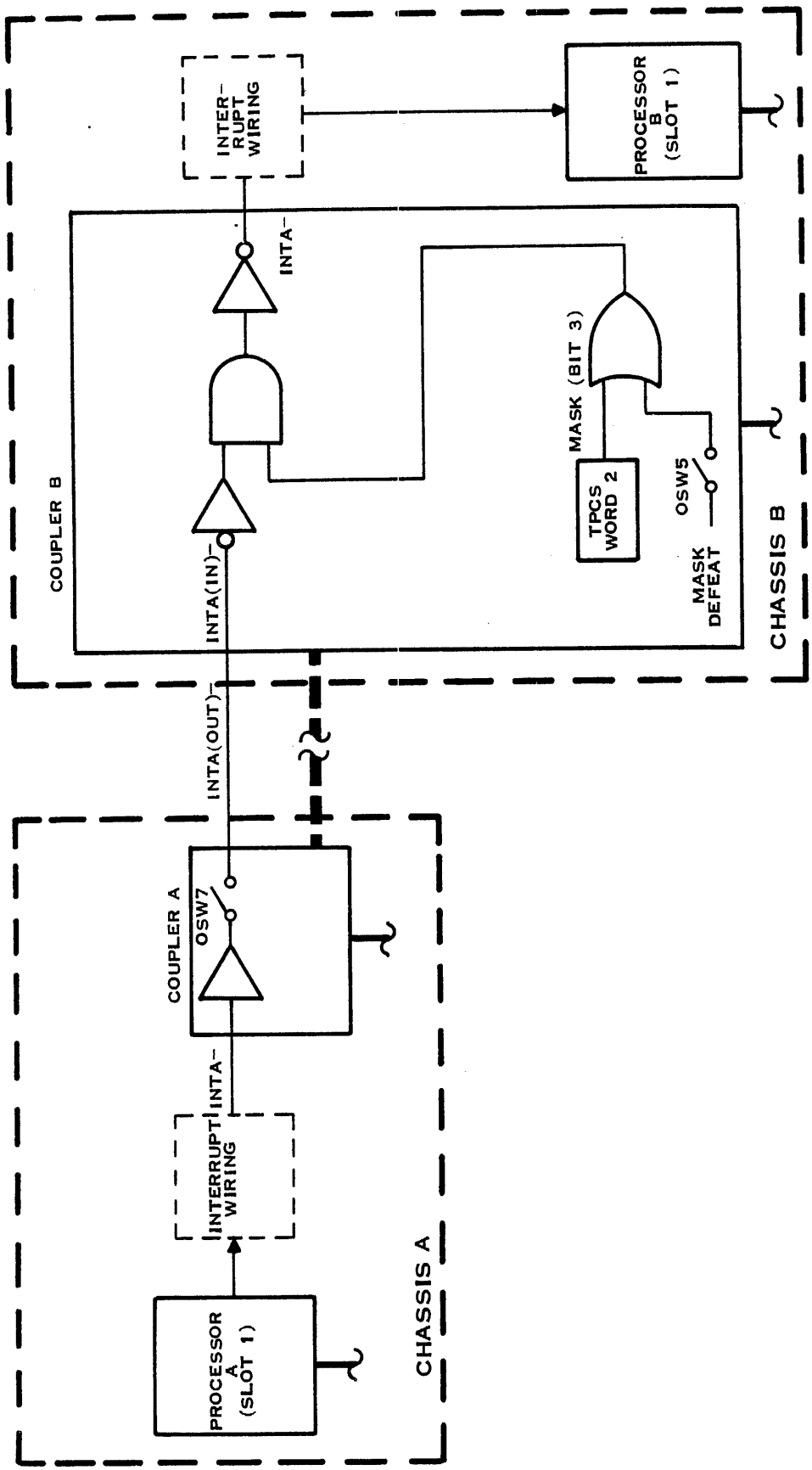


Figure 3-12. Maskable Interchassis Interrupt

3.7.1.3 Internal Interrupt (Parity Error). Both couplers check the combined parity of the intercoupler address and data words. The way that a coupler treats the parity check depends on which coupler initiated the operation.

Recall that a central processor sends an address and a read/write command to the local coupler. The local coupler, acting as a TILINE slave, initiates the intercoupler operation with a start command, START(OUT)-. The remote coupler acts as a master to read or write data, and responds with a complete command, COMP(OUT)-.

The responding coupler provides a parity signal (PARITY-) to the initiating coupler. The initiating coupler compares the remote parity with its own internal parity signal and enables a parity error latch. If the two parity signals disagree, the latch output is the parity error interrupt.

The parity error interrupt and parity error status go to both couplers. However, the interrupt is latched only at the coupler that generated it, and can be reset only at that coupler.

Multiprocessing configurations must make special provisions to assure that all processors recognize and process the interrupt before any processor clears the interrupt. Premature clearing of the interrupt can cause a processor to react unpredictably.

Separate mask bits are available on each coupler for enabling the interrupt onto P2-66. The mask is the parity error interrupt enable bit of TPCS word 0. A parity error will not affect any data transfers but is available to the CPU central processor via a read of the parity error status bit of word 0. The parity error status bit of both couplers is set to one when a parity error occurs, regardless of which coupler generated the interrupt. Resetting the latched interrupt resets the parity error status bit of both couplers simultaneously. The parity error status bit is not affected by the parity error interrupt mask.

An LED indicator lights at each coupler when a parity error occurs. The indicators remain on until the latched parity error is reset at the coupler that generated it. The parity error indicators show the state of the parity error status bit and are always either both off or both on. The parity error interrupt mask does not affect the indicator response.

On an interchassis read (only), a parity error causes TILINE memory error (TLMER-) to go active in the coupler that inititated the cycle. TILINE memory error is not maskable.

TPCS read operations are an exception to the rules for generating parity errors. Parity is not checked when one coupler reads from the TPCS in the remote chassis. Parity is checked on TPCS writes to the remote chassis.

Figure 3-13 summarizes the results of a parity error on both couplers. Part A shows the results of a parity error that occurs on a write to a remote slave or to the remote TPCS. The initiating coupler (coupler A) compares the incoming PARITY- signal to the onboard parity signal to detect the error.

Part B shows the results of a parity error that occurs on a read from a remote slave. Register timing in the coupler does not permit checking parity on a remote TPCS read. Again, the initiating coupler detects and latches the error. Both couplers display error status and generate maskable parity interrupts.

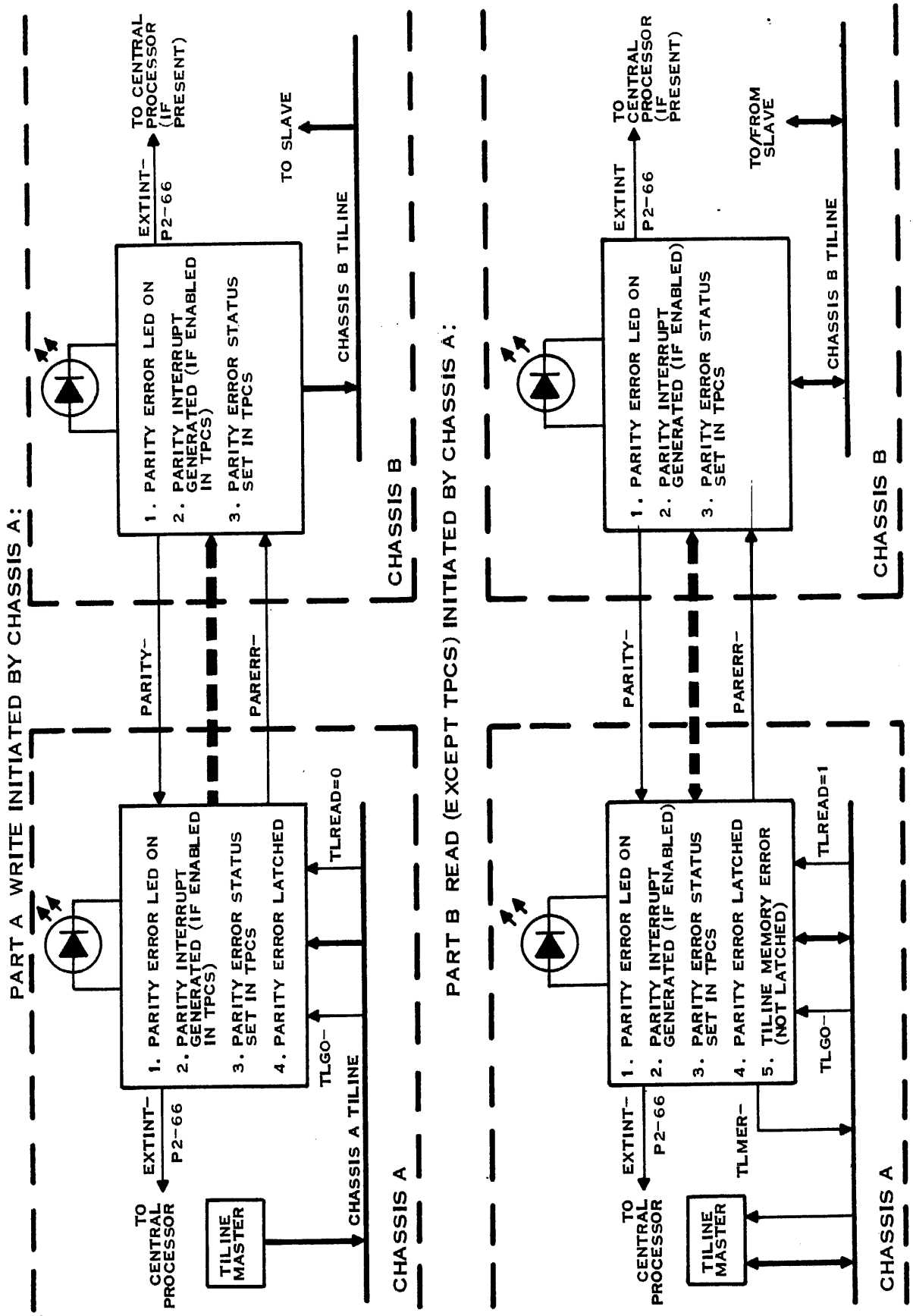


Figure 3-13. Results of a Parity Error

3.7.2 TILINE Multiprocessor Interrupts

Multiprocessor interrupts consist of attention and acknowledge interrupts that are masked and activated by reserved bits in TPCS words 2 and 3. Standard TI operating software does not use the multiprocessor interface.

NOTE

The multiprocessor interrupts emulate a Model 990/10A Computer multiprocessor interrupt structure. This structure is not directly compatible with the CRU-based 990/5 multiprocessor scheme. To use the multiprocessor interrupts, the 990/5 software must emulate the 990/10A multiprocessor scheme.

The interprocessor interface is enabled only when the direct, maskable interchassis interrupt is disabled (option switch 7 is open).

When configured for multiprocessing, one processor is designated the host processor, and the processor of the other chassis is designated the auxiliary processor. The coupler associated with the host processor is the host coupler. Host and auxiliary are software designations, and must be coordinated with interrupt option switches 5 and 7 in the couplers.

Two words in the host coupler TPCS are provided for interrupt control and monitoring. By convention, the host processor always accesses word 2 while the auxiliary processor always accesses word 3 of the same coupler TPCS.

The multiprocessor interrupt functions controlled by the TPCS are:

Interrupt or Mask Function	Mnemonic
Attention to auxiliary	ATTNTOAUX
Acknowledge to host mask	ACKTOHOST mask
Acknowledge to auxiliary	ACKTOAUX
Attention to host mask	ATTNTOHOST mask
Attention to host	ATTNTOHOST
Acknowledge to auxiliary mask	ACKTOAUX mask
Acknowledge to host	ACKTOHOST
Attention to auxiliary mask	ATTNTOAUX mask

This list shows that there are four possible types of interrupts, each of which is maskable. By convention, the auxiliary processor controls the masks for interrupts destined for the auxiliary. Also, the auxiliary processor controls the attention and acknowledge interrupts to the host.

By the same convention, the host processor controls the masks for the interrupts directed to the host, and controls interrupt outputs to the auxiliary processor. There is nothing in the coupler hardware that enforces these conventions.

Refer to Figure 3-14 and assume that the ATTNTOAUX mask is already set. In part A, the host processor sets the ATTNTOAUX interrupt in TPCS word 2. It also sets the ACKTOHOST mask bit to accept the reply. The multiprocessor interface logic sends auxiliary interrupt AUX(OUT)- to the auxiliary coupler. The auxiliary coupler recieves the interrupt as AUX(IN)-, and generates backplane interrupt INTA- on P2-66.

The auxiliary processor knows that the interrupt came from the coupler. Also, if the auxiliary processor controls the mask, it knows that this must be an attention rather than an acknowledge interrupt. The auxiliary responds with an ACKTOHOST interrupt that it writes into TPCS word 3 of the host coupler. Part B of Figure 3-14 shows this cycle. Notice that there is no actual interrupt line enabled between couplers.

The host coupler combines the ACKTOHOST with the previously-set ACKTOHOST mask to generate backplane interrupt INTA-. The arrival of the ACKTOHOST bit also clears the ATTNTOAUX interrupt.

