CM02 MEMORY MODULE

PRELIMINARY INFORMATION
DO NOT DISSEMINATE

Figures

Figures (continued)

Tables

CM02 Module Description

The CM02 memory module contains the central memory that is common to all processors in a CRAY T90 series mainframe. The memory module consists of a printed circuit board with memory stacks on board 1 and logic options on board 2. Memory stacks contain the memory chips that store the data. When fully populated with 4 Mbit (2 Mbit x 2) chips, a CM02 memory module contains 32 Mwords of memory with 2 million words per bank.

Memory modules are arranged in stacks within the mainframe. A CRAY T932 system can include two or four module stacks with each stack containing two, four, or eight memory modules. A CRAY T916 system can include two module stacks with each stack containing either four or eight memory modules. A CRAY T94 system includes only one module stack that contains either two or four memory modules.

In CRAY T916 and CRAY T932 systems, each module stack connects to four network modules. The network modules enable all processors to communicate with memory at the same time. In CRAY T94 systems, a module stack connects directly to the CP modules; therefore, the final levels of memory interconnecting and routing take place on the memory modules.

Memory Organization

Central memory is organized by sections, subsections, and banks; however, CRAY T94 systems do not have subsections. Refer to the "CRAY T94 Memory," "CRAY T916 Memory," and "CRAY T932 Memory" subsections on pages 9, 12, and 16, respectively, for information on memory organization for the specific CRAY T90 series systems.

Sections

A memory section is the range of memory components that can be the destination of a request from a CPU through a single path. In a CRAY T90 series system, each memory section is independent and corresponds with one of eight CPU memory access paths. Each access path has its own connector; therefore, each CPU can access any section of memory. During a read request, a CPU can make a request to each access path each clock period. During a write request, a CPU can make a request to each access path every two clock periods.

A section conflict occurs when one CPU tries to address the same section of memory more than once during the same clock period. The CPU resolves a section conflict by sending one request and delaying each subsequent request for 1 clock period.

Subsections

In CRAY T916 and CRAY T932 systems, each section is divided into subsections. Each subsection corresponds to one memory module in a memory module stack. There are no subsections in a CRAY T94 system.

A subsection conflict occurs when two CPUs in the same CP module stack request the same subsection. The network module resolves a subsection conflict by sending one request and delaying each subsequent request for 1 clock period.

Banks

Banks are organized into bank pairs as n/n+4 (for example 0/4, 1/5, 2/6, and so on). Each memory stack stores the upper or lower bits of the bank pair. Each memory bank is an independent unit of memory and is controlled as such; therefore, requests to different memory bank pairs do not cause access conflicts.

A bank conflict occurs when a CPU requests a bank that is busy or if two CPUs in two different CP module stacks request the same bank. The memory module resolves a bank conflict by delaying the request for "bank busy time."

Word

Each request to a bank receives one 64-bit data word (along with the 12 error-correction bits) from the requested bank.

Memory Configuration

Software address mapping through the maintenance channel controls memory configuration. The System Configuration Environment (SCE) provides the interface for selecting sections, subsections, banks, and groups in a particular configuration to control memory degrades and/or logical memory partitions.

When you degrade memory, you are bypassing areas in memory where failures occur; then, the system can continue to operate and you may perform system maintenance on the degraded portion of memory. By degrading memory, you are forcing selected section, subsection, and bank bits. Therefore, the memory address referenced by the CPU is different than the address received by memory. Figure 1 illustrates an example of bits 2, 5, 9, 10, and 11 being forced to either a 1 or a 0, which degrades the lower sections (0 through 4) and the upper subsections (4 through 7) of memory.

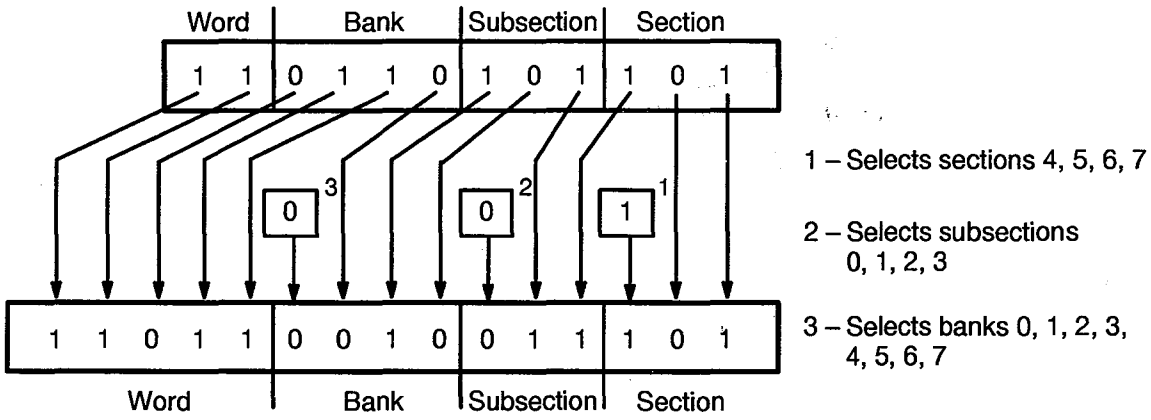


Figure 1. Example of Degrading Sections 0–3 and Subsections 4–7

Degrading memory should be done on a system-wide basis. That is, if your mainframe has logical partitions, each partition is affected when you degrade memory.

By logically partitioning memory, you are forcing CPUs and I/O channels to access different memory partitions. When you logically partition memory, you are forcing one or two group bits to either a 0 or a 1, which divides memory in either halves or quarters. Table 1 shows how SCE logically partitions memory. For more information on SCE and degrading and partitioning memory, refer to the *System Configuration Environment*, publication number HDM-xxx-x.

All configuration/degradation selections apply symmetrically within a processor; for example, selecting 4-bank mode within a CPU means that all requests on all memory ports will use 4-bank addressing. It is not possible, for example, to use 4-bank addressing through one port and 8-bank addressing through the remaining ports. The CRAY T94, CRAY T916, and CRAY T932 memory subsections describe the configurations available for the specific systems.

Table 1. Logical Memory Partitions (SCE Default)

Number of Partitions	Partition	Group Profile Bits †		Group Select Bits †		Memory Size
		1	2	1	2	
1	0	--	--	--	--	All
2	0 1	NU NU	1 1	NU NU	0 1	1/2 1/2
2	0 1	1 1	NU NU	0 1	NU NU	1/2 1/2
3 ‡	0 1 2	NU 1 1	1 1 1	NU 0 1	0 1 1	1/2 1/4 1/4
4	0 1 2 3	1 1 1 1	1 1 1 1	0 0 1 1	0 1 0 1	1/4 1/4 1/4 1/4

† NU means that the profile/select is not set. This is not an SCE function.
‡ There are other ways of getting 3 partitions.

Memory Address Mapping

Memory addressing is dependent on how a CRAY T90 series system is configured. The CP, network, and memory modules determine how memory addressing occurs. Table 2 describes the address map used to design the CRAY T90 series CPU. This address map accommodates future CRAY T90 series systems and is capable of addressing up to 16 Gwords of memory. The "CRAY T94 Memory," "CRAY T916 Memory," and "CRAY T932 Memory" subsections include address maps for the specific CRAY T90 series systems.

Table 2. Address Map

Address Bits	Function
0 – 2	Section select
3 – 5	Subsection select
6 – 9	Bank select
10 – 34	Word select †

Bits 10 and 11 of a word address are used for group selection.

CRAY T94 Memory

CRAY T94 memory is organized by sections and banks. A fully configured system has 8 sections of memory. Each section contains 8 banks for a total of 64 banks. Figure 2 shows memory organization in a CRAY T94 system.

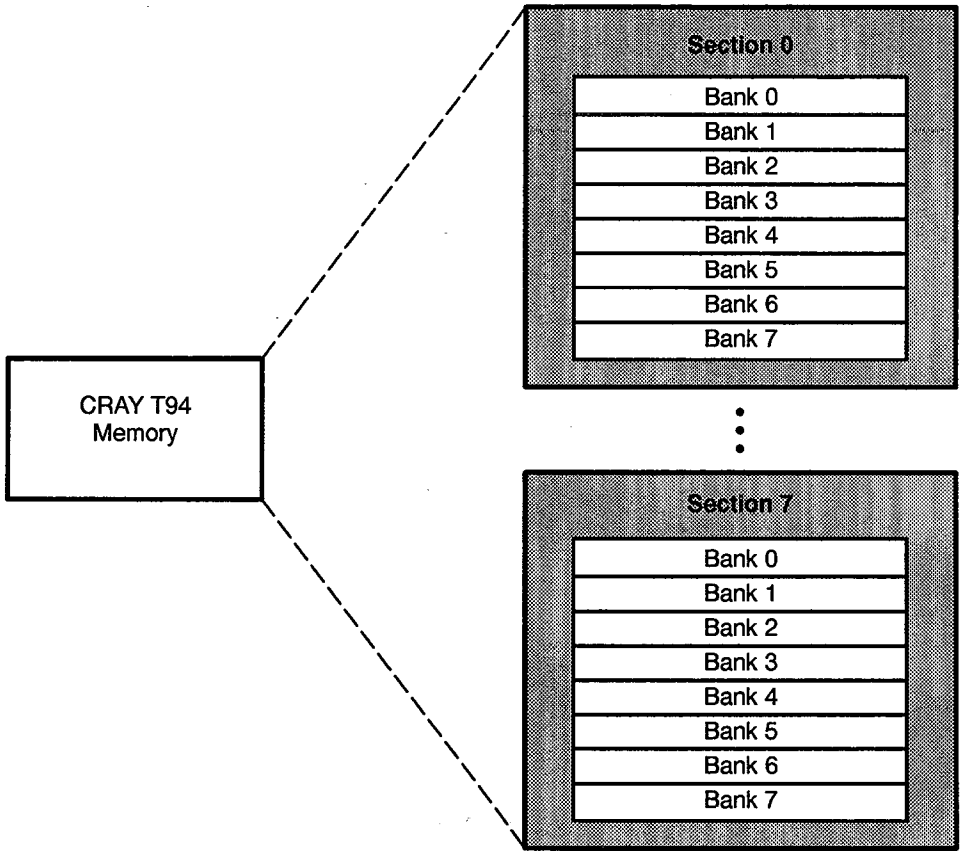


Figure 2. CRAY T94 Memory Organization

The CP and memory modules determine which section, bank, and word to address. Figure 3 shows the addressing map for a fully configured CRAY T94 system and the module type that determines the section select, bank select, and word select. Bits 27 through 30 are presently not used; however, these bits may be used in future systems to address up to 2 Gwords of memory.

