General Description

Model 990/12 LR Computer

Part No. 2268239-9701 *A
15 October 1983

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INSERT LATEST CHANGED PAGES AND DISCARD SUPERSEDED PAGES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.

Model 990/12 LR Computer (2268239-9701)

Original Issue . . . . 1 January 1983
Change 1 . . . . 15 October 1983

Total number of pages in this publication is 46 consisting of the following:

PAGE NO.	CHANGE NO.	PAGE NO.	CHANGE NO.	PAGE NO.	CHANGE NO.
Cover	1				
Effective Pages	1				
iii/iv	1				
v - vi	0				
1-1 - 1-6	0				
2-1	0				
2-2	1				
3-1 - 3-9	0				
3-10	1				
3-11 - 3-26	0				
Index-1 - Index-2	0				
User's Response	1				
Business Reply	1				
Inside Cover	1				
Cover	1				

Preface

This manual contains general and installation information as well as a functional description of the 990/12 LR Central Processing Unit (CPU). The 990/12 LR CPU is modified to exhibit low radiation characteristics.

This manual is organized into three sections as follows:

Section

1 General Information — Contains a general description of the 990/12 LR CPU and also contains a functional and physical description with emphasis on the operation of the CPU within a 990 minicomputer system.

2 Installation — Contains information needed to unpack/pack, inspect, and install the 990/12 LR SMI board, part number 2309325-0001, and the 990/12 LR AU board, part number 2309320-0001.

3 Functional Description — Contains a functional description of the 990 concept, and in particular of the 990/12 LR Central Processor.

Additional information related to the 990/12 LR Central Processor can be found in the following **manuals:**

Title	Part Number
ROM Loader User's Guide	2270534-9701
Model 990/12 Computer Assembly Language Programmer's Guide	2250077-9701
Model 990/12 LR Computer, Field Theory and Maintenance (available as part of manual kit 2268237-0001)	2268240-9701
Model 990/12 LR Computer, Depot Theory and Maintenance (available as part of manual kit 2268237-0002)	2268241-9701

Contents

Illustrations

Tables

1

General Information

1.1 GENERAL

This manual contains a general description of the Model 990/12 LR Central Processing Unit (CPU) manufactured by Texas Instruments Incorporated.

This section contains functional and physical descriptions of the CPU with emphasis on the operation of the CPU within a 990 minicomputer system. Additionally, electrical characteristics of the CPU are defined.

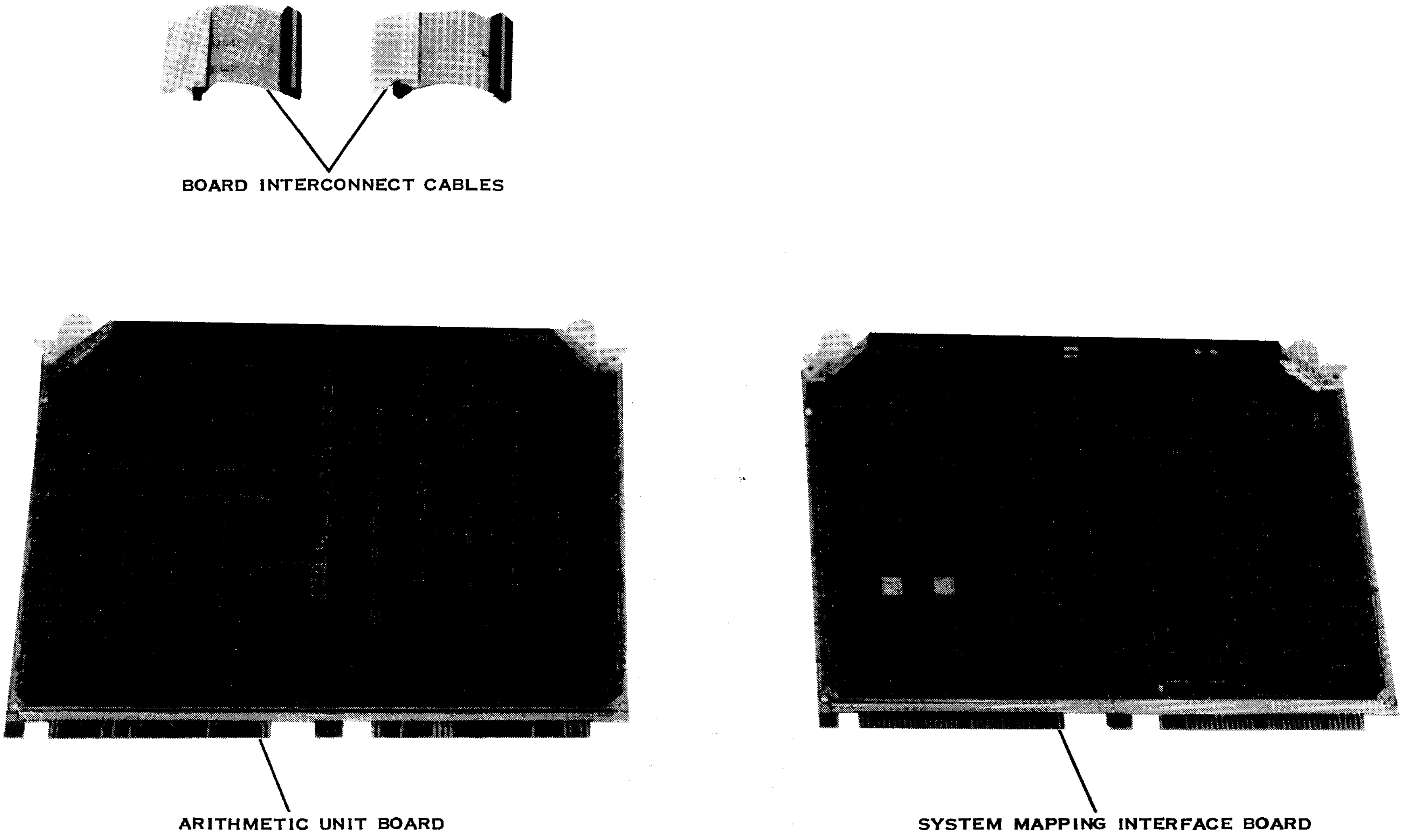
The CPU (Figure 1-1) is implemented with transistor, transistor logic (TTL) Schottky and low-power Schottky technology on two multilayer circuit boards: the arithmetic unit (AU) board, and the system mapping interface (SMI) board.

1.2 PURPOSE OF EQUIPMENT

When installed in a 990 chassis, the two boards of the 990/12 LR CPU and a memory board function as a 990/12 LR minicomputer. When augmented with interface modules, input/output modules, and optional additional memory, the 990/12 LR minicomputer becomes a high speed, flexible, powerful minicomputer system capable of handling a wide range of computer applications at a low cost.

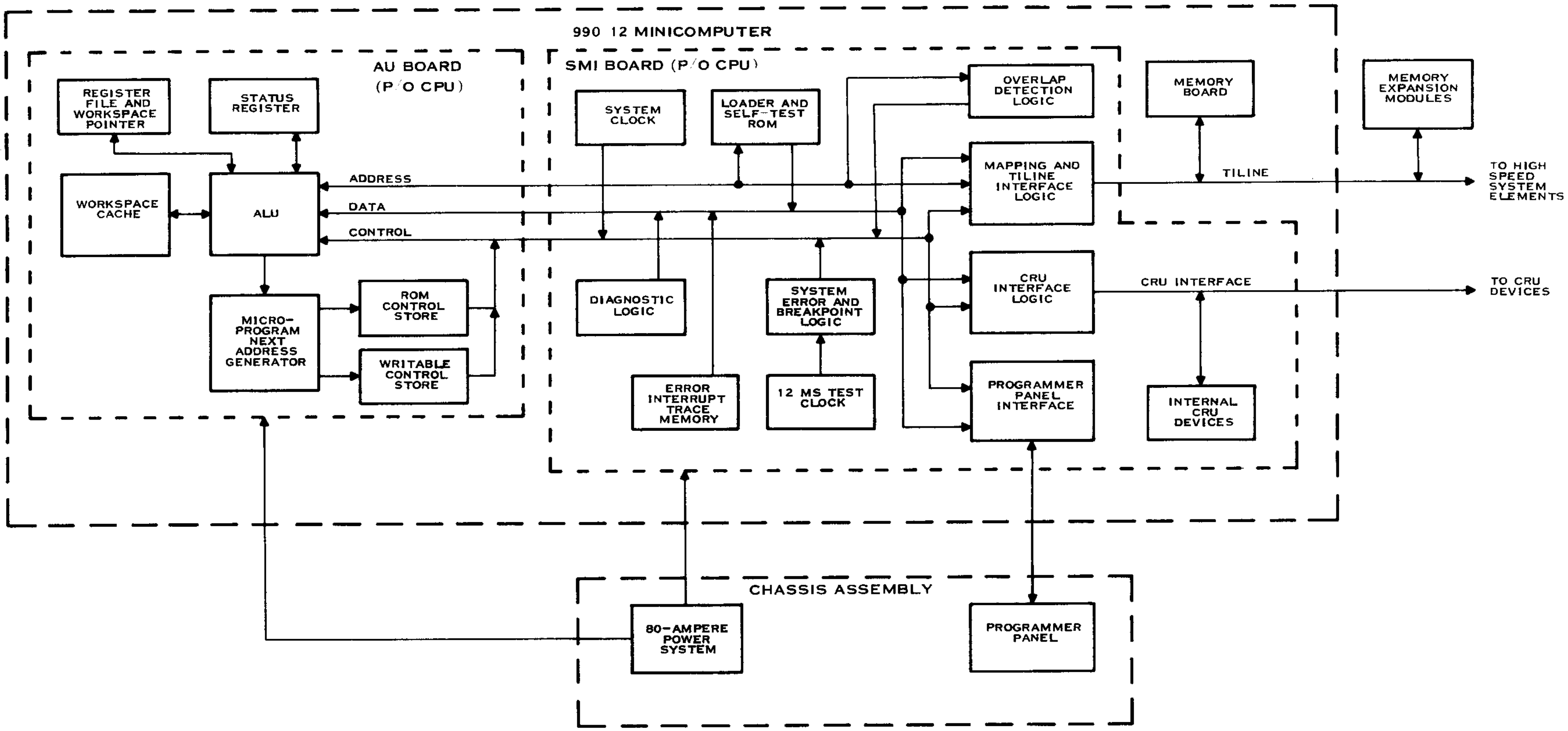
1.3 HARDWARE DESCRIPTION

The 990/12 LR CPU is a full-function processor, incorporating floating point arithmetic, byte string operation, bit-array instructions, and multiprecision integer and decimal conversion. The 990/12 LR CPU uses Texas Instruments 74S481 and 74S482 large scale integration (LSI) processor elements and has an instruction set that is expanded more than either the 990/10 or 990/10A CPU. The block diagram of Figure 1-2 shows the operation of the 990/12 LR CPU as part of a minicomputer system.