Either processor can determine the status of any interrupt by reading the status bits in host coupler TPCS words 2 and 3. More detailed information about TPCS word 2 and word 3 bit assignments appears in the following TPCS word descriptions. Refer to Appendix A for additional information about the multiprocessor attention/acknowledge protocol.

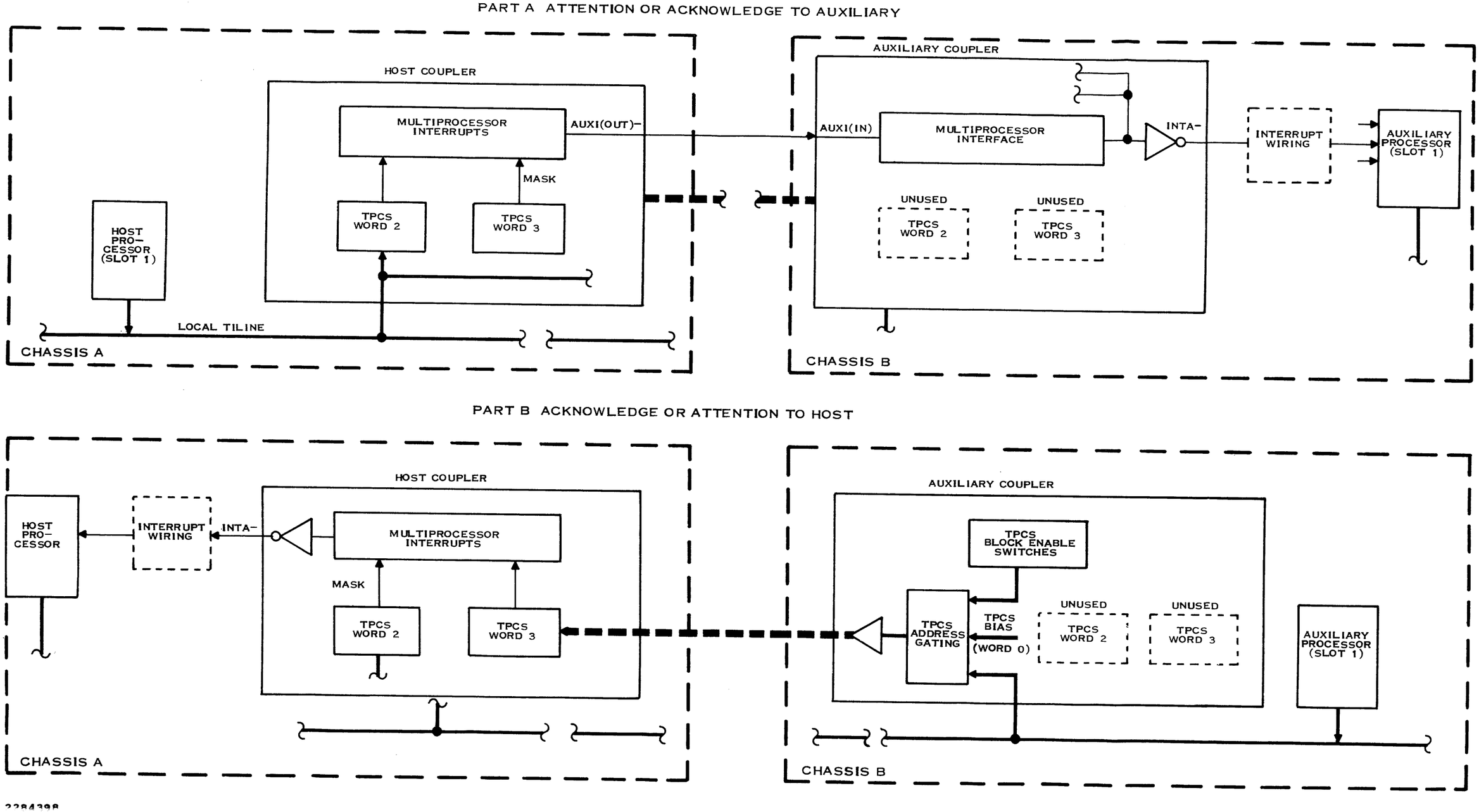
3.8 TPCS WORDS — BIT ASSIGNMENTS AND INTERPRETATION

TPCS words contain status information on the interrupts, the address and bias switches, the option switches, and the user-controlled enable bits, and feedback information on the latched address and data bits. Refer to Figure 3-15, which summarizes TPCS bit assignments for the TILINE coupler.

Word 0 (W0) contains the user-controlled enable bits and some read-only status bits. It has two modes of operation. A read from word 0 operates in a word mode that allows all 16 bits to be read simultaneously. A write to word 0 operates in a bit mode where only one of the user-controlled bits can be set or reset at a time.

Word 1 (W1) is a read-only word containing the 16 latched data bits. Words 2 and 3 contain the attention and acknowledge interrupts and mask bits of the multiprocessor interrupt interface. Word 2 (W2) is the host interface word. During normal multiprocessing operations it is accessed by a single designated host processor. Word 3 (W3) is the auxiliary interface word. During normal multiprocessing operations it is accessed by a single designated auxiliary processor.

Word 4 (W4) is a read-only word associated with the parallel-in, serial-out, switch-feedback shift register. The shift register provides a feedback path for the address limit switches, the bias switches, the option switches, and the 20 latched address lines.



2284398

Figure 3-14. Multiprocessor Interrupts

2272075-9701

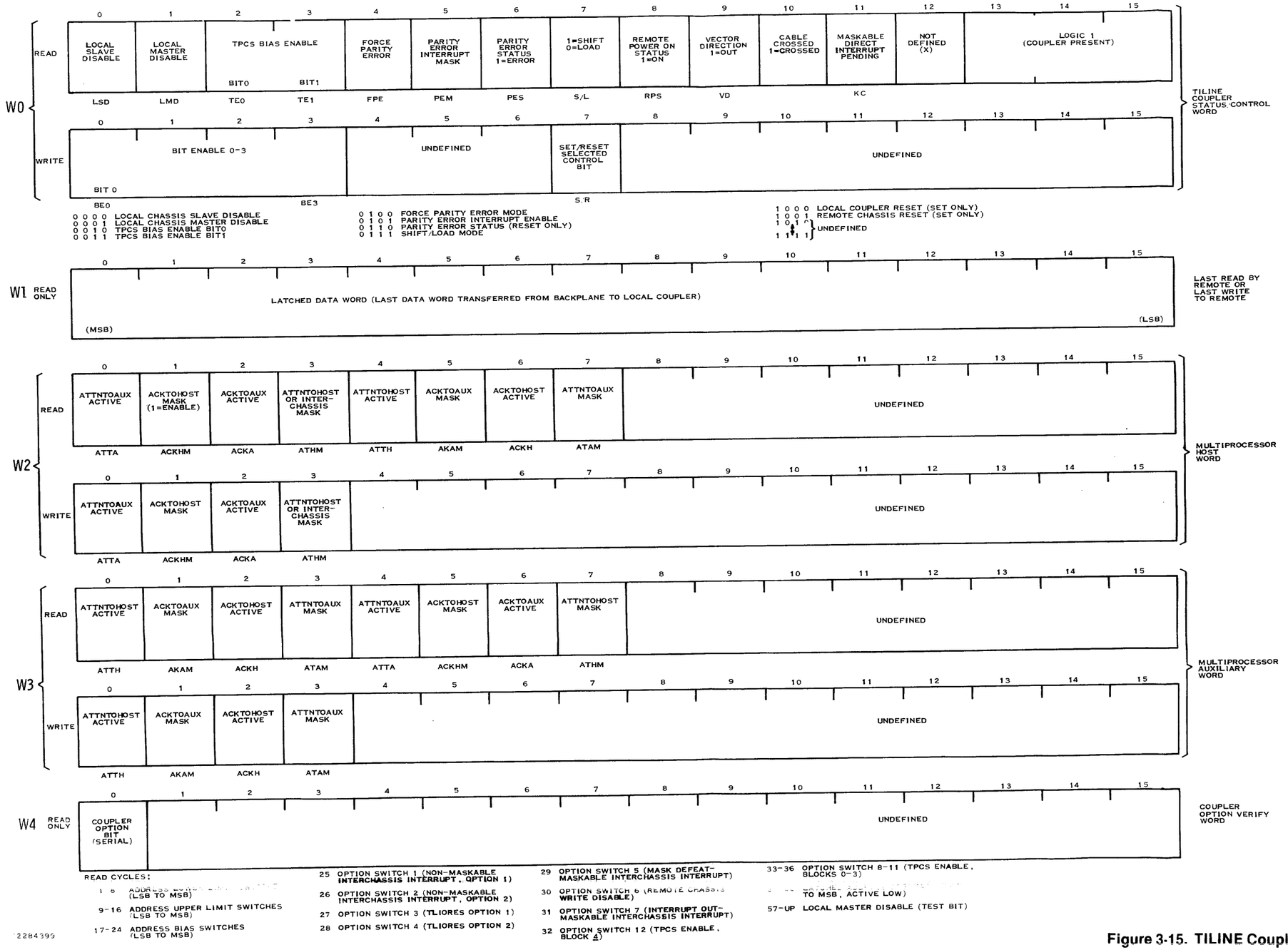


Figure 3-15. TILINE Coupler TPCS Control/Status Word Formats

3.8.1 TPCS Word 0 (Read Mode) — Coupler Status

The following status bits are included in W0:

Bit 0 — Local chassis slave disable

Bit 1 — Local chassis master disable

Bit 2 — TPCS bias enable bit 0 (MSB)

Bit 3 — TPCS bias enable bit 1 (LSB)

Bit 4 — Force parity error mode

Bit 5 — Parity error interrupt enable

Bit 6 — Parity error status

Bit 7 — Shift/load mode bit

Bit 8* — Remote power-on status

Bit 9* — Vector direction

Bit 10* — Cables crossed indicator

Bit 11* — Maskable direct interrupt pending

Bit 12* — Not defined

Bit 13* — Logic 1 (not used)

Bit 14* — Logic 1 (not used)

Bit 15* — Logic 1 (not used)

Note:

* Bits 8 through 15 are read-only bits.

3.8.1.1 Local Chassis Slave Disable — W0, Bit 0. This bit disables the TILINE slave function of the local coupler. When set to a logic 1, this bit prevents the local coupler from answering on the local TILINE and initiating a TILINE master cycle in the remote chassis. This bit does not prohibit access to the coupler local TPCS.

Any of the following conditions reset the slave disable bit:

- Explicit control by a write to TPCS word 0 (bit enable = 0000, set/reset bit = 0)
- A coupler internal I/O reset due to:
  - — Local coupler reset command in TPCS word 0 (bit enable = 1000, set/reset bit = 1)
  - — TILINE I/O reset from chassis
  - — TILINE power reset from chassis
  - — Reset command from remote coupler, with local OSW4 closed

3.8.1.2 Local Chassis Master Disable — W0, Bit 1. This bit disables the TILINE master function of the local coupler. When set to a logic 1, this bit prevents the local coupler from accessing the local TILINE. This also prevents the remote coupler from accessing the local coupler TPCS.

W0, bit 1 is also an input to the diagnostic shift register where it is used as a test input (W4).

Any of the following conditions reset the local master disable bit:

- Explicit control by a write to TPCS word 0 (bit enable = 0001, set/reset bit = 0)
- A coupler internal I/O reset (refer back to W0, bit 0 for a desciption of the coupler internal reset conditions)

3.8.1.3 TPCS Bias Enable Bits 0 and 1 — W0, Bits 2 and 3. These bits specify the block of TPCS addresses which will be mapped to block 0 of the remote chassis. Refer to Table 3-5.

Table 3-5. TPCS Mapping

Bits 2	Bits 3	Function
0	0	Disable mapping
0	1	Map TPCS addresses of block 1 to block 0
1	0	Map TPCS addresses of block 2 to block 0
1	1	Map TPCS addresses of block 3 to block 0

Any of the following operations reset the TPCS mapping bits:

- Explicit control by two writes to TPCS word 0 (bit enable = 0010, set/reset bit = 0 and bit enable = 0011, set/reset bit = 0)
- A coupler internal I/O reset (refer back to W0, bit 0 for a list of the coupler internal reset conditions)

3.8.1.4 Force Parity Error Mode — W0, Bit 4. When set to a logic 1, this bit toggles the least significant address line as it enters the local parity checking circuitry, forcing a parity error.

NOTE

If the force parity error bit is set on both couplers, the incorrect parities will cancel each other and no parity error will be generated.

Either of the following resets the force parity error mode bit:

- A write to TPCS word 0 (bit enable = 0100)
- A coupler internal I/O reset (refer back to W0, bit 0 for a description of the reset conditions)

3.8.1.5 Parity Error Interrupt Enable — W0, Bit 5. This bit allows an interrupt on P2-66 when the parity error interrupt is set and the proper switch options are selected. Setting this bit to a logic 0 prevents the interrupt from appearing.

A write to TPCS word 0 (bit enable = 0101) or a local coupler reset clears the parity error interrupt enable bit. Refer to W0, bit 0 for a list of local coupler reset conditions.

3.8.1.6 Parity Error Status — W0, Bit 6. This bit reflects the logic level of the coupler parity error status line (active high). The coupler executing a slave cycle generates a parity error. The parity error status bit is set or reset at both couplers. This bit remains set until the error is cleared at the coupler that generated it.

The parity error LED remains on at both couplers as long as this bit is active high. This bit is not affected by the parity error interrupt enable.