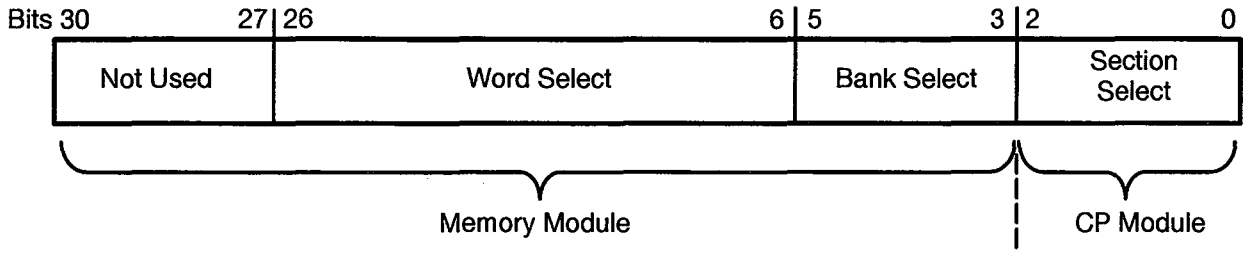


Figure 3. CRAY T94 Addressing (Full Memory and No Degrading)

The address bit layout varies for the different CRAY T94 configurations. Table 3 lists the available CRAY T94 configurations and the address bit layout for each of the configurations.

Table 3. CRAY T94 System Configurations

Module Counts			Section Configuration		Address Bit Layout		
Processor	Network	Memory	Sections	Banks	Section	Bank	Word Address
1 to 4	0	4	8	64	00, 01, 02	03, 04, 05, –	06 – 30
1 to 4	0	4	8	32	00, 01, 02	03, 04, –, –	05 – 29
1 to 4	0	2	4	32	00, 01, –	02, 03, 04, –	05 – 29
1 to 4	0	2	4	16	00, 01, –	02, 03, –, –	04 – 28

A CRAY T94 system has one memory module stack containing either two or four memory modules. The memory modules connect directly to the CP modules. Figure 4 shows the interconnections between the CP modules and memory modules in a CRAY T94 system. Each connection between the modules in the CP module stack and the memory module stack is actually two connections; one physical connector on the CP module connects to two memory section access paths.

Each memory module in a CRAY T94 system comprises two sections of memory. Figure 4 shows the memory module layout for a CRAY T94 system.

Assuming the first reference is to section 0 and the memory references are sequential, the first two references go to sections 0 and 1 (bank 0) on the memory module at location C2, the next two references to sections 2 and 3 to the memory module at location C4, and so on, until each module in the stack receives two references. References nine through sixteen follow the

same pattern but address bank 1, references 17 through 24 address bank 2, and so on, until all banks (64 in a fully configured system) have been referenced.

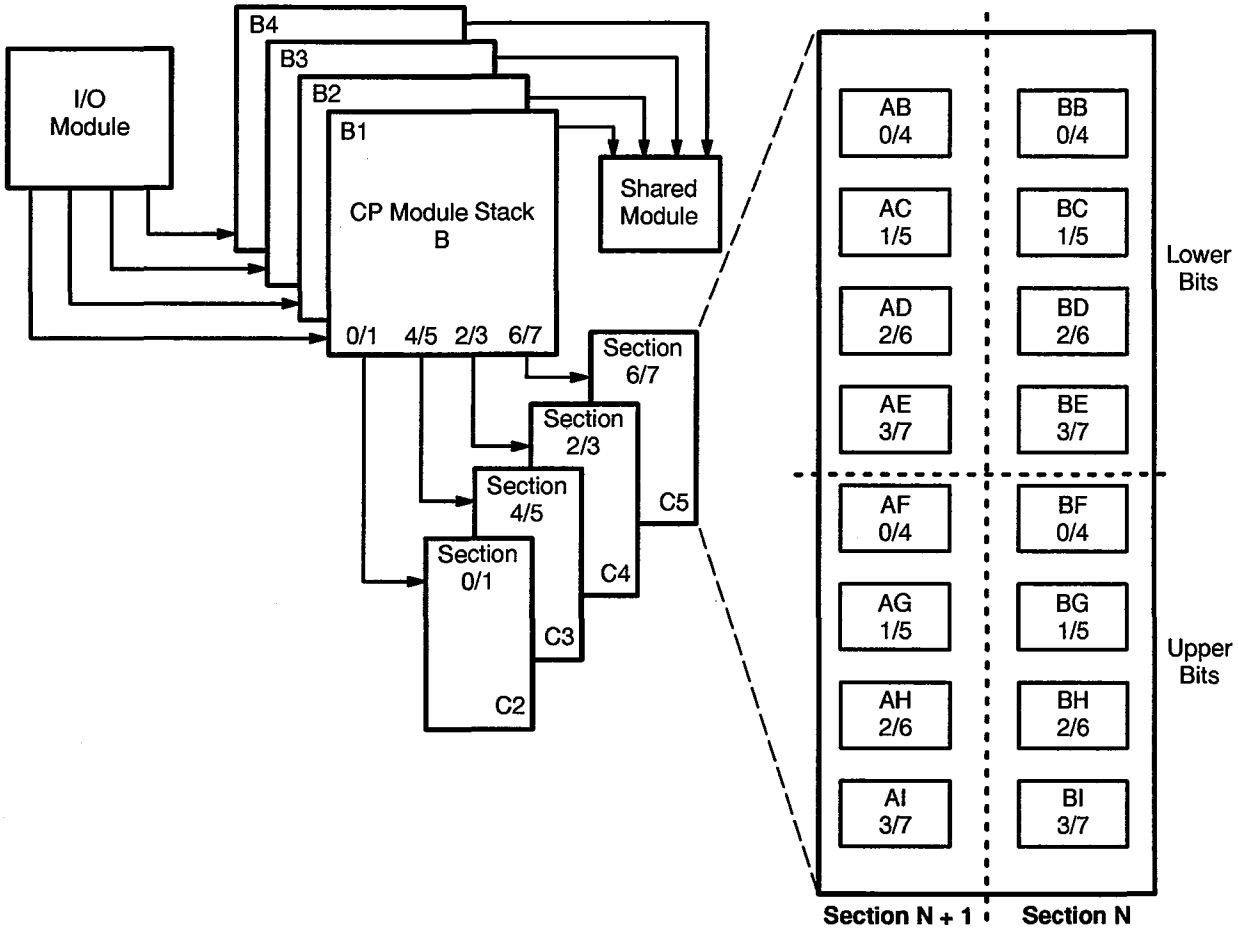


Figure 4. CRAY T94 CPU-to-Memory Interconnections

CRAY T916 Memory

CRAY T916 memory is organized by sections, subsections, and banks. A fully configured system has 8 sections of memory. Each section has 8 subsections and each subsection has 4 banks for a total of 256 banks. Figure 5 shows memory organization in a CRAY T916 system.

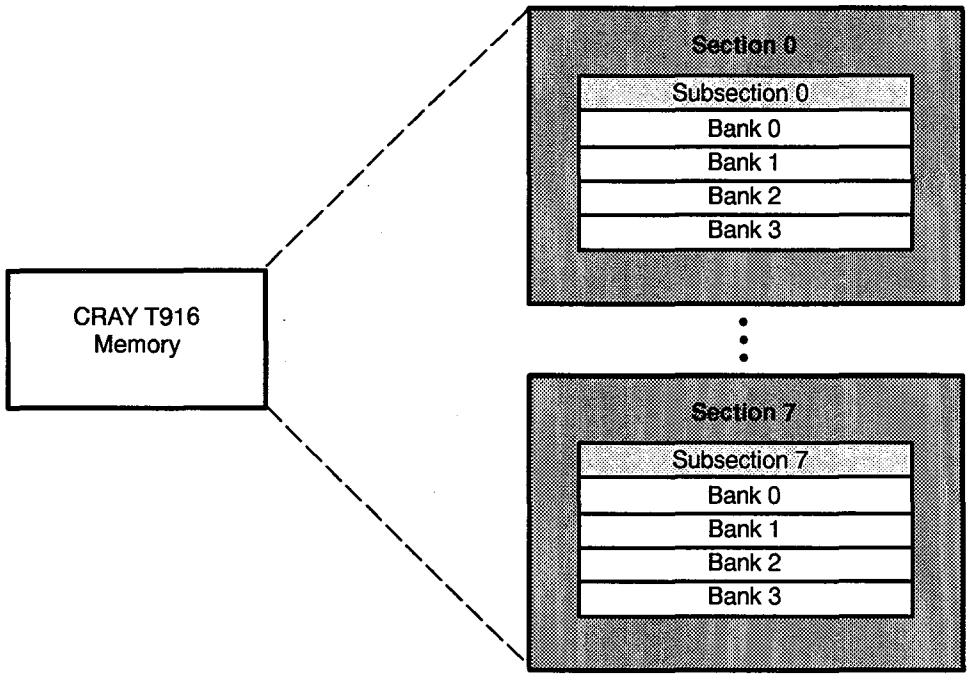


Figure 5. CRAY T916 Memory Organization

The CP, network, and memory modules determine which section, subsection, bank, and word in memory to address. Figure 6 shows the addressing map for CRAY T916 memory and the module type that determines the section select, subsection select, bank select, and word select. Bits 29 through 32 are presently not used; however, these bits may be used in future systems to address up to 4 Gwords of memory.