2283708

Figure 1-1. Model 990/12 LR Computer Central Processor Boards



2280063

Figure 1-2. Model 990/12 LR Minicomputer System Block Diagram

Features of the CPU include the following:

- 4K 16-bit words of mask, read-only memory (ROM) installed on the SMI board for front panel, loader and self-test programs
- 16 vectored interrupts (13 external)
- Real-time clock
- TILINE* bus interface to memory, memory expansion, and high-speed direct memory-access controllers
- Communications register unit (CRU) interface for communications link with a variety of external serial and parallel data transfer devices
- One CRU interface for the front panel
- Error interrupt logic
- Processor capability that can access from 64K bytes to 2M bytes of memory through mapping logic contained on the SMI board
- Error interrupt trace memory that permits software to characterize error interrupts
- Workspace cache, which is a 16-register bipolar memory located on the AU board. In this cache, data accessed as workspace registers is stored as it is fetched from TILINE memory. Subsequent accesses to the same data are from the on-board cache; this alleviates the need for a memory cycle and decreases the amount of time required for processing.
- Writable control store (WCS). The WCS is a 1024 by 64-bit memory containing microcode instructions (except on power-up). A Load Control Store instruction writes and stores instructions in the WCS. Control is transferred to the WCS by a hardware Extended Operation (XOP) instruction with status register bit 11 set to one.

1.3.1 Arithmetic Unit Board

The AU board is a two-edged printed circuit board that contains the TTL devices. The TTL devices implement the arithmetic logic unit (ALU) and read-only memory (ROM) microsequencing and control, as described in the following list:

- Four cascaded SN74S481 4-bit slice processor elements used as the ALU of the AU board. The SN74S481 is a highly complex Schottky TTL processor element and is the only bipolar micro/macroprogrammable element featuring automatically sequenced iterative multiply and divide and cyclical-redundancy algorithms.
- Three SN74S482 4-bit slice expandable control elements cascaded to perform next-address generation functions. The three elements can address up to 4096 words of microprogram. Comprised of an output register, a push-pop stack and a full adder, the SN74S482 provides the capability to implement multiway testing needed to generate the next microprogram address.

* TILINE is a trademark of Texas Instruments Incorporated.

- ROM Control Store. Bipolar programmable read-only memory (PROM) chips provide storage for 2048 64-bit microcode instruction words in the ROM control store.
- WCS. Sixteen fully static random-access memory (RAM) devices organized as 1024 words by 4 bits provide storage for 1024 64-bit microcode instruction words. The three-state common data I/O pins facilitate the interface to a bidirectional bus.

The AU board normally resides in slot 2 of the 13-slot chassis, and it connects to the SMI board by two ribbon cables that connect to the top edge of the two boards. The AU resides in slot 2 because of its proximity to slot 1 and the length of the ribbon cables. The interface to the TILINE data bus and to the chassis power supply for required voltages and the power reset signal is at the edge of **the AU board.**

1.3.2 System Mapping Interface Board

The SMI board contains the TILINE bus interface, the CRU interface, interrupt logic, loader and self-test ROM, the system clock, a 12-millisecond test clock, front panel interface, a breakpoint system, error interrupt trace memory, diagnostic registers that monitor selected signals for input to the AU board, and memory mapping. The SMI board installs into slot 1 of the chassis. The interface with the power supply, memory and peripheral devices is at the edge of the board that makes contact with the backpanel. As described in the previous paragraph, ribbon cables provide the interface to the AU board. A separate ribbon cable provides an interface to the front panel. Both interfaces are at the top edge of the SMI board.

1.4 ELECTRICAL CHARACTERISTICS

Power requirements for the CPU are as shown in Table 1-1.

Table 1-1. 990/12 LR CPU Power Requirements

Board	Voltage	Current
AU	+ 5.0 V (50 mV ripple)	12A
SMI	+ 5.0 V (50 mV ripple)	8A*

Note:

* Includes power drain of installed front panel.

2

Installation

2.1 INTRODUCTION

This section provides information needed to unpack/pack, inspect, and install the 990/12 LR SMI and AU boards. The procedures do not require a detailed or extensive knowledge of computer hardware or software. Refer to the installation procedures of the related system maintenance manual for additional board installation information.

2.2 UNPACKING

Boards shipped with systems are usually installed in the chassis, ready to use. However, boards shipped individually or as part of an upgrade kit are packed together in one box with an opaque gray antistatic film covering and a bubble-wrap material. Perform the following to unpack the **boards:**

1. To inspect the box for signs of possible damage to the contents, look for such signs as:

   a. Crumpled corners

   b. Tears

   c. Water stains

   d. Loosened packing tape

   Notify supervisory personnel of any abnormalities.

2. Open the box and remove the boards.

3. Carefully unwrap the bubble-wrap and antistatic material from the boards.

4. Verify that the assembly numbers on both the AU and SMI board actually match the numbers on the packing slip.

5. Save all packing material until you verify these numbers.

2.3 INSPECTION

Visually inspect the circuit boards before installing them in a chassis slot. Look for cracks, corrosion, loose components and any other damage to the board.

2.4 INSTALLATION AND CABLING

CAUTION

Always turn off power to the chassis before attempting logic board removal or installation. Failure to observe this precaution can result in board damage because connector pins are temporarily misaligned during board removal and installation.

To install both the AU and SMI boards in the computer chassis and to connect the ribbon cable to the top-edge connectors, perform the following:

1. Gain access to the chassis backpanel connectors as described in your computer system chassis manual.

2. Insert each board individually; put the SMI board in slot one and the AU board in slot two (though the AU can go in another slot provided the two ribbon cables will still connect the two boards). Push down evenly on both sides of the board until the board locks in the slot.

3. Align the ribbon cables so that the carets on the cable connector match the carets on the board connectors.

4. Push down firmly on the ribbon cable until it is seated evenly on the top-edge connector.

5. Remove the boards by reversing the previous steps; remember to turn the power to the chassis off before attempting removal of either board.

NOTE

When the system is equipped with a battery backup a jumper can be installed on the AU board at location GJ067. This jumper is in a plastic bag, attached to the left rear side of the chassis. Refer to paragraph 3.1.4 for installation instructions.

3

Functional Description

3.1 FUNCTIONAL DESCRIPTION OF THE CPU

The 990/12 LR minicomputer uses a 20-bit address and a 16 bit data bus. The 990 concept features multiple register files (16 registers) that reside in memory. The advanced architecture of the 990/12 LR CPU permits efficient programming with bit, byte and word addressing capability, and multiple register files allow rapid context switching. Table 3-1 shows the characteristics of the 990/12 LR.

Table 3-1. 990/12 LR CPU Characteristics

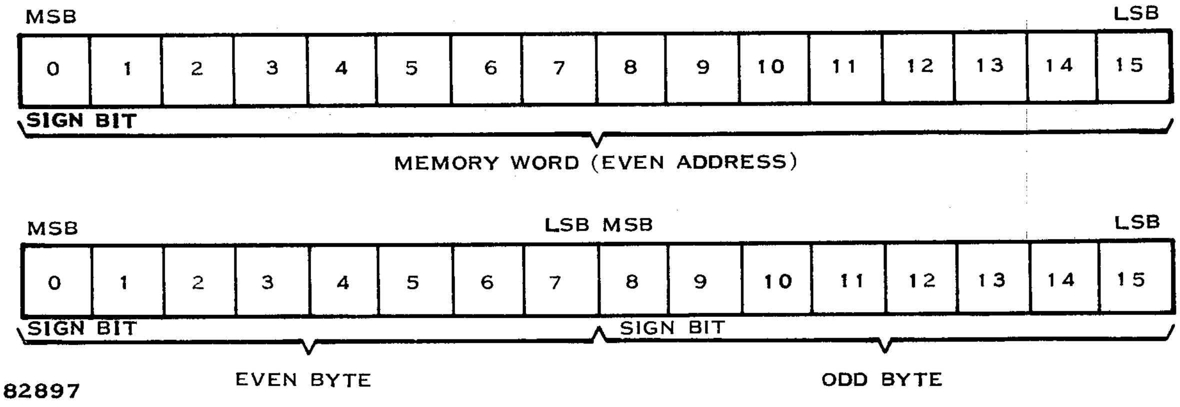
Item	Characteristic
Word Size	16 bits (2 bytes, 8 bits each)
Maximum memory addressing capability	64K bytes without mapping, 2M bytes with mapping
Clock rate	Approximately 4.5 MHz
Addressing modes	Immediate Workspace register Workspace indirect Symbolic memory (direct) Indexed memory Workspace register indirect Autoincrement Program counter relative CRU base register relative
Interrupts	16 interrupts, 13 external
Registers	16
Input/Output	CRU and TILINE bus
Address bus	16 bits, internal processor; 20 bits, TILINE
Data bus	16 bits
Power	± 5 Vdc
Board size	274 mm (10.25 in.) by 362 mm (14.25 in.)

The memory data word of the CPU is 16 bits, or 2 bytes of 8 bits each. The instruction set for the 990/12 LR permits both word and byte operands. Thus, all memory word locations are on even byte address boundaries. Byte instructions can address either the even or odd byte of a word. The memory space without mapping is 65,536 bytes, or 32,768 words. Figure 3-1 illustrates the word and byte formats of the processor.

The processor employs an advanced memory-to-memory architecture. Blocks of memory designated as workspace hold the program context. Figure 3-2 illustrates the memory map. The first 32 words are the 16 interrupt trap vectors, with the first 6 words reserved for internal interrupts and the remaining 26 words reserved for 13 external equipment interrupts. The next contiguous block of 32 memory words uses the XOP instructions for trap vectors. Those addresses in the range of >F800 through >FBFE (512 words) map to the TILINE peripheral control space (TPCS) to become >FFC00 through >FFDFE. The last 512 words at addresses >FC00 through >FFFE address the loader and self-test ROM on the SMI board. The remaining memory is available for programs, data and workspace registers; if desired, you can designate any of the special areas as general memory.