NOTE

This bit cannot be set by writing to word 0. It can be set by forcing a parity error as described with W0, bit 4.

A write to TPCS word 0 (bit enable = 0110) or a local coupler reset can reset the parity error status bit. Refer to W0, bit 0 for a list of the local coupler reset conditions.

3.8.1.7 Shift/Load Mode — W0, Bit 7. This bit determines the mode of the parallel-in, serial-out switch feedback shift register. When reset to a logic 0, the shift register is placed in the load mode. While in this mode, a read of word 4 latches the switch configurations into the parallel inputs of the shift register.

When set to a logic 1, the shift register is placed in the shift mode. In shift mode, the switch configuration can be read, one switch at a time, through the serial output of the shift register to the most significant data bit of word 4. Each successive read of word 4 shifts the register by one bit. This bit is reset to a logic 0 by a local coupler I/O reset, as described with W0, bit 0.

3.8.1.8 Remote Power On Status — W0, Bit 8. This bit reflects the status of the remote chassis. A logic 1 indicates that the remote chassis is powered up. This bit is read only.

3.8.1.9 Vector Direction — W0, Bit 9. This bit is true (logic 1) if the preferred direction of TILINE transfers is from the local to the remote coupler. This direction is established by the vector control cable. This bit is read only.

3.8.1.10 Cables Crossed Indicator — W0, Bit 10. When this bit is a logic 1, it indicates that the inter-coupler address and data cables are cross-connected. A logic 0 indicates that the data cable is connected correctly or that no cable is connected to P4.

3.8.1.11 Maskable Direct Interrupt Pending — W0, Bit 11. When this bit is a logic 1, it indicates that the maskable direct interchassis interrupt is pending. This bit is not affected by the interrupt mask.

3.8.1.12 W0, Bit 12. This bit is not defined.

3.8.1.13 W0, Bits 13 Through 15. These bits are read as logic 1 and can be used to identify the presence of a coupler.

3.8.2 TPCS Word 0 (Write Mode) — Coupler Control

A write to WORD 0 can only set or reset one bit at a time. The data word sent to word 0 is defined as follows:

Bit 0 — Bit enable 0 (MSB)

Bit 1 — Bit enable 1

Bit 2 — Bit enable 2

Bit 3 — Bit enable 3 (LSB)

Bit 4 — X

Bit 5 — X

Bit 6 — X

Bit 7 — Set/reset

Bit 8 — X

Bit 9 — X

Bit 10 — X

Bit 11 — X

Bit 12 — X

Bit 13 — X

Bit 14 — X

Bit 15 — X

Note:

X = Don't care.

3.8.2.1 Bit Enable — W0, Bits 0 Through 3. These bits indicate which status bit in word 0 is to be set or reset, as shown in Table 3-6.

Table 3-6. TPCS Word 0 Write Enable Bits

Bit Enable 0 1 2 3	Status Bit Affected
0 0 0 0	Local chassis slave disable
0 0 0 1	Local chassis master disable
0 0 1 0	TPCS bias enable bit 0 (MSB)
0 0 1 1	TPCS bias enable bit 1 (LSB)
0 1 0 0	Force parity error mode
0 1 0 1	Parity error interrupt enable
0 1 1 0	Parity error status (reset only)
0 1 1 1	Shift/load mode
1 0 0 0	Local coupler reset
1 0 0 1	Local chassis reset
1 0 1 0	Not defined
1 0 1 1	Not defined
1 1 0 0	Not defined
1 1 0 1	Not defined
1 1 1 0	Not defined
1 1 1 1	Not defined

Writing a logic 1 or logic 0 to the local coupler reset bit issues an IORES- signal to the local coupler only. Writing a logic 1 or logic 0 to the remote chassis reset bit issues an IORES- signal to the remote chassis only. The IORES- pulse has a duration of 250 nanoseconds.

3.8.2.2 W0, Bits 4 Through 6. These are don't care bits.

3.8.2.3 Set/Reset — W0, Bit 7. A logic 1 sets the bit specified by bit enable 0 through 3 (unless the specified bit is a reset-only bit). A logic 0 resets the appropriate bit.

3.8.2.4 W0, Bits 8 Through 15. These are don't care bits.

3.8.3 TPCS Word 1 (Read Only) — Latched Data

These bits reflect the data last transferred from the local TILINE to the local coupler. Data is transferred in this direction during a slave write cycle or a master read cycle.

3.8.4 TPCS Word 2 — Multiprocessor Host Word

Word 2 contains the following bits:

Bit 0 — Attention to auxiliary (ATTNTOAUX) interrupt active

Bit 1 — Acknowledge to host (ACKTOHOST) interrupt mask

Bit 2 — Acknowledge to auxiliary (ACKTOAUX) interrupt active

Bit 3 — Attention to host (ATTNTOHOST) interrupt mask or interchassis interrupt mask

Bit 4* — Attention to host (ATTNTTOHOST) interrupt active

Bit 5* — Acknowledge to auxiliary (ACKTOAUX) interrupt mask

Bit 6* — Acknowledge to host (ACKTOHOST) interrupt active

Bit 7* — Attention to auxiliary (ATTNTOAUX) interrupt mask

Note:

* Bits 4 through 7 are read-only bits.

A coupler internal I/O reset command resets all bits of W2. Reset conditions include:

- Local coupler reset command in TPCS word 0 (bit enable = 1000, set/reset bit = 1)
- TILINE I/O reset from chassis
- TILINE power reset from chassis
- Reset command from remote coupler, with local OSW4 closed

3.8.4.1 ATTNTOAUX Active — W2, Bit 0. This bit reflects the status of the attention to auxiliary interrupt line (active high). Writing a logic 1 to this bit sets the interrupt. Writing a logic 0 clears the interrupt. Option switch 7, if closed, overrides the TPCS inputs to hold the bit reset.

During normal multiprocessing operations, this interrupt is cleared when the acknowledge to host interrupt is activated. The attention to auxiliary mask has no effect on W2, bit 0.

A local coupler internal reset clears W2, bit 0.

3.8.4.2 ACKTOHOST Mask — W2, Bit 1. When this bit is set to a logic 0, the acknowledge to host interrupt is disabled. When set to a logic 1, the interrupt is enabled. A local coupler reset clears the bit.

3.8.4.3 ACKTOAUX Active — W2, Bit 2. This bit reflects the status of the acknowledge to auxiliary interrupt line (active high). Writing a logic 1 to this bit automatically clears the attention to host interrupt and activates the acknowledge to auxiliary interrupt. The acknowledge interrupt is activated by the high to low transition of the attention to host interrupt line. Writing a logic 0 to this bit clears the acknowledge interrupt.

During normal multiprocessing operations, the interrupt is cleared when the acknowledge to auxiliary interrupt mask is set to a logic 0. This bit is cleared by IORES- or if option switch 7 is closed.

3.8.4.4 ATTNTOHOST or Interchassis Interrupt Mask — W2, Bit 3. When using the multiprocessing interface, this bit is the ATTENTION INTERRUPT FROM AUXILIARY MASK. When using the maskable interchassis interrupt as coming into the local coupler from the remote coupler, this bit enables/disables the interrupt onto the backplane.

In either mode, when this bit is set to a logic 0, the interrupt is disabled. When set to a logic 1, the interrupt is enabled. This bit is cleared by IORES-. When option switch 5 is closed, the interchassis interrupt is always enabled and this bit always reads as a logic 1.

3.8.4.5 ATTNTOHOST Active — W2, Bit 4. This bit shows the status of the attention to host interrupt line (active high).This bit is read only and is not affected by the attention to host bit. Option switch 7, if closed, holds this bit reset. A coupler internal reset will clear this bit.

3.8.4.6 ACKTOAUX Mask — W2, Bit 5. A logic 0 in this bit indicates that the acknowledge to auxiliary interrupt is disabled. A logic 1 indicates that the interrupt is enabled. This bit is read only. A coupler internal reset clears this bit.

3.8.4.7 ACKTOHOST Active — W2, Bit 6. This bit reflects the status of the acknowledge to host interrupt line (active high). This bit is read only. Option switch 7, if closed, holds this bit reset. A coupler internal reset clears this bit.

3.8.4.8 ATTNTOAUX Mask — W2, Bit 7. A logic 0 in this bit indicates that the attention to auxiliary interrupt is disabled. A logic 1 indicates that the interrupt is enabled. This bit is read only. A coupler internal reset clears the bit.

3.8.5 TPCS Word 3 — Multiprocessor Auxiliary Word

Word 3 consists of the following bits:

Bit 0 — Attention to host (ATTNTOHOST) interrupt active

Bit 1 — Acknowledge to auxiliary (ACKTOAUX) interrupt mask

Bit 2 — Acknowledge to host (ACKTOHOST) interrupt active

Bit 3 — Attention to auxiliary (ATTNTOAUX) interrupt mask

Bit 4* — Attention to auxiliary (ATTNTOAUX) interrupt active

Bit 5* — Acknowledge to host (ACKTOHOST) interrupt mask

Bit 6* — Acknowledge to auxiliary (ACKTOAUX) interrupt active

Bit 7* — Attention to host (ATTNTOHOST) interrupt mask

Note:

* Bits 4 through 7 are read-only bits.

A coupler internal I/O reset command resets all bits of W3. Reset conditions include:

- Local coupler reset command in TPCS word 0 (bit enable = 1000, set/reset bit = 1)
- TILINE I/O reset from chassis
- TILINE power reset from chassis
- Reset command from remote coupler, with local OSW4 closed

3.8.5.1 ATTNTOHOST Active — W3, Bit 0. This bit shows the status of the attention to host interrupt line (active high). Writing a logic 1 to this bit sets the interrupt. Writing a logic 0 clears the interrupt. During normal multiprocessing operations, this interrupt is cleared when the acknowledge to auxiliary interrupt is activated. The attention to host mask has no effect on this bit. A coupler internal reset clears this bit. Option switch 7, if closed, keeps the bit reset.

3.8.5.2 ACKTOAUX Mask — W3, Bit 1. When this bit is set to a logic 0, the acknowledge to auxiliary interrupt is disabled. When set to a logic 1, the interrupt is enabled. A coupler internal reset clears this bit.

3.8.5.3 ACKTOHOST Active — W3, Bit 2. This bit reflects the status of the acknowledge to host interrupt line (active high). Writing a logic 1 to this bit automatically clears the attention to auxiliary interrupt and activates the acknowledge to host interrupt. The acknowledge interrupt is activated by the high to low transition of the attention to auxiliary interrupt line. Writing a logic 0 to this bit clears the acknowledge interrupt.

During normal multiprocessing operations, the interrupt is cleared when the acknowledge to host interrupt mask is set to a logic 0. A coupler internal reset clears this bit. Option switch 7, if closed, holds the bit reset.

3.8.5.4 ATTNTOAUX Mask — W3, Bit 3. When this bit is set to a logic 0, the attention to auxiliary interrupt is disabled. When set to a logic 1, the interrupt is enabled. A coupler internal reset clears the bit.

3.8.5.5 ATTNTOAUX Active — W3, Bit 4. This bit shows the status of the attention to auxiliary interrupt line (active high). This bit is read only and is not affected by the attention to auxiliary mask bit. A coupler internal reset clears this bit. Option switch 7, if closed, holds the bit reset.

3.8.5.6 ACKTOHOST Mask — W3, Bit 5. A logic 0 in this bit indicates that the acknowledge to host interrupt is disabled. A logic 1 indicates the interrupt is enabled. This bit is read only, and is cleared by a coupler internal reset.

3.8.5.7 ACKTOAUX Active — W3, Bit 6. This bit reflects the status of the acknowledge to auxiliary interrupt line (active high). This bit is read only, and is cleared by a coupler internal reset. Option switch 7, if closed, holds the bit reset.

3.8.5.8 ATTNTOHOST Mask — W3, Bit 7. A logic 0 in this bit indicates that the attention to host interrupt has been disabled. A logic 1 indicates that the interrupt is enabled. This bit is read only, and is cleared by a coupler internal reset. Option switch 5, if closed, forces the bit to appear as a logic 1.

3.8.6 TPCS Word 4 (Read Only) — Coupler Option Verify

TPCS word 4 is controlled by the shift/load bit of word 0 (W0, bit 7). In load mode, the coupler loads the switch setting data in parallel into a parallel-in/serial-out shift register. You cannot read W4 in load mode. In shift mode, word 4 data is available, one bit at a time.

Bit 0 is the only data bit defined for word 4. Bit 0 is the serial output of the parallel-in/serial-out shift register. Successive read operations shift the switch data out on W4, bit 0. For all option switches, a logic 1 indicates switch closure. Word 4 is a diagnostic word that is not required for single-CPU or multiprocessor data transfers.

Table 3-7 lists the coupler option verify bit assignments.