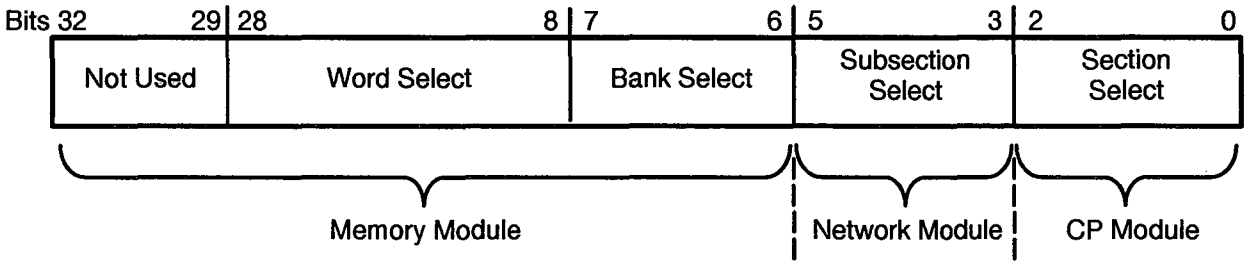


Figure 6. CRAY T916 Addressing

The address bit layout varies for the different CRAY T916 systems. Table 4 lists the various configurations of CRAY T916 systems and the address bit layout for each configuration.

Table 4. CRAY T916 System Configurations

Module Counts			Section Configuration			Address Bit Layout			
Processor	Network	Memory	Sections	Sub-sections	Banks	Section	Sub-section	Bank	Word Address
4 to 8	4	8	8	4	128	00, 01, 02	–, 03, 04	05, 06, –, –	07 – 31
4 to 8 †	4	8	8	2	64	00, 01, 02	–, –, 03	04, 05, –, –	06 – 30
4 to 8 †	2	8	4	4	64	00, 01, –	–, 02, 03	04, 05, –, –	06 – 30
4 to 8 †	2	4	4	2	32	00, 01, –	–, –, 02	03, 04, –, –	05 – 29
4 to 8	4	16	8	8	256	00, 01, 02	03, 04, 05	06, 07, –, –	08 – 32
4 to 8 †	4	8	8	4	128	00, 01, 02	–, 03, 04	05, 06, –, –	07 – 31
4 to 8 †	2	8	4	8	128	00, 01, –	02, 03, 04	05, 06, –, –	07 – 31
4 to 8 †	2	4	4	4	64	00, 01, –	–, 02, 03	04, 05, –, –	06 – 30
8 to 16	8	16	8	8	256	00, 01, 02	03, 04, 05	06, 07, –, –	08 – 32
8 to 16 †	8	8	8	4	128	00, 01, 02	–, 03, 04	05, 06, –, –	07 – 31
8 to 16 †	4	8	4	8	128	00, 01, –	02, 03, 04	05, 06, –, –	07 – 31
8 to 16 †	4	8	4	4	64	00, 01, –	–, 02, 03	04, 05, –, –	06 – 30

† This indicates a degraded or partitioned system; the system is not usually sold with this configuration.

A CRAY T916 mainframe has two memory module stacks consisting of four or eight memory modules. The memory modules connect to the network modules, which connect to the CP modules. Connections between network and CP modules are actually through the system interconnect board (SIB).

Figure 7 shows the logical interconnections between the CP modules and memory modules in a CRAY T916 system.

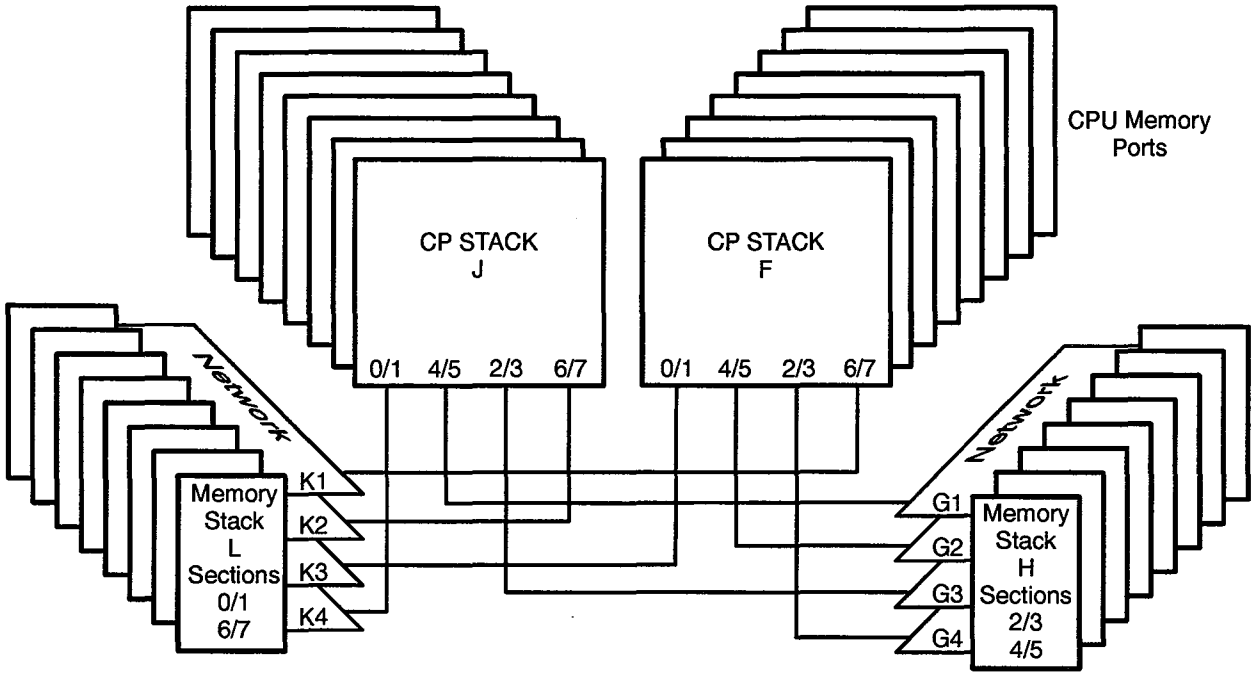


Figure 7. CRAY T916 CPU-to-Memory Interconnection

Each memory module stack handles 4 sections of memory. Each memory module within the stack is 1 subsection. Each subsection has 4 banks (instead of 8 banks as in the CRAY T932 and CRAY T94 systems). Figure 8 shows the memory module layout for a CRAY T916 system.

Assuming the first reference is to section 0 and the memory references are sequential, reference one goes to section 0, subsection 0 at module location L1. Reference two goes to the same module but to section 1. References three through six go to sections 2, 3, 4, and 5 and subsection 0 on the memory module at location H1. References seven and eight go to sections 6 and 7 subsection 0 on the memory module at location L1. The next eight references follow the same sequence for subsection 1 but reference the modules at locations L3 and H3. This referencing pattern continues until all banks have been addressed.

