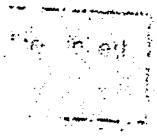
CRAY Y-MP Series
Theory of Operations Manual

CMM-



Cray Research, Inc.

Any shipment to a country outside of the United States requires a letter of assurance from Cray Research, Inc.

Requests for copies of Cray Research, Inc. publications should be directed to:

CRAY RESEARCH, INC.
LOGISTICS
6251 South Prairie View Road
Chippewa Falls, WI 54729

Comments about this publication should be directed to:

CRAY RESEARCH, INC.
HARDWARE PUBLICATIONS
890 Industrial Blvd.
Technical Operations Building
Chippewa Falls, WI 54729

Record of Revision

Each time this manual is revised and reprinted, all changes issued against the previous version are incorporated into the new version and the new version is assigned an alphabetic level which is indicated in the publication number on each page of the manual.

Changes to part of a page are indicated by a change bar in the margin directly opposite the change. A change bar in the footer indicates that most, if not all, of the page is new. If the manual is rewritten, the revision level changes but the manual does not contain change bars.

REVISION	DESCRIPTION
	May 1988 - Original printing

PREFACE

This manual describes the features and internal operations of the CRAY Y-MP computer system manufactured by Cray Research, Inc. (CRI).

This manual describes the operation of the Central Processing Units (CPUs) that execute instructions, provide memory protection, report hardware exceptions, and provide interprocessor communications within the computer systems. Central Memory and I/O channels are also described.

AUDIENCE

This manual is written for Cray field engineers and systems test personnel maintaining the computer and assumes the reader is familiar with digital computers and the basic architecture of the CRAY Y-MP computer system.

ORGANIZATION

This manual is organized into the following tabbed sections. A detailed Table of Contents is included at the beginning of each tabbed section.

SECTION 1 - CRAY Y-MP PHYSICAL DESCRIPTION describes the CRAY Y-MP computer system's mainframe, power supplies, refrigeration condensing unit, peripherals. Also included in this section are the physical characteristics of the memory, and CPU, and the system clock module.

SECTION 2 - CENTRAL MEMORY describes the organization and operation of Central Memory. This section also describes the data, address and control paths, and includes a description of each option used.

SECTION 3 - CPU CONTROL describes instruction issue and control. Also included in this section are the descriptions of instruction buffers, the Program Address register, Next Instruction Parcel (NIP) register, Current Instruction Parcel (CIP) register, and the Lower Instruction Parcel (LIP and LIP-1) registers.

SECTION 4 - CPU DATA REGISTERS describes the Address (A) and Intermediate Address (B) registers, the Scalar (S) and Intermediate Scalar (T) registers, the Vector (V) registers, the Vector Length (VL) register, and the Vector Mask (VM) register. Each subsection includes a description of the options involved and how the registers operate. These subsections are supported by block diagrams.

SECTION 5 - FUNCTIONAL UNITS describes the operation of the Address, Scalar, Vector, and Floating -point functional units.

SECTION 6 - PROGRAMMABLE CLOCK describes the operation of the Programmable Clock.

SECTION 7 - REAL TIME CLOCK describes the operation of the Real-time Clock.

SECTION 8 - INTER-CPU COMMUNICATIONS describes communication and control between CPUs. This section also describes shared registers, semaphore registers, and interprocessor interrupts.

SECTION 9 - CPU INPUT/OUTPUT describes the channels that transfer data between the mainframe and external devices. Separate subsections describe the operation of the three channel types. A fourth subsection describes the hardware that comprises the channels.

SECTION 10 - PERFORMANCE MONITOR describes the Performance Monitor and its Performance Counters (PC)

SECTION 11 - SYSTEM CLOCK MODULE describes the different CRAY Y-MP mainframe clock speeds, clock options, clock fanout, and tuning of the system clock circuitry.

SECTION 12 - SHARED RESOURCES MODULE includes the physical and functional characteristics of the Shared Resources Module.

SECTION 13 - MCA2500ECL MACROCELL ARRAY describes the concepts of the MCA2500ECL Macrocell Array used in the CRAY Y-MP computer system.

APPENDIX A - ENGINEERING DOCUMENTATION explains how to interpret the engineering documentation associated with the CRAY Y-MP computer system.

NOTATIONAL CONVENTIONS

The following conventions are used throughout this manual.

Convention	Description
Lowercase italic	Variable information is denoted by the use of lower case italics.
X or x or *x*	An unused value is denoted by the use of this convention..
n	A specified value is denoted by the use of n.
(value)	The contents of the register or memory location designated by value.
Register bit designators	Register bits are numbered from right to left as powers of 2. Bit 2^0 corresponds to the least significant bit of the register. One exception is the Vector Mask register. The Vector Mask register bits correspond to a word element in a vector register; bit 2^{63} corresponds to element 0 and bit 2^0 corresponds to element 63.

Number base	All numbers used in this manual are decimal, unless otherwise indicated. Octal numbers are indicated with an 8 subscript. Exceptions are register numbers, the instruction parcel in instruction buffers, and instruction forms, which are given in octal without the subscript.

The following are examples of the preceding conventions.

Example	Description
Transmit (A*k*) to S*i*	Transmit the contents of the A register specified by the *k* field to the S register specified by the *i* field.
167*ixk*	Interpreted as machine instruction 167 with the *j* field not used.
Read n words from memory	Interpreted as reading a specified number of words from memory.
Bit 2^{63}	The value represents the most significant bit of an S register or element of a V register.
1000_8	The number base is octal.

RELATED PUBLICATIONS

For additional information on the CRAY Y-MP computer system, refer to the following publications.

- CMM-0405-000 *CRAY Y-MP Power Distribution And Refrigeration Maintenance Manual.* This manual describes operation and maintenance of the power distribution and refrigeration systems for the CRAY Y-MP computer system. The manual describes the microprocessor-based control system, Motor Generator Set (MGS), power supplies, the Heat Exchange Unit (HEU), and the Refrigeration Condensing Unit (RCU-1). A Troubleshooting and Maintenance section is included.

- CQH-0403-000 *CRAY Y-MP/832 Hardware Reference Booklet.* This booklet is a pocket-sized quick reference book. It contains the CAL instruction summary, Exchange Package, off-line diagnostic standard locations, functional unit times, a mainframe functional block diagram, and other information needed by hardware personnel repairing the computer system.

- CSM-0400-000 *CRAY Y-MP System Programmer Reference Manual.* This manual provides a detailed architectural overview for the CRAY Y-MP mainframe from a programmers perspective. It provides information to help system programmers write and optimize program code. The manual contains the following sections: CRAY Y-MP Computer System Overview, Shared

Resources, CPU Control, CPU Computation, Parallel Processing Features, Maintenance Mode, and CPU Instruction Descriptions.

- HR-0080 — *Cray Peripheral Equipment Site Planning Reference Manual.* This manual provides site planning information for operator workstation (OWS) and maintenance workstation (MWS) equipment, Disk Storage units (DSUs), and Front-end Interface (FEI) cabinets.

- HR-0082 — *Cray Support Equipment Site Planning Reference Manual.* This manual provides site planning information for refrigeration condensing units (RCUs) and motor-generator sets (MGSs).

- HR-4000 — *CRAY Y-MP/8 Site Planning Reference Manual.* This manual provides site planning information for the CRAY Y-MP mainframe, the mainframe Heat Exchanger Unit (HEU), the I/O Subsystem (IOS), the SSD solid-state storage device, and the IOS and SSD Power Distribution Units (PDUs).

- HR-4001 — *CRAY Y-MP Computer Systems Functional Description Manual.* This manual describes all components of the CRAY Y-MP computer system. The manual contains the following tabbed sections: System Overview, Mainframe Architecture and CPU Instructions, I/O Subsystem, SSD Solid-state Storage Device, Peripheral Equipment, and Software Overview.

CONTENTS

CONTENTS

FIGURES

TABLES

1 - CRAY Y-MP PHYSICAL DESCRIPTION

CRAY Y-MP computer systems are powerful, general purpose machines that contain eight Central Processing Units (CPUs). The CPUs achieve extremely high multiprocessing rates by efficiently using the scalar and vector capabilities of all CPUs combined with the systems' bipolar memory and shared registers. The CRAY Y-MP mainframe uses a technology based on 2500-gate macrocell array logic. Table 1-1 lists the features of the CRAY Y-MP computer system. Figure 1-1 shows the mainframe with its attached I/O Subsystem (IOS) and an SSD solid-state storage device.

The CRAY Y-MP mainframe is a rectangular-shaped cabinet as shown in Figure 1-1. The dimensions of the mainframe chassis are 79 in. × 32 in. × 76 in. (201 cm × 81 cm × 193 cm). The mainframe chassis weighs approximately 5,000 lbs (2,300 kg).

Figure 1-2 is a top view of the mainframe, showing its mechanical organization. Twelve power supplies are mounted on a vertical plate that rises through the center of the cabinet. Dielectric coolant circulates through the vertical plate and through the power supplies. A panel on the back of the mainframe contains the variacs required to adjust power-supply voltages. Dielectric-coolant hoses at the front of the cabinet distribute coolant to all of the logic chassis modules.

The following subsections describe the CRAY Y-MP mainframe architecture. The subsections describe:

- Mainframe and power supplies
- Refrigeration condensing unit and available peripherals
- Physical and functional characteristics of the memory, CPU, and clock modules
- CRAY Y-MP system configuration

WARNING and CONTROL SYSTEM

The warning and control system console is located in the center of the power-supply end of the mainframe chassis. It contains the fault indicating devices. The warning and control system console monitors voltages, temperatures, pressures, flow rates, moisture, control contacts, motor-generator set (MGS) conditions, and the state of the coolant chiller. The system control console also contains a number of safety indicators that detect abnormal operating conditions.

Table 1-1. CRAY Y-MP Features

Configuration	• Mainframe with eight Central Processing Units (CPUs) • One standard I/O Subsystem, one optional I/O Subsystem • One SSD solid-state storage device
CPU speed Registers	• 6.0 ns clock period (CP) • 32-bit A registers, B registers, Shared B registers; 64-bit S registers,T registers, Shared T registers, and V registers; one-bit Semaphore registers
Functional Units	• 32-bit Address Add, Address Multiply • 64-bit Floating-point Add, Multiply, Reciprocal Approximation • 64-bit Scalar Add, Logical, Shift, Pop/Parity, Leading Zeros • 64-bit Vector Add, Logical, Shift, Pop/Parity and Second Vector Logical
Central Memory	• 32 million 64-bit words plus 8 check bits • 256 banks • 64 k × 1 bit, 15-ns bipolar storage chip • Single-error correction/double-error detection (SECDED) data protection†
Input/Output	• Four very high speed (VHISP) channels, at 1000 Mbyte/s each • Eight high speed (HISP) channels, at 100 Mbyte/s each • Eight low speed (LOSP) channels, at 6 Mbyte/s each
Physical Description	• 17.6 ft^2 (1.64 m^2) floor space for CRAY Y-MP mainframe • 15 ft^2 (1.39 m^2) floor space for each IOS • 15 ft^2 (1.39 m^2) floor space for the SSD • 2.5 tons (2.27 metric tons) CRAY Y-MP mainframe weight • 1.5 tons (1.36 metric tons) for each IOS • Dielectric fluid cooling for mainframe modules and power supplies, using Freon chiller unit heat exchanger • 400-Hz power from the MGSs; PDU inside mainframe cabinet • Three module types: 32 memory, 8 CPU, 1 clock
Maintenance	• The maintenance work station (MWS) is 68020 based, 4-Mbyte memory, 20-slot VME chassis • CPUs have a selectable maintenance mode • Memory tester unit • CPU tester unit, one CPU module and four memory modules • Centralized repair station (off-site) • Shared Resources module

† Hamming, R.W., "Error Detection and Correcting Codes," *Bell System Technical Journal*, 29, No.2, pp. 147-160 (April, 1950).

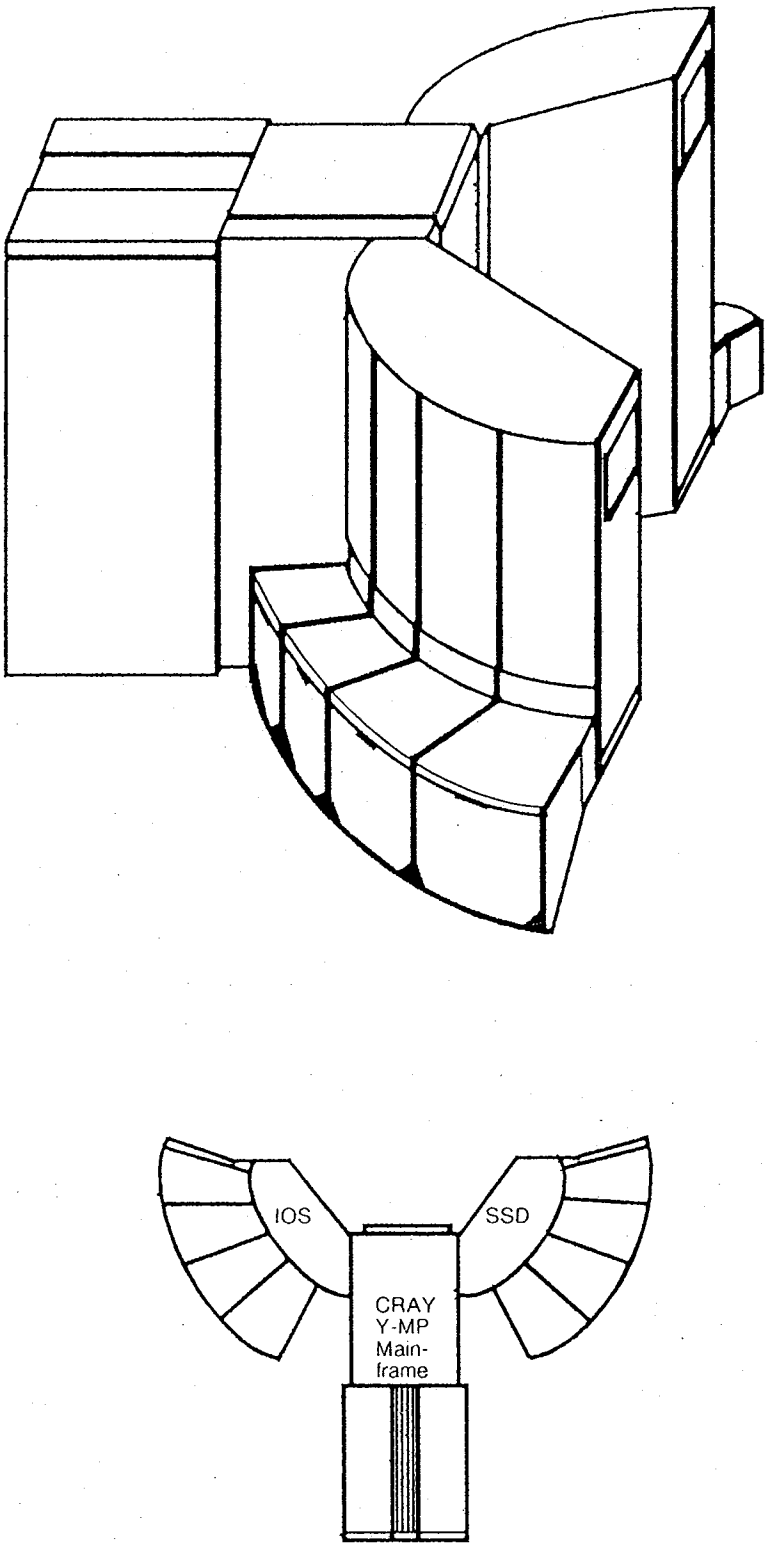


Figure 1-1. CRAY Y-MP Computer System

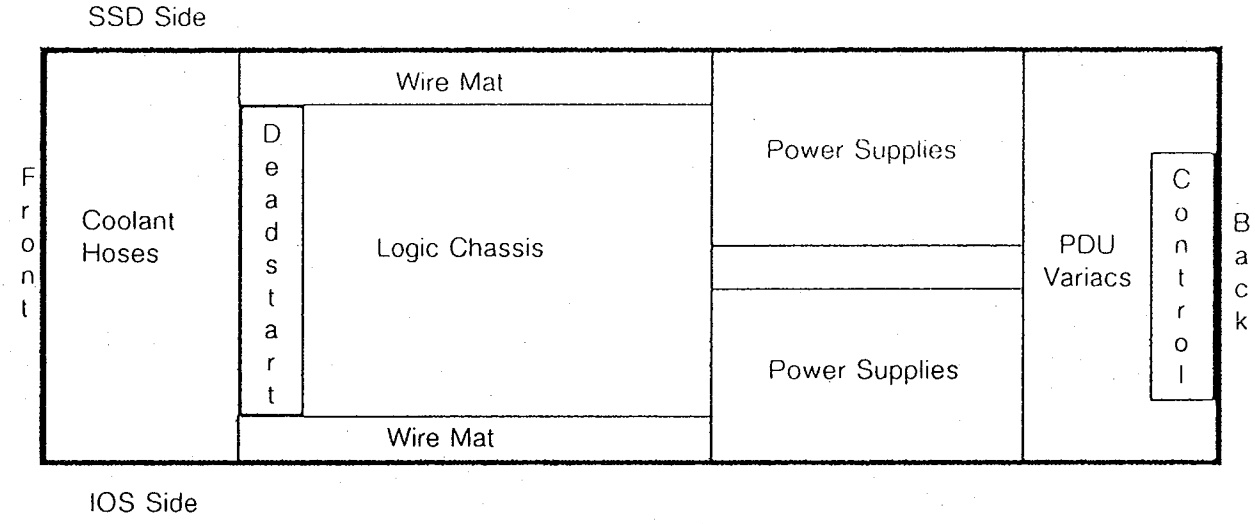


Figure 1-2. Mainframe Mechanical Organization (Top View)

POWER SUPPLIES

The CRAY Y-MP mainframe operates from a 400-Hz, 3-phase power supply. The MGSs transfer 400-Hz power to each of the 12 power supplies. The power supplies provide power to the modules through the bus bars attached to the back of each module. The 12 power supplies are mounted on a vertical plate that rises through the center of the cabinet (refer to Figure 1-3 for power supply to module assignments). Dielectric coolant circulates through the vertical plate and through the power supplies. A panel on the back of the mainframe contains the variacs required to adjust power-supply voltages. The presence of the Power Distribution Unit (PDU) variacs eliminates the need for a separate PDU for the mainframe.

COOLING

The mainframe is cooled by chilled dielectric fluid circulating through each module, each power supply, and the power-supply mounting plate. The refrigeration unit is a freon-based condensing unit that cools the dielectric fluid by means of a Freon-to-dielectric fluid heat exchanger. The Refrigeration Condensing Unit (RCU) is cooled by customer-supplied chilled water. The heat exchanger contains a pump that circulates the dielectric fluid through the mainframe (Figure 1-4 shows the dielectric fluid circulation paths). Each dielectric fluid circulation loop has an adjustable ball valve that controls the flow rate. The power supply module temperature can be raised or lowered by performing evaporator adjustments at the Heat Exchanger Unit (HEU). There is an optimum temperature and flow rate for each circulation loop.

Dielectric-fluid piping and hose connections are made with quick-disconnect couplings. Connections to power supplies and modules are made with self-sealing quick disconnect couplings. The dielectric fluid is sealed off in both sides of the coupling when the coupling is opened. For additional information refer to the CRAY Y-MP Power Distribution and Refrigeration Maintenance Manual, CMM-0405-000.

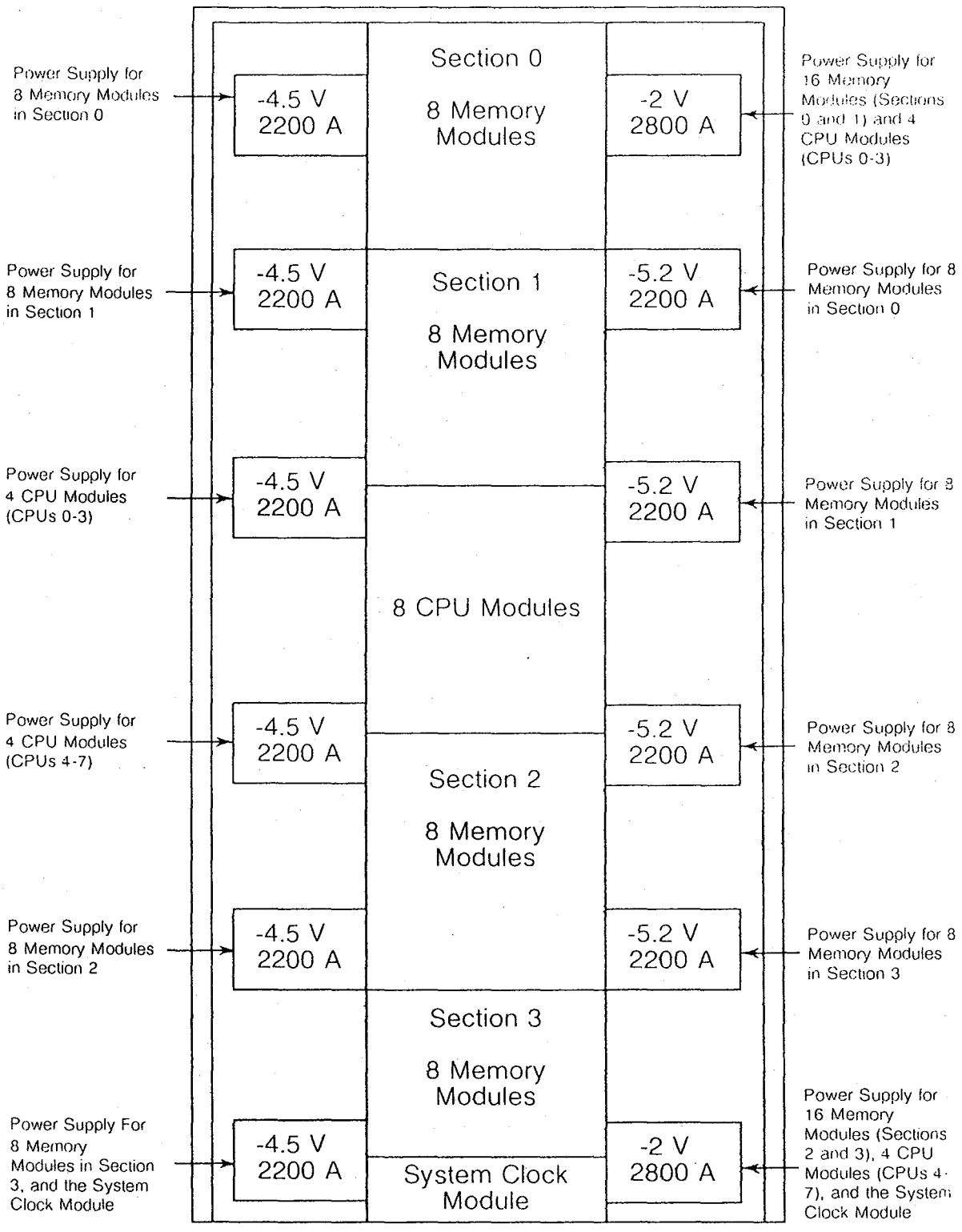


Figure 1-3. Power-supply Configuration (Rear View)

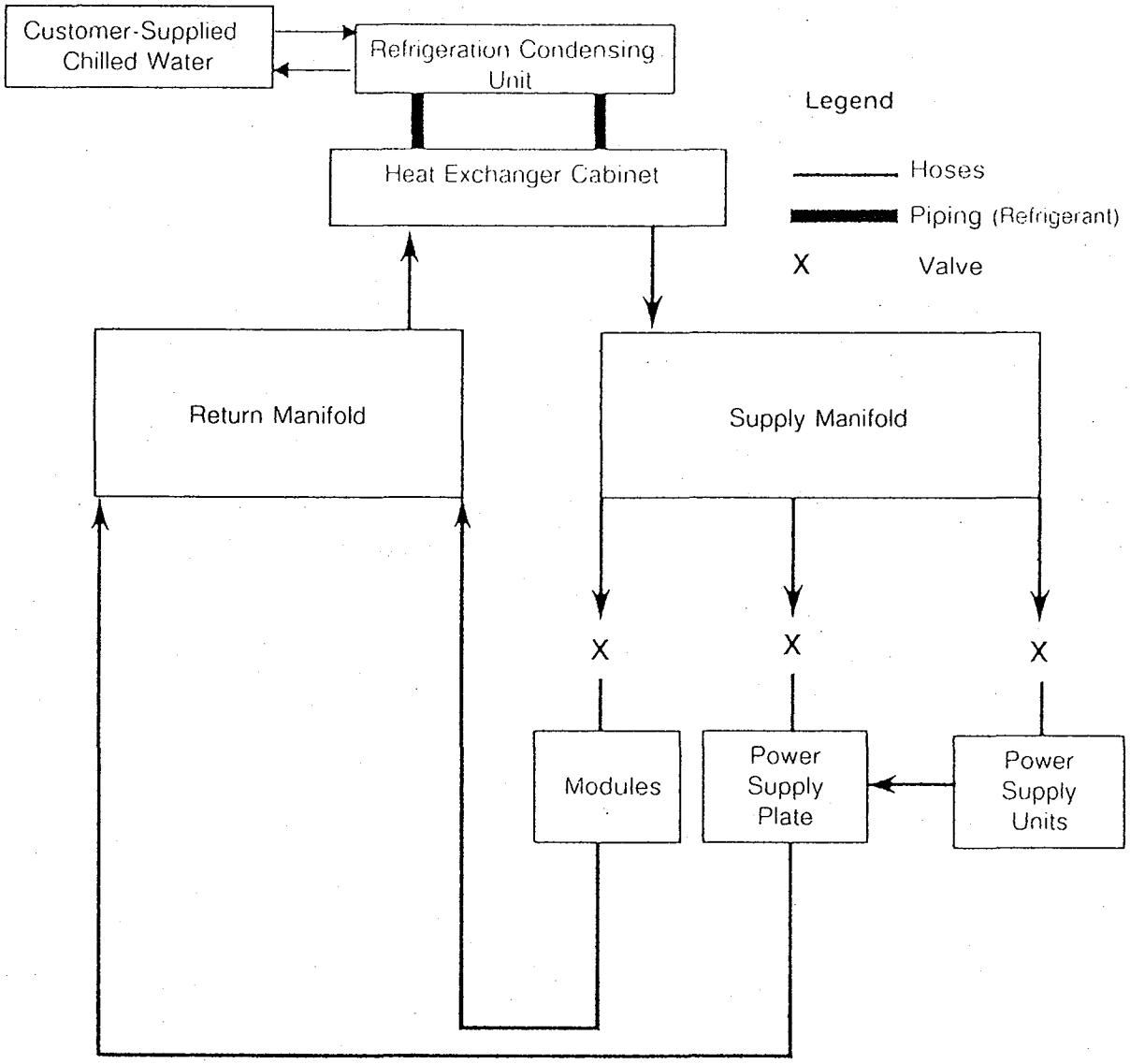


Figure 1-4. Dielectric-fluid Circulation Paths

LOGIC CHASSIS

The logic chassis contains eight CPU modules, 32 memory modules, and the System Clock module (refer to Figure 1-5). Coolant manifolds at the front of the cabinet distribute coolant to all of the logic modules. The coolant connections to each module are located at the front of the module. The DC power bus connections are located at the back of the modules, near the power supplies. The signal connectors, located along the sides of the logic chassis, are special zero-insertion-force connectors that disengage to allow the modules to slide out the front of the chassis.

Slot # 1	Memory Module/Section 0 Data Bits 0 to 8
Slot # 2	Memory Module/Section 0 Data Bits 9 to 17
Slot # 3	Memory Module/Section 0 Data Bits 18 to 26
Slot # 4	Memory Module/Section 0 Data Bits 27 to 35
Slot # 5	Memory Module/Section 0 Data Bits 36 to 44
Slot # 6	Memory Module/Section 0 Data Bits 45 to 53
Slot # 7	Memory Module/Section 0 Data Bits 54 to 62
Slot # 8	Memory Module/Section 0 Data Bit 63, Check Bits 0 to 7
Slot # 9	Memory Module/Section 1 Data Bits 0 to 8
Slot # 10	Memory Module/Section 1 Data Bits 9 to 17
Slot # 11	Memory Module/Section 1 Data Bits 18 to 26
Slot # 12	Memory Module/Section 1 Data Bits 27 to 35
Slot # 13	Memory Module/Section 1 Data Bits 36 to 44
Slot # 14	Memory Module/Section 1 Data Bits 45 to 53
Slot # 15	Memory Module/Section 1 Data Bits 54 to 62
Slot # 16	Memory Module/Section 1 Data Bit 63, Check Bits 0 to 7
Slot # 17	CPU 0
Slot # 18	CPU 1
Slot # 19	CPU 2
Slot # 20	CPU 3
Slot # 21	CPU 4
Slot # 22	CPU 5
Slot # 23	CPU 6
Slot # 24	CPU 7
Slot # 25	Memory Module/Section 2 Data Bits 0 to 8
Slot # 26	Memory Module/Section 2 Data Bits 9 to 17
Slot # 27	Memory Module /Section 2 Data Bits 18 to 26
Slot # 28	Memory Module/Section 2 Data Bits 27 to 35
Slot # 29	Memory Module/Section 2 Data Bits 36 to 44
Slot # 30	Memory Module/Section 2 Data Bits 45 to 53
Slot # 31	Memory Module/Section 2 Data Bits 54 to 62
Slot # 32	Memory Module/Section 2 Data Bit 63, Check Bits 0 to 7
Slot # 33	Memory Module/Section 3 Data Bits 0 to 8
Slot # 34	Memory Module/Section 3 Data Bits 9 to 17
Slot # 35	Memory Module/Section 3 Data Bits 18 to 26
Slot # 36	Memory Module/Section 3 Data Bits 27 to 35
Slot # 37	Memory Module/Section 3 Data Bits 36 to 44
Slot # 38	Memory Module/Section 3 Data Bits 45 to 53
Slot # 39	Memory Module/Section 3 Data Bits 54 to 62
Slot # 40	Memory Module/Section 3 Data Bit 63, Check Bits 0 to 7
Slot # 41	System Clock Module

Figure 1-5. Module Locations within the Logic Chassis

MODULE DESCRIPTION

Coolant flows through the two hollow cold plates contained on each CPU and memory module. The System Clock module has one hollow cold plate. A printed circuit board is

mounted on each side of both cold plates for CPU and memory modules. Figure 1-6 shows the standard module construction.

Each circuit board has 12 layers and measures 11 in × 21.2 in. (28 cm × 54 cm). There is space on the board for up to 78 of the 2500 macrocell array chips. These boards are grouped into pairs and mounted flat on two hollow cold plates. Boards A and B are mounted on the upper cold plate. Board A is mounted on top of the cold plate with its component side up. Board B is mounted on the bottom of the cold plate, component side down. Board A and Board B mounted on a cold plate makes up the upper cold-plate assembly. Boards C and D are mounted to the second cold plate to form the lower cold-plate assembly. The two cold-plate assemblies are then stacked and connected together to form a complete module. Cooling pads on the module, where the board is cut away to allow the chips to directly contact the cold plate, provide more efficient cooling of the chips.

Memory Module

The CRAY Y-MP computer system contains 32 million words of memory divided into 4 sections, 32 subsections, and 256 banks. Within a section, each of the eight modules is responsible for 9 bits of the 72-bit (64 data bits and 8 check bits) data word (refer to Figure 1-5). The memory modules contain all Central Memory data storage chips and a large portion of the data paths, address paths, and control logic needed to transfer data between the storage chips and the eight CPU's. Memory access conflicts are resolved on the memory modules and on the CPU modules. SECDED generation and checking is also done on the CPU module.

The CRAY Y-MP computer system contains 32 identical memory modules. The overall dimensions of the memory module are 23.30 in. × 12.80 in. × 1.39 in. (59.18 cm × 32.51 cm × 3.53 cm). A sample layout of the memory module boards is shown in Figure 1-7, Sheets 1 through 4.

Each memory board has thirty-five 2500-gate macrocell array chips and 288 64 k × 1-bit ECL-compatible RAM chips. The RAM chips are arranged in 36 groups of 8. Figure 1-8 shows the locations of individual memory chips within a group.

There are 20 edge connectors on each board, 8 on the Y side and 12 on the Z side. Each edge connector has 26 pairs of pins for electrical contacts between the module and the chassis wiring. The module also has 57 feed-through connectors. Each of these connectors has up to 48 pins for electrical connections between the four boards on the same module. Each memory board has room for 121 test points that total 484 per memory module.

Power is supplied to the module through a ground connector and three power busses at the rear of the module. The busses supply -2.0 V to the terminator resistors (internal to the circuit boards), -4.5 V to the 2500-gate macrocell array chips, and -5.2 V to the RAM chips. Module cooling is provided by circulating dielectric coolant through hollow channels in the cold plates. The coolant enters and exits the cold plates through quick-disconnect connectors at the front of the module.

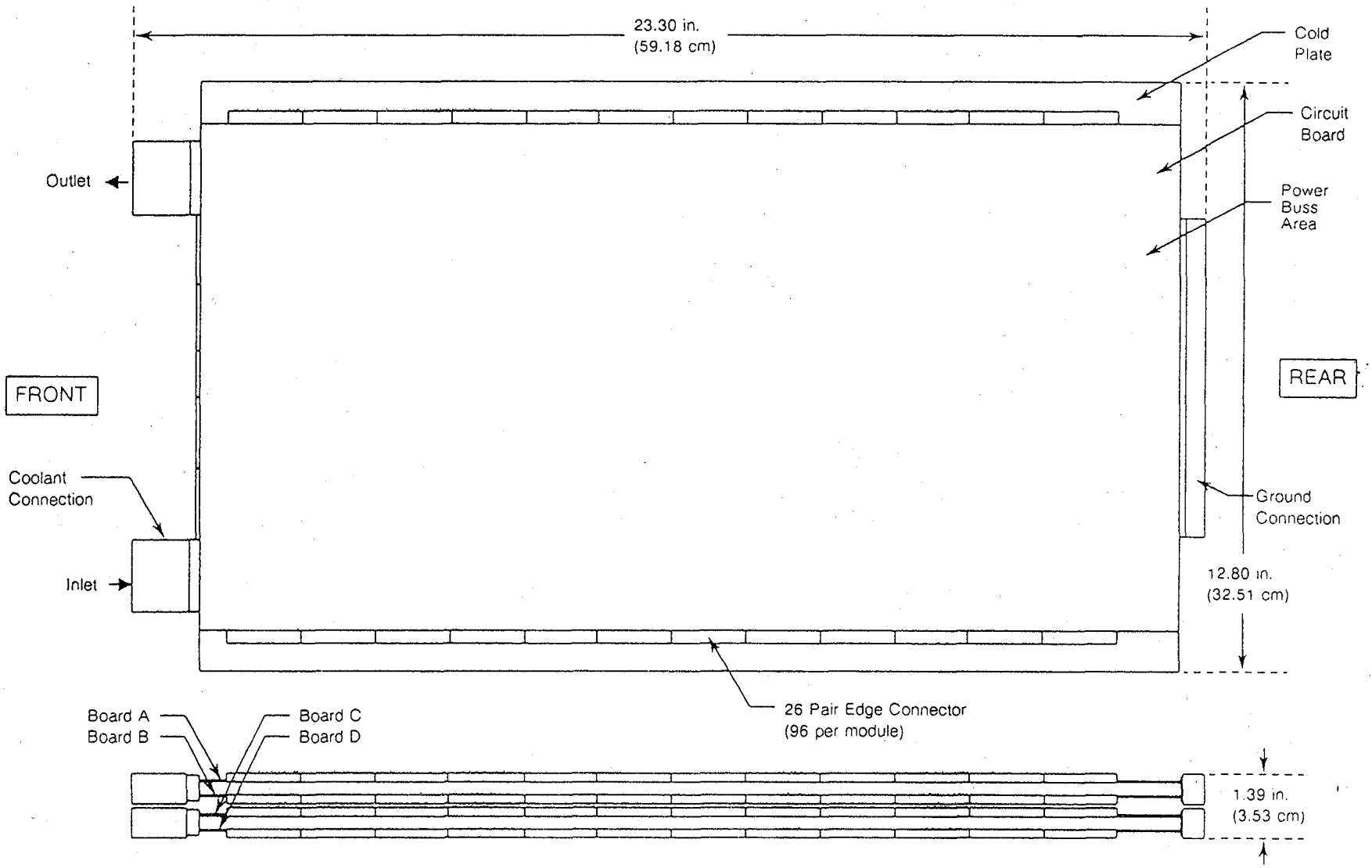


Figure 1-6. Module Construction

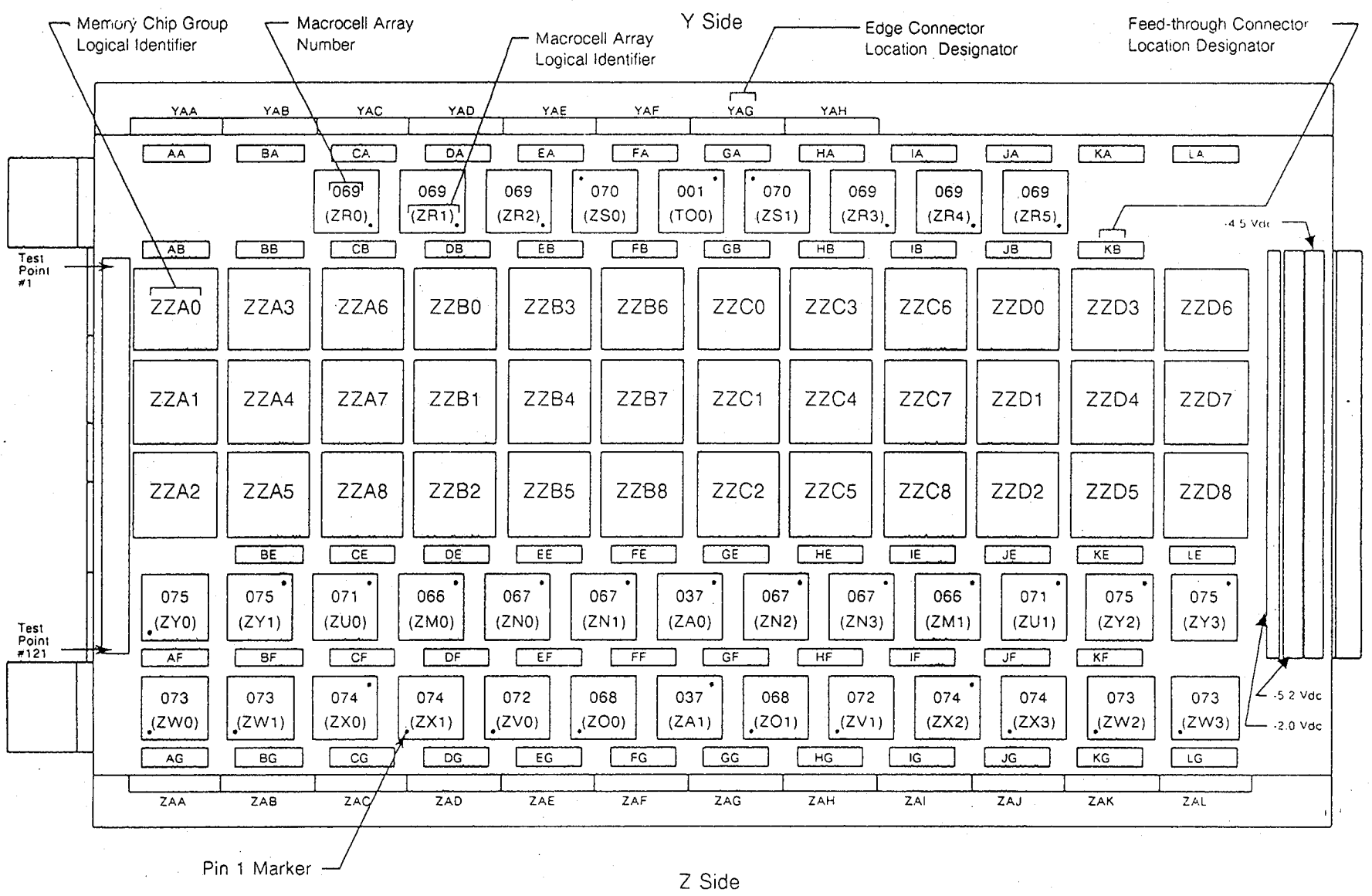


Figure 1-7. Sample of Memory Module Board A Layout (Sheet 1 of 4)

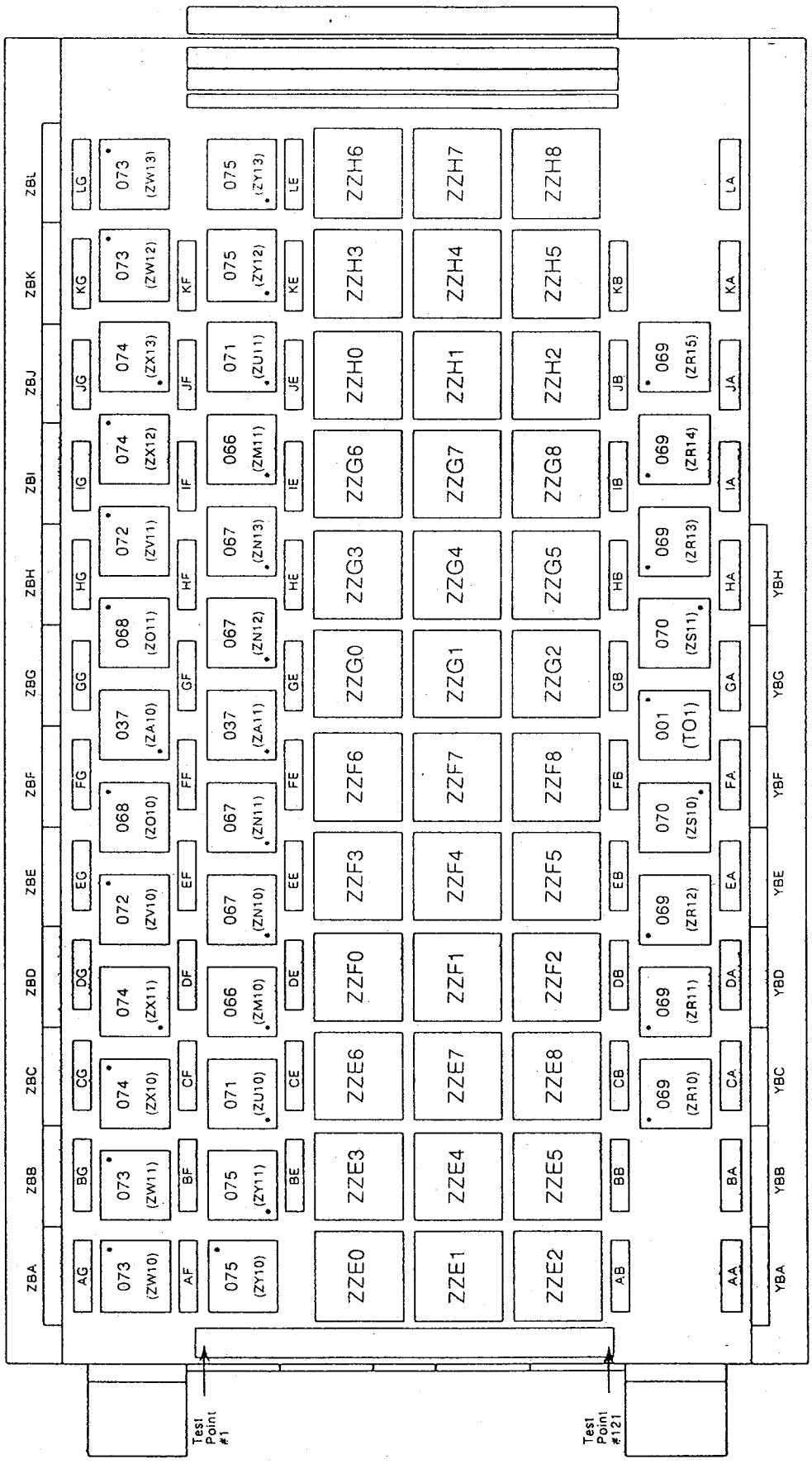


Figure 1-7. Sample of Memory Module Board B Layout (Sheet 2 of 4)

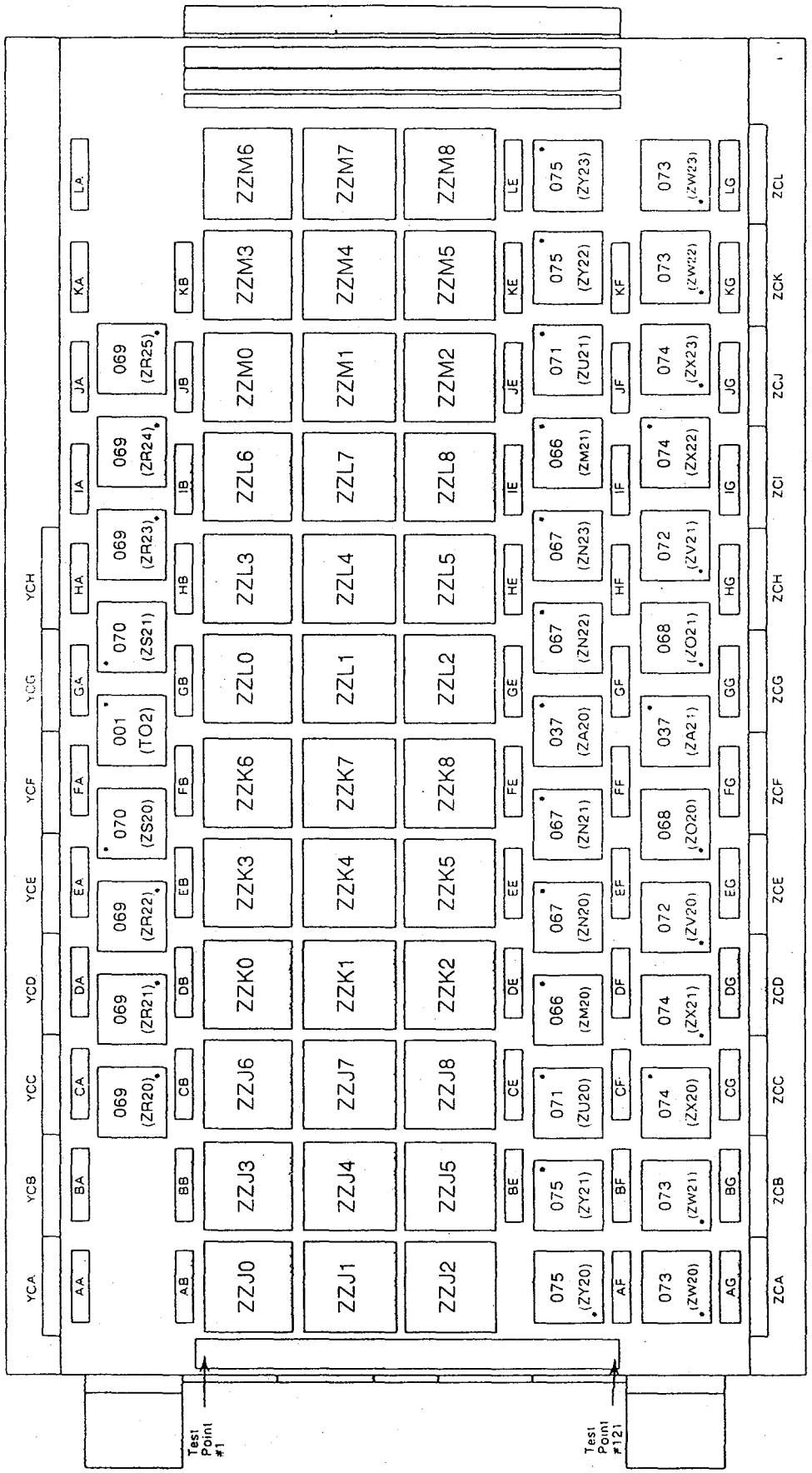


Figure 1-7. Sample of Memory Module Board C Layout (Sheet 3 of 4)

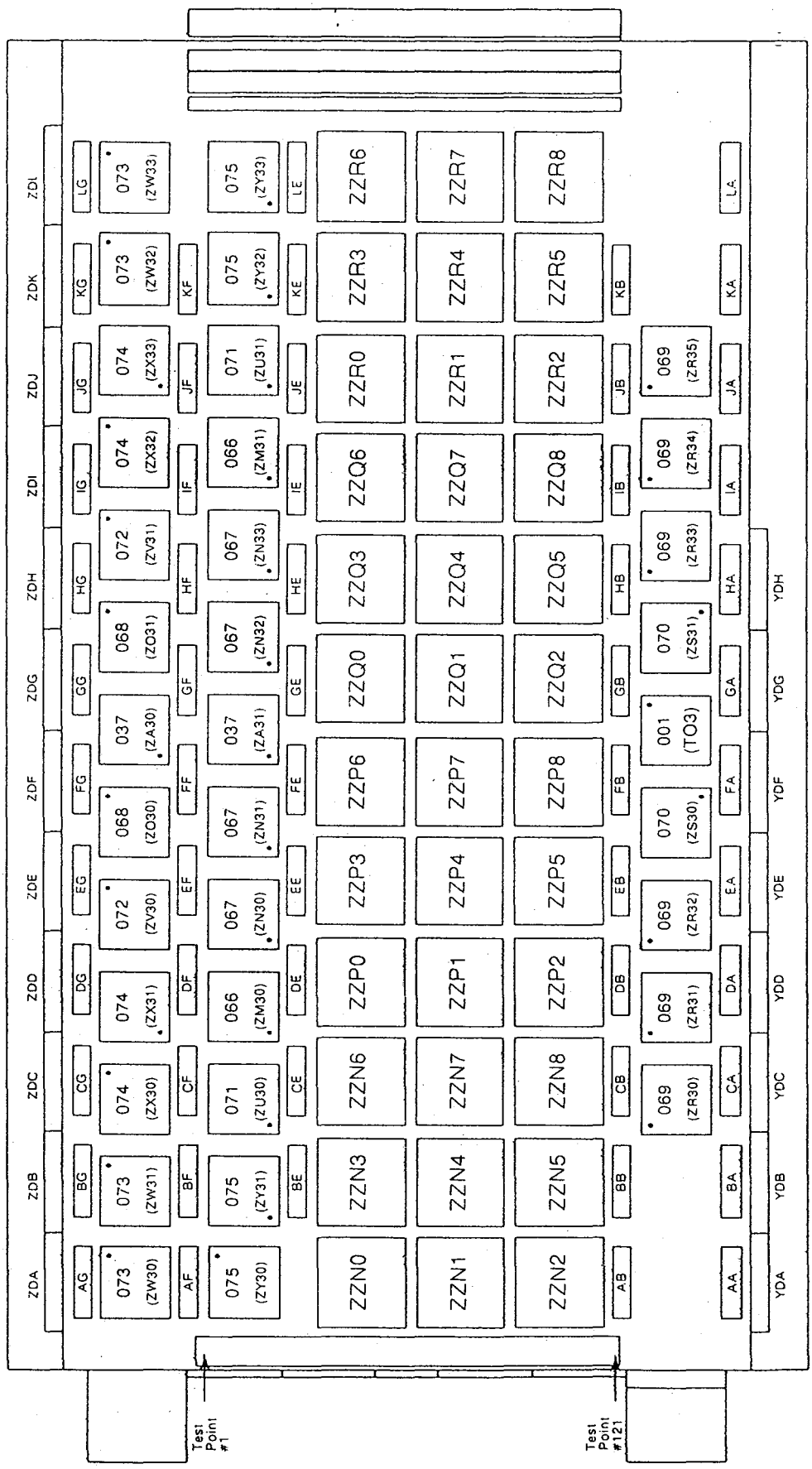


Figure 1-7. Sample of Memory Module Board D Layout (Sheet 4 of 4)

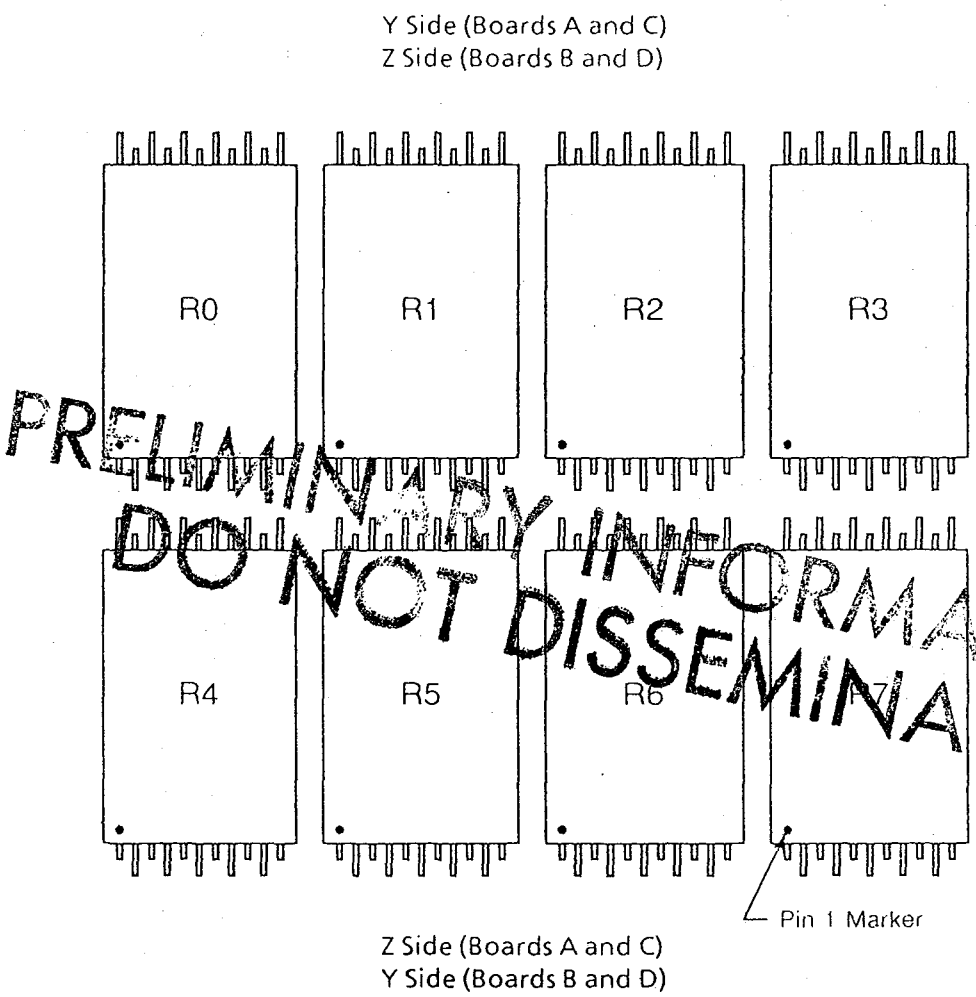


Figure 1-8. Memory Chip Group Layout

CPU Module

Each CPU is contained on one module; all CPU modules are identical (refer to Figure 1-9, Sheets 1 through 4).

All eight CPUs share the mainframe's Central Memory, the inter-CPU communication section, Real-time Clock, and the I/O section.

Each CPU module has four 12-layer printed circuit boards per module. The overall dimensions of the CPU module are 23.30 in. × 12.80 in. × 1.39 in. (59.18 cm × 32.51 cm × 3.53 cm). The CPU modules also consist of Boards A through D mounted on two cold plates.

Each CPU board can contain up to seventy-eight 2500-gate macrocell array chips. There are 24 edge connectors on each board, 12 on the Y side and 12 on the Z side. Each edge

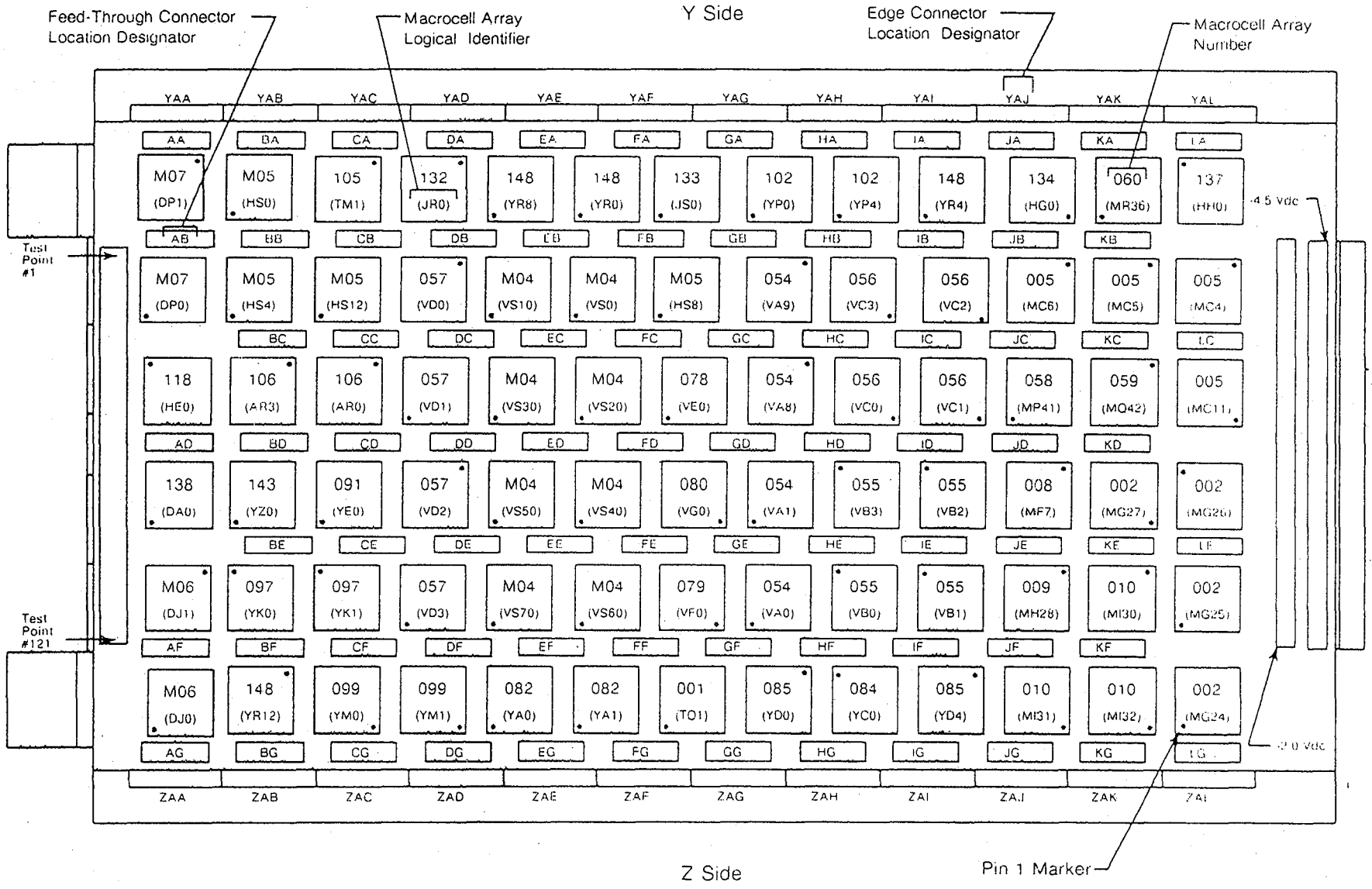


Figure 1-9. Sample of CPU Module Board A Layout Rev 4 (Sheet 1 of 4)

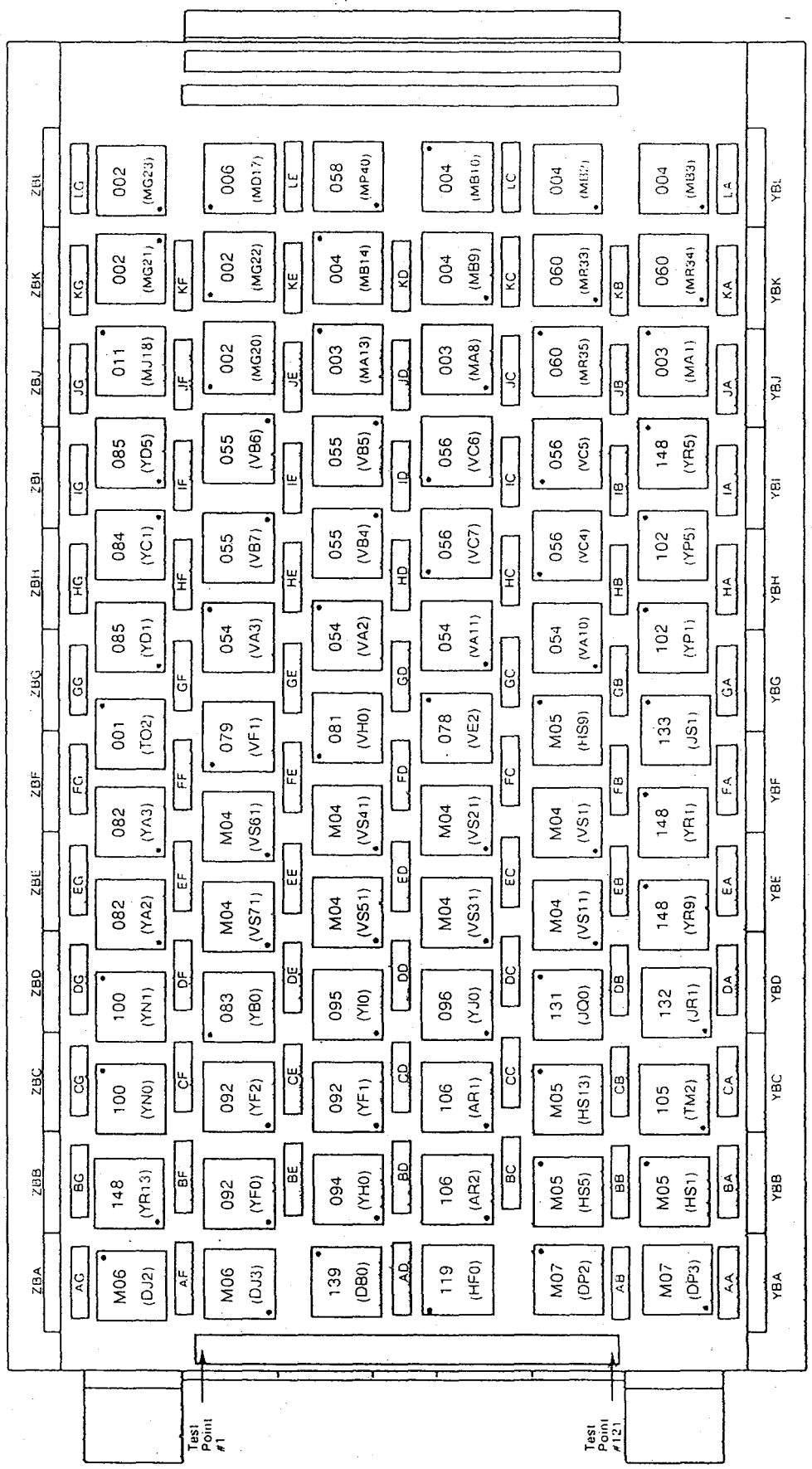


Figure 1-9. Sample of CPU Module Board B Layout Rev 4 (Sheet 2 of 4)

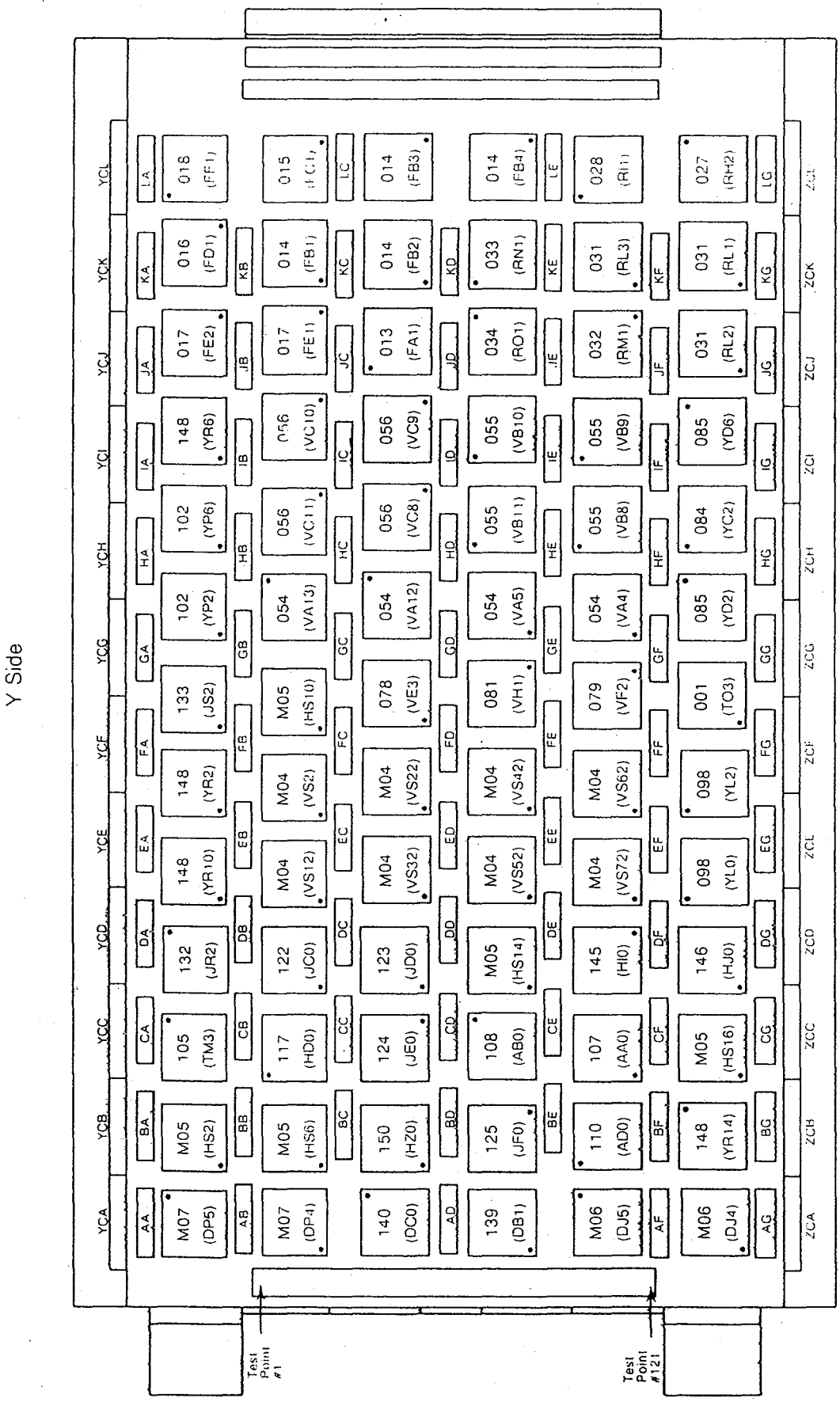


Figure 1-9. Sample of CPU Module Board C Layout Rev 4 (Sheet 3 of 4)

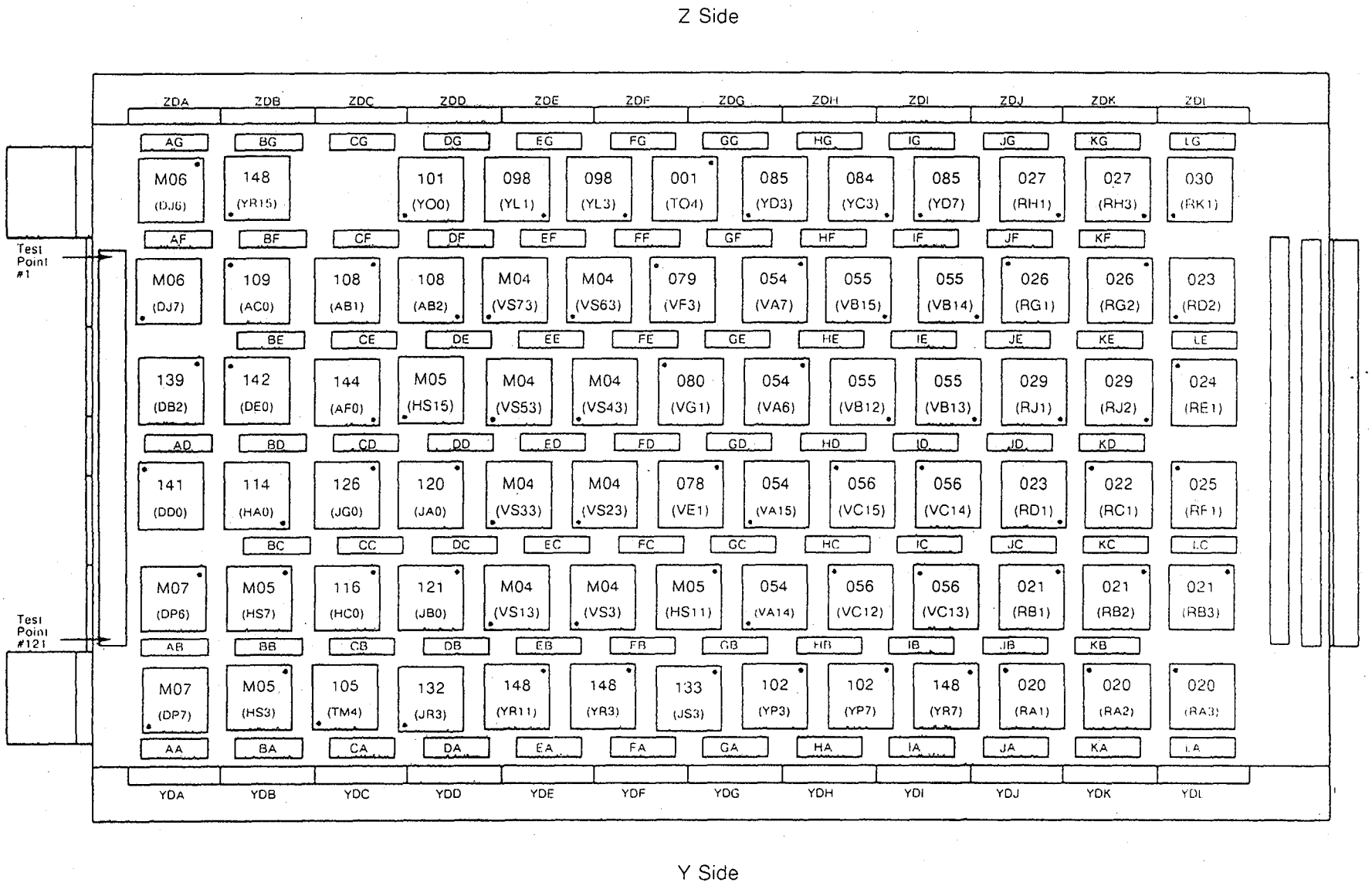


Figure 1-9. Sample of CPU Module Board D Layout Rev 4 (Sheet 4 of 4)

connector has 26 pairs of pins that make electrical contacts between the module and the chassis wiring. The module also has 79 feed-through connectors. Each of these connectors has 48 pins that make electrical connections between boards A through D on a CPU module.