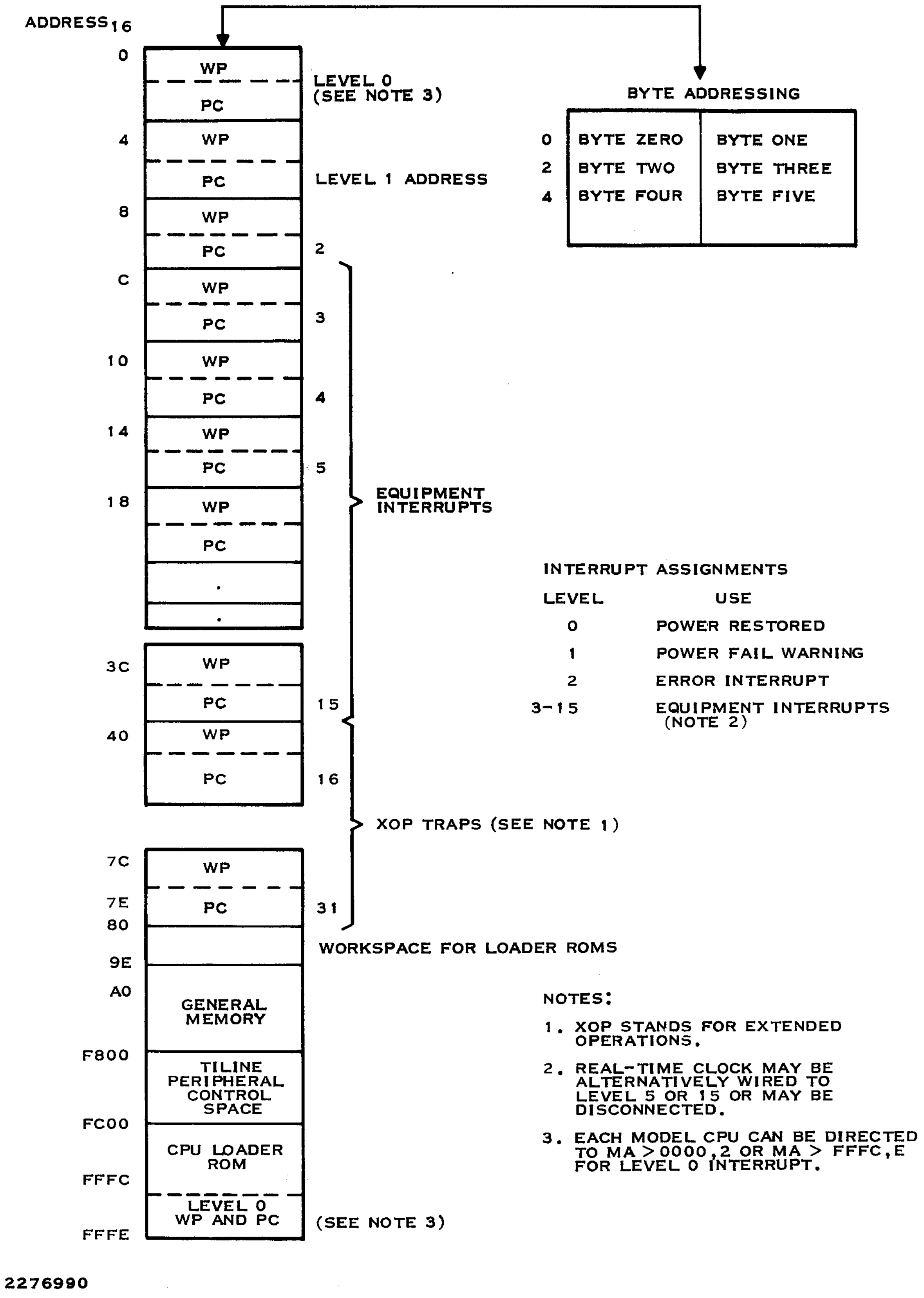
There are internal registers of the CPU that you can access. Of these, the program counter register (PC) and the memory counter register (MC) (accessible through the PC and MA ENTER switches, respectively, on the programmer panel) are integral to the ALU on the AU board. The PC and MC can hold or increment by one or two counts on each system clock to the ALU. The contents of either the PC or MC registers can be selected and applied to the address bus.

The status register (ST) contains the interrupt mask level and status information pertinent to the instruction operation. Each bit position in the register signifies a particular function or condition that exists in the processor. Figure 3-3 illustrates the bit position assignments. Some instructions use the ST to check for a prerequisite condition, others affect the values of the bits in the register, and others load the entire status register with a new set of parameters. A description of the instruction set in the *Model 990/12 Computer Assembly Language Programmer's Guide* details the effect of each instruction on the ST.



2282897

Figure 3-1. Processor Word and Byte Format



INTERRUPT ASSIGNMENTS

LEVEL	USE
0	POWER RESTORED
1	POWER FAIL WARNING
2	ERROR INTERRUPT
3-15	EQUIPMENT INTERRUPTS (NOTE 2)

NOTES:

1. XOP STANDS FOR EXTENDED OPERATIONS.
2. REAL-TIME CLOCK MAY BE ALTERNATIVELY WIRED TO LEVEL 5 OR 15 OR MAY BE DISCONNECTED.
3. EACH MODEL CPU CAN BE DIRECTED TO MA >0000,2 OR MA > FFFC,E FOR LEVEL 0 INTERRUPT.

Figure 3-2. 990/12 LR Processor Memory Map

The workspace pointer register (WP) contains the address of the first word in the currently active set of workspace registers. Figure 3-4 illustrates a workspace register file that occupies 16 contiguous memory words in the general memory area. Each workspace register holds data or addresses and functions as an operand register, accumulator, address register or index register. Some workspace registers take on special significance during execution of certain instructions. Table 3-2 lists dedicated workspace registers and instructions that use them. In addition to the list shown in Table 3-2, register R0 can contain the byte count for string instructions, and registers R0 through R3 are dedicated floating point accumulators in some arithmetic instructions, conversion instructions and load and move instructions. During an instruction execution, the processor addresses any register in the workspace by adding the register number to the contents of the workspace pointer (refer to Figure 3-4).

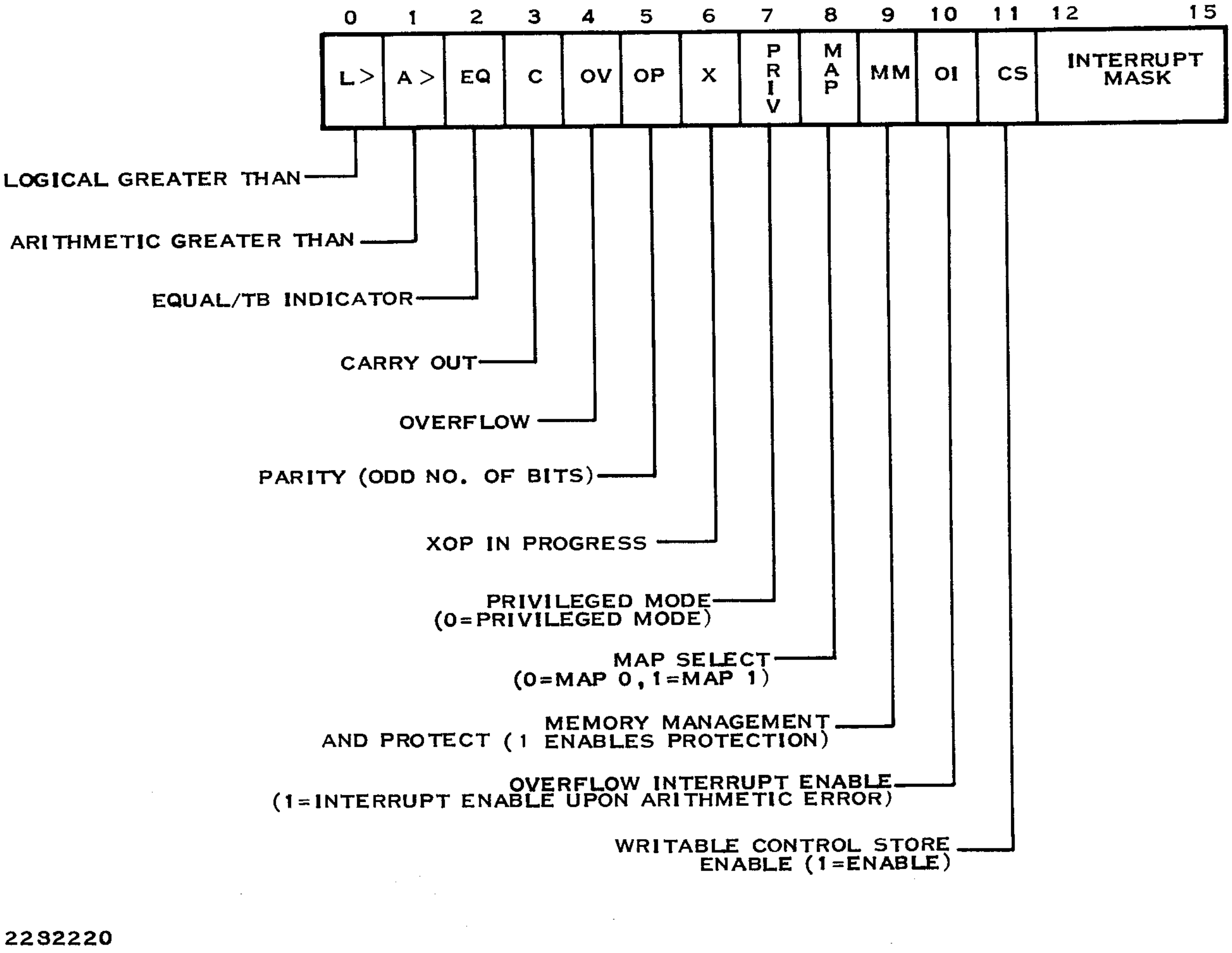
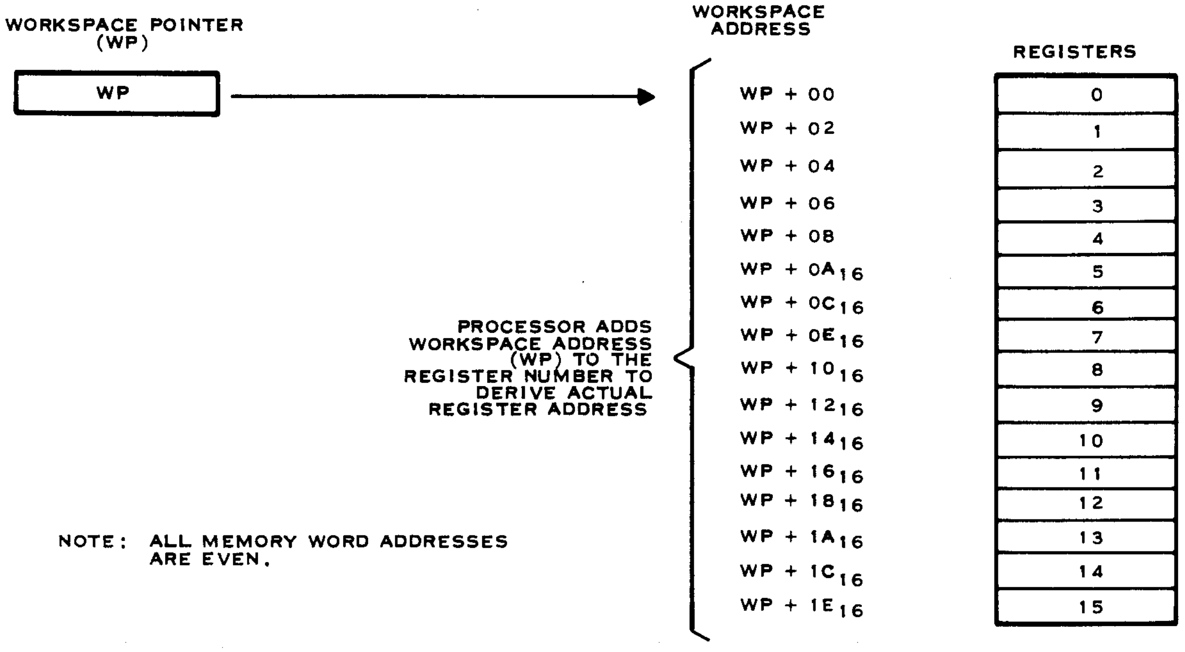