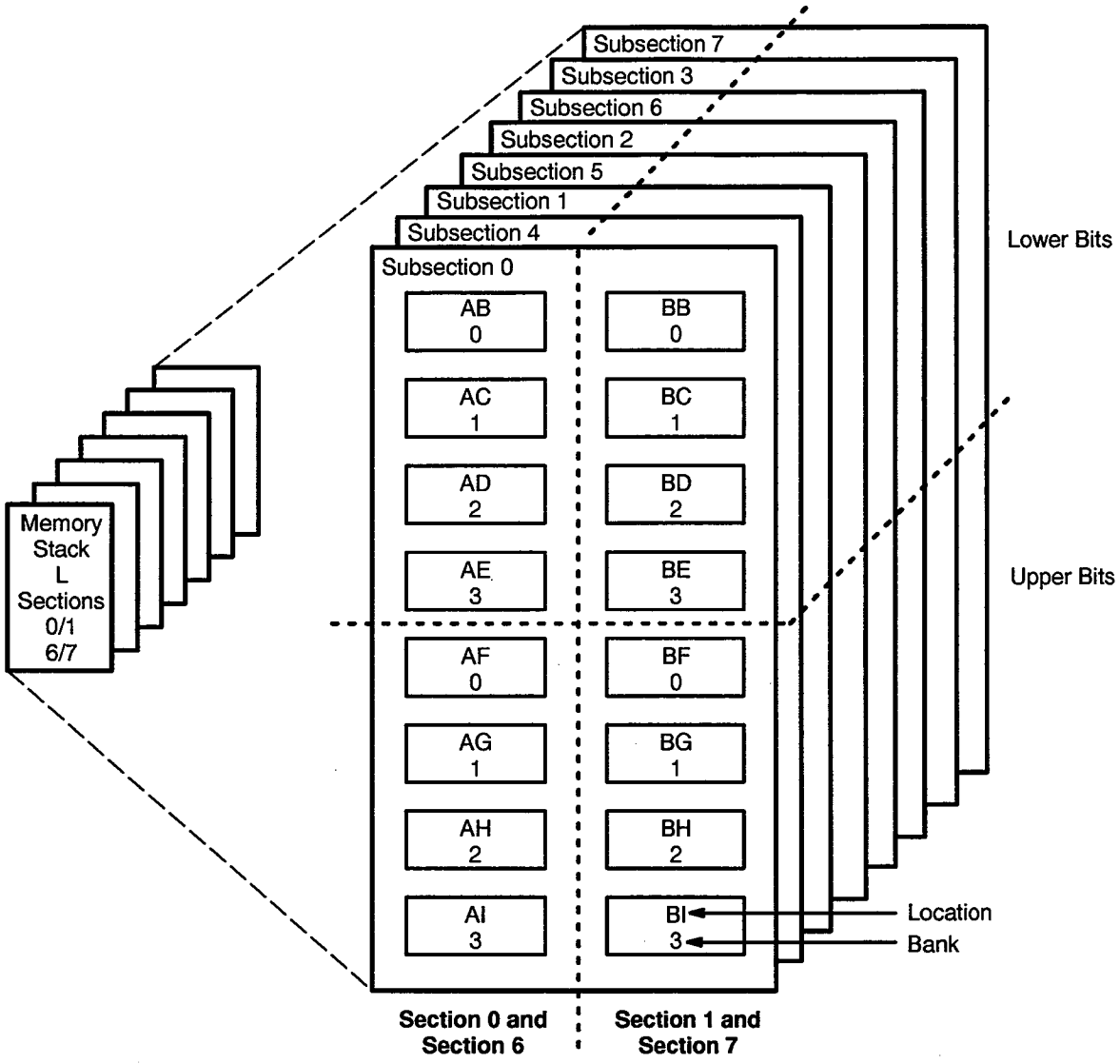


Figure 8. CRAY T916 Memory Module Layout

CRAY T932 Memory

CRAY T932 memory is organized by sections, subsections, and banks. A fully configured system has 8 sections of memory. Each section has 8 subsections and each subsection has 8 banks for a total of 512 banks. Figure 9 shows memory organization in a CRAY T932 system.

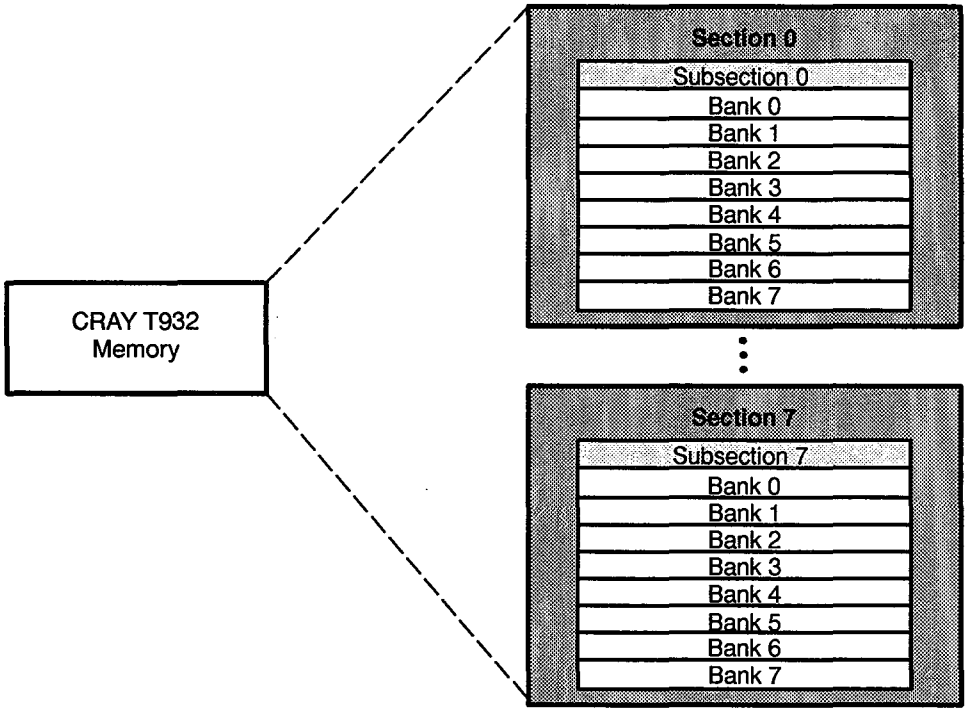


Figure 9. CRAY T932 Memory Organization

The CP, network, and memory modules determine which section, subsection, bank, and word in memory to address. Figure 10 shows the addressing map for CRAY T932 memory and the module type that determines the section select, subsection select, bank select, and word select. Bit 9 is presently used as part of the word select; however, this bit can be used as part of the bank select in future systems to address 16 banks per subsection. Bits 30 through 33 are presently not used; however, these bits may be used in future systems to address up to 16 Gwords of memory.

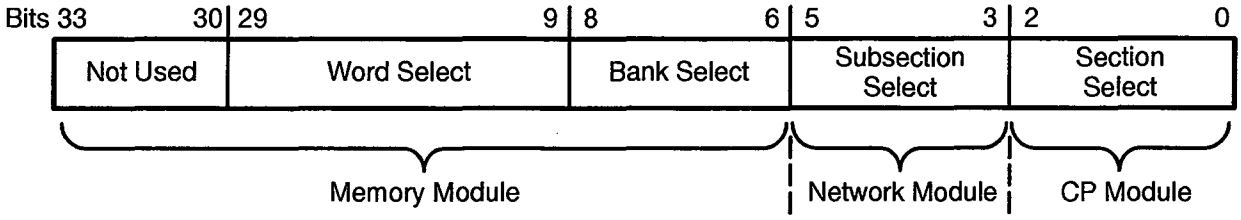


Figure 10. CRAY T932 Addressing

The address bit layout varies for the different CRAY T932 configurations. Table 5 lists the various configurations of CRAY T932 systems and the address bit layout for each of the configurations.

Table 5. CRAY T932 System Configurations

Module Counts			Section Configuration			Address Bit Layout			
Processor	Network	Memory	Sections	Sub-sections	Banks	Section	Sub-section	Bank	Word Address
8	4	32	8	8	512	00, 01, 02	03, 04, 05	06, 07, 08, –	09 – 33
8	4	16	8	4	256	00, 01, 02	–, 03, 04	05, 06, 07, –	08 – 32
8 †	4	8	8	2	128	00, 01, 02	–, –, 03	04, 05, 06, –	07 – 31
8	8	8	8	2	128	00, 01, 02	–, –, 03	04, 05, 06, –	07 – 31
8 †	8	8	8	2	64	00, 01, 02	–, –, 03	04, 05, –, –	06 – 30
8 †	4	4	4	2	64	00, 01, –	–, –, 02	03, 04, 05, –	06 – 30
8 †	4	4	4	2	32	00, 01, –	–, –, 02	03, 04, –, –	05 – 29
8 to 16	8	16	8	4	256	00, 01, 02	–, 03, 04	05, 06, 07, –	08 – 32
8 to 16 †	8	16	8	4	128	00, 01, 02	–, 03, 04	05, 06, –, –	07 – 31
8 to 16 †	8	8	8	2	128	00, 01, 02	–, –, 03	04, 05, 06, –	07 – 31
8 to 16 †	4	8	4	4	128	00, 01, –	–, 02, 03	04, 05, 06, –	07 – 31
16	16	16	8	4	256	00, 01, 02	–, 03, 04	05, 06, 07, –	08 – 32
16 †	16	16	8	4	128	00, 01, 02	–, 03, 04	05, 06, –, –	07 – 31
16 †	16	8	8	2	128	00, 01, 02	–, –, 03	04, 05, 06, –	07 – 31
16 †	8	8	4	4	128	00, 01, –	–, 02, 03	04, 05, 06, –	07 – 31
16 to 32	16	32	8	8	512	00, 01, 02	03, 04, 05	06, 07, 08, –	09 – 33
16 to 32 †	16	32	8	8	256	00, 01, 02	03, 04, 05	06, 07, –, –	08 – 32
16 to 32 †	16	16	8	4	256	00, 01, 02	–, 03, 04	05, 06, 07, –	08 – 32
16 to 32 †	8	16	4	8	256	00, 01, –	02, 03, 04	05, 06, 07, –	08 – 32

† This indicates a degraded or partitioned system; the system is not usually sold with this configuration.

A CRAY T932 system has two or four memory module stacks with each module stack containing either two, four, or eight memory modules. The memory modules connect to the network modules, which connect to the CP modules. The system interconnect board (SIB) connects these memory modules to the network modules and to the CP modules.

Figure 11 shows the logical interconnections between the CP modules and memory modules in a CRAY T932 system.