Each CPU board has 121 test points, totaling 484 test points per CPU module.

A ground connector and two power busses at the back of the module supply power to the CPU module. The busses supply -2.0 V to the terminator resistors (internal to the circuit boards) and -4.5 V to the 2500-gate macrocell array chips.

Module cooling is provided by circulating dielectric coolant through hollow channels in the cold plates. The coolant enters and exits the cold plates through quick-disconnect connectors at the front of the module.

System Clock Module

The mainframe is synchronized by the system clock. The System Clock module generates and fans out the 6-ns clock signal for each CPU and memory module (refer to Figure 1-10). The System Clock module can generate crystal-controlled frequencies that are switch selectable from the deadstart panel. There is also a provision for running from an external frequency source that is also selectable from the deadstart panel switches.

The module contains only Board A, which is mounted in a C board slot in the chassis. This module arrangement allows for the placement of higher profile components on the top surface of the module. Because the system clock module contains only a single board mounted on a single cold plate, the components will not cause a clearance problem with the module directly above it. The single system clock module board is mounted on a cold plate and bolted with screws running through the cold plate.

The system clock module has the same physical dimensions as the memory and CPU modules.

Like the memory module and the CPU module, the board used on the System Clock module has 12 layers.

The board contains twelve 2500-gate macrocell array chips and four crystals that control the clock frequencies. The two conhexes located on the front edge of the module are used as auxilliary connectors. One conhex provides a clock from an external signal source and the other conhex provides for reading clock signals out of the module. Layout of the edge connectors on the system clock module is exactly the same as on the the memory and CPU modules. Because the System Clock module has only one board, feed-through connectors are not needed.

The System Clock module has no test points. The clock is test pointed on both the CPU and memory modules.

Power is supplied to the System Clock module through a ground connector and two power busses at the rear of the module. The busses supply -2.0 V to the terminator resistors (internal to the circuit boards) and -4.5 V to the oscillators and 2500-gate macrocell array chips.

System Clock module cooling is provided by circulating dielectric coolant through hollow channels in the cold plate. The coolant enters and exits the cold plate through quick-disconnect connectors at the front of the module.

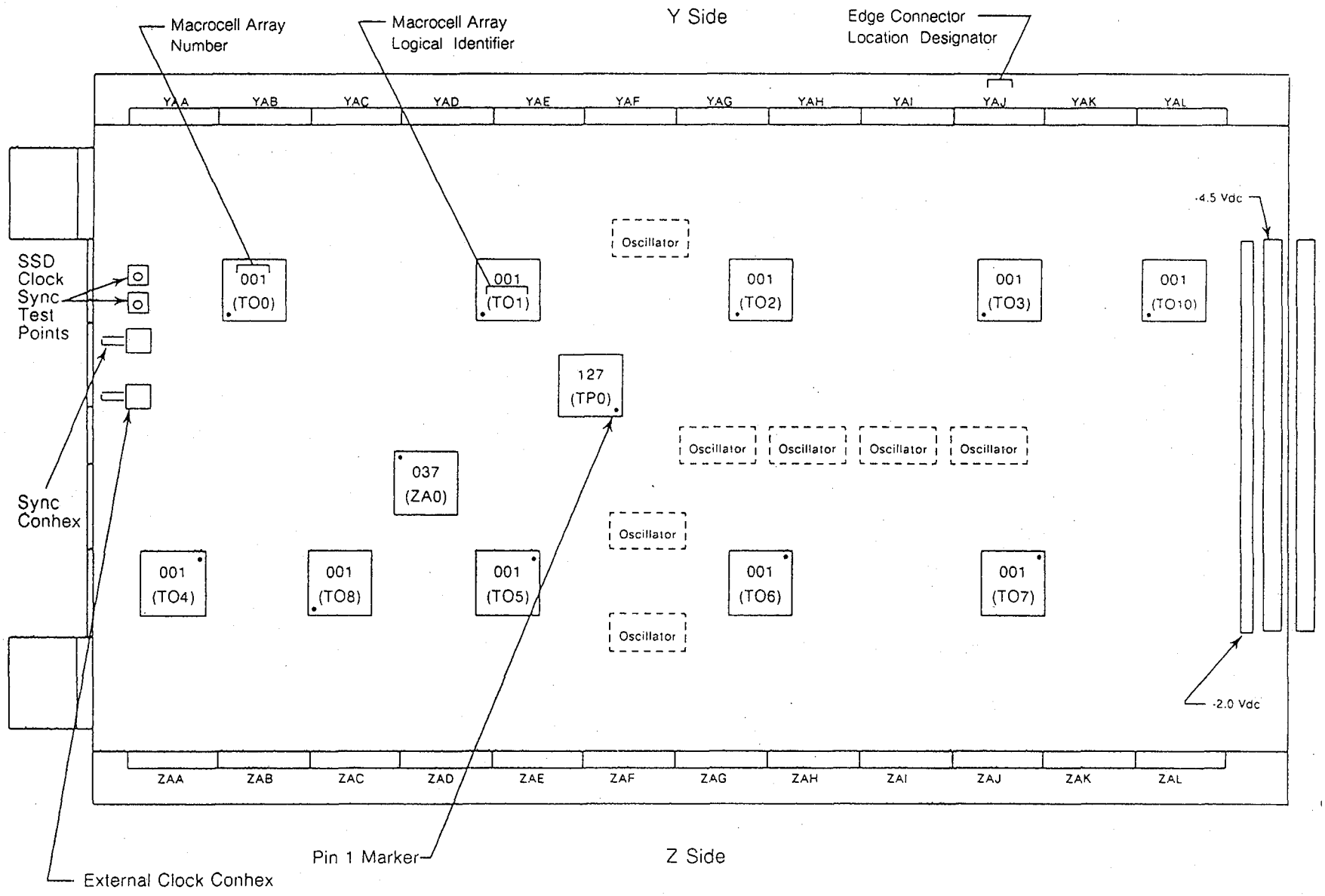


Figure 1-10. Sample of System Clock Module

2 - CENTRAL MEMORY

The CRAY Y-MP computer system contains a large high-speed Central Memory that is shared by all eight central processing units (CPUs) and the I/O section. This section describes the major functions and components of Central Memory. It covers three major topics:

- An overview of Central Memory
- A description of Central Memory functions and components on the CPU Module
- A description of Central Memory functions and components on the Memory Module

OVERVIEW

Central Memory consists of 32 Mwords of solid-state random-access memory (RAM). Each memory word consists of 72 bits–64 data bits and 8 error-correction bits (check bits). Storage for data and check bits is provided by 64k × 1 bit ECL-compatible RAM chips with a 15 ns access time. In order to improve memory access speed, Central Memory is divided into multiple banks that can be active simultaneously. The banks have a 5 clock period (CP) cycle time; each bank can be accessed once every 5 CPs.

In each CPU, the A, B, S, T, and V registers and the instruction buffers have access to Central Memory through memory ports. Each CPU has four ports, so that up to four memory references can occur simultaneously in each CPU. The I/O section does not have its own memory ports, but shares one port in each CPU.

Memory Instructions

Table 2-1 shows all the CPU machine instructions that transfer data between CPU registers and Central Memory or that affect memory operation. Instructions 10h to 13h perform scalar references; each instruction causes only one word to be transferred to or from memory. Instructions 034 to 037 perform block transfers. Each instruction transfers a block (1 to 64 words) to or from consecutive locations in memory. Instructions 176i0k and 1770jk perform stride references. One to 64 words are transferred to or from memory locations that are separated by a constant increment (stride). Instructions 176i1k and 1771jk perform gather and scatter references. They transfer 1 to 64 words to or from randomly-programmable locations in memory.

Instructions 002300 to 002700 affect memory operation. Instructions 002300 and 002400 set and clear the Interrupt on Operand Range Error (IOR) flag in the Exchange Package Mode register. Read "Address Range Checking" in this section for an explanation of the IOR flag. Instructions 002500 and 002600 clear and set the

Table 2-1. Memory Instructions

Instruction		Description	Type of Memory Reference
Machine Code	CAL		
10hijkm (X-mode) 10hi00mn (Y-mode)	Ai exp,Ah	Read from ((Ah) + exp) to Ai exp = ijkm or nm	Scalar
11hijkm (X-mode) 11hi00mn (Y-mode)	exp,Ah Ai	Write (Ai) to ((Ah) + exp) exp = ijkm or nm	Scalar
12hijkm (X-mode) 12hi00mn (Y-mode)	Si exp,Ah	Read from ((Ah) + exp) to Si exp = ijkm or nm	Scalar
13hijkm (X-mode) 13hi00mn (Y-mode)	exp,Ah Si	Write (Si) to ((Ah) + exp) exp = ijkm or nm	Scalar
034ijk	Bjk,Ai ,A0	Read (Ai) words from (A0) to B register jk	Block
035ijk	,A0 Bjk,Ai	Write (Ai) words to (A0) from B register jk	Block
036ijk	Tjk,Ai ,A0	Read (Ai) words from (A0) to T register jk	Block
037ijk	,A0 Tjk,Ai	Write (Ai) words to (A0) from T register jk	Block
176i0k	Vi ,A0,Ak	Read (VL) words from (A0) incremented by (Ak) to Vi	Stride
1770jk	,A0,Ak Vi	Write (VL) words to (A0) incremented by (Ak) from Vi	Stride
176i1k	Vi ,A0,Vk	Read (VL) words from ((A0) + (Vk)) to Vi	Gather
1771jk	,A0,Vk Vj	Write (VL) words to ((A0) + (Vk)) from Vi	Scatter
002300	ERI	Enable operand range error interrupts	None
002400	DRI	Disable operand range error interrupts	None
002500	DBM	Disable bidirectional memory transfers	None
002600	EBM	Enable bidirectional memory transfers	None
002700	CMR	Complete memory references	None

Bidirectional Memory (BM) flag in the Mode register. Instruction 002700 performs no operation, but it holds issue until all previously issued instructions have completed all memory references. Read "Memory Ports" in this section for an explanation of the BM flag and the 002700 instruction.

In addition to memory references that are generated directly by CPU machine instructions, there are three ways that memory references are generated indirectly. A no-coincidence condition in a CPU causes an instruction fetch sequence to begin, which

causes 32 consecutive words to be read from memory to an instruction buffer. An Exchange Sequence in a CPU causes 16 words to be read from and 16 words to be written to Central Memory. For details on the Fetch and Exchange Sequences, read "Instruction Fetch" and "Exchange Mechanism" in Section 3 of this manual. An I/O transfer to or from an external device causes a block of words to be read from or written to memory. Read Section 9 in this manual for details on I/O transfers.

Logical Organization

Figure 2-1 show the major architectural features of Central Memory. Refer to this figure while reading the following paragraphs.

Central Memory is divided into four sections, each section into eight subsections, and each subsection into eight banks. This arrangement permits simultaneous memory references (that is, two or more memory references that begin in the same CP) and overlapping memory references (one or more memory references that begin while another reference is in progress).

Memory Paths

Each CPU has an independent path into each memory section. (The I/O section does not have its own paths, but shares the paths of each CPU.) This allows each CPU to make up to four simultaneous memory references, one reference to each section. Each CPU can have overlapping references in different sections without restrictions, and can have overlapping references within a section as long as each reference uses a different subsection. Because each CPU has only one path into each memory section, simultaneous references to the same section are not permitted.

Simultaneous and overlapping memory references involving two or more CPUs have fewer restrictions than those involving only a single CPU. Simultaneous and overlapping memory references from different CPUs are permitted within a section and within a subsection. The only limitation is that each reference must use a different bank.

Memory Ports

Each CPU has four memory ports which it uses to access its paths to Central Memory. As shown in Table 2-2, each memory port has for specific purposes. Ports A, B, and C are used by memory reference instructions and by the Exchange Sequence. Port D is used by the instruction buffers and the I/O section.

With the exception of memory reads to V registers (176i0k and 176i1k instructions), each type of memory reference uses one specific port. On a read to a V register, Port B is used if it is available. If Port B is reserved, Port A is used if it is available. If both ports are reserved, the instruction holds issue until one of the ports is available. If both ports become available at the same time, Port B is used.

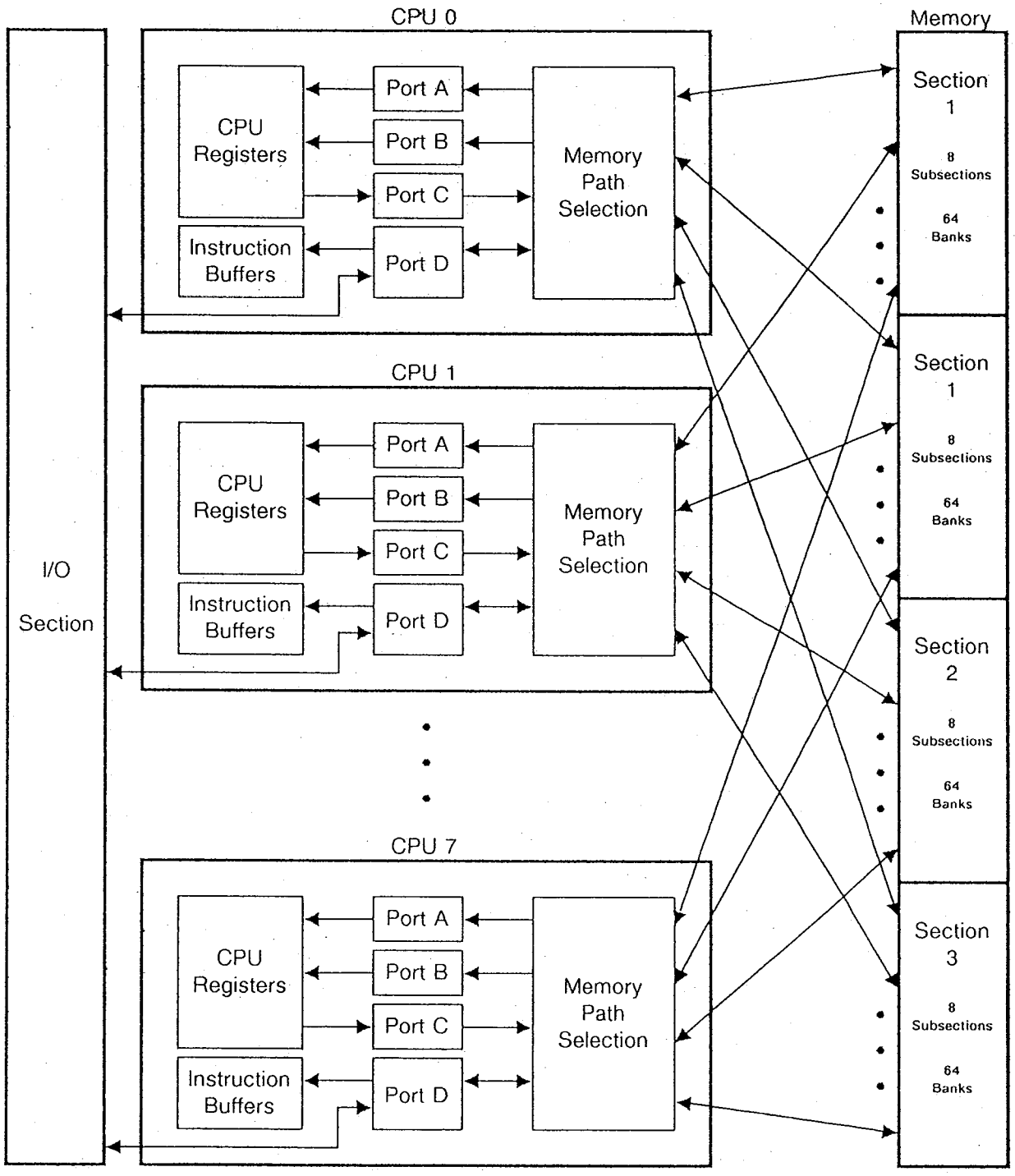


Figure 2-1. Central Memory Architecture

Ports A, B, and C

Ports A, B, and C operate differently for block and vector transfers (B, T, and V registers) than for scalar transfers (A and S registers). A memory reference instruction that transfers data to or from B, T, or V registers holds issue if the associated port is in use by another memory operation. When the port becomes available, the instruction issues and reserves the port. The port remains reserved until the instruction has completed all its memory references. Then the port reservation is cleared, making the port available for other memory operations. Normally, a block or vector transfer can read or write one word of data each CP. However, if the instruction encounters a memory conflict during its execution (read "Memory Conflicts" in this section), it temporarily suspends operation until the conflict is resolved. Therefore, the number of clock periods that the instruction executes and that the port is reserved is unpredictable.

Table 2-2. Memory Ports

Port	Types of References	Used By
A	Read Only	A Registers (10h instruction) B Registers (034 instruction) V Registers (176 instructions) Exchange Data (first eight words)
B	Read Only	~~S Registers (12h instruction)~~ T Registers (036 instruction) V Registers (176 instructions) Exchange Data (last eight words)
C	Write Only	A Registers (11h instruction) B Registers (035 instruction) S Registers (13h instruction) T Registers (037 instruction) V Registers (177 instructions) Exchange Data
D	Read and Write	Instruction Buffers I/O Section

Block and vector transfer instructions that use different ports are normally allowed to operate simultaneously. Under some circumstances this can cause memory references to occur in an unwanted sequence. For example, if an 035ijk instruction (write to memory from a block of T registers) precedes a 176i0k instruction (read from memory to a V register) that uses one or more of the same memory addresses, it is possible that some memory addresses will be read before the data is written to them; both instructions can operate simultaneously, and the read instruction may reference an address before the write instruction.

There are two ways to prevent this problem. An 002700 instruction (complete memory references) inserted between the write and read instructions holds issue until Ports A, B, and C are available. Although this instruction does not perform any operation, it prevents the read instruction from issuing until the write instruction has completed all its memory references and cleared the Port C reservation.

Alternatively, if the BM flag is clear, there is no problem with out-of-sequence references. In this case, instructions that use Port A or B also require Port C to be available, and instructions that use Port C require Ports A and B to be available. The memory read instruction holds issue until the write instruction has completed all its references.

A scalar transfer instruction (A or S register) requires that Ports A, B, and C be available before it can issue. Ports A, B, and C are reserved until the instruction has made its memory reference. This prevents out-of-sequence references involving A and S registers.

Port D

For Port D, an instruction fetch sequence has priority over an I/O transfer. That is, if a fetch request occurs while an I/O transfer is in progress, the I/O transfer is suspended and the fetch begins. When the fetch is completed. the I/O transfer is allowed to continue.

Conflict Resolution

A memory conflict occurs whenever a memory port tries to access a shared part of memory that is in use or whenever two or more ports try to access a shared part of memory at the same time. Intra-CPU conflicts involve ports in the same CPU. Inter-CPU conflicts involve ports in different CPUs. In both cases, conflict resolution logic uses predefined priority schemes to sequence the conflicting memory references and maximize overall machine throughput.

There are four types of memory conflicts: section, subsection, simultaneous bank, and bank busy. The following paragraphs explain each type of conflict and how the conflict is resolved.

A section conflict occurs when two or more ports in the same CPU attempt to access the same memory section simultaneously. It occurs because there is only one path from each CPU to each memory section. The port with the highest priority is allowed to begin its reference. All other conflicting ports hold reference for 1 CP. The following rules determine relative priorities between conflicting ports:

1. Port D has priority over Ports A, B, and C when it is being used for an instruction fetch sequence.

2. Among Ports A, B, and C, any port that has an odd memory address increment has priority over every port that has an even increment. The following rules determine the type of increment (even or odd) for each port:

a. A port being used by a block reference instruction has an address increment of one, which is odd.

b. A port being used by a stride reference instruction can have any constant increment (even or odd).

c. A port being used by a gather or scatter instruction can have an increment that changes after each reference. For the porpose of conflict resolution, a gather or scatter instruction is always considered to have an odd increment.

3. Among Ports A, B, and C with the same type of memory increment, priority is determined by the relative time of instruction issue. The port being used by the instruction first issued has the highest priority.

4. Port D normally has a lower priority than Ports A, B, and C when it is being used for an I/O transfer. However, if a Port D memory reference has been forced to hold for 32 CPs, Port D is temporarily given top priority so that one memory reference can proceed. After the reference has begun, Port D returns to its low-priority status.

Subsection conflicts occur because each memory reference by a CPU makes an entire memory subsection unvailable to all ports in the same CPU for 5 CPs. A subsection conflict occurs if any port in the same CPU attempts to make a reference to the same subsection during this interval. The new refererence holds until the old reference no longer needs the subsection–1 to 4 CPs. Subsection conflicts usually involve two or more ports, but may involve two references from the same port.

If two or more references are holding because of the same subsection conflict, a section conflict occurs immediately following the resolution of the subsection conflict. Another subsection conflict will occur 1 CP after the section conflict. For example, if Port A is using a subsection and Ports B and C attempt to use the same subsection while it is busy, Ports B and C will hold due to the subsection conflict. When the reference from Port A no longer needs the subsection, the subsection conflicts will disappear. Ports B and C will be involved in a section conflict, which will be resolved according to the priority rules listed above. The port with the higher priority will make its reference, and the one with the lower priority will encounterr a subsection conflict.

A simultaneous bank conflict occurs when two or more ports in different CPUs attempt to access the same memory bank at the same time. The CPU with the highest priority is allowed to make its reference. All other CPUs attempting to access the same bank hold reference for 1 CP. Relative priorities between CPUs are determined by memory section and subsection number. Table 2-3 shows that each CPU is given a fixed priority in each section and subsection. Following a simultaneous bank conflict, each CPU port that was forced to hold reference encounters a bank busy conflict.

Bank conflicts occur because each memory reference by a CPU makes the referenced memory bank unavailable to all ports in all other CPUs for 5 CPs. A bank busy conflict occurs if any port in a different CPU attempts to make a reference to the same bank during this interval. The new reference holds until the old reference no longer needs the bank–1 to 4 CPs. If two or more ports are holding because of the same bank busy conflict, a simultaneous bank conflict will occur immediately following resolution of the bank busy conflict.

Table 2-3. CPU Priorities

CPU	Priority							
	Section 0		Section 1		Section 2		Section 3	
	Subsections 0, 1, 4, 5	Subsections 2, 3, 6, 7	Subsections 0, 1, 4, 5	Subsections 2, 3, 6, 7	Subsections 0, 1, 4, 5	Subsections 2, 3, 6, 7	Subsections 0, 1, 4, 5	Subsections 2, 3, 6, 7
CPU 0	A (highest priority)	H (lowest priority)	D	E	C	F	B	G
CPU 1	B	G	A	H	D	E	C	F
CPU 2	C	F	B	G	A	H	D	E
CPU 3	D	E	C	F	B	G	A	H
CPU 4	H	A	E	D	F	C	G	B
CPU 5	G	B	H	A	E	D	F	C
CPU 6	F	C	G	B	H	A	E	D
CPU 7	E	D	F	C	G	B	H	A

The four types of memory conflicts are summarized in Table 2-4.

Memory Addressing

Twenty five address bits, numbered 2^0 to 2^{24}, are used to address 32 Mwords of memory. Figure 2-2 shows the function of each address bit. Bits 2^0 and 2^1 select one of the four memory sections. Bits 2^2 to 2^4 select one of the eight subsections within the section. Bits 2^5 to 2^7 select one of the eight banks within the subsection. In each of these fields, the highest-numbered bit is most significant. With this arrangement, if the memory address advances sequentially, all four memory sections are stepped through in turn. Next all 32 subsections are stepped through. Finally all 256 banks are stepped through.

Address bits 2^8 to 2^{24} select one address within a section, subsection, and bank. Bit 2^{19} selects one of two sets of chips in the bank. Bits 2^8 to 2^{18} and 2^{20} to 2^{24} determine the internal address within each chip.

Memory sections are numbered 0 to 3, according to address bits 2^0 and 2^1. There are two methods for numbering the 32 subsections. Using the absolute method, address bits 2^0 to 2^4 are viewed as the subsection number. Using the relative method, bits 2^0 and 2^1 comprise the section number and bits 2^2 to 2^4 comprise the subsection number. For example, assume address bits 2^4 to 2^0 are 10101_2. Using the relative method , the

Table 2-4. Memory Conflicts

Conflict	Type	Duration	How Resolved	Remarks
Section	Intra-CPU	1 CP	Highest priority port makes reference. Other ports hold reference.	Followed by a subsection conflict if references are to the same subsection.
Subsection	Intra-CPU	1 to 4 CPs	All memory references hold until reference in progress is complete.	Followed by a section conflict if two or more references are forced to hold.
Simultaneous Bank	Inter-CPU	1 CP	Highest-priority CPU makes reference. Other CPUs hold reference.	Always followed by a bank busy conflict.
Bank Busy	Inter-CPU	1 to 4 CPs	All memory references hold until reference in progress is complete.	Followed by a simultaneous bank conflict if two or more references are forced to hold.

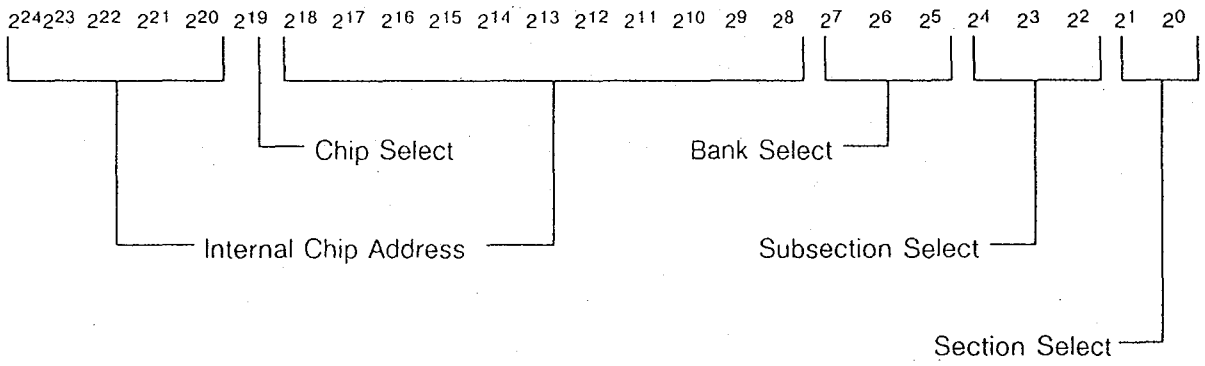


Figure 2-2. Memory Addressing

address is in Section 1, Subsection 5. Using the absolute method, the address is in Subsection 25_8 (Subsection 21_{10}).

Similarly, there are two methods for numbering the 256 banks. Using the absolute method, address bits 2^0 to 2^7 are viewed as the bank number. Using the relative method, bits 2^0 and 2^1 comprise the section number, bits 2^2 to 2^4 the subsection number, and bits 2^5 to 2^7 the bank number. For example, assume address bits 2^7 to 2^0 are 01010101_2. Using the relative method , the address is in Section 1, Subsection 5, Bank 2. Using the absolute method, the address is in Bank 125_8 (Bank 85_{10}).

The remainder of this section uses the relative method of numbering subsections and banks.

Address Range Checking

Four registers in the Exchange Package are used to place a program's data and instruction fields in specific locations in memory and to allocate specific amounts of memory to the fields. This has two benefits. First, all programs are relocatable. When a program is written, the programmer does not need to know where in memory the instruction and data fields will be located. Second, each program can have its memory access restricted to certain parts of memory. A program can be halted if it tries to execute an instruction outside of its allowed instruction range or if it tries to read or write data outside of its allowed data range. This is especially important where more than one program occupies memory at the same time: programs can be prevented from executing instructions or operating on data that belongs to other programs.

The Data Base Address (DBA) register determines where in memory a program's data field is located. Addresses generated by memory reference instructions are relative to the DBA register. Each time an instruction makes a memory reference, the memory address generated by the instruction is added to the contents of the DBA register to form the absolute memory address.

The Data Limit Address (DLA) register determines the highest absolute memory address the program can use for reading or writing data. Each time an instruction makes a memory reference, the absolute address generated is compared to the DLA register. If the absolute address is less than the DLA register, the reference is allowed to proceed. If the absolute address is equal to or greater than the DLA register, an out-of-range condition exists and the memory reference is aborted by disabling all chip selects and write enables in the referenced memory bank. For a memory write reference, no write operation is performed. For read reference, all bits are set to zero.

If the Interrupt on Operand Range Error (IOR) flag (in the Exchange Package Mode register) is set, the out-of-range condition sets the Operand Range Error (ORE) flag (in the Exchange Package Flag register) and causes an Exchange Sequence to begin. If the IOR flag is clear, program execution is allowed to continue.

The Instruction Base Address (IBA) register functions similarly to the DBA register, except that it operates on a program's instruction field. Each time an instruction fetch sequence takes place, absolute memory addresses are formed by adding the relative addresses generated by the fetch control logic to the contents of the IBA register.

The Instruction Limit Address (ILA) register functions similarly to the DLA register, except that it operates on a program's instruction field and there is no provision for continuing program execution when an out-of-range condition occurs. If an absolute memory address generated by an instruction fetch sequence is less than the ILA register, the fetch sequence is allowed to proceed. If the absolute address is equal to or greater than the ILA register, an out-of-range condition exists. This sets the Program Range Error flag (in the Exchange Package Flag register) and disables all chip selects in the referenced memory bank. All bits in the referenced bank are read as zeros and transmitted to the active instruction buffer. Because all bits are zero, the instruction is executed as an Error Exit (000 instruction), which causes an Exchange Sequence to begin.

The DBA, DLA, IBA, and ILA registers contain only address bits 2^8 to 2^{27}. Bits 2^0 to 2^7 are always 0. Therefore the contents of these registers is always a multiple of 400_8

(256_{10}). The data and and instructions fields must begin on a 400_8 word boundary and must be a multiple of 400_8 words long. Adding the contents of the DBA or IBA register to a relative memory address does not change the section, subsection, and bank number. Therefore, memory conflicts can be determined from the relative addresses generated by instructions and the fetch control logic. It is not necessary use absolute addresses to determine if conflicts exist.

Address range checking is not performed during Exchange Sequences and I/O transfers. Memory addresses generated by these operations are absolute addresses.

Error Detection and Correction

Single Error Correction, Double Error Detection (SECDED) monitors Central Memory for data errors. Memory errors involving only one bit in each data word (single-bit errors) can be detected and corrected. Double-bit errors can be detected, but cannot be corrected. Errors involving more than two bits cannot be reliably detected.

The SECDED error processing scheme was developed by R. W. Hamming.† When a 64-bit word (bits 2^0 to 2^{63}) is written to memory, an 8-bit check byte is generated and stored in memory with the data word. (The check bits are number 0 to 7 and are stored as data bits 2^{64} to 2^{71}.) When the word is read from memory, a check byte is again generated and compared with the original check byte, using a bit-by-bit Exclusive-OR. The resulting comparison is called a syndrome. If all the bits in the syndrome are 0, the two check bytes are identical and no memory error occurred.

If there are one or more 1s in the syndrome, some type of memory error occurred. The type of memory error (single-bit or double-bit) can be determined by interpreting the syndrome. If a single-bit error occurred, the syndrome indicates the bit in error. The SECDED logic toggles the incorrect bit to its correct value. If a double-bit error occurred, the syndrome indicates that there was an error, but it cannot pinpoint the incorrect bits. Errors involving more than two bits produce unpredictable results. In some cases they produce unique syndromes that can be detected by the SECDED logic. In other cases the syndrome appears to be a no-error condition or a single- or double-bit error.

Table 2-5 shows the data bits used to generate each bit in the check byte. All data bits marked with an *X* in a row contribute to the corresponding check bit. The parity of all such data bits determines the state of the check bit. If the parity is even, the check bit is set to zero. If it is odd, the check bit is set to one. For example, the data bits that make up check bit 0 are bits 2^1, 2^3, 2^5, 2^7, 2^9,2^{11}, 2^{13}, 2^{15}, 2^{17}, 2^{19}, 2^{21}, 2^{23}, 2^{25}, 2^{27}, 2^{29}, and 2^{31} to 2^{55}. If an even number of these bits is one, check bit 0 is set to logic 0. Otherwise it is set to logic 1.

If a syndrome other than zero is generated, memory error information is sent to the error channel (read "Error Channel" in section 9 of this manual) to help pinpoint the hardware failure. A non-zero syndrome may also initiate an Exchange Sequence, depending on the state of two flags in the Exchange Package Mode register. If the

† Hamming, R. W., "Error Detection and Correcting Codes," *Bell System Technical Journal*, 29, No. 2, pp. 147-160 (April, 1950).

Table 2-5. Check Bit Generation

	Data Bits								Data Bits							
	2^{63}	2^{62}	2^{61}	2^{60}	2^{59}	2^{58}	2^{57}	2^{56}	2^{55}	2^{54}	2^{53}	2^{52}	2^{51}	2^{50}	2^{49}	2^{48}
Check Bit 0									X	X	X	X	X	X	X	X
Check Bit 1	X	X	X	X	X	X	X	X								
Check Bit 2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Check Bit 3	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Check Bit 4	X		X		X		X		X		X		X		X	
Check Bit 5	X	X			X	X			X	X			X	X		
Check Bit 6	X	X	X	X					X	X	X	X				
Check Bit 7	X			X		X	X		X			X		X	X	
	2^{47}	2^{46}	2^{45}	2^{44}	2^{43}	2^{42}	2^{41}	2^{40}	2^{39}	2^{38}	2^{37}	2^{36}	2^{35}	2^{34}	233	2^{32}
Check Bit 0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Check Bit 1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Check Bit 2									X	X	X	X	X	X	X	X
Check Bit 3	X	X	X	X	X	X	X	X								
Check Bit 4	X		X		X		X		X		X		X		X	
Check Bit 5	X	X			X	X			X	X			X	X		
Check Bit 6	X	X	X	X					X	X	X	X				
Check Bit 7	X			X		X	X		X			X		X	X	
	2^{31}	2^{30}	2^{29}	2^{28}	2^{27}	2^{26}	2^{25}	22^{4}	2^{23}	2^{22}	2^{21}	2^{20}	2^{19}	2^{18}	2^{17}	2^{16}
Check Bit 0	X		X		X		X		X		X		X		X	
Check Bit 1	X	X			X	X			X	X			X	X		
Check Bit 2	X	X	X	X					X	X	X	X				
Check Bit 3	X			X		X	X		X			X		X	X	
Check Bit 4									X	X	X	X	X	X	X	X
Check Bit 5	X	X	X	X	X	X	X	X								
Check Bit 6	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Check Bit 7	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
Check Bit 0	X		X		X		X		X		X		X		X	
Check Bit 1	X	X			X	X			X	X			X	X		
Check Bit 2	X	X	X	X					X	X	X	X				
Check Bit 3	X			X		X	X		X			X		X	X	
Check Bit 4	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Check Bit 5	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Check Bit 6									X	X	X	X	X	X	X	X
Check Bit 7	X	X	X	X	X	X	X	X								

Interrupt on Correctable Memory Error (ICM) flag is set, a single-bit (correctable) memory error sets the Memory Error (ME) flag in the Exchange Package Flag register and starts an Exchange Sequence. If the Interrupt on Uncorrectable Memory Error (IUM) flag is set, a double-bit or detectable multiple-bit (uncorrectable) error sets the ME flag and starts an Exchange Sequence. If either the ICM or the IUM flag is clear, the corresponding type of memory error does not start an Exchange Sequence and does not set the ME flag.

CPU MODULE

The remainder of this section is organized according to the physical location of Central Memory hardware in the mainframe. The subsection explains Central Memory hardware on the CPU module. The next subsection explains the Memory module.

Central Memory is the largest functional area on the CPU module. Fifty-four chips, consisting of seventeen option types, are used exclusively by Central Memory. Two other chips, each a unique option type, are used primarily by Central Memory but are shared with other parts of the CPU. With the exception of inter-CPU conflict detection and resolution logic, each CPU module operates independently. Inter-CPU conflicts are detected and resolved jointly among the eight CPU modules; each module handles conflicts for four memory subsections. This section describes the major Central Memory components on the CPU module.

Input and Output Terms

The CPU module has several types of input and output terms that are used by Central Memory. The major are:

- Write data terms to each memory section
- Read data terms from each memory section
- Address and control terms to each memory section and to the inter-CPU conflict resolution logic
- Control terms from the inter-CPU conflict resolution logic

Other terms are used by the inter-CPU conflict detection and resolution logic on each CPU module. These terms are:

- Address and control terms from each CPU
- Control terms to each CPU

Table 2-6 lists the memory input and output terms on the CPU module. Each of the terms listed using logical connector CNN is actually four terms, one for each memory section. Read and write data bits 0 to 71 connect to the memory modules. They consist of 64 bits of actual data and 8 check bits. Address bits 2^5 to 2^7 perform bank selection. Copies 0 and 1 go to the memory modules. Copies 2 and 3 are used by the inter-CPU conflict resolution logic. Address bits 2^8 to 2^{24} go to the memory modules. Address bits

Table 2-6. CPU Module Inputs and Outputs

Logical Edge Connector	Boolean Term(s)	Name	Type	Module Side
CNN	R0 to R71	Section N Write Data Bits 0 to 71	Data Ouput	Z Side
CNN	I0 to I71	Section N Read Data Bits 0 to 71	Data Input	Y Side
CNN	R72 , R76, and R80	Section N Address Bits 2^5 to 2^7 (copy 0)	Address Output	Z Side
CNN	R73 , R77, and R81	Section N Address Bits 2^5 to 2^7 (copy 1)	Address Output	Z Side
CNN	R74 , R78, and R82	Section N Address Bits 2^5 to 2^7 (copy 2)	Address Output	Z Side
CNN	R75 , R79, and R83	Section N Address Bits 2^5 to 2^7 (copy 3)	Address Output	Z Side
CNN	R84 to R103	Section N Address Bits 2^8 to 2^{27}	Address Output	Z Side
CNN	R104	Section N Go Write	Control Output	Z Side
CNN	R105	Section N Abort	Control Output	Z Side
CNN	R106 to R120 even	Section N GOSS 0 to 7 (copy 0)	Control Output	Z Side
CNN	R107 to R121 odd	Section N GOSS 0 to 7 (copy 1)	Control Output	Z Side
CNN	R122 to R124	Section N Subsection Read Select 2^0 to 2^2	Control Output	Y Side
CN4	I0 to I31	Release Subsection 0 to 31	Control Input	Z Side
CN5	I0, I7, ... I49	CPU A to H GOSS X	Control Input	Z Side
CN5	I1, I8, ... I50	CPU A to H GOSS X + 1	Control Input	Z Side
CN5	I2, I9, ... I51	CPU A to H GOSS X + 2	Control Input	Z Side
CN5	I3, I10, ...I52	CPU A to H GOSS X + 3	Control Input	Z Side
CN5	I4 to I6, I11 to I13, ... I53 to I55	CPU A to H Address Bits 2^5 to 2^7	Control Input	Z Side
CN5	R0, R4, ... R28	CPU A to H Release Subsection X	Control Output	Z Side
CN5	R1, R5, ... R29	CPU A to H Release Subsection X + 1	Control Output	Z Side
CN5	R2, R6, ... R30	CPU A to H Release Subsection X + 2	Control Output	Z Side
CN5	R3, R7, ... R31	CPU A to H Release Subsection X + 3	Control Output	Z Side

go as high as bit 2^{27}, making the CPU module capable of addressing 256 Mwords of memory. However, with 32 Mwords of memory, address bits 2^{25} to 2^{27} are not used.

The control terms perform several functions. Go Write and Abort are used by the CPU modules. Go Write is logic 0 for a read reference and logic 1 for a write reference. Abort prevents data from being read from or written to memory in case of a memory range error. Go Subsection (GOSS) terms are used to activate a subsection for reading or writing. Copy 0 is used by the memory modules. Copy 1 is used by the inter-CPU conflict resolution logic. Subsection Read Select 2^0 to 2^2 are used by the memory modules during a read reference. They determine which subsection is the source of the read data. Release Subsection terms come from the inter-CPU conflict resolution logic and determine when a subsection is available for use by the CPU (that is, when the previous reference by the same CPU to the same subsection has completed).

The inter-CPU conflict resolution section has several control and address terms. The GOSS terms that are received from each CPU indicate when the CPU has made a reference to a subsection. Address bits 2^5 to 2^7 determine which bank within the subsection each CPU has referenced. A Release Subsection is generated when a CPU completes its reference to a subsection.

Write Data Paths and Check Bit Generation

Write data paths transmit data from CPU registers and the I/O section to the memory modules. Twelve options comprise the write data paths–eight YD and four YC options. The YD options steer data from the CPU registers and the I/O section to the memory modules. For CPU data, the YC options generate a check byte for each data word and steer the check byte to the memory modules. For I/O data, the YC options steer the check byte generated in the I/O section. The YD and YC options can also stack CPU data and check bits when there is a memory conflict. Data and check bits can be held in the options until the conflict is resolved, then sent to the memory modules. Each YD and YC option also contain a fanout. Most of these fanouts are used by write data control signals; the other fanouts are not used.

Figure 2-3 shows the write data paths on the CPU module. Each YD option steers eight data bits. Each YC option generates and steers two check bits. A-register and exchange data comes into the YDs as I0 to I7 and into the YCs as I0 to I15. In an Exchange Sequence, all 64 bits are active. In an A-register write instruction, only the lower 24 bits in X-mode and the lower 32 bits in Y-mode are active; the inactive bits are forced to 0 by the AR and HF options.

B-register data bits 2^0 to 2^{23} are delayed on the VR options. Data bits 2^{24} to 2^{31} are gated on the YL and YO options; they are forced to 0 in X-mode. B-register data comes into the YD options as I8 to I15 and into the YC options as I16 to I31. Data bits 2^{32} to 2^{63} are forced to 0 on the YD and YC inputs.

S, T, and V-register data enter the YD options as I16 to I23 and into the YC options as I32 to I47. All 64 bits are active. I/O data enters the YD options as I24 to I31. The I/O check byte enters the YC options as I48 to I49.

Select A Data and Select B data come from the JA and JD options and are fanned out by YD options. They then enter as I40 and I41 on the YDs and as I62 and I63 on the YCs. As shown in Table 2-7, they normally select the source for CPU write data. They also

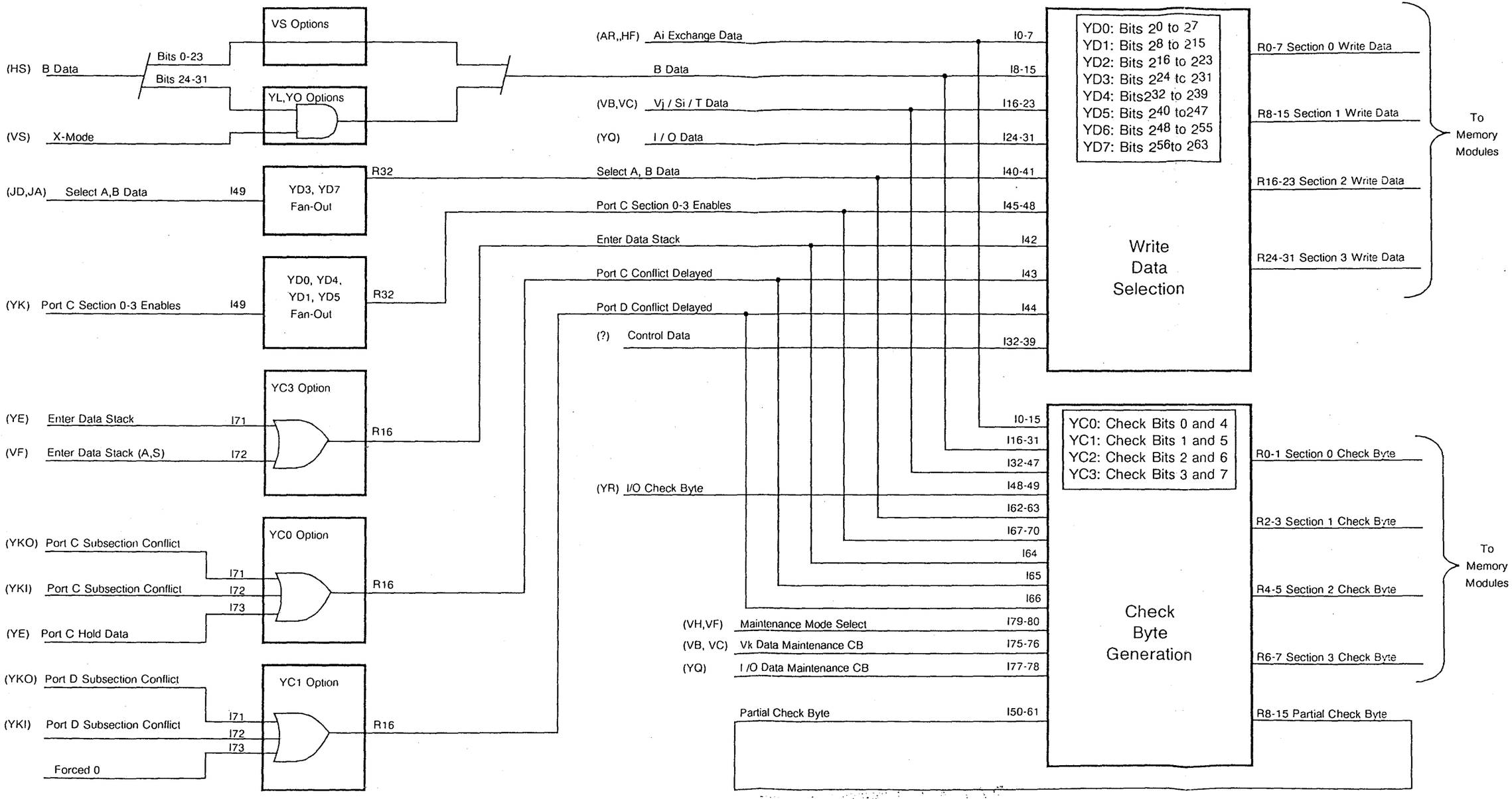


Figure 2-3. Write Data Paths

can select a maintenance function in which all data bits are forced to one and all check bits are forced to zero.

Table 2-7. CPU Write Data Selection

Select A Data	Select B Data	Data Source
0	0	S, T or V register
1	0	A register or exchange data
0	1	B register
1	1	Data bits–forced 1 Check bits–forced 0

Port C Section 0 to 3 Enables come from the YK options and are fanned out by the YD options. They enter the YD options as I45 to I48 and the YC options as I67 to I70. They determine the source of the data and check bits that are sent to the memory modules. If a Port C Section Enable is logic 1, CPU register data is sent to the corresponding memory section. If it is 0, I/O data is sent.

Enter Data Stack and Port C Conflict Delayed (I42 and I43 on the YDs, I64 and I65 on the YCs) are used to resolve Port C memory conflicts. Enter Data Stack comes from the YE option and is fanned out by the YC3 option. When this signal is active, it causes data and check bits to be stacked in the YD and YC options. Port C Conflict Delayed consists of Port C Hold Data from the YE option and Port C Subsection Conflict from the YK options ORed together and fanned out on the YC0 option. This term causes data and check bits to be held in the stack. When the conflict is resolved, data and check bits are sent to the CPU module in the same order that they were placed in the stack. Up to eight sets of data and check bits can be held in the stack at one time.

Port D Conflict (I44 on the YDs, I66 on the YCs) consists of Port D Subsection Conflict terms from the YK options which are ORed together and fanned out by the YC1 option. When this term is logic 1, I/O data and check bits are held on the YD and YC options. When it is logic 0 the bits are released, allowing new I/O data and check bits to enter the options.

Terms R0 to R31 from the YD options are the write data bits which go to the memory modules. Terms R0 to R7 from the YC options are the check bits which go to the memory modules. YC option terms R8 to R15 are partial check bytes which feed into other YC options as terms I50 to I61.

Several maintenance functions are built into the YC and YD options. When Select A Data and Select B Data are both at logic 1, all CPU data bits are forced to logic 1 and all CPU check bits are forced to 0, regardless of the values in the CPU registers. Control data (YD options, term I32 to I39) from the J? option also replace the normal I/O data in this case.

The Maintenance Mode input (term I79' on the YC options) from the VH option permits maintenance check bits to be used in place of the normal check bits. When this term is active (I79 is logic 0), the Maintenance Select input (term I80') determines which of two maintenance functions is active. If I80 is logic 1, CPU check bits are generated normally. I/O check bits are taken from I77 and I78, which are maintenance check bits from the I/O section. If I80 is logic 0, I/O check bits are taken from the normal path, I48 and I49. CPU check bits are modified by terms I75 and I76, which are maintenance check bits from the Vector registers.

Read Data Paths and Error Correction

Read data paths transmit data from the memory modules to the CPU registers and the I/O section. Twenty options comprise the read data paths–8 YP and 12 YQ options. The YP options steer data from the memory modules to memory Ports A, B, and D. The YQ options perform error detection and correction for each port. Each YP option also contains delays for control signals. The delays which pertain directly to the read data paths are explained in the following paragraphs. For information on the other delays, read "Control Paths" in this section.

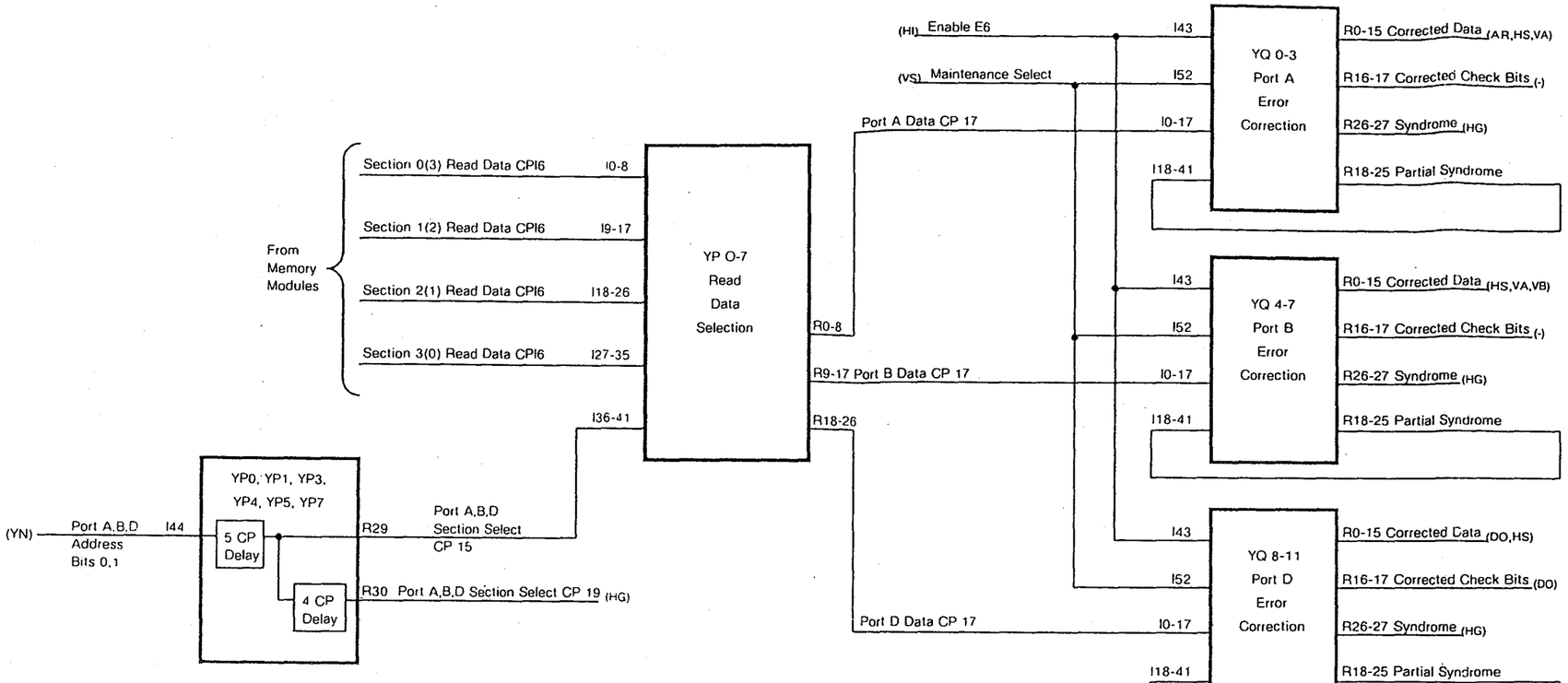
Figure 2-4 shows the read data paths on the CPU module. Each YP option steers eight data bits and one check bit. Six of the YPs delay control signals for the read data paths. In YP0 to YP3, Section 0 data and check bits enter as I0 to I8, Section 1 as I9 to I17, Section 2 as I18 to I26, and Section 3 as I27 to I35. In YP4 to YP7, the sections are reversed (section 0 bits enter as I27 to I35, section 1 as I18 to I26, etc.) Port A, B, and D address bits 2^0 and 2^1 enter six of the YP options as I44, are delayed 5 CPs, and exit as R29. These terms enter the YP options as I36 to I41 and determine the source (section number) of data and check bits for each port. On YP3 to YP7, I36 to I44 are connected to R29' from the YP delays, which reverses the order that the memory sections are connected to I0 to I35. The R29 terms are delayed an additional 4 CPs in the YP options and exit as R30. These terms are sent to the HG option and determine the section number in case of a memory error. Port A data and check bits exit the YP options as R0 to R8, Port B as R9 to R17, and Port D as R18 to R26.

Each YQ option generates 16 corrected data bits and two corrected check bits for one port. The data and check bits from the YP options enter the YQ options as I0 to I17. Partial syndromes are generated by each YQ option and exit as R18 to R25. The partial syndromes enter other YQs as I18 to I41. The partial syndromes generate corrected data (R0 to R15), corrected check bits (R16 and R17), and a complete syndrome (R26 and R27).

Port A corrected data goes to the address registers (AR options), B registers (HRs), and V registers (VAs). Port B corrected data goes to the S registers (VBs), B registers (HRs), and V registers (VAs). Port D corrected data goes to the instruction buffers (HR options), and the I/O section (DOs). Port A and B corrected check bits are not used. Port D corrected check bits go to the DO option in the I/O section.

Syndromes from all ports are sent to the HG option. If the syndrome is non-zero, a memory error has occurred. The HG option reports the error to the error channel and may initiate an Exchange Sequence, depending on the type of error (correctable or uncorrectable) and the state of two interrupt flags in the Mode register.

Two maintenance features on the YQ options help pinpoint memory hardware failures. First, error correction can be disabled. When the Enable Error Correction input (term



Option	Bits		R29, R30 Delay	Module Location
	Data Bits	Check Bit		
YP0	2^0 to 2^7	0	Port A Section Select 2^0	?
YP1	2^8 to 2^{15}	1	Port A Section Select 2^1	?
YP2	2^{16} to 2^{23}	2	Not used for read data selection	?
YP3	2^{24} to 2^{31}	3	Port D Section Select 2^0	?
YP4	2^{32} to 2^{39}	4	Port B Section Select 2^0	?
YP5	2^{40} to 2^{47}	5	Port B Section Select 2^1	?
YP6	2^{48} to 2^{55}	6	Not used for read data selection	?
YP7	2^{56} to 2^{63}	7	Port D Section Select 2^1	?

Option	Port	Bits		Module Location
		Data Bits	Check Bits	
YQ0	Port A	2^0 to 2^7 and 2^{32} to 2^{39}	0 and 4	?
YQ1	Port A	2^8 to 2^{15} and 2^{40} to 2^{47}	1 and 5	?
YQ2	Port A	2^{16} to 2^{23} and 2^{48} to 2^{55}	2 and 6	?
YQ3	Port A	2^{24} to 2^{31} and 2^{56} to 2^{63}	3 and 7	?
YQ4	Port B	2^0 to 2^7 and 2^{32} to 2^{39}	0 and 4	?
YQ5	Port B	2^8 to 2^{15} and 2^{40} to 2^{47}	1 and 5	?
YQ6	Port B	2^{16} to 2^{23} and 2^{48} to 2^{55}	2 and 6	?
YQ7	Port B	2^{24} to 2^{31} and 2^{56} to 2^{63}	3 and 7	?
YQ8	Port D	2^0 to 2^7 and 2^{32} to 2^{39}	0 and 4	?
YQ9	Port D	2^8 to 2^{15} and 2^{40} to 2^{47}	1 and 5	?
YQ10	Port D	2^{16} to 2^{23} and 2^{48} to 2^{55}	2 and 6	?
YQ11	Port D	2^{24} to 2^{31} and 2^{56} to 2^{63}	3 and 7	?

Figure 2-4. Read Data Paths

I43) from the HI option is logic 0, error correction is disabled. In this condition, data bits pass through YQ options without correction and the corrected check bit outputs (R16 and R17) are replaced by the syndrome. The syndrome outputs (R26 and R27) are forced to 0.

When the Maintenance Select input (I52) from the VS option is logic 1 (regardless of the state of I43), error correction is also disabled. Data bits 2^1 to 2^7, 2^9 to 2^{15}, 2^{17} to 2^{23}, 2^{25} to 2^{31}, 2^{33} to 2^{39}, 2^{41} to 2^{47}, 2^{49} to 2^{55}, and 2^{57} to 2^{63} pass through without correction. Data bits 2^0, 2^8, 2^{16}, 2^{24}, 2^{32}, 2^{40}, 2^{48}, and 2^{56} are replaced by uncorrected check bits 0 to 7. The corrected check bit outputs are replaced by the syndrome. The syndrome outputs are forced to 0.

Address Paths and Address Range Checking

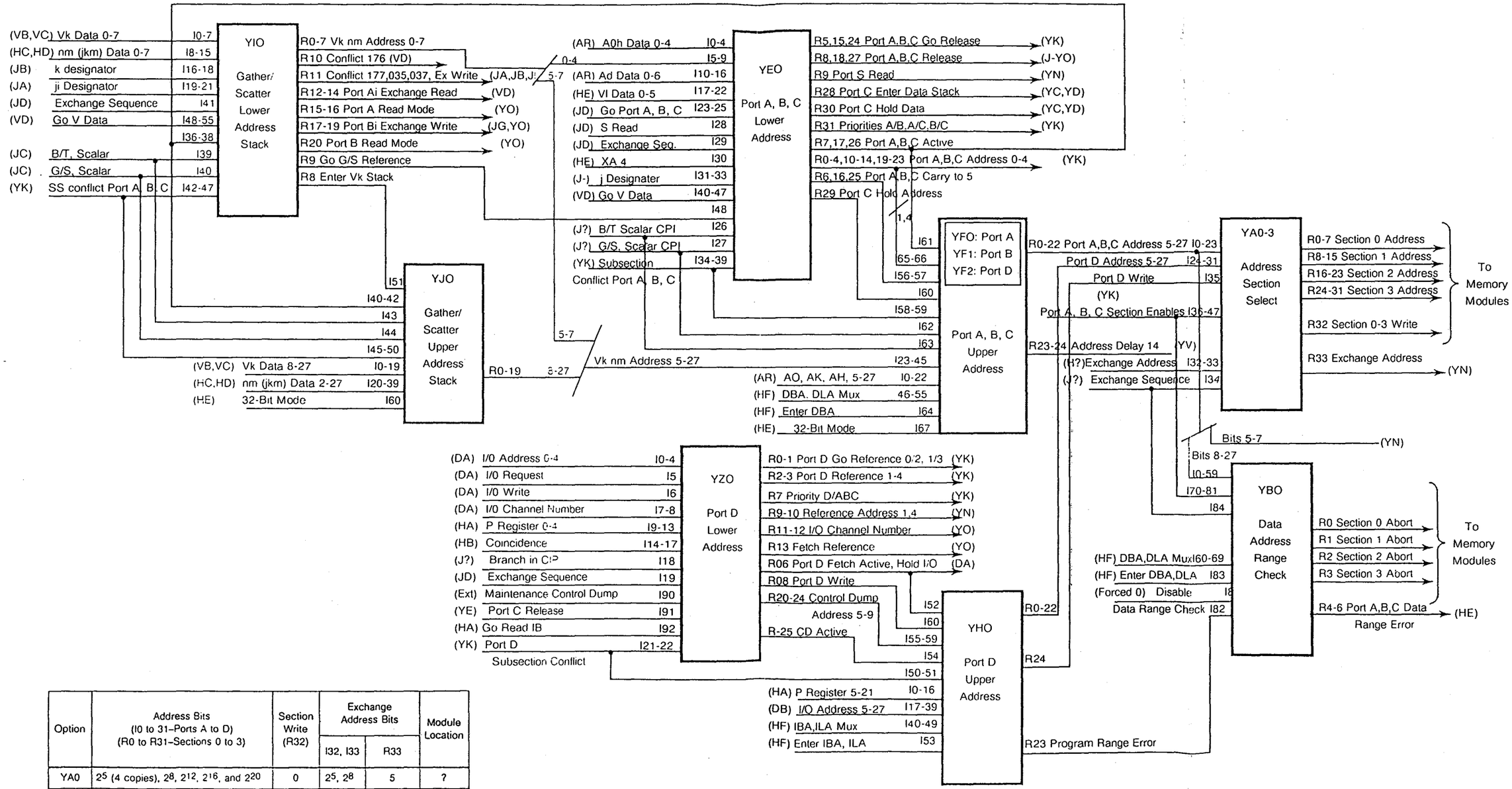
Address paths transmit address information from instruction fields, CPU registers, and the I/O section to the memory modules. For block and vector transfers, the address paths generate the addresses needed for the successive memory references and stop the operation when all references have been completed. Thirteen options comprise the address paths: four YA, one YB, one YE, three YF, one YZ , one YH, one YI, and one YJ option. The YA options steer address bits from each memory port to the memory sections and, for write references, send Section Write to the memory sections. The YB option performs address range checking for ports A, B, and C. The YE and YF options generate addresses for Ports A, B, and C, determine when block and vector instructions have completed, and perform address range checking for all memory reference instructions. The YZ and YH options generate addresses for Port D memory references. For instruction fetches, the YH option performs address range checking. The YI and YJ options are used mainly for stacking gather/scatter addresses during memory conflicts. They also transmit address information to the YE and YF options for scalar transfers. Figure 2-5 shows the address paths on the CPU module. Refer to this figure while reading the following subsections.

YA and YB Options

Each YA option can steer eight address bits to each memory section, generate one Section Write, and supply one Exchange Address bit to the YN options. Ports A to D address bits enter the YA options as I0 to I31. Four copies of address bits 2^5 to 2^7 are used in order to meet fanout requirements on the memory module. Only one copy of address bits 2^8 to 2^{27} is needed. The inverted side of Port A, B and C Section Enables enter the YA options as I36 to I47. Whenever one of these signals is active (for example Port A Section 0 Enable), address bits are steered from the named port to the corresponding memory section. If all three Section Enables are inactive for any section, Port D address bits are steered to that section. Section 0 to 3 address bits exit the YA options as R0 to R31 and go to the memory modules.

A write reference to a memory section activates the corresponding Section 0 to 3 Write term (R32). Since Ports A and B are read-only ports, a Section Write signal is never generated for references coming from those ports. Port C is write-only; a reference from Port C enables the appropriate Section Go Write. Port D references can be either read or write references; a Section Write is generated if Port D Write (term I35) from the YH option is logic 1.

The YA option operates slightly differently during an Exchange Sequence. When Exchange Sequence (term I34) is logic 1, Port A, B, and C address inputs are not used.



Option	Address Bits (I0 to 31–Ports A to D) (R0 to R31–Sections 0 to 3)	Section Write (R32)	Exchange Address Bits		Module Location
			I32, I33	R33	
YA0	2^5 (4 copies), 2^8, 2^{12}, 2^{16}, and 2^{20}	0	2^5, 2^8	5	?
YA1	2^6 (4 copies), 2^9, 2^{13}, 2^{17}, and 2^{21}	1	2^6, 2^9	6	?
YA2	2^7 (4 copies), 2^{10}, 2^{14}, 2^{18}, and 2^{22}	2	2^7, 2^{10}	7	?
YA3	2^{24} to 2^{27}, 2^8, 2^{11}, 2^{15}, and 2^{23}	3	–, 2^{11}	not used	?

Figure 2-5. Address Paths

Exchange Address bits (terms I32 and I33) take the place of address bits 2^5 to 2^{11} and address bits 2^{12} to 2^{27} are forced to 0 for any section in which the Port A, B, or C Section Select is logic 1. This forces all exchange addresses to be less than 10000_8. Exchange address bits 2^5 to 2^7 (bank selection) exit YA0 to YA2 as R33 and enter the YN options. Port D memory references can operate while an Exchange Sequence is in progress.

The YB option performs address range checking for data transfers between Central Memory and CPU registers (Ports A, B, and C). During each Exchange Sequence, Enter DBA,DLA (term I83) goes active. This loads copies of the DBA and DLA registers into the YB option. The contents of the registers are multiplexed through I60 to I69. DBA bits 2^8 to 2^{17} are loaded first, followed by DBA bits 2^{18} to 2^{27}, then by DLA bits 2^8 to 2^{17}, and finally by DLA bits 2^{18} to 2^{27}. Address bits 2^8 to 2^{27} from Ports A, B, and C enter as terms I0 to I19. Each address is compared with the DLA register. If the address is greater than the DLA register, an address range error has occurred.

Each address is also compared with the DBA register to detect address overflows (that is, addresses beyond the maximum physical memory address). An address overflow may occur in the YF options, where the contents of the DBA register is added to the relative addresses generated by instructions. Since relative addresses and the DBA register contain address bits up to 2^{27}, their sum may require address bits up to 2^{28}. However, the YF options do not contain address bit 2^{28} and do not detect addition overflow; they send only address bits up to 2^{27} to the YA options. The YB option detects the overflow condition by comparing the address (up to bit 2^{27}) with the DBA register. An overflow makes the address less than the DBA register.

When the YB option detects a data range error, it sends Port A, B, or C Data Range Error (terms R4 to R6) to the HE option and looks at the Ports A, B, and C Section Enables (terms I70 to I81) to determine which section is receiving the address. It then sends a Section Abort signal (terms R0 to R3) to the proper memory section. Data range checking can be disabled by activating Disable Data Range Check (term I85'). This forces all Port Data Range Error outputs to logic 0 and disables the Section Aborts.

The YB option also generates Section Aborts when an instruction fetch sequence attempts to access illegal addresses. In this case, the YB option receives Program Range Error as I82'. This generates a Section Abort for each memory section that is not enabled by a Port A, B, or C Section Enable. Section aborts generated by program range errors are not affected by the state of Disable Data Range Check.

YE and YF Options

The YE and YF options generate addresses for Ports A, B, and C. The YE option generates address bits 2^0 to 2^4 for all three ports. Each YF option generates address bits 2^5 to 2^{27} for one port. YF0 is used for Port A, YF1 for Port B, and YF2 for Port C. The YE and YF options also generate some control terms.

From the YE option, terms R0 to R4, R10 to R14, and R19 to R23 are address bits 2^0 to 2^4 for Ports A, B, and C. Terms R5, R15, and R24 are the Go Reference terms for each port. Go Reference goes to logic 1 for 1 CP when a port begins a memory reference. Terms R6, R16, and R25 are the Carry 5 terms for each port. When a memory address is generated by addition (for example, a block transfer address increment), Carry 5 is the carry bit into address bit 2^5. R7, R17, and R26 are the Active terms for each port. Active goes to logic 1 at the beginning of a memory reference instruction and remains in that state for the duration of the instruction. R8, R18, and R27 are Release terms for each port.

Release goes active for 1 CP at the end of a memory reference instruction. R9 is Port A S Read. It goes active for 1 CP during an memory read to an S register read and indicates that the read data must use Port B paths. R28 to R30 are Port C Enter Data Stack, Hold Data, and Hold Address. They temporarily suspend memory operations when memory conflicts occur. R31 to R33 indicate relative priorities between Ports A, B, and C for use in resolving section conflicts.