Table 3-7. TPCS Coupler LR Option Verify Bit Assignments

Bit 0 (Shift Mode)	Switch Setting or Latched Address
Read 1	Address lower limit switch (LSB)
Read 2	Address lower limit
Read 3	Address lower limit
Read 4	Address lower limit
Read 5	Address lower limit
Read 6	Address lower limit
Read 7	Address lower limit
Read 8	Address lower limit switch (MSB)
Read 9	Address upper limit switch (LSB)
Read 10	Address upper limit
Read 11	Address upper limit
Read 12	Address upper limit
Read 13	Address upper limit
Read 14	Address upper limit
Read 15	Address upper limit
Read 16	Address upper limit switch (MSB)
Read 17	Address bias switch (LSB)
Read 18	Address bias
Read 19	Address bias
Read 20	Address bias
Read 21	Address bias
Read 22	Address bias
Read 23	Address bias
Read 24	Address bias switch (MSB)
Read 25	OSW1 — Nonmaskable interchassis interrupt option 1
Read 26	OSW2 — Nonmaskable interchassis interrupt option 2
Read 27	OSW3 — TLIORES- option 1
Read 28	OSW4 — TLIORES- option 2
Read 29	OSW5 — Mask defeat, maskable interchassis interrupt
Read 30	OSW6 — Remote chassis write disable
Read 31	OSW7 — Interrupt out, maskable interchassis interrupt
Read 32	OSW12 — TPCS enable block 4
Read 33	OSW8 — TPCS enable block 0
Read 34	OSW9 — TPCS enable block 1
Read 35	OSW10 — TPCS enable block 2
Read 36	OSW11 — TPCS enable block 3
Read 37	Latched address (TLAD19 + bias)- (LSB)
Read 38	Latched address (TLAD18 + bias)-
Read 39	Latched address (TLAD17 + bias)-
Read 40	Latched address (TLAD16 + bias)-
Read 41	Latched address (TLAD15 + bias)-
Read 42	Latched address (TLAD14 + bias)-
Read 43	Latched address (TLAD13 + bias)-
Read 44	Latched address (TLAD12 + bias)-
Read 45	Latched address (TLAD11 + bias)-
Read 46	Latched address (TLAD10 + bias)-

Table 3-7. TPCS Coupler LR Option Verify Bit Assignments (Continued)

Bit 0 (Shift Mode)	Switch Setting or Latched Address
Read 47	Latched address (TLAD09 + bias)-
Read 48	Latched address (TLAD08 + bias)-
Read 49	Latched address (TLAD07 + bias)-
Read 50	Latched address (TLAD06 + bias)-
Read 51	Latched address (TLAD05 + bias)-
Read 52	Latched address (TLAD04 + bias)-
Read 53	Latched address (TLAD03 + bias)-
Read 54	Latched address (TLAD02 + bias)-
Read 55	Latched address (TLAD01 + bias)-
Read 56	Latched address (TLAD00 + bias)- (MSB)
Read 57 upward — Local master disable (used as a test bit)	

3.8.6.1 Address Lower and Upper Limit Switches — Read 1 Through 16. These bits reflect the setting of the lower and upper address limit switches. A logic 1 indicates a switch closure.

3.8.6.2 Address Bias Switches (Read 17 Through 24). These bits reflect the address bias switch settings. A logic 1 indicates switch closure.

3.8.6.3 Nonmaskable Interchassis Interrupt Option 1 — Read 25. When closed, OSW1 enables interchassis interrupts from the back panel (at P2-66) out to the remote coupler. A logic 1 indicates switch closure.

3.8.6.4 Nonmaskable Interchassis Interrupt Option 2 — Read 26. When closed, OSW2 enables interchassis interrupts from the remote coupler into the back panel (at P2-66). A logic 1 indicates switch closure.

3.8.6.5 TLIORES- Option 1 — Read 27. When closed, OSW3 enables the TLIORES- signal from the local chassis onto the remote coupler. A logic 1 indicates switch closure.

3.8.6.6 TLIORES- Option 2 — Read 28. When closed, OSW4 enables the TLIORES- signal of the remote chassis onto the local coupler. A logic 1 indicates switch closure.

3.8.6.7 Mask Defeat for Maskable Interchassis Interrupt — Read 29. When closed, OSW5 always enables the direct maskable interchassis interrupt from the remote chassis to the coupler. A logic 1 indicates switch closure.

3.8.6.8 Remote Chassis Write Disable — Read 30. When closed, OSW6 disables write operations to the remote coupler. When open, both read and write operations are enabled. A logic 1 indicates switch closure.

3.8.6.9 Interrupt Out, Maskable Interchassis Interrupt — Read 31. When closed, OSW7 allows an interrupt originating on P1-66 of the backplane to be sent to the remote coupler. Also, when closed, this switch disables the attention and acknowledge multiprocessor interrupts.

3.8.6.10 TPCS Enable Bits, Block 4 and Blocks 0 Through 3 — Read 32 Through 36. These bits reflect the TPCS block enable switches. A logic 1 indicates switch closure.

3.8.6.11 Latched Address Bits (Including Bias) — Read 37 Through 56. These bits are low true and reflect the 20-bit TILINE address which was presented to the coupler during the last TILINE coupler slave cycle. The latched address is equal to the TILINE address added to the bias set in the offset switches.

NOTE

The latched address bits are the only bits that the remote coupler cannot load into the local feedback shift register. This is due to the fact that the address that the remote coupler maps to the local coupler uses the same address bus that is needed to feedback the latched address bits. However, if the values have already been loaded into the feedback register, then the remote coupler is able to place the register in the shift mode and read all of the switch and address feedback bits.

3.8.6.12 Local Master Disable (Test Bit) — Read 57 and Beyond. This bit is the same bit as described in W0, bit 1. It is provided here as a test bit. By setting or resetting this bit (through word 0), a known state can be shifted through the entire serial path of the shift register.

Appendix A

Using the Multiprocessor Interface

A.1 GENERAL

This appendix describes how to use the multiprocessor interface of the TILINE coupler. The appendix describes a few possible configurations, a typical sequence of operations, and similarities with the 990/10A multiprocessor interface. The last part of the appendix describes the maskable and nonmaskable interchassis interrupts and some considerations for multiprocessing configurations.

A.2 MULTIPROCESSOR OPERATION

The multiprocessor interface of the TILINE coupler facilitates supervisory communication between the host and auxiliary processors by means of interrupts. Two words in the coupler's TILINE peripheral control space (TPCS) set and clear these interrupts. The on-board TPCS base address switches determine the addresses of these words.

Both the host and auxiliary processors receive interrupts off the backplane. A host is presumed to be in slot 1 of the main chassis. A slot 1 processor receives the interrupts off the backplane on P1-66. One or more auxiliary processors can be in the expansion chassis. An auxiliary processor in slot 1 receives the interrupts off the backplane in the same way as the host. When the auxiliary is in any slot other than slot 1, the backplane jumper plug must be modified so that the interrupts coming in from the coupler on P1-66 are directed out to the auxiliary on P1-66 or P2-66.

The attention interrupt begins interaction between the host and auxiliary processors, and the acknowledge interrupt is the response. Each coupler contains a complete multiprocessor interface; however, only one interface is ever used during an attention-acknowledge interrupt sequence. The other coupler's interface is never accessed.

NOTE

While it is conceivable to use one coupler's interface for one interrupt sequence and the other coupler's interface for the next sequence, extra care must be taken to insure that all the interrupts are disabled on the interface that is not in use. Switching from one interface to the other is not recommended.

To decide which coupler's multiprocessor interface to use, take the following steps:

1. Designate one processor as the host and the processor in the other chassis as the auxiliary. This first step is necessary to be consistent with the interface bit definitions in Tables A-1 and A-2.

2. Designate the coupler in the host's chassis as the host coupler. It is the host coupler's multiprocessor interface that is used. The host processor accesses TPCS word 2, while the auxiliary processor accesses TPCS word 3 of the host coupler.

A.2.1 Host Processor Interface

The host can interrupt the auxiliary processor by setting the attention to auxiliary interrupt bit to one in the host coupler's multiprocessor interface. The auxiliary can enable this interrupt by setting the attention interrupt mask to one, but the power-up sequence initially leaves the mask reset to zero. The host must initialize the auxiliary with an appropriate interrupt vector and then enable the attention mask before generating the attention interrupt. The auxiliary acknowledges the interrupt by clearing the attention interrupt from the host. This automatically causes an acknowledge interrupt to the host. The host enables acknowledge interrupts by setting the acknowledge mask to one. Table A-1 lists the bit definitions of the host interrupts.

Table A-1. Interface Bits as Seen by the Host Processor

Address	Bit	Value	Meaning When Read	Value	Meaning When Written
Word 2	0	0	ATTNTOAUX inactive	0	Clear ATTNTOAUX (test mode)
		1	ATTNTOAUX active	1	Send ATTNTOAUX interrupt
Word 2	1	0	ACKTOHOST masked	0	Mask out, clear ACKTOHOST interrupt from auxiliary
		1	ACKTOHOST unmasked	1	Enable ACKTOHOST interrupt from auxiliary
Word 2	2	0	ACKTOAUX inactive	0	Clear ACKTOAUX (test mode)
		1	ACKTOAUX active	1	Clear ATTNTOHOST from auxiliary and send ACKTOAUX interrupt
Word 2	3	0	ATTNTOHOST masked	0	Mask out attention interrupt from auxiliary
		1	ATTNTOHOST unmasked	1	Enable attention interrupt
Word 2	4	0	ATTNTOHOST inactive		N/A
		1	ATTNTOHOST active		
Word 2	5	0	ACKTOAUX from host masked		N/A
		1	ACKTOAUX unmasked		
Word 2	6	0	ACKTOHOST inactive		N/A
		1	ACKTOHOST active		
Word 2	7	0	ATTNTOAUX from host masked		N/A
		1	ATTNTOAUX unmasked		

A.2.2 Auxiliary Processor Interface

An auxiliary processor can interrupt the host by setting the attention to host interrupt bit. If the host sets the attention interrupt mask to one, the host is interrupted from the backplane. The host acknowledges this interrupt by clearing the attention interrupt. Clearing the attention interrupt generates an acknowledge interrupt back to the auxiliary. Table A-2 lists the bit definitions of the auxiliary interrupts.

Table A-2. Interface Bits as Seen by the Auxiliary Processor

Address	Bit	Value	Meaning When Read	Value	Meaning When Written
Word 3	0	0	ATTNTOHOST inactive	0	Clear attention interrupt (test mode)
		1	ATTNTOHOST active	1	Send ATTNTOHOST interrupt
Word 3	1	0	ACKTOAUX from host masked	0	Mask out and clear interrupt from auxiliary
		1	ACKTOAUX unmasked	1	Enable acknowledge interrupt
Word 3	2	0	ACKTOHOST not active	0	Clear acknowledge interrupt (test mode)
		1	ACKTOHOST active	1	Clear attention interrupt from host and send ACKTOHOST
Word 3	3	0	ATTNTOAUX masked	0	Mask out attention interrupt from host
		1	ATTNTOAUX unmasked	1	Enable attention interrupt from host
Word 3	4	0	ATTNTOAUX inactive		N/A
		1	ATTNTOAUX active		
Word 3	5	0	ACKTOHOST masked		N/A
		1	ACKTOHOST unmasked		
Word 3	6	0	ACKTOAUX not active		N/A
		1	ACKTOAUX active		
Word 3	7	0	ATTNTOHOST masked		N/A
		1	ATTNTOHOST unmasked		

A.3 MULTIPROCESSOR CONFIGURATIONS

Figure A-1 shows one possible configuration of a multiprocessor system.

In Figure A-1, the relationship between the host and the auxiliary processors can assume many forms. Only one is discussed here. In this case, the operating system of each auxiliary processor requires a host processor to be present in the system. The host uses auxiliary processors 1 and slot 1 to execute tasks. The auxiliary processors are loaded with a kernel that supports the coupler's multiprocessor interface and, to a limited extent, the communications port (if applicable). The kernel used by each auxiliary processor is assumed not to support a user without interaction with the host.

The slot 1 auxiliary processor controls auxiliary processor S1 with the same relationship as the host processor controls its auxiliary processors. The slot 1 auxiliary processor is loaded with a kernel that allows it to schedule tasks with its auxiliary processor and to perform I/O. Thus, the slot 1 auxiliary appears as an auxiliary to the host and as a host to auxiliary S1. One primary advantage of this configuration is the expansion of the host's I/O capabilities by utilizing the slot 1 auxiliary.