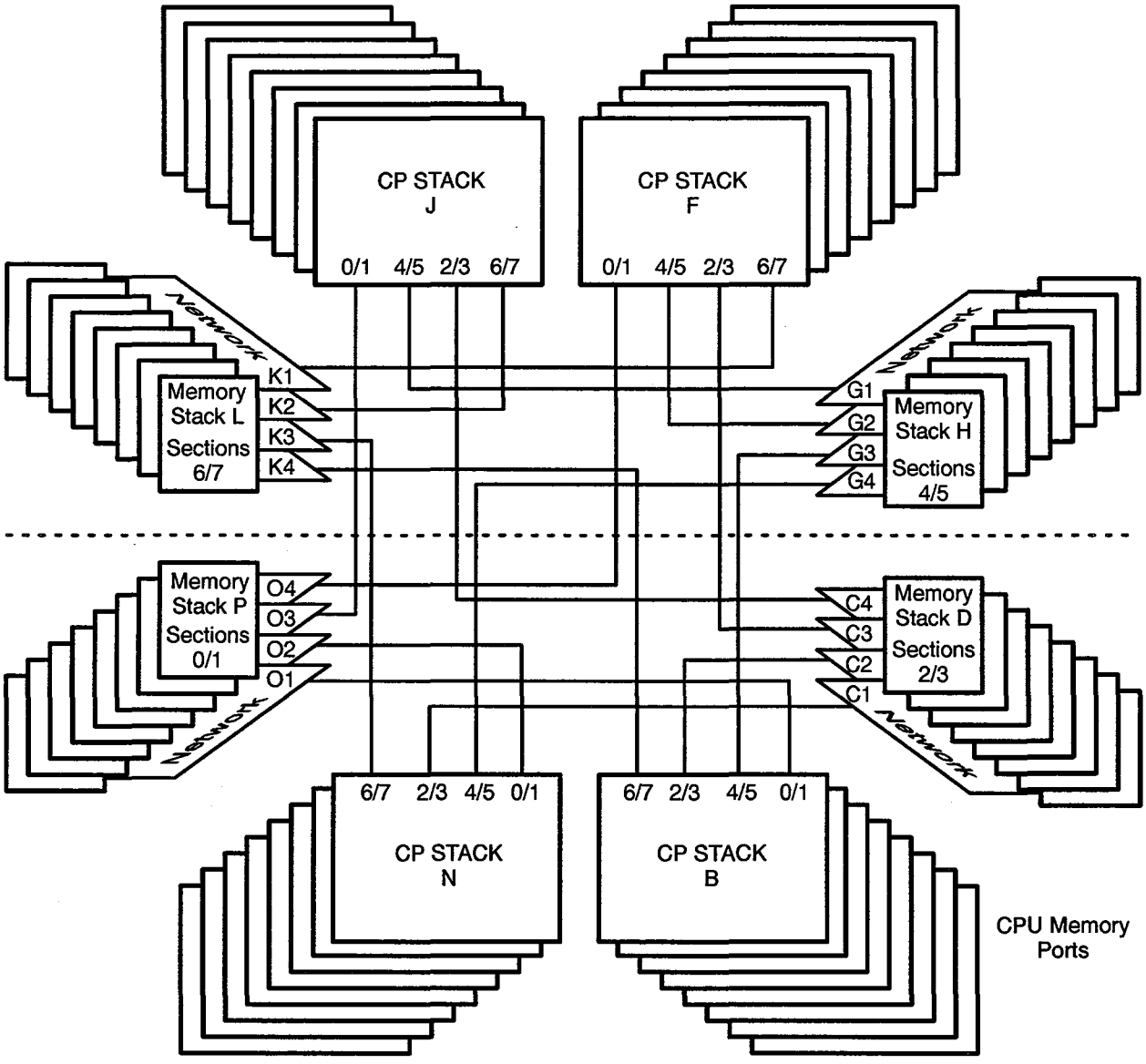


Figure 11. CRAY T932 CPU-to-Memory Interconnections

Each memory module stack handles two sections of memory. Each memory module within the module stack is one subsection. Figure 12 shows the memory module layout for a CRAY T932 system.

Assuming the first reference is to section 0 and the memory references are sequential, references one and two go to sections 0 and 1 subsection 0 at module location P1. References three and four go to sections 2 and 3 subsection 0 at module location D1. References five and six go to sections 4 and 5 subsection 0 at module location H1. References seven and eight go to sections 6 and 7 subsection 0 at module location L1. This sequence continues as a descending spiral through the module stacks and memory subsections until all subsections and banks have been addressed.

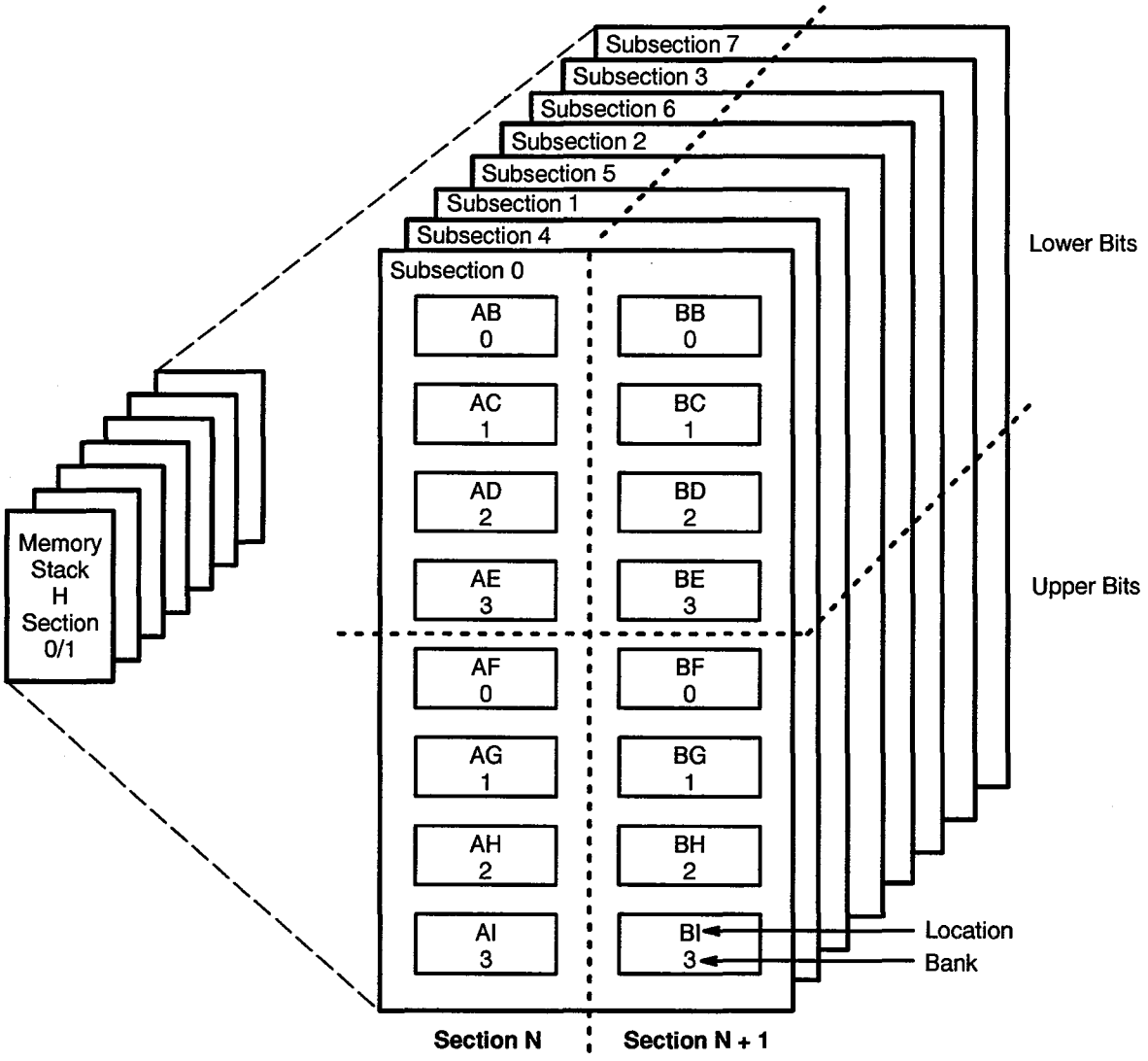


Figure 12. CRAY T932 Memory Module Layout

Memory Module Components

A CM02 memory module consists of two boards: board 1 and board 2. Figure 13 and Figure 14 show the components contained on board 1 and board 2, respectively. The following subsections describe the components on the memory module.