0
1
2
3
4
5
6
7
8
9
10
11
12
15
L>
A>
EQ
C
OV
OP
X
PRIV
MAP
MM
OI
CS
INTERRUPT MASK
LOGICAL GREATER THAN
ARITHMETIC GREATER THAN
EQUAL/TB INDICATOR
CARRY OUT
OVERFLOW
PARITY (ODD NO. OF BITS)
XOP IN PROGRESS
PRIVILEGED MODE (0=PRIVILEGED MODE)
MAP SELECT (0=MAP 0, 1=MAP 1)
MEMORY MANAGEMENT AND PROTECT (1 ENABLES PROTECTION)
OVERFLOW INTERRUPT ENABLE (1=INTERRUPT ENABLE UPON ARITHMETIC ERROR)
WRITABLE CONTROL STORE ENABLE (1=ENABLE)
2232220

Figure 3-3. 990/12 LR CPU Status Register Bit Assignments



2282900

Figure 3-4. Workspace Pointer and Registers

Table 3-2. Dedicated Workspace Registers

Register Number	Contents	Used During
0	Shift count (optional)	Shift instructions (SLA, SRA SRC, SRL)
11	Return address	Branch and Link (BL) instruction
	Effective address	Software implemented Extended Operation (XOP) instruction
12	CRU base address	CRU instructions (SBO, SBZ, TB, LDCR, and STCR)
13	Saved WP register	Context switching (BLWP, RTWP, software XOP, recognized interrupt, LOAD and RESET)
14	Saved PC register	Context switching (BLWP, RTWP, software XOP, recognized interrupt, LOAD and RESET)
15	Saved ST register	Context switching (BLWP, RTWP, software XOP, recognized interrupt, LOAD and RESET)

A unique feature of the 990/12 LR is the workspace cache, a 16-register bipolar memory located on the AU board. The cache fetches the accessed data of the WP registers from TILINE memory and stores it. Subsequent accesses to the same workspace data are from the on-board workspace cache. Having the WP cache data on board eliminates the need for a memory cycle and decreases the amount of time required for processing. However, when a WP register is addressed in a mode other than the direct mode, the efficiency of the WP cache is lost. This occurs because the program will try to access memory before it is able to manipulate the data in the WP register.

The workspace concept is particularly valuable during operations that require a context switch (a change from one program to another, or to a subroutine, as in the case of an interrupt). The processor accomplishes a complete context switch with only the three store cycles and the three fetch cycles required to exchange the contents of the PC, ST, and WP registers, plus a store cycle for each register in the workspace cache into which data has been written during the present context. After the switch, the WP contains the starting address of a new 16-word workspace in memory for the new routine. The new workspace registers 13, 14, and 15, respectively, save the contents from the previous WP, PC, and ST registers. A corresponding savings in time occurs when the original context is restored.

3.1.1 Interrupts

The 990/12 LR minicomputer implements 16 priority-vectored interrupt levels. Those interrupts that the XOP instructions generate are not a part of the priority structure. When hardware recognizes an interrupt (that is, an interrupt pending at a level not masked by the ST mask), a forced BLWP instruction examines the information stored at the trap address for that interrupt level to determine the new values of the WP and PC. These new values represent the address of the interrupt service routine. The interrupt service routine concludes with an RTWP instruction, which exits the routine and returns control of execution to the point of interrupt. The processor continuously compares the interrupt code (ILVE0– through ILEV3–) with the interrupt mask in ST bits 12 through 15. When the level of the pending interrupt is less than or equal to the enabling mask level (higher or equal priority interrupt), the processor recognizes the interrupt and initiates a context switch upon completion of the currently executing instruction. The processor fetches the new context WP and PC from the interrupt vector locations. The new workspace registers 13, 14, and 15, respectively, retain the previous context WP, PC, and ST. The processor then forces the interrupt mask to a value that is one less than the level of the interrupt being serviced, except for the level 0 interrupt that loads 0 into the mask. This allows only interrupts of higher priority to interrupt a service routine. The processor also inhibits interrupts until the first instruction of the service routine executes, thus preserving the program linkage if a higher-priority interrupt should occur. All interrupt requests remain active until the processor in the device service routine recognizes them. The individual service routines must reset the interrupt requests before the routine is complete. If a higher-priority interrupt occurs, a second context switch processes the higher-priority interrupt. When that routine completes, RTWP restores the first service routine parameters to the processor to complete processing of the lower-priority interrupt. To restore the original program parameters, all interrupt subroutines terminate with the RTWP instruction.

Internal interrupts use the three highest priority level interrupts. The highest of these, level 0, is the power restored interrupt. When ac power is restored to the CPU chassis, program execution begins using the PC and WP previously stored at the level 0 trap locations. When the mask field of the ST is set to 0, it disables all interrupts except level 0. The level 0 interrupt cannot be disabled. When the 990/12 LR power supply senses that a loss of ac power is imminent, it generates a level 1 interrupt. The processor then has 12 milliseconds of program time before the system is reset for the level 0, power-loss state. This function is wired to interrupt level 1. The third internal interrupt, wired to interrupt level 2, is a merging of all system errors plus breakpoint and the 12 millisecond clock. These conditions are:

- Arithmetic overflow.
- 12 millisecond test clock.
- Breakpoint address encountered.
- Stack overflow/underflow.
- Write attempt in write-protected memory.
- Execution violation. Indicates an attempt to execute from a mapped memory segment flagged as nonexecutable.
- Mapping error. Indicates an attempt to address memory beyond the limits set in the active map file limit register.
- Memory data error.
- Illegal operation code.
- Privileged instruction fetch with privileged mode off.
- TILINE timeout. Indicates an attempt to address unimplemented TILINE addresses (memory).

Interrupt levels 3 through 15 use the externally requested equipment interrupts. Figure 3-2 illustrates the reserved trap addresses for these levels. The interrupt request from the external devices (or from internal interrupt functions) can be wired to any of the 13 interrupt request lines on the edge connector of the SMI board. The lines form 13 separate wired-OR interrupt buses. Each line on the board has a 1-K ohm pull-up resistor, and each interrupt request signal should be an active (low) signal driven by an open-collector TTL gate. The request signal remains on its respective interrupt bus until software resets it.

The request signal must reset before the interrupt service program executes RTWP, or else the processor will repeat the trap.

3.1.2 TILINE Peripheral Control Space

The TPCS consists of those CPU addresses in the range of >F800 through >FBFE (see the memory map, Figure 3-2). These addresses are modified before presentation to the TILINE address space. Five address bits are appended to the left (most significant bit side) of each address in order to form a 20-bit TILINE word address. That is, addresses >F800 through >FBFE map to addresses >FFC00 through >FFDFE. This particular mapping occurs only when map file zero is invoked (ST bit 8 equals 0) or if mapping is disabled. The TPCS is a range of addresses reserved for the peripheral device controllers.

3.1.3 Input/Output

The 990/12 LR minicomputer uses a versatile, direct command-driven I/O interface designated as the CRU. The CRU provides up to 4096 directly addressable input bits and 4096 directly addressable output bits. It addresses the input and output bits either individually or in fields of from 1 to 16 bits. The processor employs three dedicated I/O pins (CRUBITIN, CRUBITOUT and STORECLK- and 12 bits (CRUBIT4 through CRUBIT15) of the address bus at the interface to the CRU system. The processor instructions that drive the CRU interface can set, reset or test any bit in the CRU array or move data between memory and CRU data fields.

Because of its extremely flexible data format, the CRU interface can effectively use a wide range of control and data transaction operations. These applications divide into two broad categories: those involving a single control bit transfer, and those requiring input or output of several data or status bits.

The processor performs three single-bit CRU functions: test bit (TB), set bit to one (SBO), and set bit to zero (SBZ). To identify the bit to be operated upon, the processor develops a CRU-bit address and places it on the address bus, CRUBIT4 through CRUBIT15.

For the two output operations, SBO and SBZ, the processor generates a STORECLK- pulse that indicates to the CRU device when an operation is an output operation. It then causes bit 7 of the instruction word on the CRUBITOUT line to go either high or low to accomplish the specified operation (bit 7 is a 1 for SBO and a 0 for SBZ). The TB instruction is an input operation that transfers the addressed CRU bit from the CRUBITIN input line to bit 2 (equal bit; refer to Figure 3-3) of the ST.

The processor develops a CRU-bit address for the single-bit operations from the CRU base address contained in workspace register 12 (W12) and the signed displacement contained in bits 8 through 15 of the instruction. The displacement allows two's complement addressing from base minus 128 bits through base plus 127 bits. The base address from W12 is added (in bits 3 through 14) to the signed displacement specified in the instruction (anything above >E00 is a privileged CRU address), and the result is added onto the address bus. Figure 3-5 illustrates the development of a single-bit CRU address.