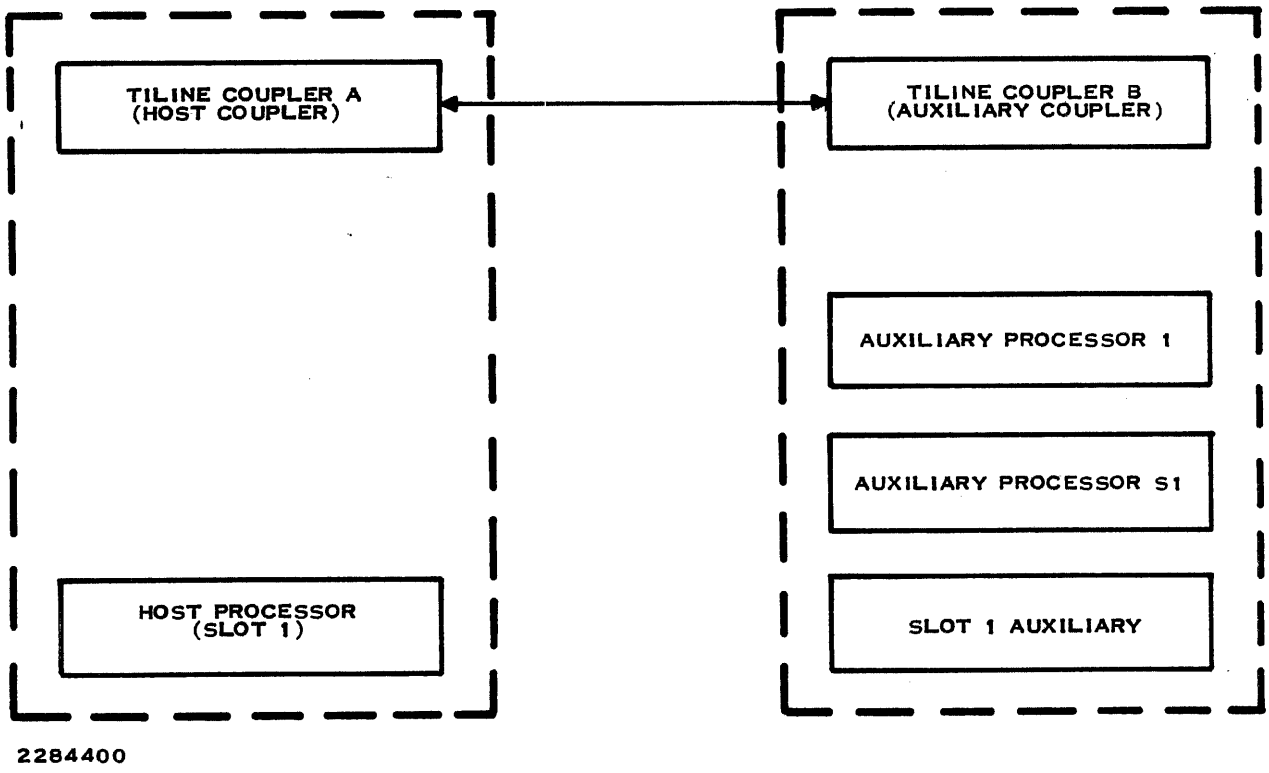


Figure A-1. Multiprocessor Configuration With Host and Auxiliaries

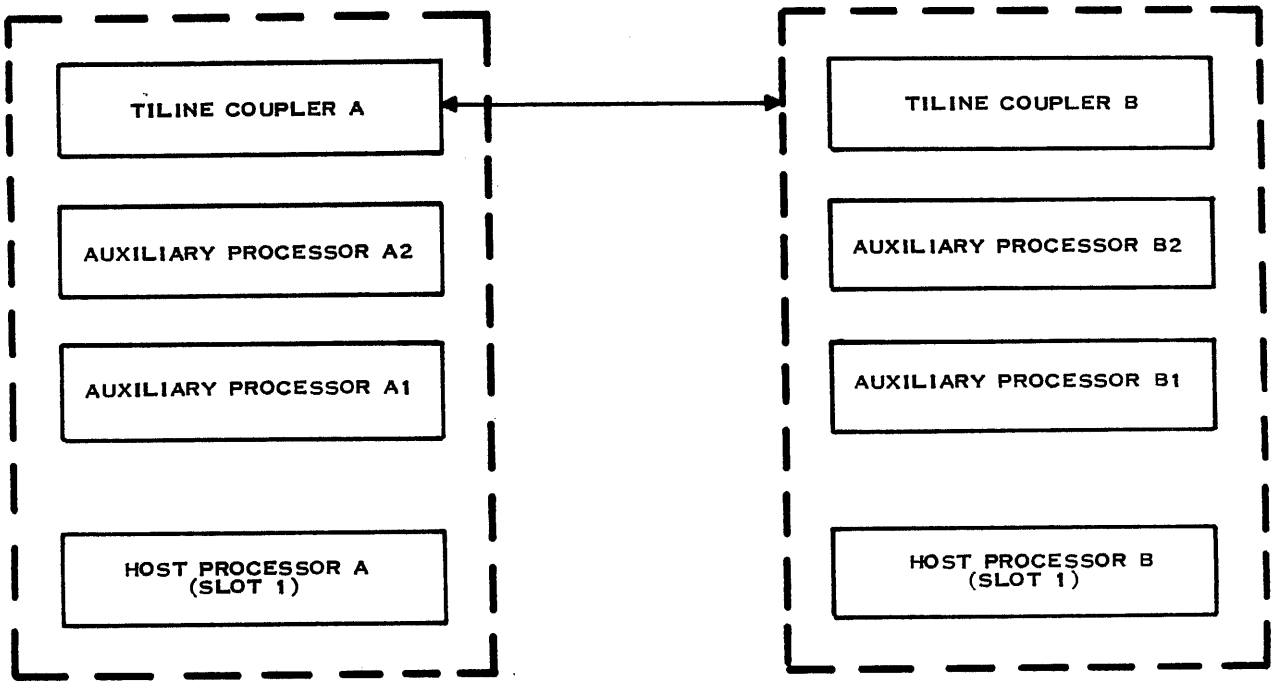
In Figure A-1, coupler A can be designated the host coupler. Both the slot 1 processor and auxiliary processor 1 access the multiprocessor interface of the host coupler to control interrupts to the host processor. Since the slot 1 auxiliary and auxiliary processor 1 share the same interrupt level from the host's point of view, the host processor must have some method for determining which processor interrupted it. One method is to set up a flag in memory, which all processors have access to, that indicates which auxiliary generated the interrupt.

Interrupts sent by the host will be received by both the slot 1 processor and the auxiliary 1 processor at the same time (assuming the backplane jumper plug is configured correctly). Therefore, the auxiliary processors need a method to determine which of them was meant to receive the interrupt. Again, one such method is to set up a flag in memory, which all processors have access to, that indicates which auxiliary the interrupt was meant for.

Figure A-2 shows a second configuration that includes multiple hosts.

The configuration in Figure A-2 requires that each host operate under its own operating system. This configuration does not allow host A to legally control an auxiliary of host B or vice-versa. Multiprocessing interrupts are passed between a host and its auxiliaries or between the two hosts.

Only the host-to-host communication can use the coupler's multiprocessing interface. In this configuration, it makes no difference whether coupler A or coupler B provides the multiprocessor interface.



2284401

Figure A-2. Multiprocessors With Multiple Hosts

A.4 TYPICAL SEQUENCE OF OPERATIONS

In the following discussion, all references to setting and resetting interrupts and interrupt masks involve the multiprocessing interface of the host coupler only. The term local refers to the host chassis; remote refers to the expansion chassis.

A typical sequence of operations involving communications between a host and an auxiliary in a multiprocessor system might be:

1. The auxiliary processor powers up and executes self-test (except the load portion), sets the attention to auxiliary interrupt mask to enable attention interrupts by the host processor, and goes to IDLE.

2. The host processor checks to see if the auxiliary processor has powered up. This can be done by checking the TPCS of the local coupler for RPWRON to be true. To provide initial instructions to the auxiliary processor, the host may write a task into the auxiliary processor's memory for it to execute, or the auxiliary may be loaded with a kernel causing it to read an interprocessor message area in the system table area. The host processor then sets the attention interrupt to the auxiliary processor and simultaneously enables the acknowledge interrupt back from the auxiliary processor.

3. The auxiliary processor acknowledges that it received the attention interrupt (and possibly that it understood its instructions) by setting the acknowledge interrupt. The setting of the acknowledge interrupt clears the attention interrupt.

4. The host responds to the acknowledge by testing the acknowledge bit for a 1. If the host processor has multiple auxiliary processors on the same interrupt level and more than one is capable of generating the acknowledge interrupt at any one time, the host needs a method to determine which auxiliary processor(s) has generated the interrupt. There is no information residing on the coupler that indicates which processor accessed its TPCS. The host then sets the enable acknowledge bit to zero to clear the acknowledge bit. The acknowledge status latch is cleared on the high to low transition of the enable acknowledge bit.

5. The auxiliary processor, upon completion of its task or when in need of service, begins the preceding sequence. In this case, the words auxiliary and host are interchanged in the preceding description.

If the processor that initiates the sequence (sets attention to one) is to poll for acknowledgement, rather than to accept interrupts, the following sequence applies:

1. The auxiliary processor powers up and executes self-test (except the load portion), sets the attention to auxiliary interrupt mask to enable attention interrupts by the host processor, and goes to IDLE.

2. The host processor checks to see if the auxiliary processor has powered up by checking the TPCS of the local coupler for remote power on. To provide initial instructions to the auxiliary processor, the host may write a task into the auxiliary processor's memory for it to execute, or the auxiliary may be loaded with a kernel causing it to read an interprocessor message area in the system table area. The host processor then sets the attention interrupt to the auxiliary processor and simultaneously enables the acknowledge interrupt back from the auxiliary processor.

3. The auxiliary processor acknowledges that it received the attention interrupt (and possibly that it understood its instructions) by setting the acknowledge interrupt. Setting the acknowledge interrupt clears the attention interrupt. In this case, the acknowledge interrupt is not generated because it is masked.

4. The host polls for the acknowledge by testing the acknowledge bit for a 1. The host then sets the enable acknowledge bit to a 1. Finally, the host sets the enable acknowledge bit to zero. This sequence is necessary because the acknowledge bit is cleared only on the falling edge of the enable acknowledge interrupt bit. Logic in the interface prevents the acknowledge interrupt from being generated when the enable acknowledge bit is set to 1 in this mode.

5. The auxiliary processor, upon completion of its task or when in need of service, will begin the preceding sequence. Interchange the words auxiliary and host in the preceding description.

NOTE

In *no* case should a processor write to its control word (after setting its attention bit to a 1) until either it has seen an acknowledge from the other processor or the other processor is verified to have halted (FAULT is true). If a processor were to attempt to modify a bit in its control word with the attention bit set to a 1, it could not correctly predict the value to be written into the attention bit. The reason is that the state of the attention bit may be changed at any time by the processor responding to the attention. A read followed by a write will not guarantee successful prediction because the responding processor may have its memory access interleaved with the desired read-write sequence.

The processor that sets the acknowledge bit should not clear the bit. Acknowledge will be cleared by the processor responding to the acknowledge. If the processor that sets the acknowledge also resets the acknowledge, the processor responding to the acknowledge may not be able to successfully test the acknowledge bit before it is reset and therefore may not be able to determine the source of the interrupt.

The preceding examples will suffice when messages are not queued across the interface. If queued requests are desired, then the following sequence is required:

1. The host writes a message in the interprocessor communication area and sets attention to the auxiliary. The auxiliary processor reads (copies) the message and responds with acknowledge. When the auxiliary is finished, it will respond with an attention interrupt to the host.

2. The host wishes to put the next message in the auxiliary queue. It is possible that both the host and the auxiliary will get their attention set on consecutive TILINE cycles and lock up the interface (see the preceding note). To prevent this, the host disables interrupts (LIMI 2), sets the attention mask, and then re-enables interrupts. This is necessary because the auxiliary could set the interrupt during the TILINE cycle before the mask was set. When the auxiliary responds with an acknowledge for the message just sent, the host can then disable interrupts, enable attention, and re-enable interrupts. When the auxiliary sees the attention interrupt from the host, it verifies that the host has masked off the attention interrupt from the auxiliary so that the host will not acknowledge the attention. The state of the attention will not change, allowing the auxiliary to write into its control word even though attention is set.

The preceding queuing routine requires that interrupts be disabled for a short period of time. Another approach can be used to overcome the requirement of disabling the interrupts. A lockword convention can be established, using a dedicated memory location. This location signals availability by showing a negative value. A processor desiring access tests the lockword, using the ABS instruction. The ABS instruction sets a positive value if the location contains a negative value. The status register indicates whether the value was changed or was already positive, indicating another processor had access. When access is complete, the processor sets the lockword negative. (A negate instruction is sufficient for this operation, since a conflict is not posssible.) A processor finding the lockword positive postpones its access until a negative value is found.

A.5 SIMILARITY TO THE 990/10A MULTIPROCESSING INTERFACE

The TILINE coupler's multiprocessor interface has been designed to simulate the 990/10A interface. The coupler's multiprocessor interface appears to all other processors, in terms of both hardware and software, as the multiprocessor interface of a 990/10A acting in the auxiliary mode with the following exceptions:

- Bit 8 of WORD 2, auxiliary processor busy/idle, has been omitted from the coupler's interface.
- Bit 9 of WORD 2, auxiliary passed/failed self-test, has been omitted from the coupler's interface.

In order to check if the auxiliary processor has power, the host can check if the remote chassis power on status bit (RPWRON) is set to one. As long as these exceptions are allowed for, software developed for communication with the 990/10A interface can be used for communicating with the coupler's interface.

A.6 CONSIDERATIONS IN MULTIPROCESSING CONFIGURATIONS

The TILINE coupler's multiprocessor interface has been designed to allow any host-to-auxiliary(s) combination of 990 CPU microcomputers. Its versatility lies in the fact that any microcomputer that can access the TPCS and that can receive interrupts off the backplane interrupt jumper plug, can control interrupts through the coupler's multiprocessor interface without regard to what type of processor it is communicating with. Several configurations that were not previously possible are now allowed. For example, the 990/12 can now act as a slot 1 auxiliary (as shown in Figure A-1) with either a 990/10, a 990/10A, or a 990/12 as the host processor. Similarly, the 990/10 can now act as a slot 1 auxiliary (as shown in Figure A-1) with either a 990/10, a 990/10A, or a 990/12 as the host processor.