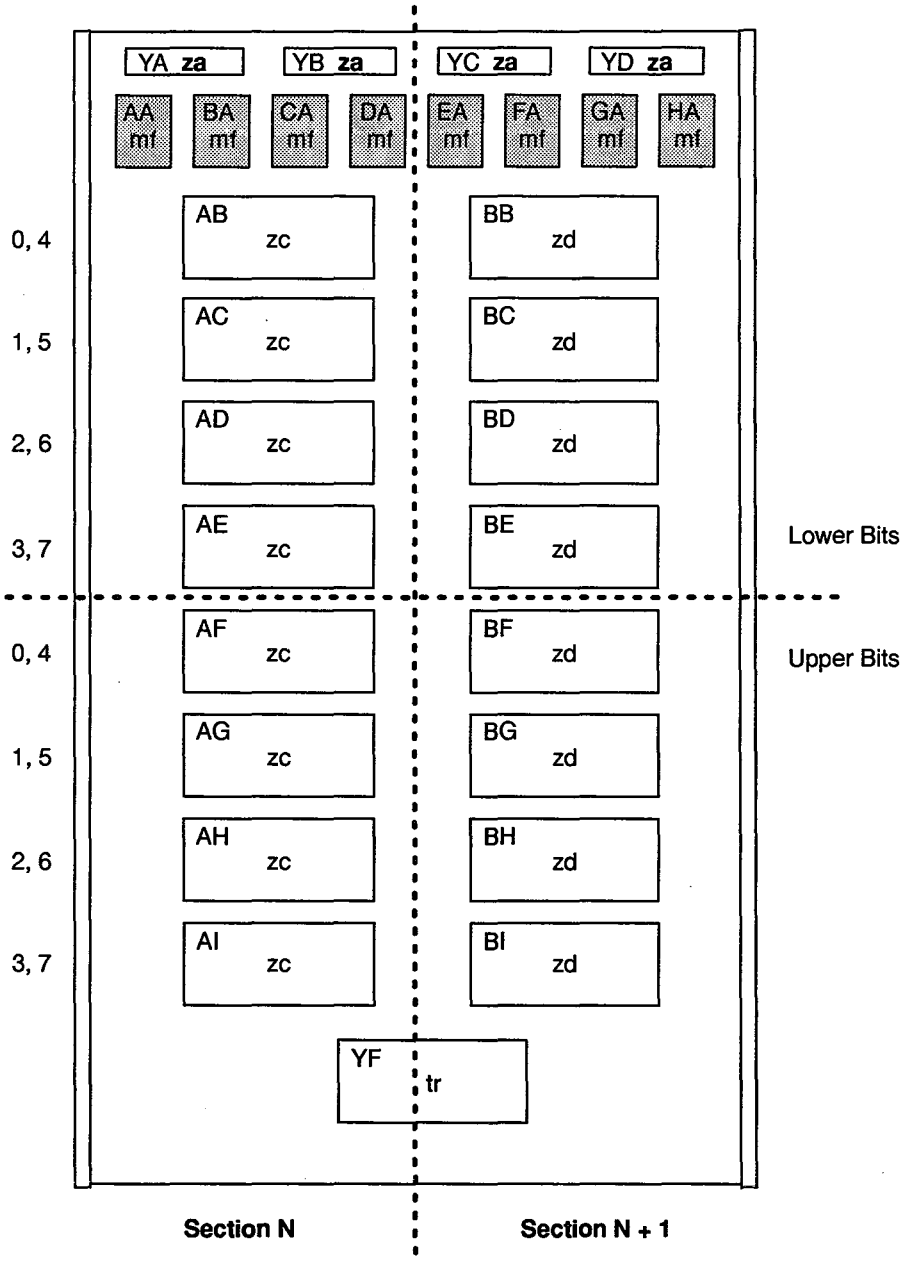


Figure 13. CM02 Memory Module Board 1

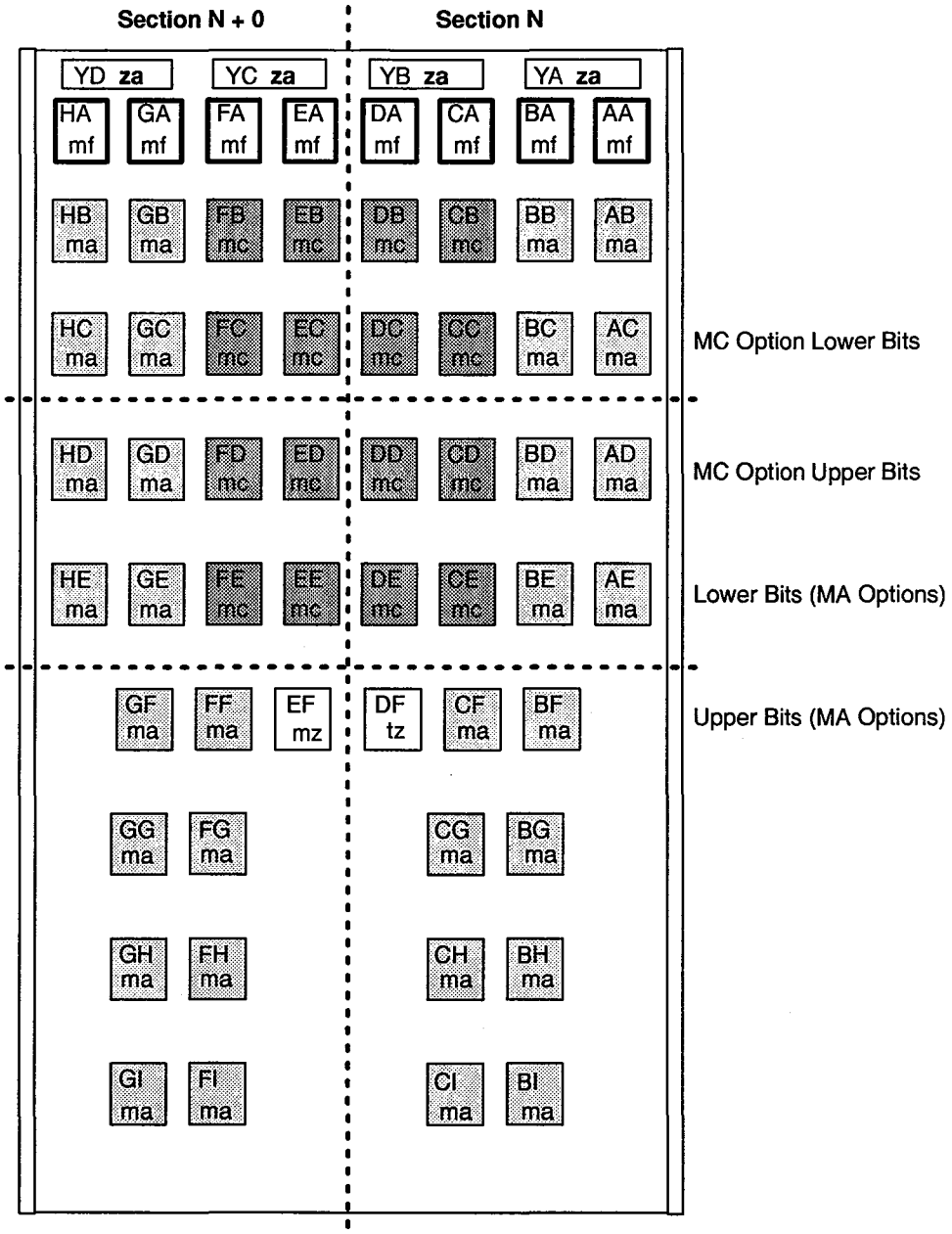


Figure 14. CM02 Memory Module Board 2

Memory Module Connectors

There are eight ZA connectors on each memory module; four connectors on board 1 and four connectors on board 2. The ZA connectors on board 1 handle the memory access paths for processors 4, 5, 6, and 7, and the ZA connectors on board 2 handle the memory access paths for processors 0, 1, 2, and 3. Each ZA connector is associated with one of the eight memory access paths.

Memory Module Options

Table 6 lists the memory module options. In addition to the options listed in the table, the memory module has one MZ and one TZ option. These options are common to all CRAY T90 series modules. The MZ option receives and fans out boundary scan and burnline signals, and the TZ option receives and fans out the system clock.

Table 6. CM02 Memory Module Options

Option Name	Option Quantity	Option Description
MF	16	The MF option is a transmitter and a receiver that can talk to another module 2 to 6 phases away. This option also receives the sanity code bit pattern.
MC	32	The MC option provides a 4 x 4 multiplexer that determines which bank to steer data to or retrieve data from on a write from or read to memory, respectively.
MA	16	The MA option controls memory cycle and memory access time and also controls the spare memory chip selection when a bad memory chip is detected. This option gates data, address, and control information through a four rank FIFO buffer to one of two banks.

Memory Stacks

Each memory module has 16 memory stacks. Each memory stack consists of two printed circuit boards (ST01 A and ST01 B) with 40 2 Mbit x 2 SRAM memory chips between them. The memory chips are arranged in two vertical rows of 20 chips each; 19 chips in each row store data and 1 chip in each row is a spare memory chip.

Figure 15 shows a memory stack and the pin-outs for each memory chip within the memory stack. Memory stack connectors (MSCs) connect memory stacks to the memory module (refer to Figure 15). An MSC is an EZIF connector that requires the application of electric current to insert or remove the memory stack from the module. Refer to the *CRAY T90 Series Field Repair Procedures* document, publication number HMM-xxx-x, for a procedure on replacing a memory stack.

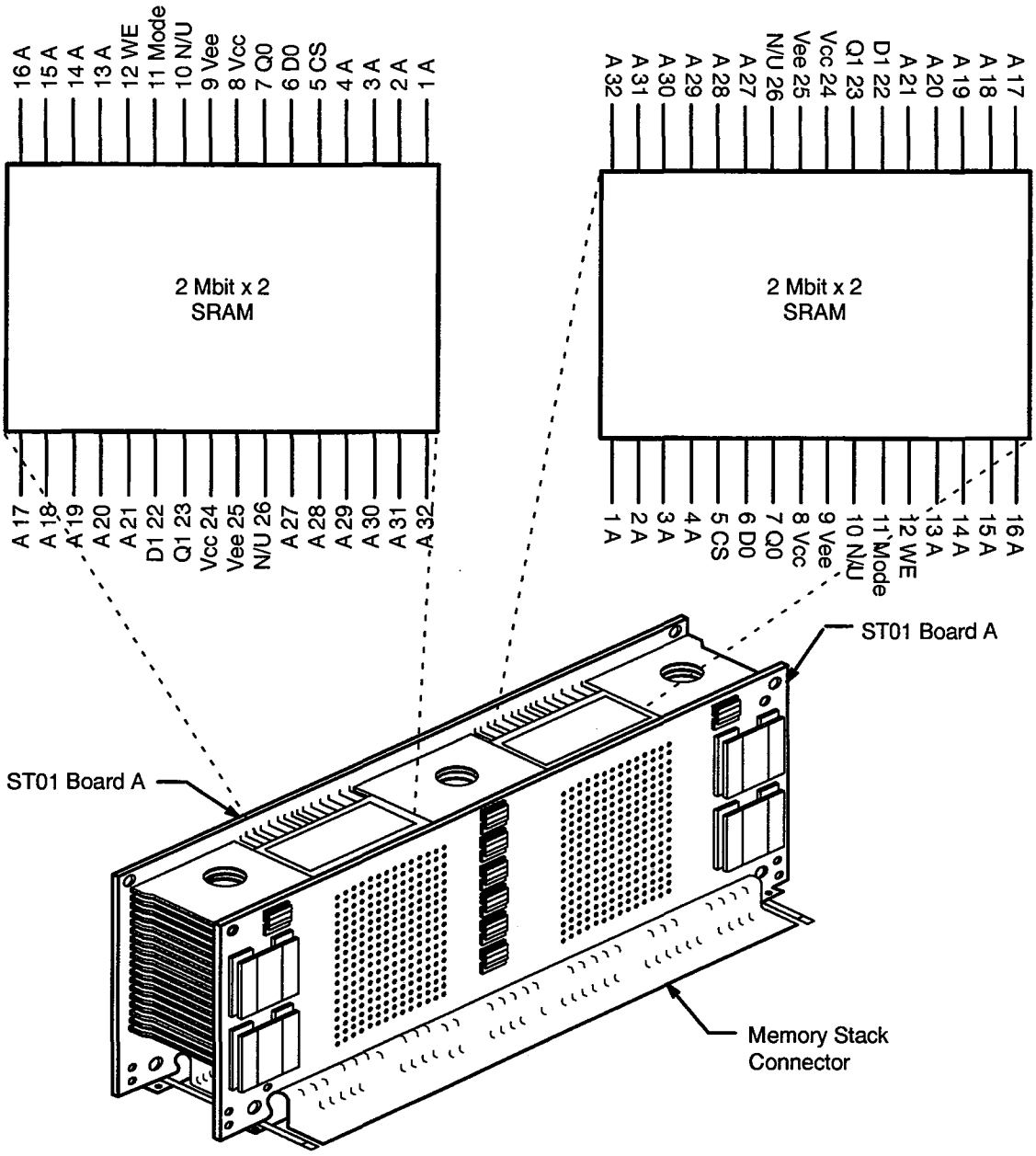


Figure 15. Memory Stack

Figure 16 shows the bit-to-chip layout for the chips within a memory stack. Each memory stack handles either the upper or lower bits of a word for two different banks; each row of chips is one bank. Each memory chip within the rows handles 2 bits of the word. The top chip in each row is a spare memory chip. This spare memory chip can be used in place of a defective memory chip.