From each YF option, terms R0 to R22 are address bits 2^5 to 2^{27} for one port. Terms R23 and R24 are address bits 2^1 and 2^4 from one port, delayed until memory conflicts for the port have been resolved.

On the input side to the YE and YF options are address inputs from the A registers, V registers. and the instruction field. Terms I0 to I4 on the YE option are base address bits 2^0 to 2^4 for all memory reference instructions. They come from register A*h* for scalar memory transfers and from register A0 for block and vector transfers. Terms I5 to I9 serve different functions, depending on the instruction type. For scalar instructions, they are bits 2^0 to 2^4 of the *nm* field (*jkm* field in X-mode) of the instruction. For gather and scatter instructions, they are bits 2^0 to 2^4 of the V*k* elements. In each of these cases, the YE option adds I5 to I9 to the base address to form absolute addresses. I5 to I9 are not used for block and stride instructions. I10 to I16 serve different functions, depending on the instruction type. For block transfers, these terms indicate the block count. For stride transfers, they are the lower 7 bits of the increment value from the A*k* register. They are not used for scalar and scatter/gather instructions. Terms I17 to I22 are the vector length and are used only for vector instructions.

YF option terms I0 to I22 are address bits 2^5 to 2^{27}. They come from register A*h* for scalar memory transfers and from register A0 for block and vector transfers. For stride transfers, the address increment value from the A*k* registers is multiplexed on I0 to I22; at the beginning of a stride transfer, A0 bits are loaded into the YF option, followed by the Ak bits the next CP. Terms I23 to I45 serve different functions, depending on the instruction type. For scalar instructions, they are bits 2^8 to 2^{27} of the *nm* field (*jkm* field in X-mode) of the instruction. For gather and scatter instructions, they are bits 2^0 to 2^4 of the V*k* elements. In each of these cases, the YE option adds I5 to I9 to the base address and to the DBA register (bit 2^8 to 2^{27} only) to form absolute addresses.

There are several control inputs to the YE and YF options. Go Port A, B, and C enter the YE option as I23 to I25. One of these terms is active for 1 CP to at the beginning of each memory reference instruction. Terms B/T, Scalar and G/S, Scalar enter the YE option as I26 and I27 and the YF options as I62 and I63 at the beginning of a memory reference instruction. As shown in Table 2-8, these terms determine the type of memory operation for each port. The term S Read enters the YE option as I28. This term indicates that read data should use Port B memory paths. Terms I29 and I30 on the YE options are used during Exchange Sequences. Term I29 indicates that an Exchange Sequence is in progress. Term I30 is exchange address bit 2^4.

Terms I33 to I47 and I40 to I47are used only for vector transfers to memory. I31 to I33 are the *j* field of the instruction, which indicates the V register that is the source of the data. I40 to I47 are Go V Data for V registers 0 to 7. For a vector write instruction to execute without interruption, the appropriate Go V Data must be at logic 1 until the write instruction has completed all its references. If Go V data goes to logic 0 temporarily, the instruction suspends operation until Go V Data returns to logic 1. Such suspensions can occur during chained operations, where the V register that is the source of the write data is waiting to receive data from a previous operation. Term I48 on the

YE option is Go G/S Reference. During a gather or scatter instruction, this term goes to logic 1 each time a memory reference begins.

During each Exchange Sequence, Enter DBA (YF option term I64) goes active. This loads a copy of the DBA registers into the YF option. The contents of the register are multiplexed through I46 to I55. DBA bits 2^8 to 2^{17} are loaded first, followed by bits 2^{18} to 2^{27}. YF option term I67 is logic 1 in Y-mode, allowing instructions to access data in the full 28-bit physical range. In X-mode, I67 is logic 0. This limits instruction addresses to a 24-bit range by setting the upper four address bits (2^{24} to 2^{27}) equal to the bits 2^{24} to 2^{27} of the DBA register.

Table 2-8. Memory Reference Instruction Types

Boolean Terms		Type of Memory Reference
B/T, Scalar	G/S, Scalar	
0	0	Stride (V register)
0	1	Gather or Scatter (V register)
1	0	Block (B or T registers)
1	1	Scalar (A or S register)

YZ and YH Options

The YZ and YH options handle memory addresses for Port D. The YZ option handles address bits 2^0 to 2^4. The YH option handles address bits 2^5 to 2^{27}. The YZ and YH options also generate some control terms. Those control terms that deal with memory address paths are explained in this section. For information on the other control terms generated by the YZ and YH options, read "Control Paths" in this section.

Each time the I/O section requests a memory reference, it sends information regarding the reference to the YZ and YH options. The YZ option receives the following terms: I/O address bits 2^0 to 2^4 (terms I0 to I4), I/O request (I5), I/O Write (term I6–logic 0 for a read request, logic 1 for a write request), and I/O channel number 2^0 and 2^1 (I7 and I8). The YH option receives I/O address bits 2^5 to 2^{27} as I17 to I39.

Instruction fetch requests are generated by the YZ option. When there is a no-coincidence condition in all four instruction buffers, the Coincidence terms (YZ option terms I14 to I17 are all logic 0. This causes an the YZ option to initiate an instruction fetch sequence. The initial fetch address is determined by P-register bits 2^0 to 2^4, which enter the YZ option as I9 to I13, and bits 2^5 to 2^{21}, which enter the YH option as I0 to I16. After the fetch begins, the YZ option increments address bits 2^0 to 2^4 until all 32 memory locations have been addressed. The YZ option does not initiate a fetch sequence if there is a branch instruction in CIP (term I18 active) or if an Exchange Sequence is in progress (term I19 active).

Port D subsection conflicts enter the YZ option as I21' and I22' and the YH option as I50 and I51. If one of these terms is active, Port D references hold until the conflict is resolved. During each Exchange Sequence, Enter IBA,ILA (YH option term I53) goes active. This causes copies of the IBA and ILA registers to be loaded into the YH option. The contents of the registers are multiplexed through I40 to I49. IBA bits 2^8 to 2^{17} are loaded first, followed by IBA bits 2^{18} to 2^{27}, then by ILA bits 2^8 to 2^{17}, and finally by ILA bits 2^{18} to 2^{27}.

YZ output terms include Port D Go Reference 0/2 (term R0) which goes to logic 1 when Port D requests a reference to Section 0 or 2 and remains logic 1 until the reference has begun. Port D Go Reference 1/3 (term R1) operates the same way for references to Sections 1 or 3. Port D address bits 2^1 to 2^4 exit as R2 to R3.

During an instruction fetch sequence, the term Port D Fetch Active, Hold I/O (R6) becomes active, forcing the I/O section to hold I/O operations until the fetch is completed. Priority D/ABC (R7) indicates the priority of Port D relative to Ports A,B, and C in order to resolve section conflicts. R7 is normally logic 0, but goes to logic 1 during fetch sequences. It also goes to logic 1 if an I/O reference request has held for 32 CPs and remains logic 1 until the I/O reference is made. Port D Write (R8) is logic 1 for I/O write references and logic 0 for I/O read and fetch references. Terms R9 to R13 go active when Port D requests a memory reference and remains active until the reference has begun. R9 and R10 are fetch addresses 2^1 and 2^4. R11 and R12 are I/O channel number 2^0 and 2^1. Fetch Reference (R13) is logic 0 if an I/O reference is holding and logic 1 if a fetch request is holding.

On the YH option, terms R0 to R22 are Port D address bits 2^5 to 2^{27}. For an I/O reference, the address bits come directly from the I/O section. For a fetch reference, the YH option adds bits 2^8 to 2^{27} of the P register to the contents of the IBA register to obtain the absolute address. If bits 2^8 to 2^{27} of the absolute address are equal to or greater than the ILA register, Program Range Error (term R23) goes to logic 1.

An instruction buffer dump feature in the YZ and YH options can be used to detect hardware failures in the instruction buffers. When YZ option term I90 is logic 1, the dump mode is enabled. The dump mode is then activated at the end of the next Exchange Sequence (term I19 active) when Port C is released (term I91 active). The actual dump begins when four fetch sequences have been completed and Go Read IB (term I92) goes to logic 1. This causes the YZ option to send Port D Write and CD active (terms R8 and R25) to the YH option. The YH option forces address bits 2^{10} to logic 1 and bits 2^{11} to 2^{27} to logic 0. Address bits 2^5 to 2^9 are taken from Control Dump Address 5-9 (YZ terms R20 to R24). This addressing scheme causes the instruction buffers to be dumped to memory addresses between 2000_8 to 3777_8.

YI and YJ Options

The YI and YJ options select *nm* data (*jkm* data in X-mode) or V*k* data for the YE and YF options, depending on the type of memory reference instruction. Scalar instructions require *nm* data. Scatter and gather instructions require V*k* data. The YI and YJ options stack V*k* data when a scatter or gather reference incurs a memory conflict. When a conflict occurs, the memory reference is temporarily suspended. The YI option immediately stops the V*k* register from sending data for later references, but data for the next four references has already left the V*k* register. This data is stacked in the YI and YJ registers until the conflict is resolved. Then V*k* data is taken from the stack until the

stack is empty. At this point, the YI and YJ options again begin receiving data from register V*k*. The YI option does multiplexing and stacking for address bits 2^0 to 2^7. The YJ option handles address bits 2^8 to 2^{27}. The YI option also generates some control terms.

Data from register V*k* enters the YI option as I0 to I7 and the YJ option as I0 to I19. Instruction field *nm* (*jkm* in X-mode) enters the YI option as I8 to I15 and the YJ option as I20 to I39. In Y-mode all 32-bits of the *nm* field are used. In X-mode, the YJ option sign-extends the 22-bit *jkm* field to 24 bits and ignores the upper eight bits. Term I60 makes the selection between X-mode and Y-mode. After selection between V*k* data and *nm* data, address information for the YE and YF options exits the YI option as R0 to R7 and the YJ option as R0 to R19.

The terms B/T, Scalar and G/S,Scalar enter the YI option as I39 and I40 and the YJ option as I43 and I44. They determine the type of memory reference in progress and operate identically to YE option terms I26 and I27. Their primary purpose in the YI and YJ options is to select between V*k* and *nm* data for the YE and YF options. Port A, B, and C Active (YI option terms I36 to I38, YJ terms I40 to I42) are used during a gather or scatter reference to determine which port is in use.

YI option terms I16 to I18 and I48 to I55 are used during gather and scatter references. I16 to I18 are the *k* field of the instruction. They designate the V register that is the source of the address information. Terms I48 to I55 are Go V Data for V registers 0 to 7. The YI and YJ options can only receive an element from register V*k* when the corresponding Go V Data input is active. If Go V Data is not active, which can happen during a vector chain, the YI and YJ options temporarily suspend receiving V*k* data. When Go V Data goes active, the YI option begins receiving Vk data and sends Enter Vk stack (term R8) to the YJ option to instruct it to receive V*k* data. When the YI and YJ options have data ready for the YE and YF options, the YI option sends GO G/S Reference (term R9) to the YE option.

Subsection Conflict Port A, B, and C (YI terms I42 to I47, YJ terms I45 to I50) are used for scalar, gather, and scatter references. For a scalar reference, a memory conflict causes the YI and YJ options to hold the *nm* field until the conflict is resolved. When a memory conflict occurs during a gather reference, the YI option activates Conflict 176 (R10). For a scatter conflict, Conflict 177 (R11) is activated. These terms cause the VD options to stop sending V*k* data. Elements that have already left the Vk register are stacked in the YI and YJ options. When the conflict is resolved, R10 or R11 is deactivated, and the VD options resume sending V*k* elements.

The YI option generates several control terms. During execution of scalar or vector memory read instructions, terms R12 to R14 and R17 to R19 are the *i* field of the instruction for Ports A and B. During Exchange Sequences they are 3-bit counters for Read and Write data that increment each time exchange data is read from or written to memory. R15 and R16 are Port A Read Mode bits 2^0 and 2^1. R20 is Port B Read Mode. Read "Error Channel" in Section 9 for information on read modes.

Control Paths

Control paths perform three primary functions:

- They generate the appropriate Go Subsection signal for each memory reference.

- They generate Subsection Readout Selects to steer data from the subsection to the read data paths for memory read references.

- They detect and resolve all intra- and inter-CPU memory conflicts.

Eleven options comprise the major part of the control paths: two YK, four YL, two YM, two YN and one YO option. The YK options detect intra-CPU conflicts. The YL options detect and resolve inter-CPU conflicts and generate Go Subsection terms. The YM options determine when conflicts have been resolved. The YN and YO options delay address and control signals during a conflict. Portions of the YZ and YP options also comprise the control paths. Figure 2-6 shows the control paths on the CPU module.

YK Option

The YK options detect section and subsection conflicts and resolve section conflicts for all memory ports. YK0 detects conflicts in memory sections 0 and 2. YK1 detects conflicts in sections 1 and 3. Requests from ports A,B, and C come from the YE option. Requests from Port D come from the YZ option. Port A requests enter the YK options as I0 to I5. Terms I0 to I3 are address bits 2^2, 2^3, 2^4, and 2^1. Term I4 is address bit 2^0. The inverted side of this bit enters YK0, the upright side YK1. Each option responds to a reference request only when I4 is logic 1. Therefore YK0 handles section 0 and 2 requests and YK1 handles section 1 and 3 requests. Port A Go enters the YK options as I5, which initiates the reference request. Port B and C reference requests operate identically to Port A requests and enter as I6 to I11 and I12 to I17. Port D requests are similar, except that the YZ option sends a separate Go term to each YK option. This eliminates the need to send address bit 2^1 to the YKs. Port D address bits 2^2, 2^3, 2^4, and 2^1 and Go enter the YKs as I18 to I22.

If there are no conflicts which interfere with a port's reference request, YK0 or YK1 grants the request. For a reference to section 0 (YK0) or section 1 (YK1), the YK option activates Section N Go Subsection 0 to 3 or 4 to 7 (terms R14 and R15) and one of the Section N Subsection Decodes (terms R10 to R13). Term R14 or R15 selects a group of four subsections to be active; terms R10 to R13 narrow the selection to one subsection. If the reference request is from port A, B, or C, the YK option activates Port A, B, or C Enable Section N (term R4, R6, or R8). Port D references do not generate any corresponding term. References to Section 1 (YK0) and Section 3 (YK1) are handled in the same way as those to Sections 0 and 2. R5, R7 and R9 take the place of R4, R6, and R8, and R17 to R22 take the place of R10 to R15. For a reference to either section handled by a YK option, the option generates Port A, B, C, or D Go Section N, N+2 (R28 to R31) and Port A, B, C, or D address bits 2^2 and 2^3 (R24 to R27).

If a section or subsection conflict interferes with a port's references request, the output terms explained above are not generated until the the conflict is resolved. Port A, B, C or D Subsection Conflict (terms R0 to R3) is active while the conflict is in progress. The conflict is terminated when a YK option receives the appropriate Release Subsection (terms I37 to I52). For Ports A, B, and D conflicts, the appropriate Unable to Release terms (I31 to I36) must be inactive for the reference to begin.

The YK option resolves Section conflicts. The port with the highest priority is allowed to make its reference; all other ports with reference requests to the same section incur a 1 CP conflict. YK inputs I27 to I30 determine relative port priorities. Terms I27 to I29

Terms	Port	YN0	YN1
I0 to I4	Port A	Go Section 0/2, 1/3, Address Bits 2^1, 2^5, and 2^7	Address Bits 2^2, 2^3, 2^4, 2^6, and 2^{19}
I5 to I9	Port B	Go Section 0/2, 1/3, Address Bits 2^1, 2^5, and 2^7	Address Bits 2^2, 2^3, 2^4, 2^6, and 2^{19}
I10 to I12	Port C	Go Section 0/2, 1/3, Address Bit 2^1	Address Bits 2^2, 2^3, and 2^4
I13 to I17	Port D	Go Section 0/2, 1/3, Address Bits 2^1, 2^5, and 2^7	Address Bits 2^2, 2^3, 2^4, 2^6, and 2^{19}
I18 to I19	–	Exchange Address Bits 2^5 and 2^7	Exchange Address Bits 2^6 (I19 not used)
R0 to R2	Port A	Go Section 0/2, 1/3, Address Bit 2^1	Address Bits 2^2, 2^3, and 2^4
R3 to R5	Port B	Go Section 0/2, 1/3, Address Bit 2^1	Address Bits 2^2, 2^3, and 2^4
R6 to R8	Port C	Go Section 0/2, 1/3, Address Bit 2^1	Address Bits 2^2, 2^3, and 2^4
R9 to R11	Port D	Go Section 0.2, 1/3, Address Bit 2^1	Address Bits 2^2, 2^3, and 2^4
R12 to R16	Port A	Go Read, Address Bits 2^1, 2^5, and 2^7	Address Bits 2^2, 2^3, 2^4, 2^6, and 2^{19}
R17 to R21	Port B	Go Read, Address Bits 2^1, 2^5, and 2^7	Address Bits 2^2, 2^3, 2^4, 2^6, and 2^{19}
R22 to R26	Port D	Go Read, Address Bits 2^1, 2^5, and 2^7	Address Bits 2^2, 2^3, 2^4, 2^6, and 2^{19}

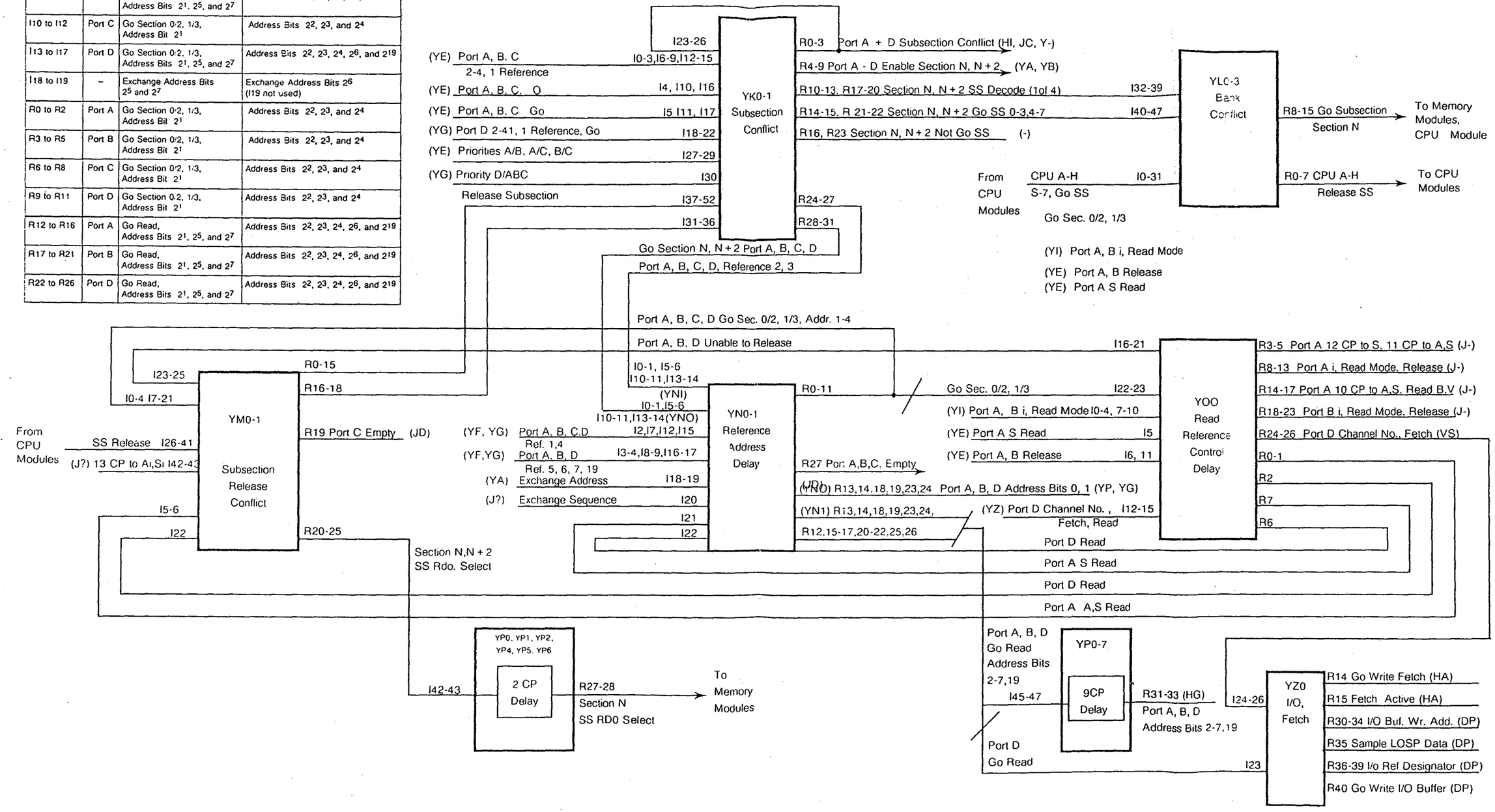


Figure 2-6. Control Paths

determine the priorities between Ports A, B, and C. Term I30 determines if Port D has the highest or lowest priority.

YL Option

The four YL options perform two independent functions. First, they generate Go Subsection terms for each memory section.. Each YL option generates Go Subsection terms for one memory section. YL0 handles section 0, YL1 section 1, YL2 section 2 and YL3 section 3. Go Subsection 0 to 7 exit the YL options as R8 to R15 and exit the module. There are two copies of each of these terms. One copy goes to the the memory modules; the other copy goes to the CPU module that handles inter-CPU conflicts for the subsection. The Go Subsection terms are generated from terms I32 to I47. Terms I40 and I44 are Go Subsection 0 to 3 and 4 to 7. If one of these terms is active, the YL looks at the Subsection Decodes (terms I32, I34, I36, and I38) to determine which Go Subsection output term to activate. YL option terms I33, I35, I37, and I39 are identical to I32, I34, I36, and I38. I41 to I43 are identical to I40. I45 to I47 are identical to I44.

Second, the YL options detect and resolve the two inter-CPU conflicts–the bank busy and the simultaneous bank conflict; each of the 32 YL options in the mainframe detects and resolves conflicts for one memory subsection. Table 2-9 show the section and subsection handled by each YL option. When a CPU makes a memory reference, it sends Go Subsection to the corresponding YL option. It also sends address bits 2^5 to 2^7 (Bank Select). The Go Subsection and address bits enter the YL option as four terms between I0 and I31. The CPU with the highest priority for the subsection (CPU A) uses terms I0 to I3, and the CPU with the next priority (CPU B) uses I4 to I7. CPU H, which has the lowest priority uses I28 to I31.

Each YL option uses terms I0 to I31 to detect and resolve inter-CPU conflicts and to generate Release Subsection for each CPU (terms R0 to R7). If a CPU reference does not encounter an inter-CPU conflict, the YL option activates Release Subsection immediately. If conflicts are encountered, Release Subsection is delayed until the conflicts are resolved.

When an inter-CPU conflict occurs, the conflicting references are not delayed on the CPU modules. All such references are passed to the Memory modules immediately and delayed there until the conflict is resolved. The Release Subsection terms from the YL options are used to inform each CPU about the status of the delays. Read "Memory Module" in this section for information on how references are delayed when there are inter-CPU conflicts.

YM Option

Two YM options determine when memory conflicts are resolved. YM0 handles Sections 0 and 2, YM1 handles Sections 1 and 3. Terms I0 to I4 are address bits 2^2, 2^3, 2^4, and 2^1 and Go Section 0/2 or 1/3 for a Port A reference that is waiting for a conflict to be resolved. I7 to I11, I12 to I17 and I18 to I21 are the corresponding terms for Ports B, C, and D. Address bit 2^0 determines which YM option is used for each reference. Terms I26 to I41 are Subsection Release terms from the YL options. The YM options use these terms to generate Release Subsection terms (R0 to R15); a Release Subsection is generated each time a port requests a memory reference, but not until all conflicts have been resolved.

Table 2-9. Inter-CPU Conflict Resolution Options

CPU	Option	Section	Subsection	CPU	Option	Section	Subsection
0	YL0	0	0	4	YL0	0	4
0	YL1	0	1	4	YL1	0	5
0	YL2	0	2	4	YL2	0	6
0	YL3	0	3	4	YL3	0	7
1	YL0	1	0	5	YL0	1	4
1	YL1	1	1	5	YL1	1	5
1	YL2	1	2	5	YL2	1	6
1	YL3	1	3	5	YL3	1	7
2	YL0	2	0	6	YL0	2	4
2	YL1	2	1	6	YL1	2	5
2	YL2	2	2	6	YL2	2	6
2	YL3	2	3	6	YL3	2	7
3	YL0	3	0	7	YL0	3	4
3	YL1	3	1	7	YL1	3	5
3	YL2	3	2	7	YL2	3	6
3	YL3	3	3	7	YL3	3	7

For memory read references, the YM options generate Subsection Readout Select 2^0 to 2^2 (R20 to R25) for the appropriate section when all conflicts are resolved. The Subsection Readout Selects select one subsection asthe source of the read data. If a conflict delays a read reference, the YM option activates Port A, B, and D Unable to Release (terms R16 to R18) until the conflict is resolved. For scalar read references, the terms 13 CP to Ai and Si (I42 and I43) indicate when the destination register is not available; if one of these terms is logic 1, the port is not released until the term returns to logic 0. For Port D references, Port D Read (term I22) differentiates between read and write references.

Port C Empty (term R19) indicates when there is no Port C reference waiting for conflicts to be resolved.

YN Option

The primary purpose of the two YN options is to delay address bits for conflict resolution and for use by the error channel. The YN options have one set of primary input terms

and two sets of output terms. Terms I0 to I17 are Port A, B, C, and D address bits and Go Section terms from the YF and YK options. These terms become active when a port clears all conflicts and begins its reference. Some of these terms (Go Section 0/2 and 1/3 and address bits 2^1 to 2^4) are delayed and exit as R0 to R11. These terms are used by the YM options to determine when conflicts can be released. Address bits from Ports A, B, and D are further delayed and exit as R12 to R26. For all Port A and B references and for Port D read references, a Go Read term is generated. These terms enter the YP options, are delayed 9 CPs, and exit to the error channel. Port D Go Read also go to the YZ option.

During Exchange Sequences, term I20 is active. This causes Port A address bits 2^5 to 2^7 (bank select) to be taken from Exchange Address terms (I18 and I19) instead of the normal Port A input terms. Port A S Read (term I21) is active when an S register is the destination for a scalar read reference. In this case, the data uses the Port B read data paths; if a memory error occurs, it is considered to be a Port B error. Port D Read (term I22) differentiates between Port D read and write references.

YO Option

The YO option delays Port A, B, and D control control terms until memory conflicts are resolved. The output terms from the YO option are used for conflict resolution, the issue control options, and the I/O section.

The YO option generates the following Port A delays for the issue control options:

- A and S Read (R0 and R1)
- 12 CP to S (R3)
- 11 CP to A and S (R4and R5)
- Instruction *i* field (R8 to R10)
- Read Mode (R11 and R12)
- Release (R13)
- 10 CP to A and S (R14 and R15)
- Read B and V (R16 and R17)

R0 and R1 are also used by the YM options to determine if a Port A conflict involving a scalar reference can be released. Port A S Read (term R7) is sent to the YN options for scalar read references to S registers. All of these terms are generated from Port A *i* (I0 to I2), Read Mode (I3 and I4), S Read (I5) and Release (I6). With the exception of Release, all Port A terms are controlled by Port A Unable to Release (terms I16 and I17). If Unable to Release remains inactive, the terms are delayed a fixed time period within the option. Each CP that Unable to Release is active further delays the terms by 1 CP. Port A Release is unaffected by Unable to Release; it is always delayed a fixed time interval.

The YO option generates the following Port B delays for the issue control options:

- Instruction *i* field (R18 to R20)
- Read Mode (R21)
- Release (R22)

They are delayed from corresponding input terms: *i* (I7 to I9), Read Mode (R10), and Release (R11). Port B Unable to Release (terms I18 to I19) controls *i* and Read Mode in the same way that I16 and I17 control Port A terms. Port B Release is delayed a fixed time interval.

Delays that are generated for Port D are Read (R6) , Channel Number 2^0 and 2^1 (R23 and R24), and Fetch (R25). All of these terms go to the YZ option and are controlled by Port D Unable to Release (terms I20 to 121).

YZ Option

A small part of the YZ option is used to generate Port D control terms. The YZ option receives Port D Go Read (term I23) from YN0 and I/O Channel Number 2^0 and 2^1 (I24 and I25) and Fetch (I26) from the YO option. All of these terms are delayed 6 CPs in the YZ option. If I23 and I26 are active, Go Write Fetch (R14) and Fetch Active (R15) are enabled. Go Write Fetch indicates that fetch data is ready to be written to an instruction buffer and is active for only 1 CP. Fetch Active remains active as long as Fetch stays active.

Terms R30 to R40 are used by the I/O section. Read Section 9 in this manual for an explanation of these terms.

YP Option

Parts of the YP options are used to delay control terms. R27 and R28 on YP0 to YP2 and on YP4 to YP7 delay Subsection Readout Selects 2^0 to 2^2 before they go to the memory modules. Each term is delayed 2 CPs. R31 to R33 delay Port A, B, and D Go Read and address bits 2^2 to $2^7$and 2^{19} for the error channel. Each term is delayed 9 CPs.

MEMORY MODULE

The CRAY Y-MP/832 computer system contains 32 identical memory modules divided into four memory sections. Each section has 8 Mwords of Central Memory. Within a section, each of the eight modules is responsible for nine bits of the 72-bit data word. The memory modules contain all Central Memory data storage chips and a large portion of the data paths, address paths, and control logic needed to transfer data between the storage chips and the eight CPUs. The memory modules also contain part of the inter-CPU conflict resolution logic.

The memory module is divided into two independent funtional areas. First-level fanouts receive address and control signals from the CPU modules and fan them out to all memory modules with a section. The main logic receives the address and control signals from the first-level fanouts and write data bits directly from the CPU modules. It also transmits read data bits directly to the CPU modules. All inputs and output to the first-level fanouts and the main logic go through the module edge connectors and the wire mat; there are no direct connectors on the module between the first-level fanouts and the main logic. Figure 2-7 shows the interconnections between the CPU modules and the memory module first-level fanouts and main logic for one memory section.

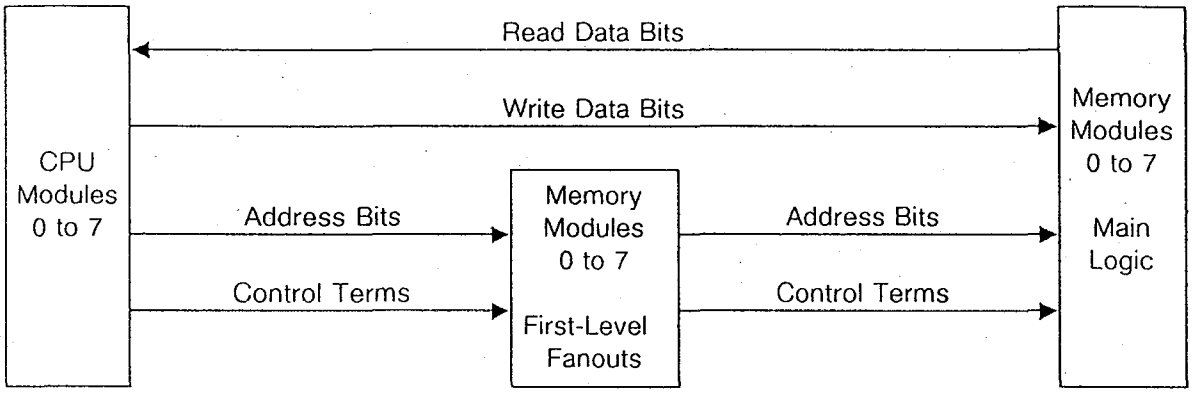


Figure 2-7. Module Interconnections

First-Level Fanouts

The first-level fanouts are used to connect address and control signals from the CPU and clock modules to the main logic of the memory modules. Although these fanouts are implemented on the memory modules, they are completely independent from the rest of the module (except the system clock). All inputs to the first level fanouts come directly from the edge connectors and all outputs leave the module via the edge connectors. There are two types of first-level fanouts:

- Buffer fanouts make eight copies of a signal. Buffer fanouts are implemented in ZA and ZS options.
- AND-gate fanouts perform the logical product of two signals. Eight AND gates are used to make eight copies of the logical product. AND-gate fanouts are implemented in ZO and ZW options.

Figure 2-8 shows how the first-level fanouts are implemented in each option type. There are multiple fanouts in the ZA, ZO, and ZW options. For simplicity, only one fanout is shown for each option type and only three of the eight circuits in each fanout are shown. All inputs and outputs leave the module via the edge connectors. The fanouts in the ZA, ZO, and ZW options use logical connector CN10 on the Z side of the module. The fanouts in the ZS options use CN20 on the Y side. All fanouts are latched, causing a 1 CP delay between input and output.

Table 2-10 and Table 2-11 show the first-level fanouts on the Y and Z sides of the module. The first column in these tables is the name of the fanout. The second and third columns show the Boolean input and output terms to the module. Columns four through seven show where the fanout is implemented. Column four is the option logical identifier. Columns five and seven are the Boolean input and output terms to the option. They are connected directly to the module input and output terms. Column six shows required force-1 inputs to the option. Many of the fanouts in the ZA and ZS options are not shown in these tables. They are used in the main logic of the memory module and are explained in the next subsection.

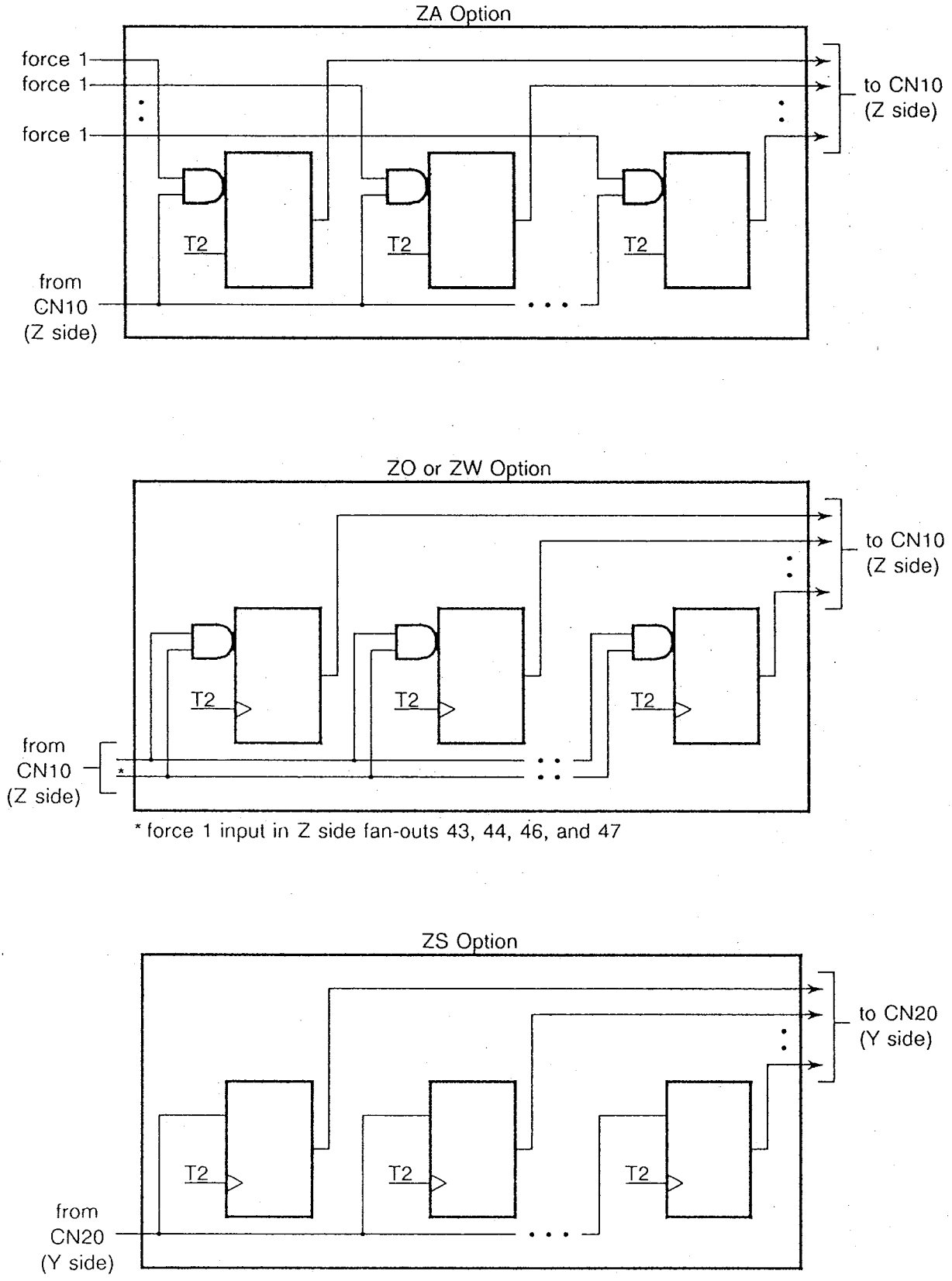


Figure 2-8. First-Level Fan-Out Block Diagram

Table 2-10. Y Side First-Level Fanouts

Name	Edge Connector CN20		Option Used			
	Input	Outputs*	Option	Input	Force 1s	Outputs
Fanout 0	I20	RX0	ZS0	I80		R10 - R17
Fanout 1	I21	RX1	ZS1	I80		R10 - R17
Fanout 2	I22	RX2	ZS20	I80		R10 - R17
Fanout 3	I23	RX3	ZS21	I80		R10 - R17

*X represents all digits 0 to 7. For example, RX0 means R0, R10...R70.

Because the first-level fanouts are used for different purposes depending on where the module is located in the chassis, information on how the first level fanouts are used is not shown in Tables 2-10 and 2-11. Refer to the *Wire Tabs* Engineering Documentation for information on that subject.

Main Logic

This subsection explains all of the logic on the memory module except the first-level fanouts and the system clock fanout. For clarity, the following notations are used:

- CPU N is an abbreviation meaning any one of the eight CPUs.
- Subsection N refers to any one of the eight subsections on the memory module.
- Bank N refers to any one of the eight banks within a subsection.

Input and Output Terms

There are several types of inputs and outputs to the main logic of the memory module :

- Control and address terms from each of the CPU modules
- Write data terms from each of the CPU modules
- Read data terms to each of the CPU modules
- Master Clear from the deadstart panel
- Static terms from the clock module

The control and address terms, Master Clear, and the static terms go through the first-level fanouts on their way to the memory module main logic. The write data terms go directly from the CPU modules to the memory module main logic. The read data terms go directly to the CPU modules. For details on the paths between modules for all these signals, refer to the *Wire Tabs* Engineering documentation.

Table 2-12 lists the main logic inputs and outputs. Each of the control, address, and data terms listed represents eight terms, one for each CPU. The write and read data bits are

Table 2-11. Z Side First-Level Fan-Outs

Name	Edge Connector CN10		Option Used			
	Input(s)	Outputs*	Option	Input(s)	Force 1s	Outputs
Fan-Out 0	I0' · I1	RX00	ZW0	I80 · I82		R30 - R37
Fan-Out 1	I0 · I1	RX01	ZW0	I81 · I83		R40 - R47
Fan-Out 2	I1' · I48	RX02	ZW1	I80 · I82		R30 - R37
Fan-Out 3	I3 · I2	RX03	ZW1	I81 · I83		R40 - R47
Fan-Out 4	I4 · I5	RX04	ZO0	I80 · I82		R30 - R37
Fan-Out 5	I1 · I6	RX05	ZO0	I81 · I83		R40 - R47
Fan-Out 6	I7	RX06	ZA1	I101	I8 - I15	R8 - R15
Fan-Out 7	I8 · I9	RX07	ZO1	I80 · I82		R30 - R37
Fan-Out 8	I8 · I10	RX08	ZO1	I81 · I83		R40 - R47
Fan-Out 9	I11 · I12	RX09	ZW2	I80 · I82		R30 - R37
Fan-Out 10	I14' · I49	RX10	ZW2	I81 · I83		R40 - R47
Fan-Out 11	I13' · I14	RX11	ZW3	I80 · I82		R30 - R37
Fan-Out 12	I13 · I14	RX12	ZW3	I81 · I83		R40 - R47
Fan-Out 13	I0 · I15	RX13	ZW10	I80 · I82		R30 - R37
Fan-Out 14	I0' · I15	RX14	ZW10	I81 · I83		R40 - R47
Fan-Out 15	I16 · I2	RX15	ZW11	I80 · I82		R30 - R37
Fan-Out 16	I15' · I50	RX16	ZW11	I81 · I83		R40 - R47
Fan-Out 17	I17 · I18	RX17	ZO10	I80 · I82		R30 - R37
Fan-Out 18	I17 · I19	RX18	ZO10	I81 · I83		R40 - R47
Fan-Out 19	I20	RX19	ZO10	I101	I8 - I15	R8 - R15
Fan-Out 20	I21 · I22	RX20	ZO11	I80 · I82		R30 - R37
Fan-Out 21	I21 · I23	RX21	ZO11	I81 · I83		R40 - R47
Fan-Out 22	I25' · I51	RX22	ZW12	I80 · I82		R30 - R37
Fan-Out 23	I24 · I12	RX23	ZW12	I81 · I83		R40 - R47
Fan-Out 24	I13 · I25	RX24	ZW13	I80 · I82		R30 - R37
Fan-Out 25	I13' · I25	RX25	ZW13	I81 · I83		R40 - R47

* X represents all digits 0 to 7. For example, RX00 means R00, R100...R700.

Table 2-11. Z Side First-Level Fan-Outs

Name	Edge Connector CN10		Option Used			
	Input(s)	Outputs*	Option	Input(s)	Force 1s	Outputs
Fan-Out 26	I0' · I26	RrX26	ZW20	I80 · I82		R30 - R37
Fan-Out 27	I0 · I26	RX27	ZW20	I81 · I83		R40 - R47
Fan-Out 28	I26' · I52	RX28	ZW21	I80 · I82		R30 - R37
Fan-Out 29	I27 · I39	RX29	ZW21	I81 · I83		R40 - R47
Fan-Out 30	I28 · I29	RX30	ZO20	I80 · I82		R30 - R37
Fan-Out 31	I28 · I30	RX31	ZO20	I81 · I83		R40 - R47
Fan-Out 32	I31	RX32	ZA21	I101	I8 - I15	R8 - R15
Fan-Out 33	I32 · I33	RX33	ZO21	I80 · I82		R30 - R37
Fan-Out 34	I32 · I34	RX34	ZO21	I81 · I83		R40 - R47
Fan-Out 35	I35 · I46	RX35	ZW22	I80 · I82		R30 - R37
Fan-Out 36	I36' · I53	RX36	ZW22	I81 · I83		R40 - R47
Fan-Out 37	I13' · I36	RX37	ZW23	I80 · I82		R30 - R37
Fan-Out 38	I13 · I36	RX38	ZW23	I81 · I83		R40 - R47
Fan-Out 39	I0 · I37	RX39	ZW30	I80 · I82		R30 - R37
Fan-Out 40	I0' · I37	RX40	ZW30	I81 · I83		R40 - R47
Fan-Out 41	I38 · I39	RX41	ZW31	I80 · I82		R30 - R37
Fan-Out 42	I37' · I54	RX42	ZW31	I81 · I83		R40 - R47
Fan-Out 43	I40	RX43	ZO30	I80	I82	R30 - R37
Fan-Out 44	I41	RX44	ZO30	I81	I83	R40 - R47
Fan-Out 45	I42	RX44	ZA30	I100	I0 - I15	R0 - R15
Fan-Out 46	I43	RX46	ZO31	I80	I82	R30 - R37
Fan-Out 47	I44	RrX47	ZO31	I81	I83	R40 - R47
Fan-Out 48	I47' · I55	RX48	ZW32	I80 · I82		R30 - R37
Fan-Out 49	I45 · I46	RX49	ZW32	I81 · I83		R40 - R47
Fan-Out 50	il3 · I47	RX50	ZW13	I80 · I82		R30 - R37
Fan-Out 51	I13' · I47	RX51	ZW13	I81 · I83		R40 - R47

* *X* represents all digits 0 to 7. For example, RX00 means R00, R100...R700.

the input and output paths for the nine data bits handled by the module. The address bits select the internal address within the memory chips.

Table 2-12. Main Logic Inputs and Outputs

Logical Edge Connector	Boolean Term(s)	Name	Type	Module Side
CNN	I0 to I15	CPU N Address Bits 0 to 15	Address Input	Z Side
CNN	I16	CPU N Chip Select	Control Input	Z Side
CNN	I17	CPU N Go Write	Control Input	Z Side
CNN	I18	CPU N Abort	Control Input	Z Side
CNN	I19	CPU N Abort (copy)	Control Input	Z Side
CNN	I20 to I28	CPU N Write Data Bits 0 to 8	Data Input	Z Side
CNN	I30 to I37	CPU N GOSS 0 to 7	Control Input	Z Side
CNN	I40 to I54 even	CPU N GOSS 0 to 7 · Bank Select $2^{2'}$	Control Input	Z Side
CNN	I41 to I 55 odd	CPU N GOSS 0 to 7 · Bank Select 2^{2}	Control Input	Z Side
CNN	I56 to I57	CPU N Bank Select 2^0 and 2^1 (copy 0)	Control Input	Z Side
CNN	I58 to I59	CPU N Bank Select 2^0 and 2^1 (copy 1)	Control Input	Z Side
CNN	I60 to I61	CPU N Bank Select 2^0 and 2^1 (copy 2)	Control Input	Z Side
CNN	I62 to I63	CPU N Bank Select 2^0 and 2^1 (copy 3)	Control Input	Z Side
CNN	I70 to I72	CPU N Subsection Read Select 2^0 to 2^2	Control Input	Y Side
CNN	R0 to R8	CPU N Read Data Bits 0 to 8	Data Output	Y Side
CN20	I0	Master Clear	Control Input	Y Side
CN20	I1	3 CP Write Enable	Static Input	Y Side
CN20	I2 to I3	Bank Busy Select 2^0 and 2^1	Static Input	Y Side

The control terms perform several functions. Go Subsection (GOSS) terms are used to activate a subsection for reading or writing to a CPU. When a GOSS is activated, the corresponding GOSS ·Bank Select $2^{2'}$ is activated if the memory reference is to banks 0 through 3 within the subsection. If the reference is to banks 4 through 7, GOSS ·Bank

Select 2^2 is activated. Simultaneously with the GOSS terms, Bank Select 2^0 to 2^2 (all four copies) select a bank in the active half-subsection. Chip Select determines which 9 of the 18 chips in the bank to use. Go Write is logic 0 for a read reference and logic 1 for a write reference. Abort and Abort (copy) prevent data from being written to or read from the memory chips in case of a memory range error. Subsection Read Select 2^0 to 2^2 are used during a read reference. They determine which subsection is the source of the read data. When Master Clear is activated, it stops all memory references in progress and prevents new references from starting.

The static terms configure the ZR and ZX options to the access and cycle times of the memory storage chips. 3 CP Write Enable selects the length of the write enable signal supplied to memory chips during a write reference. It is set to logic 1, causing the write enable signal to last 3 CPs. Bank Busy 2^0 and 2^1 determine the memory bank cycle time. Table 2-13 shows all possible cycle times. Currently, Bank Busy 2^0 is set to logic 1 and Bank Busy 2^1 is set to logic 0, causing a 5 CP cycle time.

Table 2-13. Bank Cycle Times

Bank Busy		Bank Cycle Time
2^1	2^0	
0	0	4 CPs
0	1	5 CPs*
1	0	6 CPs
1	1	7 CPs

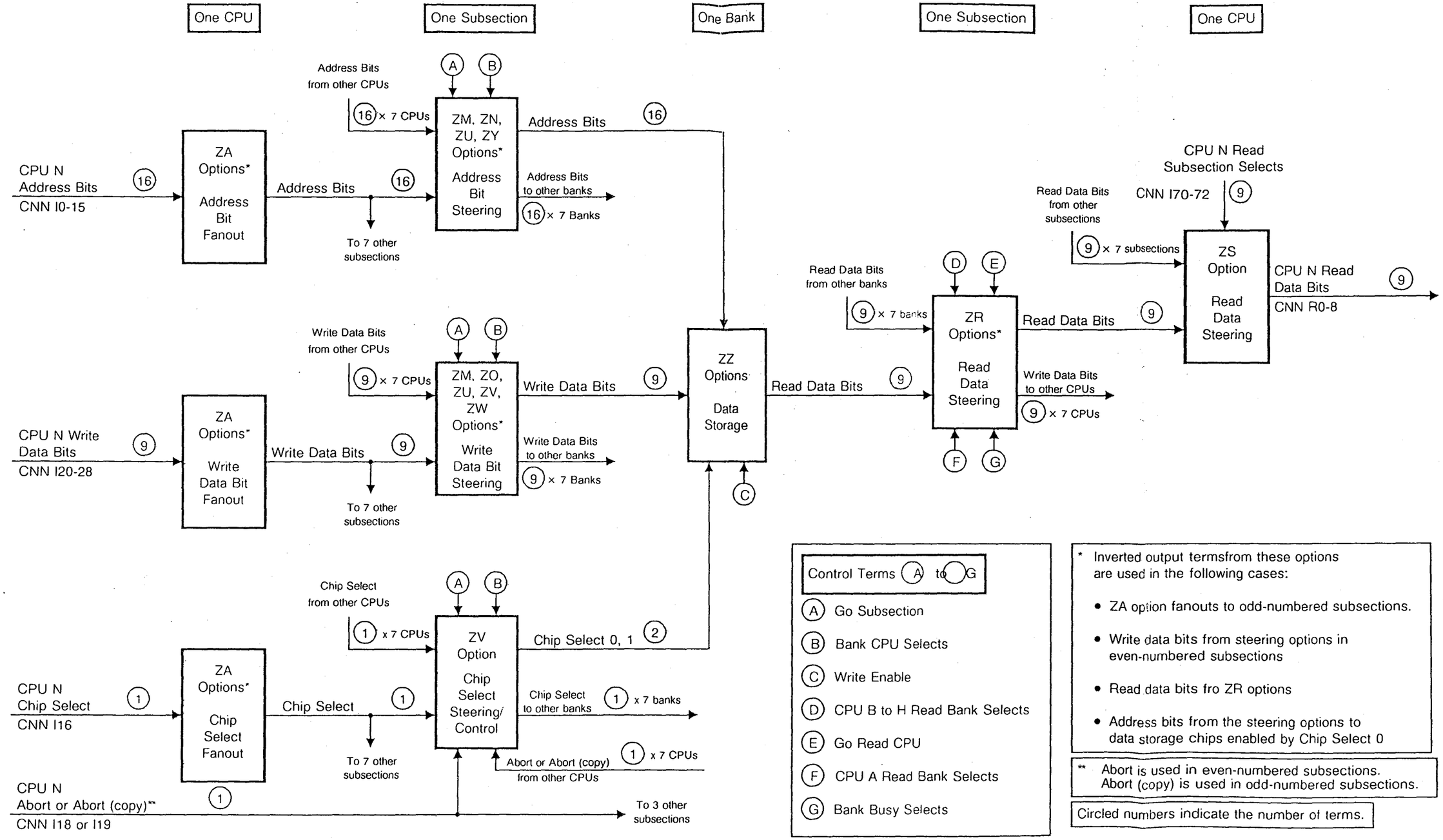
* currently used

Memory Write Reference

Figure 2-9 and Figure 2-10 are block diagrams of the main logic. Figure 2-9 shows the Address, Data, Chip Select, and Abort Paths. Figure 2-10 shows the Control and Static terms. Figure 2-11 is a timing diagram of the major signals on the memory module. Refer to these diagrams while reading this subsection.

During a memory write reference to subsection M, the address bits, write data bits, and the control terms Chip Select, Go Write, GOSS N, GOSS N-Bank Bit 2^2' (or GOSS M-Bank Bit 2^2), and Bank Selects 2^0 and 2^1 arrive at the memory module during the same CP. The address bits, write data bits, Chip Select, and Go Write are fanned out by ZA options, causing a 1 CP delay. GOSS N is fanned out by a ZM or ZU option, depending on the CPU generating the memory reference and the subsection being referenced. This fanout also causes a 1 CP delay.

After the ZA fanouts, the address bits arrive at the ZM, ZN, ZU and ZY options in each subsection, the write data bits at the ZM, ZO, ZU, ZV, and ZW options, and Chip Select at the ZV options. These signals are ignored by all the subsections except subsection N,



Control Terms (A) to (G)

- (A) Go Subsection
- (B) Bank CPU Selects
- (C) Write Enable
- (D) CPU B to H Read Bank Selects
- (E) Go Read CPU
- (F) CPU A Read Bank Selects
- (G) Bank Busy Selects

* Inverted output termsfrom these options are used in the following cases:

- ZA option fanouts to odd-numbered subsections.
- Write data bits from steering options in even-numbered subsections
- Read data bits fro ZR options
- Address bits from the steering options to data storage chips enabled by Chip Select 0

** Abort is used in even-numbered subsections. Abort (copy) is used in odd-numbered subsections.

Circled numbers indicate the number of terms.

Figure 2-9. Address, Data, Chip Select, and Abort Paths

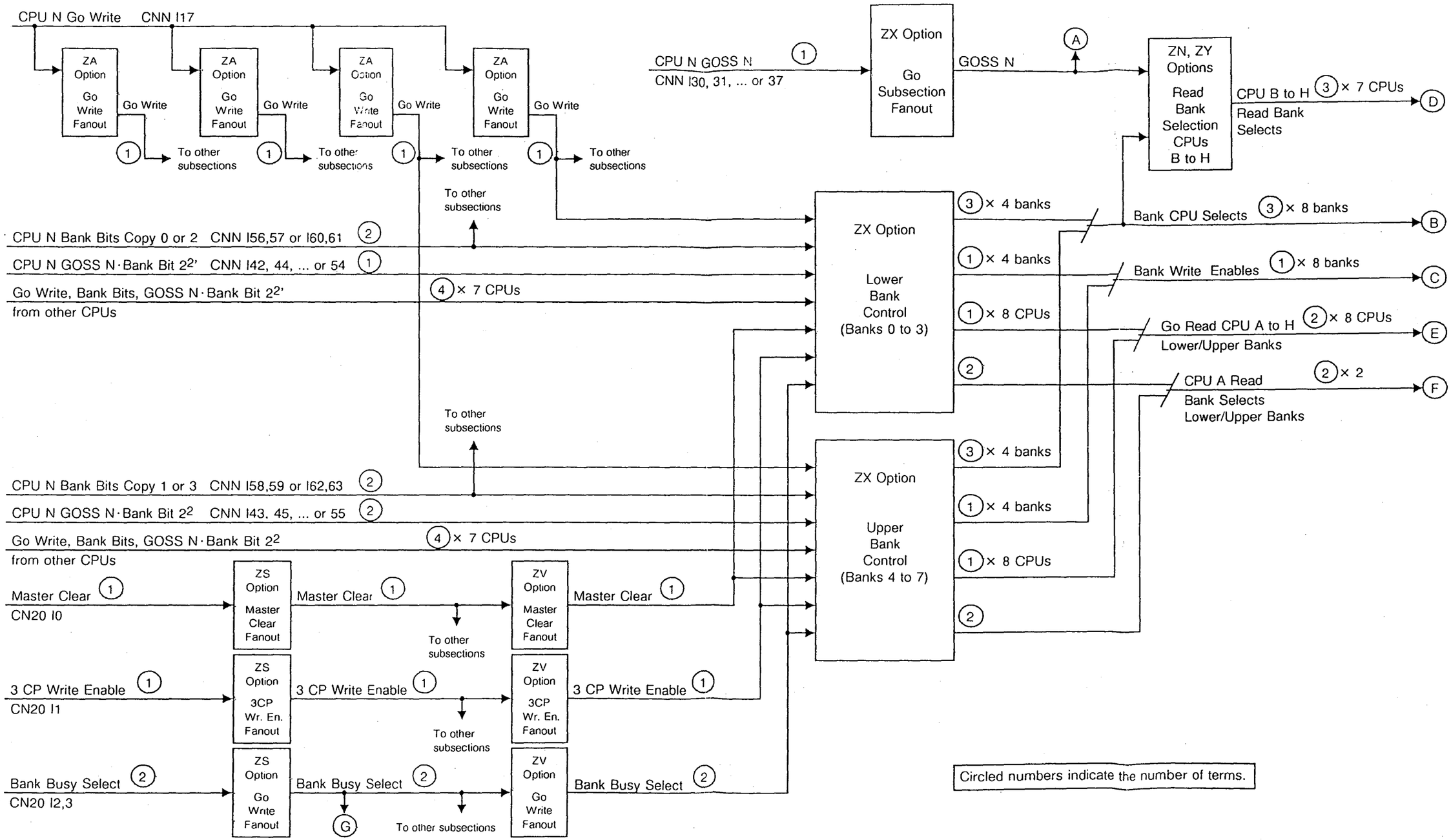


Figure 2-10. Control and Static Terms

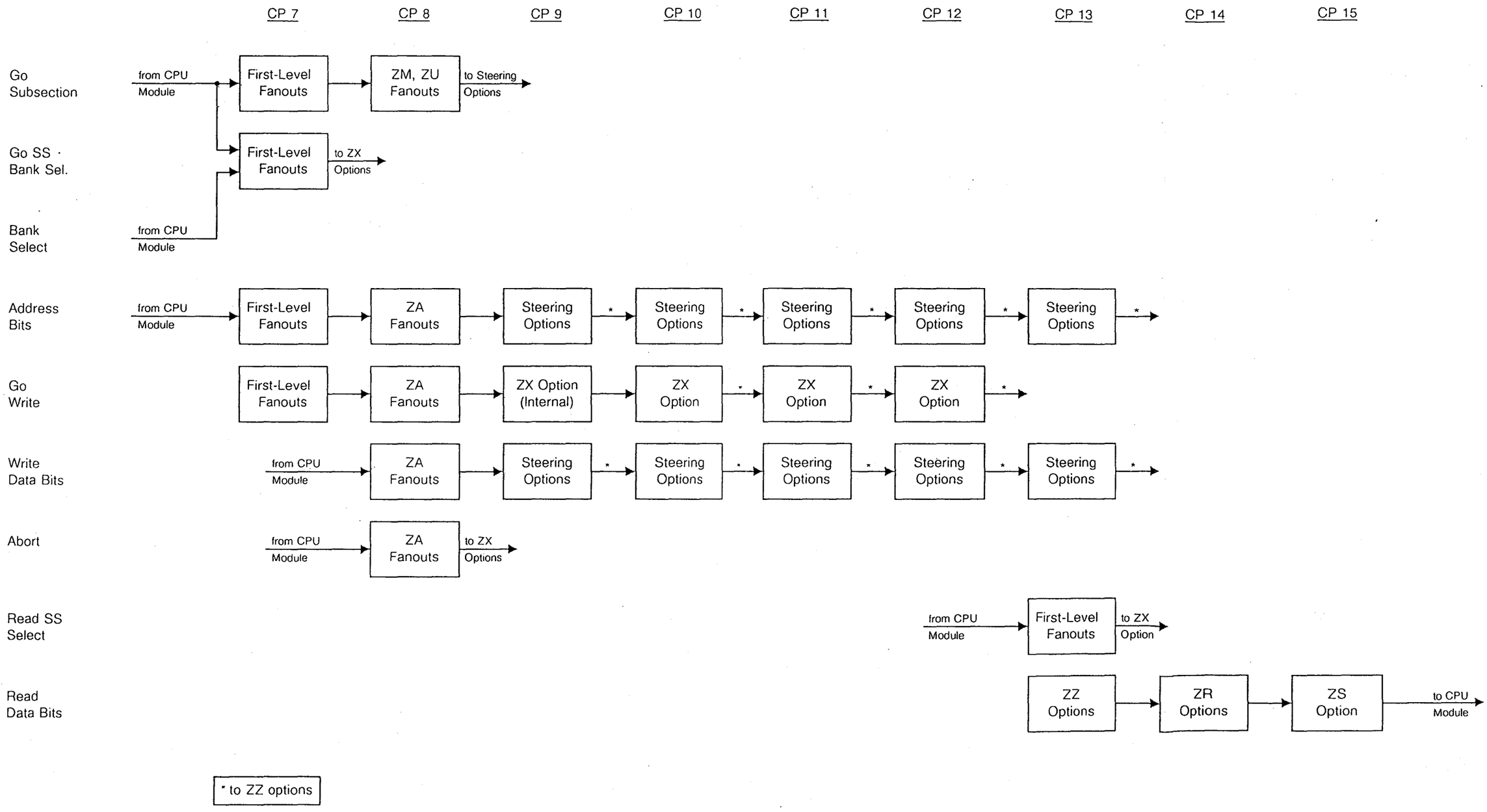


Figure 2-11. Memory Module Timing

the one that is being referenced. In subsection N, the fanned-out GOSS N causes the address bits, write data bits, and Chip Select to be latched in the steering options listed above. The terms remain latched during the entire write reference and are not released until there is another reference to the same subsection by the same CPU.

GOSS M-Bank Bit 2^2' (or GOSS M-Bank Bit 2^2) and one copy of Bank Selects 2^0 and 2^1 go directly to the ZX option which controls the bank being referenced. (Each ZX option controls four banks within a subsection, either banks 0 to 3 or banks 4 to 7.) The ZX option accepts the reference request for the bank and allows the reference to proceed when two conditions have been satisfied: the reference in progress (if any) to the same bank by another CPU must have completed and there must not be another CPU with a higher priority waiting for the same bank. As soon as these conditions exist, the ZX option sends Bank L CPU Selects 2^0 to 2^2 to the steering options where the address bits, data bits, and Chip Select are latched. This steers the address and data bits to all 18 memory chips in the bank being referenced. It also activates either Chip Select 0 or Chip Select 1, which enables nine chips in the bank. The address and data bits and Chip Select 0 or 1 remain active for 5 CPs, completing the memory reference.

During the reference, Write Enable is presented to all the memory chips in the banks by the ZX option. Write Enable becomes active 1 CP after the address and data bits and Chip Select 0 or 1 enter the memory chips. It remains active for 3 CPs. After the reference is completed, the bank is ready for another reference. The next reference can come from the same or a different CPU, but each CPU can be using or waiting for only one bank in a subsection at any time.

In case of a memory range error, the CPU which requested the memory reference stops the reference while it is in progress by sending Abort and Abort (copy). One of these terms is latched by GOSS N in the same ZV option where Chip Select is latched. This deactivates Chip Select 0 and Chip Select 1 for the duration of the reference, disabling all memory chips in the bank. In all other respects an aborted reference is identical to a normal reference.

Memory Read Reference

In many respects, a read reference is identical to a write reference. The address bits and Chip Select are fanned out, latched, and presented to the memory chips in the same way. GOSS N, GOSS N-Bank Bit 2^2' (or GOSS N-Bank Bit 2^2), Bank Selects 2^0 and 2^1, Abort, and Abort (copy) are used in the same manner. Most of the bank control logic in the ZX option serves the same function.

The main differences in a read reference are that Go Write and Write Enable are inactive and that additional control logic is necessary to move the read data bits from the memory chips to the proper CPU module. This control logic consists of two parts. Read data steering logic steers the data from the memory chips to correct CPU data path for the subsection. Read data selection logic selects the correct subsection to be the data source for the CPU requesting the reference. The first part of this control logic is slightly different depending upon which CPU requested the reference. CPU A (the CPU with the highest priority for the subsection) uses one type of control signals, and CPUs B to H use another. The second part of the read control logic is identical for all CPU paths. The following paragraphs explain both parts of the read control logic.

In the read data steering logic, read data bits are sent from the memory chips to the ZR options in the subsection 5 CPs after the memory chips are selected. For a read reference

from CPU A, the ZX option controlling the bank being referenced sends CPU A Go Read to the ZR options. Simultaneously, it sends CPU A Bank Select Bits 2^0 and 2^1 to the ZRs. By determining which of the two ZX options in the subsection sent the CPU Go Read, the ZRs decide which half of the subsection (either Banks 0 to 3 or Banks 4 to 7) is the data source for CPU A. By decoding Bank Select Bits 2^0 and 2^1, the ZRs narrow their selection to a single bank. The ZR options then steer the read data bits from the proper bank to CPU A's data path and latch the data until CPU A is ready to use it. This completes the read data steering logic for CPU A.

In the read data steering logic for a CPU other than CPU A, the read data bits are sent from the memory chips to the ZR options. Depending on the subsection and which CPU requested the reference, a ZN or ZY option sends CPU N Bank Select 2^0 to 2^2 to the ZR options in the subsection. (The ZN and ZY options generate the CPU N Bank Selects by decoding the Banks 0 to 7 CPU Selects received from the ZX options.) The ZR options use the CPU N bank selects to steer the data bits from the proper bank to CPU N's data path. CPU N Go Read is sent from the ZX option to the ZR options, causing the data to be latched until CPU N is ready for it. This completes the read data steering logic for CPUs B to H.

In the read data selection logic, the ZS options are used to gate the data from the proper ZR options to the CPU. Each ZS option is responsible for the read data bits for one CPU. When a CPU is ready to receive the read data, it sends CPU N Read Subsection Select 2^0 to 2^2 to the ZS option that it controls. This gates the data from the ZR options in the subsection being referenced to CPU N.

Data Paths

Write Data Bits 0 to 8 from each CPU are fanned out by the ZA options. The upright side of each output term goes to the even-numbered subsections and the inverted side goes to the odd-numbered subsections. After the fanouts, the write data bits are steered to the proper bank in each subsection by the ZM, ZO, ZU, ZV, and ZW options. In the even-numbered subsections, the inverted side of the steering option outputs is used. In the odd-numbered subsections, the upright side is used. The result of the data inversions is that all write data bits are inverted once, either after being fanned out to a subsection, or after being steered to a bank. Table 2-14 shows the write data bit fanouts. Table 2-15 and Table 2-16 show the steering options.

Data is stored inverted in the ZZ options. Each ZZ option consists of eight memory chips. Table 2-17 and Table 2-18 show the locations of the memory chips in each subsection based on the bank number and the state of Chip Select from the CPU module.

Read Data Bits 0 to 9 from each bank in a subsection are fanned in and steered to the proper CPU by the ZR options. After the ZR options, read data bits from each subsection are fanned in by a ZS option for each CPU. The inverted side of the ZR outputs is used by the ZS options, compensating for the inversion that is done to the write data bits. Table 2-19 shows the ZR options used in each subsection. Table 2-20 shows the ZS options used by each CPU.

Address Paths

Address Bits 0 to 15 from each CPU are fanned out by the ZA options. The upright side of each output term goes to the even-numbered subsections and the inverted side goes to

Table 2-14. Write Data Bit Fan-Outs

Write Data Bit	Option and Output Term*							
	CPU 0	CPU 1	CPU 2	CPU 3	CPU 4	CPU 5	CPU 6	CPU 7
Bit 0	ZA1 R16	ZA1 R18	ZA1 R20	ZA1 R22	ZA30 R15	ZA30 R13	ZA30 R11	ZA30 R9
Bit 1	ZA1 R17	ZA1 R19	ZA1 R21	ZA1 R23	ZA30 R14	ZA30 R12	ZA30 R10	ZA30 R8
Bit 2	ZA10 R0	ZA10 R2	ZA10 R4	ZA10 R6	ZA21 R16	ZA21 R18	ZA21 R20	ZA21 R22
Bit 3	ZA10 R1	ZA10 R3	ZA10 R5	ZA10 R7	ZA21 R17	ZA21 R19	ZA21 R21	ZA21 R23
Bit 4	ZA0 R12	ZA0 R13	ZA11 R12	ZA11 R13	ZA20 R12	ZA20 R13	ZA31 R12	ZA31 R13
Bit 5	ZA10 R22	ZA10 R20	ZA10 R18	ZA10 R16	ZA21 R6	ZA21 R4	ZA21 R2	ZA21 R0
Bit 6	ZA0 R23	ZA10 R21	ZA10 R19	ZA10 R17	ZA21 R7	ZA2 R5	ZA21 R3	ZA21 R1
Bit 7	ZA1 R6	ZA1 R4	ZA1 R2	ZA1 R0	ZA30 R22	ZA30 R20	ZA30 R18	ZA30 R16
Bit 8	ZA1 R7	ZA1 R5	ZA1 R3	ZA1 R1	ZA30 R23	ZA30 R21	ZA30 R19	ZA30 R17

* The upright side of each term goes to the even memory subsections.
The inverted side of each term goes to the odd memory subsections.

Table 2-15. Write Data Bit Steering Options
Subsections 0,2,4, and 6

Write Data Bits	Option*			
	SS 0	SS 2	SS 4	SS 6
Bits 0 and 1	ZW0	ZW10	ZW20	ZW30
Bits 2 and 3	ZW1	ZW11	ZW21	ZW31
Bit 4	ZU0	ZU10	ZU20	ZU30
Bit 5	ZM0	ZM10	ZM20	ZM30
Bit 6	ZV0	ZV10	ZV20	ZV30
Bits 7 and 8	ZO0	ZO10	ZO20	ZO30

* Only the inverted sides of the option outputs are used.

the odd-numbered subsections. After the fanouts, the address bits are steered to the proper bank in each subsection by the ZM, ZN, ZU, and ZY options. The upright side of the steering module outputs goes to the memory chips in each bank enabled by Chip

Table 2-16. Write Data Bit Steering Options
Subsections 1,3,5, and 7

Write Data Bits	Option*			
	SS 1	SS 3	SS 5	SS 7
Bits 0 and 1	ZO1	ZO11	ZO21	ZO31
Bit 2	ZV1	ZV11	ZV21	ZV31
Bit 3	ZM1	ZM11	ZM21	ZM31
Bit 4	ZU1	ZU11	ZU21	ZU31
Bits 5 and 6	ZW2	ZW12	ZW22	ZW32
Bits 7 and 8	ZW3	ZW13	ZW23	ZW33

* Only the upright side of the option outputs is used.