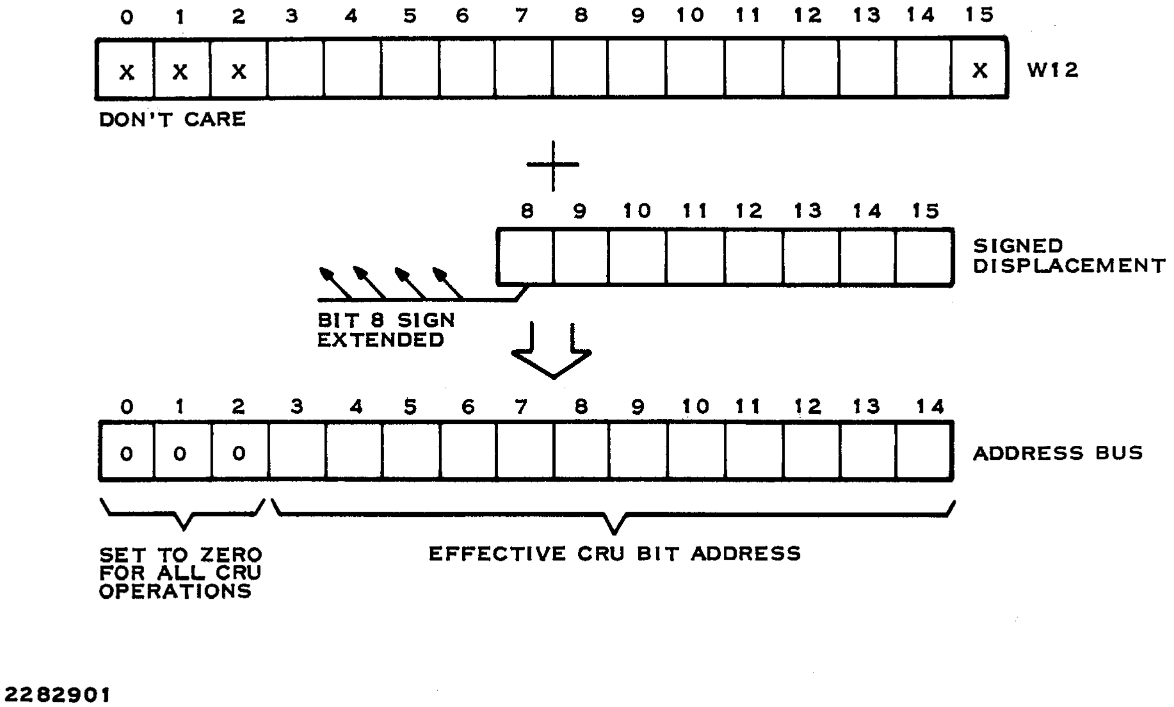
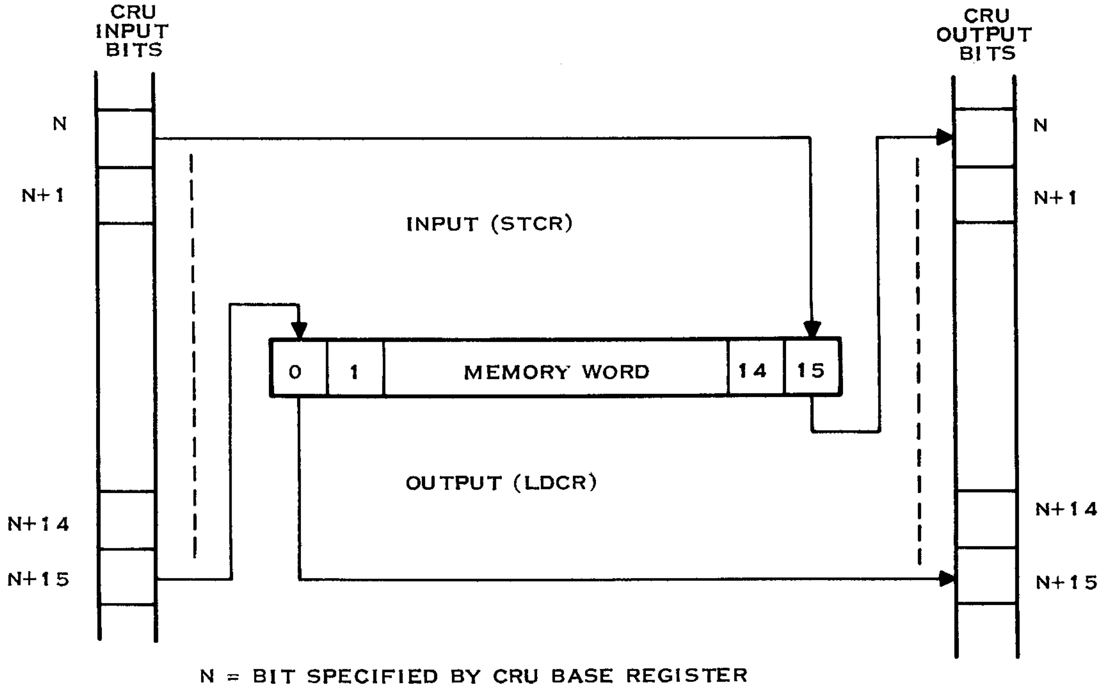


Figure 3-5. Single Bit CRU Address Development

The processor performs two multiple-bit CRU functions: store communications register (STCR) and load communications register (LDCR). Both functions perform a data transfer from CRU-to-memory or from memory-to-CRU (refer to Figure 3-6). Although Figure 3-6 shows a full 16-bit transfer operation, such operations can involve from 1 to 16 bits. The LDCR instructions fetch a word from memory and right-shift it to transfer it serially to CRU output bits. If the LDCR has eight or fewer bits, those bits come from the right-justified field within the addressed byte of the memory word. If the LDCR involves nine or more bits, those bits come from the right-justified field within the whole memory word. As the bits transfer to the CRU interface, the CRU address increments for each successive bit. This addressing mechanism reverses the order of the bit; that is, bit 15 of the memory word (or bit 7) becomes the lowest-addressed bit in the CRU, and bit 0 becomes the highest-addressed bit in the CRU field.

An STCR instruction transfers data from the CRU to memory. If the operation involves a byte or less transfer, the transferred data stores the memory byte right-justified with the leading bits set to zero. When the input from the CRU device completes, the first bit from the CRU is in the least significant bit position in the memory word or byte.



2282902

Figure 3-6. LDCR/STCR Data Transfer

3.1.4 Load Function

There are four ways to perform a load on the 990/12 LR computer system:

1. Power-up, jumper installed
2. Power-up, jumper removed
3. Front panel load switch
4. Alternate load (front panel not present)

When the 990/12 LR is equipped with battery backup it can also have a jumper installed on the AU board at location GJ067. This jumper prevents the system from performing a load function from the program vector at power-up thus preventing loss of any user data. Instead the program traps to the WP and PC of memory locations >0000 and >0002. This is the power restored level zero interrupt vector trap.

The normal mode for the 990/12 LR is the jumper plug removed (no batteries). In this mode, the hardware registers clear at power-up and the program vector loads from the last two locations of the SMI self-test ROM where WP = >FFFC and PC = >FFFE. This location contains the address for the loader, front panel code, and self-test programs. A RESTART- signal permits warm start ROM loaders and front panel routines to be implemented for the processor. When active, RESTART- causes the processor to initiate a trap immediately following the instruction being executed. RESTART- is an unmaskable interrupt (level one interrupt) that traps to the loader and self-test ROM location >FFFC, to obtain the trap vector (WP and PC). The old PC, WP and ST load into the new workspace, and the interrupt mask is set to zero. Then program execution resumes, using the new PC and WP. The RESTART- signal enters the SMI board either from the backpanel or from the front panel connector.

3.1.5 Privileged Instructions

Certain machine instructions of the 990/12 LR execute only when the computer is in the privileged mode (bit 7 of the ST equals 0). Attempting to execute a privileged instruction when the computer is not in the privileged mode causes an error condition and a trap through interrupt level 2 (except for instruction RTWP). The privileged instructions are:

Mnemonic	Instruction
LIMI	Load Interrupt Mask Immediate
DINT	Disable Interrupt
EINT	Enable Interrupts
LIM	Load Interrupt Mask
SLSP	Search List Physical Address
LMF	Load Mapfile
LDS	Long Distance Source
LDD	Long Distance Destination
LREX	Load or Restart Execution
RSET	Reset
CKON	Clock On
CKOF	Clock Off
IDLE	Computer Idle
RTWP	Return from Subroutine

When any interrupt trap occurs, bit 7 of the ST clears to 0 to allow proper interrupt processing. The instruction RTWP executes normally except that during the workspace register 15 to ST transfer, bits 7, 8, and 12 through 15 do not load. Additionally, the addressing of CRU output bits is privileged at an address greater than or equal to >E00 (CRU base address >1C00).

3.1.6 Illegal Operation Codes

A level 2 error interrupt occurs when the processor acquires an instruction that cannot be executed. Table 3-3 lists the illegal instruction operation codes.

Table 3-3. Illegal Instructions

Operation Code in Hexadecimal
0000 through 001B
002C
0100 through 013F
0210 through 021F
0230 through 023F
0250 through 025F
0270 through 027F
0290 through 029F
02B0 through 02BF
02D0 through 02DF
02E1 through 02EF
02F0 through 02FF
0301 through 030F
0310 through 031F
0341 through 034F
0350 through 035F
0361 through 036F
0370 through 037F
0381 through 038F
0390 through 039F
03A1 through 03AF
03B0 through 03BF
03C1 through 03CF
03D0 through 03DF
03E1 through 03EF

3.1.7 Real-Time Clock

A line-frequency synchronized oscillator on the power supply is an input to the central processor. Every cycle of the oscillator generates the real-time clock interrupt function. This function can connect with jumper wires to interrrupt levels 5 or 15 or it can be disconnected. The CKON and CKOF instructions enable and disable, respectively, the real-time clock interrupt function, independent of the ST mask.

3.1.8 Extended Operations

The XOP instruction causes microcode instructions in the WCS to execute if the WCS enable bit in the ST (bit 11) is set. If the WCS enable bit is not set, the XOP instruction traps to the appropriate software subroutine.

3.1.9 TILINE Bus

The 990/12 LR minicomputer uses a high-speed, bidirectional data bus called the TILINE bus, which, along with associated control lines, transfers data among all high-speed system elements. These elements include the CPU, memory, and other rapid data transfer devices such as disk units and magnetic tape transports. The TILINE bus also serves as a computer-to-computer link and is the backbone of multiprocessor systems. The TILINE bus operates asynchronously and the speed of data transfers over it is governed by the distance between and the speed of the devices connected. Consequently, the right configuration of devices can easily tailor the computer to specific applications and upgrade the performance of the system.

The devices connected to the TILINE bus compete for access to it through a positional priority system. High-speed peripherals are usually assigned highest priority and the CPU is assigned lowest priority. In operation, an efficient cycle-stealing action occurs. The overhead time required for switching from CPU access to another device overlaps with the data transfer. This permits a very high rate of device switching without sacrificing overall data bandwith.

3.1.10 Programmer Panel Interface

The 990/12 LR provides an interface to an optional programmer panel through the test connector on the control and display panel. The programmer panel is a CRU device addressed at CRU base address >1FE0. The *990A13 Programmer Panel, Operation and Maintenance Manual* part number 2308789-9701 discusses the programmer panel in detail. The following programmer panel functions are implemented on the SMI board:

- POWER LED — Power reset is inverted and supplied to the programmer panel connector through a 180-ohm resistor.

- FAULT LED — An SBO instruction to CRU bit 11 causes the FAULT indicator on both the programmer panel and the SMI board to light. The fault indicator flip-flop is buffered to drive both lamps. An I/O reset or an SBZ instruction turns off the FAULT indicators. A power reset turns on the indicators. The FAULT signal is supplied to the programmer panel through a 180-ohm resistor.