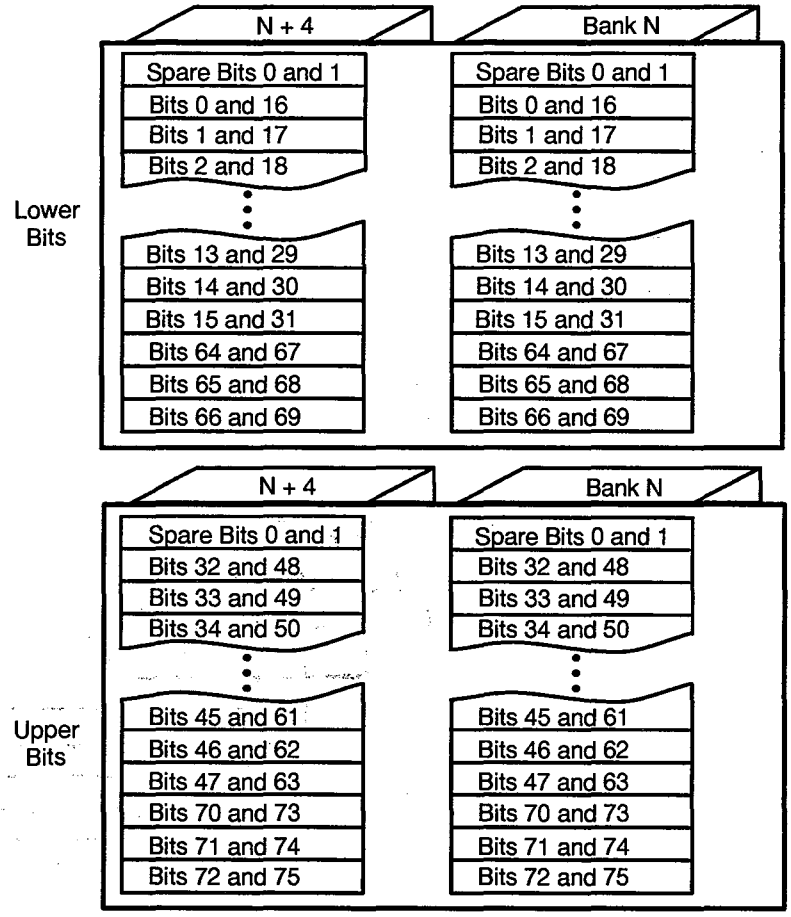


Figure 16. Bit-to-Chip Layout on the Memory Stack

Spare Memory Chip Configuration

Each memory stack has 2 spare memory chips (refer to Figure 16). Using special direct memory access (DMA) commands from the maintenance channel, you can flaw a bad memory chip and use one of the spare memory chips that is located in the same row in the memory stack. Because each memory chip handles 2 bits, both bits on the chip are automatically flawed when a spare memory chip is configured into the system. Refer to the *System Configuration Environment (SCE)* document, publication number HDM-xxx-x, for information on how to configure a spare memory chip.

The MA option configures the data around the bad memory chip. When the system receives the command to flaw a memory chip, the MA option determines which bits to shift around, and the MA option sets a *Special bit* that enables the bits to shift and bypass the flawed memory chip. Figure 17 illustrates the bit shifting between the MA option and the memory stack. In this figure, bits 9 and 25 are in error and are bypassed.

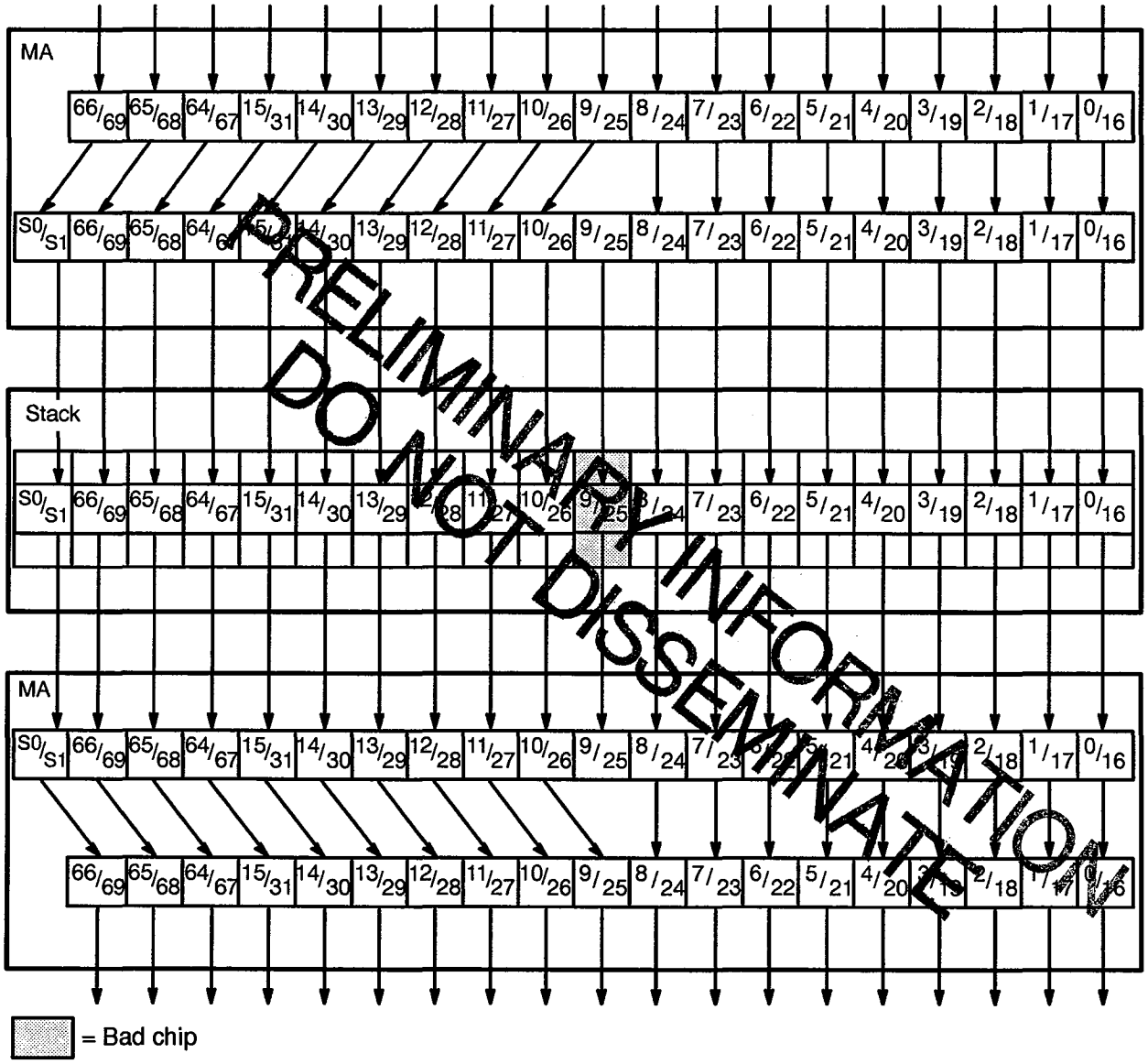


Figure 17. Data Flow through a Memory Stack with Bits 9 and 25 in Error

Error Correction and Detection

CRAY T90 series error correction and detection is designed for "x 2" memory chips where each chip stores 2 bits of each data word. **These 2 bits are called a byte.** The error-correction code is designed to detect and correct byte errors, in which a byte error is an error in either or both bits of the byte. This method of error correction and detection is known as single-byte correction/double-byte detection (SBCDBD). With SBCDBD, single-byte errors can be detected and corrected, double-byte errors can be detected but not corrected, and errors involving 3 or more bytes cannot be reliably detected.

Each memory data word consists of 76 bits: 64 data bits and 12 check bits. During a write operation, a matrix algorithm generates the 12 check bits and stores them with the data bits. The matrix algorithm groups the data into six groups that are consistent with the bit-to-chip layout illustrated in Figure 18. The memory stack groups data this way so that any single data-path error between a CPU and memory stack can be corrected as if it were a memory error.

During a read operation, the check bits are regenerated and compared to the original check bits. The result of this comparison is called a *syndrome*. This syndrome can be broken down into 6 (2-bit) bytes.