The multiprocessor interrupts emulate a Model 990/10A computer multiprocessor interrupt structure, and are not directly compatible with the CRU-based 990/5 multiprocessor scheme. To use the multiprocessor interrupts, the 990/5 software must emulate the 990/10A multiprocessor scheme.

It should be noted that a 990/5 CPU can conceivably act as the host to any of the higher level 990 microcomputers, but because of its limitations in addressing this would be a highly improbable configuration.

Appendix B

Configuring the Multiprocessor Interface

B.1 ENABLING THE DIRECT NONMASKABLE INTERRUPT

In addition to the multiprocessing interface, two direct interchassis interrupts are available for multiprocessing. One of the interrupts is direct and nonmaskable. This interrupt appears on P2-66 and can be configured as either an incoming or outgoing interrupt. When configured as an incoming interrupt, option switch 1 is closed on the local coupler and open on the remote coupler, while option switch 2 is open on the local coupler and closed on the remote coupler. When configured as an outgoing interrupt, the option switches are reversed on both couplers.

NOTE

Never close option switches 1 and 2 on the same coupler.

The nonmaskable interchassis interrupt is independent of the multiprocessor interface and can be used during multiprocessing operations.

B.2 ENABLING THE DIRECT MASKABLE INTERRUPT

The second interrupt is also direct, and it can be configured as either maskable or nonmaskable and as either incoming or outgoing. This interrupt appears on P1-66. When configured as an outgoing, maskable interrupt, option switch 7 is closed on the local coupler and open on the remote coupler. Option switch 5 (mask defeat) is also open on the remote coupler.

NOTE

Because of the danger of the interrupt latching itself across the two chassis, *never* close option switch 7 (OSW7) on both couplers at the same time.

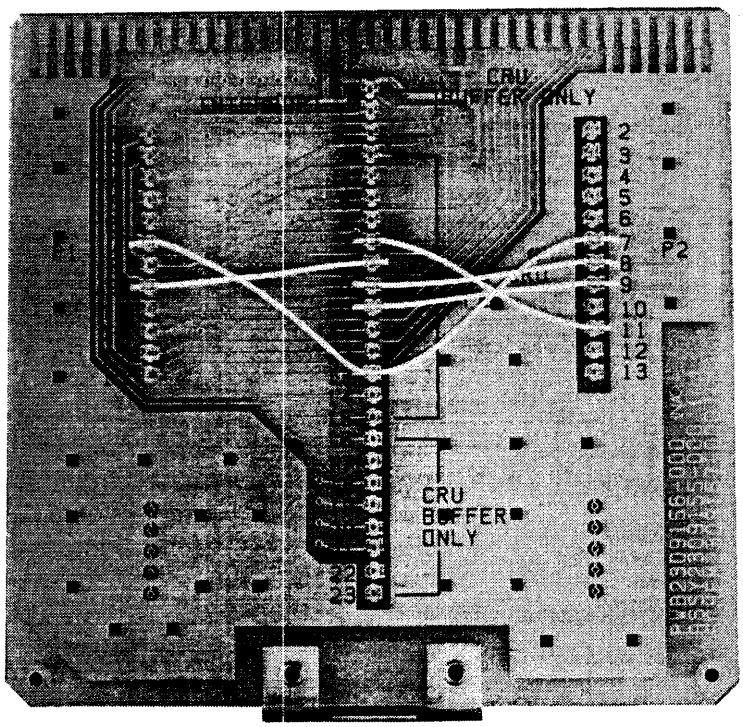
The mask for the outgoing interrupt is on the remote coupler (Word 2, Bit 3—interchassis interrupt mask). To make the outgoing interrupt unmaskable, close option switch 5 (OSW5) on the remote coupler.

When configured as an incoming, maskable interrupt, option switch 7 is open on the local coupler and closed on the remote coupler, and option switch 5 is open on the remote coupler. The mask for the incoming interrupt is on the local coupler (word 2, bit 3). To make the incoming interrupt unmaskable, close option switch 5 on the local coupler.

The maskable interchassis interrupt is *not* independent of the multiprocessing interface, and you must choose one or the other for handling interrupts on P1-66.

B.3 WIRING THE INTERCHASSIS INTERRUPTS

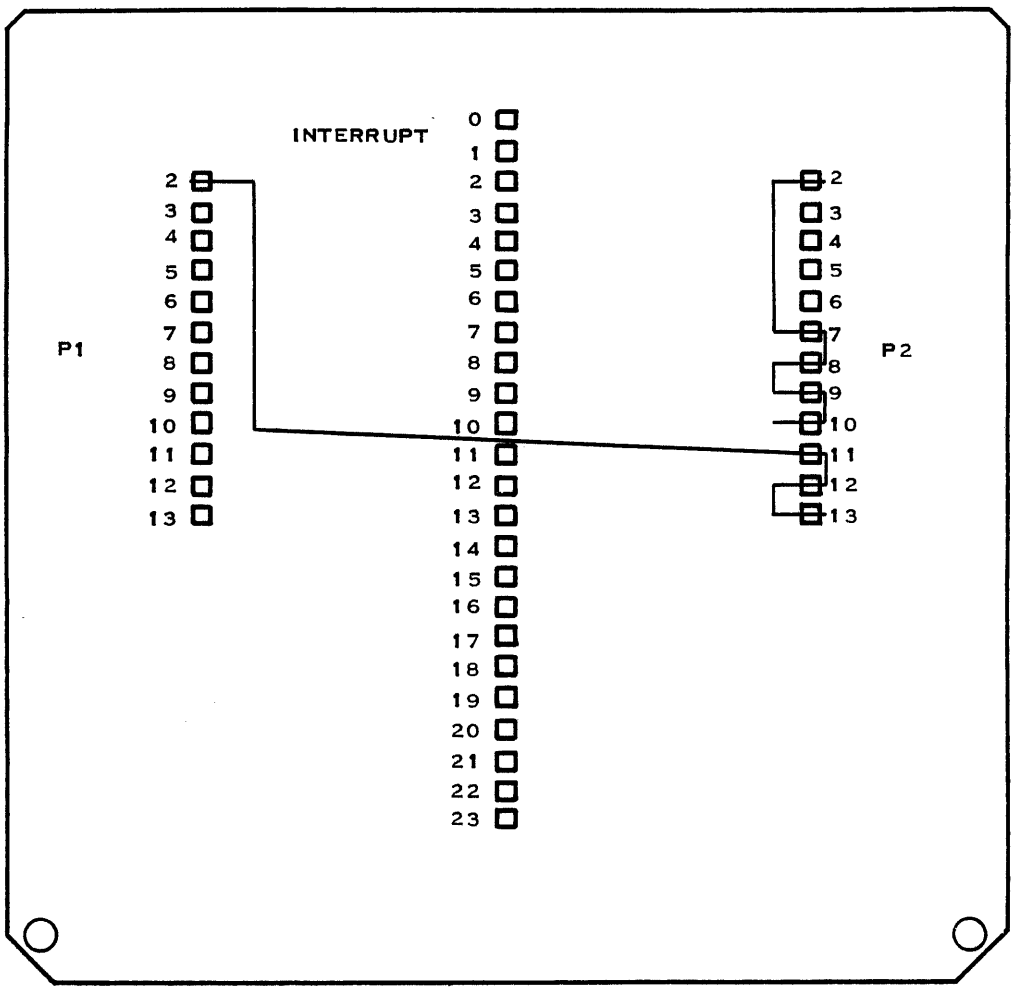
In almost every expansion system that uses the direct interrupts, you need to install the programmable interrupt board in the expansion chassis. Figure B-1 shows the pattern of terminals on the programmable interrupt board. Interrupt outputs from all the P1 connectors in slots 2 through 13 are on the left side, and the P2 interrupts are on the right side. The middle row of terminals are the interrupt level inputs to slot 1. Interrupt levels 3 through 15 are the only levels available in a computer chassis. Do not make any connections to levels 0 through 2 or 16 through 23 except in an expansion chassis. The Programmable Interrupt Kit, TI part number 2309285-1, contains general instructions for wiring the interrupt board. Some additional remarks are needed for use with the TILINE couplers.



2283014

Figure B-1. Programmable Interrupt Board

If the expansion coupler is located in any slot other than slot 1, the wiring is straightforward. When using the direct nonmaskable interrupt, connect all the interrupt outputs in a daisy chain from the P1 and/or P2 side to the P2 side of the coupler at its respective slot. Similarly, when using the direct maskable interrupt, connect all the interrupt outputs in a daisy chain from the P1 and/or P2 side to the P1 side of the coupler at its respective slot. An example appears in Figure B-2. In this example, three P2 interrupt outputs (from boards located in slots 7 through 10) are connected in a daisy chain to the P2 side of the coupler (located in slot 2). Three other P2 interrupt outputs (from boards located in slots 11 through 13) are also connected in a daisy chain to the P1 side of the coupler.



2281257

Figure B-2. Sample Interrupt Wiring Schematic With TILINE Coupler in Slot 2

If the expansion coupler is located in slot 1, the wiring is less obvious. In this case, the coupler's P2 side corresponds to the terminal marked INTERRUPT 14, while the P1 side corresponds to INTERRUPT 2. The remainder of the wiring procedure is the same as that described for an expansion coupler located in a slot other than slot 1. An example appears in Figure B-3. This example is the same as that in Figure B-2 except that the coupler is in slot 1.

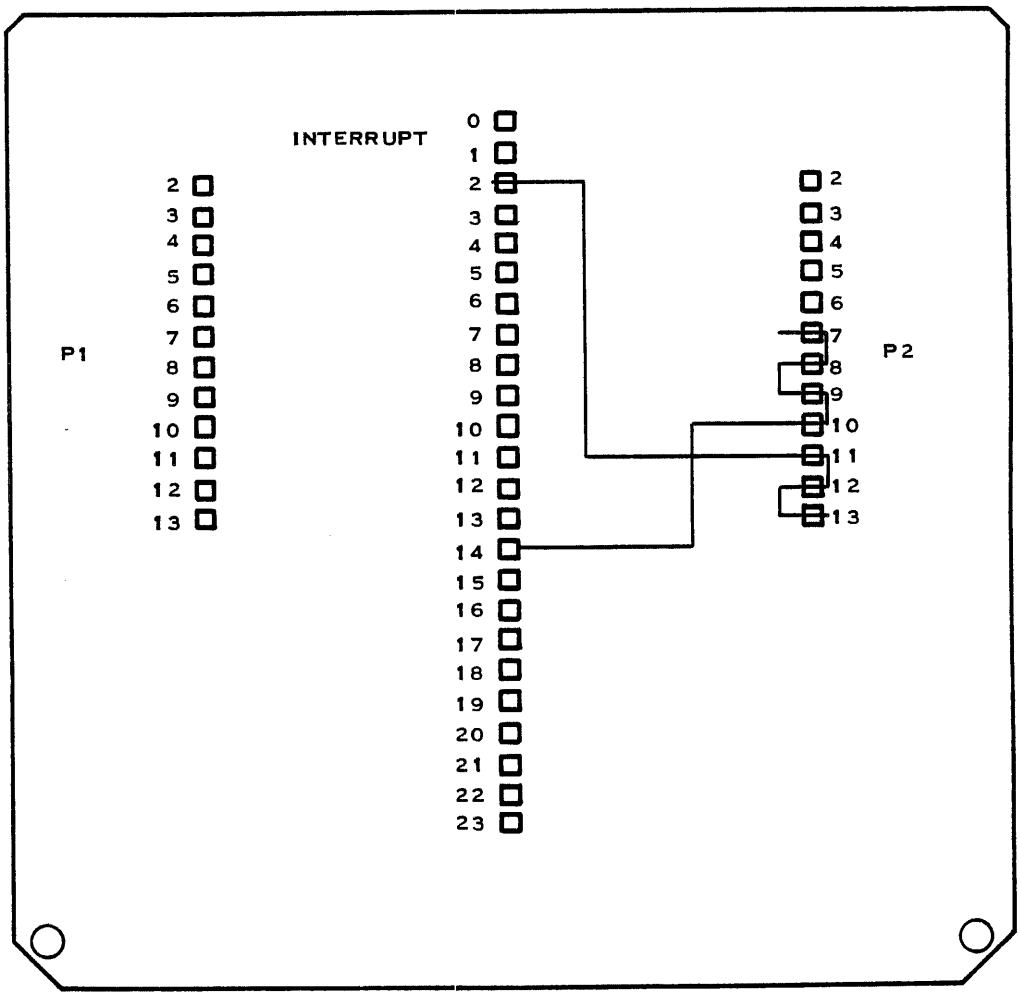


Figure B-3. Sample Interrupt Wiring Schematic With TILINE Coupler in Slot 1

NOTE

Regardless of how the interrupts are wired through the coupler in the expansion chassis, the main chassis coupler determines the actual interrupt level (as recognized by the main chassis CPU). All interrupt outputs connected to the P2 side of the expansion coupler have the same interrupt level as the P2 side of the main coupler. All interrupt outputs connected to the P1 side of the expansion coupler have the same interrupt level as the P1 side of the main coupler. Thus, if the coupler's P1 side has a different interrupt level than the P2 side, the main coupler should not be in a slot that has the P1 and P2 sides connected together; that is, the coupler should not be in slot 6 when using the standard 990 interrupt levels in the main chassis.