- RUN LED — An SBO or SBZ instruction to CRU bit 10 causes the RUN indicator on the programmer panel to light. A power reset also lights the RUN indicator. The run function is automatically cleared when a RESTART- signal is generated or the SIE function is set (programmer panel CRU output bit 14). The RUN signal is supplied to the programmer panel through a 180-ohm resistor.

- Memory Error Interrupt Clear — An SBO or SBZ instruction to CRU bit 12 clears a memory error interrupt.

- Single Instruction Execute (SIE) — An SBO or SBZ instruction to CRU bit 14 causes the level one interrupt to be executed after two additional instructions.

- IDLE LED — The IDLE- signal from the processor is buffered and provided to the programmer panel connector through a 390-ohm resistor as the IDLELED- signal.

3.1.11 CRU Interface

Logic for the CRU interface is on the SMI board and exerts control over the interface data and control lines. Except for the module select signals, and the slot location used by the SMI board (slot 1), these lines are available to all main chassis locations. Twenty-four module select signals, which the CRU interface logic decodes, are available to chassis slot locations 2 through 13. Each full slot chassis location accommodates one double-connector circuit board or two single-connector circuit boards. All CRU output operations perform at a 3-megahertz rate. Input operations from CRU base addresses zero through >03FE perform at a 3-megahertz rate, whereas those operations from CRU base addresses >0400 and above perform at a 2-megahertz rate to allow input from an expansion chassis. Table 3-4 lists the specific CRU address ranges reserved for dedicated functions by the 990/12 LR minicomputer system.

Table 3-4. Dedicated CRU Addresses

Hexadecimal Range (CRU Base)	Function
1B00-1BFE	990/12 LR maintenance
1F00-1F3E	CRU interrupt expansion control
1F40-1F5E	TILINE coupler interrupt control
1F60-1F7E	In a newer version of the TILINE coupler, this control has been moved to the TPCS
1F80-1F9E	Breakpoint register
1FA0-1FBE	Error interrupt trace control and map control; also, loader and self-test ROM address control
1FC0-1FDE	System error interrupt, breakpoint interrupt, 12 ms clock interrupt
1FE0-1FFE	Programmer panel (or control and display panel)

3.1.12 Breakpoint System

A CRU-addressable 16-bit breakpoint register, the error interrupt status register and the error trace memory implement the 990/12 LR breakpoint system. The breakpoint register address is CRU base address >1F80. A CRU output instruction loads the desired breakpoint word address into the breakpoint register. CRU output bits 1 through 15 in the breakpoint register correspond to the breakpoint word address, and CRU output bit 0 is the breakpoint enable signal (logic 1 equals breakpoint enabled). When the breakpoint system is enabled and a memory reference is made to the unmapped address that corresponds to the value placed in the breakpoint register, the error interrupt trap (level 2) executes and the breakpoint flag sets the error interrupt status register (bit 6, CRU base address >1FC0). When the breakpoint interrupt occurs, the error interrupt trace memory contains a trace of memory cycles that occurred prior to and including the error condition (as described in paragraph 3.1.14). Hardware clears the breakpoint address and breakpoint enable bits when:

- The breakpoint occurs
- A RSET is generated
- A power-up occurs

The breakpoint register used in diagnostic testing of interrupt levels 3 through 15 disables the breakpoint system and enables the diagnostic interrupt logic.

The breakpoint system is disabled if the processor executes a long distance mapping instruction. When memory mapping is used, the breakpoint system is enabled for either map 0 or map 1. Table 3-6 shows that CRU output bit 13 at CRU base address >1FA0 addresses the processor in the privileged mode and can enable or disable the breakpoint system in this mode. CRU output bits 6 and 7 at CRU base >1FA0 further qualify the breakpoint system. When addressed, these bits (shown in Table 3-6) permit a breakpoint to occur under four conditions:

1. When an instruction fetch is made to the breakpoint address
2. When memory is read from the address
3. When memory is written to the address
4. When the breakpoint is addressed for any condition

3.1.13 12-Millisecond Test Clock

The 12-millisecond test clock resides on the SMI board. When enabled, it generates an interrupt from which a service routine determines the percentage of time used by various system routines and the percentage of time available to users. An SBO instruction addressed to CRU output bit 1 at CRU base address >1FA0 enables the test clock to provide a 12-millisecond test clock interrupt input to the error interrupt status register. When the clock interrupt ocurs, the system error interrupt (level 2) occurs, and the 12-millisecond test clock flag is set in the error interrupt status register (CRU input bit 5 of CRU base address >1FC0). The interrupt is cleared by addressing an SBZ instruction to CRU output bit 5 at CRU base address >1FC0. A power-up or RSET instruction clears and disables the clock interrupt.

3.1.14 Error Interrupt Trace Memory

When an error interrupt or breakpoint occurs, the error interrupt trace memory contains a trace of the 15 memory cycles prior to and including the fetch of the error interrupt trap vector. The trace memory does not stop immediately when the error interrupt occurs but stores one more workspace access or memory cycle to make the sixteenth word. The trace memory contains 16 words of 32 bits each. This 32-bit word contains a 20-bit address field and a 12-bit status code. Table 3-5 describes the function of the bits.

Software reads the error interupt trace memory through a 16-bit data selector/multiplexer that is enabled at CRU base address >1FA0. The first 16-bit word read from the data selector/multiplexer after an error interrupt is the first 16 bits of the last 32-bit trace word. The second 16-bit word read is the second 16 bits of the last trace word. The entire trace memory may be read in this manner. Each read instruction (STCR) must be followed by a set-bit-to-one (SBO) or set-bit-to-zero (SBZ) instruction addressed to CRU output bit 0 at base address >1FA0 to decrement the trace pointer.

Table 3-5. Error Interrupt Trace Memory Data Word Bit Functions

Trace Memory Bit Number	CRU Bit Number	Comment
	Word 1	
0–15	0–15	Saved address; 16 least significant bits
	Word 2	
16–19	0–3	Saved address; 4 most significant bits
20	4	Instruction fetch flag; 1 = Instruction fetch
21	5	Workspace access flag; 1 = Workspace access
22	6	TILINE read/write flag; 1 = Write
23	7	TILINE access flag; 1 = TILINE access
24	8	Workspace read/write flag; 1 = Write
25	9	Privileged instruction attempt when not in privileged mode; 1 = Privilege violation
26	10	An illegal operation code was decoded; 1 = Illegal operation code
27	11	Mapping error; indicates an attempt to address memory beyond the limits set in the active map file limit register; 1 = Error
28	12	Memory data error; 1 = Error
29	13	TILINE timeout; indicates an attempt to address unimplemented TILINE addresses; 1 = Timeout
30	14	Execution violation; indicates memory attempted to execute from a mapped segment that had been flagged as nonexecutable; 1 = Execution violation
31	15	Write violation; indicates an attempt to write to a mapped memory segment that was flagged as nonwritable; 1 = Write violation

3.1.15 Loader and Self-Test Read-Only Memory

Two Texas Instruments TMS 4732 devices that are 32,768-bit, read-only memories (ROMs), implement the loader and self-test ROM on the SMI board. The two devices (each organized as 4K by 8 bits) provide 4K 16-bit words of loader and self-test ROM.

The 990 ROM address space allocation is 512 words, addressed at central processor addresses >FC00 through >FFFE when map 0 is selected (refer to Figure 3-2). To facilitate addressing the 4K words of the ROM in the allocated address space, the ROM has eight 512-word sections. Three address lines that are addressed by CRU output instructions to CRU output bits 10, 11, and 12 at CRU base address >1FA0 (shown in Table 3-6) provide the 512-word section selection. Note that CRU base address >1FA0 also addresses error interrupt trace control and map control. The first section of the ROM is selected on power-up.

The self-test ROM is a fault detection diagnostic with some AU/SMI board fault isolation. The test divides into a test of the microcode and an assembly language self-test. To provide an indication of test failure, the self-test diagnostic addresses one fault lamp on the AU board, two fault lamps on the SMI board, and one fault lamp on the front panel. The front panel fault lamp corresponds to one of the fault lamps (KG033) on the SMI board. The SMI fault lamp KG105 is the one nearest to the front panel connector and the other fault lamp, KG033, is on the opposite side of the board, farthest away from the front panel connector.

The self-test initiates at power-up before control transfers to the power-up trap. The self-test can also initiate by pressing the HALT, RESET and LOAD switches on the optional programmer panel. Table 3-7 provides the meaning of the fault indicators. The table shows that an assembly language code self-test failure results in an error code display on the programmer panel (control and display panel on the 990A13 chassis) data display indicators. The meaning of the error code in hexadecimal is as follows:

- >0100 — Self-test hung
- >0400 — Instruction failure
- >0800 — TILINE failure
- >1000 — Memory failure
- >2000 — Level 2 interrupt failure
- >4000 — Levels 3 through 15 interrupt failure

The error code is also present in programmer panel WP R6. On memory error, the 20-bit TILINE address is stored in programmer panel WP R8 and R10.

Table 3-6. CRU Output Bit Assignments for Error Interrupt Trace Control and Map Control (CRU Base Address >1FA0)

Output Bit	Function
0	SBO or SBZ instruction to bit 0 decrements the display pointer of the error interrupt trace memory and enables the CRU register containing the trace memory for the CRU read.
1	Test clock enable. SBO enables the test clock; SBZ disables the test clock.
2	TILINE cache enable. SBO disables the cache, SBZ enables the cache.
3	Mapping enable. SBO enables memory mapping. SBZ disables mapping except for the addresses to TPCS and to the loader and self-test ROM.
4[1]	SBO or SBZ to bit 4 clears the mapping violation bit in the error interrupt status register.
5	Breakpoint on map 0 or map 1. SBZ = map 0; SBO = map 1.
6, 7 = 0, 0[2]	Breakpoint occurs on any read.
6, 7 = 0, 1[2]	Breakpoint occurs on any instruction fetch.
6, 7 = 1, 0[2]	Breakpoint occurs on any write.
6, 7 = 1, 1[3]	Breakpoint on any address occurrence.
8	Diagnostic interrupt enable. SBO enables the diagnostic interrupt.
9	Diagnostic memory error. SBO forces a TILINE memory error.
10, 11, 12	Bits 10, 11, and 12 develop ROM address lines to address one of eight 512-word sections of the loader and self-test ROM. Bit 10 is the least significant bit, bit 12 is the most significant bit.
13	Breakpoint qualifier in privileged mode. SBO disables the breakpoint system when the processor is in privileged mode.
14, 15	Reserved.

Notes:

[1] Also cleared by SBZ to CRU output bit 11 at CRU base address >1FC0, power-up, or RSET instruction.

[2] Breakpoint bit in the error interrupt status register is set after the memory request that satisfied the breakpoint.

[3] Breakpoint bit in the error interrupt status register is set after the map request.