Select 1. The inverted side goes to the chips enabled by Chip Select 0. Table 2-21 shows the address bit fanouts. Table 2-22 show the steering options.

Control Paths

Chip Select from each CPU is fanned out by the ZA options. The upright side of each output term goes to the even-numbered subsections and the inverted side goes to the odd-numbered subsections. For Go Write from each CPU, four copies are made and each copy is fanned out by the ZA options. The upright side of each output term is used in all cases. Go Subsection terms from each CPU are fanned out by the ZM and ZU options. The upright side of each output goes to the ZU, ZW, and ZY options in the subsection. The inverted side goes to the ZM, ZN, ZO, and ZV options. Table 2-23 shows the control term fanouts.

After being fanned out, Chip Select from each CPU is decoded into Chip Select 0 and Chip Select 1 and steered to the proper bank in each subsection by the ZV options. Each ZV option also uses Abort or Abort (copy) to disable the bank in case of an aborted reference. Table 2-24 shows the Chip Select decoding and steering options and the copy of Abort used by each ZX option.

Each ZX option contains the main control logic for four memory banks. Table 2-25 shows the banks controlled by each ZX option and the copy of CPU N Go Write and Bank Bits 2^0 and 2^1 used by each option.

During read references by CPUs B to H, the ZN and ZY options generate CPU N Read Bank Select 2^0 to 2^2. Table 2-26 shows the option that generates the Read Bank Selects in each subsection for each CPU.

Table 2-17. Subsections 0,1, 4, and 5 Data Storage

Bank and Chip Select		Option* and Output Term								
		Data Bit 0	Data Bit 1	Data Bit 2	Data Bit 3	Data Bit 4	Data Bit 5	Data Bit 6	Data Bit 7	Data Bit 8
Bank 0	CS = 0	ZZX0 R2	ZZX0 R6	ZZX0 R1	ZZX1 R2	ZZX1 R6	ZZX1 R1	ZZX2 R2	ZZX2 R6	ZZX2 R1
	CS = 1	ZZX0 R0	ZZX0 R4	ZZX0 R5	ZZX1 R0	ZZX1 R4	ZZX1 R5	zzX2 R0	ZZX2 R4	ZZX2 R5
Bank 1	CS = 0	ZZX3 R1	ZZX3 R5	ZZX3 R0	ZZX4 R1	ZZX4 R5	ZZX4 R0	ZZX5 R1	ZRX5 R5	ZZX5 R0
	CS = 1	ZZX0 R3	ZZX0 R7	ZZX3 R4	ZZX1 R3	ZZX1 R7	ZZX4 R4	ZZX2 R3	ZZX2 r7	ZZX5 R4
Bank 2	CS = 0	ZZX6 R0	ZZX6 R4	ZZX3 R3	ZZX7 R0	ZZX7 R4	ZZX4 R3	ZZX8 R0	ZZX8 R4	ZZX5 R3
	CS = 1	ZZaX R2	ZZX3 R6	ZZX3 R7	ZZX4 R2	ZZX4 R6	ZZX4 R7	ZZX5 R2	ZZX5 R6	ZZX5 R7
Bank 3	CS = 0	ZZX6 R3	ZZX6 R7	ZZX6 R2	ZZX7 R3	ZZX7 R7	ZZX7 R2	ZZX8 R3	ZZX8 R7	ZZX8 R2
	CS = 1	ZZX6 R1	ZZX6 R5	ZZX6 R6	ZZX7 R1	ZZX7 R5	ZZX7 R6	ZZX8 R1	ZZX8 R5	ZZX8 R6
Bank 4	CS = 0	ZZY2 R6	ZZY2 R2	ZZY2 R5	ZZY1 R6	ZZY1 R2	ZZY1 R5	ZZY0 R6	ZZY0 R2	ZZY0 R5
	CS = 1	ZZY2 R4	ZZY2 R0	ZZY2 R1	ZZY1 R4	ZZY1 R0	ZZY1 R1	ZZY0 R4	ZZY0 R0	ZZY0 R1
Bank 5	CS = 0	ZZY5 R5	ZZY5 R1	ZZY5 R4	ZZY4 R5	ZZY4 R1	ZZY4 R4	ZZY3 R5	ZZY3 R1	ZZY3 R4
	CS = 1	ZZY2 R7	ZZY2 R3	ZZY5 R0	ZZY1 R7	ZZY1 R3	ZZY4 R0	ZZY0 R7	ZZY0 R3	ZZY3 R0
Bank 6	CS = 0	ZZY8 R4	ZZY8 R0	ZZY5 R7	ZZY7 R4	ZZY7 R0	ZZY4 R7	ZZY6 r4	ZZY6 R0	ZZY3 R7
	CS = 1	ZZY5 R6	ZZY5 R2	ZZY5 R3	ZZY4 R6	ZZY4 R2	ZZY4 R3	ZZY3 R6	ZZY3 R2	ZZY3 R3
Bank 7	CS = 0	ZZY8 R7	zZZ8 R3	ZZY8 R6	ZZY7 R7	ZZY7 R3	ZZY7 R6	ZZY6 R7	ZZY6 R3	ZZY6 R6
	CS = 1	ZZY8 R5	ZZY8 R1	ZZY8 R2	ZZY7 R5	ZZY7 R1	ZZY7 R2	ZZY6 R5	ZZY6 R1	ZZY6 R2

* *X* represents A in subsection 0, C in subsection 1, J in subsection 4, L in subsection 5.
Y represents B in subsection 0, D in subsection 1, K in subsection 4, M in subsection 5.

Table 2-18. Subsections 2, 3, 6, and 7 Data Storage

Bank and Chip Select		Option and Output Term								
		Data Bit 0	Data Bit 1	Data Bit 2	Data Bit 3	Data Bit 4	Data Bit 5	Data Bit 6	Data Bit 7	Data Bit 8
Bank 0	CS = 0	ZZX2 R6	ZZX2 R2	ZZX2 R5	ZZX1 R6	ZZX1 R2	ZZX1 R5	ZZX0 R6	ZZX0 R2	ZZX0 R5
	CS = 1	ZZX2 R4	ZZX2 R0	ZZX2 R1	ZZX1 R4	ZZX1 R0	ZZX1 R1	ZZX0 R4	ZZX0 R0	ZZX0 R1
Bank 1	CS = 0	ZZX5 R5	ZZX5 R1	ZZX5 R4	ZZX4 R5	ZZX4 R1	ZZX4 R4	ZZX3 R5	ZZX3 R1	ZZX3 R4
	CS = 1	ZZX2 R7	ZZX2 R3	ZZX5 R0	ZZX1 R7	ZZX1 R3	ZZX4 R0	ZZX0 R7	ZZX0 R3	ZZX3 R0
Bank 2	CS = 0	ZZX8 R4	ZZX8 R0	ZZX5 R7	ZZX7 R4	ZZX7 R0	ZZX4 R7	ZZX6 R4	ZZX6 R0	ZZX3 R7
	CS = 1	ZZX5 R6	ZZX5 R2	ZZX5 R3	ZZX4 R6	ZZX4 R2	ZZX4 R3	ZZX3 R6	ZZX3 R2	ZZX3 R3
Bank 3	CS = 0	ZZX8 R7	ZZX8 R3	ZZX8 R6	ZZX7 R7	ZZX7 R3	ZZX7 R6	ZZX6 R7	ZZX6 R3	ZZX6 R6
	CS = 1	ZZX8 R5	ZZX8 R1	ZZX8 R2	ZZX7 R5	ZZX7 R1	ZZX7 R2	ZZX6 R5	ZZX6 R1	ZZX6 R2
Bank 4	CS = 0	ZZY0 R2	ZZY0 R6	ZZY0 R1	ZZY1 R2	ZZY1 R6	ZZY1 R1	ZZY2 R2	ZZY2 R6	ZZY2 R1
	CS = 1	ZZY0 R0	ZZY0 R4	ZZY0 R5	ZZY1 R0	ZZY1 R4	ZZY1 R5	ZZY2 R0	ZZY2 R4	ZZfY R5
Bank 5	CS = 0	ZZY3 R1	ZZY3 R5	ZZY3 R0	ZZY4 R1	ZZY4 R5	ZZY4 R0	ZZY5 R1	ZZY5 R5	ZZY5 R0
	CS = 1	ZZY0 R3	ZZY0 R7	ZZY3 R4	ZZY1 R3	ZZY1 R7	ZZY4 R4	ZZY2 R3	ZZY2 R7	ZZY5 R4
Bank 6	CS = 0	ZZY6 R0	ZZY6 R4	ZZY3 R3	ZZY7 R0	ZZY7 R4	ZZY4 R3	ZZY8 R0	ZZY8 R4	ZZY5 R3
	CS = 1	ZZY3 R2	ZZY3 R6	ZZY3 R7	ZZY4 R2	ZZY4 R6	ZZY4 R7	ZZY5 R2	ZZY5 R6	ZZY5 R7
Bank 7	CS = 0	ZZY6 R3	ZZY6 R7	ZZY6 R2	ZZY7 R3	ZZY7 R3	zZZ7 R2	ZZY8 R3	ZZY8 R7	ZZY8 R2
	CS = 1	ZZY6 R1	ZZY6 R5	ZZY6 R6	ZZY7 R1	ZZY7 R1	ZZY7 R6	ZZY8 R1	ZZY8 R5	ZZY8 R6

* *X* represents E in subsection 2, G in subsection 3, N in subsection 6, Q in subsection 7.
Y represents F in subsection 2, J in subsection 3, P in subsection 6, R in subsection 7.

Table 2-19. Read Data Bit Steering Options

Read Data Bits	Option*							
	SS 0	SS 1	SS 2	SS 3	SS 4	SS 5	SS 6	SS 7
Bits 0 to 2	ZR0	ZR3	ZR10	ZR13	ZR20	ZR23	ZR30	ZR33
Bits 3 to 5	ZR1	ZR4	ZR11	ZR14	ZR21	ZR24	ZR31	ZR34
Bits 6 to 8	ZR2	ZR5	ZR12	ZR15	ZR22	ZR25	ZR32	ZR35

* The inverted side of each output term is used

Table 2-20. CPU Read Subsection Selects

CPU	CPU 0	CPU 1	CPU 2	CPU 3	CPU 4	CPU 5	CPU 6	CPU 7
Option	ZS0	ZS1	ZS10	ZS11	ZS20	ZS21	ZS30	ZS31

Master Clear and Static Term Fanouts

When activated, Master Clear stops all memory references in progress and clears all pending memory requests. The static term 3 CP Write Enable determines the length of the Write Enable term sent to the memory chips during a write reference. If 3 CP is at logic 0, Write Enable lasts 2 CPs. If it is at logic 1, Write Enable lasts 3 CPs. The static terms Bank Busy 2^0 and 2^1 establish the bank cycle time, which can be set between 4 and 7 CPs..

Master Clear and the static terms have two stages of fanout in the main logic. In the first stage, ZS options fan out the terms to each subsection. Within a subsection, the inverted side of Bank Busy 2^0 and 2^1 drive the three ZR options. ZV options fan out Master Clear and all the static terms to the two ZX options. The upright side of 3 CP Write Enable and Bank Busy 2^0 and 2^1 is used by the ZX options. The inverted side of Master Clear is used. Table 2-27 shows both stages of Master Clear and static term fanouts.

Inter-CPU Conflict Resolution

Each CPU module resolves memory conflicts between its own memory ports. The memory module is responsible only for resolving conflicts between ports in different CPUs. Memory references from multiple CPUs can overlap in time or occur simultaneously if each CPU is accessing a different bank. When a CPU tries to access a

Table 2-21. Address Bit Fan-Outs

Address Bit	Option and Output Term*							
	CPU 0	CPU 1	CPU 2	CPU 3	CPU 4	CPU 5	CPU 6	CPU 7
Bit 0	ZA0 R27	ZA0 R24	ZA0 R21	ZA0 R18	ZA31 R11	ZA31 R8	ZA31 R5	ZA31 R2
Bit 1	ZA0 R26	ZA0 R23	ZA0 R20	ZA0 R17	ZA31 R10	ZA31 R7	ZA31 R4	ZA31 R1
Bit 2	ZA0 R25	ZA0 R22	ZA0 R19	ZA0 R16	ZA31 R9	ZA31 R6	ZA31 R3	ZA31 R0
Bit 3	ZA11 R11	ZA11 R8	ZA11 R5	ZA11 R2	ZA20 R27	ZA20 rR4	ZA20 R21	ZA20 R18
Bit 4	ZA11 R10	ZA11 R7	ZA11 R4	ZA11 R1	ZA20 R26	ZA20 R23	ZA20 R20	ZA20 R17
Bit 5	ZA11 R9	ZA11 R6	ZA11 R3	ZA11 R0	ZA20 R25	ZA20 R22	ZA20 R19	ZA20 R16
Bit 6	ZA0 R28	ZA0 R30	ZA10 R32	ZA11 R32	ZA20 R28	ZA20 R30	ZA30 R32	ZA31 R32
Bit 7	ZA0 R29	ZA0 R31	ZA10 R33	ZA11 R33	ZA20 R29	ZA20 R31	ZA30 R33	ZA31 R33
Bit 8	ZA0 R32	ZA1 R32	ZA10 R32	ZA11 R32	ZA20 R32	ZA21 R32	ZA31 R16	ZA31 R18
Bit 9	ZA0 R33	ZA1 R33	ZA10 R33	ZA11 R33	ZA20 R33	ZA21 R33	ZA31 R17	ZA31 R19
Bit 10	ZA11 R27	ZA11 R24	ZA11 R21	ZA11 R18	ZA20 R11	ZA20 R8	ZA20 R5	ZA20 R2
Bit 11	ZA11 R26	ZA11 R23	ZA11 R20	ZA11 R17	ZA20 R10	ZA20 R7	ZA20 R4	ZA20 R1
Bit 12	ZA11 R25	ZA11 R22	ZA11 R19	ZA11 R16	ZA0 R9	ZA20 R6	ZA20 R3	ZA20 R0
Bit 13	ZA0 R11	ZA0 R8	ZA0 R5	ZA0 R2	ZA31 R20	ZA31 R23	ZA31 rR6	ZA31 R29
Bit 14	ZA0 R10	ZA0 R7	ZA0 R4	ZA0 R1	ZA31 R21	ZA31 R24	ZA31 R27	ZA31 R30
Bit 15	ZA0 R9	ZA0 R6	ZA0 R3	ZA0 R0	ZA31 R22	ZA31 R25	ZA31 R28	ZA31 R31

* The upright side of each term goes to the even memory subsections.
The inverted side of each term goes to the odd memory subsections.

bank that is busy or when two or more CPUs try to access the same bank at the same time, a conflict occurs.

Each ZX option resolves inter-CPU conflicts in the four banks that it controls. If a ZX option receives a request for a bank that is busy, the new reference is forced to wait until the reference in progress is completed. If two or more CPUs request the same bank at the same time, or if multiple CPUs are waiting for the same bank, the ZX option uses a simple priority scheme to determine which CPU gets first access to the bank: each CPU is given a fixed priority in each subsection. The CPU with the highest priority is referred to as CPU A and the one with the lowest priority as CPU H. When there is a conflict, the CPU with the highest priority is allowed to access memory first. Each of the other CPUs are forced to wait until all CPUs with higher priorities have completed their references.

Table 2-22. Address Bit Steering Options

Address Bits	Option							
	SS 0	SS 1	SS 2	SS 3	SS 4	SS 5	SS 6	SS 7
Bits 0 to 2	ZY0	ZN2	ZY10	zn12	ZY20	ZN22	ZY30	ZN32
Bits 3 to 5	ZY1	ZN3	ZY11	zn13	ZY21	ZN23	ZY31	ZN33
Bits 6 to 7	ZU0	ZM1	ZU10	ZM11	ZU20	ZM21	ZU30	ZM31
Bits 8 to 9	ZM0	ZU1	ZM10	ZU11	ZM20	ZU21	ZM30	ZU31
Bits 10 to 12	ZN0	ZY2	ZN10	ZY12	ZN20	ZY22	ZN30	ZY32
Bits 13 to 15	ZN1	ZY3	ZN11	ZY13	ZN21	ZY23	ZN31	ZY33

System Clock Fanout

Each of the four printed circuit boards in the memory module has a TO option which fans out the system clock to the other macrocell array options on the board. The memory storage chips do not require a clock signal.

Table 2-23. Control Term Fan-Outs

Control Term	Option and Output Term							
	CPU 0	CPU 1	CPU 2	CPU 3	CPU 4	CPU 5	CPU 6	CPU 7
Chip Select*	ZA0 R14	ZA0 R15	ZA11 R14	ZA11 R15	ZA20 R14	ZA20 R15	ZA31 R5	ZA31 R2
Go Write	ZA1	ZA1	ZA10	ZA10	ZA21	ZA21	ZA30	ZA30
Copy 0	R24	R28	R24	R28	R24	R28	R24	R28
Copy 1	R25	R29	R25	R29	R25	R29	R25	R29
Copy 2	R26	R30	R26	R30	R26	R30	R26	R30
Copy3	R27	R31	R27	R31	R27	R31	R27	R31
GOSS 0	ZU0 R30	ZU0 R31	ZU0 R32	ZU0 R33	ZM0 R30	ZM0 R31	ZM0 R32	ZM0 R33
GOSS 1	ZU1 R30	ZU1 R31	ZU1 R32	ZU1 R33	ZM1 R30	ZM1 R31	ZM1 R32	ZM1 R33
GOSS 2	ZM10R33	ZM10R32	ZM10R31	ZM10R30	ZU10 R33	ZU10 R32	ZU10 R31	ZU10 R30
GOSS 3	ZM11R33	ZM11R32	ZM11R31	ZM11R30	ZU11 R33	ZU11 R32	ZU11 R31	ZU11 R30
GOSS 4	ZU20R30	ZU20R31	ZU20R32	ZU20R33	ZM20R30	ZM20rR1	ZM20R32	ZM20R33
GOSS 5	ZU21R30	ZU21R31	ZU21R32	ZU21R33	ZM21R30	ZM21R31	ZM21R32	ZM21R33
GOSS 6	ZM30R33	ZM30R32	ZM30R31	ZM30R30	ZU30 R33	ZU30 rR2	ZU30 R31	ZU30 R30
GOSS 7	ZM31R33	ZM31R32	ZM31R31	ZM31R30	ZU31 R33	ZU31 R32	ZU31 R31	ZU31 R30

* The upright side of each Chip Select term goes to the even memory subsections.
The inverted side of each Chip Select term goes to the odd memory subsections.

Table 2-24. Chip Select Steering and Control Options

	Chip Selects							
	SS 0	SS 1	SS 2	SS 3	SS 4	SS 5	SS 6	SS 7
Option	ZV0	ZV1	ZV10	ZV11	ZV20	ZV21	ZV30	ZV31
Copy of CPU N Abort Used	Abort	Abort (copy)	Abort	Abort (copy)	Abort	Abort (copy)	Abort	Abort (copy)

Table 2-25. Bank Control Options

Subsection and Banks	Option	Copy Used	
		CPU N Go Write	CPU N Bank Bits 2^0 and 2^1
SS 0 Banks 0 to 3	ZX0	Copy 0	Copy 0
SS 0 Banks 4 to 7	ZX1	Copy 1	Copy 1
SS 1 Banks 0 to 3	ZX2	Copy 2	Copy 2
SS 1 Banks 4 to 7	ZX3	Copy 3	Copy 3
SS 2 Banks 0 to 3	ZX10	Copy 0	Copy 0
SS 2 Banks 4 to 7	ZX11	Copy 1	Copy 1
SS 3 Banks 0 to 3	ZX12	Copy 2	Copy 2
SS 3 Banks 4 to 7	ZX13	Copy 3	Copy 3
SS 4 Banks 0 to 3	ZX20	Copy 0	Copy 0
SS 4 Banks 4 to 7	ZX21	Copy 1	Copy 1
SS 5 Banks 0 to 3	ZX22	Copy 2	Copy 2
SS 5 Banks 4 to 7	ZX23	Copy 3	Copy 3
SS 6 Banks 0 to 3	ZX30	Copy 0	Copy 0
SS 6 Banks 4 to 7	ZX31	Copy 1	Copy 1
SS 7 Banks 0 to 3	ZX32	Copy 2	Copy 2
SS 7 Banks 4 to 7	ZX33	Copy 3	Copy 3

Table 2-26. Read Banks Select Source Options

Subsection	Option			
	CPUs 0 and 4	CPUs 1 and 5	CPUs 2 and 6	CPUs 3 and 7
SS 0	ZY0*	ZY 1	ZN0	ZN1
SS 1	ZY3*	ZY2	ZN3	ZN2
SS 2	ZN11	ZN10	ZY11	ZY10*
SS 3	ZN12	ZN13	ZY12	ZY13*
SS 4	ZY20*	ZY21	ZN20	ZN21
SS 5	ZY23*	ZY22	ZN23	ZN22
SS 6	ZN31	ZN30	ZY31	ZY30*
SS 7	ZN32	ZN33	ZY32	ZY33*

*The Read Bank Selects for CPU 0 or CPU 7 from this option are not used.

Table 2-27. Master Clear and Static Term Fan-Outs

Fan-Out	Option and Output Term(s)			
	Master Clear	3 CP Write Enable	Bank Busy	
			2^0	2^1
First-Stage	ZS30 R10 to SS 1 R11 to SS 3 R12 to SS 5 R13 to SS 7 R14 to SS 6 R15 to SS 4 R16 to SS 2 R17 to SS 0	ZS31 R10 to SS 1 R11 to SS 3 R12 to SS 5 R13 to SS 7 R14 to SS 6 R15 to SS 4 R16 to SS 2 R17 to SS 0	ZS10 R10 to SS 1 R11 to SS 3 R12 to SS 5 R13 to SS 7 R14 to SS 6 R15 to SS 4 R16 to SS 2 R17 to SS 0	ZS11 R10 to SS 1 R11 to SS 3 R12 to SS 5 R13 to SS 7 R14 to SS 6 R15 to SS 4 R16 to SS 2 R17 to SS 0
SS 0	ZV0 R30	ZV0 R31	ZV0 R32	ZV0 R33
SS 1	ZV1 R30	ZV1 R31	ZV1 R32	ZV1 R33
SS 2	ZV10 R30	ZV10 R31	ZV10 R32	ZV10 R33
SS 3	ZV11 R30	ZV11 R31	ZV11 R32	ZV11 R33
SS 4	ZV20 R30	ZV20 R31	ZV20 R32	ZV20 R33
SS 5	ZV21 R30	ZV21 R31	ZV21 R32	ZV21 R33
SS 6	ZV30 R30	ZV30 R31	ZV30 R32	ZV30 R33
SS 7	ZV31 R30	ZV31 R31	ZV31 R32	ZV31 R33

Contents

Figures

Tables

CPU DATA REGISTERS

The CPU data registers hold temporary results and data for CPU operations. These registers speed up the computational process of the CRAY Y-MP computer system, and are associated with three types of processing: address, scalar, and vector. Address (A), Scalar (S), and Vector (V) registers are the three primary sets of CPU registers. Primary registers can be directly accessed by functional units and memory. Intermediate Address (B) and Intermediate Scalar (T) registers are the intermediate sets of CPU registers. Intermediate registers are not accessible to the functional units, but act as a buffer between the primary registers and memory. This section deals with the following CPU registers:

- Address (A) registers
- Intermediate Address (B) registers
- Scalar (S) registers
- Intermediate Scalar (T) registers
- Vector registers
- Vector Length (VL) register
- Vector Mask (VM) register

ADDRESS AND INTERMEDIATE ADDRESS REGISTERS

Address registers process address information. There are two types of address registers: Address (A) registers and Intermediate Address (B) registers. The A registers are the primary address registers and B registers are the intermediate address registers used to buffer data between A registers and memory. This section explains the function and operation of the address registers and includes the following subsections:

- A Registers
  - A register options
  - A register operation
- B Registers
  - B register options
  - B register operation

A Registers

There are eight A registers designated $A0_8$ through $A7_8$. In X-mode, the A registers are 24-bits long (2^0 through 2^{23}). In Y-mode, they are 32-bits long (2^0 through 2^{31}). The A registers are used for memory referencing, and are also used as index registers for block transfers and loop control. They provide stride values for vector reads and writes as well as values for shift counts, I/O channel operations, the Vector Length (VL) register and the Exchange Address (XA) register. The A registers are used as the source of operands

for the address functional units (refer to "Address Functional Units" in this manual) and are the destination for the results. They are also the destination for the results of the POP/Leading Zero (POP/LZ) functional units.

Data can move directly between A registers and memory, B registers, Shared Address (SB) registers, the VL register, or S registers. All data transfers occur within the CPU except memory transfers and SB transfers. A0 is the only A register that can be referenced when it is not specified in one of the instruction fields on block transfer instructions (034 through 037, 176, and 177).

A Register Options

Each CPU module has four AR options (AR0 through AR3). Each option contains 8 bits of the A registers $A0_8$ through $A7_8$. AR0 contains bits 2^0 through 2^7; AR1, bits 2^8 through 2^{15}; AR2, bits 2^{16} through 2^{23}; and AR3, bits 2^{24} through 2^{32}. All AR options are internally the same; however, their function may differ depending on their location. AR0 is located on board A, AR1 on board B, AR2 on board B, and AR3 on board A. The function of the AR option is explained in the following subsection "A Register Operation".

Most of the Address Add functional unit is also contained in the AR options. Refer to "Address Add Functional Units" in this manual for more information on the Address Add functional unit.

A Register Operation

The A registers can be written to or read from. The A registers can also be bypassed. Bypass occurs when a read operation is followed by a write operation. The following subsections describe the write, read, and bypass operation for A registers. These subsections support Figure 4-1, A Register Block Diagram.

Write

Data enters the A registers through an eight-to-one write data multiplexer. The multiplexer selects one of eight signal groups to be passed on to the A registers. The incoming data and control signals (I terms) are explained in the following paragraphs.

Terms I0 through I7 come from Port A of memory. Any 10X instruction reads data from memory and writes it into an A register. The 10X instructions and descriptions are listed below:

Instruction	CAL	Unit	Description
X 10*hijkm*	A*i* *exp*,A*h*	Memory	Read from ((A*h*) + *jkm*) to A*i* (*h*≠0)
Y 10*hi*00*mn*	A*i* *exp*,A*h*	Memory	Read from ((A*h*) + *nm*) to A*i* (*h*≠0)
X 100*ijkm*	A*i* *exp*,0	Memory	Read from (*jkm*) to A*i*
Y 100*i*00*mn*	A*i* *exp*,0	Memory	Read from (*nm*) to A*i*
X10h*i*00 0	A*i* ,A*h*	Memory	Read from (A*h*) to A*i* (*h*≠0)
Y10h*i*00 00	A*i* ,A*h*	Memory	Read from (A*h*) to A*i* (*h*≠0)

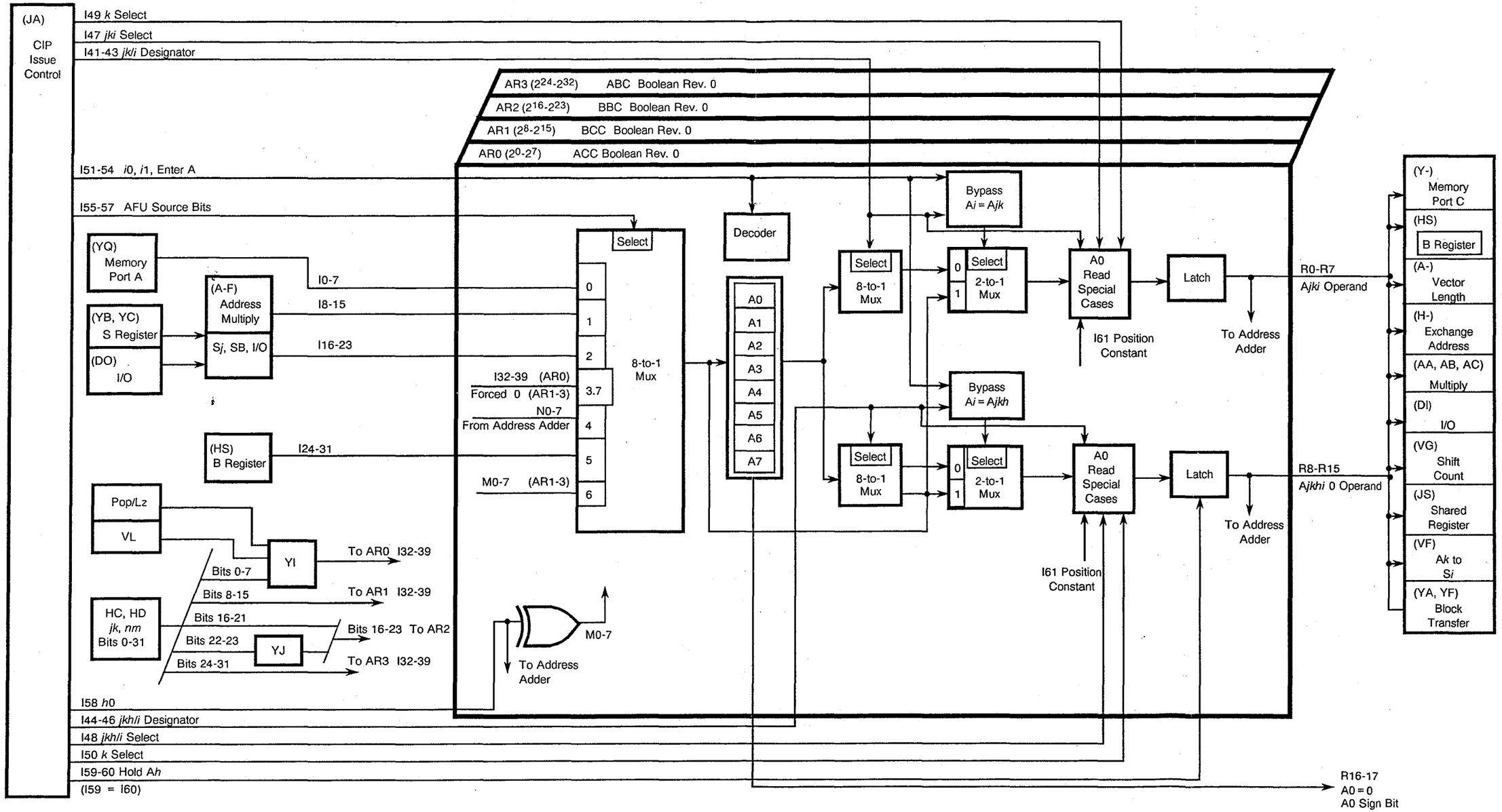


Figure 4-1. A Register Block Diagram

Terms I8 through I15 come from the Address Multiply functional unit. The 032*ijk* instruction initiates the multiply function and writes the result to an A register. The 032*ijk* instruction and its description follow.

Instruction	CAL	Unit	Description
032*ijk*	A*i* A*j**A*k*	A Int Mult	Transmit integer product of (A*j*) and (A*k*) to A*i*.

Terms I16 through I23 come from the JS option. The JS option selects one of three data groups: S register data, SB register data, or I/O channel data. The data group selected is instruction dependent. The instructions are listed below.

Instruction	CAL	Unit	Description
023*ij*0	A*i* S*j*	N/A	Transmit (S*j*) to A*i*
026*ij*7	A*i* SB*j*	N/A	Transmit (SB*j*) to A*i*
033*i*00	A*i* Cl	N/A	Transmit channel number to A*i* ($j=0$)
033*ij*0	A*i* CA,A*j*	N/A	Transmit address of channel (A*j*) to A*i* ($j\neq 0$)
033*ij*1	A*i* CE,A*j*	N/A	Transmit Error flag of channel (A*j*) to A*i* ($j\neq 0$)

Terms I24 through I31 come from the B registers. An 024 instruction generates a data transmit from a B register to an A register. The 024 instruction is listed below.

Instruction	CAL	Unit	Description
024*ijk*	A*i* B*jk*	N/A	Transmit (B*jk*) to A*i*

Terms I32 through I39 come from the YI and the YJ options. The YI and YJ options are used in memory operations. Refer to Section 2 "Memory" in this manual for more information on memory operations in the YI and YJ options. The operation of the YI and YJ options for the A registers is described in the following paragraphs.

The YI option selects data from the POP/LZ functional unit, the VL register, or bits 2^0 through 2^7 of the *mn* field, and sends it to the write data multiplexer. Because each AR option contains 8 bits, and the POP/LZ functional unit and the VL register send 7 bits of information, only AR0 is used for data coming from the POP/LZ functional unit, the VL register, and the lower 8 bits of the *nm* field.

The YJ option operates three ways depending on the mode and instructions generated. When the computer system is running in Y-mode, the YJ option sends the 32-bit constant to options AR0 through AR3. When the system is running in X-mode, either a 22-bit or 24-bit constant is sent to the AR options. When an 020 through 022 instruction is generated, the YJ option forces bits 2^{22} and 2^{23} to 0 and passes the 22-bit constant to options AR0 through AR2. When the 01*hijkm* instruction is generated, the YJ option passes the full 24-bit constant to options AR0 through AR2. The 020 through 021, 01*hijkm*, POP/LZ, and VL instructions and descriptions follow.

Instruction	CAL	Unit	Description
X 01h*ijkm*	A*h exp*	N/A	Transmit *ijkm* to A*h*
X 020*ijkm*	A*i exp*	N/A	Transmit *jkm* to A*i*
Y 020*i*00*mn*	A*i exp*	N/A	Transmit *nm* to A*i*
X 021*ijkm*	A*i exp*	N/A	Transmit one's complement of *jkm* to A*i*
Y 021*i*00*mn*	A*i exp*	N/A	Transmit one's complement of *nm* to A*i*
022*ijk*	A*i exp*	N/A	Transmit *jk* to A*i*
023*i*01	A*i* VL	N/A	Transmit (VL) to A*i*
026*ij*0	A*i* PS*j*	S Pop	Transmit population count of (S*j*) to A*i*
027*ij*0	A*i* ZS*j*	S/LZ	Transmit leading zero count of (S*j*) to A*i*

Term I40 is only used by the Address Add functional unit and is not shown in Figure 4-1. For more information on the Address Add functional unit, refer to the subsection "Address Add Functional Unit" in Section 5 of this manual.

Terms I55 through I57 are the Address Functional Unit (AFU) source bits. The AFU source bits are instruction dependent and determine which input group is routed to the A registers. Table 4-1 lists the AFU source bits, the description, and the instruction that generates them.

Table 4-1. AFU Source Code

AFU bits 2^2, 2^1, 2^0 (I57, I56, I55)	Description	Instruction
000	Memory Port A	10*h*
001	Address Multiply	032
010	S*j*, SB*j*, I/O	023*ij*0, 026*ij*7, 033
100	Address Add	030, 031
101	B*jk*	024
110	*nm*	020, 021
X11†	*jk*, VL, POP/LZ	020-022,023*i*01, 026*ij*1, 027*ij*0

† X = Don't Care

Terms I51 through I54 control Write Enable and Bypass. Refer to "Bypass" in this section for more information on bypass. I53 and I54 are the Write Enable bits. I53 is equal to *i*2' (the inverted term of 2^2 in the *i* field) ANDed with the Enter A signal. I54 is equal to *i*2 (bit 2^2 in the *i* field) ANDed with the Enter A signal. The *i* field determines the destination register. If I53 is a 1, then data is written to A registers 0 through 3 (A0 through A3). If I54 is a 1, then data is written to A registers 4 through 7 (A4 through

A7). I51 and I52 select which A register within the A register groups (A0 through A3, or A4 through A7) the data is written to. I51 is bit 2^0 of the *i* field (*i*0) and I52 is bit 2^1 of the *i* field (*i*1).

Term I58 is bit 2^0 of the *h* field (*h*0). If I58 equals 1 (data generated by an 021 or 031 instruction), the data is inverted before entering the write data multiplexer in options AR1 through AR3. If I58 equals 0 (data generated by an 020, 022, or 030 instruciton), the data goes directly to the write data multiplexer in options AR1 through AR3 at path 6.

Term I58 also determines whether the Address Add functional unit performs an add or subtract operation. For more information on an add or subtract operation refer to "Address Add Functional Unit" in Section 5 of this manual.

Read

Two readout paths allow more than one A register to be read from at the same time. The paths are designated A*jk* and A*jkh* and function in a similar manner; however, the A*jkh* readout path has a holding path controlled by terms I59 and I60 (Hold A*h*). The following paragraph gives an overview of a readout operation for the A*jk* readout path.

Data leaves an A register and enters the eight-to-one read data multiplexer. The multiplexer selects the A register that data is read from. The data selected enters a two-to-one multiplexer. This multiplexer selects either the data coming from the read data multiplexer or the data coming from the bypass path, and sends it to special case control logic. The special case control logic determines if the *j*,*k*, or *h* field of the instruction is 0. This is controlled by term I47 (term I48 for the A*jkh* readout path). If I47 is a 1, and data is read from A0, then the special case logic is used. If I47 is a 0, or if data is not read from A0, then data is passed through the control logic to the read data latch. Data leaves the A*jk* read-data latch as terms R0 through R7. The control and output terms for a read operation are explained in the following paragraphs.

Terms I41 through I46 are the control terms that select which A register data is read from. I41 through I43 select the A register that data is read from (A0 through A7), and I44 through I46 designate the *jkh*/*i* readout path.

Terms I47 and I48 are the control terms that determine whether A0 data has been selected for either the *jk*/*i* or *jkh*/*i* readout path. I47 checks the *jk*/*i* readout path and I48 checks the *jkh*/*i* readout path.

Terms I49 and I50 have different functions depending on the AR option. Table 4-2 shows the function of terms I49 and I50.

Table 4-2. I49 and I50 Control Terms

AR Option	I49	I50
AR0	*k* Select	*k* Select
AR1	X†	Forced 0
AR2	Forced 1	X-mode
AR3	Y-mode	Forced 1

† X = Don't Care

Terms I49 and I50 select either X-mode (24-bit) or Y-mode (32-bit). I49 selects Y-mode and I50 selects X-mode and occurs on AR2 or AR3. Terms I49 and I50 are also used for *k* select. *k* select is used in special cases to set A*k* equal to 1 when A0 is in the *k* field. *k* select affects only the lowest bit (2^0) and therefore, occurs only in option AR0. Table 4-3 shows the special cases for the *j*, *k*, and *h* fields.

Table 4-3. Special Case Values

Register	Value
A*i*, *i* = 0	0
A*j*, *j* = 0	0
A*k*, *k* = 0	1

Terms I59 and I60 hold A*h* data in the A*jkhi* readout latch. I59 is always equal to I60'. These terms are used to hold the base address for block transfers between memory and B, T, or V registers.

Term I61 is the position constant term. This term is a forced 0 on option AR0 and a forced 1 on options AR1 through AR3.

Terms R0 through R7 are the terms for the A*jk* data. Table 4-4 shows the destination for A register data, and includes the instructions that generate an A register read operation.

Terms R8 through R15 are the terms for the A*jkh* data. Table 4-4 shows the destination for A register data and includes the instructions that generate an A register read operation.

Table 4-4. A Register Read Destination and Instructions

Destination	Instructions
Memory	10*h*,11*h*, 12*h*, 13*h*
B register	025*ijk*
Vector Length register	00200*k*
Exchange Address register	0013*j*0
S Register	071*i*0*k*, 071*i*1*k*, 071*i*2*k*
Shared register	027*ij*7
Add	030*ijk*, 031*ijk*
Multiply	032*ijk*
I/O	0010-0012, 033*ij*0, 033*ij*1
Block Transfer	034-037, 176-177

Term R16 determines if all the bits in A0 are equal to 0 on options AR0 through AR3. If the CPU is running in X-mode, then R16 is forced to a 1 on AR3. R16 is used for branch instructions 010 through 013. These instructions are listed below.

Instruction	CAL	Unit	Description
010*ijkm*	JAZ *exp*	N/A	Jump to *ijkm* if (A0) = 0
011*ijkm*	JAN *exp*	N/A	Jump to *ijkm* if (A0) ≠ 0
012*ijkm*	JAP *exp*	N/A	Jump to *ijkm* if (A0) ≥ 0
013*ijkm*	JAM *exp*	N/A	Jump to *ijkm* if (A0) < 0

Term R17 is the sign bit. The sign bit is the most significant bit; therefore, bit 2^{23} on option AR2 is the sign bit in X-mode and bit 2^{31} on option AR3 is the sign bit in Y-mode.

Bypass

Bypass allows data to go into an A register and the readout register simultaneously. Bypass occurs when the destination register of one instruction is equal to an operand register of a following instruction (A*i* = A*j* or A*i* = A*k*). Figure 4-3 shows the timing of two instructions that cause data to bypass the A registers. Bypass occurs because instruction 030415 is trying to read data from A1 before the data in A1 is valid. Bypass saves 1 clock period (CP) by allowing data to go directly into the readout register without entering the A registers. Bypass is controlled by terms I51 through I54.

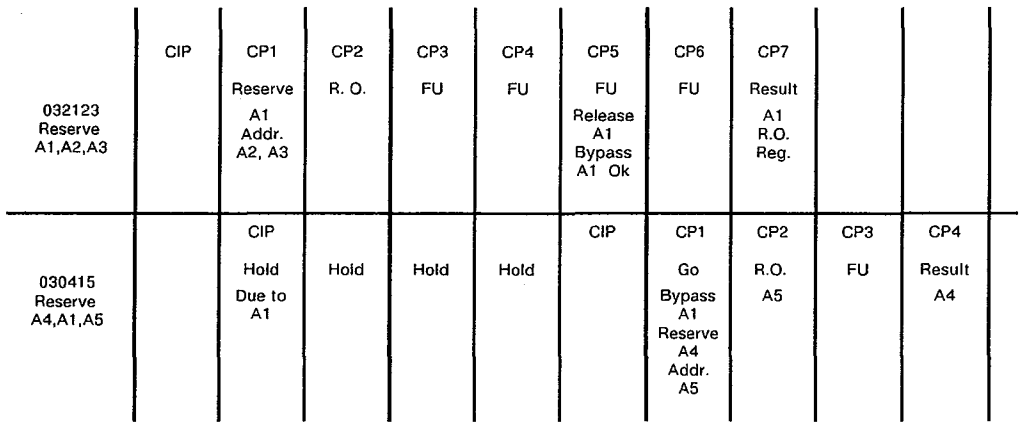


Figure 4-2. Timing Diagram of a Bypass Example

B Registers

The sixty-four 32-bit B registers are designated $B0_8$ through $B77_8$. B registers are used to buffer data between memory and A registers and operate in 32-bit mode. If X-mode is used, the upper 8 bits of the 32-bit operand are zero allowing B registers to operate in 24-bit mode. B registers can speed up block transfers to and from memory when more than 1 word is transferred. B0 is the only B register that can be referenced when it is not specified in one of the instruction fields, for example:

Instruction	CAL	Unit	Description
007*ijkm*	R exp	N/A	Return jump to *ijkm*; set B00 to (P) + 2.

B Register Options

There are 14 HS options (HS0 through HS13) used for B registers, T registers, and instruction buffers. HS0 through HS11 are used for the instruction buffers and T registers. Refer to "T Registers" in this section for more information on T registers, and "Instruction Buffers" in Section 3 of this manual for more information on instruction buffers. The HS option is also used for the performance monitor. Refer to "Performance Monitor" in Section 10 of this manual for more information on the performance monitor. Some of the circuitry in the HS option is used for instruction buffers, T registers and the performance monitor and is not used in the B registers.

Options HS12 and HS13 are used for the B registers. HS12 handles bits 0 through 7 and bits 24 through 31. HS13 handles bits 8 through 15 and bits 16 through 23. Figure 4-3 shows the HS options used for the B registers. The following subsections explain the operation of the B registers in the HS options.

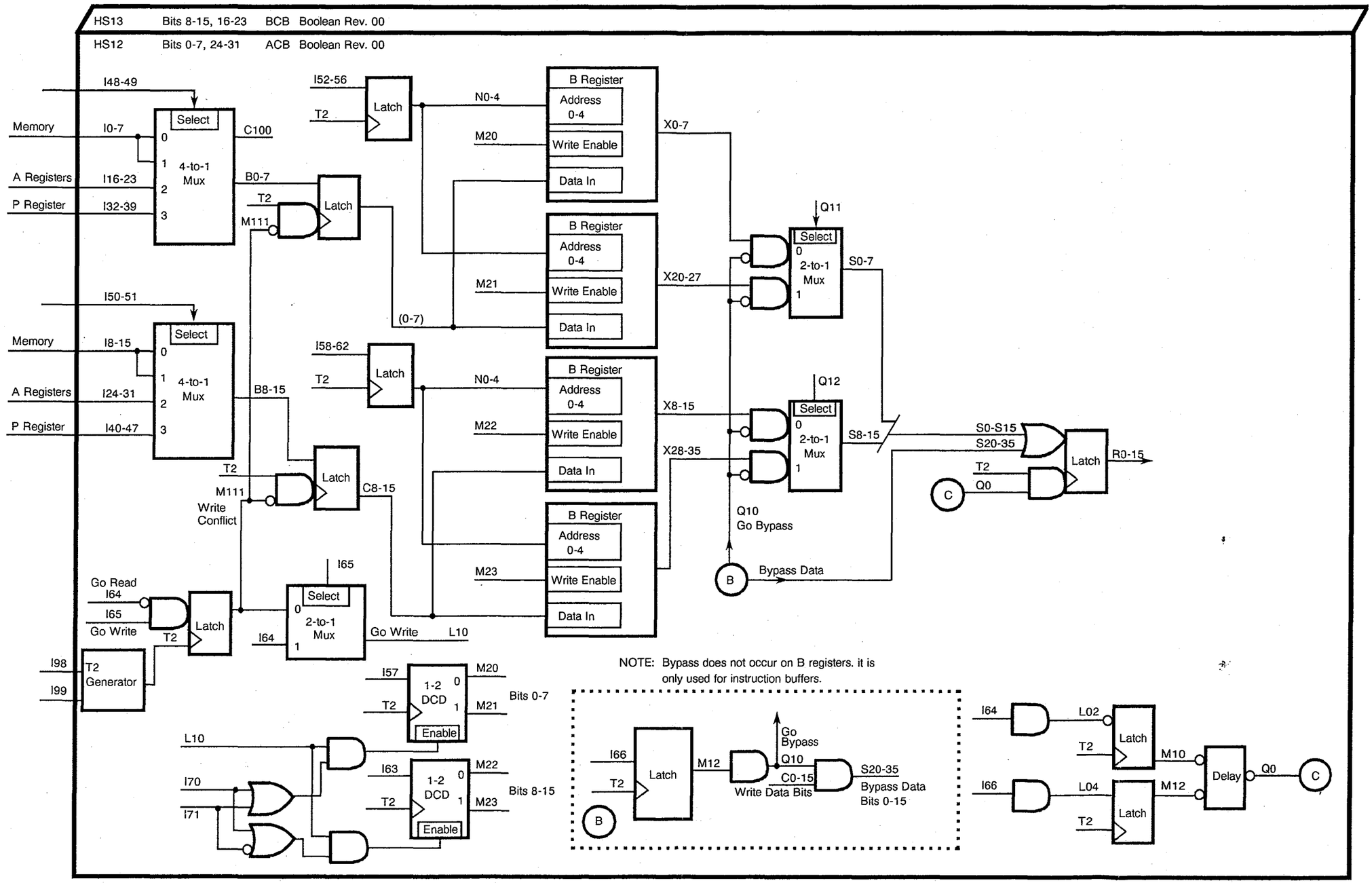


Figure 4-2. B Register Block Diagram

B Register Operation

The B registers can be written to or read from. There are two sets of 8-bit registers on the HS option. These registers work together to form the B registers. The following subsections explain the operation of the B registers for a write and read operation. These subsections support Figure 4-3, B Register Block Diagram.

Write

Data enters the B registers through a four-to-one write data multiplexer. The multiplexer selects the data to be passed on to the registers. Data enters the register at the location specified by the address terms also entering the register. The data and control terms for a write operation are explained in the following paragraphs.

Terms I0 through I47 are the data terms written to the B registers. Data can be transmitted to B registers from A registers (025 instruction), memory (035 instruction), or Program Address (P) register (007 instruction). The following instructions initiate a data transfer to the B registers.

Instruction	CAL	Unit	Description
007*ijk*	R *exp*	N/A	Return jump to *exp*; set B0 to (P) + 2
025*ijk*	B*jk* A*i*	N/A	Transmit (A*i*) to B*jk*
035*ijk*	A0 B*jk*,A*i*	Memory	Write (A*i*) words at B register *jk* into memory address ((A0) + (DBA))

Terms I48 through I51 are the control terms that select the data to enter the B registers. I48 and I49 are the select terms for I0 through I7, I16 through I23, and I32 through I39. I50 and I51 are the select terms for I8 through I15, I24 through I31, and I40 through I47. Table 4-4 shows the select bits for terms I48 through I50 and gives a description of their function.

Table 4-4. Select Bits for B Register Data

Select Bits 2^1, 2^0 (I49, I48) (I51, I50)	Description	Instruction
0X†	Memory	034-035
10	A Registers	025*ijk*
11	P Register	007*ijk*

† X = Don't Care

Terms I52 through I56 and I58 through I62 are the control terms that select the address location of the register that data is written to. I52 through I56 select the address for data

terms C0 through C7, and I58 through I62 select the address for data terms C8 through C15.

Terms I57 and I63 determine which write enable (M20, M21, M22, or M23) is selected or which register (upper or lower) data is written to.

Term I70 is the write data term for bit groups 0 through 7 and 8 through 15. If I70 is active, both bit groups are enabled.

Read

Data leaves the B registers and enters a two-to-one read data multiplexer. The selected data, controlled by address bit 5, is latched and leaves the HS option as terms R0 through R15. The following paragraphs explain the R terms and the control term that generates a read operation.

Terms R0 through R15 on option HS13 are the read data bit terms for bits 0 through 15. R0 through R7 on option HS12 handle read data bits 0 through 7, and R8 through R15 on option HS12 handle read data bits 8 through 15. The following instructions initiate a read operation from a B register.

Instruction	CAL	Unit	Description
0050*jk*	J B*jk*	N/A	Jump to (B*jk*)
024*ijk*	B*jk* A*i*	N/A	Transmit (B*jk*) to A*i*
034ijk	B*jk*,A*i* A0	Memory	Read (A*i*) words to B registers starting at B*jk* from (A0)

Term I64' controls a read operation from a B register. I64' latches the data before leaving the HS option as terms R0 through R15.

Contents

Figures

5-FUNCTIONAL UNITS

An introductory paragraph will be included here when all the functional unit subsections are completed. I included this section to give you an idea of the organization of this section. The organization is taken from the original outline.

ADDRESS FUNCTIONAL UNITS

The address functional units perform integer arithmetic on operands obtained from the Address (A) registers and deliver the result back to an A register. There are two address functional units: Address Add and Address Multiply. The following subsections describe the Address Add and the Address Multiply functional units. A description of the options used and the functional unit operation is included.

Address Add Functional Unit

The Address Add functional unit performs integer addition and subtraction on 32-bit operands. When running in X-mode, the upper 8-bits of the 32-bit operand are 0 allowing integer addition and subtraction on 24-bit operands. The 030 and 031 instructions initiate either an add or a subtract operation on integers obtained from the A registers and send the results back into an A register. The instructions and their descriptions are listed below:

Instruction	CAL	Unit	Description
030*ijk*	A*i* A*j*+A*k*	A Int Add	Transmit integer sum of (A*j*) and (A*k*) to A*i*
031*ijk*	A*i* A*j*-A*k*	A Int Add	Transmit integer difference of (A*j*) less (A*k*) to A*i*

Add and subtract operations are performed in a similar manner; that is, both operations perform addition on the operands. For an add operation, the *k* operand is added to the *j* operand. For a subtract operation, the one's complement of the *k* operand is added to the *j* operand and then a 1 is added in the lowest bit position (2^0). An example of a subtraction operation is illustrated below:

For an 031*ijk* instruction, if A*j* = 11011_2, and A*k* = 10000_2, then A*i* = 01011_2.

11011	A*j*
+ 01111	One's complement of A*k* (A*k*')
01010	Partial result
+ 1	Increment by 1 (forced Carry into bit position 2^0)
01011	Final result (A*i*)

For either an add or subtract operation, it takes 2 clock periods (CPs) from the time the *j* and *k* operands leave the A registers until the result enters the A*i* register. The following subsections explain the options used by the Address Add functional unit and give a detailed explanation of their operation.

Address Add Functional Unit Options

The AR options and the YJ option make up the Address Add functional unit. To understand the function of the options and their operation, you must understand the definitions of a Carry, Sum, and Enable. Figure 5-1 shows a Carry, a Sum, and an Enable. The definitions are listed below.

Carry Both the *j* and *k* operand bits in the same bit position equal 1. This forces a Carry to the next bit position.

Enable Both the *j* and *k* operand bits in the same bit position equal 0. A Carry entering an Enable position is stopped.

Sum Either the *j* operand or the *k* operand in the same bit position equals 1 and the other bit equals 0. A Carry entering a Sum position propagates to the next bit position.

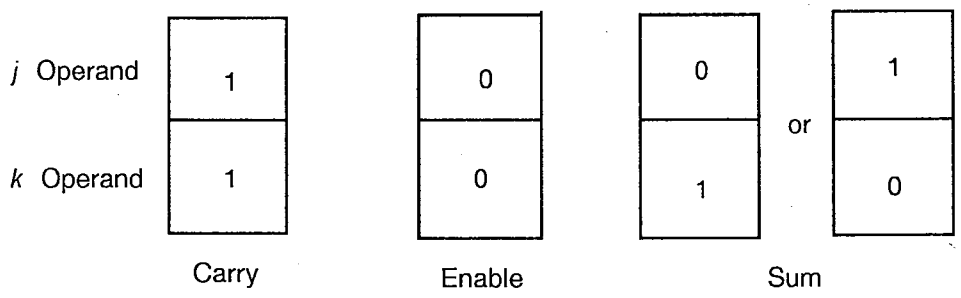


Figure 5-1. Carry, Sum, and Enable

Figure 5-2 shows an example of Carries, Sums, and Enables generated by adding the *j* and *k* operands. The arrows point to the bit positions (2^0, 2^1, and so on) that contain either Carries, Sums, or Enables.

Note: For a subtract operation, the Carries, Sums, and Enables are generated using the complemented *k* operand (*k*').

The following subsections describe the AR and YJ options used to generate Carries, Sums, and Enables for integer arithmetic performed by the Address Add functional unit.

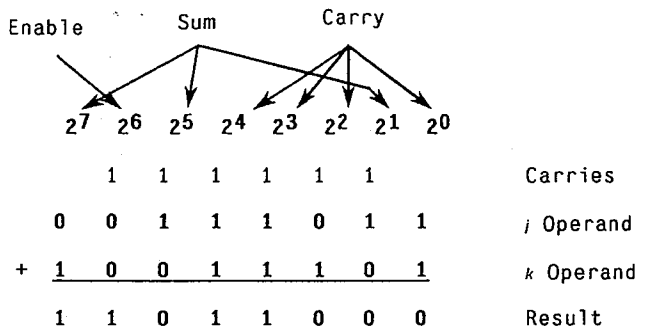


Figure 5-2. Carries, Sums, and Enables for Two 8-bit Operands

AR Option

The four AR options are designated AR0 through AR3. Each AR option handles 8 bits of the *j* and *k* operands. AR0 handles bits 2^0 through 2^7 and is located on CPU board A. AR1 handles bits 2^8 through 2^{15} and is located on CPU board B. AR2 handles bits 2^{16} through 2^{23} and is located on CPU board B. AR3 handles bits 2^{24} through 2^{31} and is located on CPU board A. The 8 bits handled on each AR option are referred to as sections. Each section is divided into two groups referred to as 4-bit groups.

Each AR option generates results for its 8-bit section by first determining Partial Sums, Section Carries, and Section Enables and second forming the final sum. The Section Carries and Section Enables leave the AR option and go to the YJ option to process the Final Section Carries. The Final Section Carries and the Partial Sum form the Final Sum on the AR option. The Partial Sum and the Final Sum do not leave the AR option. The Final Sum is stored in the A register designated by the *i* field of the instruction. For more information on A registers, refer to "A Registers" in Section 4 of this manual.

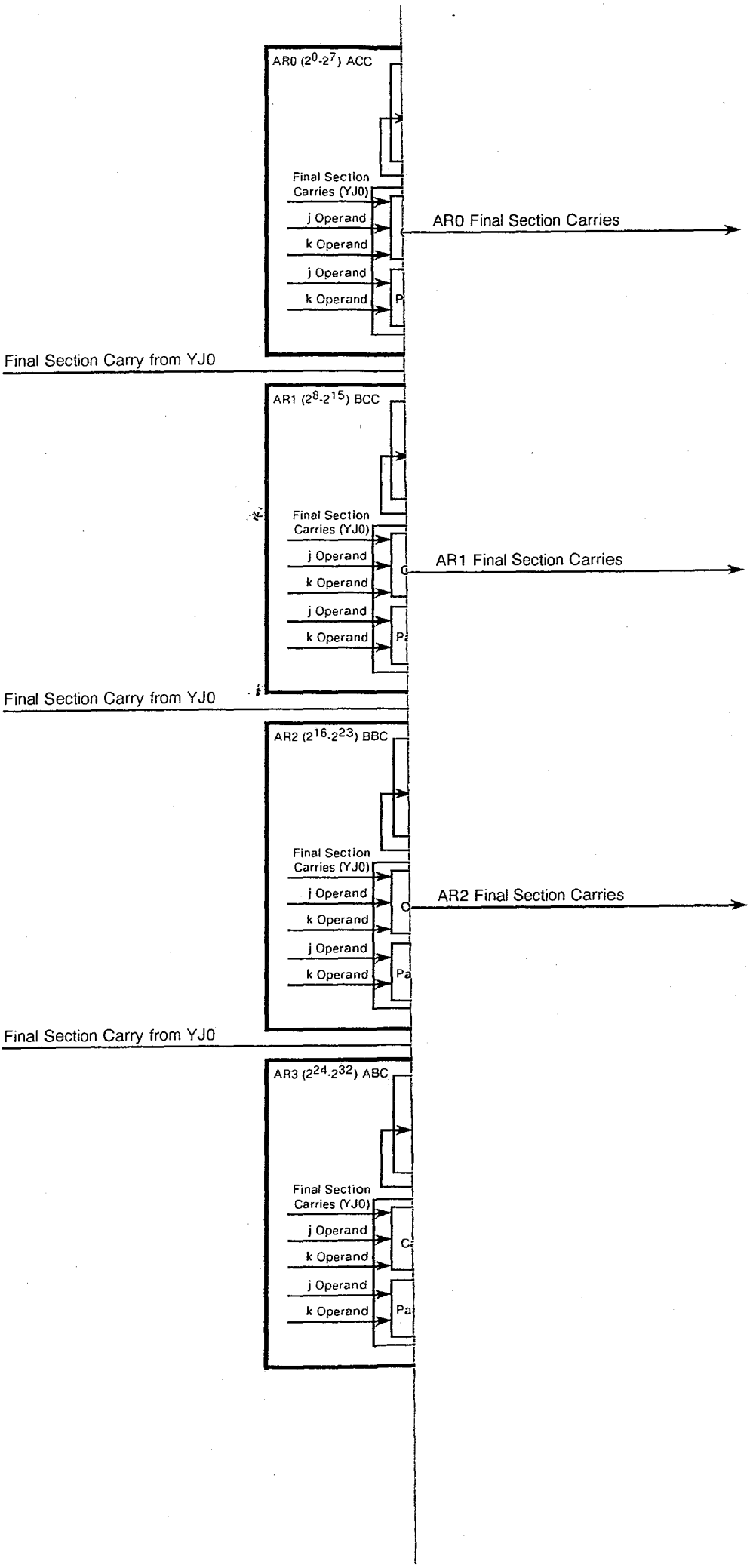
YJ Option

The one YJ option is designated YJ0 and is located on CPU Board B. YJ0 combines all the Section Carries and Section Enables from options AR0 through AR2 to generate the Final Section Carries. The Final Section Carries determine the bit position a Carry is generated to. The Final Section Carries are sent back to the AR options and combined with the Partial Sum to form the final result. For a subtraction operation, the YJ option sends a Final Section Carry to AR0. This causes a 1 to be added into bit position 2^0 to complete the operation.

Address Add Functional Unit Operation

Figure 5-3 is a block diagram of the Address Add functional unit. Refer to this diagram while reading the following paragraphs.

In each AR option, data leaves the A*jki* and A*jkhi* readout paths as the *j* and *k* operands. The operands enter the functional unit logic that performs a partial add to determine the


AR0 (2^0-2^7) ACC
Final Section Carries (YJ0)
j Operand
k Operand
j Operand
k Operand
AR0 Final Section Carries
Final Section Carry from YJ0
AR1 (2^8-2^{15}) BCC
Final Section Carries (YJ0)
j Operand
k Operand
j Operand
k Operand
AR1 Final Section Carries
Final Section Carry from YJ0
AR2 (2^{16}-2^{23}) BBC
Final Section Carries (YJ0)
j Operand
k Operand
j Operand
k Operand
AR2 Final Section Carries
Final Section Carry from YJ0
AR3 (2^{24}-2^{32}) ABC
Final Section Carries (YJ0)
j Operand
k Operand
j Operand
k Operand

Section Carries and Section Enables for the 8-bits handled on each AR option. The Section Carries and Section Enables are sent to the YJ option.

The YJ option interconnects all of the Section Carries and Section Enables from three AR options (AR0 through AR2) to obtain the Final Section Carries. The Section Carries and Section Enables from AR3 are not used. The Final Section Carries indicate whether there is a Carry into a particular 8-bit section. The YJ option sends the Final Section Carries back to the AR options.

At the same time the YJ option is processing the Final Section Carries, the AR options are computing a Partial Sum. The Partial Sum and the Final Section Carries form the final result on the AR option. The final result is stored in the A register designated by the *i* field of the instruction.

Figure 5-4 shows an example of the *j* and *k* operand bits flowing through the Address Add functional unit. The *k* operand is equal to 10101101_2 and the *j* operand is equal to 10101110_2. The operands form a final result equal to 101011011_2. The following paragraphs explain the sequence performed by the Address Add functional unit to obtain the final result. Some of the operations in the sequence occur at the same time. The figure refers to Boolean terms on the AR and YJ options. These terms are noted in the text in parenthesis, for example (G10 through 17).

During CP 1, the AR option checks the *j* and *k* operands bit-by-bit to determine the following status of the bits:

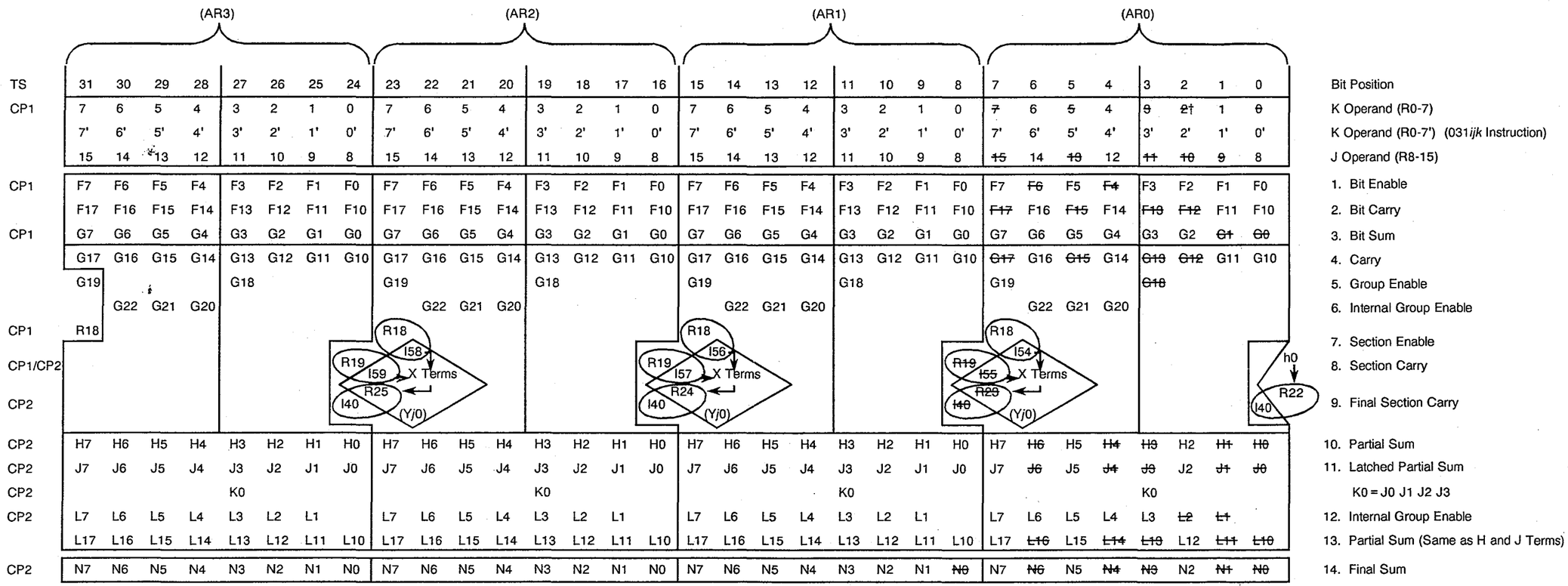
- If both the *j* and *k* operands equal 0 (Bit Enable) (F0 through F7)
- If both the *j* and *k* operands equal 1 (Bit Carry) (F10 through F17)
- If the *j* and *k* operands are different (Bit Sum) (G0 through G7)

Next, the AR options check the *j* and *k* operands to determine if there is a Carry propagating into the bit position indicated (G10 through G17). For example, if both the *j* and *k* operands in bit position 2^3 equal 1, then there is a Carry into bit position 2^4. These Carries are computed for separate 4-bit groups.

After Carries are determined for the 4-bit groups, the AR option checks for Group Enables (G18 through G19). A Group Enable sets if there are no Bit Enables within the 4-bit group; that is, if one Bit Enable within the group equals 1, then the Group Enable equals 0. If there are no Bit Enables, a Carry entering the 4-bit group is propagated out.

The AR option then checks for Internal Group Enables (G20 through G22). An Internal Group Enable sets if there are no Bit Enables at or below the bit position in the upper 4-bit group (bit positions 2^4 through 2^7, 2^{12} through 2^{15}, and so on) of an 8-bit section. The Group Enables determine the exact bit position an entering Carry propagates for each 8-bit section.

A Section Enable is an Enable within an 8-bit section (R18). A Section Enable sets if both Group Enables equal 1. If there are no Bit Enables within the 8-bit section a Carry entering the section is propagated out. For example, both terms G18 and G19 must be set for a Section Enable on option AR0. Section Enables leave the AR options as term R18 and enter the YJ option as term I54, I56, or I58. The Section Enable on option AR3 is not used.



† The strikeout indicates the term in that bit position is set.

Figure 5-4. CRAY Y-MP Address Add Flow Diagram

A Section Carry is a Carry out of the 8-bit section (R19). A Section Carry leaves option AR0, AR1, or AR2 as term R19 and enters the YJ option as terms I55, I57, or I59 respectively. Overflow is not detected; therefore, the Section Carry on option AR3 is not used.

The YJ option combines the Section Enables and the Section Carries to form the Final Section Carries (term I40 on options AR0 through AR3). If the following conditions occur, there is a Final Section Carry into an 8-bit section.

- Term I40 on AR0 is set if *h*0 equals 1 (031 instruction).
- Term I40 on AR1 is set if AR0's Section Carry is set or if AR0's Section Enable is set and *h*0 equals 1.
- Term I40 on AR2 is set if AR1's Section Carry is set, or if AR0's Section Carry and AR1's Section Enable are set, or if AR0's and AR1's Section Enable are set and *h*0 equals 1.
- Term I40 on AR3 is set if AR2's Section Carry is set, or if AR1's Section Carry and AR2's Section Enable are set, or if AR1's and AR2's Section Enable and AR0's Section Carry are set, or if AR0, AR1, and AR2's Section Enables are set and *h*0 equals 1.

At the same time the YJ option processes the Final Section Carries, the AR options compute a Partial Sum (H0 through H7). The Partial Sum is computed by an Exclusive OR with the Bit Sum and Bit Carry terms.

During CP 2, the Partial Sum is latched (J0 through J7) making the J terms equal to the H terms.

After the Partial Sum is latched, the Internal Group Enables are generated to determine the exact bit position an entering Carry propagates to; that is, all Partial Sum bits up to and including the bit position (within the 8-bit section) are toggled, so an entering carry is propagated to that bit position. Figure 5-5 is an example of a Carry from term I40 entering an 8-bit section. The Carry propagates to the first bit position containing a 0 and forces a 1 in that bit position

When the Internal Group Enables determine the exact bit position Carries propagate to, another Partial Sum (L10 through L17) is generated from the H terms and J terms. terms L10 through L17 (Partial Sum) combined with I40 (the Final Section Carries) form the final result. The final result is stored in the A register designated by the *i* field of the instruction. For more information on A registers, refer to "A registers" in Section 4 of this manual.