Appendix C

Examples of TILINE Expansion Configurations

C.1 INTRODUCTION

This appendix provides examples of different TILINE expansion configurations with particular attention paid to TPCS base address switch settings. Every possible configuration is not covered, but the concepts discussed should make it possible to configure a wide variety of special application systems.

NOTE

Before you review the following examples, read paragraphs 1.5 through 1.6.4 for a general introduction to the TILINE coupler and its various switches.

All TPCS addresses referred to in the following examples are central processor byte addresses in the range from >F800 to >FBF0. All memory addresses referred to in the following examples are TILINE byte addresses in the range from >000000 to >1FF7FF.

C.2 TILINE EXPANSION WITH SLAVES ONLY

Tables C-1 and C-2 show an expansion chassis containing communication controllers. Except for the TILINE coupler, the expansion chassis contains only slave devices. In this system, the direction the vector is pointing on the control cable does not matter, since the possibility of having simultaneous coupler requests for the TILINE does not exist. Since the expansion chassis contains only slave devices, option switch 12 can be closed on the main coupler to allow access to all the CI403s in the expansion chassis. Later examples demonstrate that when the expansion (or remote) chassis contains a master device, option switch 12 must be open.

In Tables C-1 and C-2, all the CI403s in the expansion chassis are connected on the same interrupt level, which necessitates a programmable interrupt card. The CI403s are then fed back to the main chassis on the couplers' P2 direct interrupt line. The P2 interrupt is enabled with option switch 1 closed on the expansion coupler and option switch 2 closed on the main coupler (as described in Tables C-1 and C-2).

NOTE

In Tables C-1 through C-8, each CI403 full slot board has four ports, with one 931 VDT connected to each port.

Table C-1. Main Chassis TILINE Expansion — Slaves Only

Main Chassis — S800 (Includes Standard Interrupt Card)						
Slot	**Address**	**P1 Circuit Board**	**Interrupt Level**	**Address**	**P2 Circuit Board**	**Interrupt Level**
1	N/A	990/12 LR SMI	N/A	N/A	990/12 LR SMI	N/A
2	N/A	990/12 LR AU	N/A	N/A	990/12 LR AU	N/A
3	>FB10	512KB Cache Controller	N/A	>FB10	512K Cache Controller	N/A
4	N/A	1MB Memory Array	N/A	N/A	1MB Memory Array	N/A
5	>0B00	810 on CI402	7	>0200	3270 on BCAIM	3
	and >0B80	810 on CI402	7			
6	>FA80	TILINE Coupler	11	>FA80	TILINE Coupler	11
7	>F800	Disk Control A	9	>F800	Disk Control A	13
8	>F880	Tape	8	>F880	Tape	14
9	>F9A0	931 on CI403	N/A	>F9A0	931 on CI403	10
10	>F980	931 on CI403	12	>F980	931 on CI403	11
11	>00A0		3	>0080	LP600 on LP1	7
12	>F990	931 on CI403	14	>F990	931 on CI403	4
13	>0020	Auto Call	15	>0000	3780 on BCAIM	6

Examples of switch settings for the main chassis TILINE coupler are as follows:

Switch	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Option Switch	OFF	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON
Lower Limit	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF				
Upper Limit	ON	ON	ON	ON	ON	ON	ON	ON				
Bias Switches	X	X	X	X	X	X	X	X				
TPCS Base	OFF	ON	OFF	ON	ON	ON						

Switch	**Memory Address Window**
Lower limit Upper limit Bias	No address window

SW8-SW12 are TPCS address block enables.

Vector points in either direction.

Table C-2. Expansion Chassis TILINE Expansion — Slaves Only

Expansion Chassis — S800 (Includes Programmable Interrupt Card)						
Slot	Address	P1 Circuit Board	Interrupt Level	Address	P2 Circuit Board	Interrupt Level
1	>FAA0	TILINE Coupler		>FAA0	TILINE Coupler	
2						
3						
4						
5						
6						
7	>FA10	931 on CI403		>FA10	931 on CI403	11
8	>FA00	931 on CI403		>FA00	931 on CI403	11
9	>F9F0	931 on CI403		>F9F0	931 on CI403	11
10	>F9E0	931 on CI403		>F9E0	931 on CI403	11
11	>F9D0	931 on CI403		>F9D0	931 on CI403	11
12	>F9C0	931 on CI403		>F9C0	931 on CI403	11
13	>F9B0	931 on CI403		>F9B0	931 on CI403	11

Examples of switch settings for the expansion chassis TILINE coupler are as follows:

Switch	1	2	3	4	5	6	7	8	9	10	11	12
Option Switches	ON	OFF	OFF	ON	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF
Lower Limit	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF				
Upper Limit	ON	ON	ON	ON	ON	ON	ON	ON				
Bias Switches	X	X	X	X	X	X	X	X				
TPCS Base	OFF	ON	OFF	ON	OFF	ON						

Switch	Memory Address Window
Lower Limit Upper Limit Bias	No address window

SW8-SW12 are TPCS address block enables.

Vector points in either direction.

C.3 TILINE EXPANSION WITH MASTER AND SLAVE DEVICES

In Tables C-3 and C-4, the TILINE is expanded to both master and slave devices. In this configuration, the CPU accesses only TPCS addresses in the remote chassis, as in Tables C-1 and C-2. Cycles originating in the expansion chassis are transfers between disk control B and main chassis memory. Thus the possibility of a *wait state* exists, and only valid addresses should be sent across the couplers (in either direction). That is, only addresses that are expected to access a device in the other chassis should be sent. TILINE cycles from the remote to the main chassis are all memory cycles, and the only requirements are setting the upper limit, lower limit, and bias switches on the remote coupler (as described in Tables C-3 and C-4).

TILINE cycles from the main to remote chassis are all TPCS cycles. The CPU must be allowed to access all the TPCS addresses in the remote chassis, but when the CPU accesses a local TPCS address (such as the system disk at >F800), that address must not be allowed across the coupler. To accomplish this, allocate all the TPCS addresses of the main chassis into blocks 0 and 1, and allocate all the TPCS addresses of the remote chassis into blocks 2 and 3. The TPCS block enable switches (option switches 8 through 12) are then set on the main coupler as described in Table C-3.

NOTE

The TILINE coupler automatically prevents its own TPCS address from going across to the other chassis, so it is not necessary to restrict this address to any particular block.

In this system, the vector direction is important in determining which chassis has priority when simultaneous transfers across the couplers occur. The vector direction should point away from the slowest master device that is transferring data across the couplers. In most cases, this means the vector direction should be pointed from the remote to the main chassis, as noted in Tables C-3 and C-4.

As described in Tables C-1 and C-2, the interrupts from the CI403s in the expansion chassis are fed back to the main chassis through the P2 interrupt of the couplers. The interrupt for disk control B is fed back on the P1 interrupt line. To allow the P1 interrupt, option switch 7 is closed on the expansion coupler, and option switch 5 is closed on the main coupler.

Table C-3. Master Chassis TILINE Expansion — Multiprocessing and Slaves

Master Chassis — S800 (Includes Standard Interrupt Card)						
Slot	**Address**	**P1 Circuit Board**	**Interrupt Level**	**Address**	**P2 Circuit Board**	**Interrupt Level**
1	N/A	990/12 LR SMI	N/A	N/A	990/12 LR SMI	N/A
2	N/A	990/12 LR AU	N/A	N/A	990/12 LR AU	N/A
3	>FB10	512KB Cache Controller	N/A	>FB10	512KB Cache Controller	N/A
4	N/A	1MB Memory Array	N/A	N/A	1MB Memory Array	N/A
5	>0B00 or >0B80	810 on CI402 810 on CI402	7 7	>0200	3270 on BCAIM	3
6	>F980	931 on CI403	11	>F980	931 on CI403	11
7	>F800	Disk Control A	9	>F800	Disk Control A	13
8	>F880	Tape	8	>F880	Tape	14
9	>FAA0	TILINE Coupler	8	>FAA0	TILINE Coupler	10
10	>F990	931 on CI403	12	>F990	931 on CI403	11
11	>00A0		3	>0080	LP600 on LPI	7
12	>F9C0	931 on CI403	14	>F9C0	931 on CI403	4
13	>0020	Auto Call	15	>0000	3780 on BCAIM	6

Examples of switch settings for the chassis A TILINE coupler are as follows:

Switch	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Option Switch	OFF	ON	ON	OFF	ON	OFF	OFF	OFF	OFF	ON	ON	OFF
Lower Limit	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF				
Upper Limit	ON	ON	ON	ON	ON	ON	ON	ON				
Bias Switches	X	X	X	X	X	X	X	X				
TPCS Base	OFF	ON	OFF	OFF	ON							

Switch	**Memory Address Window**
Lower limit Upper limit Bias	No address window

SW8–SW12 are TPCS address block enables.

Vector points toward main chassis.

Table C-4. Expansion Chassis TILINE Expansion — Multiprocessing and Slaves

Expansion Chassis — S800 (Includes Programmable Interrupt Card)						
Slot	Address	P1 Circuit Board	Interrupt Level	Address	P2 Circuit Board	Interrupt Level
1	>FAC0	TILINE Coupler		>FAC0	TILINE Coupler	
2						
3						
4						
5						
6		931 on CI403		>FA60	931 on CI403	11
7		931 on CI403		>FA50	931 on CI403	11
8		931 on CI403		>FA40	931 on CI403	11
9		931 on CI403		>F9F0	931 on CI403	11
10		931 on CI403		>F9E0	931 on CI403	11
11		931 on CI403		>F9D0	931 on CI403	11
12		931 on CI403		>F9C0	931 on CI403	11
13	>F840	Disk Control B	12	>F840	Disk Control B	

Examples of switch settings for the chassis B TILINE coupler are as follows:

Switch	1	2	3	4	5	6	7	8	9	10	11	12
Option Switches	ON	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF	OFF	OFF	OFF
Lower Limit	ON	ON	ON	ON	ON	ON	ON	ON				
Upper Limit	OFF	ON	OFF	OFF	OFF	OFF	OFF	OFF				
Bias Switches	ON	ON	ON	ON	ON	ON	ON	ON				
TPCS Base	OFF	ON	OFF	OFF	ON	ON						

Switch	Memory Address Window
Lower Limit	>000000
Upper Limit	>17FFFF
Bias	No bias

SW8–SW12 are TPCS address block enables.

Vector points toward main chassis.

C.4 TILINE EXPANSION IN MULTIPROCESSING WITH SHARED MEMORY

Tables C-5 and C-6 describe TILINE expansion in a multiprocessor system with shared memory in chassis B. In this system, CPU A has access to 1.75M bytes of physical memory. This accessible memory consists of the 1.5M bytes in CPU A's own chassis, plus the last 256K bytes of chassis B's memory. The last 256K bytes of chassis B's memory appear to CPU A in the address range from >180000 to >1BFFFF. The lower limit, upper limit, and bias switches on the coupler in chassis A are set as described in Tables C-5 and C-6.

CPU B, on the other hand, has access to 1.5M bytes of memory, all physically located in its own chassis. Thus, the last 256K bytes in chassis B are shared by both CPUs. To CPU B, the shared memory space appears in the range from >140000 to >17FFFF. The shared memory space can be used for exchanging blocks of data, maintaining multiprocessing status flags, and so on.

In addition to sharing some of chassis B's memory, both CPUs may require access to either coupler's TPCS. As described in Tables C-3 and C-4, it is extremely important that all local TPCS addresses *not* be allowed across the couplers, while the couplers must still allow access to certain remote TPCS addresses. This is accomplished by the following steps:

1. Allocate remote TPCS addresses that need to be accessed into different TPCS blocks from those for local TPCS addresses.

2. Set the correct TPCS block enable switches on each coupler.

In this example, coupler A is in TPCS block 1, coupler B is in TPCS block 3, and all other TPCS addresses are in TPCS blocks 0 and 2. The TPCS block enable switches of each coupler are set as described in Tables C-5 and C-6. Each chassis can have many of the same TPCS addresses allocated (including system disks at >F800), as long as those addresses are not enabled across the couplers.

Communication between the two 990/12s can be accomplished by using either coupler's multiprocessor interface. (See Appendix A.) However, interprocessor communication requires software not presently supplied by Texas Instruments.

Table C-5. Chassis A TILINE Expansion With Multiprocessing and Shared Memory

Chassis B — S800 (Includes Standard Interrupt Card)						
Slot	P1 Address	P1 Circuit Board	Interrupt Level	P2 Address	P2 Circuit Board	Interrupt Level
1	N/A	990/12 LR SMI	N/A	N/A	990/12 LR SMI	N/A
2	N/A	990/12 LR AU	N/A	N/A	990/12 LR AU	N/A
3	>FB10	512KB Cache Controller	N/A	>FB10	512KB Cache Controller	N/A
4		1MB Memory Array	N/A		1MB Memory Array	N/A
5	>0B00 and >0B80	810 on CI402 810 on CI402	7 7	>0200	3270 on BCAIM	3
6	>F990	931 on CI403	11	>F990	931 on CI403	11
7	>F800	Disk Control B	9	>F800	Disk Control B	13
8	>F880	Tape	8	>F880	Tape	14
9	>FAE0	TILINE Coupler	8	>FAE0	TILINE Coupler	10
10	>F980	931 on CI403	12	>F980	931 on CI403	11
11	>00A0		3	>0080	LP600 on LP1	7
12	>F9C0	931 on CI403	14	>F9C0	931 on CI403	4
13	>0020	Auto Call	15	>0000	3780 on BCAIM	6

Examples of switch settings for the chassis A TILINE coupler are as follows:

Switch	1	2	3	4	5	6	7	8	9	10	11	12
Option Switches	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF	OFF
Lower Limit	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF				
Upper Limit	ON	ON	ON	ON	ON	ON	ON	ON				
Bias Switches	X	X	X	X	X	X	X	X				
TPCS Base	OFF	ON	OFF	OFF	OFF	ON						

Switch	Memory Address Window
Lower limit Upper limit Bias	No Address Window

SW8–SW12 are TPCS address block enables.