Table 3-7. 990/12 LR Self-Test Fault Indicators

Condition of System or Lamp					
	Front Panel				
System Lockup	RUN LED	FAULT LED	AU	SMI (KG105)	Meaning
No	On	Off	Off	Off	Microcode and assembly language code self-test passed, loader program entered.
No	On	On	Off	Off	Assembly language code self-test failed, error code on front panel.
Yes	Off	On	Off	Off	Microcode self-test failed, unable to isolate.
Yes	Off	On	On	Off	Microcode self-test failed, AU is the probable cause, the SMI is not tested.
Yes	Off	On	Off	On	Microcode self-test failed, SMI is the probable cause.
Yes	Off	On	On	On	Microcode self-test not initiated, probable cause is a very basic failure.

Note:

* SMI LED, KG033, lights and extinguishes in step with the FAULT LED on the control and display panel or optional programmer panel.

3.1.16 System Error Interrupt (Level 2)

The system error interrupt merges all system errors, plus breakpoint and test clock inputs in the error interrupt status register. This register is implemented by a 12-bit CRU register at address >1FC0. Each error is read and/or cleared by the CRU interface and is also cleared by either an I/O reset or master clear. Table 3-8 shows the CRU input and output bit assignments that address the system error interrupt logic.

Table 3-8. System Error Interrupt CRU Bit Assignments (CRU Base Address >1FC0)

Input Bit	Output Bit[1]	Error Condition
0-3	0-3	Reserved.
4	4	Arithmetic overflow.
5	5	12 ms test clock.
6	6	Breakpoint address encountered.
7	7	Stack overflow/underflow.
8	8	Write attempt in protected memory.
9	9	Execution violation. Indicates an attempt to execute from a mapped memory segment that was flagged as nonexecutable.
10	10	Not used.
11	11[2]	Mapping error. Indicates an attempt to address memory beyond the limits set in the active map file limit register.
12	12[3]	Memory data error.
13	13	Illegal operation.
14	14	Privileged instruction fetch with privileged mode off.
15	15	TILINE timeout. Indicates an attempt to address unimplemented TILINE address (memory).

Notes:

[1] Individually cleared by an SBZ instruction. An SBO instruction to any of bits 0 through 7 of this register sets all bits for diagnostic purposes.

[2] Also cleared by an SBZ instruction to mapping feature bit 4 (CRU base address >1FA0) for 990/10 compatibility.

[3] Also cleared by SBO or SBZ instruction to programmer panel bit 12 (CRU base address >1FE0) for 990/4 compatibility.

In addition to being applied to the error interrupt status register (where they can be read with a CRU input instruction to CRU base address >1FC0), the error status signals are also applied to interrupt logic and to error interrupt trace memory logic. The interrupt logic and error interrupt trace memory logic are discussed in separate paragraphs.

3.1.17 Diagnostic Registers

The diagnostic registers monitor selected signals on the SMI board as well as provide a diagnostic serial data stream input to the AU board that is representative of the monitored signals. The serial data input to the AU board is applied to the processor and the status multiplexer on the AU board. The logic of the diagnostic registers consists of shift/storage registers and a data selector/multiplexer. Each shift/storage register loads parallel with the signals to be monitored by that register; these signals are serially clocked out of the shift/storage register as input to a data selector/multiplexer. Additionally, the TILINE reset and hold signals are applied as inputs to the data selector/multiplexer. Signals monitored by the shift/storage registers include TILINE address signals, mapping base register select signals, the EPROM- address enable, module select signals, front panel signals, CRU signals and SMI board clock signals. The diagnostic registers increase the probability of isolating a fault to a particular board.

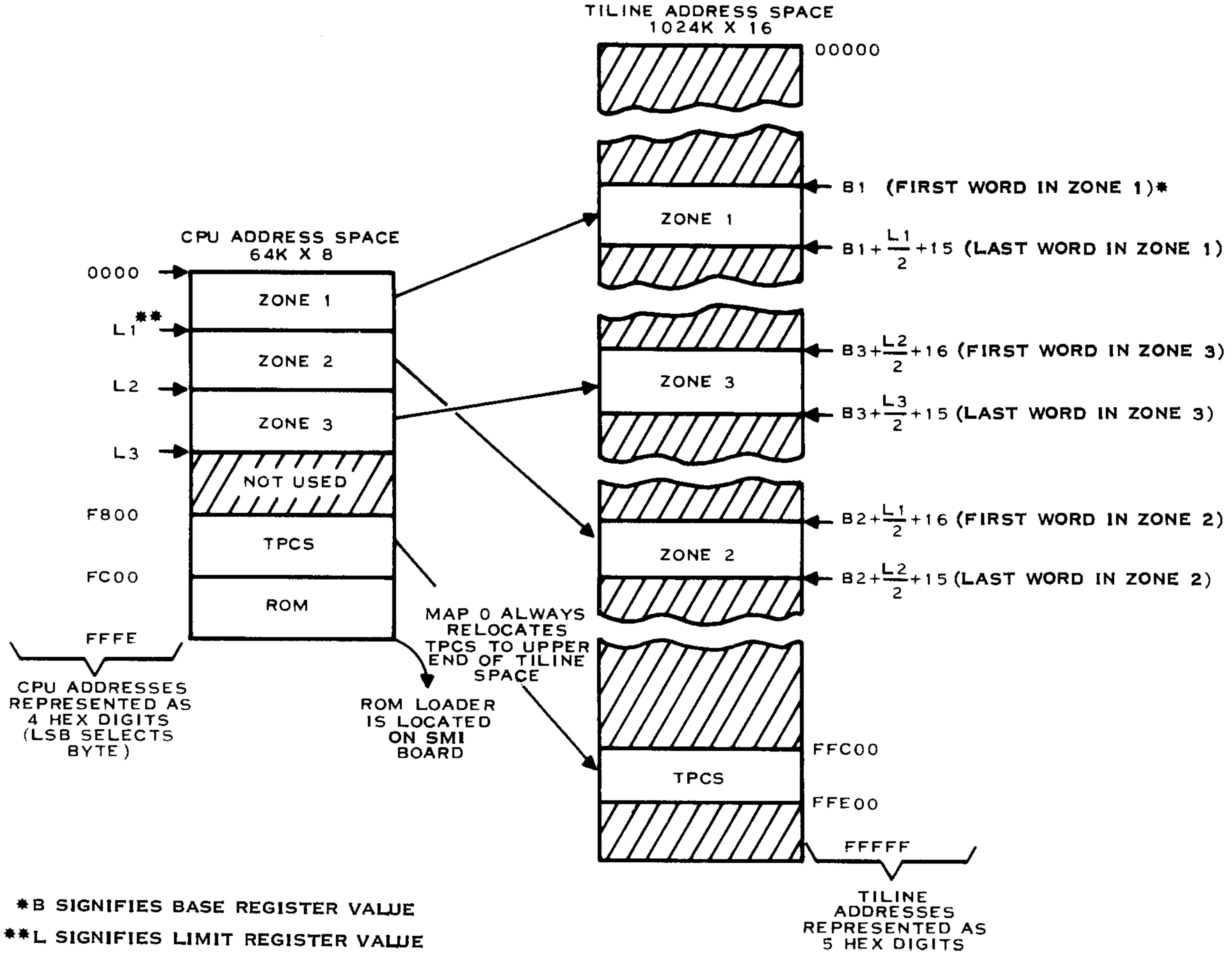
3.1.18 Memory Mapping

Memory mapping expands the inherent addressing capability of the 990/12 LR processor from 64K bytes to 2048K bytes. Figure 3-7 illustrates how memory mapping expands the CPU address space. Basically, memory mapping works by relocating the address generated by the processor within blocks of memory defined by preloaded displacement values. The mapping logic (or map file) consists of four sets of mapping registers (map 0, 1, 2 and 3). The operating system, user programs, and long distance instructions, respectively, use maps 0, 1, and 2. Certain instructions use map 3 as an alternative map file. User code cannot specify map 3 since it is reserved for microcode use only. By using three base registers and three limit registers, each map set can define three displacement values for programs using that map. The total memory available with one set of map registers is 64K bytes. The full 2048K bytes can be accessed by changing the parameters within the map with a Load Map File instruction. In addition to the expanded addressing capability, memory mapping provides the following:

- Error detection logic that generates an interrupt and prevents further memory write operations or workspace cache write operations when a program attempts to access memory above its defined area

- Automatic detection of accesses to TILINE peripheral controllers, and mapping of those accesses to the high-order addresses when operating under map 0 only (the operating system map)

- Memory management and protect logic


TILINE ADDRESS SPACE
1024K X 16
00000
ZONE 1
B1 (FIRST WORD IN ZONE 1)*
B1+L1/2+15 (LAST WORD IN ZONE 1)
CPU ADDRESS SPACE
64K X 8
0000
ZONE 1
L1**
ZONE 2
L2
ZONE 3
L3
NOT USED
F800
TPCS
FC00
ROM
FFFE
CPU ADDRESSES REPRESENTED AS 4 HEX DIGITS (LSB SELECTS BYTE)
ZONE 3
B3+L2/2+16 (FIRST WORD IN ZONE 3)
B3+L3/2+15 (LAST WORD IN ZONE 3)
ZONE 2
B2+L1/2+16 (FIRST WORD IN ZONE 2)
B2+L2/2+15 (LAST WORD IN ZONE 2)
MAP 0 ALWAYS RELOCATES TPCS TO UPPER END OF TILINE SPACE
ROM LOADER IS LOCATED ON SMI BOARD
FFC00
TPCS
FFE00
FFFFF
TILINE ADDRESSES REPRESENTED AS 5 HEX DIGITS
*B SIGNIFIES BASE REGISTER VALUE
**L SIGNIFIES LIMIT REGISTER VALUE

2283711

Figure 3-7. Example of CPU Address Space as Expanded with Memory Mapping

3.1.18.1 Map File Loading. During a load map file operation, the AU board activates an interface signal to the SMI board to indicate that a load operation is in progress. Simultaneously, the AU board places an address on the AU bus. This address indicates the starting address of six contiguous words in memory that contain the map file parameters to be loaded. After passing through the mapping circuits, the address directed to memory fetches the first parameter word. This first parameter word transfers into limit register A on the SMI board. Subsequent memory cycles increment a counter so that incoming data is distributed to registers in the selected map file in the following parameter word order:

1. Limit register A
2. Base register A
3. Limit register B
4. Base register B
5. Limit register C
6. Base register C

Additionally, bits 14 and 15 of parameter words 1, 3, and 5 contain memory protect information that the memory protect registers store and retrieve when the appropriate base register and map file are selected. Because of the sequence used in accessing the mapping parameters and the manner in which the data loads into the map file registers, the data must be formatted correctly in memory before the load operation begins. Figure 3-8 illustrates the proper format for all the parameters. Table 3-9 provides a decode of the meaning of bits 14 and 15. All parameters must be in contiguous memory locations in the order in which they will load. In addition, the values must be left-justified within the memory word, since bits 11, 12, and 13 are ignored when the limit registers and the memory protect register are loaded. Also, due to the method of comparison of the limit registers to the incoming address, the true values of the limit registers is loaded so that L1 < L2 < L3; and L1 takes priority over L2, and L2 takes priority over L3.