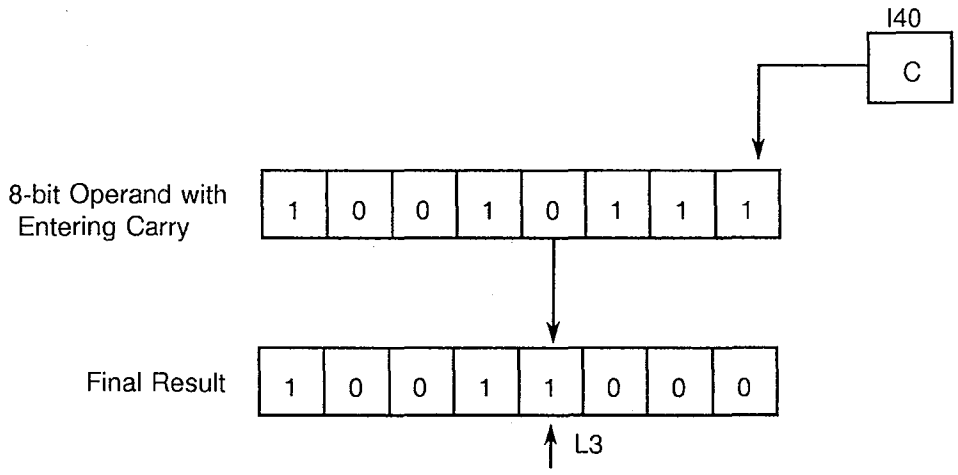


Figure 5-5. Internal Group Enable with Entering Carry

CONTENTS

FIGURES

TABLES

9 - I/O SECTION

The Input/Output (I/O) section of the CRAY Y-MP mainframe is shared by all CPUs. The I/O section contains three channel types identified by their maximum transfer rates: 6-Mbyte/s, 100-Mbyte/s, and 1000-Mbyte/s. For conciseness and consistency with engineering documentation (such as Boolean and wire tabs) this section uses following notations for the channel types:

- Low-speed (LOSP) channel refers to a 6-Mbyte/s channel.
- High-speed (HISP) channel refers to a 100-Mbyte/s channel.
- Very high-speed (VHISP) channel refers to a 1000-Mbyte/s channel.

The LOSP and HISP channels are further identified by the direction they transfer data. (The VHISP channels are bidirectional.) This section classifies them according to the way they appear from the mainframe. That is, input channels transfer data from external devices to the mainframe; output channels transfer data from the mainframe to external devices.

The LOSP channels are normally used to transfer control information between the mainframe and a CRAY I/O Subsystem (IOS). Each LOSP channel can be programmed from any CPU in the mainframe. The HISP channels are used to transmit data between the mainframe and an IOS and are programmed from the IOS. The VHISP channels transfer data between the mainframe and a CRAY SSD Solid-State Storage Device. Like the LOSP channels, the VHISP channels can be programmed from any CPU in the mainframe.

The I/O section uses Port D in each CPU to transfer data between Central Memory and the I/O channels. Table 9-1 shows each CPU and its associated I/O channels. Each of the eight CPUs provides access to Central Memory for one LOSP channel pair (that is, one input and one output channel) and one HISP channel pair. CPUs are paired to provide Central Memory access for the bidirectional VHISP channels. CPUs 0 and 1 are used by channel 1. CPUs 2 and 3 are used by channel 5, and so on

This section contains six subsections. The first three subsections describe the operation of the three channel types. The fourth subsection explains I/O channel access to Central Memory. The fifth subsection explains I/O interrupts. The last subsection describes the hardware that comprises the I/O section.

LOSP CHANNEL PAIRS

The LOSP channel pairs transmit data between Central Memory and an external device, normally a Cray IOS. Each channel pair consists of an input and an output channel. Each channel in the pair sends data in 16-bit parcels. An input channel assembles four parcels to make a 64-bit word. An output channel disassembles a 64-bit word into four

Table 9-1. CPU I/O Channel Assignments

CPU†	LOSP Channel		HISP Channel (Input and Output)	VHISP Channel (Input and Output)
	Input	Output		
0	20_8	21_8	0	1
1	22_8	23_8	1	1
2	24_8	25_8	2	5
3	26_8	27_8	3	5
4	30_8	31_8	4	11_8‡
5	32_8	33_8	5	11_8‡
6	34_8	35_8	6	15_8‡
7	36_8	37_8	7	15_8‡

† Each CPU provided memory access for the specific I/O channels shown, but each CPU can program all the 6-Mbyte/s and 1000-Mbyte/s channels.

‡ Channels 11_8 and 15_8 are reserved for future use.

parcels. Each channel provides data error detection (but not correction) by sending four parity bits with each parcel.

The following subsections describe the LOSP channel signals, CPU instructions, and error conditions.

LOSP Channel Signals

Each LOSP channel (input or output) uses the following signals to communicate between the mainframe and an external device:

- 16 data bits (numbered 0 to 15)
- 4 parity bits (numbered 0 to 3)
- 1 Ready signal
- 1 Resume signal
- 1 Disconnect signal
- 1 Device Master Clear signal (output channels only)
- 1 Deadstart signal (channel 20_8 only)
- 1 Deaddump signal (channel 21_8 only)

Figures 9-1 and 9-2 show the channel signals for the LOSP input and output channels. The following paragraphs explain the signals in detail.

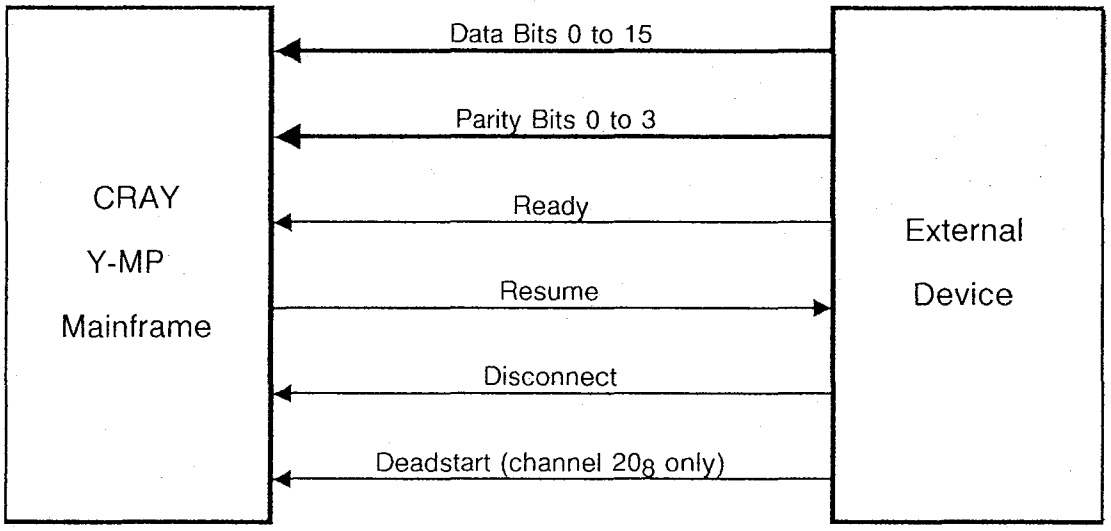


Figure 9-1. LOSP Input Channel Signals

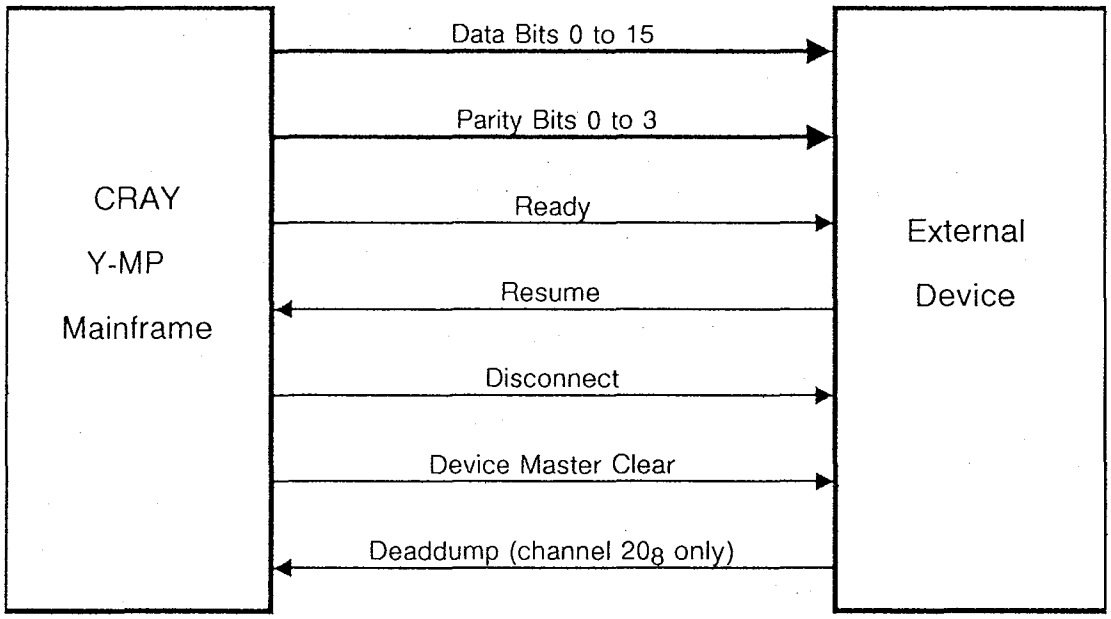


Figure 9-2. LOSP Output Channel Signals

Data is sent across the channel in 16-bit parcels. Input channels assemble four parcels into one Central Memory word. Output channels disassemble one Central Memory word into four parcels. Parcel 0 (data bits 2^{48} to 2^{63}) is sent across the channel first, followed by Parcels 1, 2, and 3.

Error detection (but not correction) is provided by transmitting four parity bits with each parcel. Each parity bit is assigned to four bits in the parcel. Odd parity is used. Table 9-2 shows the LOSP channel data and parity bits. The first column shows the parcel numbers in the order they are sent across the channel. The second column shows the data bits in each parcel. The third through sixth columns show the data bits assigned to each parity bit.

Table 9-2. LOSP Channel Data and Parity Bits

Parcel	Data Bits 0 to 15	Parity Bits			
		0	1	2	3
0	2^{48} to 2^{63}	2^{48} to 2^{51}	2^{52} to 2^{55}	2^{56} to 2^{59}	2^{60} to 2^{63}
1	2^{32} to 2^{47}	2^{32} to 2^{35}	2^{36} to 2^{39}	2^{40} to 2^{43}	2^{47} to 2^{47}
2	2^{16} to 2^{23}	2^{16} to 2^{19}	2^{20} to 2^{23}	2^{24} to 2^{27}	2^{28} to 2^{31}
3	2^{0} to 2^{15}	2^{0} to 2^{3}	2^{4} to 2^{7}	2^{8} to 2^{11}	2^{12} to 2^{15}

Control for each channel is provided by three signals: Ready, Resume, and Disconnect. Each of these signals is 50 ± 10 ns wide. On an input channel, the external device transmits Ready and Disconnect signals to the mainframe and the mainframe transmits Resume to the external device. The normal control sequence for an input channel is as follows:

1. The external device places a parcel of data and four check bits on the channel. It then activates Ready to inform the mainframe that data is waiting on the channel.

2. The mainframe reads the data and parity bits from the channel and checks for parity errors. It then activates Resume to indicate that it has received the data.

3. Steps 1 and 2 are repeated until all data has been transferred across the channel.

4. The external device activates Disconnect to indicate that the transfer is complete.

An output channel uses the same control signals, but their directions are reversed. That is, the mainframe transmits Ready and Disconnect signals to the external device and the external device transmits Resume to the mainframe. Each output channel also sends an additional control signal, Master Clear, to the external device. Master Clear can be set or cleared under program control from any CPU.

Each LOSP channel channel has two registers that can be loaded from any CPU. The Channel Address (CA) register contains the next Central Memory address to be transferred. When an I/O transfer begins, the CA register contains the address of the

the first word. After the first word is transferred, the CA register increments. The next word is transferred and the CA register again increments. This process continues until all words have been transferred.

The Channel Limit (CL) register determines the ending address in Central Memory. An I/O transfer is completed when the contents of the CA register is equal to the contents of the CL register. The word at address CL is not transferred; address CL - 1 contains the last word transferred.

The Deadstart and Deaddump signals (channels 20_8 and 21_8, respectively) permit an external device to initiate a LOSP input or output transfer. When one of these signals is received, the channel's CA register is forced to 0 and the CL register is forced to all 1s (that is, to an address beyond the physical range of Central Memory). The Deadstart signal causes channel 20_8 to receive data until it receives the Disconnect signal; this is the normal way the mainframe is initialized. (The Deadstart signal also generate I/O Master Clear in the mainframe). The Deaddump signal causes channel 21_8 to send data until it stops receiving the Ready signal. Deaddump is normally used only for maintenance.

LOSP Channel Programming

Data transfers through a LOSP channel can be initiated by any CPU in Monitor Mode. Once a transfer is initiated, the CPU does not need to take any further action. The transfer operates as a background activity and the CPU may resume other processing. When the transfer is completed, the channel sets an interrupt request flag in a CPU. The CPU that receives the interrupt request is not necessarily the same CPU that initiated the transfer. Refer to "I/O Interrupts" later in this section for more information.

Table 9-3 shows all the instructions that are applicable to the LOSP channels. Instructions 0010*jk* through 0012*j*1 perform channel control and can be executed only by a CPU in Monitor Mode. There is no hardware interlock between CPUs; the programmer must ensure that two CPUs do not try to control the same channel at the same time. Instructions 033*i*00 through 033*ij*1 transmit I/O status information to register A*i*. These instructions are not limited to Monitor Mode and can be executed by any number of CPUs simultaneously.

To initiate a transfer across a LOSP channel, use the following sequence of instructions:

Machine Instruction	CAL	Comment
0011*jk*	CL,A*j* A*k*	Set CL register to last memory address + 1
0010*jk*	CA,A*j* A*k*	Set CA register to first memory address and begin I/O sequence.

Table 9-3. LOSP Channel Instructions

Machine Instruction	CAL	Description
0010*jk*†	CA,A*j* A*k*	Set channel (A*j*) CA register to (A*k*) and begin I/O sequence.
0011*jk*†	CL,A*j* A*k*	Set channel (A*j*) CL register to (A*k*).
0012*j*0†	CI,A*j*	Clear channel (A*j*) interrupt and error flags. Clear device Master Clear (output channel).
0012*j*1†	MC,A*j*	Clear channel (A*j*) interrupt and error flags. Set device Master Clear (output channel). Clear device ready-held (input channel).
033*i*00	A*i* CI	Transmit interrupting channel number to A*i*.
033*ij*0	A*i* CA,A*j*	Transmit (CA) of channel (A*j*) to A*i*.
033*ij*1	A*i* CE,A*j*	Transmit error flag of channel (A*j*) to A*i*.

† Privileged to Monitor mode

This sequence starts the I/O transfer and increments the CA register after each data word is transferred to or from the mainframe. On an output channel, the transfer stops when CA = CL. On an input channel, the transfer stops when CA = CL or when the mainframe receives the Disconnect signal, which ever comes first.

There are two important characteristics of the LOSP channels that must be kept in mind when programming an I/O transfer. First, the CL register must be loaded before the CA register; the transfer begins when the CA register is loaded, regardless of the contents of the CL register. Second, the CA register must be loaded with a value less than the contents of the CL register. If the CA register is loaded with a value equal to or greater than the CL register, unpredictable results will occur.

Two auxiliary operations can also be programmed to a LOSP channel. They are most commonly used to initialize a channel after a deadstart or resynchronize a channel after an error. The first operation involves the Ready signal received by an input channel. When a Ready signal is received, it is held (latched) until the channel is ready to receive the data. Since a Ready signal may be received when the channel is not active, it is sometimes useful to clear the ready-held condition. Instruction 0012*j*1 performs this function.

The second auxiliary operation performs a Master Clear of a external device through an output channel. The external Master Clear sequence is as follows:

Machine Instruction	CAL	Comment
0012*j*0	CI,A*j*	Clear input channel to ensure external activity on the channel pair has stopped.
0012*j*1	MC,A*j*	Clear output channel to ensure external activity on the channel pair has stopped. Set device Master Clear.
–	–	Delay. The required delay time is determined by the external device.
0012*j*0	MC,A*j*	Clear output channel. Clear device Master Clear.
–	–	Delay. The required delay time is determined by the external device.

The 0012*j*0 and 0012*j*1 instructions used in the auxiliary functions also clear the channel interrupt and error flags. Refer to "I/O Interrupts" later in this section for more information.

LOSP Channel Errors

The LOSP channels detect two types of errors. Input channels detect parity errors. Output channels detect unexpected Resume signals received from external devices. Either type of channel error sets the Channel Error flag, which can can be read using instruction 033*ij*1. This instruction transfers the Error flag to bit 2^0 of register A*i* and clears all other bits of the register .

When an input channel detects a parity error, it sets an internal parity error flag but does not interrupt the data transfer. All data received after the parity error is zeroed before being written to Central Memory. When the transfer completes, the parity error flag sets the channel error flag. There is no way to inform the external device of the parity error.

When an output channel receives an unexpected Resume signal (that is, a Resume signal that is received without the channel sending a Ready signal), the channel error and interrupt flags are set. Refer to "I/O Interrupts" later in this section for more information.

In addition to errors detected by the LOSP channels, errors can occur when transferring data from Central Memory to a LOSP output channel. The mainframe handles these errors like any other memory error. Refer to "Error Detection and Correction" in Section 2 of this manual for more information.

HISP CHANNELS

The HISP channels transfer data between Central Memory and an external device, normally a Cray IOS. Each channel uses 64-bit data words with eight check bits for error detection and correction, identical to the SECDED scheme used by Central Memory. (Refer to "Error Detection and Correction" in Section 2 of this manual for more information.)

The HISP channels are under complete control of the external devices. The external devices initiate all data transfers, determine the number of words to be transferred, and determine the starting address in Central Memory. Their operation is completely transparent to the CPUs in the mainframe; there are no CPU instructions to control or monitor channel operations and the channels do not generate CPU interrupt requests. Channel errors can be detected at either end, but are reported to the external device.

Data is usually transmitted across HISP channels in 16-word blocks. However, the last block of a transfer may contain 1 to 16 words. The mainframe contains two 16-word data buffers for each HISP channel. The buffers allow an entire 16-word block to be transferred across a channel without delays caused by memory conflicts.

The input channel data buffers operate as follows: When a data transfer begins, only one buffer, Buffer A, is used. Buffer A begins receiving data from the channel. When Buffer A is full, the second buffer, Buffer B, begins to receive the channel data, and Buffer A begins to send its data to Central Memory. When Buffer B is full and Buffer A is empty, they exchange roles; Buffer A receives the channel data and Buffer B sends its data to Central Memory. The buffers continue to exchange roles, with one buffer receiving channel data and the other buffer sending data to Central Memory, until the data transfer is complete.

The output channel buffers operate similarly to the input channel buffers, except that data is transferred in the opposite direction. One buffer receives data from Central Memory while the other buffer transmits its data across the channel. The buffers exchange roles for each 16-word block.

HISP Channel Signals

The signals used by each HISP channel (input or output) can be divided into two categories:

- Initialization signals:
  - 1 Clear Channel signal
  - 1 Transmit Address signal
  - 1 Address Ready signal
  - 16 address channel bits (numbered 0 to 15)
  - 4 address parity bits (numbered 0 to 3)
  - 1 Address Error signal

- Data transfer signals:
  - 1 Transmit Data signal
  - 1 Data Ready signal
  - 64 data bits (numbered 2^0 to 2^{63})
  - 8 data check bits (numbered 0 to 7)
  - 1 Last Word flag
  - 1 Uncorrectable Error signal
  - 1 Disable Error Correction signal (input channels only)

Figures 9-3 and 9-4 show the channel signals for the HISP input and output channels. The following subsections explain the signals in detail.

Initialization Signals

The external device prepares the mainframe for a data transfer by sending Clear Channel. This signal stops any transfer already in progress on the channel and prepares the mainframe to receive address information. The mainframe then sends Address Ready to indicate that it is ready to receive address channel information.

Address channel information consists of two parts: 32 Starting Address (SA) bits and 14 Transfer Word Count (TWC) bits. The SA bits indicate the starting address in Central Memory that the data is to be transferred to or from. The TWC bits indicate the number of words to be transferred. The external device generates the SA and TWC bits and combines them into three 16-bit parcels, with four parity bits for each parcel (odd parity). The parcels are transmitted to the mainframe as address bits 0 to 15 and the parity bits are transmitted as address parity bits 0 to 3. Table 9-4 shows the how the SA and TWC bits are combined into parcels and how the parity bits are arranged.

Each time the external device places a parcel of information (and four parity bits) on the address channel, it indicates that the information is available by activating Address Ready. It then deactivates Address Ready until the next parcel is ready. The mainframe activates Address Error if it detects an error in the transmission of address information. Address errors are discussed in detail in the next subsection.

Data Transfer Signals

A data transfer begins after all three parcels have been transmitted across the address channel. For each block of data, the receiving device (the mainframe for input channels, the external device for output channels) activates Transmit Data to indicate that it is ready to receive the data. The transmitting device (the external device for input channels, the mainframe for output channels) places a 64-bit data word and 8 check bits on the channel and activates Data Ready to indicate that the data is available. The transmitting devices then places the next data word (and check bits) on the channel and activates Data Ready again. This process continues until the entire block has been transmitted; each word is transmitted without any acknowledgment from the receiving device.

The transmitting device stops sending data at the end of the block. When the receiving device is ready to receive another block, it activates Transmit Data. The transmitting device then sends another block and waits for another Transmit Data. This process

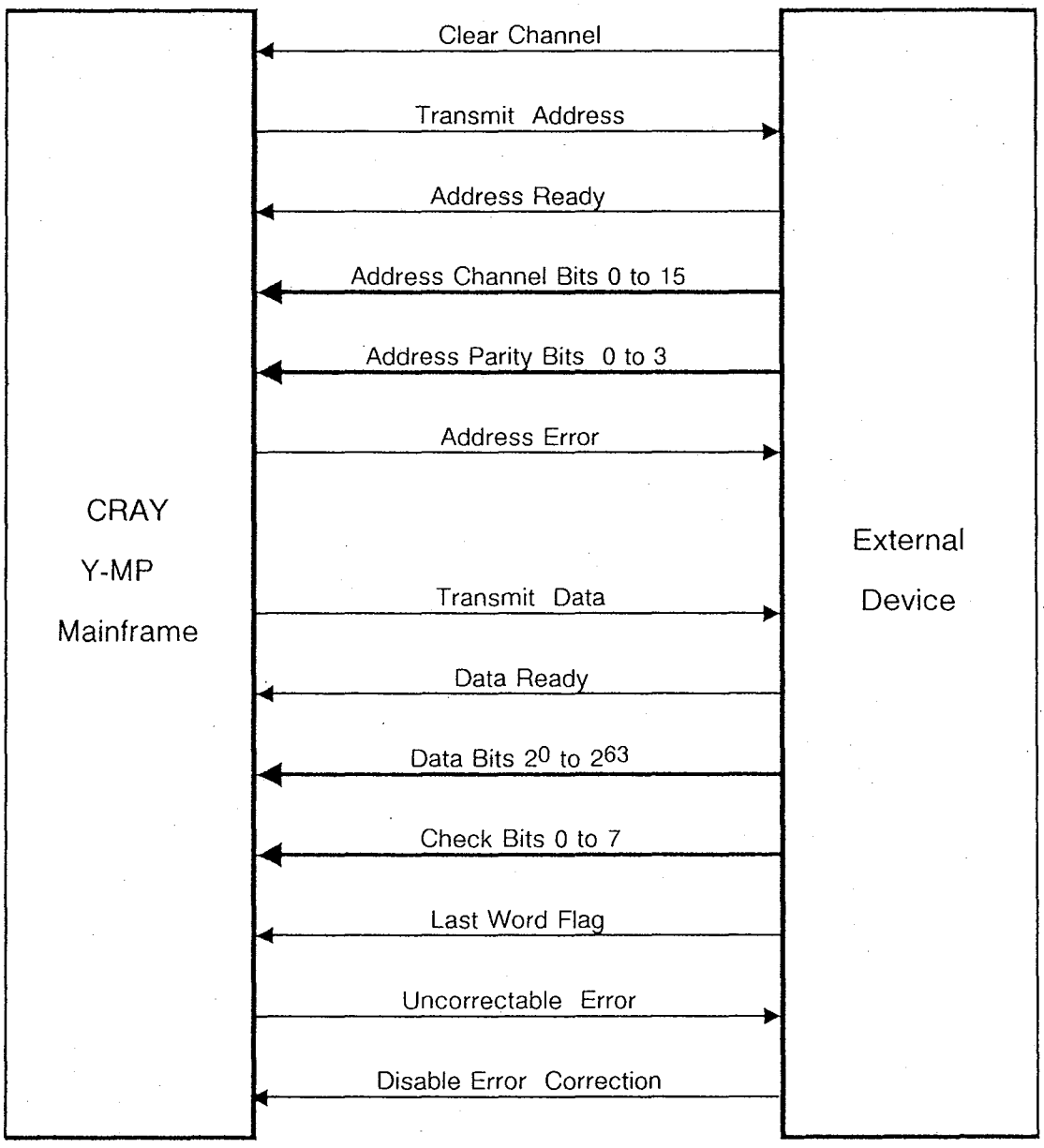


Figure 9-3. HISP Input Channel Signals

continues until all the data has been transmitted. When the last word of the last block has been placed on the channel, the transmitting device indicates that the transfer is complete by activating the Last Word flag.

The mainframe activates Uncorrectable Error if it detects an error during the the transmission of address information or data. Errors are discussed in detail in the next subsection.

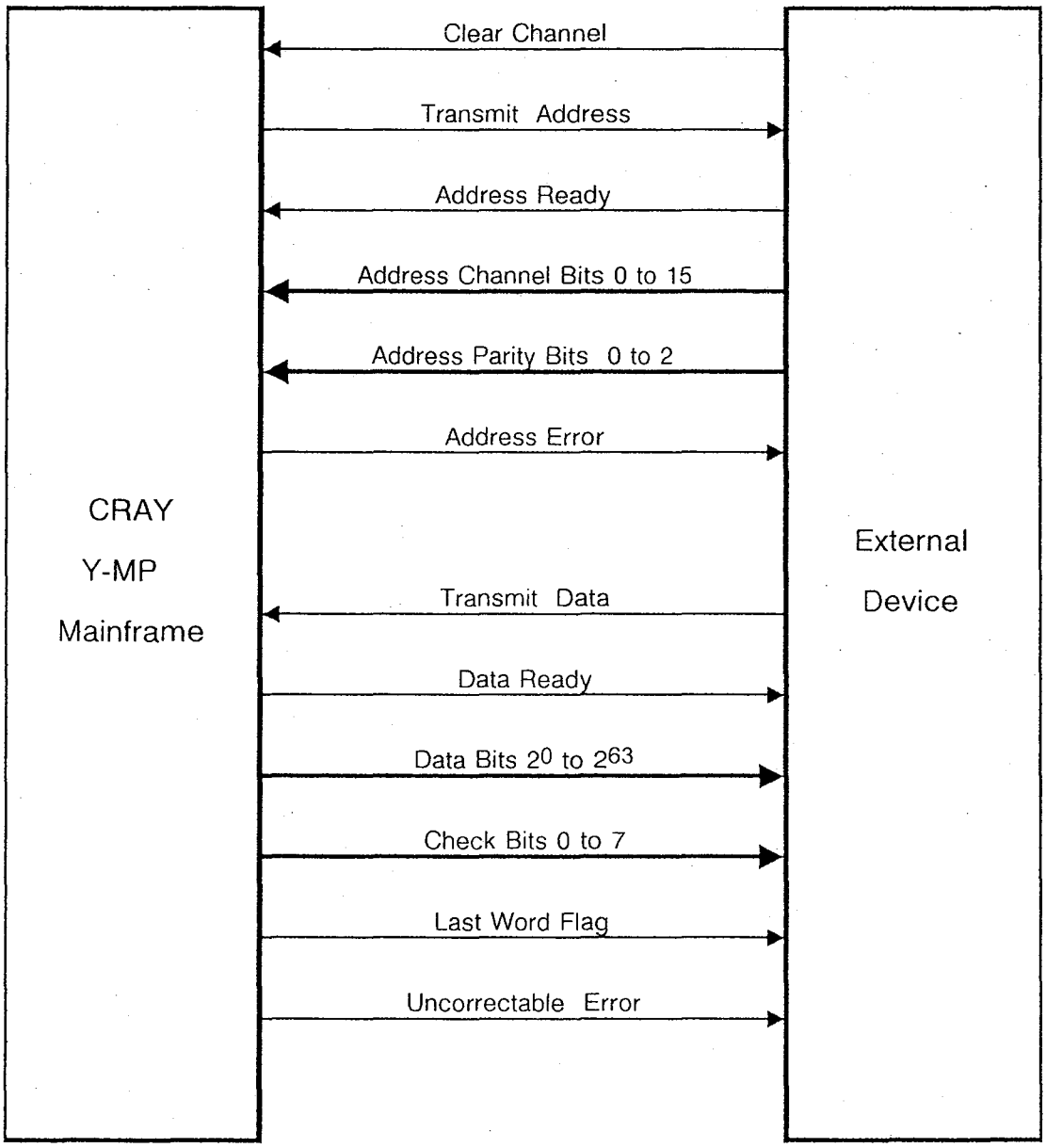


Figure 9-4. HISP Output Channel Signals

For input channel maintenance, an external device can send Disable Error Correction to the mainframe. When this signal is active, SECDED errors occurring on the channel are not detected or corrected; the data and check bits are written to Central Memory without modification.

Table 9-4. HISP Address Channel Bits

Parcel	Address Bits															
	Parity Bit 3				Parity Bit 2				Parity Bit 1				Parity Bit 0			
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	SA 2^9	SA 2^{31}	SA 2^{30}	SA 2^{29}	SA 2^{28}	SA 2^{27}	SA 2^{26}	SA 2^{25}	SA 2^{24}	SA 2^{23}	SA 2^{22}	SA 2^{21}	SA 2^{20}	SA 2^{19}	SA 2^{18}	SA 2^{17}
2	SA 2^{13}	SA 2^{12}	SA 2^{11}	SA 2^{10}	SA 2^{16}	SA 2^8	SA 2^7	SA 2^6	SA 2^5	SA 2^4	SA 2^3	SA 2^2	SA 2^1	SA 2^0	TWC 2^{13}	TWC 2^{12}
3	Not Used	Not Used	SA 2^{15}	SA 2^{14}	TWC 2^{11}	TWC 2^{10}	TWC 2^9	TWC 2^8	TWC 2^7	TWC 2^6	TWC 2^5	TWC 2^4	TWC 2^3	TWC 2^2	TWC 2^1	TWC 2^0

HISP Channel Programming

The HISP channels are programmed from the external devices to which they are connected; they cannot be programmed from the mainframe. For more information, refer to the programming manual for the appropriate external device .

HISP Channel Errors

The mainframe can detect several types errors on the HISP channels. When it detects an error, it stops channel operation and informs the external device using two signals, Address Error and Uncorrectable Error. (With the exception of errors which occur when transferring data from Central Memory to a HISP output channel, HISP channel errors are not reported to any CPU in the mainframe.)

If the mainframe detects an error involving address information, it activates Address Error and Uncorrectable Error. There are four types of errors in this category:

- Address parity errors.
- Unexpected Address Ready received. This condition occurs if the mainframe receives four Address Ready signals in succession or if it receives an Address Ready during data transmission.
- Unexpected Data Ready received (input channels only). This condition occurs if the mainframe receives Data Ready when it is expecting to receive Address Ready.

- Unexpected Transmit Data received (output channels only). This condition occurs if the mainframe receives Transmit Data when it is expecting to receive Address Ready.

If the mainframe detects any other type of error, it activate only Uncorrectable Error. There are two types of errors in this category:

- Uncorrectable (double-bit) data errors. The mainframe can detect data errors on input channels (when the data is received across the channel) and output channels (when the data is read from Central Memory). An uncorrectable error of either type activates the Uncorrectable Error signal. Errors occurring while reading data from Central Memory are also reported to the mainframe. Refer to "Error Detection and Correction" in Section 2 of this manual for more information.

- Word count mismatch (input channels only). This type of error can occur because the mainframe and the external device keep track separately of the number of word remaining to be transferred. If an I/O transfer operates properly, the word count maintained by the mainframe should decrement to 0 when the last Data Ready is received. Simultaneously, the external device should send the Last Word flag to the mainframe. A word count mismatch occurs if the mainframe's word count reaches 0 and the Last Word flag is not received or if the Last Word flag is received with the mainframe's word count not equal to 0.

Once activated, Address Error and Uncorrectable Error remain active until the mainframe receives Clear Channel from the external device. The Clear Channel signal terminates the transfer in which the error occurred and prepares the mainframe to begin a new channel transfer.

VHISP CHANNELS

The VHISP channels transfer data between Central Memory and a Cray Solid-state Storage Device (SSD). Each channel consists of two parallel 64-bit channels, allowing two successive Central Memory words to be transferred across the channel simultaneously. Each 64-bit channel also contains 8 check bits for data protection, identical to the SECDED scheme used by Central Memory. (Refer to "Error Detection and Correction" in Section 2 of this manual for more information.) Unlike the LOSP and HISP channels, the VHISP channels are bidirectional. One VHISP channel number applies to both input and output, and data transfers can take place in only one direction at a time.

Data is transmitted across the VHISP channels in blocks. Each block contains 64 successive Central Memory words and is transferred across a VHISP channel in 32 operations. Partial blocks are not permitted.

Two VHISP registers for each channel can be loaded from any CPU. The Channel Address (CA) register contains the next Central Memory address to be transferred. When a data transfer begins, the CA register contains the address of the the first word to be transferred. As each word is written to or read from Central Memory (depending on

whether the channel is being used for input or output) the CA register increments. This process continues until all words have been transferred.

The Block Length (BL) register determines the number of blocks to be transferred. The BL register decrements after each block is transferred. An I/O transfer is completed when the contents of the BL register is equal to zero.

Each VHISP channel uses two adjacent CPUs for access to Central Memory. Two successive Central Memory words are transferred across the channel (to or from the SSD) simultaneously. However, because the CPUs contain data buffers, the transfer of successive words between Central Memory and the CPUs does not necessarily occur simultaneously.

Each CPU contains two input and two output VHISP data buffers. Each buffer can hold 16 64-bit data words (plus check bits). The buffers in two adjacent CPUs operate simultaneously when transferring data across the channel. This allows 16 128-bit words to be transferred without delays caused by memory conflicts. However, the buffers in the adjacent CPUs operate independently when transferring data to or from Central Memory.

The input channel data buffers operate as follows: When a data transfer begins, only one buffer in each CPU, Buffer A, is used. Buffer A begins receiving data from the channel. When Buffer A is full, the second buffer, Buffer B, begins to receive the channel data, and Buffer A begins to send its data to Central Memory. When Buffer B is full and Buffer A is empty, they exchange roles; Buffer A receives the channel data and Buffer B sends its data to Central Memory. The buffers continue to exchange roles, with one buffer receiving channel data and the other buffer sending data to Central Memory, until the data transfer is complete.

The output channel buffers operate similarly to the input channel buffers, except that data is transferred in the opposite direction. One buffer receives data from Central Memory while the other buffer transmits its data across the channel. The buffers exchange roles for each group of 16 words .

VHISP Channel Signals

The VHISP channel signals can be divided into three categories:

- Status and initialization signals:
    - Offline
    - Busy
    - Clear Control
    - Acknowledge Clear
    - Block Address Bits 2^0 to 2^{28}
    - Block Length Bits 2^0 to 2^{17}
    - Write
    - Set CA
    - Uncorrectable Error
    - Block Length Error

- Input transfer signals:
  - Transmit Enable (mainframe to SSD)
  - Data Ready (SSD to mainframe)
  - Even CPU Data bits 2^0 to 2^{63}
  - Even CPU Check bits 0 to 7
  - Odd CPU Data bits 2^0 to 2^{63}
  - Odd CPU Check bits 0 to 7
  - Done and Empty
- Output transfer signals:
  - Transmit Enable (SSD to mainframe)
  - Data Ready (mainframe to SSD)
  - Even CPU Data bits 2^0 to 2^{63}
  - Even CPU Check bits 0 to 7
  - Odd CPU Data bits 2^0 to 2^{63}
  - Odd CPU Check bits 0 to 7
  - Done and Empty

Figure 9-5 shows the channel signals for a VHISP channel. The following subsections explain the signals in detail.

Status and Initialization Signals

The status and channel initialization signals transmit SSD status and error information from the SSD to the mainframe and allow the mainframe to initialize the SSD for a data transfer. The Offline signal indicates the status of the SSD port to which the VHISP channel is connected. If the signal is active, the port is not in operation and cannot perform data transfers. The Busy signal becomes active when the SSD begins a data transfer and remains active until the transfer is complete.

The following signals are used to begin a data transfer: Clear Control, Acknowledge Clear, Block Address, Block Length, Write, and Set CA. The normal sequence is as follows:

1. The mainframe sends Clear Control to the SSD. This initializes the SSD port to which the VHISP channel is connected, stopping any data transfer in progress, clearing all error conditions, and preparing the port to begin a transfer.

2. When the SSD is ready to begin the transfer, the SSD sends Acknowledge Clear to the mainframe.

3. The mainframe sends Block Address bits 2^0 to 2^{28} to the SSD. The Block Address indicate the starting block address in the SSD. (Not all the Block Address bits are used by the SSD. The number of bits that are used is determined by the memory size of the SSD.)

4. The mainframe sends Block Length bits 2^0 to 2^{17} and Write to the SSD. The Block Length indicates the number of bits to be transferred. Write is logic 0 for an input transfer (SSD to mainframe) and logic 1 for an output transfer (mainframe to SSD).

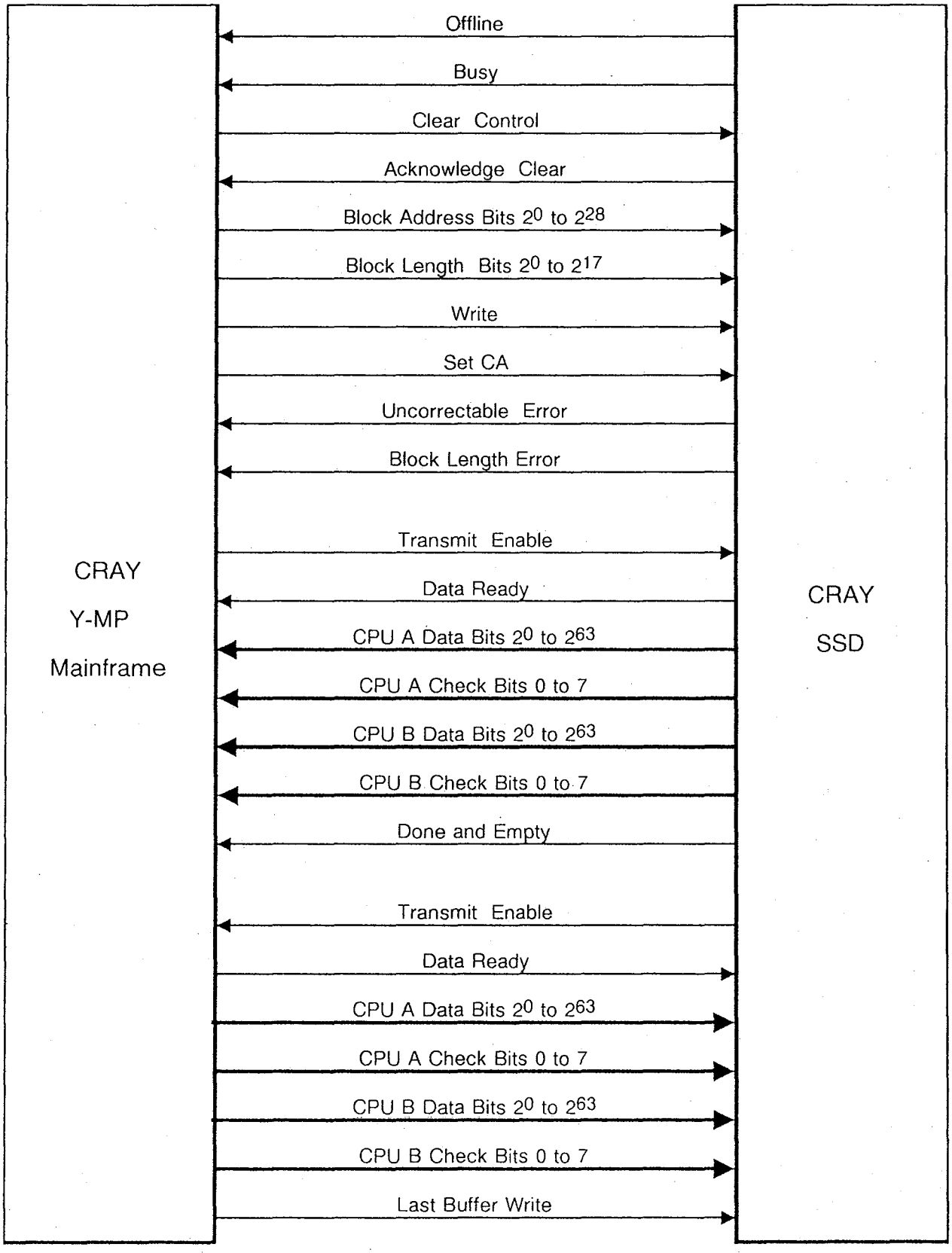


Figure 9-5. VHISP Channel Signals

5. The mainframe sends Set CA to the SSD. This signal causes the SSD to latch Block Address, Block Length and Write and starts the data transfer.

6. An input or output data transfer begins. The signals used for the transfers are explained in the following two subsections.

Uncorrectable Error and Block Length Error are activated by the SSD if it detects errors during a data transfer. For further information, refer to "VHISP Channel Errors" later in this section.

Input Transfer Signals

The input transfer signals are used during a transfer from the SSD to the mainframe. The normal sequence is as follows:

1. The mainframe sends Transmit Data to the SSD. This signal indicates that the mainframe is ready to receive 1 buffer (16 128-bit words plus check bits) of data.

2. The SSD sends Data Ready to the mainframe. This signal indicates that the data will be sent across the channel beginning in 4? CPs.

3. The SSD sends 16 128-bit words of data (with 16 check bits for each word) to the mainframe at 2 CP intervals. Each 128-bit word corresponds to two consecutive Central Memory addresses. The data and check bits are received by two adjacent CPUs. The lower-numbered CPU receives the data for the lower Central Memory address. The higher-numbered CPU receives the data for the higher Central Memory address.

4. Steps 1 to 3 are repeated until all blocks of data have been transferred across the channel.

5. The SSD sends Done and Empty to the mainframe to indicate that the transfer is complete.

Certain error conditions can cause this sequence to terminate prematurely. Refer to "VHISP Channel Errors" later in this section for more information.

Output Transfer Signals

The data output signals are used during a transfer from the SSD to the mainframe. The normal sequence is as follows:

1. The SSD sends Transmit Data to the mainframe. This signal indicates that the mainframe is ready to receive 1 buffer (16 128-bit words plus check bits) of data.

2. The mainframe sends Data Ready to the SSD. This signal indicates that the data will be sent across the channel beginning in 4? CPs.

3. The mainframe sends 16 128-bit words of data (with 16 check bits for each word) to the SSD at 2 CP intervals. Each 128-bit word corresponds to two consecutive Central Memory addresses. The data and check bits are sent by two adjacent CPUs. The lower-numbered CPU sends the data from the lower Central Memory address. The higher-numbered CPU sends the data from the higher Central Memory address.

4. Steps 1 to 3 are repeated until all blocks of data have been transferred across the channel.

5. The mainframe sends Last Buffer Write to the SSD to indicate that the transfer is complete.

Certain error conditions can cause this sequence to terminate prematurely. Refer to "VHISP Channel Errors" later in this section for more information.

VHISP Channel Programming

Programming a VHISP channel is similar to programming a LOSP channel. Most of the same instructions (privileged to Monitor Mode) are used, although their operation is slightly different. Table 9-5 shows the instructions that are applicable to the VHISP channel. All the instructions except 033*i*00 operate slightly differently than they do in the LOSP channels.

Table 9-5. VHISP Channel Instructions

Machine Instruction	CAL	Description
0010*jk*†	CA,A*j* A*k*	First occurrence: Set channel (A*j*) SSD starting block address to (A*k*) . Second occurrence: Set channel (A*j*) CA register to (A*k*) .
0011*jk*†	CL,A*j* A*k*	Set channel (A*j*) BL register to (A*k*), select input or output transfer, and begin I/O sequence.
0012*j*0†	Cl,A*j*	Clear channel (A*j*) .
033*i*00	A*i* Cl	Transmit interrupting channel number to A*i*.
033*ij*1	A*i* CE,A*j*	Transmit channel (A*j*) status word to A*i*.

† Privileged to Monitor Mode

Instruction 0010*jk* performs two functions. The first time it is executed, it determines the starting block address in the SSD. The second time it is executed, it loads the channel CA register, which determines the starting address in Central Memory.

Instruction 0011*jk* loads the channel BL register and determines whether to do an input or an output transfer. A*k* register bits 2^0 to 2^{17} are loaded into the BL register. A*k* register bit 2^{23} determines the transfer direction. A logic 0 causes data to be transferred from the SSD to the mainframe. A logic 1 transfers data from the mainframe to the SSD.

Instruction 0012*j*0 performs several functions to prepare the mainframe and SSD for a data transfer. It clears the channel interrupt flag and status word, initializes the mainframe side of the channel, and sends Clear Control to the SSD.

Instruction 033*ij*1 transmits a channel status word to register A*i* which indicates the number of block remaining to be transferred and the state of several error flags. Table 9-6 shows how the bits of the status word are used. The error flags are discussed in more detail in the next subsection.

Table 9-6. VHISP Channel Status Word

Bit	Description
2^0 to 2^{17}	BL register bits 2^0 to 2^{17}
2^{18}	Channel transfer in progress
2^{19}	Not used (forced to 0)
2^{20}	Block length error
2^{21}	Uncorrectable (double-bit) error in SSD
2^{22}	Uncorrectable (double-bit) error in mainframe
2^{23}	Fatal error (Block length error or uncorrectable memory error in SSD or mainframe)

To initiate a transfer across a VHISP channel, execute the following sequence of instructions:

	Machine Instruction	CAL	Comment
1.	0012*j*0	CI,A*j*	Clear channel .
2.	0010*jk*	CA,A*j* A*k*	Load SSD starting block address
3.	0010*jk*	CA,A*j* A*k*	Load CA register
4.	0011*jk*	CL,A*j* A*k*	Load BL register, select transfer direction, and begin I/O sequence.

VHISP Channel Errors

Two types of errors can occur on the VHISP channels: block length errors and data errors. Block length errors and uncorrectable (double-bit) data errors stop the channel transfer, set bits in the VHISP channel status word, and generate an I/O interrupt request.

Block length errors can occur because the mainframe and the SSD use separate block length (BL) registers to keep track of the number of blocks remaining to be transferred. The block lengths maintained by both devices decrement to 0 simultaneously if an I/O transfer operates properly. On the other hand, if the block length of one device reaches 0 while the other device is still expecting data to be transferred, a block length error occurs. A block length error stops the transfer and sets bits 2^{20} and 2^{23} of the VHISP channel status word.

Block length errors can occur on both input and output transfers. There are two ways a block length error occur can during an input transfer:

- The mainframe's BL register reaches 0, but the SSD sends Data Ready to indicate that there is more data to be sent. This error is detected by the mainframe.
- The SSD's BL register reaches 0,but the mainframe sends Transmit Enable to indicate that it is ready to receive more data. This error is detected by the SSD, which reports the error to the mainframe through the Block Length Error signal.

There are also two ways a block length error can during an output transfer:

- The SSD's BL register reaches 0, but the mainframe sends Data Ready to indicate that there is more data to be sent. This error is detected by the SSD, which reports the error to the mainframe through the Block Length Error signal.
- The mainframe's BL register reaches 0, but the SSD sends Transmit Enable to indicate that it is ready to receive more data. This error is detected by the mainframe.

Data errors can occur on both input and output transfers. There are two places data errors can be detected during an input transfer:

- The readout path from SSD memory to the channel. An uncorrectable error detected here activates the Uncorrectable Error signal to the mainframe, which sets bits 2^{21} and 2^{23} of the VHISP channel status word.

- The input path from the channel to the mainframe. An uncorrectable error detected here sets bits 2^{22} and 2^{23} of the VHISP channel status word.

There are also two places data errors can be detected during an output transfer:

- The readout path from Central Memory. An uncorrectable error detected here sets bits 2^{22} and 2^{23} of the VHISP channel status word. This type of error is in the mainframe. In addition to affecting the channel operation, the CPU detecting this type of error responds the same way it responds to any Central Memory error. Refer to "Error Detection and Correction" in section 2 for more information.

- The output path from the channel to the SSD. An uncorrectable error detected here activates the Uncorrectable Error signal to the mainframe, which sets bits 2^{21} and 2^{23} of the VHISP channel status word.

I/O CHANNEL MEMORY ACCESS

The I/O section uses Port D in each CPU for access to Central Memory. Each LOSP and HISP channel uses Port D in a specific CPU. Each VHISP channel uses Port D in two adjacent CPUs. Port D is also used by the instruction buffers in each CPU.

The I/O channels shares access to Central Memory with many other resources (registers, instruction buffers, and the Exchange Package) in each CPU. Because of this, memory conflicts can occur. There are three levels of conflicts that affect the I/O channels. The first level involves conflicts between Port D and all other ports in all the CPUs. The next level involves conflicts between the I/O channels and the instruction buffers for use of Port D. The last level involves conflicts between the five channels (LOSP input and output, HISP input and output, and VHISP) in each CPU.

The remainder of this subsection discusses memory conflicts that are specific to Port D and the I/O section. It assumes you are familiar with the basic concepts of memory conflicts. If you need more information in this area, refer to "Memory Conflicts" in Section 2 of this manual.

The following rules apply to intra-CPU memory conflicts involving Port D:

- In each CPU, if Port D is being use for an instruction fetch sequence, it has priority over Ports A, B, and C.

- If Port D is being used for an I/O transfer, it normally has lower priority than Ports A, B, and C. However, if a Port D memory reference has been forced to hold for 32 CPs, Port D is temporarily given top priority so that one memory

reference can proceed. After the reference has begun, Port D returns to its low-priority status.

- Instruction fetches have priority over I/O transfers for use of Port D.
- The order of priority for the I/O channels for use of Port D is as follows (from highest to lowest priority):

  1. LOSP input channel
  2. LOSP output channel
  3. HISP input channel
  4. HISP output channel
  5. VHISP channel

- When a HISP or VHISP channel begins a memory reference, it is temporarily given priority over all other I/O channels. The channel retains the highest priority until it has made 16 memory references. This allows an entire HISP or VHISP data buffer to be emptied or filled without delays caused by memory requests from other channels.
- LOSP channels can make a maximum of 1 memory reference every 4 CPs. When a LOSP channel begins a memory reference (that is, after it has passed all conflicts), a lockout circuit disables LOSP channel memory requests for the next 3 CPs. A memory reference by either LOSP channel (input or output) locks out both LOSP channels for the next 3 CPs, providing an interval for HISP and VHISP channel memory requests.
- A second lockout circuit disables HISP channel memory requests for 3 CPs following a HISP channel data transfer to or from memory. When a HISP input or output channel memory request is granted, the channel is allowed to transfer an entire 16-word buffer to or from Central Memory without interruption from other channels. After the last of the 16 memory references begins, both HISP channels are prevented from making memory requests for the next 3 CPs, providing an interval for VHISP channel memory requests.

I/O INTERRUPTS

I/O interrupts are generated by the LOSP and VHISP channels to indicate that a data transfer is complete or that an unexpected error occurred. (Parity errors on LOSP input channels and correctable data errors on VHISP channels do not cause interrupts.)

When an I/O transfer is complete, or an interrupt-generating error occurs, an interrupt request is sent to a CPU. The CPU that receives the interrupt request is not necessarily the CPU that initiated the transfer or the CPU that controls the channel that was used. The following rules determine the CPU that receives the request:

1. If one CPU has its Selected for External Interrupt (SEI) bit (in the Mode register) set, that CPU will receive all I/O interrupt requests. If more than one CPU has its SEI bit set, the lowest-numbered CPU with its SEI bit set will receive all I/O interrupt requests.

2. If condition 1 does not apply, a CPU holding on a test and set instruction (0034*jk*) will receive the interrupt request. If more than one CPU is holding on a test and set instruction, the lowest-numbered CPU holding on a test and set instruction will receive all interrupt requests.

3. If conditions 1 and 2 do not apply, an interrupt request from a channel is directed to the CPU that last issued a clear interrupt instruction (0012*j*0 or 0012*j*1) to that channel.

A CPU that receives an interrupt request will honor the request (that is, execute an Exchange sequence) only if it is not in Monitor Mode. If two or more I/O channels generate interrupt requests to the same CPU, instruction 033*i*00 will return the lowest-numbered I/O channel requesting service. When that channel's interrupt flag is cleared, the instruction will return the next-lowest I/O channel number. When all interrupt flags are cleared, 033*i*00 will return a value of 0.

HARDWARE IMPLEMENTATION

This subsection describes the major hardware components (options and interconnections) that comprise the I/O section. The I/O section is divided equally among all eight CPU modules. On each CPU, the I/O section can be further divided into four functional areas:

- Channel control paths
- HISP address channels
- Input data paths
- Output data paths

The following subsections describe these functional areas in detail.

Channel Control Paths

Channel control paths perform several major functions:

- They transmit all control signals needed by the external devices and receives all the control signals sent by the external devices.
- They initiate transfers across the LOSP and VHISP channels.
- They control the transmission of data between the channels and Central Memory.
- They resolve conflicts between channels for memory access.
- Along with the external devices, they controls the transfer of data across the channels.
- They determine when a channel transfer is complete.
- They generate interrupt requests at the end of LOSP and VHISP channel transfers.

- They respond to errors detected during channel operation.

The channel control hardware consists of seven options: one DA, three DB, one DC, one DD, and one DE option. The DA and DB options are used by all the channel types, while the DC, DD, and DE options are used primarily by the LOSP, HISP, and VHISP channels, respectively.

Figure 9-6 is block diagram of the channel control options. Refer to this diagram while reading the following option descriptions.

DA and DB Options

The DA and DB options have two primary functions. First, they contain the Channel Address (CA) register for each channel. The CA register indicates the Central Memory address that will be used for the next data transfer between the channel and Central Memory. After each word is transferred, the CA register increments. Second, the DA option determines which channel is allowed to make a memory request each CP. If two or more channels request memory access simultaneously, the DA option uses a predetermined priority scheme to sequence the requests. Refer to "I/O Channel Memory Access" in this section for more information.

It addition to their primary functions, the DA and DB options perform several secondary functions:

- They contain the LOSP channel Channel Limit (CL) registers.
- They contain the VHISP channel Block Length (BL) register.
- They contain the HISP output channel Transfer Word Count (TWC) register.
- The DA options on two adjacent CPUs contain a copy of the lowest four bits of the HISP input channel TWC register.
- The DA option transmits the Block Address and Block Length signals to the VHISP channel.
- For LOSP channel instruction 033*ij*0, the DA and DB options transmit the contents of a CA register to register A*i*.
- For VHISP channel instruction 033*ij*1, they transmit the contents of the Status register to register A*i*.
- The DA option determines which input channel can transmit its data to Central Memory. It also controls the reading of data from the HISP and VHISP channel input buffers.

The CA, CL, BL, and TWC registers and the readout path to register A*i* are divided bit-wise among the DA and DB options. Table 9-7 shows the arrangement.

The following paragraphs describe the input and output terms of the DA and DB options.

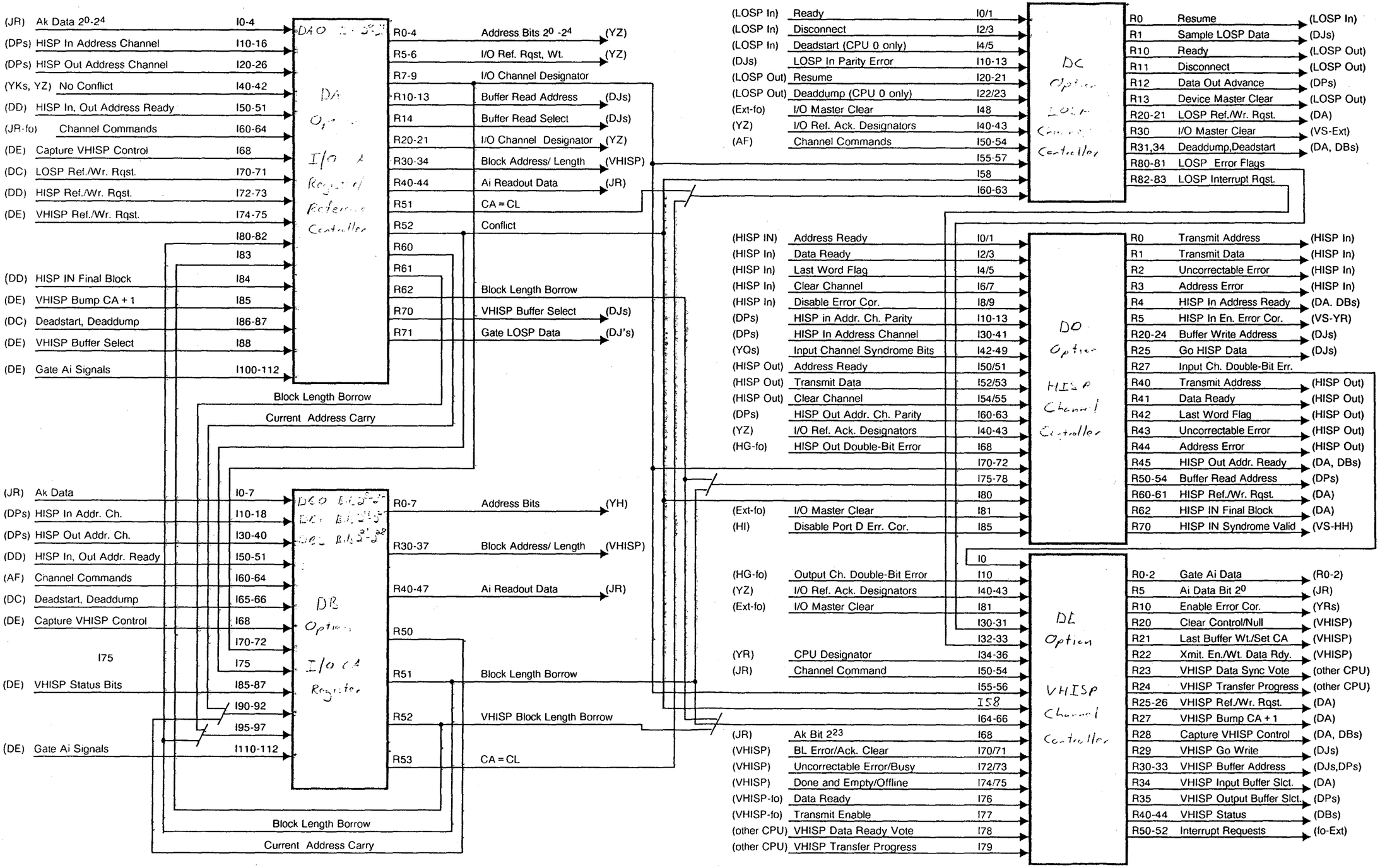


Figure 9-6. I/O Control Block Diagram

Table 9-7. DA and DB Options Bit Assignments

	DA0	DB0	DB1	DB2
CA registers (all channels) CL registers (LOSP channels) Path to register A*i*	2^0 to 2^4	2^5 to 2^{12}	2^{13} to 2^{20}	2^{21} to 2^{28}
TWC register (HISP input channel)†	2^0 to 2^3	not used	not used	not used
TWC register (HISP output channel)	2^0 to 2^4	2^5 to 2^7	2^8 to 2^{10}	2^{11} to 2^{13}
BL register (VHISP channel)	2^0 to 2^4	2^5 to 2^{12}	2^{13} to 2^{17}	not used

† The entire HISP input channel TWC register (bits 2^0 to 2^{13}) is held on the DD option.

The LOSP channel CA and CL registers and the VHISP channel CA and BL registers can be loaded from A registers under program control. The registers are loaded through the A*k* Data terms, I0 to I4 on the DA option and I0 to I7 on the DB options. Channel Command bits 2^0 to 2^4, terms I60 to I64 on both option types, are derived from CPU instructions and control which register is loaded. (The channel commands perform additional functions on the DC and DE options.) Figure 9-7 shows the functions of the Channel Command bits.

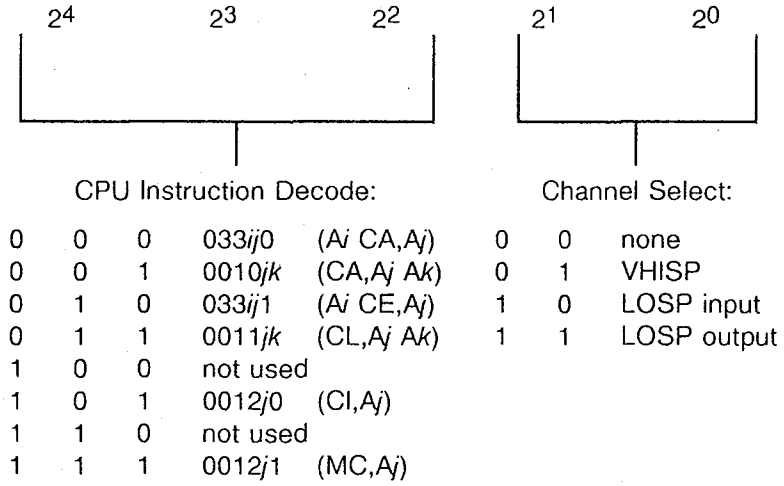


Figure 9-7. Channel Comand Bits

On CPU 0 only, the LOSP channel CA and CL registers are also loaded during Deadstart and Deaddump sequences. The Deadstart and Deaddump signals (DA option terms I86 and I87, DB option terms I65 and I66) affect the LOSP input and output channels, respectively. Deadstart sets the input channel CA register to 0 and the CL

register to 2^{29}-1 (all 1s). Deaddump sets the output channel CA register to 0 and the CL register to 2^{29}-1.

The HISP channel CA and TWC registers are loaded into the DA and DB options from the HISP address channels. The HISP In Address Channel enters the DA option as I10 to I16 and the DB options as I10 to I18. The HISP Out Address Channel enters the DA option as I20 to I26 and the DB options as I30 to I40. HISP In Address Ready and HISP Out Address Ready (terms I50 and I51 on both option types) from the DD option control the loading of the CA and TWC registers from the address channels. Refer to "HISP Address Channels" in this section for more information.

When an I/O channel is ready to transfer data to or from Central memory, it makes its request to the DA option. Terms I70 and I71 are requests from the LOSP channels. Term I70, LOSP Reference Request, indicates that either LOSP channel is requesting memory access. Term I71, LOSP Write Request, indicates that the input channel is making the request (that is, a memory write request). Terms I72 and I73 indicate requests from the HISP channels, and terms I74 and I75 are requests from the VHISP channel.

The DA option resolves any memory access conflicts which occur between the channels. It then sends I/O Reference Request (term R5) to the YZ option. If a channel is requesting a memory write, the DA option also sends I/O Write Request (term R6). The DA option sends Channel Designator bits 2^0 to 2^2 to indicate the channel that is making a memory request. Table 9-8 shows how the Channel designator bits are used. Terms R20 and R21 are Channel Designator bits 2^0 and 2^1 to the YZ option. Terms R7 to R9 are Channel Designator bits 2^0 to 2^2 to the DB, DC, DD, and DE options.

Table 9-8. Channel Designator Bits

Designator Bits			Channel
2^2	2^1	2^0	
X	0	0	not used
0	0	1	LOSP output
0	1	0	HISP output
0	1	1	VHISP output
1	0	1	LOSP input
1	1	0	HISP input
1	1	1	VHISP input

X = 0 or 1

The DA and DB options send Address Bits 2^0 to 2^{28} with the memory requests. These bits come from the CA register of the channel making the request. The DA options send

bits 2^0 to 2^4 (terms R0 to R4) to the YZ option. The DB options send bits 2^5 to 2^{28} (terms R0 to R7) to the YH option.

The DA option receives memory conflict information from the YK options (subsection conflicts) and from the YZ option (instruction fetch conflict) as terms I40 to I42, No Conflict. If all of these terms are active, an I/O memory request can proceed. If one or more of these terms is inactive, I/O memory requests are forced to delay until all three terms become active simultaneously. The DA option uses this information internally, and also passes it to the DB, DC, DD, and DE options using term R70, Conflict.

After a channel memory reference, the CA register increments. A LOSP or HISP channel memory reference simply increments the appropriate CA register by 1. However, because the VHISP channels use two adjacent CPUs for memory access, their operation is slightly different.

During a VHISP channel operation, each of the adjacent CPUs transfers every other word to or from Central Memory. The lower-numbered CPU transfers the first, third, and fifth words, and so on. The higher-numbered CPU transfers the second, fourth, and sixth words, and so on. Each CPU uses its own CA register to keep track of its own memory references. At the beginning of a VHISP transfer, both CA registers are loaded with the first Central Memory address. Therefore, the CA register on the higher-numbered CPU (CPU 1, 3, 5, or 7) must be incremented by 1 so that it points to the second address. DA option term I85 (active only on odd-numbered CPUs), Bump CA + 1, performs this function. For the remainder of the channel transfer, after a CPU transfers a word to or from Central Memory, its CA register increments by 2.

When a CA register increments, DA option term R60 is the carry from bits 2^0 to 2^4 to all the DB options. Term R60 on DB0 and DB1 is the carry from bits 2^5 to 2^{12} and 2^{13} to 2^{20}, respectively. DB0 term R60 propagates carries to the DB1 and DB2 options. DB1 term R60 propagates carries only to the DB2 option.

The DA and DB registers determine when a LOSP, HISP output, or VHISP channel transfer is complete. For the LOSP channels, they transmit CA = CL (DA term R51, DB term R53) to the DC option. For the HISP output channel, they decrement the TWC register after each word is transferred; when the TWC register reaches 0, the transfer is complete. When the TWC register decrements, DA option term R61 is the borrow from bits 2^0 to 2^4 to all the DB options. (A borrow term is logic 1 when all bits are 0). Term R51 on DB0 and DB1 is the borrow from bits 2^5 to 2^7 and 2^8 to 2^{10}, respectively. DB0 term R51 propagates borrows to the DB1 and DB2 options. DB1 term R51 propagates borrows only to the DB2 option.

To determine when the HISP output TWC register reaches 0, the DB options send term R51 to the DA and DD options and the DA option sends term R62 to the DD option. Each of these terms indicate that a range of bits in the TWC register is equal to 0. When all these terms are active, the TWC register is equal to 0; the DA register stops making HISP output channel memory requests and the DD option stops channel activity.

The DA option does not determine when a HISP input transfer is complete, but relies on the DD option for this information. The DA option receives term I84, HISP In Final Block when the final block has been received by the HISP input buffer. Because this block may not contain a full 16 words, the DA option uses TWC register bits 2^0 to 2^3 to determine how many memory requests to make.

The DA and DB options (DB0 and DB1) determine when a VHISP channel transfer is complete by decrementing the BL register after each block of data is transferred; when the BL register reaches 0, the transfer is complete.When the BL register decrements, DA option term R61 is the borrow from bits 2^0 to 2^4 to DB0 and DB1. Term R51 on DB0 is the borrow from bits 2^5 to 2^{12} to option DB1.

To determine when the VHISP channel BL register reaches 0, option DB0 sends term R51 to the DA and DE options, option DB1 sends term R52 to the DA and DE options, and the DA option sends term R62 to the DE option. When all these terms are active, the TWC register is equal to 0; the DA register stops making VHISP output channel memory requests and the DE option stops channel activity.

The DA and DB options transmit the contents of a LOSP channel CA register to register A*i* for instruction 033*ij*0 (Ai CA,Aj). They also transmit the VHISP Status Word to register A*i* for instruction 033*ij*1 (Ai CE,Aj). The DA and DB options receive Gate Ai Signals for the LOSP input, LOSP output and VHISP channels as I110, I111, and I112, respectively. If I110 or I111 is activated, the DA option gates the appropriate CA register to terms R40 to R44 and the DB options gate the CA register to terms R40 to R47. If I112 is activated, the BL register is gated to DA0 terms R40 to R44 (bits 2^0 to 2^4), DB0 terms R40 to R47 (bits 2^5 to 2^{12}), and DB1 terms R40 to R44 (bits 2^{13} to 2^{17}). VHISP channel flags from the DE option enter DB1 option as I85 and I87 and DB2 as I85, I86, and I87. The flags are gated to DB1 terms R5 and R7 (bits 2^{18} and 2^{20}) and DB2 terms R40 to R42 (bits 2^{21} to 2^{23}). Bits 2^{19} and 2^{24} to 2^{28} of the VHISP status word are forced to 0 on DB1 and DB2.