Vector points in either direction.

Table C-6. Chassis B TILINE Expansion With Multiprocessing and Shared Memory

Chassis B — S800 (Includes Standard Interrupt Card)						
Slot	**Address**	**P1 Circuit Board**	**Interrupt Level**	**Address**	**P2 Circuit Board**	**Interrupt Level**
1	N/A	990/12 LR SMI	N/A	N/A	990/12 LR SMI	N/A
2	N/A	990/12 LR AU	N/A	N/A	990/12 LR AU	N/A
3	>FB10	512KB Cache Controller	N/A	>FB10	512KB Cache Controller	N/A
4	N/A	1MB Memory Array	N/A	N/A	1MB Memory Array	N/A
5	>0B00 or >0B80	810 on CI402	7	>0200	3270 on BCAIM	3
		810 on CI402	7			
6	>F990	931 on CI403	11	>F990	931 on CI403	11
7	>F800	Disk Control B	9	>F800	Disk Control B	13
8	>F880	Tape	8	>F880	Tape	14
9	>FAE0	TILINE Coupler	8	>FAE0	TILINE Coupler	10
10	>F980	931 on CI403	12	>F980	931 on CI403	11
11	>00A0		3	>0080	LP600 on LPI	7
12	>F9C0	931 on CI403	14	>F9C0	931 on CI403	4
13	>0020	Auto Call	15	>0000	3780 on BCAIM	6

Examples of switch settings for the chassis B TILINE coupler are as follows:

Switch	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Option Switches	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF	OFF
Lower Limit	OFF	OFF	ON	ON	ON	ON	ON	ON				
Upper Limit	OFF	OFF	ON	ON	OFF	OFF	OFF	OFF				
Bias Switches	OFF	OFF	ON	ON	ON	ON	ON	ON				

Switch	**Address Window**
Lower Limit	>180000
Upper Limit	>19FFFF
Bias	> −080000

SW8–SW12 are TPCS address block enables.

Vector points in either direction.

C.5 TILINE EXPANSION IN MULTIPROCESSING WITH SHARED RESOURCES

Tables C-7 and C-8 show TILINE expansion in a multiprocessing system, with a shared memory space in each chassis and a shared disk controller in chassis B. In this system, CPU A's address space looks exactly the same as that in Tables C-5 and C-6, including the memory that is shared with CPU B. The address space for CPU B, on the other hand, now ranges from >000000 to >17FFFF (1.5M bytes of memory located in chassis B) and from >180000 to >19FFFF (128K bytes of memory located in chassis A). To allow CPU B access to the 128K bytes in chassis A, the lower limit, upper limit, and bias switches on the coupler in chassis B are set as described in Tables C-7 and C-8. These 128K bytes of memory also appear in the middle of CPU A's address space (the middle was an arbitrary choice) in the range from >100000 to >11FFFF. Thus, both CPUs share a 256K-byte memory space in chassis B and a 128K-byte memory space in chassis A.

In addition to the shared memory, the CPUs share use of disk control B. As in previous examples, local TPCS addresses must not be allowed across the couplers, but the couplers must still allow any remote TPCS addresses that need to be accessed to cross. In this example, however, the two system disks, disk control A and disk control B, should be kept at >F800 to be consistent with the recommended TPCS addresses. For both system disks to remain at >F800 and still allow CPU A to access disk control B, the operating system in chassis A must use the TPCS bias enable bits of TILINE coupler A. Setting word 0, bits 2 and 3 of coupler A causes block 3 TPCS addresses to be mapped to block 0 in the remote chassis. TPCS block 3 addresses are enabled by option switch 11, which allows CPU A to access disk control B by accessing >F860. Placing TPCS block 3 in the biased mode does not affect TPCS blocks 1 or 2. As described in Tables C-7 and C-8, option switch 9 is also closed (enabling block 1) on the coupler in chassis A, which allows CPU A to access the coupler in chassis B at >FAB0.

With the two shared memory spaces and the shared disk controller, the following transfers are possible:

- CPU A can directly access the shared memory spaces in either chassis.
- CPU B can directly access the shared memory spaces in either chassis.
- CPU B can direct disk control B to execute transfers between the disk and the shared memory space in chassis B.
- CPU B can direct disk control B to execute transfers between the disk and the shared memory space in chassis A.
- CPU A can direct disk control B to execute transfers between the disk and the shared memory space in chassis B.
- CPU A can direct disk control B to execute transfers between the disk and the shared memory space in chassis A.

The last type of transfer is possibly the most interesting demonstration of the TILINE coupler's capabilities. To CPU A, the shared 128K-byte memory space in chassis A appears directly as >100000 to >11FFFF, but to disk control B, this same memory space appears as >180000 to >19FFFF. CPU A must determine how addresses on chassis B's TILINE are mapped to chassis A's TILINE. CPU A may not have any prior knowledge as to where the shared memory space is located in chassis A, or even how large the space is. All this information—the address window as it appears in chassis B, the location of the memory space within chassis A, and the size of this memory space—is available to CPU A by accessing the TPCS of chassis B's coupler. Once CPU A has enabled access to chassis B's coupler (by enabling TPCS biasing of block 3 on the coupler in chassis A), it can read the switch settings of the lower limit, the upper limit, and the bias, and calculate all the necessary information.

NOTE

CPU B can access disk control A in a similar fashion. By closing option switch 11 on the coupler in chassis B, CPU B can set up biasing of TPCS block 3, just as CPU A did, and access disk control A at >F860. Meanwhile, CPU B can still access the coupler in chassis A at >FAA0.

The preferred vector direction for this example is towards chassis A, which gives priority to disk control B (slowest master device transferring data across couplers). Note that no interrupt exists between disk control B and CPU A. This example assumes that CPU A's device service routine (DSR) uses polling of disk control B, and that interprocessor communication prevents the CPUs from attempting to send commands simultaneously.

As described in Tables C-5 and C-6, communication between the two 990/12s can be accomplished by using either coupler's multiprocessor interface. (See Appendix A.) Polling of disk control B, interprocessor communication, and control of the TPCS bias enable bits all require special software not presently supplied by Texas Instruments.

Table C-7. Chassis A TILINE Expansion — Multiprocessing and Shared Resources

Chassis A — S800 (Includes Standard Interrupt Card)						
Slot	Address	P1 Circuit Board	Interrupt Level	Address	P2 Circuit Board	Interrupt Level
1	N/A	990/12 LR SMI	N/A	N/A	990/12 LR SMI	N/A
2	N/A	990/12 LR AU	N/A	N/A	990/12 LR AU	N/A
3	>FB10	512KB Cache Controller	N/A	>FB10	512K Cache Controller	N/A
4	N/A	1MB Memory Array	N/A	N/A	1MB Memory Array	N/A
5	>0B00 or >0B80	810 on CI402 810 on CI402	7 7	>0200	3270 on BCAIM	3
6	>F980	931 on CI403	11	>F980	931 on CI403	11
7	>F800	Disk Control A	9	>F800	Disk Control A	13
8	>F880	Tape	8	>F880	Tape	14
9	>FAA0	TILINE Coupler	8	>FAA0	TILINE Coupler	10
10	>F990	931 on CI403	12	>F990	931 on CI403	11
11	>00A0		3	>0080	LP600 on LPI	7
12	>F9C0	931 on CI403	14	>F9C0	931 on CI403	4
13	>0020	Auto Call	15	>0000	3780 on BCAIM	6

Examples of switch settings for the chassis A TILINE coupler are as follows:

Switch	1	2	3	4	5	6	7	8	9	10	11	12
Option Switch	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	ON	OFF
Lower Limit	OFF	OFF	ON	ON	ON	ON	ON	ON				
Upper Limit	OFF	OFF	ON	OFF	OFF	OFF	OFF	OFF				
Bias Switches	OFF	OFF	OFF	ON	ON	ON	ON	ON				
TPCS Base	OFF	ON	OFF	ON	OFF	ON						

Switch	Memory Address Window
Lower limit	>180000
Upper limit	>1BFFFF
Bias	> −040000

SW8–SW12 are TPCS address block enables.

Vector points toward Chassis A.

Table C-8. Chassis B TILINE Expansion — Multiprocessing and Shared Resources

Chassis B — S800 (Includes Standard Interrupt Card)						
Slot	**Address**	**P1 Circuit Board**	**Interrupt Level**	**Address**	**P2 Circuit Board**	**Interrupt Level**
1	N/A	990/12 LR SMI	N/A	N/A	990/12 LR SMI	N/A
2	N/A	990/12 LR AU	N/A	N/A	990/12 LR AU	N/A
3	>FB10	512KB Cache Controller	N/A	>FB10	512KB Cache Controller	N/A
4	N/A	1MB Memory Array	N/A	N/A	1MB Memory Array	N/A
5	>0B00 or >0B80	810 on CI402 810 on CI402	7 7	>0200	3270 on BCAIM	3
6	>F990	931 on CI403	11	>F990	931 on CI403	11
7	>F800	Disk Control B	9	>F800	Disk Control B	13
8	>F880	Tape	8	>F880	Tape	14
9	>FAE0	TILINE Coupler	8	>FAE0	TILINE Coupler	10
10	>F980	931 on CI403	12	>F980	931 on CI403	11
11	>00A0		3	>0080	LP600 on LPI	7
12	>F9C0	931 on CI403	14	>F9C0	931 on CI403	4
13	>0020	Auto Call	15	>0000	3780 on BCAIM	6

Examples of switch settings for the chassis B TILINE coupler are as follows:

Switch	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Option Switches	OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF	OFF
Lower Limit	OFF	OFF	ON	ON	ON	ON	ON	ON				
Upper Limit	OFF	OFF	ON	ON	OFF	OFF	OFF	OFF				
Bias Switches	OFF	OFF	ON	ON	ON	ON	ON	ON				

Switch	**Address Window**
Lower Limit	>180000
Upper Limit	>19FFFF
Bias	> −080000

SW8–SW12 are TPCS address block enables.

Vector points toward Chassis A.

Alphabetical Index

Introduction

HOW TO USE INDEX

The index, table of contents, list of illustrations, and list of tables are used in conjunction to obtain the location of the desired subject. Once the subject or topic has been located in the index, use the appropriate paragraph number, figure number, or table number to obtain the corresponding page number from the table of contents, list of illustrations, or list of tables.

INDEX ENTRIES

The following index lists key words and concepts from the subject material of the manual together with the area(s) in the manual that supply major coverage of the listed concept. The numbers along the right side of the listing reference the following manual areas:

- Sections — Reference to Sections of the manual appear as "Sections x" with the symbol x representing any numeric quantity.

- Appendixes — Reference to Appendixes of the manual appear as "Appendix y" with the symbol y representing any capital letter.

- Paragraphs — Reference to paragraphs of the manual appear as a series of alphanumeric or numeric characters punctuated with decimal points. Only the first character of the string may be a letter; all subsequent characters are numbers. The first character refers to the section or appendix of the manual in which the paragraph may be found.

- Tables — References to tables in the manual are represented by the capital letter T followed immediately by another alphanumeric character (representing the section or appendix of the manual containing the table). The second character is followed by a dash (-) and a number.

  Tx-yy

- Figures — References to figures in the manual are represented by the capital letter F followed immediately by another alphanumeric character (representing the section or appendix of the manual containing the figure). The second character is followed by a dash (-) and a number.

  Fx-yy

- Other entries in the Index — References to other entries in the index preceded by the word "See" followed by the referenced entry.

USER'S RESPONSE SHEET

Manual Title: 990 CRU/TILINE Expansion Installation and Operation (2272075-9701)

Manual Date: March 1984 **Date of This Letter:** _____

User's Name: _____ **Telephone:** _____

Company: _____ **Office/Department:** _____

Street Address: _____

City/State/Zip Code: _____

Please list any discrepancy found in this manual by page, paragraph, figure, or table number in the following space. If there are any other suggestions that you wish to make, feel free to include them. Thank you.

Location in Manual	Comment/Suggestion
_____	_____
	_____
	_____
	_____
_____	_____
	_____
	_____
	_____
_____	_____
	_____
	_____
	_____

CUT ALONG LINE

NO POSTAGE NECESSARY IF MAILED IN U.S.A.
FOLD ON TWO LINES (LOCATED ON REVERSE SIDE), TAPE AND MAIL

FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 7284 DALLAS, TX

POSTAGE WILL BE PAID BY ADDRESSEE

TEXAS INSTRUMENTS INCORPORATED
DATA SYSTEMS GROUP

ATTN: TECHNICAL PUBLICATIONS
P.O. Box 2909 M/S 2146
Austin, Texas 78769

FOLD