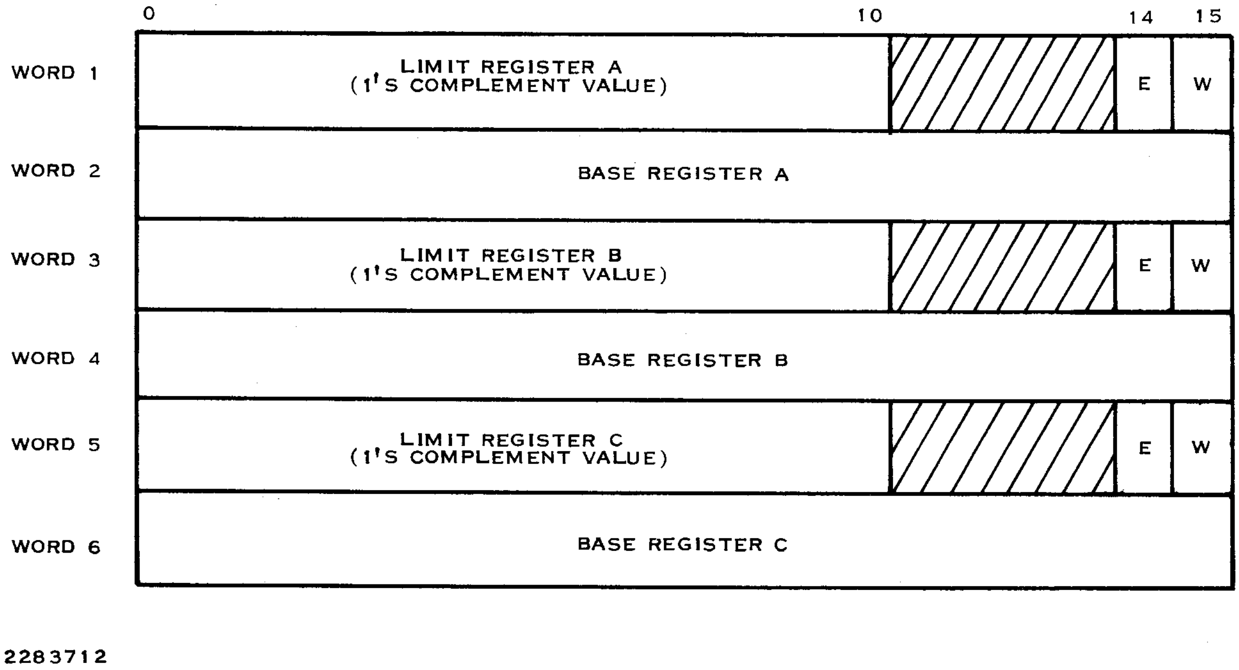


Figure 3-8. Memory Format of Mapping Parameters

Table 3-9. Memory Protect Decode of Limit Register Parameter Words

Mapping Limit Register Bits		Memory Cycles Permitted in Segment
14 (E)	15 (W)	
0	0	Read, Write, Execute
0	1	Read, Execute
1	0	Read, Write
1	1	Read only

3.1.18.2 Memory Management and Protection. Three parameter words address the limit register during a map loading operation. The addressed map file of the memory protect register stores the memory protect information contained in bits 14 and 15 of the three parameter words Signals developed at the output of the memory protect register correspond to the complement of bits 14 and 15, respectively. Table 3-9 lists the limit register's parameter words with the decoded **functions.**

When bit 9 in the status register is set, it invokes memory management and protection for a predetermined memory map address. Invoking memory protection and management provides an alternate means of generating the TILINE write enable signal to prohibit execute and write operations to segments of protected memory. Segments of memory protected from write operations and predetermined from the stored status of bits 14 and 15 of the three parameter words, are loaded into the limit registers during the map loading operations. When a level 2 error interrupt occurs, writes into memory and the workspace cache are inhibited until the interrupts are tested.

TPCS addresses are excluded from memory protection. Attempts to write to or execute in protected memory generate write violation signals. These signals provide inputs generated as level 2 interrupts to the error interrupt trace memory and to the system error interrupt status register.

3.1.18.3 TPCS Mapping. The TPCS is a range of TILINE addresses reserved for assignment to peripheral device controllers. The address range is 512 words. This 512 word range of addresses extends from TILINE word addresses >FFC00 through >FDFFF. The processor maps CPU addresses >F800 through >FBFF to the TPCS (see Figure 3-7). Map file 0 invokes this particular mapping when ST bit 8 equals 0, or when the CPU is in the unmapped mode. The computer uses the TPCS to precondition TILINE peripheral device controllers to perform data transfers. Each controller is assigned a block of 16 addresses from the TPCS; however, a specific controller may not require all 16 addresses. The TPCS addresses allow the computer to access directly a hardware register file within the SLAVE portion of the TILINE peripheral device controller. By writing into this register file, the computer defines a data transfer operation that the MASTER portion of the controller will perform. The actual function of each word of the TPCS depends on the particular controller. Finally, the computer can examine the contents of the controller's register file by performing a read operation with the TPCS addresses assigned to that controller. Figure 3-9 illustrates the TPCS concept.

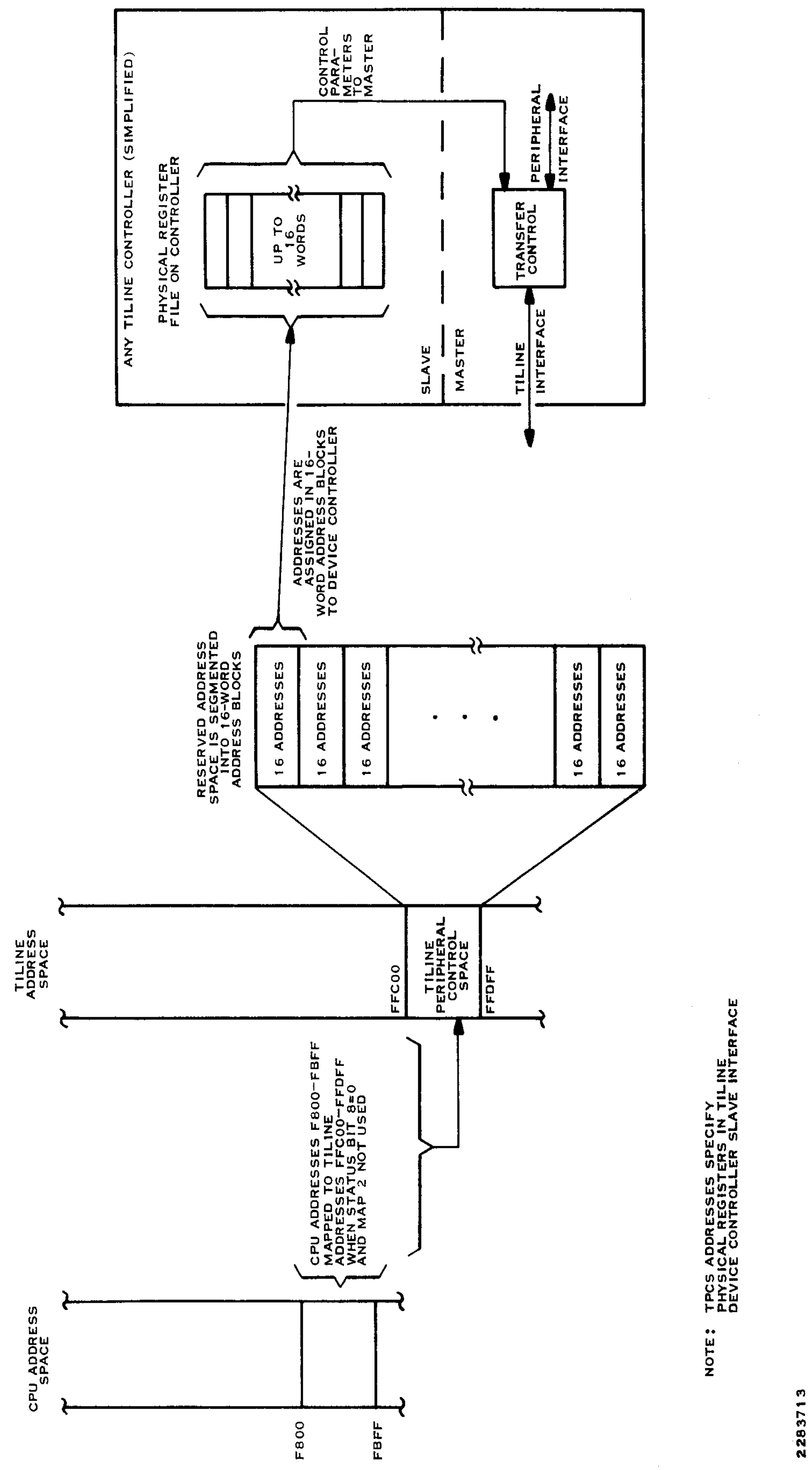

ANY TILINE CONTROLLER (SIMPLIFIED)
PHYSICAL REGISTER FILE ON CONTROLLER
UP TO 16 WORDS
CONTROL PARA-METERS TO MASTER
PERIPHERAL INTERFACE
TRANSFER CONTROL
SLAVE
MASTER
TILINE INTERFACE
ADDRESSES ARE ASSIGNED IN 16-WORD ADDRESS BLOCKS TO DEVICE CONTROLLER
RESERVED ADDRESS SPACE IS SEGMENTED INTO 16-WORD ADDRESS BLOCKS
16 ADDRESSES
16 ADDRESSES
16 ADDRESSES
. . .
16 ADDRESSES
16 ADDRESSES
TILINE ADDRESS SPACE
FFC00
TILINE PERIPHERAL CONTROL SPACE
FFDFF
CPU ADDRESSES F800—FBFF MAPPED TO TILINE ADDRESSES FFC00—FFDFF WHEN STATUS BIT 8=0 AND MAP 2 NOT USED
CPU ADDRESS SPACE
F800
FBFF
NOTE: TPCS ADDRESSES SPECIFY PHYSICAL REGISTERS IN TILINE DEVICE CONTROLLER SLAVE INTERFACE
228371 3

Figure 3-9. TPCS Implementation

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: Model 990/12 LR Computer General Description (2268239-9701)

Manual Date: 15 October 1983 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
____________	____________
	____________
	____________
	____________
____________	____________
	____________
	____________
	____________
____________	____________
	____________
	____________
	____________

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

CUT ALONG LINE