The DA and DB options transmit the VHISP Block Address and Block Length to the VHISP channel. The CPU pair that comprises a single VHISP channel shares this task; the lower-numbered CPU transmits Block Address Bits 2^0 to 2^{28}. The higher-numbered CPU transmits Block Length bits 2^0 to 2^{17}. Instruction 0010*jk* transmits the Block Address to DA option terms I0 to I4 and DB option terms I0 to I7 on the lower-numbered CPU. On the higher-numbered CPU, instruction 0011*jk* transmits the Block Length to DA0 terms I0 to I4, DB0 terms I0 to I7, and DB1 terms I0 to I4. All of theses terms are latched and transmitted to the VHISP channel when the DA and DB options receive Capture VHISP Control, term I68.

The DA option controls which input channel data is transmitted to Central Memory. Term R71, Go LOSP Data, is logic 1 when LOSP channel data is used and logic 0 when HISP or VHISP channel is used. Term R14, Buffer Read Select, selects between HISP data and VHISP data. Logic 0 selects HISP data; logic 1 selects VHISP data. Finally, if HISP or VHISP data is selected, terms R10 to R13, Buffer Read Address selects 1 of 16 words in the channel input buffer. These terms are the output of a counter which increments after each word is read from the buffer.

The DA option determines which VHISP input buffer, buffer A or buffer B, receives channel data at any time. The DA option receives term I88, VHISP Buffer Select, delays it for memory conflicts, and retransmits it as term R70.

DC Option

The DC option is the LOSP channel controller. It performs several major functions:

- It transmits all the control signals needed by the external device and receives all the control signals sent by the external device.
- It initiates all LOSP data transfers.
- Along with the external device, it controls all data transfers across the LOSP channels.
- It makes memory requests for the LOSP channels.
- It controls data assembly (input channels) and disassembly (output channel) between the the LOSP channels and Central Memory.
- It generates error flags and interrupt requests.

The following paragraphs describe the input and output terms of the DC option.

Many of the DC option input and output terms are connected directly to the LOSP channels. The input channel terms are Ready (differential pair I0/1), Resume (R0), and Disconnect (I2/3). The output channel terms are Ready (R10), Resume (I20/21), and Disconnect (R11). On CPU 0 only, the input channel (20_8) receives Deadstart as terms I4/5 and the output channel (21_8) receives Deaddump as terms I22/23. The DC option on CPU 0 retransmits Deadstart and Deaddump to the DA and DB options as R31 and R34. It also retransmits Deadstart as I/O Master Clear, term R30. I/0 Master Clear is fanned out to every CPU module. It enters the DC option on each CPU module as term I48.

The DC option receives CPU instructions as Channel Command bits 2^0 to 2^4, terms R50 to R54. The Channel Commands are identical to those received by the DA and DB options. The DC option responds to CPU instructions 0010*jk* (CA,A*j* A*k*), 0012*j*0 (CI,A*j*), 0012*j*1 (MC,A*j*) and 033*ij*1 (A*j* CE,A*j*).

The DC option makes memory requests to the DA option using LOSP Reference Request and Write Request, terms R20 and R21. When a LOSP request makes it past conflicts with other channels, the DA option acknowledges the request using the I/O Channel designator bits, DC option terms I55 to I57. The DC option receives memory conflict information through term I58. All of these terms are explained in "DA and DB Options" in this section.

When an I/O channel memory request is granted, the YZ option transmits Reference Acknowledge Designator bits 2^0 to 2^3 to the I/O control options. Bit 2^0 indicates a memory read reference. Bits 2^1 and 2^2 indicate the channel type (0 = none, 1 = LOSP, 2 = HISP, 3 = VHISP). Bit 2^3 indicates a memory write reference. The bits enter the YC option as I40 to I43.

The DC option assembles input data by using Sample Data, term R1. Each time this term is activated, it cause a parcel of input data to be read from the input channel. Similarly, the DC option uses Data Out Advance, term R12 to disassemble output data. Each time R12 is activated, a new parcel of data is sent to the output channel.

The DC option generates LOSP Error Flags (terms R80 and R81) for instruction 033*ij*1. Input channel errors (parity errors) are received by the DC option as terms I10 to I13 and set term R80. Output channel errors (unexpected Ready received) are detected by the DC option and set term R81.

The DC option generates LOSP Input and Output Interrupt Requests (terms R80 and R81) when a channel transfer completes or, for the output channel only, when an error occurs.

DD Option

The DD option is the HISP channel controller. It performs several major functions:

- It transmits all the control signals needed by the external device and receives most of the control signals sent by the external device.
- Along with the external device, it controls all transfers across the HISP channels.
- It contains the HISP input channel Transfer Word Count (TWC) register.
- It makes memory requests for the HISP channels.
- It controls the writing of data to the input buffers the the reading of data from the the output buffers.

The DD option also detects uncorrectable (double-bit) errors which occur on HISP and VHISP input transfers.

The following paragraphs describe the input and output terms of the DD option.

Most of the DD option input and output terms are connected directly to the HISP channels. The input channel terms are Address Ready (differential pair I0/1), Transmit Address(R0), Data Ready (I2/3), Transmit Data (R1), Last Word Flag (I4/5), Uncorrectable Error (R2), Clear Channel (I6/7), Address Error (R3), and Disable Error Correction (I8/9). The output channel terms are Address Ready (I50/51), Transmit Address (R40), Transmit Data (I52/I53), Data Ready (R41), Last Word Flag (R42), Uncorrectable Error (R43), Clear Channel (I54/55), and Address Error (R3).

The DD option loads the input TWC register from the HISP input Address Channel. The input Address Channel (bits 0 to 11) is received as terms I30 to I41. Input Address Channel parity errors are received as terms I10 to I13. The DD option does not receive the HISP output Address Channel, but it does receive output Address Channel parity errors (terms I60 to I63). The DD option controls the loading of the Address Channels into the DA and DB options. HISP In Address Ready (term R4) and HISP Out Address Ready (term 45) perform this function. Refer to "HISP Address Channels" in this section for more information.

The DD option tracks the progress of a HISP input transfer by decrementing the TWC register after each word is transferred; when the TWC register reaches 0, the transfer is complete. When the last block of data is received (possibly containing less than 16 words), the DD option activates HISP In Final Block (term R62). The DA option uses this term (along with TWC bits 2^0 to 2^3) to stop memory requests after the last word has been transferred.

For HISP output transfers, the DD option relies on the DA and DB options to keep track of the TWC. The DD option receives Block Length Borrow (terms I75 to I78) from the

DA and DB options. When the TWC reaches 0, each of these terms is logic 1, indicating that the transfer is complete.

HISP channel transfers in progress are aborted under two conditions. First, a channel error causes a transfer to stop. Second, I/O Master Clear (DD option term I81) stops transfers on both HISP channels.

The DD option handles memory access for the HISP channels the same way that the DC option handles LOSP channel memory requests. The DD options sends Memory Reference Request and Memory Write Request (terms R60 and R61) to the DA option. The DD option receives the I/O Channel Designator (terms I70 to I72) and Conflict (term I80) from the DA option. It also receives the I/O References Acknowledge Designators (terms I40 to I43) from the YZ option.

The DD option controls the writing of data from the HISP input channel to the input buffers using six control terms. Five term, R20 to R24 (Buffer Write Address), select a location in the buffers; R24 selects one of the two buffers. R20 to R23 select 1 of 16 locations within the buffer. The sixth term, R26 (Go HISP Data) latches the input data into the selected buffer location.

The DD option controls the writing of data from the output buffers to the HISP output channel. It uses five control terms, R50 to R54 (Buffer Write Address). R54 selects one of the two buffers. R50 to R53 select 1 of 16 locations within the buffer.

The DD option uses several terms to detects uncorrectable data errors during HISP input and output transfers and during VHISP input transfers. It receives Input Channel Syndrome bits (I42 to I49) when a data word is received across a channel. If the number of 1s in the syndrome is even and non-zero, a double-bit error occurred. The DD option uses this information internally and also passes it to the DE option using term R27, Input Channel Double-Bit Error. The DD option receives output channel error information (that is, errors detected when reading from Central Memory) on term I68, Output Channel Double -Bit Error. For maintenance purposes, activating Disable Port D Error Correction (term I85), causes the DD option to ignore double-bit output errors.

DE Option

The DE option is the VHISP channel controller. Two DE options on adjacent CPUs control a single VHISP channel. They perform several major functions:

- They transmit all the control signals needed by the external device and receives all of the control signals sent by the external device.
- They initiate all VHISP data transfers.
- Along with the external device, they control all transfers across the VHISP channel.
- They make memory requests for the VHISP channel.
- They control the writing of data to the input buffers the the reading of data from the the output buffers.

The DE option on each CPU also performs several functions that are not specific to the VHISP channels:

- It responds to CPU instructions 0012*j*0, 0012*j*1, 033*ij*0, and 033*ij*1.
- It generate LOSP and VHISP channel interrupt requests.

The following paragraphs describe the input and output terms of the DE option.

Many of the DE option input and output terms are connected directly to the VHISP channel. Some of the terms are identical on the DE option in both CPUs, while others serve different function in each CPU. Data Ready (I76) and Transmit Enable (I77) are received by both CPUs. However Block Length Error (differential pair I70/71) is received only by the lower-numbered CPU. The higher-numbered CPU receives Acknowledge Clear on I70/71. The separate functions are denoted in the term name by separating them with a "/" (that is, Block Length Error/Acknowledge Clear). Other input terms the operate differently are Uncorrectable Error/Busy (I72/73) and Done and Empty/Offline (I74/75).

All DE option output terms to the VHISP channel operate differently on the two CPUs. These include Clear Control/Null (R20) ("Null" means that the term is not used), Last Buffer Write/ Set CA (R21), and Transmit Enable/Write Data Ready (R22).

The DE options on the two CPUs coordinate channel operations using two sets of cross-coupled control signals. VHISP Data Ready Vote exits each DE option as R23 and enters the other DE option as I78. VHISP Transfer Progress is similarly cross-coupled using terms R24 and I79. Each DE option activates VHISP Data Ready Vote to indicate that it is ready to transfer a buffer (16 words) of data across the channel. For an input transfer, this term indicates that a buffer is empty and ready to be filled. For an output transfer, it indicates that a buffer is full and ready to be emptied. VHISP Data Ready Vote must be active on both CPUs before the transfer can proceed.

The other cross-coupled term, VHISP Transfer Progress consists of two signals that alternate every CP. One signal, Phase 1, can be active only during CPs 1, 3,5, and so on. The other signal, Phase 2, can be active only during CPs 2, 4, 6, and so on. Phase 1 indicates that a VHISP memory reference in progress on the CPU. Phase 2 is transmitted only from the higher-numbered CPU to the lower-numbered CPU. It provides three types off status information to the lower-numbered CPU by the number of CPs that it is active. If Phase 2 is active for 1 CP only, it indicates that Acknowledge Clear has been received from the external device. If Phase 2 is active for 2 alternate CPs (for example, CPs 4 and 6), it indicates that the higher-numbered CPU detected an uncorrectable VHISP data error. Finally, If Phase 2 is active for 3 or more alternate CPs, the higher-numbered CPU is receiving Offline from the external device.

The DE option receives CPU instructions as Channel Command bits 2^0 to 2^4, terms I50 to I54. The Channel Commands are identical to those received by the DA, DB, and DC options, except that the DE option receives them 3 CPs earlier. The DE option responds to VHISP channel instructions 0010*jk* (CA,A*j* A*k*) and 0011*jk* (CL,A*j* A*k*) to begin a channel transfer. For instruction 0010*jk* on the lower-numbered CPU and for instruction 0011*jk* on the higher-numbered CPU, the DE option generates Capture VHISP Control, term R28. This term causes the DA and DB options to transmit the appropriate signal (Block Address or Block Length) to the VHISP channel. For instruction 0011*jk*, the DE option on both CPUs receives A*k* Bit 2^{23} (term I68) to

indicate the transfer direction; the DE option on the higher-numbered CPU generates VHISP Bump CA + 1 (term R27) to increment the CA register to its correct initial value.

The DE option handles memory access requests for the VHISP channels almost exactly the same way that the DC and DD options handle memory requests for their respective channels. The DE options sends Memory Reference Request and Memory Write Request (terms R25 and R26) to the DA option. The DE option receives I/O Channel Designator bits 2^0 and 2^1 (terms I55 and I56) and Conflict (term I80) from the DA option. It also receives the I/O References Acknowledge Designators (terms I40 to I43) from the YZ option. The DE option generates Enable Error Correction (term R10) from the I/O channel designators to indicate that data received from a channel is to be processed by the SECDED logic. R10 is logic 0 for LOSP transfers and logic 1 for HISP and VHISP transfers.

A VHISP channel transfer stops when the BL register decrements to 0. Options DA0, DB0, and DB1 inform the DE when this occurs by sending Block Length Borrow (DE option terms I64 to I66). When all three of these terms are logic 1, the transfer is complete. A VHISP transfer in progress is stopped if the DE option receives I/O Master Clear (term I81) or if a fatal error occurs. Two types of fatal errors can occur. Block length errors are detected by the external device or by the DE option. Uncorrectable data errors are detected by the external device or by the SECDED logic in the mainframe. Uncorrectable errors detected by the mainframe enter the DE option as I0 (Input Channel Double-Bit Error) and I10 (Output Channel Double-Bit Error).

The DE option controls the transfer of data between the VHISP channel and the data buffers. For input transfers, the DE option controls the reading of data from the channel to the input buffers; term R34 selects one of two input buffers and term R29 (VHISP GO Write) latches the data into the buffer. For output transfers. the DE option controls the writing of data from the output buffers to the channel; term R35 selects one of two output buffers. For both input and output transfers, terms R30 to R34 (VHISP Buffer Address) select 1 of 16 locations within the active buffer. Because input and output transfers cannot be active simultaneously, only one set of VHISP Buffer Address terms is needed for both sets of buffers.

The DE option responds to CPU instructions 033*ij*0 (A*i* CA,A*j*) for the LOSP channels and 033*ij*1 (AI CE,A*j*) for the LOSP and VHISP channels. LOSP instruction 033*ij*0 causes the DE option to activate Gate A*i* Data (R0 for the input channel, R1 for the output channel). Gate A*i* Data causes the DA and DB options to output the appropriate CA register. LOSP instruction 033*ij*1 causes the DE option to read the appropriate LOSP Error Flag (I30 for the input channel, I31 for the output channel) and transmit it to A*i* Data Bit 2^0, term R5.

VHISP instruction 033*ij*1 causes the DA and DB options to output the VHISP Status Word. The DE option sends Gate A*i* Data (term R2) to DA and DB options. It also sends VHISP Status Bits 2^{18}, 2^{19}, and 2^{21} to 2^{22} to DB1 and DB2. The DA and DB options combine the VHISP BL register and the Status Bits to form the VHISP Status Word.

The DE option receives LOSP channel interrupt requests from the DC option as terms R82 (input channel) and R83 (output channel). The DE option generates VHISP channel interrupt requests internally. Each of these signals goes to logic 1 when the request is made and remains at logic 1 until the interrupt request is cleared by a 0012*j*0 (CI,Aj) or 0012*j*1 (MC,A*j*) instruction.

When one of the interrupt request signals goes to logic 1, it causes the DE option to activate Interrupt Request (R50-LOSP input; R51-LOSP output; R52-VHISP). Interrupt Request goes to logic for 1 CP to indicate that a request is beginning. The next 3 CPs, Interrupt Request serially transmits a CPU number (most significant bit first). The CPU number is the CPU that most recently cleared the channel's interrupt request (using a 0012*j*0 or 0012*j*1 instruction). (This CPU will normally receive the interrupt request. However, if another CPU has its SEI bit set or is holding issue on a test and set instruction, the interrupt request will be sent there.) After the CPU number is sent, Interrupt Request returns to logic 1.

Interrupt Request remains at logic 1 until cleared by a 0012*j*0 or 0012*j*1 instruction. A 0012j0 or 0012j1 instruction also transmits the number of the CPU that originated the instruction to DE option terms I34 to I36 (CPU Designator), where it is latched and used for the next interrupt request.

HISP Address Channels

Each HISP channel has a 16-bit (1 parcel) address channel which the external device uses to load the SA and TWC registers in the mainframe. Three transfers across the channel are required to load both register. Parcel 1 contains SA register bits only, while parcels 2 and 3 contain SA and TWC register bits. Four parity bits are transmitted with each parcel for error checking.

The HISP input and output address channels are identical. Each uses four DP options , with each DP option receiving four channel bits and the corresponding parity bit. Figures 9-8 and 9-9 show the address channels.

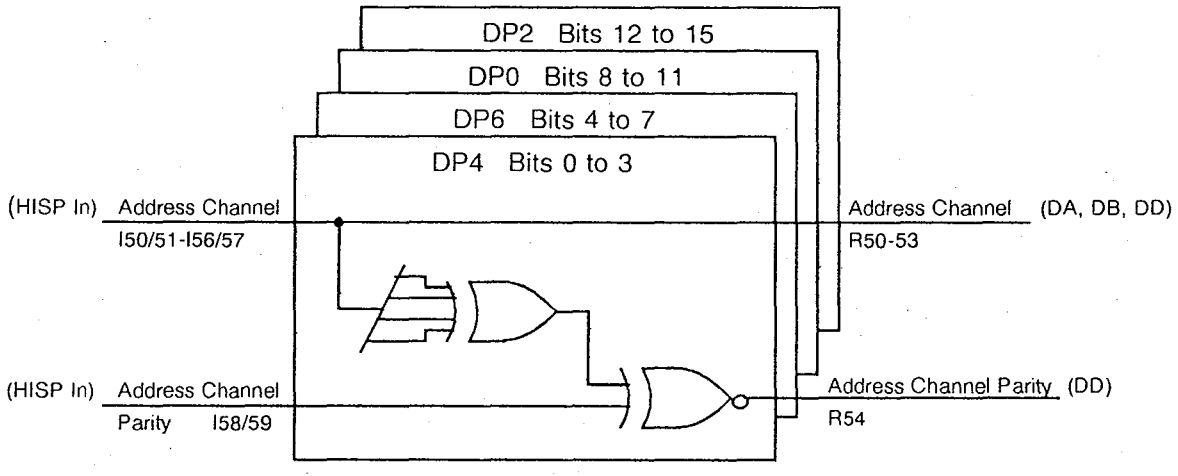


Figure 9-8. HISP Input Address Channel

The DP options pass the address channel bits to the DA, DB, and DD options. Each DP option also performs odd parity checking. If no parity error occurs, the DP option receives an odd number of 1s among the four address bits and one parity bit,and term

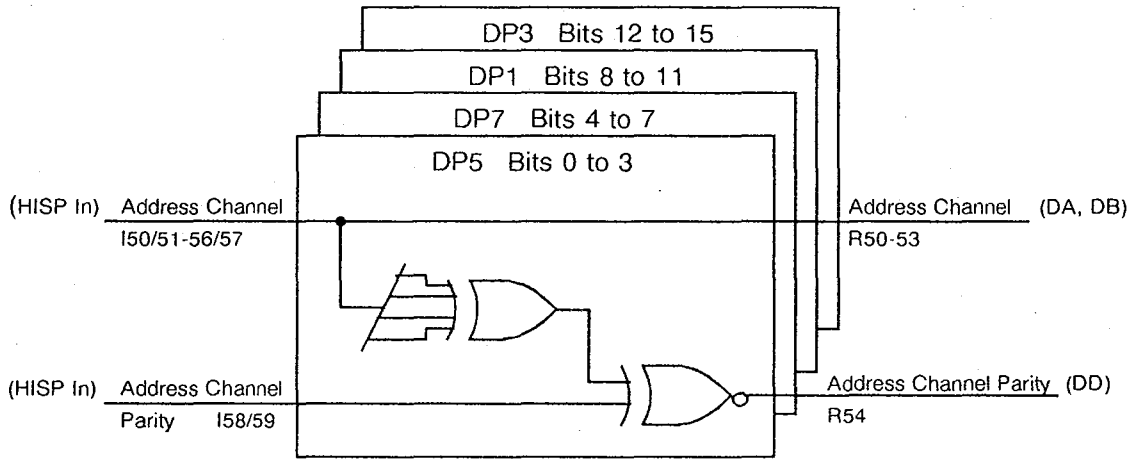


Figure 9-9. HISP Output Address Channel

R54 is logic 0. If an even number of 1s is received, a parity error has occurred and R54 is logic 1.

Because each address channel transfers data in three parcels, each bit may perform up to three functions. Tables 9-10 and 9-11 show the function of each bit in each parcel of the input and output address channels. The tables also show the the destination for each bit (a DA, DB or DD option) and the DP option for each 4 bit group.

Input Data Paths

The input data paths transmit data from the I/O channels to Central Memory. As shown in Figure 9-10, twelve options comprise the input data paths: eight DJ and four YR options. The DJ options receive data from the channels. The YR options generate check bits and, for the HISP input and VHISP channels, perform data error detection and correction. The operation of the DJ and YR options is explained in the following subsections.

DJ Option

The DJ options receive data from the LOSP input, HISP input, and VHISP channels and transmit the channel data to the YR options. The DJ options can receives data from all of the channels simultaneously. However, they can transmit only one data word each CP. The following paragraphs describe how the DJ options receive LOSP, HISP, and VHISP data and transmit the data to the YR options.

The DJ options receive LOSP input data one parcel at a time, and assemble four parcels into a 64-bit word. Four of the DJ options also perform parity checking on the data. Each DJ option receives two data bits from the LOSP input channel as differential pairs I140/141 and I42/I43. DJ4 through DJ7 also receive a parity bit as I144/145. The DJ options receive Sample LOSP Data as term I49. This term, when active for 1 CP, causes

Table 9-9. HISP Input Address Channel Bits

Parcel	Function and Destination															
	DP2 (parity bit 3)				DP0 (parity bit 2)				DP6 (parity bit 6)				DP4 (parity bit 4)			
	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	SA 2^9 DB0 I14	SA 2^{31} Not Used	SA 2^{30} Not Used	SA 2^{29} Not Used	SA 2^{28} DB2 I17	SA 2^{27} DB2 I16	SA 2^{26} DB2 I15	SA 2^{25} DB2 I14	SA 2^{24} DB2 I13	SA 2^{23} DB2 I12	SA 2^{22} DB2 I11	SA 2^{21} DB2 I10	SA 2^{20} DB1 I17	SA 2^{19} DB1 I16	SA 2^{18} DB1 I15	SA 2^{17} DB1 I14
2	SA 2^{13} DB1 I10	SA 2^{12} DB0 I17	SA 2^{11} DB0 I16	SA 2^{10} DB0 I15	SA 2^{16} DB1 I13	SA 2^8 DB0 I13	SA 2^7 DB0 I12	SA 2^6 DB0 I11	SA 2^5 DB0 I10	SA 2^4 DA0 I16	SA 2^3 DA0 I15	SA 2^2 DA0 I14	SA 2^1 DA0 I13	SA 2^0 DA0 I12	TWC 2^{13} DD0 I31	TWC 2^{12} DD0 I30
3	Not Usec	Not Used	SA 2^{15} DB1 I12	SA 2^{14} DB1 I11	TWC 2^{11} DD0 I41	TWC 2^{10} DD0 I40	TWC 2^9 DD0 I39	TWC 2^8 DD0 I38	TWC 2^7 DD0 I37	TWC 2^6 DD0 I36	TWC 2^5 DD0 I35	TWC 2^4 DD0 I34	TWC 2^3 DA0 I13 and DD0 I33	TWC 2^2 DA0 I12 and DD0 I32	TWC 2^1 DA0 I11 and DD0 I31	TWC 2^0 DA0 I10 and DD0 I30

Table 9-10. HISP Output Address Channel Bits

Parcel	Function and Destination															
	DP3 (parity bit 3)				DP1 (parity bit 2)				DP7 (parity bit 1)				DP5 (parity bit 0)			
	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	SA 2^9 DB0 I34	SA 2^{31} Not Used	SA 2^{30} Not Used	SA 2^{29} Not Used	SA 2^{28} DB2 I37	SA 2^{27} DB2 I36	SA 2^{26} DB2 I35	SA 2^{25} DB2 I34	SA 2^{24} DB2 I33	SA 2^{23} DB2 I32	SA 2^{22} DB2 I31	SA 2^{21} DB2 I30	SA 2^{20} DB1 I37	SA 2^{19} DB1 I36	SA 2^{18} DB1 I35	SA 2^{17} DB1 I34
2	SA 2^{13} DB1 I30	SA 2^{12} DB0 I37	SA 2^{11} DB0 I36	SA 2^{10} DB0 I35	SA 2^{16} DB1 I33	SA 2^8 DB0 I33	SA 2^7 DB0 I32	SA 2^6 DB0 I31	SA 2^5 DB0 I30	SA 2^4 DA0 I26	SA 2^3 DA0 I25	SA 2^2 DA0 I24	SA 2^1 DA0 I23	SA 2^0 DA0 I22	TWC 2^{13} DB2 I40	TWC 2^{12} DB2 I39
3	Not Used	Not Used	SA 2^{15} DB1 I32	SA 2^{14} DB1 I31	TWC 2^{11} DB2 I38	TWC 2^{10} DB1 I40	TWC 2^9 DB1 I39	TWC 2^8 DB1 I38	TWC 2^7 DB0 I40	TWC 2^6 DB0 I39	TWC 2^5 DB0 I38	TWC 2^4 DA0 I24	TWC 2^3 DA0 I23	TWC 2^2 DA0 I22	TWC 2^1 DA0 I21	TWC 2^0 DA0 I20

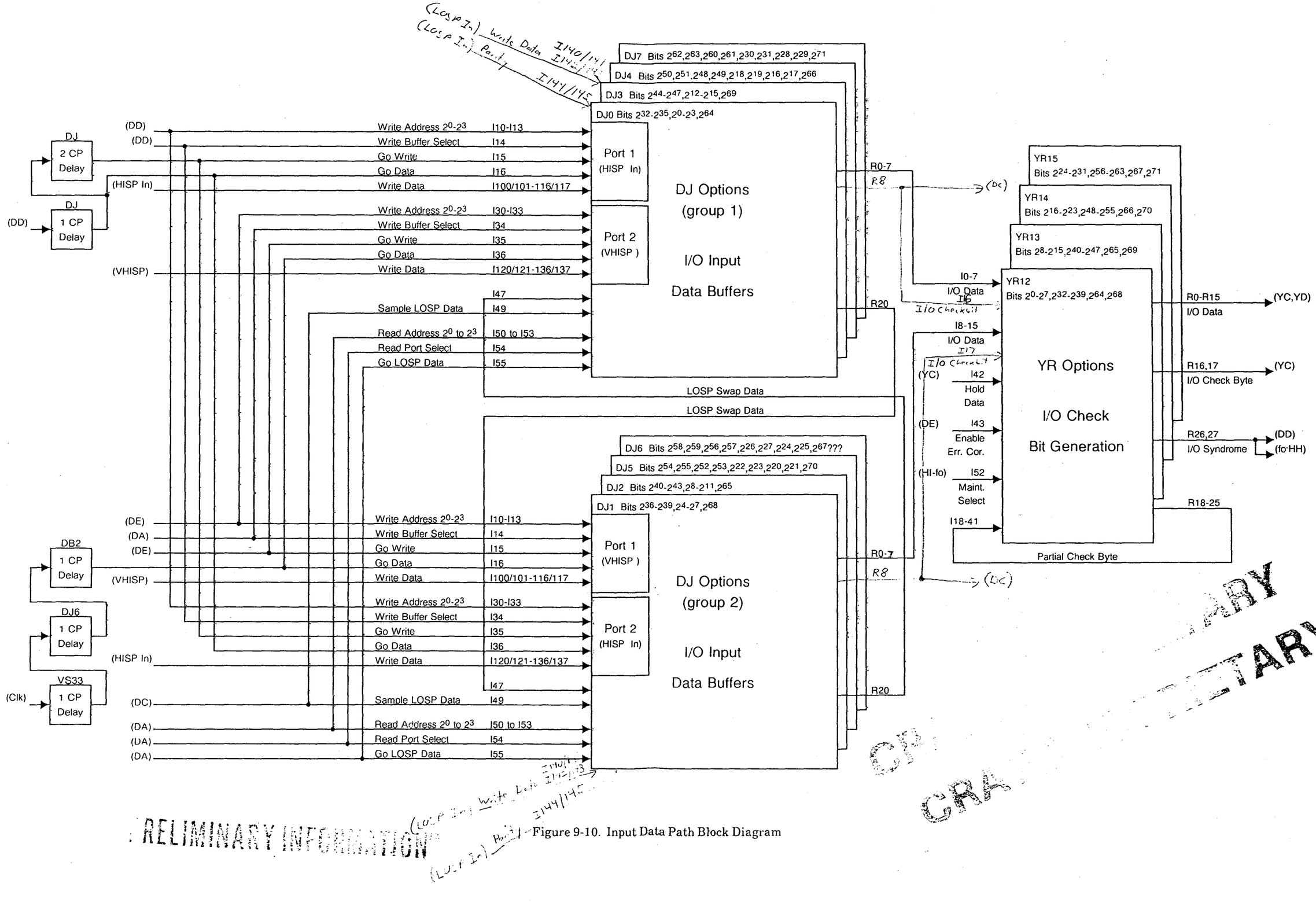


Figure 9-10. Input Data Path Block Diagram

the DJ options to latch the input data. At this time, each DJ option contains two data bits. However, the bits on four of the options are not in the correct position to be sent to the YR options. (For example, after parcel 0 is received, DJ0 contains bits 2^{48} and 2^{49}; these bits must be sent to YR14 from DJ4). An additional problem is that parity checking is done on four-bit groups and cannot be done where only two bits available.

To place the data bits in the proper positions for transmission to the YR options and for parity checking, the DJ options perform a swap sequence after each parcel is received. During the swap sequence each option retains its own data bits, but also swaps data bits with another DJ option. (DJ0 swaps data with DJ4, DJ1 with DJ5, DJ2 with DJ6, and DJ3 with DJ7.) At this point each DJ option contains four data bits, two that it received from the channel and two that it received from another DJ option. This allows DJ4 through DJ7 to perform parity checking and allows four DJ options to latch the data bits. (DJ0 to DJ3 latch the data from parcels 1 and 3; DJ4 to DJ7 latch the data from parcels 0 and 2). The data bits are latched until all four parcels have been received.

During the swap sequence, data is swapped in and out of each option serially, using terms I47 and R20. The two data bits from each option are transmitted in two successive CPs; the most significant bit is transmitted first. The swapped data bits are not sent directly from their source option to their destination option. Instead, the data bits are delayed in the swap data path of an other DJ option. For example, a swap data bit from DJ0 exits term R20 and enters DJ1 term I47. Two CPs later, it exits term R20 and enters DJ4 (its destination) term I 47.

Table 9-11 shows how each bit is swapped. Each entry in the table shows two data bits. The row indicates their parcel and the column indicates their positions within the parcel. The column also indicates the corresponding parity bit and the option which performs the parity checking. For example, data bits 2^{48} and 2^{49} are received as parcel 0, bits 0 and 1. These bits correspond to parity bit 0; parity checking is performed on option DJ4. Within each table entry, below the bits, are three options. The first option is where the bits are received from the channel. The second option is where the bits are delayed during the swap sequence. The third option is where the bits are received at the end of the swap sequence. Using the previous example, bits 2^{48} and 2^{49} are received from the channel on DJ0. During the swap sequence that occurs after parcel 0 is received, these bits exit DJ0, pass though DJ1, and enter DJ4.

Table 9-11 shows that each bit is swapped after it is received from the channel. However, not all swapped data is used. Some of the swapped data bits are used only for parity checking, while other swapped data bits are not used at all. The unused bits are shown in the table.

The DJ options receive Sample LOSP Data, term I47, for 1 CP each time a parcel is available on the channel. This signal causes each option to latch the data and begins a swap sequence. At the end of the swap sequence, the data is latched again; each bit is now correctly positioned for transmission to the YR options. A parity check is also performed at the end of the swap sequence; a parity error sets a flag in DJ4, DJ5, DJ6, or DJ7. Four swap sequences are required to latch an entire 64-bit word; a 2-bit counter in each DJ option keeps track of the number of swap sequences that have occurred. The parity error flags are cleared and the swap sequence counters are reset to 0 when term I47 is received for 2 or more successive CPs; this occurs when the channel is initialized.

The DJ options receive HISP and VHISP data in complete words; 64 data bits and 8 check bits are received simultaneously. Each DJ option receives nine bits, eight data bits and one check bit. The DJ options contain the HISP and VHISP input data buffers.

Table 9-11. LOSP Input Data Swap

Parcel	Parcel Bits							
	Parity Bit 0 (DJ4)		Parity Bit 1 (DJ5)		Parity Bit 2 (DJ6)		Parity Bit 3 (DJ7)	
	0,1	2,3	4,5	6,7	8,9	10,11	12,13	14,15
0	Bits 2^{48}, 2^{49} DJ0 →DJ1 →DJ4	Bits 2^{50}, 2^{51} DJ4 →DJ5 →DJ0‡	Bits 2^{52}, 2^{53} DJ1 →DJ4 →DJ5	Bits 2^{54}, 2^{55} DJ5 →DJ0 →DJ1‡	Bits 2^{56}, 2^{57} DJ2 →DJ3 →DJ6	Bits 2^{58}, 2^{59} DJ6 →DJ7 →DJ2‡	Bits 2^{60}, 2^{61} DJ3 →DJ6 →DJ7	Bits 2^{62}, 2^{63} DJ7 →DJ2 →DJ3‡
1	Bits 2^{32}, 2^{33} DJ0 →DJ1 →DJ4†	Bits 2^{34}, 2^{35} DJ4 →DJ5 →DJ0	Bits 2^{36}, 2^{37} DJ1 →DJ4 →DJ5†	Bits 2^{38}, 2^{39} DJ5 →DJ0 →DJ1	Bits 2^{40}, 2^{41} DJ2 →DJ3 →DJ6†	Bits 2^{42}, 2^{43} DJ6 →DJ7 →DJ2	Bits 2^{44}, 2^{45} DJ3 →DJ6 →DJ7†	Bits 2^{46} , 2^{47} DJ7 →DJ2 →DJ3
2	Bits 2^{16}, 2^{17} DJ0 →DJ1 →DJ4	Bits 2^{18}, 2^{19} DJ4 →DJ5 →DJ0‡	Bits 2^{20}, 2^{21} DJ1 →DJ4 →DJ5	Bits 2^{22}, 2^{23} DJ5 →DJ0 →DJ1‡	Bits 2^{24}, 2^{25} DJ2 →DJ3 →DJ6	Bits 2^{26}, 2^{27} DJ6 →DJ7 →DJ2‡	Bits 2^{28}, 2^{29} DJ3 →DJ6 →DJ7	Bits 2^{30}, 2^{31} DJ7 →DJ2 →DJ3‡
3	Bits 2^{0}, 2^{1} DJ0 →DJ1 →DJ4†	Bits 2^{2}, 2^{3} DJ4 →DJ5 →DJ0	Bits 2^{4}, 2^{5} DJ1 →DJ4 →DJ5†	Bits 2^{6}, 2^{7} DJ5 →DJ0 →DJ1	Bits 2^{8}, 2^{9} DJ2 →DJ3 →DJ6†	Bits 2^{10}, 2^{11} DJ6 →DJ7 →DJ2	Bits 2^{12}, 2^{13} DJ3 →DJ6 →DJ7†	Bits 2^{14}, 2^{15} DJ7 →DJ2 →DJ3

† These bits are swapped only for parity checking.
‡ These bits are used only in the option where they were received from the channel.

Each DJ option contains two inputs data ports. The HISP channel uses port 1 on DJ0, DJ3, DJ4, and DJ7 (option group 1) and port 2 on DJ1, DJ2, DJ5, and DJ6 (option group 2). The VHISP channel uses the remaining port on each option. Each port contains two 16-word data buffers. Within each port, one buffer is selected to receive data from the channel; the other buffer is then automatically selected to transmit data to the YR options.

HISP channel data is received on differential pairs I100/101 to I116/117 in the group 1 options and on I120/121 to I136/137 in group 2. When the HISP data is ready, the options receive Go Data (I16 in group 1, I36 in group 2), which latches the data in the options. Two CPs later, they receive Go Write (I15 or I35). This signal writes the data into one of the two buffers. Write Buffer Select (I14 or I34) selects the buffer and Write Address 2^0 to 2^3 (I10 to I13 or I30 to I33) select one word within the buffer.

VHISP channel data is received on differential pairs I120/121 to I136/137 in the group 1 options and on I100/101 to I116/117 in group 2. The options receive Go Data (I36 in group 1, I16 in group 2) once every 2 CPs, which latches data in the options regardless of whether the channel is active. However, the DJs receive Go Write (I15 or I35) only when data is being transmitted from the SSD. This signal writes the data into one of the two buffers. Write Buffer Select (I34 or I15) selects the buffer and Write Address 2^0 to 2^3 (I30 to I33 or I10 to I13) select one word within the buffer.

The DJ options can transmit one word of LOSP, HISP, or VHISP data to the YR options during each CP. When term I55, GO LOSP Data, is active, LOSP Data is selected. When term I55 is not active, HISP or VHISP data is selected. Term I54, Read Port Select, selects between HISP and VHISP data. When HISP or VHISP data is selected, terms I50 to I53, Read Address 2^0 to 2^3, select 1 of 16 words in a buffer. There is no input term to select a HISP or VHISP buffer; for each channel the buffer that is not selected for writing channel data is automatically selected for reading.

I/O data exits the DJ options as terms R0 to R7. When HISP or VHISP data is selected, term R9 is a check bit. When LOSP data is selected, R8 on DJ4 to DJ7 is a parity error flag; R8 on DJ0 to DJ3 is not used.

YR Option

The YR options receive channel data from the DJ options and transmit the data to Central Memory. When processing LOSP data, the YR options generate a check byte. When processing HISP and VHISP data, the YR options perform error detection and correction on the channel data, generating corrected data and check bits.

The YR options receive I/0 data as terms I0 to I15. HISP and VHISP check bits are received as terms I16 and I17. I16 and I17 are not used for LOSP data. Term I43, Enable Error Correction, is active for HISP and VHISP data and inactive for LOSP data. When this term is active, it causes the YR options to use the data and check bits received from the channel, perform error detection, and generate corrected data and check bits. When I43 is inactive, the YR options simply generate check bits from the data received from the channel. Term I42, Hold Data, is active during Port D memory conflicts. It causes I/O data to be held in the YR options.

YR option terms R0 to R15 are the I/O data bits for Central Memory. Terms R16 and R17 are the check bits. Terms R26 and R27 are syndrome bits; they are always 0 for

LOSP data. Terms R18 to R25 are partial check bytes which feed into other YR options as terms I18 to I41.

A maintenance feature is built into the YR options. When term I52, Maintenance Select, is active, normal error detection and check bit generation is disabled. Data bits 2^0 to 2^7, 2^9 to 2^{15}, 2^{17} to 2^{23}, 2^{25} to 2^{31}, 2^{33} to 2^{39}, 2^{41} to 2^{47}, 2^{49} to 2^{55},and 2^{57} to 2^{63} pass through without correction. Data bits 2^0, 2^8, 2^{16}, 2^{24}, 2^{32}, 2^{40}, 2^{48}, and 2^{56} are replaced by uncorrected check bits 0 to 7. The corrected check bits are replaced by the syndrome. The syndrome outputs are forced to 0.

Output Data Paths

The output data paths transmit data from Central Memory to the I/O channels. As shown in Figure 9-11, the input data paths consist of eight DP options. The operation of the DP options is explained in the following paragraphs.

The DP options receive data from Central Memory and transmit the data to the LOSP output, HISP output, and VHISP channels. For LOSP data, the DP options disassemble a data word into four parcels.and generate four parity bits for each parcel. For HISP and VHISP data, the DP options provide buffers between Central Memory and the channels. The DP options can receive only one data word from Central Memory each CP. However, they can transmit data to all the channels simultaneously.

The DP options receive data from Central Memory as terms I0 to I8. I0 to I7 are data bits and I8 is a check bit. If the data is for LOSP output channel, the DP options receive Sample LOSP Data, term I15. This term latches the data into the options and begins the data disassembly process. At this time, each DP option contains eight data bits and has generated two parity bits. However, half of the data bits are not in the correct option to be sent to the LOSP channel. (For example, DP0 contains bits 2^0 to 2^3 and 2^{32} to 2^{35}; bits 2^2, 2^3, 2^{34}, and 2^{35} are sent to the channel from DP4.) An additional problem is that DP0 to DP3 must send all 16 parity bits (4 bits$\times$4 parcels) to the channel; DJ4 to DJ7 now contain eight of the parity bits.

To place the data and parity bits in the proper positions for transmission to the LOSP channel, the DJ options perform a data swap sequence. During the swap sequence each option retains four of its data bits, but swaps the other four data bits with another DP option. (DP0 swaps data with DP4, DP1 with DP5, DP2 with DP6, and DP3 with DP7.) The same pairs of DP options also swap parity bits. However, each option keeps its own parity bits in addition to swapping them. At the end of the swap sequence, all of the data bits are in the correct options and DJ0 to DJ3 contain all of the parity bits. (DJ4 to DJ7 also contain all the parity bits, but they are not used.) Data and parity bits are sent out to the channel one parcel at a time; a new parcel is sent each time the DP options receive LOSP Data Advance, term I40. The LOSP data bits are transmitted through terms R30 and R31. Parity bits are transmitted through term R32 on DP0 to DP3.

During the swap sequence, data and check bits are swapped in and out of each option serially, using terms I41 and R35. The four data bits and two parity bits from each option are transmitted in six successive CPs. The swapped data bits are not sent directly from their source option to their destination option. Instead, the data bits are delayed in the swap data path of an other DJ option. For example, a swap data bit from DP0 exits term R35 and enters DJ1 term I41. Six CPs later, it exits term R35 and enters DJ4 (its destination) term I 41.

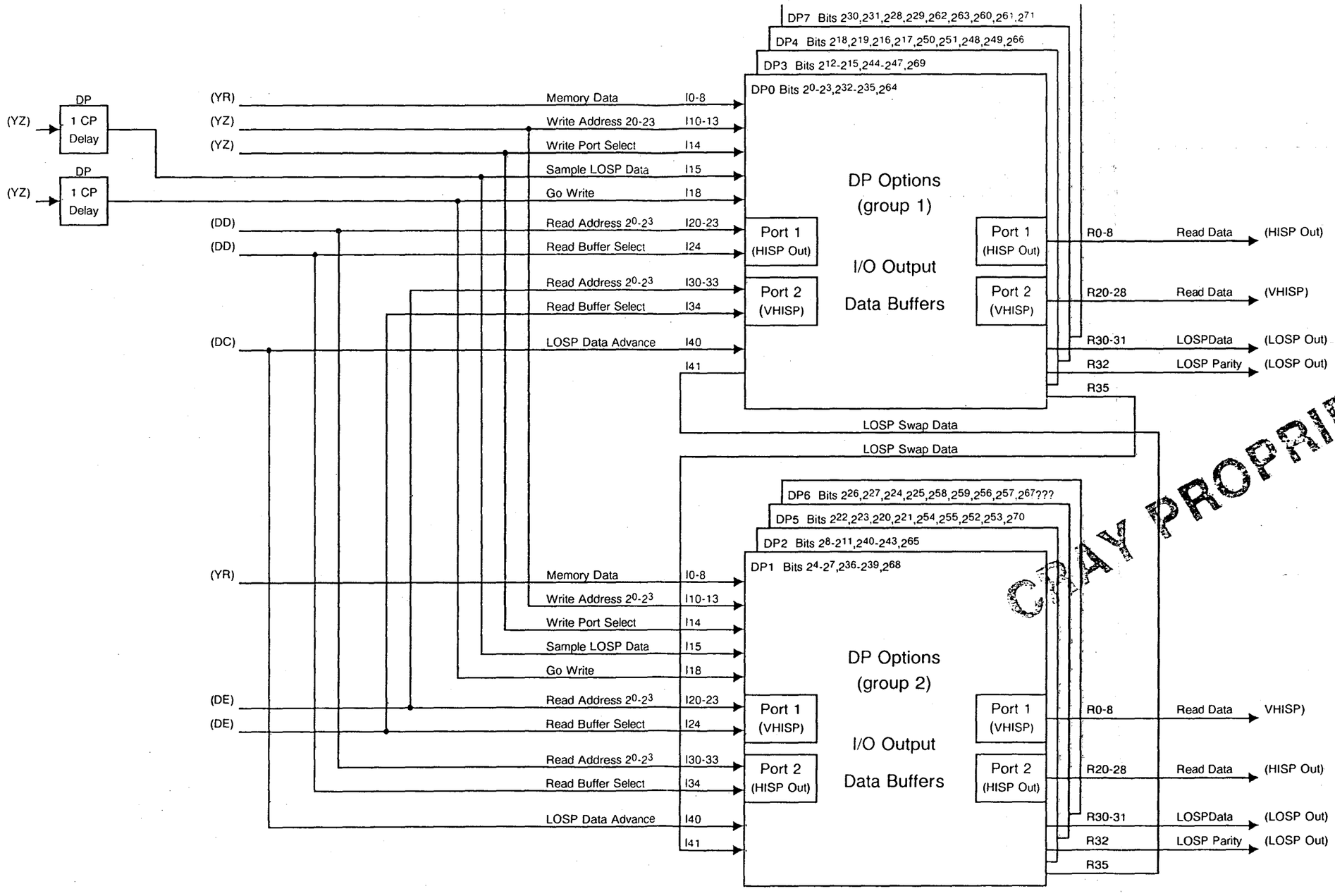


Figure 9-11. Output Data Path Block Diagram

Table 9-12 shows all the data and parity bits that are swapped. The first column shows the six bits that are swapped out of each option in the order that they are transmitted. The second column shows the options where the swap bits originate, pass through, and terminate.

Table 9-12. LOSP Output Data Swap

Bits	Swap Path
parcel 1 parity 0, data 2^{35}, 2^{34}, parcel 3 parity 0, data 2^{3}, 2^{2}	DP0→DP1→DP4
parcel 1 parity 1, data 2^{39}, 2^{38}, parcel 3 parity 1, data 2^{7}, 2^{6}	DP1→DP4→DP5
parcel 1 parity 2, data 2^{43}, 2^{42}, parcel 3 parity 2, data 2^{11}, 2^{10}	DP2→DP3→DP6
parcel 1 parity 3, data 2^{47}, 2^{46}, parcel 3 parity 3, data 2^{15}, 2^{14}	DP3→DP6→DP7
parcel 0 parity 0, data 2^{49}, 2^{48}, parcel 0 parity 0, data 2^{17}, 2^{16}	DP4→DP5→DP0
parcel 0 parity 1, data 2^{53}, 2^{52}, parcel 2 parity 1, data 2^{21}, 2^{20}	DP5→DP0→DP1
parcel 0 parity 2, data 2^{57}, 2^{56}, parcel 2 parity 2, data 2^{25}, 2^{24}	DP6→DP7→DP2
parcel 0 parity 3, data 2^{61}, 2^{60}, parcel 2 parity 2, data 2^{28}, 2^{29}	DP7→DP2→DP3

If the Central Memory data received by the DP options is for the HISP or VHISP channels, the data is sent to one of two ports. On DP0, DP3, DP4, and DP7 (option group 1), Port 1 receives HISP data and Port 2 receives VHISP data. On DP1, DP2, DP5, and DP6 (option group 2) Port 2 receives HISP data and Port 1 receives VHISP Data. When memory data is received, term I14, Write Port Select, sends the data to the proper port. Go Write, term I18, writes the data into the port. Within each port are two 16-word data buffers. At any time, one buffer is selected to transmit data to the channel. The other buffer is then automatically selected to receive data from memory. Write Address, terms I10 to I13, selects one word in the receiving buffer.

Data is read out of the buffers using two sets of control signals. Read Buffer Select (term I24 for port 1, term I34 for port 2) selects one of the two buffers in the port. Read Buffer Select 2^{0} to 2^{3} (I20 to I23 for port 1, I30 to I34 for port 2) select one of 16 words. Ports 1 read data exits as terms R0 to R8. Port 2 read data exits as R20 to R28.

CONTENTS

FIGURES

TABLES

11 - SYSTEM CLOCK

The entire mainframe is synchronized by the system clock. The System Clock module generates the system clock and fans out the clock signal to all other mainframe modules and the SSD solid-state storage device.

The System Clock module consists of one single-sided printed circuit board mounted in the "C" slot of the module chassis. It has provisions for containing up to seven crystal oscillators. It is not required to have all seven oscillators on the System Clock module. The system clock can also be supplied to the system by use of an external clock produced by a stand-alone generator. The external clock allows for variable clock speed settings.

OPTION DESCRIPTIONS

The following subsections provide a brief description of the options located on and associated with the System Clock module.

TM Option

The TM option is used exclusively on the CPU module. There is one TM option per board for a total of four per CPU module.

The TM option provides the instruction buffers (HS options), I/O data input buffers (DJ options), I/O data output buffers (DP options), and the vector registers (VS options) with a Write Enable signal and a Clock pulse.

TO Option

The TO option contains fanouts for the system clock. Each TO option has four internal levels of fanout. Option TO10 contains fanouts for forced zeros and ones.

TP Option

The TP option functions as a Clock Pulse/Timing generator and contains the first-level fanout for the system clock. The TP option interprets the Clock Select signal generated by the deadstart panel switch settings and fans out the selected clock speed to the TO option. The TP option also performs a divide-by-two on the mainframe clock that is used in data transfers over the 1000 Mbyte/s very high-speed (VHISP) channels to the SSD.

ZA Option

The ZA option contains 4 one-to-eight or 34 one-to-one fanouts. There are four groups with eight fanouts in each group. Each group has a forced 1 that allows the fanouts in the group to be active. In addition to the four groups of eight, there are two independent fanouts that do not require an enable term. The ZA option allows for the clock signal comparison of the SSD clock and CRAY Y-MP clock.

CLOCK SPEED SELECT

The System Clock module may contain up to seven crystal oscillator-controlled frequencies and a provision for running from an external frequency source. The clock sources are switch selectable from the deadstart panel.

Figure 11-1 is a block diagram of the System Clock fan-out scheme. The desired clock speed is selected by setting the mainframe deadstart panel switches to the appropriate positions shown in Table 11-1. Each switch generates its own I term (I0 through I2 for switches 0 through 2 respectively) that are sent to edge connector CN4 on the System Clock module. Edge connector CN4 sends the Clock Select signal to option TP0. The Clock Select signal on option TP0 selects one of seven clock speeds or external clock. The seven clock speeds are generated from the oscillators and the external clock is generated from an external generator. A forced 1 on option TP0 input I18 disables the test path.

An external signal generator may be used to provide clock signals through an external input. The external signal generator is cabled to the System Clock module through a conhex (defined in the "System Clock Module" subsection in Section 1 of this manual) located on the front of the module. The external source then determines the frequency of the system clock.

The external clock provides a backup in case of crystal oscillator failure. It is also useful when conducting tests that require frequencies not provided by the seven oscillators.

CRAY Y-MP MAINFRAME TO SSD DATA TRANSFER CLOCK

The I18 term enables an SSD Divide-by-two signal on the TP0 option. This signal provides a clock for the data transfer over the VHISP channel to the SSD. The CRAY Y-MP mainframe clock is divided by two because the SSD operates at one half the rate of the CRAY Y-MP mainframe's clock.

The clock pulse fanned out by option TO6 through edge connector CN13 on the clock module is sent back onto the clock module through edge connector CN19. Edge connector CN19 to TO8 to ZA0 network produces a simulated clock. This simulated clock network serves as a reference point and is used for clock tuning on the VHISP.

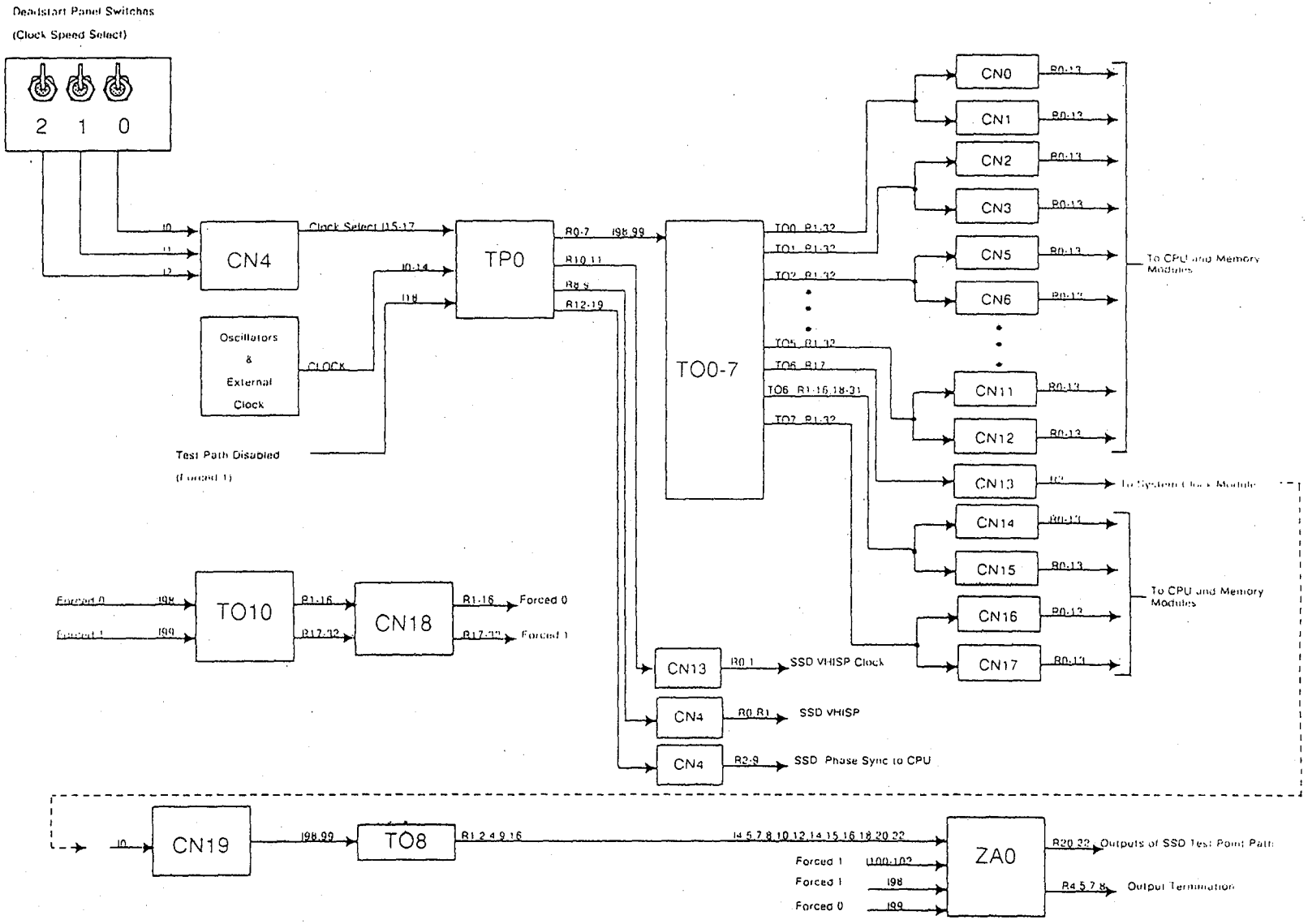


Figure 11-1. System Clock Block Diagram

Table 11-1. CRAY Y-MP Clock Speeds

Panel Switches			Description	Speed
2^2	2^1	2^0		
0	0	0	Super Slow (S Slow)	to be determined
0	0	1	Slow	to be determined
0	1	0	Normal (Norm)	6.0 ns
0	1	1	Fast	to be determined
1	0	0	Auxiliary 0 (Aux0)	to be determined
1	0	1	Auxiliary 1 (Aux1)	to be determined
1	1	0	External Clock (Ext)	to be determined
1	1	1	Auxiliary 3 (Aux3)	to be determined

FORCED 0 AND 1 FANOUTS

The TO10 option to edge connector CN18 circuitry sends forced 0s and 1s to the memory modules. The forced 0s and 1s control memory access timing. Forced 0 terms R1 through R16 control the CPU Bank Busy Select, CPU Subsection Busy Select, Select Delay, and Bank Busy Time Select Bit 1 signals generated by the memory module options. Forced 1 terms, R17 through R32, control the Bank Busy Time Select Bit 0, Select 3 CP Wide Write Enable Pulse, and Force 1 Fanout signals. For a detailed description of these signals and their functions, refer to Section 2 in this manual.

FAN-OUT SCHEME

The output terms, R0 through R7, from the TP0 option, are the first-level clock fanouts sent to the TO0 option through TO7 options on the System Clock module. Output terms R8 through R11 are copies of the SSD clock described in the "CRAY Y-MP Mainframe to SSD Data Transfer Clock" subsection. One copy is assigned to one of four 1000-Mbyte/s channels. Option TP0 outputs terms R12 through R19, SSD Clock Phase Sync signals, to edge connector CN4. The CRAY Y-MP mainframe decides which half of the divided CRAY Y-MP System Clock the SSD is using for 1000 Mbyte/s data transfer into the SSD by referencing the SSD Clock Phase Sync signal. These signals are transferred from edge connector CN4 on the System Clock module to the CPU module as terms R2 through R9.

Options TO0 through TO7 are clock fan-out chips. They fan out copies of the clock to all CPU and memory modules. All CPU and memory modules contain TO options on boards A through D. The TO options on the CPU and memory modules receive clock signal fanouts from the TO options located on the System Clock module. TO options on CPU and memory modules supply clock pulses to all the chips on their respective boards. Each TO option has a possible 32 outputs (terms R1 through R32). Refer to Table 11-2 for option TO0 through TO7 clock signal source and destination information.

Table 11-2. TO0 through TO7 Fanouts

Option	Connectors	Destination Modules (Chassis Slots)
TO0	CN0, CN1	Slot #s 14-16 and 25-28
TO1	CN2, CN3	Slot #s 7-13
TO2	CN5, CN6	Slot #s 1-6 and 17
TO3	CN7, CN8	Slot #s 18-24
TO4	CN9, CN10	Slot #s 34-40
TO5	CN11, CN12	Slot #s 29-33
TO6	CN13	Slot # 41
TO6	CN14, CN15	Slot #s 17-22 and 24
TO7	CN16, CN17	Slot #s 17-23

For module chassis locations, refer to Figure 1-5.

A Simulated Clock signal is generated by option TO6, term R17, and is sent through edge connector CN13 on the clock module. This signal is routed back onto the clock module through edge connector CN19. Option TO6 sends write enable signals through edge connectors CN14 and CN15 to the TM options on the CPU boards. The TM options send the write enable signals to the Instruction buffer (HS option), I/O Data Input buffer (DJ option), I/O Data Output buffer (DP option), and Vector register (VS option), and allow data to be written into those registers. Edge connector CN14 routes the TO6 Write Enable clock signal to the previously described registers located on CPU 2 (boards C and D) and CPU's 3 through 5 (boards A through D). CN15 routes the TO6 Write Enable signal to those same registers on CPU's 0 and 1 (boards A through D) and CPU 2 (boards A and B). Option TO6 also fans out the clock to the TM options on boards A through D of the CPU 7 module through clock module edge connector CN15.

CLOCK MODULE TUNING

Clock module tuning permits all clock outputs that leave the module to be synchronized. The TO option chips responsible for fanning out the clock signal to the CPU and memory modules each have true (terms R0 through R13) and false (terms R0' through R13') outputs. The outputs leave the board through the edge connectors and enter the CPU and memory modules through the edge connectors on those boards. The TP option signals fanned out to the TO options and the 225 output pairs (true and false outputs) of the TO fan-out options are tunable.

There are 16 tuning structures; each is designated to and located near its own edge connector. The tuning structures consist of pads and foil on layer one of the printed circuit board. The pads provide a land to solder that is capable of withstanding several tuning cycles without lifting foils. Figure 11-2 shows the physical layout of a tuning structure on the System Clock module board. Edge connectors CN4 and CN13 have no tuning structures.

Clock module tuning is done by modifying the foil lengths of the tuning structures. Approximately 50 ps may be added or subtracted from the delay by adding or deleting foil length from the tuning circuit path .

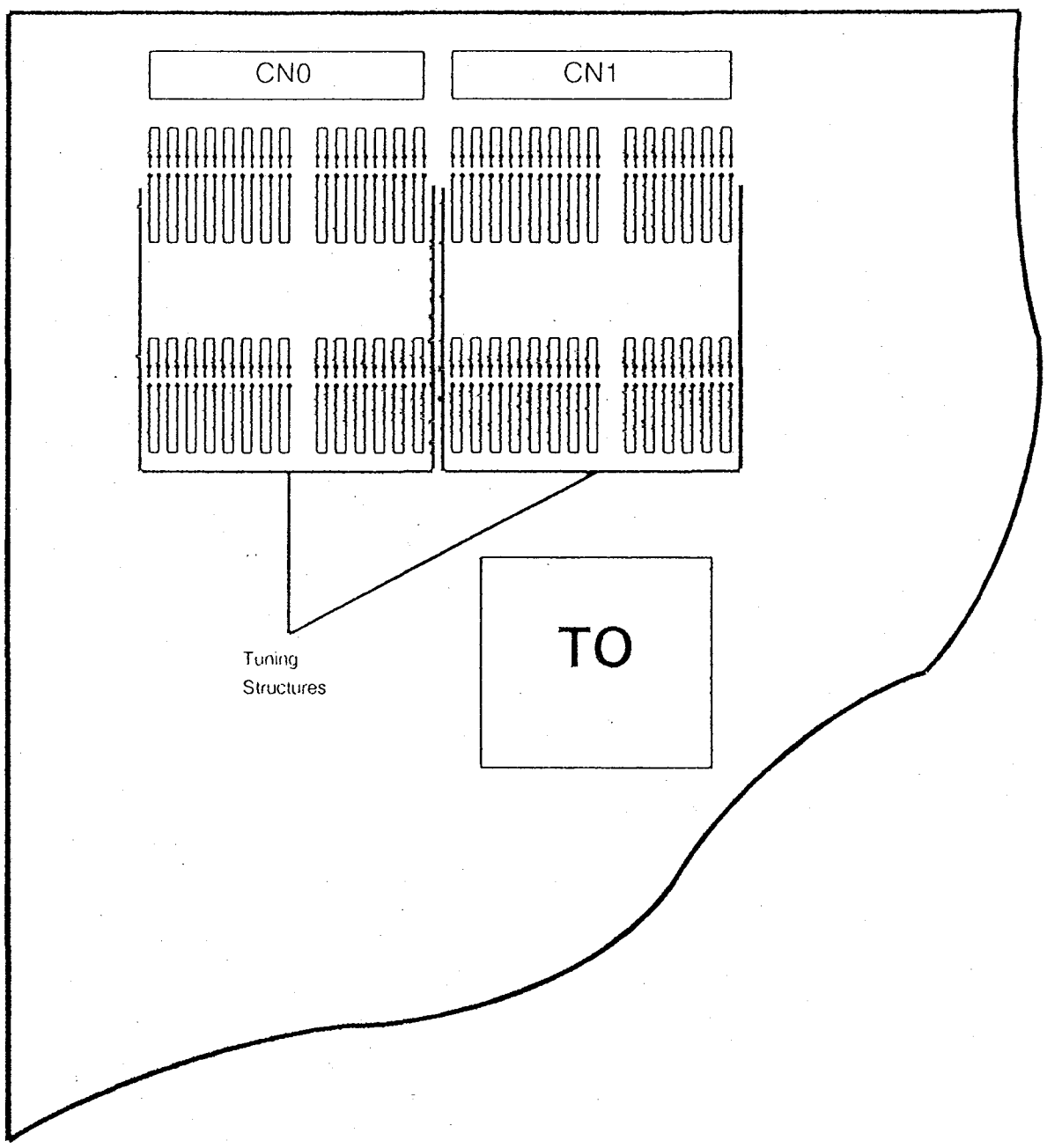


Figure 11-2. System Clock Module Tuning Structures

Table of Contents

11 - MCA2500ECL MACROCELL ARRAY

This section describes the concepts of the Motorola MCA2500ECL Macrocell Array used by Cray Research, Inc. (CRI) in the CRAY Y-MP computer system. Some of the information in this section is taken from the *Motorola MCA2500ECL Macrocell Array Design Manual*, BR165 Rev. 1. Reprint permission was granted by the Technical Communications department of Motorola Semiconductor Products, Inc., 3102 North 56th Street, Phoenix, AZ 85010.

The following conventions are used by CRI:

- Negative logic; that is, a logic 0 has a voltage range of -0.810 V to -0.960 V and a logic 1 has a voltage range of -1.65 V to -1.95 V.
- CRI provides a power supply voltage of -4.5 V for the MCA2500ECL Macrocell Array.

MACROCELL ARRAY OVERVIEW

A Macrocell Array is similar to a gate array; but instead of gates, the Macrocell Array contains unconnected transistors and resistors. When the transistors and resistors are interconnected, they form specific logic functions called macros. The macros occupy the cells which make up the Macrocell Array. There are more than 90 macros stored within the Macrocell library, some of which CRI has custom designed. For more information on macros, read the "Macrocell Library" subsections in this section.

The Macrocell Library and the specifications for interconnecting the macros are stored within a computer aided design (CAD) system. Using the CAD system, Motorola designs the prefabricated chips, and Cray customizes them to meet the CRAY Y-MP specifications. When a chip is customized it is called an option.

The information in the following subsections applies to Motorola's prefabricated chip; some of Motorola's design features are not used by CRI.

THE MCA2500ECL MACROCELL ARRAY LAYOUT

The MCA2500ECL Macrocell Array has 178 cells; 110 major cells and 68 output cells. Figure 11-1 shows the organization of the major cells and output cells.

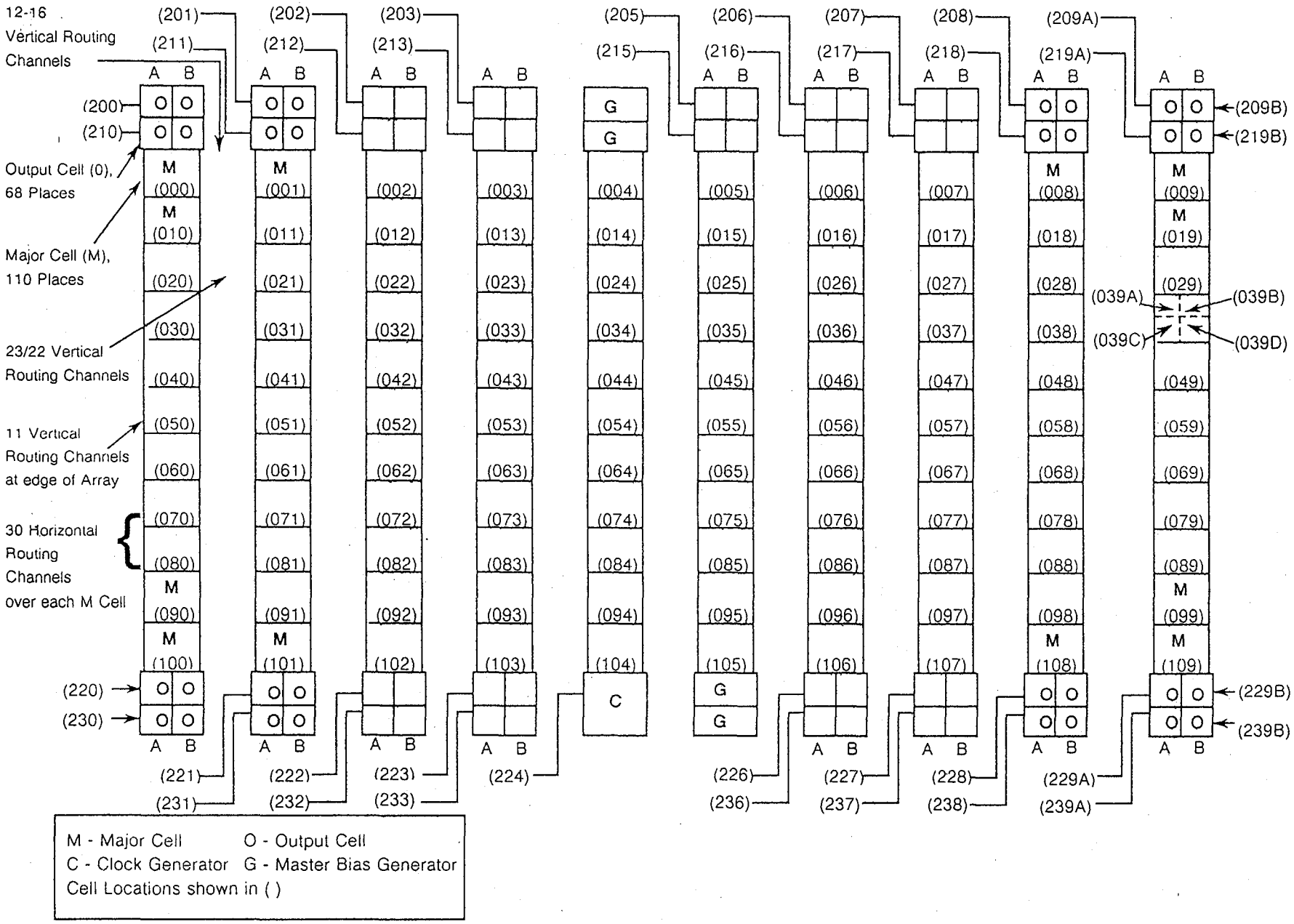


Figure 11-1. MCA2500ECL Macrocell Array Layout *

* Copyright by Motorola, Inc. Used by Permission.

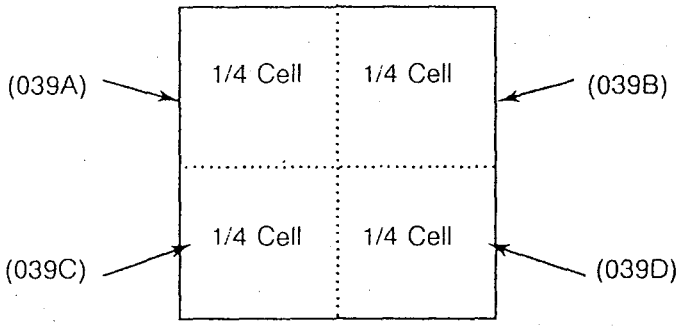
Figure 11-1 also illustrates the Clock Pulse Generator cell, the cell locations, and the routing channels. A description of these features is included in the following subsections:

- Major cells
- Output cells
- Clock Pulse Generator cell

Major Cells

Major cells are located in the internal area of the chip and are used for the majority of logic functions. There are 110 major cells in the Macrocell Array. Each major cell consists of 56 transistors and 56 resistors. The transistors and resistors are interconnected to form the major cell macros.

The major cells can be divided into independent half cells, quarter cells, three-quarter cells, or one full cell. Figure 11-2 illustrates the divisions of a major cell.



Cell Locations shown in ( )

Figure 11-2. Major Cell Division

Each major cell macro specifies how much of the major cell is needed to implement the macro function (refer to the "Macrocell Library" in this section). For example, the major cell macro M203 (eight input AND/NAND) requires one-half of a major cell. The other half of the cell could be used to implement another major cell macro requiring one-half of a major cell such as M285 (3-bit adder sum), or two macros requiring one-quarter of a major cell such as M284 (2-bit adder sum) and M228 (1,2 XOR). The arrangement of macros in the major cells is determined by the option designer.

Option inputs connect directly to the major cells. For more information on input and output pins, read "MCA2500ECL Macrocell Array Packaging" in this section.

Output Cells

Output cells are located at the top and bottom of the Macrocell Array. Output cells are used to interface the internal logic of a chip to logic outside the chip by providing 60 ohm drive capability. Each output cell contains 15 transistors and 10 resistors that are interconnected to form logic functions.

In the Macrocell Library, the primary outputs of an output cell are designated with an "X", and must be connected to the base of an output emitter follower located near the bonding pad. Outputs labeled "Z" are used to drive internal loads. All outputs are terminated to the -2.0 V power supply through 60 ohm resistors.

Clock Pulse Generator Cell

The Clock Pulse Generator cell is located on the bottom of the array, and is labeled with a "C" (refer to figure 11-1, the MCA2500ECL Macrocell Layout). The Clock Pulse Generator eliminates the necessity of supplying a narrow clock pulse to the chip. Narrow pulses are difficult to distribute between chips because pulse width shrinkage can easily occur.

CRI does not use the Clock Pulse Generator cell for the CRAY Y-MP computer system. Instead, the major cells are used to generate the clock on the chip. CRI uses two types of clock pulse circuits. Both circuits operate the same; however, the alternate clock circuit uses the inverted side of the system clock. Each option has its own clock pulse circuit. Figures 11-3 and 11-4 show the clock pulse circuits and timing diagrams. I98 is the raw clock and establishes the leading edge of the clock pulse. I99 is the pulse shaper and establishes the trailing edge of the clock pulse. It is delayed approximately 150 picoseconds from I98. I99, T100, and T101, form the pulse shaper. The pulse shaper, and I98 form the narrow clock pulse T0. T0 is delayed to form T1 which is delayed to form the narrow clock pulse T2.

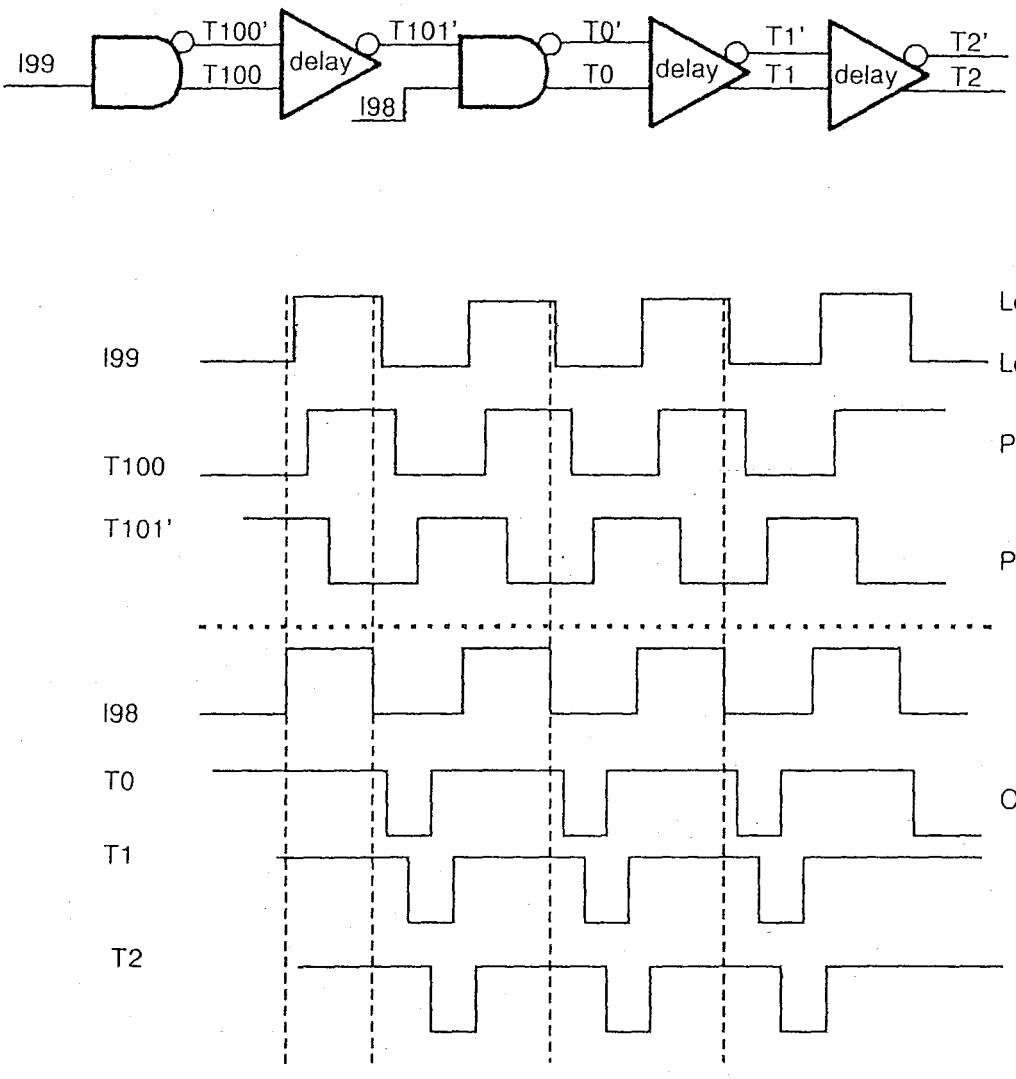


Figure 11-3. Cray Y-MP Clock Pulse Circuit

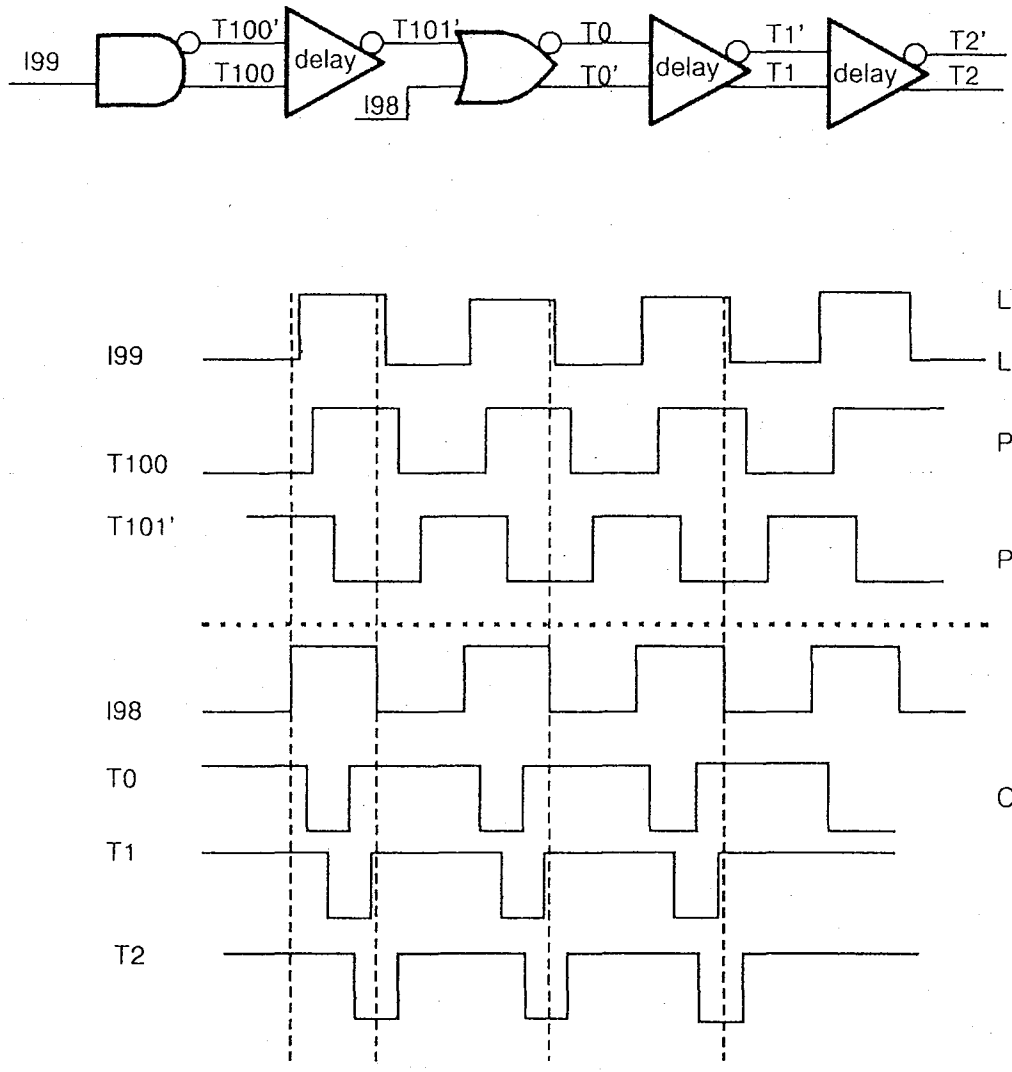


Figure 11-4. Alternate Cray Y-MP Clock Pulse Circuit

MACROCELL ARRAY PACKAGING

The MCA2500ECL Macrocell Array is packaged in a 148-pin package. Figure 11-4 shows a top view of the pin package and how the pins are numbered. There are 28 power pins, 52 designated input pins, and 68 pins that can be used for either input or output. Table 11-1 lists the pin numbers and their functions. The pins used for input or

output are located on the top and bottom of the package because of the location of the output cells. The dot indicates the location of pin 1.

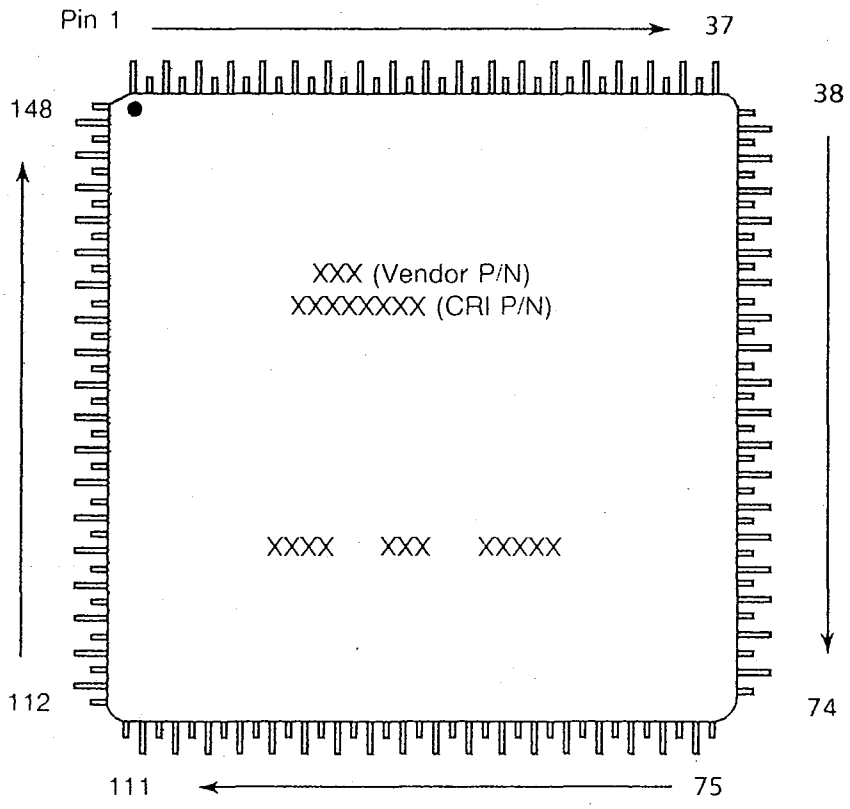


Figure 11-4. MCA2500ECL Pin Package

Pin 91 can be used for a test diode. The test diode on an option is used to test the junction temperature and the thermal characteristics of the 148-pin grid package. CRI uses the test diode if pin 91 does not need to be used as an I/O pin.

The CRAY Y-MP control system (warning and shutdown system) monitors one test diode per printed circuit board.

Table 11-1. Pin Numbers and Functions

Pin Number	Input or Output	Input Pins	Power	
			Gnd	-4.5
1-4	X			
5			X	
6-11	X			
12			X	
13-17	X			
18-20			X	
21-25	X			
26			X	
27-32	X			
33			X	
34-39	X			
40-41		X		
42				X
43-47		X		
48-49			X	
50-55		X		
56				X
57-62		X		
63-64			X	
65-69		X		
70				X
71-72		X		
73-78	X			
79			X	

Pin Number	Input or Output	Input Pins	Power	
			Gnd	-4.5
80-85	X			
86			X	
87-90	X			
91*	X			
92-94			X	
95-99	X			
100			X	
101-106	X			
107			X	
108-111	X			
112-113	X			
114-115		X		
116				X
117-121		X		
122-123			X	
124-129		X		
130				X
131-136		X		
137-138			X	
139-143		X		
144				X
145		X		
146-147	X			
148		X		

* If pin 91 is not used as an I/O pin, it is used for the test diode.

MCA2500ECL MACROCELL ARRAY ESD SENSITIVITY

The MCA2500ECL Macrocell Array is very sensitive to ESD.

MACROCELL LIBRARY TABLE

The Macrocell Library consists of major cells and output cells. Tables 11-2 and 11-3 list the macro and the Boolean equation for the major and output cells in the Macrocell Library.

MACROCELL LIBRARY DIAGRAMS

Logic diagrams for the macros are on the following pages. Each diagram provides the following information: The type of cell (M for major cells, X for output cells; for example, M203, X292), the logic function (eight input AND/NAND), and the Boolean equation.

Table 11-2. List of Major Cells in the Macrocell Library

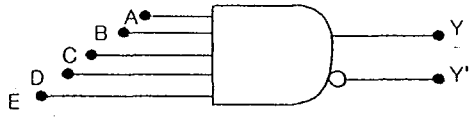
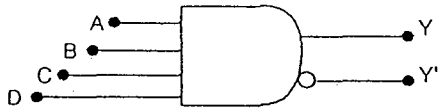
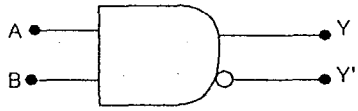
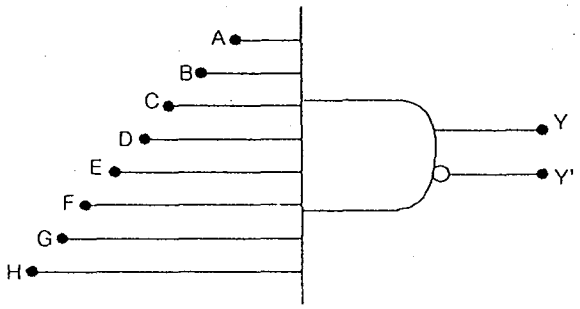
Major Cell Macro Name	Macro Description	Macro Equation
M200	Five Input AND/NAND	Y = ABCDE
M201	Four Input AND/NAND	Y = ABCD
M202	Two Input AND/NAND	Y = AB
M203	Eight Input AND/NAND	Y = ABCDEFGH
M204	12 Input AND/NAND	Y = ABCDEFGHJKLM
M207	Six Input AND/NAND	Y = ABCDEF
M211	(2,2) Sum of Products	Y = AB + CD
M212	(3,2,2,2) Sum of Products	Y = ABC + DE + FG + HJ
M213	(4,3,3,3) Sum of Products	Y = ABCD + EFG + HJK + LMN
M215	(2,2,3,3,3) Sum of Products	Y = (AB)C' + DE(F)' + GK(J)' KL + MN
M219	(3,3) Sum of Products	Y = ABC + DEF
M221	(2,2) AND/XOR	Y = AB\CD
M223	Four Input XOR	Y = A\B\C\D
M224	Four Input XNOR/Inverted XOR	Y = (A\B\C D)'
M225	(2,1,1,2) OR/AND/EXNOR	Y = (AB + C)\(DE + F)'
M226	(2,1,2,1) XOR	Y = (AB + C)\(DE + F)
M227	(1,2) XNOR\OR\NOR	Y = (A\B)(C)'
M228	(1-2) XOR	Y = A\BC
M251	4-to-1 Multiplexer with Enable	Y = MUX(A,B,C,D):DCD(F,E) + (G)'
M254	2-to-1 Multiplexer	Y = MUX(AB,CD):DCD(E)
M255	Dual 2-to-1 Multiplexer	Y = MUX(A,B):DCD(EF) Y = MUX(C,D):DCD(EF)
M256	2-to-1 Multiplexer	Y = MUX(A,B):DCD(C)
M259	4-to-1 Multiplexer	Y = MUX(A,B,C,D):DCD(G,EF)
M261	1-OF-4 Decode	Y = DCD(B,A)/C
M277	(4,4,4,2,2,2) Sum of Products	Y = ABCD + EFGH + JKLM + NP + QR + ST

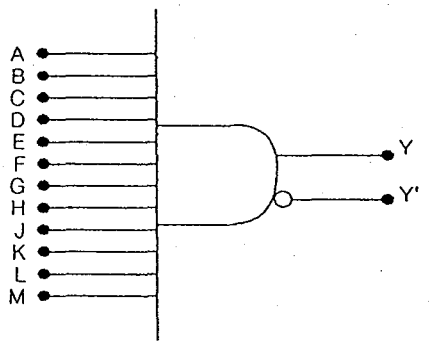
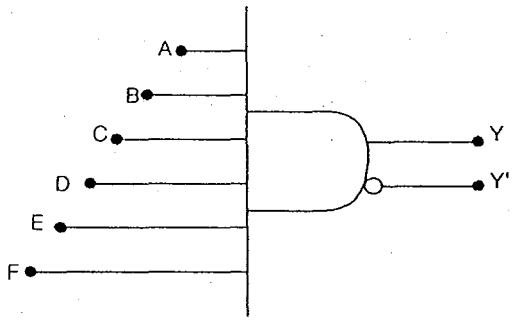
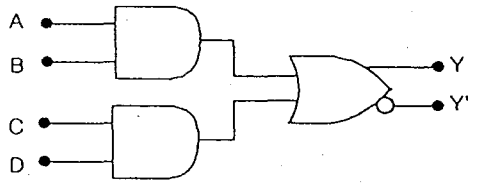
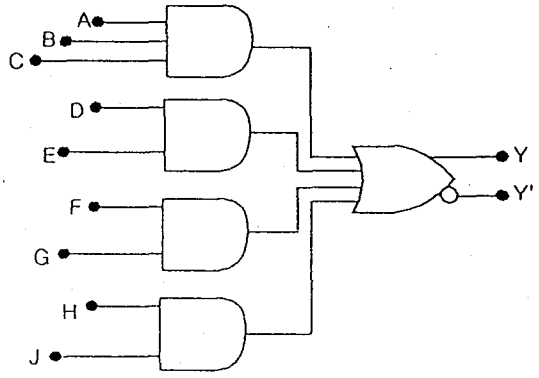
Table 11-2. List of Major Cells in the Macrocell Library

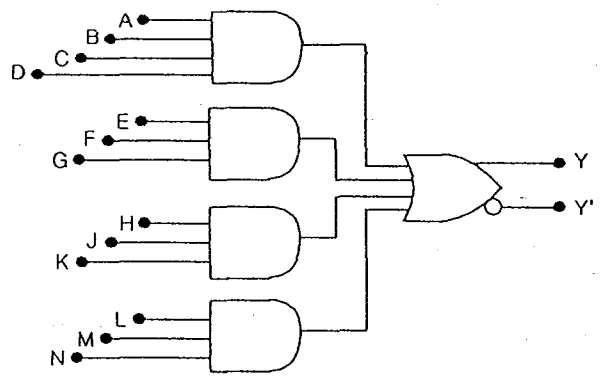
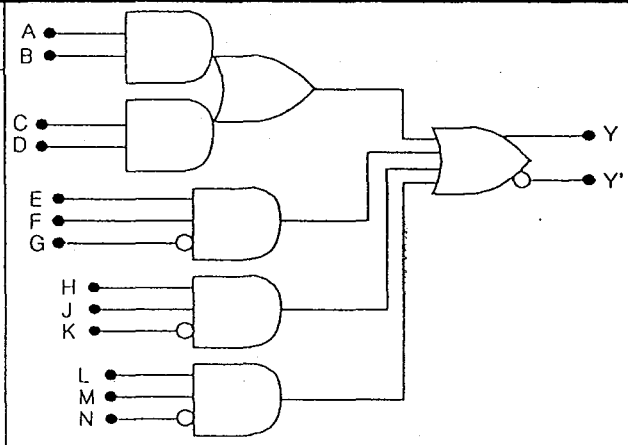
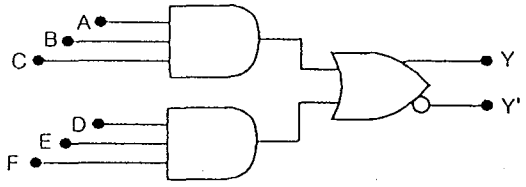
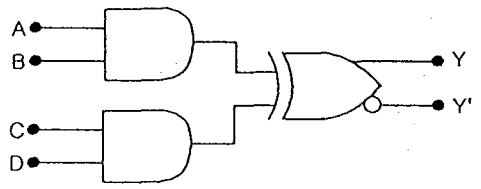
Major Cell Macro Name	Macro Description	Macro Equation
M279	(4,4,4,4,2,2,2) Sum of Products	Y = ABCD + EFGH + JKLM + NPQR + ST + UV + WX + YZ
M279	(4,4,2,2) Sum of Products	Y = ABCD + EFGH + JK + LM
M281	3 Bit Adder Sum/Carry	Y = SCY(A,BC,D)
M282	3 Bit Adder Sum/Carry	Y = SCY(A,BC,DE)
M284	2 Bit Adder/Half Add	Y = SCY(A,BC)
M285	3 Bit Adder Sum	Y = SCY(AB,CD,EF)
M286	3 Bit Adder Carry	Y = SCY(AB,CD,EF)
M291	D Flip-flop with Set (Positive Clock)	Y = AB;T2C + (E)'
M293	D Latch with Set (Positive Clock)	Y = AB;T2C + (E)'
M311	(3,3,3,3) OR/AND	Y = AB(C)' + DE(F)' + GHJ + KLM
M312	(3,3,3) OR/AND	Y = AB[(C)' + (D)' + E]
M313	(2,2) OR/AND	Y = AB(CD)'
M314	(4,3,3,2) Sum of Products	Y = ABCD + EFG + HJK + LM
M319	(3,3,2,1) Sum of Products	Y = ABC + DEF + GH + J
M321	(6,6,4,4,2,2) Sum of Products	Y = PCY(C5,C4,C3,C2,C1,C0:E5,E4, E3, E2,E1)
M322	(3,3,3) Sum of Products	Y = ABC + DEF + GHJ
M323	(3,3,2,2,2,2) OR/AND	Y = AB(C)' + DE(F)' + GH + JK + LM + NP
M324	(5,5,5,5) Sum of Products	Y = ABCDE + FGHJK + LMNPQ + RS TUV
M331	(3,2,2) OR/AND	Y = AB(C)' + DE + FG
M374	Differential Fanout	Y = DIFF(A,B)
M391	D Flip-flop with Set (Negative Clock)	Y = AB;T2C + (E)'
M393	D Latch with Set (Negative Clock)	Y = AB;T2C + (E)'

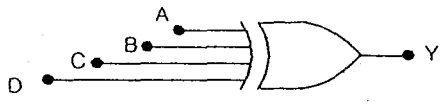
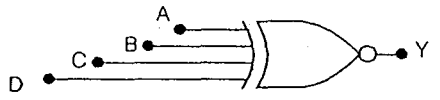
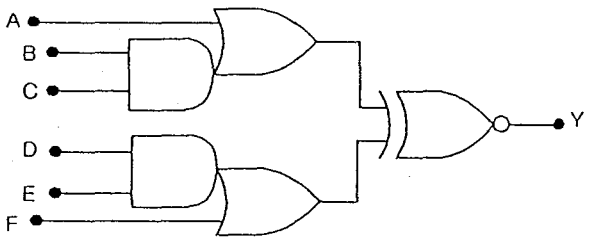
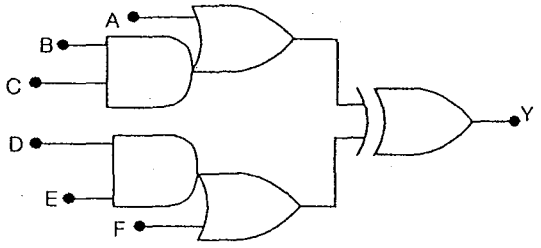
Table 11-3. List of Output Cells in the Macrocell Library

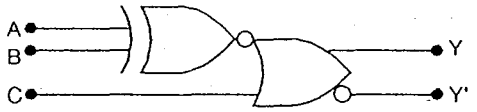
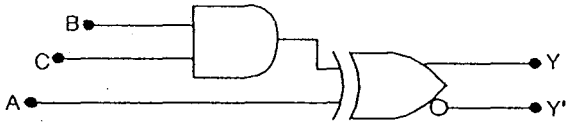
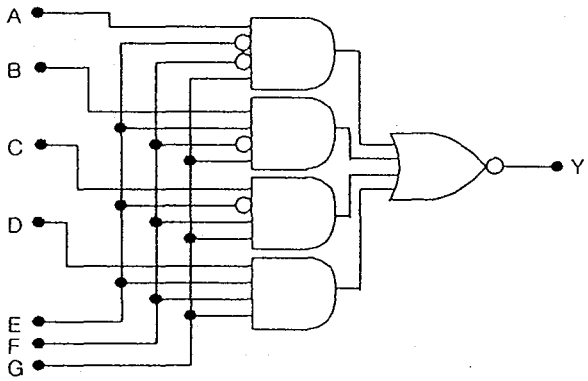
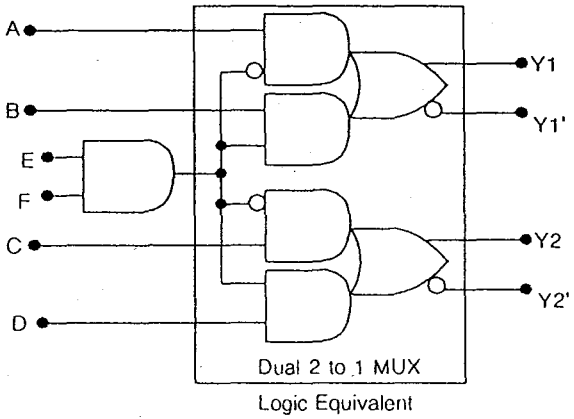
Name of Output Cell	Description	Macro Equation
X201	Two Input AND/NAND	X = AB Z =
X202	Four Input AND/NAND	X = AB Z =
X203	(2,2) AND/AND	X = AB(CD)' Z =
X204	(2,2) AND/AND	X = AB(CD) Z =
X205	Five Input AND/NAND	X = ABCDE Z =
X211	(2,2) AND/OR	Y = AB + CD Z =
X212	(3,2)AND/OR	X = ABC + DE Z =
X221	(2,2) XOR	X = AB\CD Z =
X227	Latch	X = AB;CD
X228	Latch	X = AB;(CD)'
X229	Latch	X = AB;(CD)'
X251	2-to-1 Multiplexer	X = MUX(A,B):DCD(C) Z =
X252	Dual 2-to-1 Multiplexer	X = MUX(A,B):DCD(C) X1 = MUX(C,D):DCD(C) Z = Z1 =
X254	2-to-1 Multiplexer	X = MUX(AB,CD):DCD(EF) Z =
X271	Differential Fanout	X = Dif(A,B)
X291	D Flip-flop	X = AB;T2
X292	D Latch with Set (Postitive Clock)	X = AB;T2 + (E)'

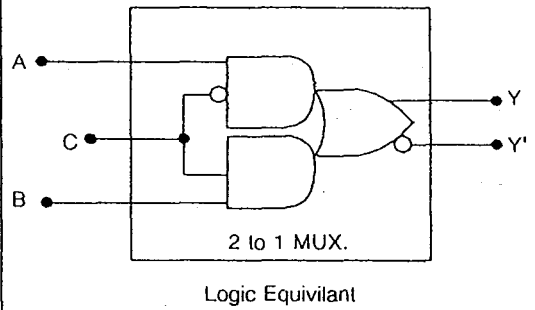
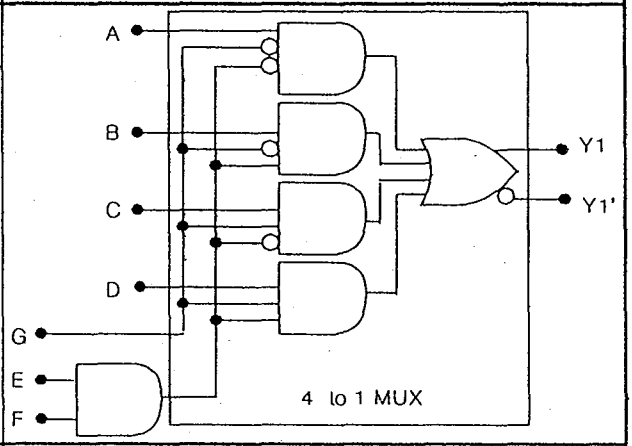
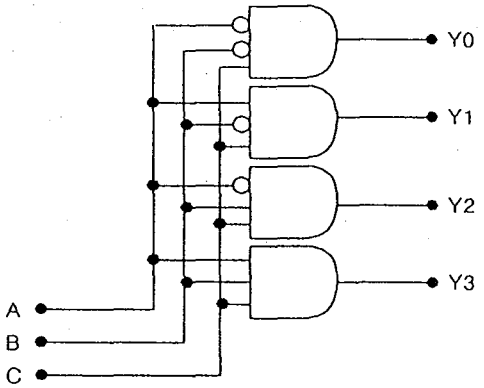
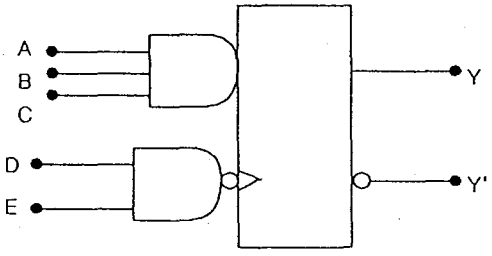
	EQUATION	CELL	SCHEMATIC
M 200	Five Input AND/NAND Macro Boolean: Y = ABCDE	1/4 M	
M 201	Four Input AND/NAND Macro Boolean: Y = ABCD	1/4 M	
M 202	Two Input AND/NAND Macro Boolean: Y = AB	1/4 M	
M 203	Eight Input AND/NAND Macro Boolean: Y = ABCDEFGH	1/2 M	

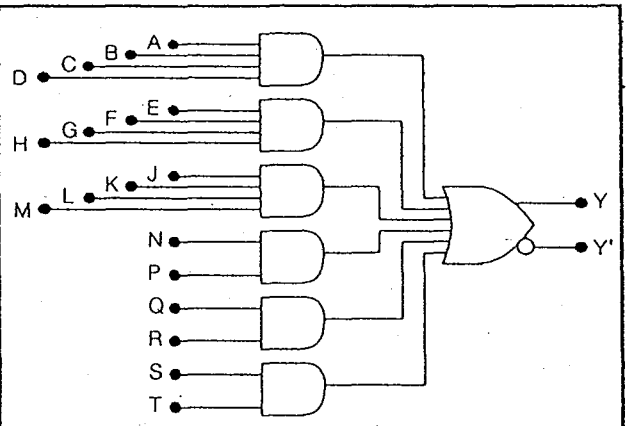
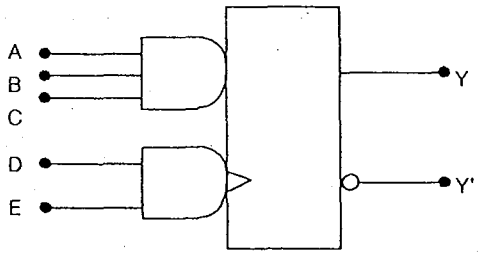
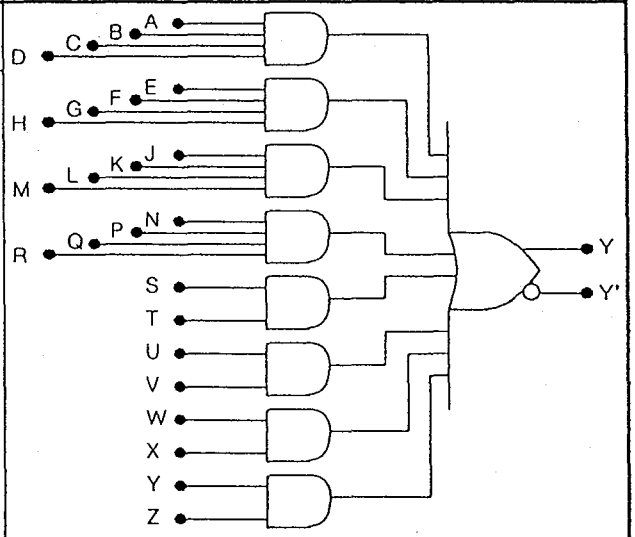
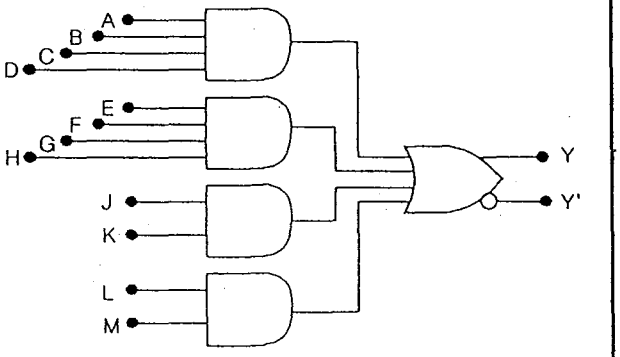
	EQUATION	CELL	SCHEMATIC
M 204	Twelve Input AND/NAND Macro Boolean: Y = ABCDEFGHJKLM	FULL M	
M 207	Six Input AND/NAND Macro Boolean: Y = ABCDEF	1/4 M	
M 211	(2,2) Sum of Products Macro Boolean: Y = AB + CD	1/4 M	
M 212	(3,2,2,2) Sum of Products Macro Boolean: Y = ABC + DE + FG + HJ	1/4 M	

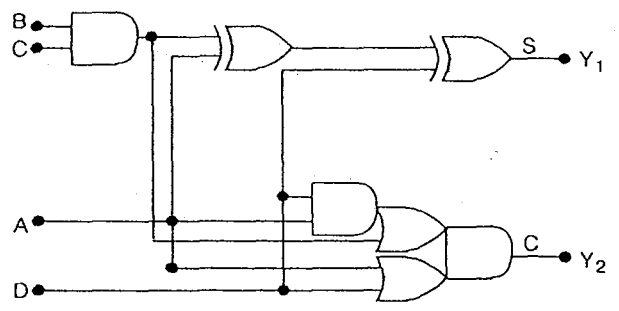
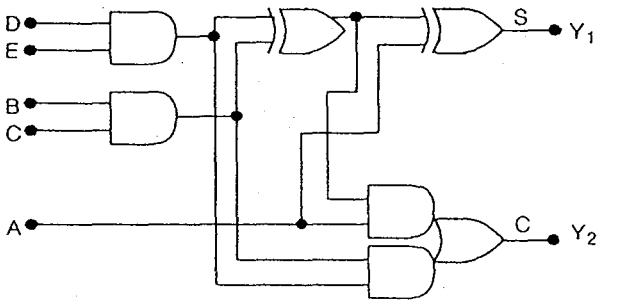
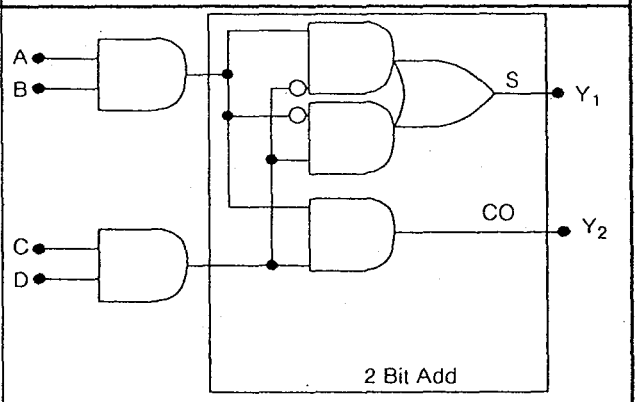
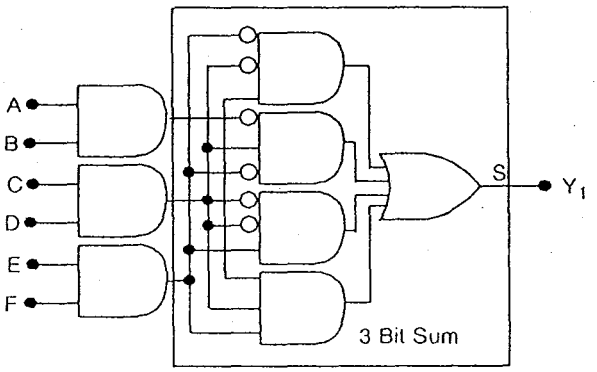
	EQUATION	CELL	SCHEMATIC
M 213	(4,3,3,3) Sum of Products Macro Boolean: Y = ABCD + EFG + HJK + LMN	Full M	
M 215	(2,2,3,3,3) Sum of Products Macro Boolean: Y = AB + CD + EF(G)' + HJ(K)' + LM(N)'	Full M	
M 219	(3,3) Sum of Products Macro Boolean: Y = ABC + DEF	1/4 M	
M 221	(3,2) AND/XNOR Macro Boolean: Y = AB\CD	1/4 M	

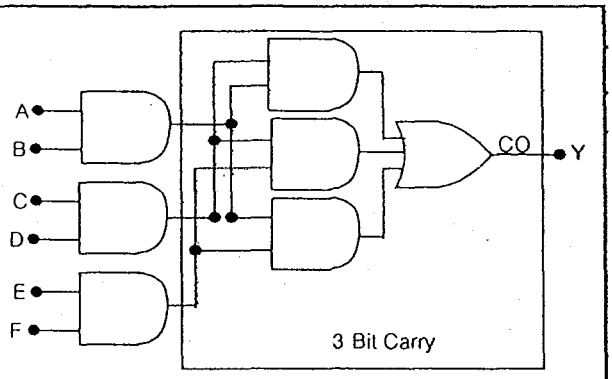
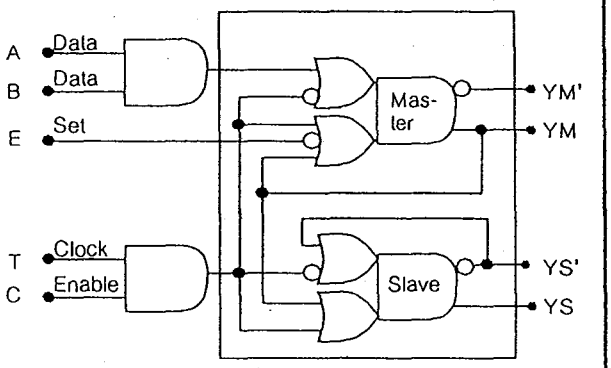
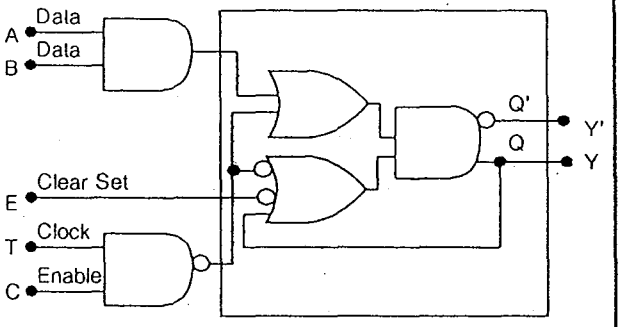
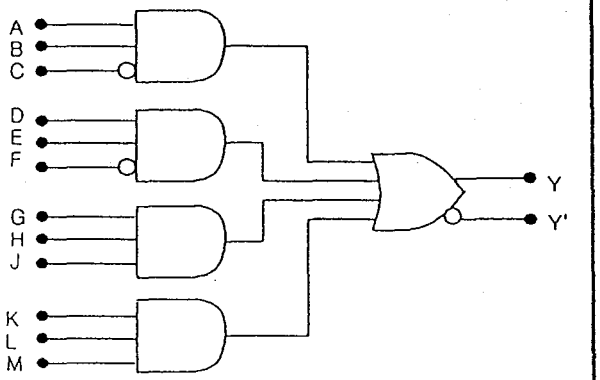
	EQUATION	CELL	SCHEMATIC
M 223	Four Input XOR Macro Boolean: Y = A\B\C\D	1/2 M	 Note: Output is logic 1 when an odd number of inputs are logic 1.
M 224	Four Input XNOR/Inverted XOR Macro Boolean: Y = (A\B\C\D)'	1/2 M	 Note: Output is logic 0 when an odd number of inputs are logic 1.
M 225	(2,1,1,2) AND/OR/XNOR Macro Boolean: Y = [(A + BC)\(DE + F)]'	1/2 M	
M 226	(1,2,2,1) XOR Macro Boolean: Y = (A + BC)\(DE + F)	1/2 M	 NOTE: Output is logic 1 when BC + A ≠ to DE + F.

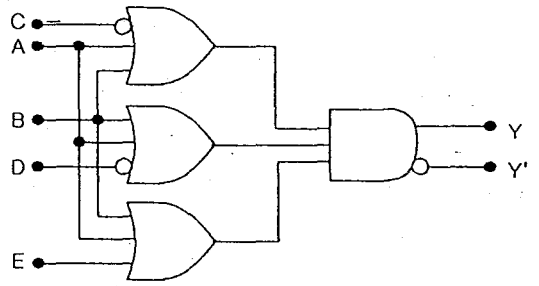
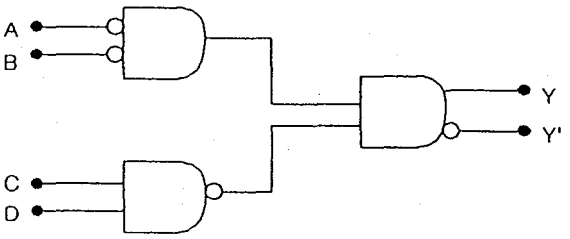
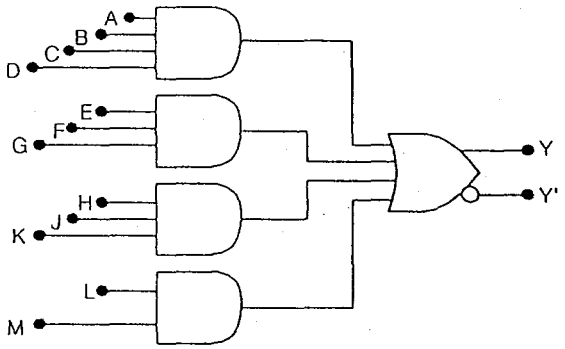
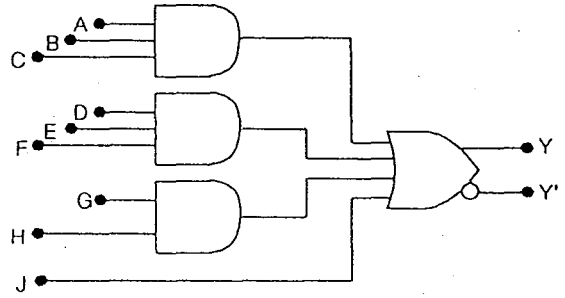
	EQUATION	CELL	SCHEMATIC
M 227	2,1 XNOR/OR/NOR Macro Boolean: Y = (A\B)' + C	1/2 M	
M 228	(1,2) XOR Macro Boolean: Y = A\BC	1/4 M	
M 251	4-to-1 Multiplexer With Enable Macro Boolean: 0 1 2 3 2^1 2^0 Y = MUX(A,B,C,D):DCD(F,E) + (G)'	Full M	
M 255	Dual 2-to-1 Multiplexer Macro Boolean: 0 1 Y1 = MUX(A,B):DCD(EF) 0 1 Y2 = MUX(C,D):DCD(EF)	1/2 M	

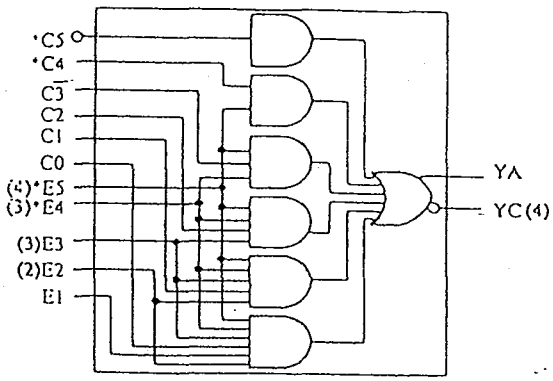
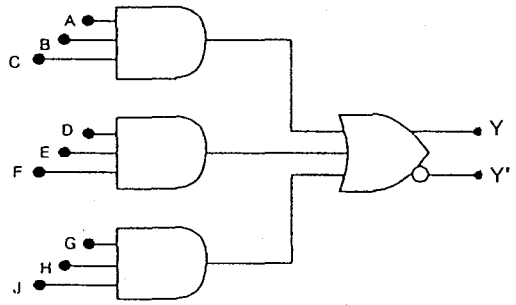
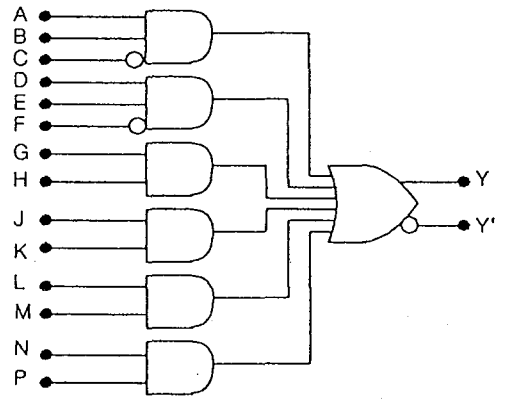
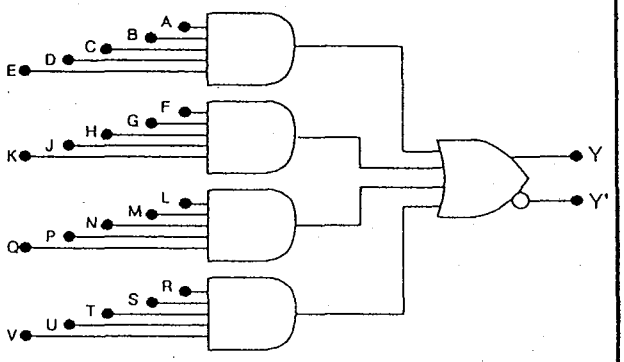
	EQUATION	CELL	SCHEMATIC
M 256	2-TO-1 Multiplexer Macro Boolean: 0 1 YA, YB = MUX(A,B) : DCD(C)	1/4 M	 Logic Equivilant
M 259	4-TO-1 Multiplexer Macro Boolean: 0 1 2 3 2^1 2^0 Y = MUX(A,B,C,D) : DCD(G,EF)	1/2 M	
M 261	1 0f 4 Decode with Enable Macro Boolean: 3 2 1 0 2^1 2^0 Y3, Y2, Y1, Y0 = DCD(B,A)/C	1/2 M	
M 272	3 Input Latch Macro Boolean: Y = ABC;(DE)'		

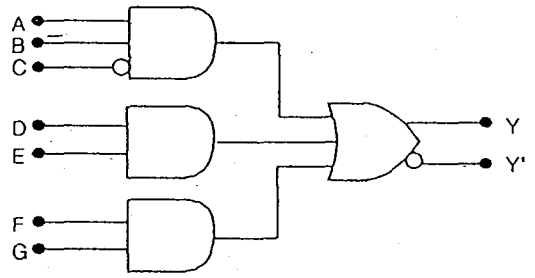
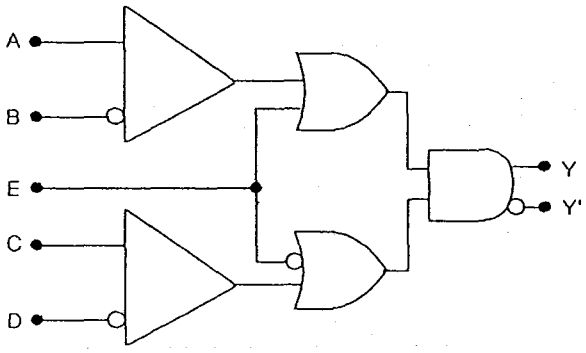
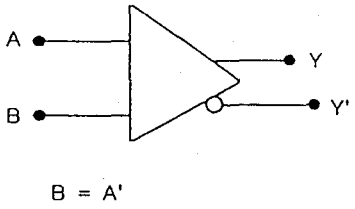
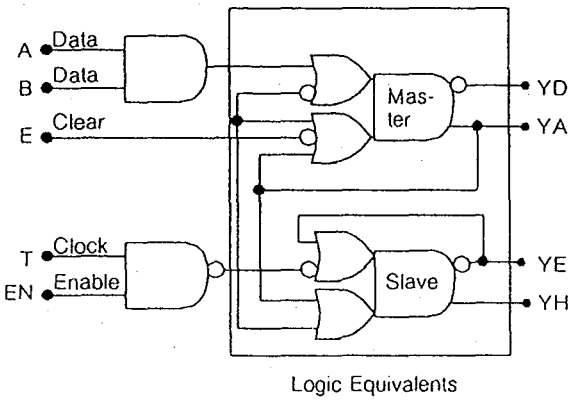
	EQUATION	CELL	SCHEMATIC
M 277	(4,4,4,2,2,2) Sum of Product Macro Boolean: ABCD + EFGH + JKLM + NP + QR + ST	3/4 M	
M 278	3 Input Latch Macro Boolean: Y = ABC;DE	1/4 M	
M 279	(4,4,4,4,2,2,2,2) Sum of Products Macro Boolean: Y = ABCD + EFGH + JKLM + NPQR + ST + UV + WX + YZ	FULL	
M 280	(4,4,2,2) Sum of Product Macro Boolean: ABCD + EFGH + JK + LM	1/2 M	

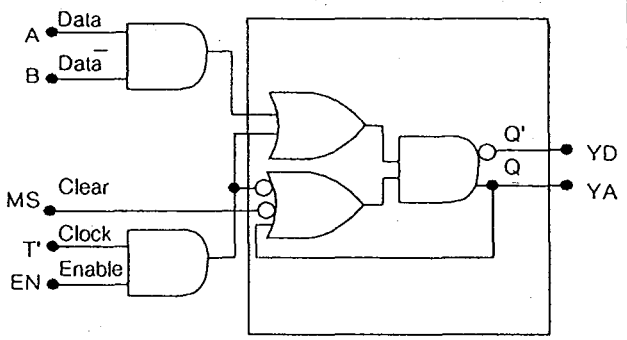
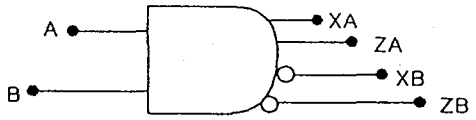
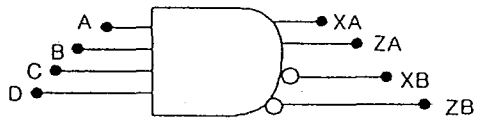
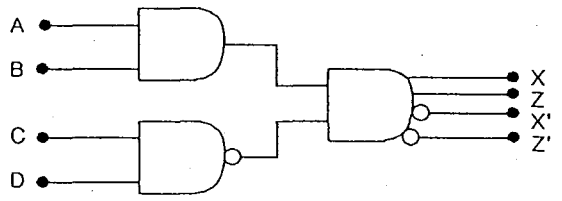
	EQUATION	CELL	SCHEMATIC
M 281	3 Bit Adder Sum/Carry Macro Boolean: (Sum, Carry) Y1,Y2 = SCY(A,BC,D)	Full M	
M 282	Full Adder with Gated Inputs Macro Boolean: (Sum, Carry) Y1,Y2 = SCY(A,BC,DE)	1/2 M	
M 284	2 Bit Adder/Half Add Macro Boolean: (Sum, Carry) Y1,Y2 = SCY(AB,CD)	1/4 M	 Logic Equivalents
M 285	3 Bit Adder Sum Macro Boolean: Y = SUM(AB,CD,EF)	1/2 M	 Logic Equivalents

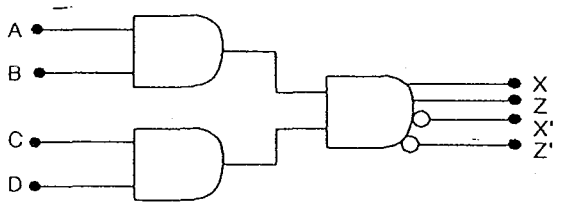
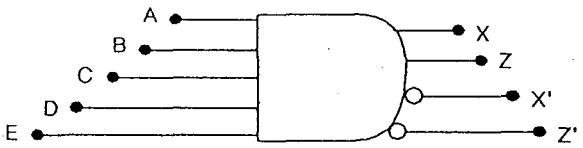
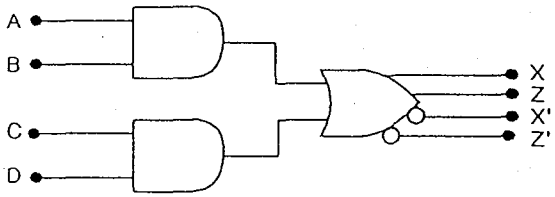
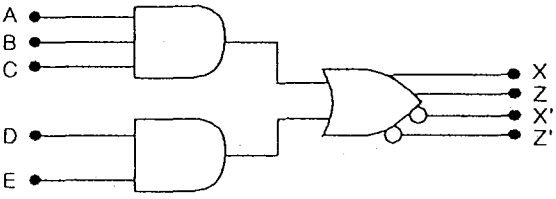
	EQUATION	CELL	SCHEMATIC
M 286	3 Bit Adder Carry Macro Boolean: Y = CARRY(AB,CD,EF)	1/2 M	 Logic Equivalents
M 291	D Flip-Flop with Set (Positive Clock) (Macro Function: D Latch-Delay with Set) Macro Boolean: YM, YS = AB;TC + (E)'	1/2 M	 Logic Equivalent
M 293	D Latch with Set (Positive Clock) Macro Boolean: Y = AB;TC + (E)'	1/4 M	 Logic Equivalent
M 311	(3,3,3,3) OR/AND Macro Boolean: Y = AB(C)' + DE(F)' + GHJ + KLM	Full M	

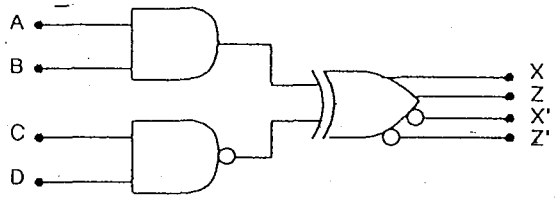
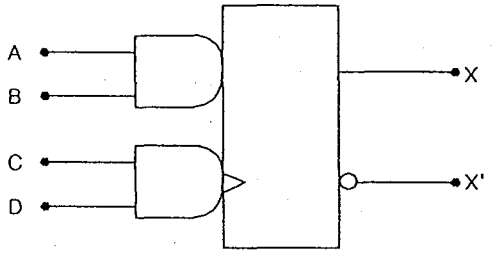
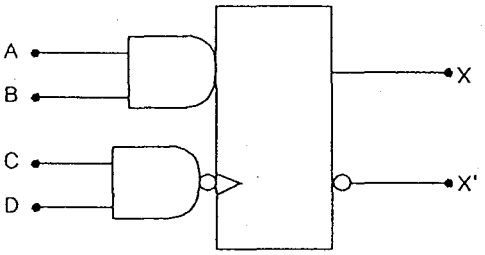
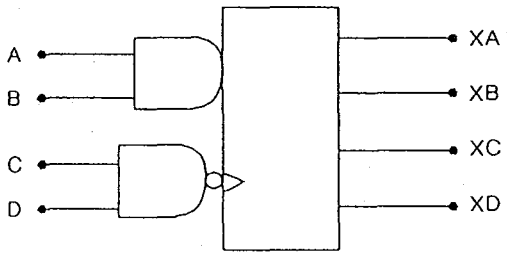
	EQUATION	CELL	SCHEMATIC
M 312	(3,3,3,3) OR/AND Macro Boolean: Y = AB[(C)' + (D)' + E]	1/2 M	
M 313	(2,2) AND/OR Macro Boolean: Y' = AB(CD)' $Y = \overline{AB} + CD$ $y = \bar{A} + \bar{B} + CD$	1/4 M	
M 314	(4,3,3,2) Sum of Products Macro Boolean: Y = ABCD + EFG + HJK + LM	1/2 M	
M 319	(3,2,2,1) Sum of Products Macro Boolean: Y = ABC + DEF + GH + J	3/4 M	

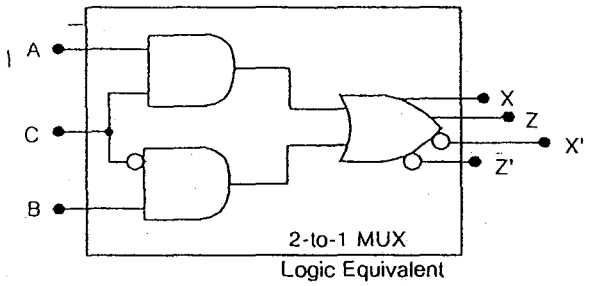
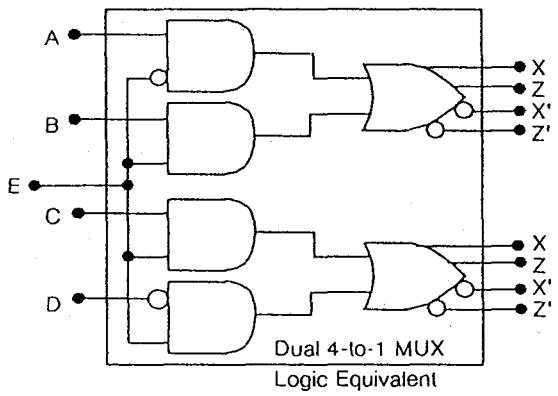
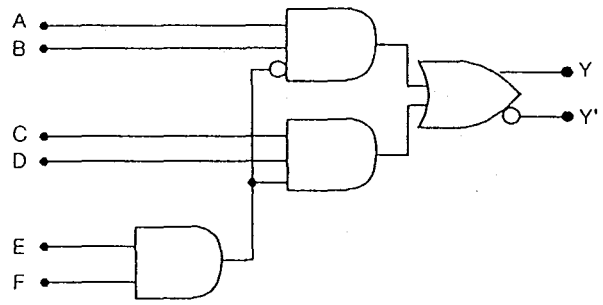
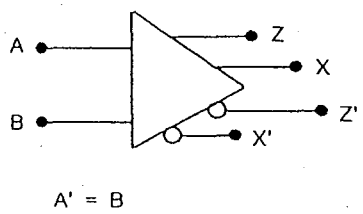
	EQUATION	CELL	SCHEMATIC
M 321	(6,6,4,4,2,2) Sum of Products (Macro Function: Propagate Carry) Macro Boolean: Y = PCarry(C5,C4,C3,C2,C1,C0: E5,E4,E3,E2,E1) Y = C0E1E2E3E4E5 + C1E2E3E4E5 + C2E3E4E5 + C3E4E5 + C4E5 + C5	Full M	
M 322	(3,3,3) Sum of Products Macro Boolean: Y = ABC + DEF + GHJ	3/4 M	
M 323	(3,3,2,2,2,2) OR/AND Macro Boolean: Y = AB(C)' + DE(F)' + GH + JK + LM + NP	M	
M 324	(5,5,5,5) Sum of Products Macro Boolean: Y = ABCDE + FGHJK + LMNPQ + RSTUV	Full M	

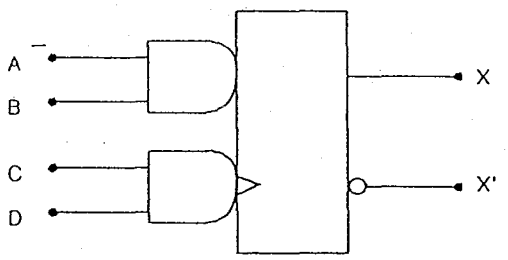
	EQUATION	CELL	SCHEMATIC
M 331	(3,2,2) OR/AND Macro Boolean: Y = AB(C)' + DE + FG	1/2 M	
M 371	3-to-1 Multiplexer Macro Boolean: Y = DMUX(AB,CD):DCD(E)	1/4 M	
M 374	Differential Fanout Macro Boolean: Y = DIFF(A,B)	1/4 M	
M 391	D Flip-Flop with Set (Negative Clock) (Macro Function: D Latch Delay with Set) Macro Boolean: Y = AB;T + (E')	1/2 M	 Logic Equivalents

	EQUATION	CELL	SCHEMATIC
M 393	D Latch with Set (Negative Clock) Macro Boolean: Y = DLY(AB)	1/4 M	 Logic Equivalents
X 201	Two Input AND/NAND Macro Boolean: X = AB Z = AB	1X	
X 202	Four Input AND/NAND Macro Boolean: X = ABCD Z = ABCD	1X	
X 203	(2,2) AND/AND Macro Boolean: X = AB(CD)' Z = AB(CD)'	1X	

	EQUATION	CELL	SCHEMATIC
X 204	(2,2) AND/AND Macro Boolean: X = ABCD Z = ABCD	1X	
X 205	Five Input AND/NAND Macro Boolean: X = ABCDE Z = ABCDE	1X	
X 211	(2,2) AND/OR Macro Boolean: X = AB + CD Z = AB + CD	1X	
X 212	(3,2) AND/OR Macro Boolean: X = ABC + DE Z = ABC + DE	1X	

	EQUATION	CELL	SCHEMATIC
X 221	(2,2) AND/XOR Macro Boolean: X = AB\CD	1X	
X 227	Latch Macro Boolean: X = AB;CD	1X	
X 228	Latch Macro Boolean: X = AB;CD'	1X	
X 229	Latch Macro Boolean: XA = XB' = AB;CD' XC = XD' = XA;CD	1X	

	EQUATION	CELL	SCHEMATIC
X 251	**2-to-1 Multiplexer** Macro Boolean: X = MUX(A,B):DCD(C) Z = MUX(A,B):DCD(C)	1X	
X 252	**Dual 2-to-1 Multiplexer** Macro Boolean: 0 1 X = MUX(A,B):DCD(E) X = MUX(C,D):DCD(E) 0 1 Z = MUX(A,B):DCD(E) Z = MUX(C,D):DCD(E)	2X	
X 254	**2-to-1 Multiplexer** Macro Boolean: 0 1 X = MUX(AB,CD):DCD(EF) 0 1 Z = MUX(AB,CD):DCD(EF)	1X	
X 271	**Differential Fanout** Macro Boolean: X = DIFF(A,B) Note: Macro Valid if A = B' Z = DIFF(A,B)	1X	

EQUATION	CELL	SCHEMATIC
D Flip-Flop Macro Boolean: X = AB;TC + (E)'	1X	

X 291

CONTENTS

FIGURES

FIGURES (continued)

TABLES

APPENDIX A - ENGINEERING DOCUMENTATION

This section explains how to interpret engineering documentation associated with the CRAY Y-MP computer system. Understanding this documentation makes it possible to follow any bit through every gate it passes in the machine.

In the following subsections, samples from appropriate documents are illustrated and explained. The notation used in these documents for headings, terms, options, location designators, etc. is not consistent with respect to capitalization. Even though the samples shown duplicate the original documents as closely as possible, all references to these terms, options, location designators, and so on in the descriptive text of this section are capitalized.

CLOCK PERIOD AND TIME SEGMENTS

A clock period (CP) is the time interval between successive latchings of a bit of information as it passes through the computer. Each CP is further divided into time segments, each about 100 picoseconds in duration. The notation TS is used when referring to a particular time segment, such as TS 20.

A bit of information can pass through up to four macrocells between successive latches on the same option. If the latches are on different options, the bit can pass through a maximum of three macrocells between the latches; the time needed to travel between the options decreases the number of macrocells through which the bit can pass. The order in which the bit passes through the macrocells is indicated by a numbering scheme in which each macrocell is assigned a number, 0 through 47_8 for the CPU modules and 0 through 4 for the memory modules. Figure A-1 shows a bit of information passing through six successive macrocells.

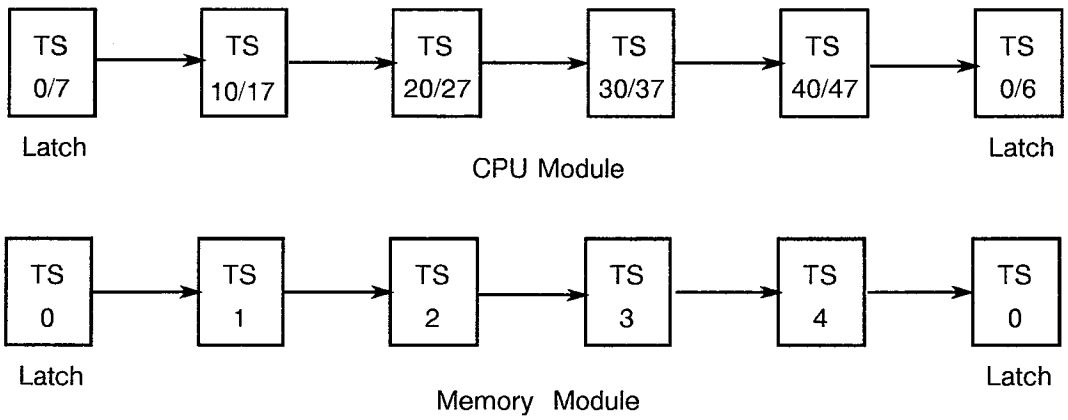


Figure A-1. Data Flow Through CPU and Memory Module Macrocells

BOOLEAN

Boolean logic equations describe the internal design and operation of each option. The Boolean documentation for each option consists of the following items:

- A cover sheet listing all input and output terms for the option
- A listing of the history of design changes to the option
- A detailed listing of the Boolean equations describing the option

Cover Sheet

In addition to the input and output terms, the cover sheet lists the option's name, revision number, date of release, usage, number of options of this type used per module (CPU options only), and the total number of pins used on the option. A sample of a cover sheet for an option is shown in Figure A-2, followed by a bulleted description of each column under the input and output terms.

```
OPTION: YA
REV:    100
DATE:   12-10-86
TITLE:  ADDRESS SECTION SELECT  (4 USED)


INPUTS:

I00-07  TS=39   LL=2.    8       PORT A ADDRESS              YF
I40-43  TS=27   LL=0.    4 C     PORT B SECTION ENABLE       YK
I50     TS=17   LL=1.    1 T(C)  PORT B 0 REFERENCE          YE
I60-63  TS=27   LL=1.    4 T/C   PORT C 2-4,1 REFERENCE      YE

OUTPUTS:

 R08-15 TS=10   LL=0.    16      SECTION N+1 ADDRESS         YN
2 R16   TS=10   LL=0.    4       4-2 FANIN/FANOUT            EXT
 R21-22 TS=29   LL=2  SE.2       SECTION N+2 GOSS 0-3,4-7    YL


TOTAL PINS 118 (50 INPUTS + 68 OUTPUTS)
```

Figure A-2. Boolean Cover Sheet

- The first column identifies the Boolean terms entering or leaving the option. I terms are used exclusively for inputs; R terms are reserved for outputs. A number preceding an R term designates the number of copies of the term that

are sent from the option. While most R terms leave the option in complementary pairs (for example, R08 and R08'), there are instances in which only either the upright or the inverted term leaves. Each true term can go to as many as four different destinations, and each false term can go to as many as four different destinations. A complete listing of all R terms and their destinations can be found in the PC load chart for each option. (Refer to "Load Charts" later in this section.)

- The second column lists the time segment at which the I term enters or the R term leaves the option.

- The third column indicates the logic level (LL) of the term, which is its relative position in time between successive latches on different options. There are four discrete logic levels, numbered 0 through 3. For output terms, the logic level corresponds to the number of macrocells the departing term has passed through since it was last latched. For input terms, subtracting the logic level from 3 determines the number of macrocells the term must pass through before it is latched. Thus an input term at LL1 must pass through two macrocells before it is latched. An R term at one logic level can connect directly to an I term at the same level. However, it is possible for an R term at LL1, for example, to connect to an I term at LL2 or LL3. In these cases, the foil path between the options has to be lengthened to delay the incoming term by the time required to pass through 1 or 2 macrocells respectively.

  In place of the logic level, input terms may have the word FORCE in this column, indicating that the particular term(s) is forced to either a logic 0 or logic 1 state and held constant. This condition can be caused either by a direct signal from an external device, such as a switch on the maintenance panel, or by internal wiring that may set the term(s) according to the location of the option. In the latter case, the status of the forced term(s) is listed at the end of the PC load chart for the option.

- The fourth column indicates the number of pins used by the associated terms. Thus, input terms I0 through I7 use 8 pins, one each for terms I0, I1, and so on; while output terms R08 through R15 use 16 pins, one for each true and false output term R08, R08', R09, R09', and so on. Because there are two copies of term R16 sent off the option, four pins are required, two copies each for terms R16 and R16'. For terms R21 and 22, only two pins are needed, one for each term. These output terms are referred to as single-ended (SE) because only either the true or false term leaves the option. To find out if the departing term is true or false, consult the PC load chart for the option.

- The fifth column is used only for input terms and indicates whether the incoming terms are true (T) or complemented (C) copies. A blank space indicates true copies. The notation T(C) is only used if there are two or more of this option type used per CPU. In this case, some of the options receive true copies while others receive false copies. The only way to determine the true or false status of the input terms to a particular option is to consult the PC load chart for the option from which the terms came, in this example, the YE option. The notation T/C means that some of the copies coming into this option are true, while others are false. In the example shown, this means that some of the terms I60 through I63 are true terms, while others are false. To determine which terms are true and which are false, check the Boolean equations to see

how the individual input terms are processed, or consult the PC load chart for the option from which these terms came.

- The next column contains a brief description of the term, or terms.
- The last column lists either the option(s) from which the term came or the option(s) to which it goes, depending on whether the column follows input or output terms respectively. The designation EXT (External) indicates that the term came from or is headed to a different module.

Option Design Change History

This information describes the history of changes to the option. The Option Design Change History starts with a Gate Array Revision Notice (GARN) that introduces the option. It may be followed by a series of Design Change Notices (DCNs) that describe minor modifications to the original Boolean.

Boolean Equations

Boolean equations for most options are printed in uppercase; however, there are some equations written in lowercase. The case used is insignificant. The equations are generally listed in alphabetic order with two exceptions: the clock terms (T terms) are always listed first, and the output terms (R terms) are usually listed last, though occasionally they may occur within the listing. The equations are further divided into groups according to the time segments during which the individual terms become active.

Frequently, the Boolean terms for a letter may not occur in consecutive numerical order. For example, if there are 20 A terms, five for each of four ports, they need not be listed as terms A0 through A19, but could be listed as terms A0 through A4, A10 through A14, A20 through A24, and A30 through A34. This symmetrical numbering of terms is often used to make corresponding data bits for different data lines easier to remember.

The following examples show several of the condensed Boolean notations currently used:

```
M256  2 A0-1=MUX(H0-1,I0-1):DCD(L0')      .PORT A 2,3
        A3=MUX(I3,H3):DCD(L0)             .       1
M213    K0-1'=H0-1 R4+H10-11 R6'          .SECTION N 2,3
X227    R6-8=F10-12' F13-15;T2            .PORT C ENABLE SEC. N,N+2
```

In each example, the first column indicates the macrocell type used to carry out the logic of the Boolean equation. If this column is blank, as it is for the A3 term, the macrocell type used is the same as for the previous term, M256. The single Boolean equation used for the A0 and A1 terms is a simplified notation used to combine the following two equations:

```
A0=MUX(H0 I0):DCD(L0')
A1=MUX(H1 I1):DCD(L0')
```

Similarly, the K0-1 equation can be expanded to give the following two equations:

```
K0'=H0 R4+H10 R6'
```

```
K1'=H1 R4+H11 R6'
```

The R6-8 equation will yield three equations, as follows:

```
R6=F10' F13;T2
R7=F11' F14;T2
R8=F12' F15;T2
```

Note that the single complement (') on terms F10-12 applies to each of the terms F10, F11, and F12 individually. The following special characters are frequently used in the Boolean logic equations:

SYMBOL	DEFINITION
#	Forced 1
!0	Forced 0
\	Exclusive OR
+	OR
'	Complement
Z999	Forced 0

OPTION PINOUTS

The pinout documentation for each option lists the specific I or R terms connected to each pin. The documentation consists of two charts: the Option Pin Description and the Option Term Description. The Option Pin Description lists each of the 148 pins in numerical order, listing the I term, R term, and power or ground connection to each pin. A sample Option Pin Description chart is shown in Figure A-3.

Option Pin Description

Pin	Term
004	r3'
005	<VCCO>
006	
129	i98
130	<VEE>

Option Term Description

Term	Pin
i44	127
i45	135
i98	129
r0	146
r0'	147

Figure A-3. Option Pinout Charts

If a pin is not used, a blank space is left in the term column, as shown for pin 006. <VCC> and <VCCO> indicate ground connections, while <VEE> is connected to –4.5 V.

The second chart, the Option Term Description, lists the I and R terms in numerical order, followed by the pin number to which each term is connected. A sample listing is shown in Figure A-3.

Figures A-4 and A-5 show the two standard 2500-gate ECL packages used: one manufactured by Cray Research, Inc. (CRI) and the other manufactured by Motorola.

Note: The location of Pin 1 is different on each of these packages.

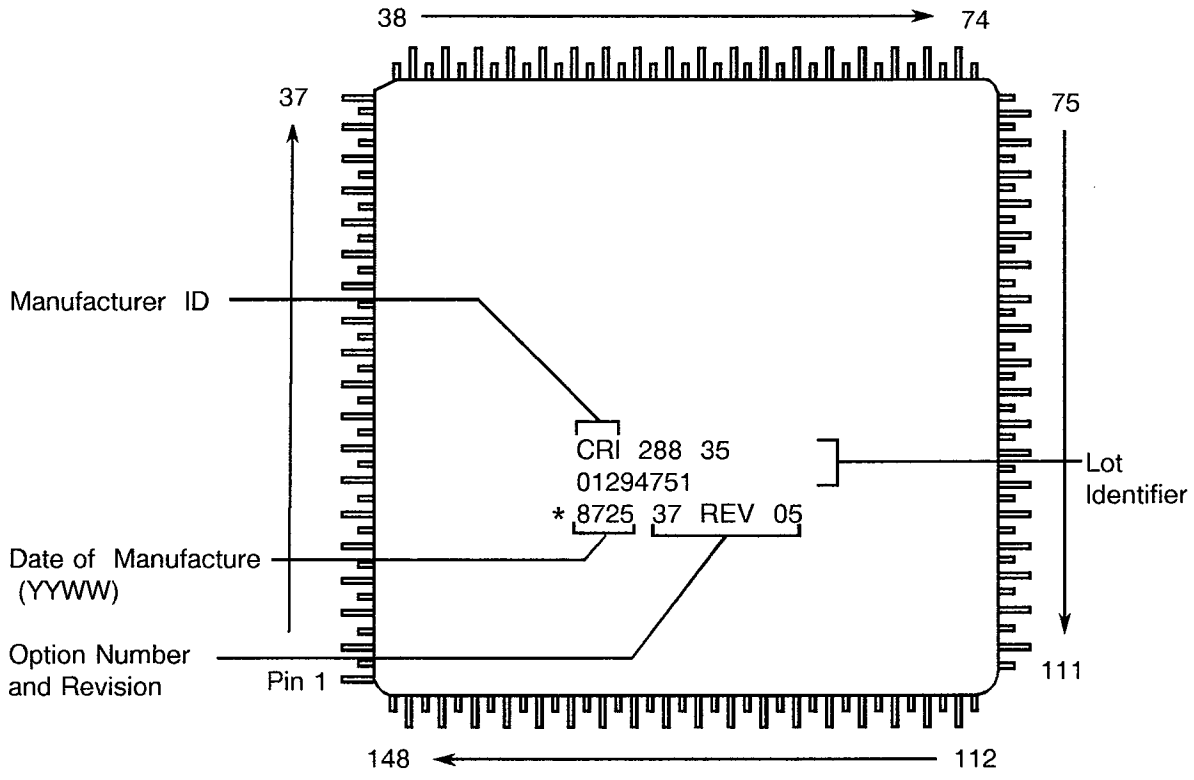


Figure A-4. CRI 2500-gate Macrocell Array Pin Package

MODULE BOARD LAYOUTS

Each CPU and memory module is made up of four 12-layer printed circuit boards labeled A through D. The clock module consists of a single board, equivalent in position to a C board of a CPU or memory module. Each memory board contains thirty-five 2500-gate macrocell array chips and 288 64 k×1 ECL-compatible RAM chips (arranged in 36 groups of 8). Each CPU board contains up to seventy-eight 2500-gate macrocell array

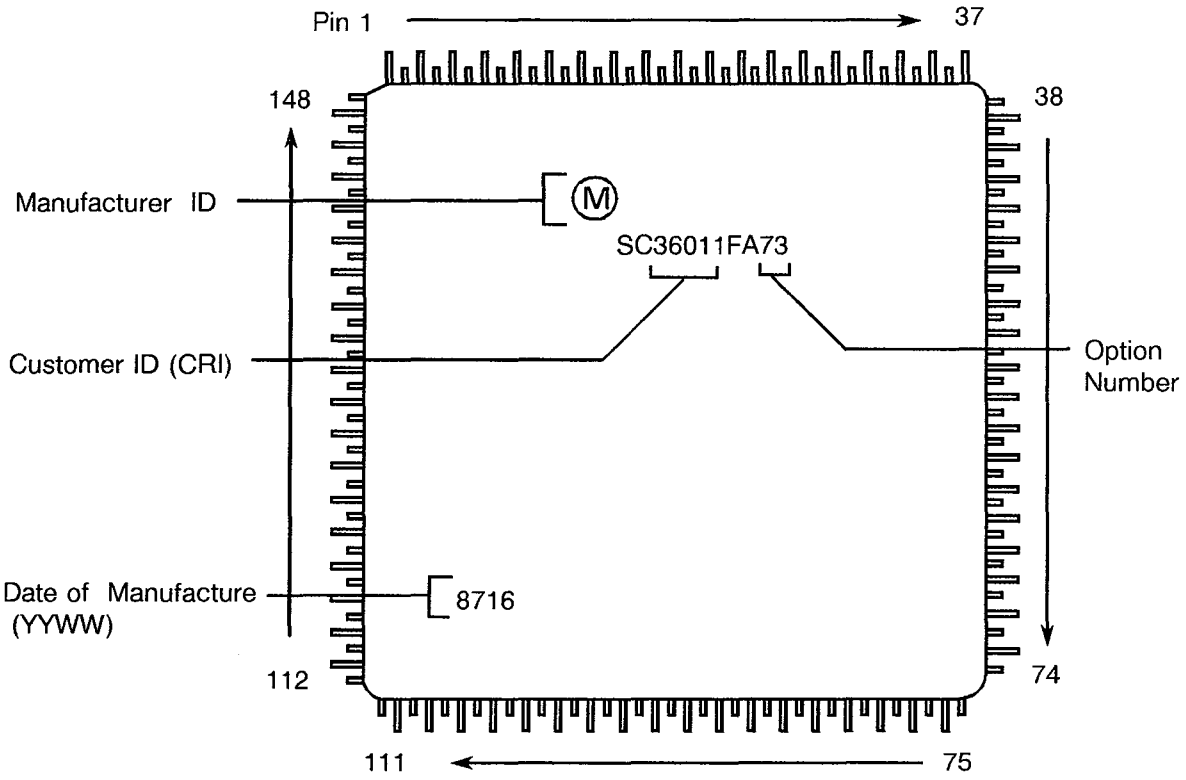


Figure A-5. Motorola MCA 2500-gate ECL Macrocell Array Pin Package

chips. The clock module contains twelve 2500-gate macrocell array chips and seven crystal oscillators. The layouts of boards A of both the memory and CPU modules are shown in Figures A-6 and A-7. Refer to Section 1, "CRAY Y-MP Physiscal Description" for more information. The layout of the clock module is shown in Figure A-8.

Each of the macrocell array chips is given a unique three- or four-character logical identifier. The first two characters are letters indicating the internal configuration (the option type) of the chip. The third and fourth characters make up a 1- or 2-digit number used to differentiate chips of the same option type. An example of a macrocell chip logical identifier is VB3.

Each of the 36 groups of memory storage chips is given a unique four-character logical identifier. The first two characters are always ZZ. The third and fourth characters are a letter and number that identify the particular memory chip group. An example of a memory chip group logical identifier is ZZA0. In order to locate a single chip within a memory chip group, the logical group identifier and the Boolean R term must be known. An example of a logical identifier for a memory chip is ZZA0 R4. Figure A-9 shows the locations of individual memory chips within a group. The groups can be properly oriented on each board by observing the Y and Z side alignments as shown in the figure.

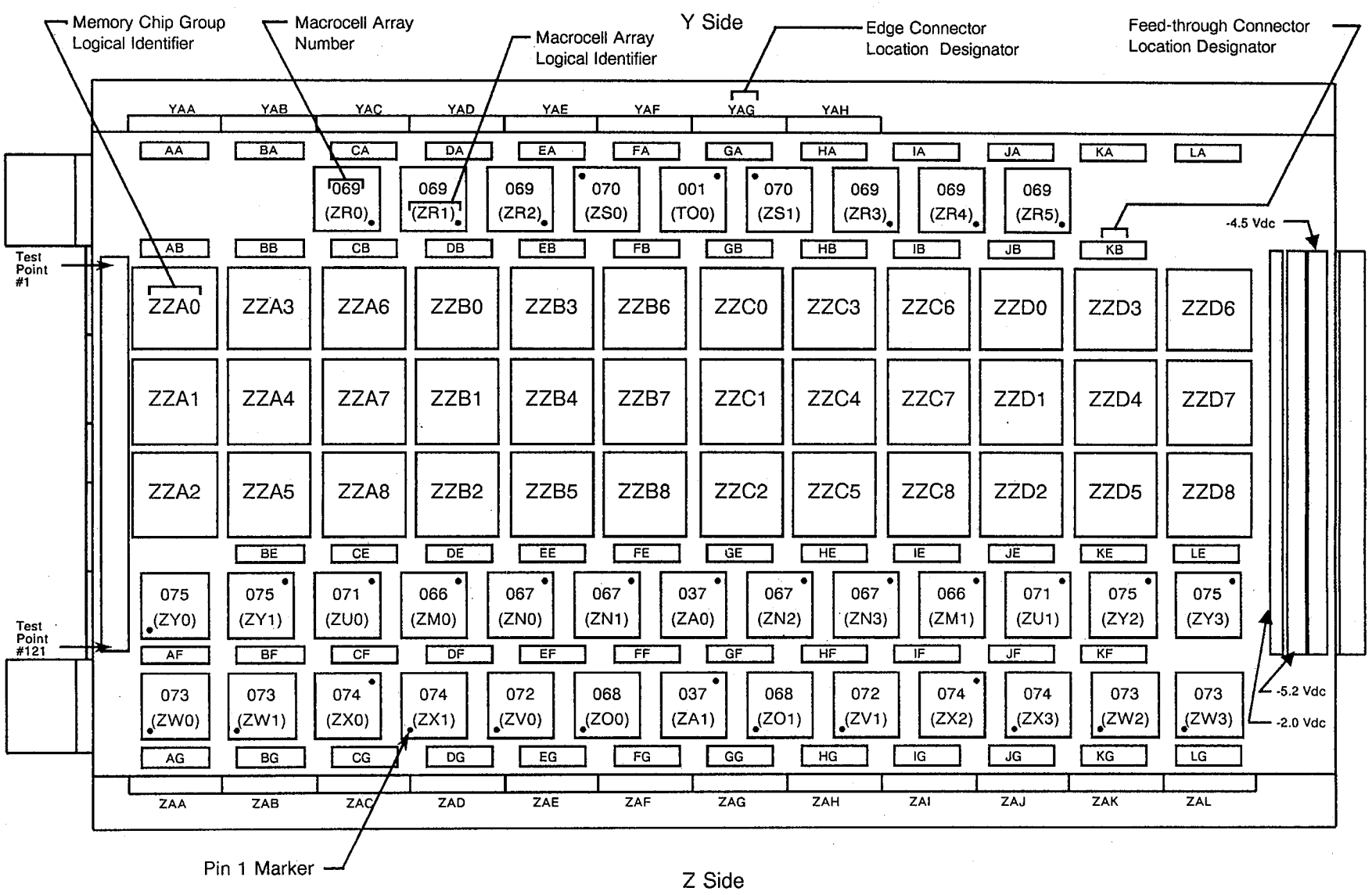


Figure A-6. Sample of Memory Module Board A Layout

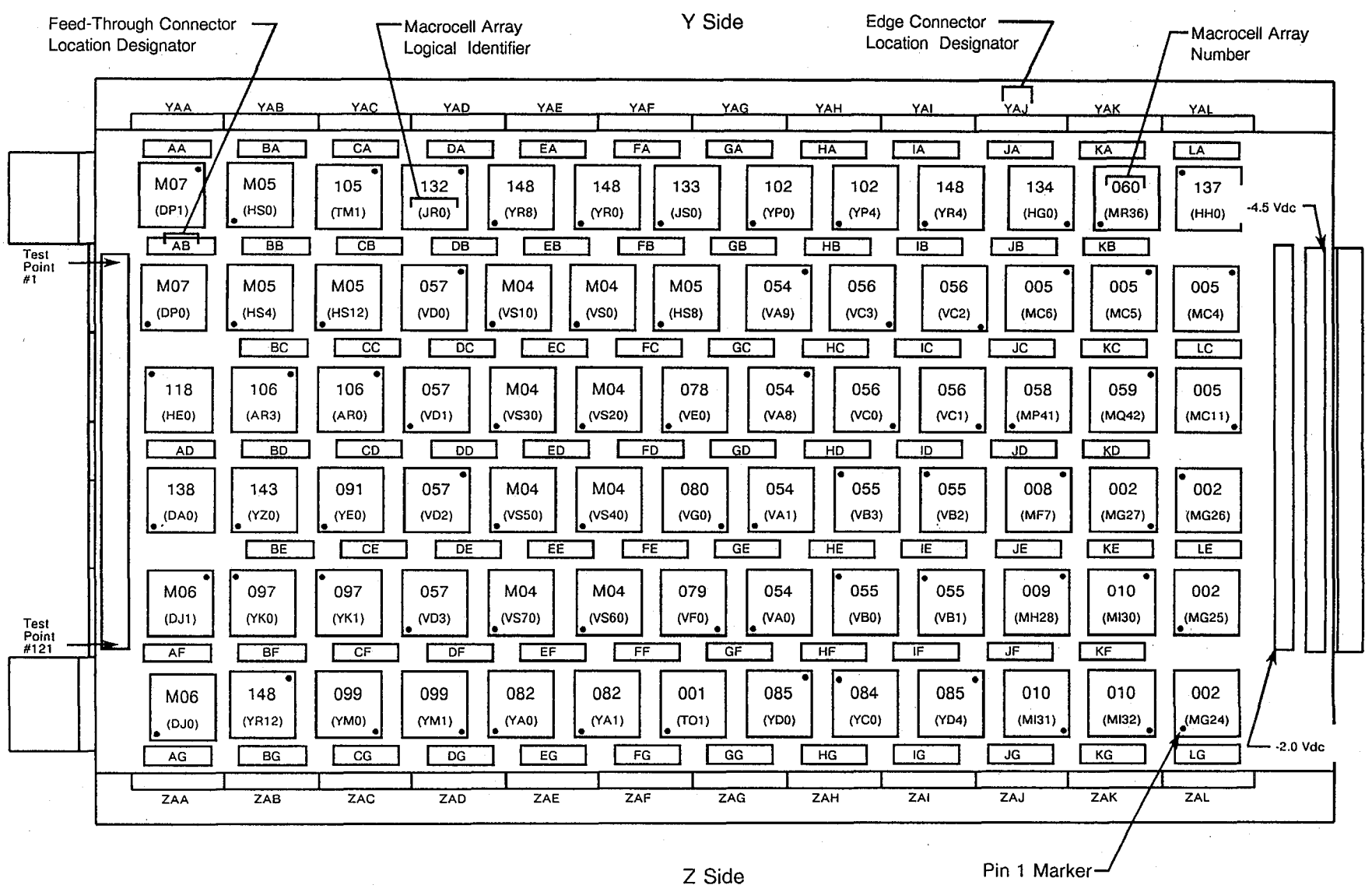


Figure A-7. Sample of CPU Module Board A Layout Rev 4

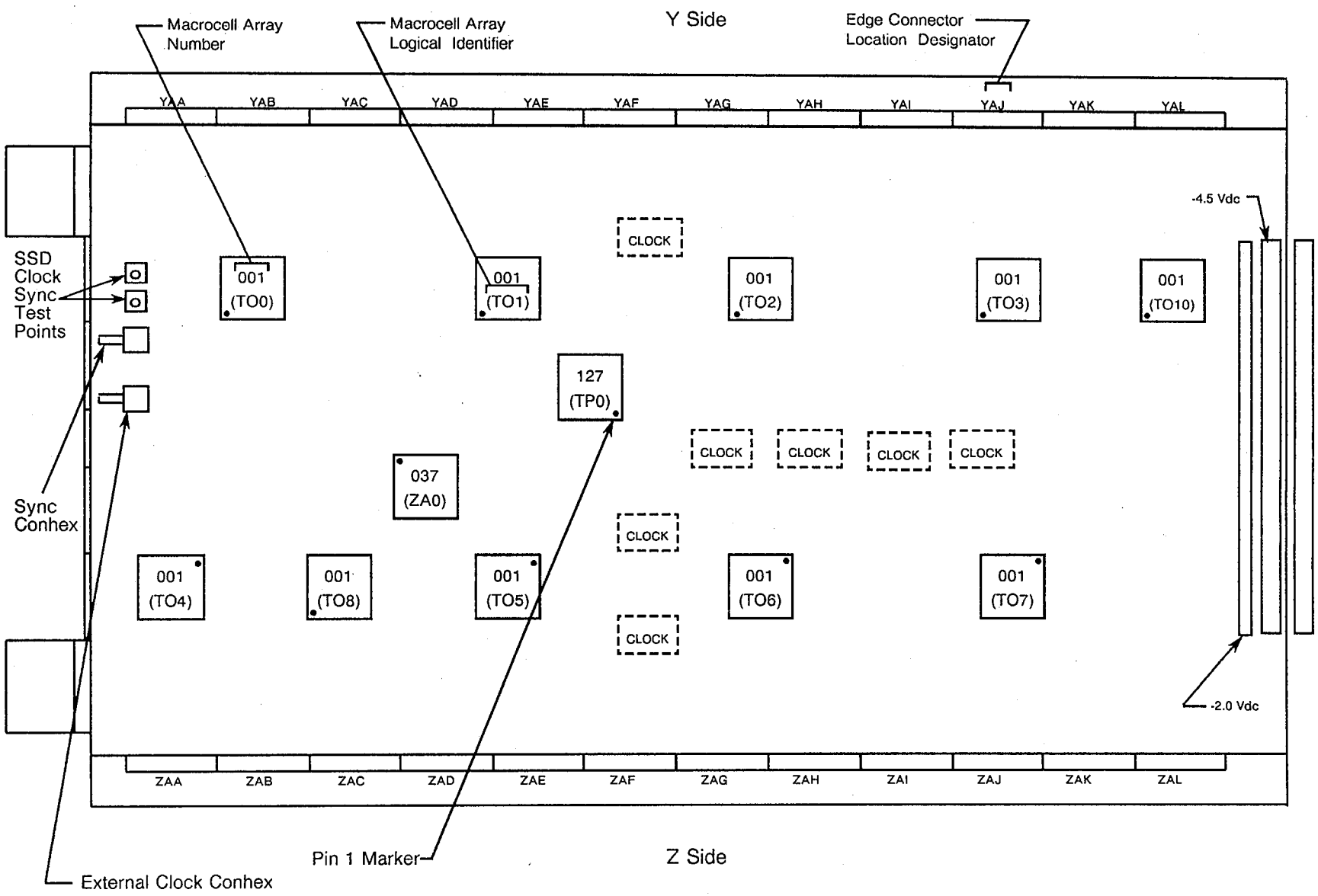


Figure A-8. System Clock Module

The physical location of a macrocell chip or memory chip group on a module is specified by a three-letter location designator. The first letter indicates the board (A through D), the second letter indicates the column (A through M) and the third letter indicates the row (A through F). The MP41 option, for example, is located on the memory module at AKC. The exact location of an option is specified in the chip map documentation explained later in this section.

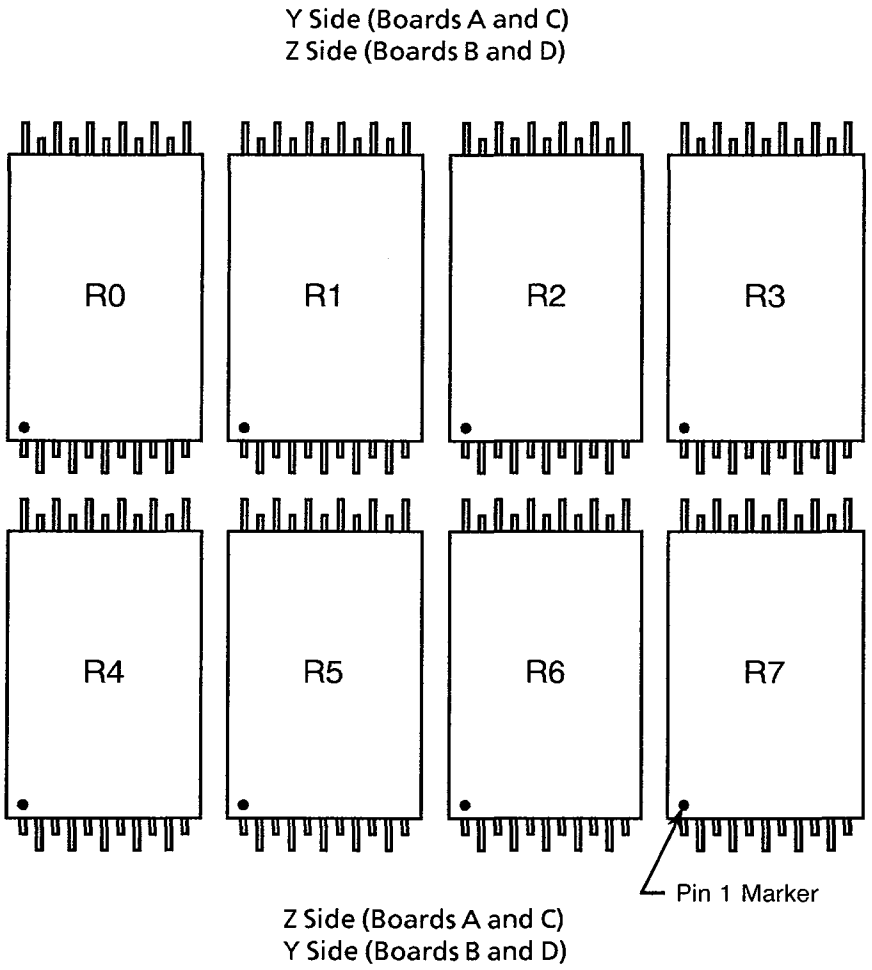


Figure A-9. Memory Chip Group Layout

Connections between the four boards of a module are made using feed-through or jumper connectors. Each memory module has 57 feed-through connectors, while the CPU modules each have 79 feed-through connectors. Though several different size connectors are used, each connector will have some subset of the 48 pins numbered and arranged as shown in Figure A-10. The connector locations are specified by the same column/row coordinate system used by the macrocell chips, except that only two letters are used. The first letter designates the column (A through L) and the second designates the row (A through G).

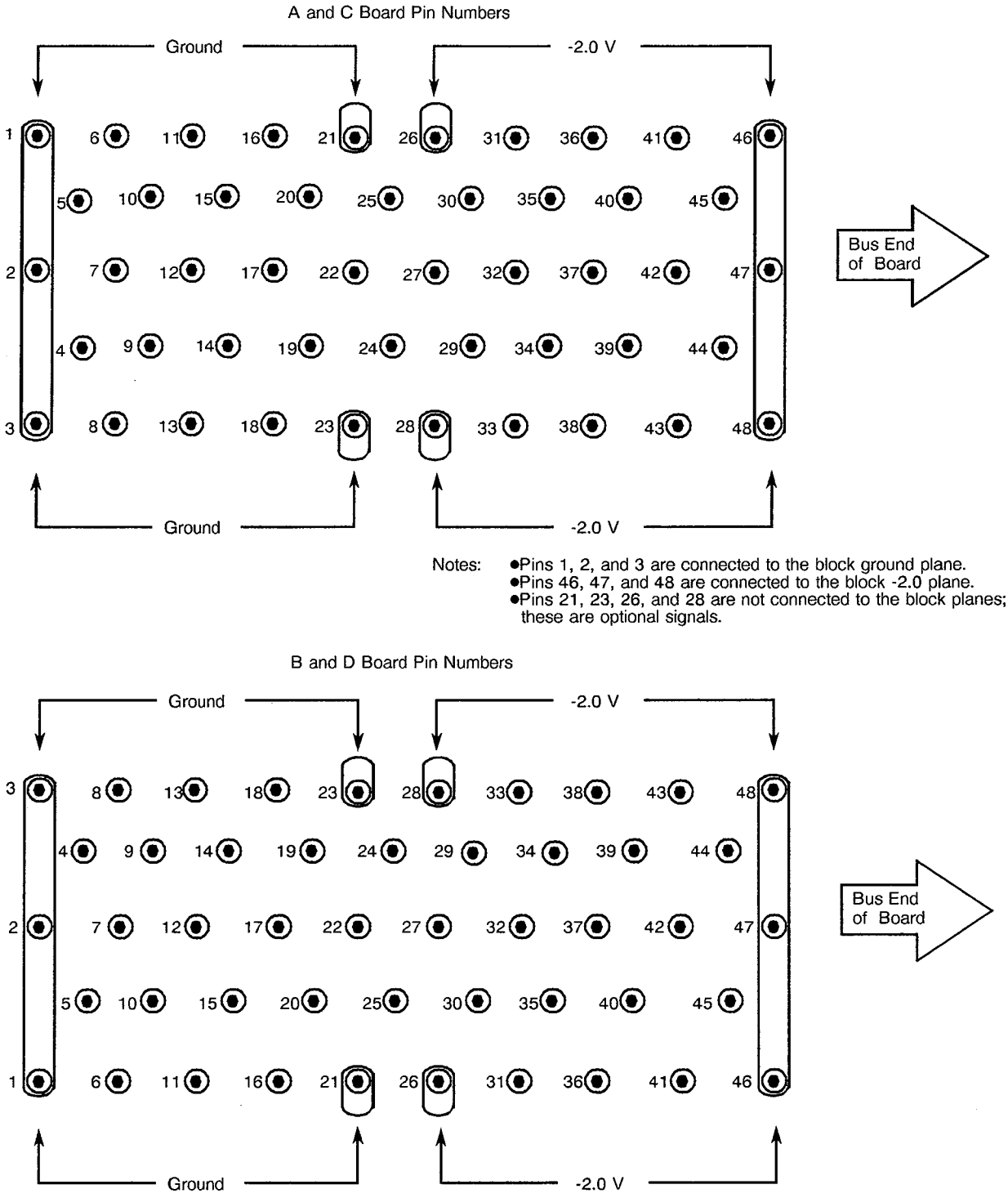
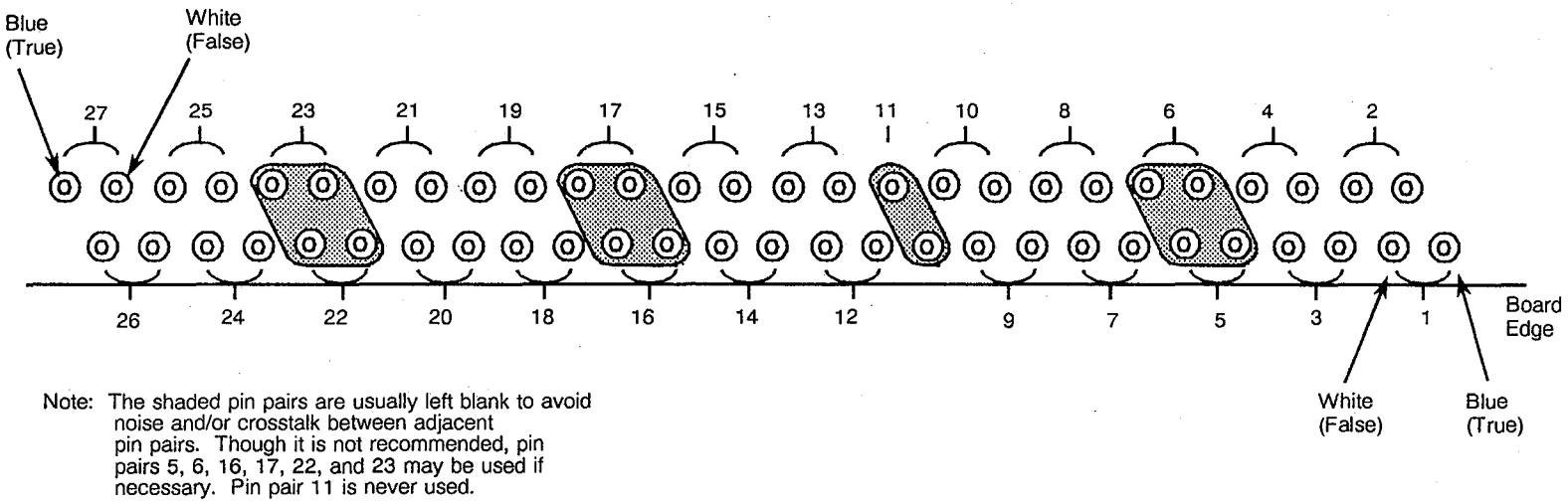


Figure A-10. CRAY Y-MP Feed-through Connector Jumper Pin Assignment



Note: The shaded pin pairs are usually left blank to avoid noise and/or crosstalk between adjacent pin pairs. Though it is not recommended, pin pairs 5, 6, 16, 17, 22, and 23 may be used if necessary. Pin pair 11 is never used.

Figure A-11. Edge Connector Pin Assignments

Connections between different modules and connections between modules and external sources are made through the edge connectors. There are 24 edge connectors on each board: 12 on the Y side and 12 on the Z side. Each connector has 27 pin pairs numbered 1 through 27, arranged as shown in Figure A-11. Each edge connector pin is specified by a two-part logical identifier. The first part consists of the lettersCN followed by a 1- or 2-digit number. The second part is a Boolean I or R term for the module, depending on whether data is entering or exiting the module through the particular pin. An example of an edge connector pin logical identifier is CN7 I40.

Because the logical identifiers do not correspond to the physical edge connectors, the identifiers are not shown in the module board layouts. Instead, three-letter edge connector location designators are shown. The first letter indicates the module side (Y or Z), the second letter indicates the module board (A through D), and the third letter indicates the column (A through L). To determine the edge connector location designator from a logical identifier, refer to either the "Wire Tabs" or the "Connector Maps" documentation described later in this section.

PCBD DOCUMENTS

The four PCBD documents (PCBDA, PCBDB, PCBDC, and PCBDD) each contain a separate chip map and 24 connector maps for one of the four memory boards. PCBDA documents board A, PCBDB documents board B, and so on.

Chip Map

The chip map shows the exact location and orientation of each option. A section of the chip map from PCBDA is shown in Figure A-12, followed by a bulleted description of each column.

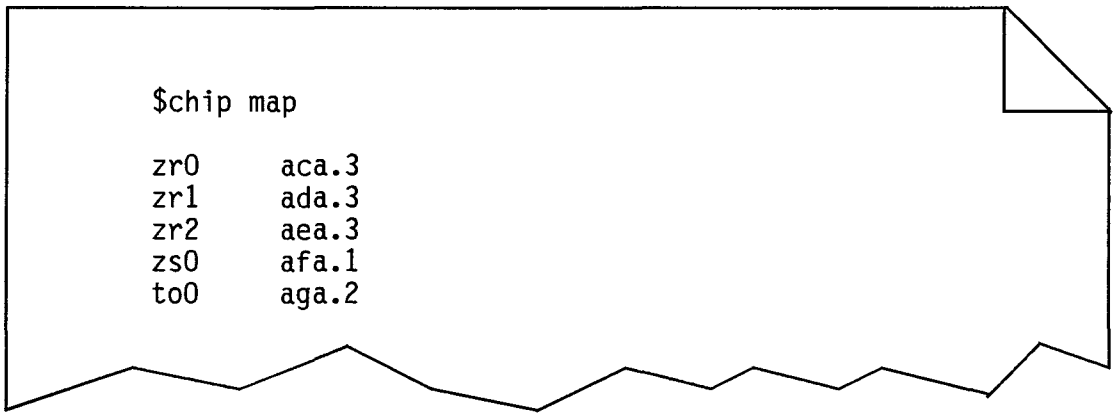


Figure A-12. PCBDA Chip Map

- The left column contains the names of the options. They are not in alphabetical order, but rather in the order in which they are arranged in successive rows on the board.

- The right column contains the three-letter chip location designator followed by a number indicating the orientation, or proper placement of the chip. The first letter of the location designator indicates the board (A through D), the second letter indicates the column (A through M), and the third letter indicates the row (A through F). The four orientations of the chip on the module, numbered 1 through 4, correspond to the four possible placements of pin 1, as shown in Figure A-13. For simplicity, the figure uses a dot to indicate the position of pin 1. To determine the location of pin 1 on a chip, refer to figures A-4 and A-5 in this section.

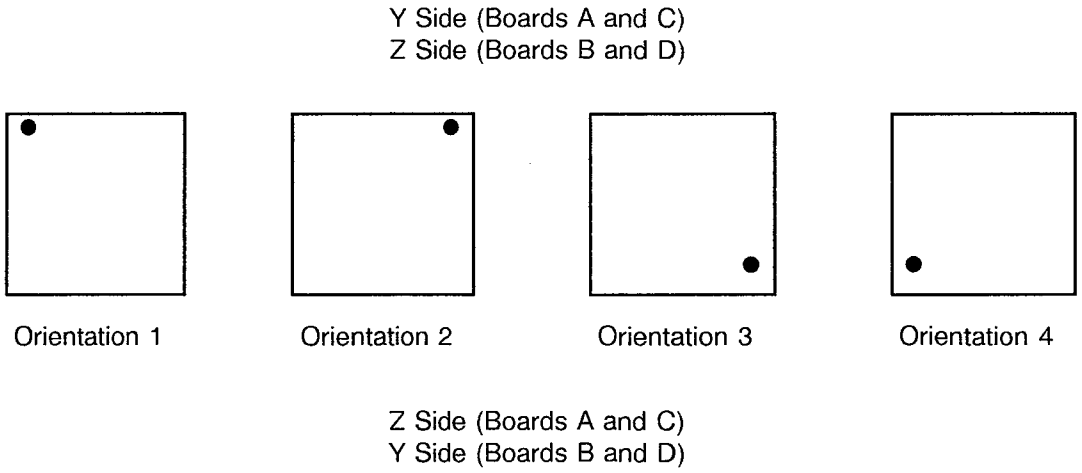


Figure A-13. Chip Orientation

Connector Maps

The connector maps show the relationship between the physical pins in each edge connector and the logical pin identifiers for the module. The connector maps are listed in alphabetical order, first along the Z side, then along the Y side of the board. A section of the connector map for connector ZAD is shown in Figure A-14, followed by a bulleted description of each column.

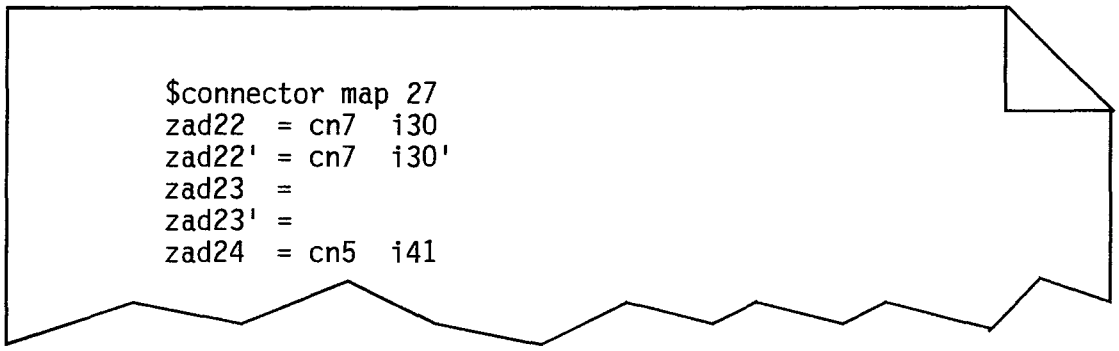


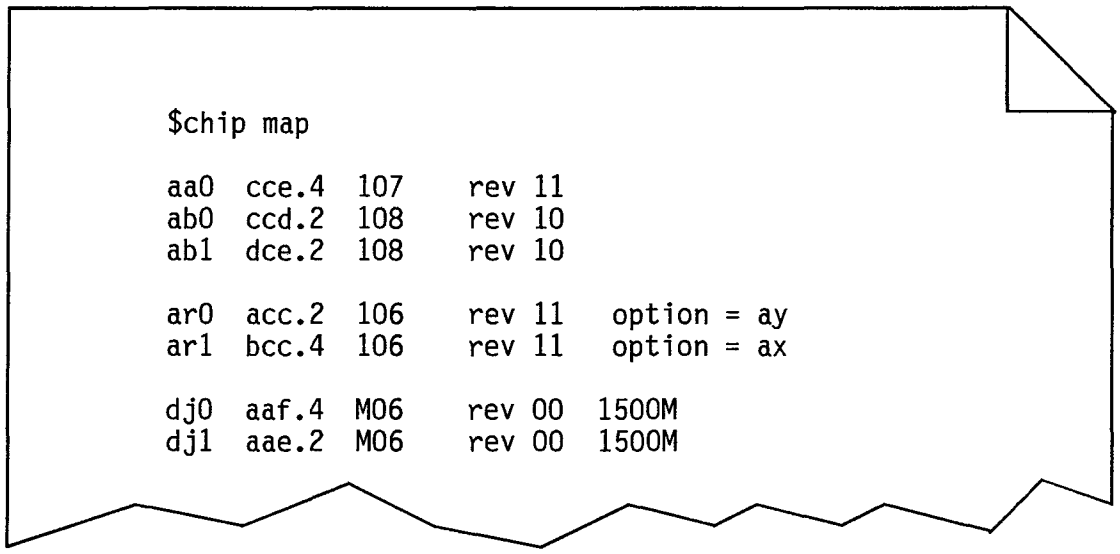
Figure A-14. Connector Map

- The column to the left of the equal sign contains the three-letter edge connector location designator followed by the pin number. The first letter of the location designator indicates the module side (Y or Z), the second letter indicates the module board (A through D), and the third letter indicates the column (A through L). The pin number specifies one of the 54 pins (1 to 27 and 1' to 27') in the connector, as shown in Figure A-9.

- The column to the right of the equal sign contains the logical pin identifiers associated with the respective pins. A blank space indicates that the designated pin is not used.

CPU CHIP MAP

The CPU chip map is similar to the memory chip maps contained in the PCBD documents, in that it shows the exact location and orientation of each option. Unlike the memory chip maps, however, the options are arranged in alphabetical order. A section of the CPU chip map is shown in Figure A-15, followed by a bulleted description of each column.



```
$chip map

aa0  cce.4  107    rev 11
ab0  ccd.2  108    rev 10
ab1  dce.2  108    rev 10

ar0  acc.2  106    rev 11   option = ay
ar1  bcc.4  106    rev 11   option = ax

dj0  aaf.4  M06    rev 00  1500M
dj1  aae.2  M06    rev 00  1500M
```

Figure A-15. CPU Chip Map

- The first column lists the name of the option.
- The second column contains the three-letter chip location designator, followed by a number indicating the orientation of the chip. This is the same notation used in the memory chip maps described earlier.
- The third column lists the number of the option.
- The fourth column lists the revision number. In the case of DJ, DP, HS, or VS options, this column is immediately followed by the notation 1500M, indicating these options contain only 1500 gates and not the standard 2500 gates. The remaining space on these options is occupied by special purpose built-in registers.
- The AR and VA options contain a fifth column listing another name for the option. This name is used in the option load chart listings found in the PCLDCPU document described later in this section.

LOAD CHARTS

The load charts serve two purposes: they show the logical paths from the edge connectors to the I terms on the individual options, and they show the destination of the R terms leaving each option. They also list the status of any forced input terms set by internal wiring; this list is attached to the end of the R terms listing for each option. The organization of the CPU and memory load charts is substantially different, as the following subsections describe. A description of the PCTAB, which is a useful variation of the CPU load chart, is also described.

PCLDCPU Document

This document is the PC load chart for the CPU module. It is several hundred pages long and is divided into two sections. The first section shows the destination of all R terms for each option on the module. The R terms can go to two different places: to other options or to an edge connector for transmission to another module. This section is arranged alphabetically by options. A sample listing for option YC1 is shown in Figure A-16, followed by a bulleted description of the entries.

```
$net list yc1

yc1 r7     cn3 r69                                    .sec 3 cb 69
yc1 r7'    cn3 r69'
yc1 r8     yc0 i56                                    .cb0,byte6 contribution
jumper  if-07
yc1 r8'
yc1 r16    yd1 i45    yc3 i60                         .cb6,byte2 contribution
jumper  hf-36
yc1 r16'
yc1 r16a  yd4 i44    yd5 i44    yd6 i44    yd7 i44   .4-2 port D sscon
jumper  ig-39    ig-40    ig-40    ig-40
yc1 r16a'
force 0    yc1 i24                                    .B Reg.Path, Bits 40-47
force 0    yc1 i25                                    .
```

Figure A-16. First Section of PCLDCPU Document

- Terms R7 and R7' leave the module through edge connector logical pins CN3 R69 and R69'. Their final destination is determined from the wire tabs.
- Term R8 leaves the YC1 option on board B enroute to the YC0 option on board A as input term I56. It travels between boards through pin 7 of the jumper connector at location IF. (The CPU Chip Map described earlier can be used to determine the location of any CPU option.)
- Term R8' is not used.
- Term R16 travels to two different options: YD1 on the same board as input term I45, and YC3 on board D as input term I60. The connection to option YC3 is through pin 36 of the jumper connector at location HF.
- Term R16A refers to the second copy of term R16. (The Boolean should indicate two copies of R16 leaving the option.) It travels to four different options: one on each board. First it leaves board B through jumper IG-39 to option YD4 on

board A; from there it travels through jumper IG-40 to options YD5, YD6, and YD7 on boards B, C, and D respectively.

- At the end of the R term listing, the status of the internally-wired forced terms is shown.

The second section of the PCLDCPU document indicates the logical connections between each of the edge connector logical pin identifiers and the input terms on the individual options. A sample listing from CN4 is shown in Figure A-17.

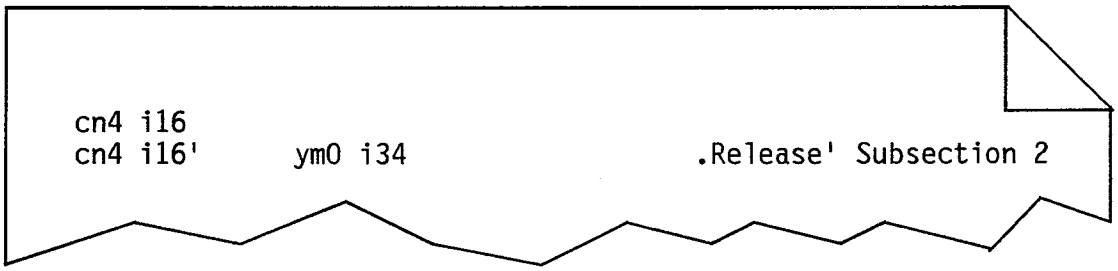
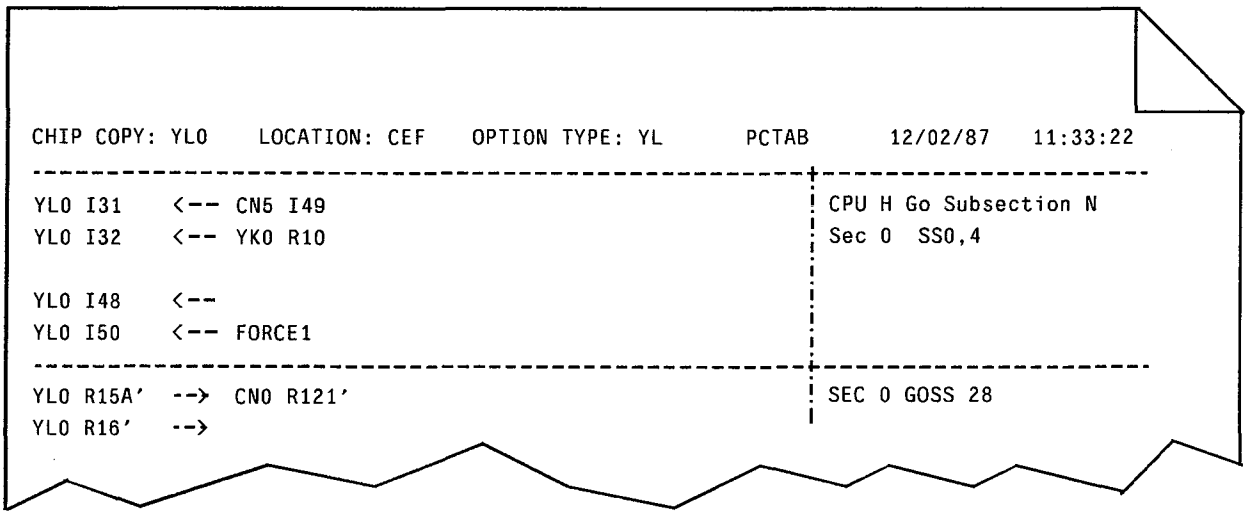


Figure A-17. Second Section of PCLDCPU Document

In this example, the edge connector pin identified as pin CN4 I16 is not used, while pin CN4 I16' goes to option YM0 as input term I34. To find the original source of this input term from another module, as well as to determine the physical location of the edge connector corresponding to the logical identifier CN4 I16', refer to the "Wire Tabs" documentation.

PCTAB Document

The PCTAB document is essentially the same document as the CPU load chart, PCLDCPU, and has the same two sections. There are, however, two notable differences. First, the listings of each option in the first section show not only the destinations of the R terms, but also the sources of the incoming I terms. This is the only documentation available that directly shows the sources of these terms. Second, the jumper connections that the I or R terms pass through enroute to the individual options are not indicated. A sample listing from the first section showing part of the listing for option YL0 is shown in Figure A-18, followed by a bulleted description of each term.



```
CHIP COPY: YL0    LOCATION: CEF    OPTION TYPE: YL      PCTAB      12/02/87   11:33:22
-------------------------------------------------------+-----------------------
YL0 I31    <-- CN5 I49                                 | CPU H Go Subsection N
YL0 I32    <-- YK0 R10                                 | Sec 0  SS0,4
                                                       |
YL0 I48    <--                                         |
YL0 I50    <-- FORCE1                                  |
-------------------------------------------------------+-----------------------
YL0 R15A'  --> CN0 R121'                               | SEC 0 GOSS 28
YL0 R16'   -->                                         |
```

Figure A-18. First Section of PCTAB Document

- Term I31 comes into the YL0 option from edge connector logical pin CN5 I49.
- Term I32 comes into the YC0 option on board C from term R10 of the YK0 option on board A. Note that the jumper connector used between boards is not indicated. To find this connector, consult the R term listing for YK0 in the first section of PCLDCPU.
- Term I48 is left open.
- Term I50 is an internally wired forced 1.
- The second copy of term R15' is sent to edge connector logical pin CN0 R121'.
- Term R16' is not used.

A sample listing of the second section of the PCTAB document is not shown because it is very similar to the corresponding listing for the PCLDCPU document described earlier.

PCLD, Subsection Data, Bank Subsection Documents

The PCLD, Subsection Data, and Bank Subsection documents make up the PC load chart for the memory module. There are four PCLD documents, four Subsection Data documents, and eight Bank Subsection documents. Together they contain the same information for the memory module as the PCLDCPU document does for the CPU module. However, the R term data is not arranged alphabetically by option in one listing, but is grouped by function (bank selects, go reads, and so on) and is spread over the 12 documents. The portion of the load chart covered in each document is explained in the following subsections.

PCLD Documents

Each of the four PCLD documents (PCLDA, PCLDB, PCLDC and PCLDD) contain information on one board of the memory module. This information is grouped under 22 different headings according to the function of the terms. The logical connections between the edge connectors and the options are listed in seven of these groups. Most of these connections are listed at the beginning of each document under two separate headings: $net list cnM and $net list cnN, where M and N stand for 0 and 1 on PCLDA, 2 and 3 on PCLDB, and so on. The rest of the connections are listed in five other places throughout each document, under the following headings:

- $ z connector fan-out inputs
- $ y connector fan-outs
- $ y connector cpuX ss select
- $ y connector cpuY ss select
- $ clock

Sample listings of $ net list cn0 and $ z connector fan-out inputs fromPCLDA are shown in Figures A-19 and A-20, followed by a bulleted description of each edge connector pin.

```
$ net list cn0      ************  cpu0  ****************

cn0 i55    zx33 i74                                      .goss 7/bank bit 2
cn0 i55'                                                 .
cn0 i56    zx0  i0   zx10 i70  zx20 i0   zx30 i70       .bank bit 0
jumper               CF-33     CF-33     CF-33
```

Figure A-19. $ Net List Cn0 listing from PCLDA

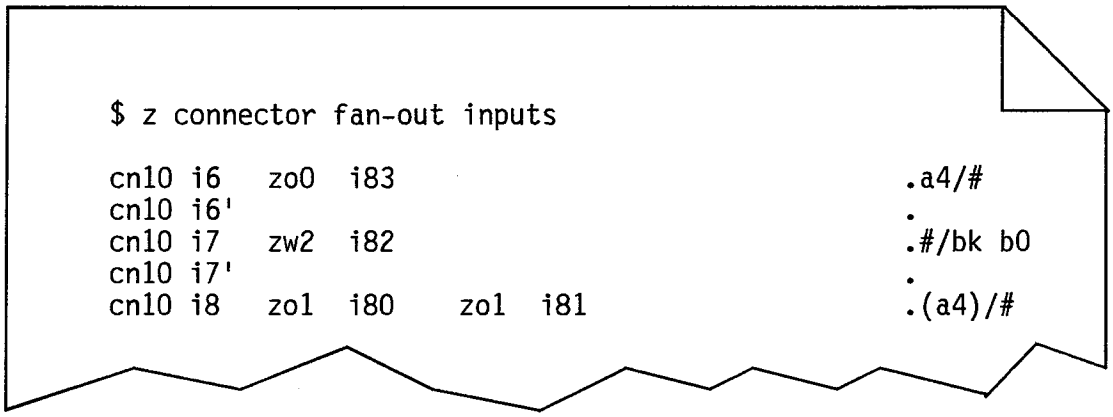
- In this example the edge connector pin identified as CN0 I55 connects to input term I74 on the ZX33 option. This is the Go Subsection 7/bank bit 2 signal from CPU 0. (In these $net list listings, the number following CN actually refers to the CPU using this path between the edge connector and the option.) To find the original source of this input term from another module, as well as to determine the physical location of the edge connector corresponding to the logical identifier CN0 I55, refer to the "Wire Tabs" documentation.

- Edge connector pin CN0 I55' is not used.

- Edge connector pin CN0 I56 brings in bank bit 0 from CPU 0 and routes it to the ZX0 option as term I0. It also sends it to options ZX10 (board B), ZX20

(board C), and ZX30 (board D) through pin 33 of the jumper connector at location CF.

Note: For memory options, the first digit of the option number indicates the board on which the option is found. A 0 denotes board A, 1 denotes board B, and so on. Thus option ZY21 would be on board C, while option ZR12 would be on board B.



```
$ z connector fan-out inputs

cn10 i6   zo0  i83                          .a4/#
cn10 i6'                                    .
cn10 i7   zw2  i82                          .#/bk b0
cn10 i7'                                    .
cn10 i8   zo1  i80    zo1  i81              .(a4)/#
```

Figure A-20. $ Z Connector Fan-out Inputs listing from PCLDA

The information in $ Z Connector Fan-out Inputs uses the following notations in its comments column:

Comment	Explanation
a4	Address bit 4
bk b0	Bank bit 0
(a4)	Complement of a4
#	Pin not used
A/B	See the following description

A/B indicates two relative positionings of the memory module within the chassis. When the module is inserted into an A position slot, the comment to the left of the / applies to the term, and when the module is inserted into a B position slot, the comment to the right of the / applies to the term. The chassis slots considered A positions and B positions are shown in Table A-1 for a section of memory.

Thus, if a memory module is inserted in an A position slot, address bit 4 enters edge connector pin CN10 I6 enroute to the ZO0 option as term I83. However, if the same module is inserted in a B position slot, edge connector pin CN10 I6 is not used.

Table A-1. Memory Module Positions

Position	Memory Module
A	Memory section n data bits 0 to 8
A	Memory section n data bits 9 to 17
B	Memory section n data bits 18 to 26
B	Memory section n data bits 27 to 35
B	Memory section n data bits 36 to 44
B	Memory section n data bits 45 to 53
A	Memory section n data bits 54 to 62
A	Memory section n data bit 63, check bits 0 to 7

The remaining groups of information on the PCLD documents contain listings of the R terms for each option showing their destinations. However, any R terms that directly enter the memory chips do not appear in these listings, but are shown in the Bank Subsection and Subsection Data documents described later. These R terms include memory write data, the memory address, and the write enable and chip select controls. All other R terms are shown on the PCLD documents and are grouped according to their function; they are not arranged alphabetically by options. Some of the groupings are Go Subsection Fan-out, Bank CPU Select, Go Read CPU a-h, and so on. Sample listings from the groups titled $delays and $subsection 1: ss cpuh are shown in Figures A-21 and A-22, followed by a bulleted description of each term.

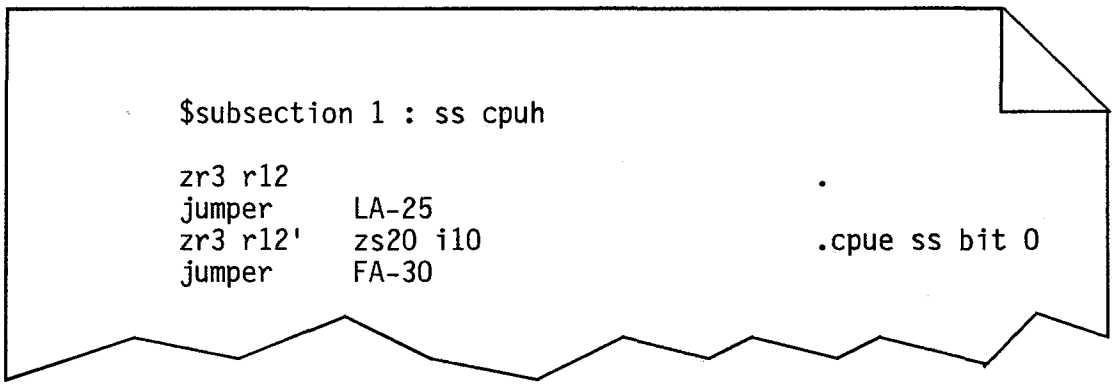
```
$delays : address a0-a16,data d0-d8,go write cpu0,1

za1 r33    zm0 i11    zm10 i61   zm20 i11   zm30 i61        .cpu1 addr 9
jumper                DE-07      DE-07      DE-07
za1 r33'   zu1 i11    zu11 i61   zu21 i11   zu31 i61        .
jumper                JE-15      JE-15      JE-15
force1     za1 i100
force1     za1 i8
```

Figure A-21. $ Delays listing from PCLDA

- Output term R33 from option ZA1 (board A) is sent to option ZM0 (board A) as input term I11, to option ZM10 (board B) as input term I61, to option ZM20 (board C) as input term I11, and to option ZM30 (board D) as input term I61. The last three connections are made through Pin 07 of the jumper connector at location DE.

- Note that output term R33' is not sent to any of the same options as the true term R33.
- The listing of any internally-wired forced terms on an option always occurs immediately after the last R term for the option, even though some of the other R terms may be listed under another heading. In the case of the ZA1 option shown in the example, terms R8 through R15 do indeed occur later in the PCLDA document under the heading $ z connector 1-8 fan-outs.



```
$subsection 1 : ss cpuh

zr3 r12                                  .
jumper      LA-25
zr3 r12'    zs20 i10                     .cpue ss bit 0
jumper      FA-30
```

Figure A-22. $ Subsection 1: ss cpuh listing from PCLDA

- In this listing, term R12' from option ZR3 (board A) is sent through pin 30 of the jumper connector at location FA to option ZS20 (board C) as input term I10.
- Term R12, while it is not used, it still sent through jumper LA-25 to another board before it is terminated.

Subsection Data Documents

There are four Subsection Data documents, one for each board of the memory module. Each document contains information on the writing or reading of data for two subsections of memory. Subsection Data A contains subsections 0 and 1; Subsection Data B contains subsections 2 and 3; Subsection Data C contains subsections 4 and 5; and Subsection Data D contains subsections 6 and 7. The documentation shows both the write and read paths of memory data. These two separate paths are shown on alternate lines of the listing. The documentation traces the logical path of each memory write data bit from the R term of the option immediately preceding the memory chip to the I terms of two memory chips. One chip is then selected by the chip select. For read data, the documentation traces the logical path of each data bit from one of two memory chips (depending on the chip select) to the I term of the appropriate ZR option. A sample listing from Subsection Data A is shown in Figure A-23, followed by a bulleted description of the write and read paths.

```
$ subsection 0 : data bits

zw1  r4'   zza3 i305  zza3 i705        .bank 2 data bit 2
zza3 r3    zza3 r7    zr0  i8
zw1  r5'   zza7 i005  zza4 i205        .bank 2 data bit 3
zza7 r0    zza4 r2    zr1  i6
```

Figure A-23. $ Subsection 0 listing from Subsection Data A

- The first column of the write path description shows the R term and option that the data comes from.
- The last two columns of the write path description show the destination memory chips for the data: ZZA3 denotes the memory chip group (8 chips per group), and I305 specifies chip 3, pin 5, as shown in Figure A-24.

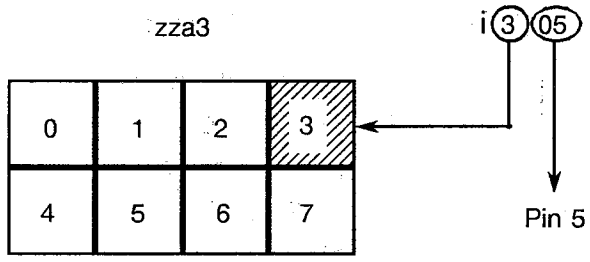


Figure A-24. Memory Write Data Destination

In the example, bank 2, data bit 2 comes from option ZW1, term R4' enroute to chips 3 and 7 of memory chip group ZZA3, on pin 5. Though both chips receive the Write Enable, only one receives the chip select.

- The read path description is similar, except that the R term itself indicates the specific chip of the memory chip group from which the data comes. Thus bank 2, data bit 2 comes from memory chip group ZZA3, chip 3 or 7 (depending on the chip select), as term R3 or R7 enroute to option ZR0 as input term I8.

Bank Subsection Documents

There are eight Bank Subsection documents, one for each subsection (0 through 7). They document the paths of the memory address, the write enable, and the chip select from the R term of the option immediately preceding the memory chips to the I terms of nine

separate memory chips. The memory chips used are indicated by the same notation used in the Subsection Data documentation. Although the source options are not listed alphabetically and there are no comments in these documents, the information is grouped into eight sections. Each section contains information on a particular bank (0 through 7) and is separated from the other sections by a blank line in the listings. There are 36 bits of information per bank and they occur in the listings in the following sequence:

1. Address bit 0
2. Address bit 0'
3. Address bit 1
4. Address bit 1'

.
.
.

31. Address bit 15
32. Address bit 15'
33. Chip Select 0
34. Chip Select 1
35. Write Enable
36. Write Enable.

A sample listing of Bank Subsection 1 is shown in Figure A-25, followed by a bulleted description of each R term.

```
zn2  r6    zzc8 i419  zzc8 i019  zzc5 i319  zzc7 i419  zzc7 i019 &
           zzc4 i319  zzc6 i419  zzc6 i019  zzc3 i319
zn2  r6'   zzc5 i719  zzc5 i619  zzc5 i219  zzc4 i719  zzc4 i619 &
           zzc4 i219  zzc3 i719  zzc3 i619  zzc3 i219
```

Figure A-25. Bank Subsection 1

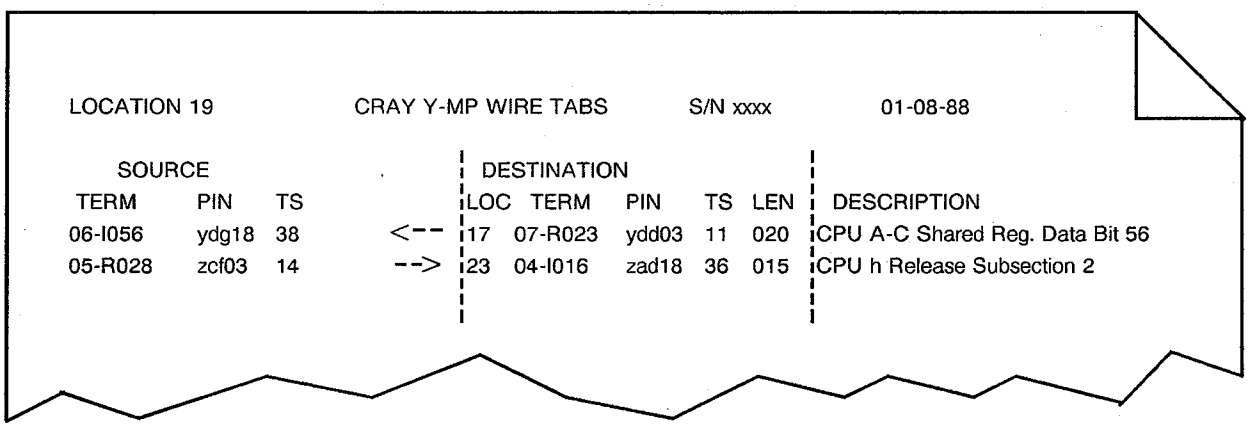
- The leftmost column shows the R term and option that the information comes from. This is followed by a two-line listing showing the nine chips that the term goes to. Thus, subsection 1, bank 2, address bit 0 comes into the memory chips from option ZN2 as term R6. It goes to chips 0 and 4 of memory chip groups ZZC6, ZZC7, and ZZC8, and to chip 3 of memory chip groups ZZC3, ZZC4, and ZZC5 on pin 19.

- An inverted copy of this bit comes from option ZN2 as term R6', and goes to chips 2, 6, and 7 of memory chip groups ZZC3, ZZC4, and ZZC5 on pin 19.

WIRE TABS

The Wire Tabs document lists the interconnections between the modules in the mainframe. It traces a bit from the R term of an edge connector on one module to the I term of an edge connector on another module. It can also be used to find the module and R term coming into an input I term on a given module. In addition, it tells the physical location of the pin corresponding to the logical pin identifier.

The document contains 41 sections, one for each module in the chassis, numbered from 1 through 41. To find the destination of an R term leaving CPU0, consult the listing for location 17; to find the source of an I term, such as data bit 6, entering memory section 1, consult the listing for location 9. A sample listing for location 19 showing an I and R term is shown in Figure A-26, followed by a bulleted description of each column.



LOCATION 19 CRAY Y-MP WIRE TABS S/N xxxx 01-08-88

SOURCE				DESTINATION					
TERM	PIN	TS		LOC	TERM	PIN	TS	LEN	DESCRIPTION
06-I056	ydg18	38	<--	17	07-R023	ydd03	11	020	CPU A-C Shared Reg. Data Bit 56
05-R028	zcf03	14	-->	23	04-I016	zad18	36	015	CPU h Release Subsection 2

Figure A-26. Wire Tab Listing for Location 19

- The first column, labeled TERM, contains the logical pin identifiers for the I or R terms, in this example, CN6 I56 or CN5 R28.
- The second column lists the physical location of the pin identified in the first column. Thus, term I56 enters the module through pin 18 of the edge connector along the Y side of the module, board D, location G; and term R28 leaves the module through pin 3 of the edge connector along the Z side of the module, board C, location F.
- The third column indicates the time segment at which the term enters or exits the module.
- The fourth column lists the location of the source or destination module.
- The fifth, sixth, and seventh columns are similar to columns 1 through 3, but with source and destination information reversed.
- The column marked LEN tells the length of the wire in inches between the two modules.
- The last column contains a description of the term.