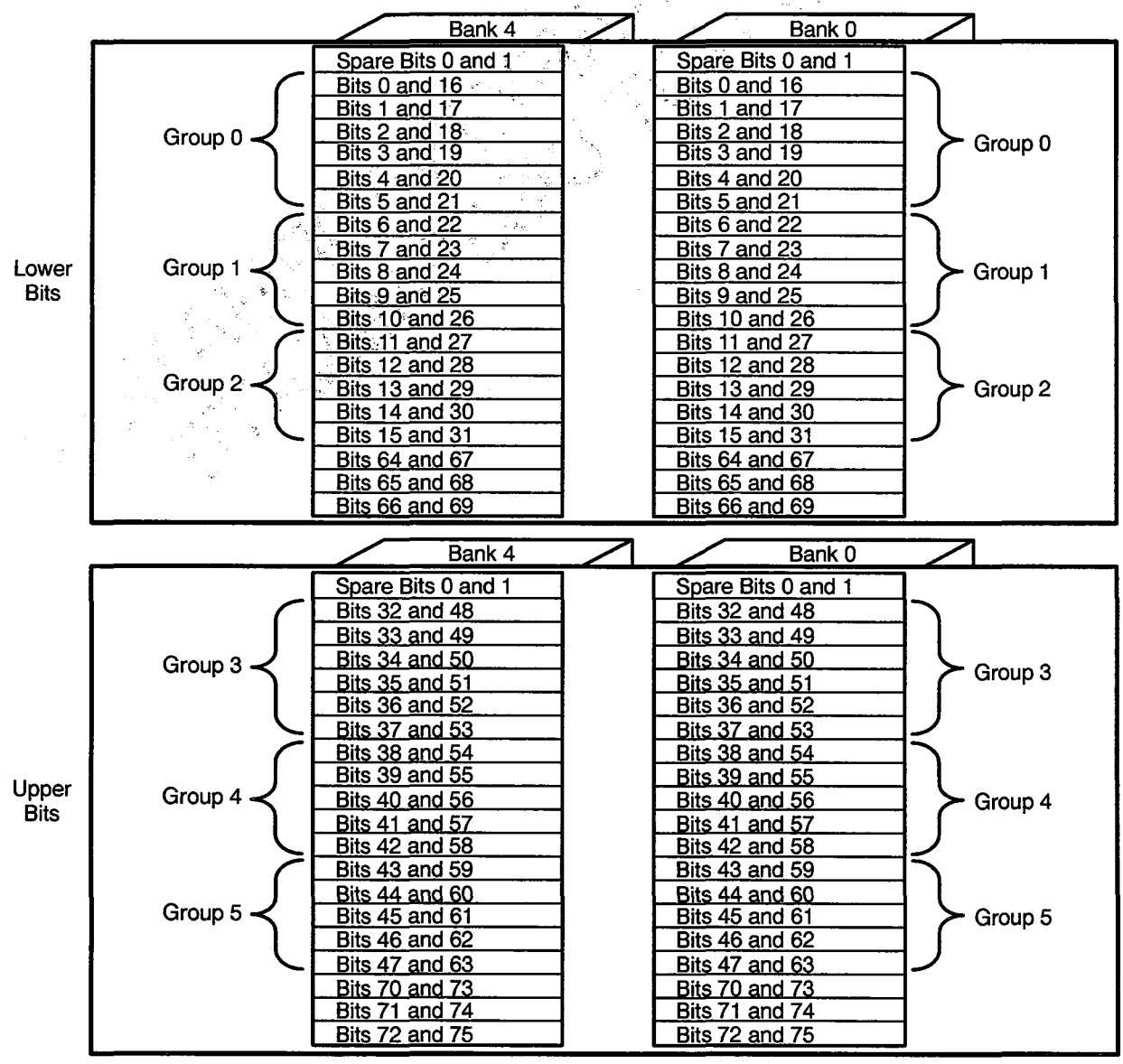


Figure 18. Data Bit Groups for SBCDBD

Syndrome Decoding

When a CPU receives an error, it decodes the syndrome to correct the failing byte or to flag the double-byte error. Table 7 and Table 8 list the different syndromes and the bits associated with each syndrome. If a syndrome is not listed in the table, then it is a multiple- or double-byte error and cannot be corrected.

Table 7. Syndrome Chart for 1 Bit in Error

Syndrome	Bit	Syndrome	Bit	Syndrome	Bit	Syndrome	Bit	Syndrome	Bit
0001	64	0501	32	2021	43	3321	47	5170	57
0002	67	0504	33	2024	6	3612	20	5200	55
0004	65	0520	34	2120	11	3621	46	5212	19
0010	68	0525	35	2125	13	4000	75	5242	61
0020	66	0555	37	2133	15	4012	18	5250	56
0025	7	0571	36	2136	14	4042	59	5330	58
0040	69	1000	74	2401	44	4050	22	5505	5
0052	23	1012	17	2420	38	4240	27	5542	63
0100	70	1202	48	2425	8	4247	30	6426	10
0105	0	1210	49	2500	39	4252	29	6612	21
0124	12	1236	52	2505	3	4255	31	7051	25
0200	73	1240	50	2521	45	4427	9	7105	4
0212	16	1252	51	2524	40	4742	62	--	--
0250	28	1266	53	2664	42	5002	60	--	--
0400	71	2000	72	2744	41	5040	54	--	--
0405	1	2005	2	3053	26	5052	24	--	--

Table 8. Syndrome Chart for 2 Bits or 1 Byte in Error

Syndrome	Bit	Syndrome	Bit	Syndrome	Bit	Syndrome	Bit	Syndrome	Bit
0003	64, 67	1417	1, 17	3476	9, 25	6366	15, 31	7634	41, 57
0014	65, 68	1703	32, 48	4717	4, 20	6371	14, 30	7700	39, 55
0060	66, 69	1714	33, 49	5475	10, 26	6663	47, 63	7717	3, 19
0077	7, 23	1733	37, 53	6000	72, 75	7163	46, 62	7763	45, 61
0300	70, 73	1747	36, 52	6017	2, 18	7403	44, 60	7774	40, 56
0317	0, 16	1760	34, 50	6063	43, 59	7460	38, 54	6377	13, 29
0374	12, 28	1777	35, 51	6074	6, 22	7477	8, 24	--	--
1400	71, 74	3317	5, 21	6360	11, 27	7554	42, 58	--	--

Special rules govern the decoding of a syndrome. Table 9 lists these rules and provides an example and description of the syndrome byte error.

Table 9. Syndrome Decoding Rules

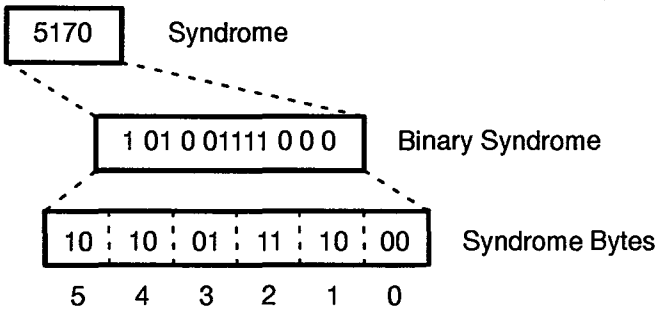
Syndrome Rule	Example: Syndrome Bytes †						Error Description
	5	4	3	2	1	0	
Syndrome = 0	0	0	0	0	0	0	No error occurred.
Even number of bytes in error	00	00	01	00	01	00	Double- or multiple-byte error.
1 byte is in error	00	01	00	00	00	00	Checkbyte error. The byte indicates which check bits are in error.
3 or 5 bytes in error	00	01	01	00	00	01	Single-byte error in data field.

† 00 = Good Byte
01, 10, 11 = Bad Byte

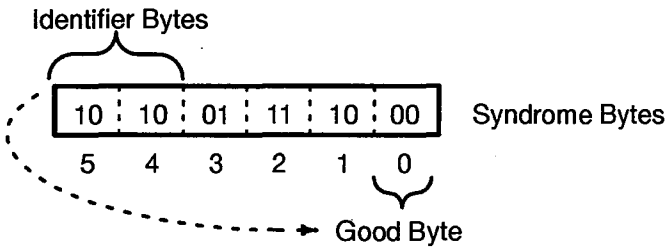
Example: Decoding a Syndrome

To decode the syndrome, the system views it as 6 bytes numbered 0 to 5 from right to left. If both bits in a particular byte of a syndrome are 0's, that syndrome byte is good. If either bit in the syndrome byte is 1, the syndrome byte is bad. Use the following sequence to understand how the system decodes the syndrome; this sequence contains the example of decoding syndrome 5170.

1. Decode the syndrome.



2. Find 2 **identical** bytes in a bad byte, bad byte, good byte pattern. Begin with byte 0 and work right to left in a counterclockwise direction. The 2 bad bytes are called identifier bytes.

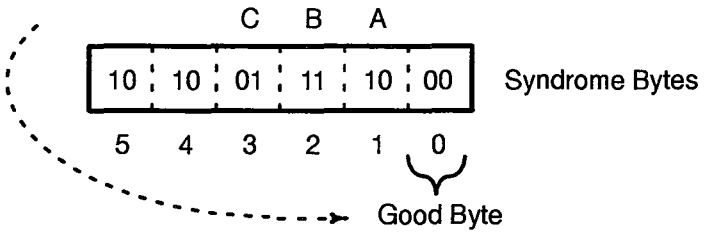


3. Determine the group in error by using Table 10. In the example, the identifier bytes (bad byte, bad byte) are bytes 4 and 5, and the good byte is byte 0. Therefore, the group in error is group 4.

Table 10. Group in Error

Bad Bytes	Good Bytes	Group in Error
0 and 1	2	0
1 and 2	3	1
2 and 3	4	2
3 and 4	5	3
4 and 5	0	4
5 and 0	1	5

4. Assign letters A, B, and C to the remaining 3 bytes. Assign letter A to the first byte to the left of the good byte and continue the assignments from right to left in a counterclockwise direction.



5. Use one of the following conditions to determine the failing data byte:

   - If exactly one of bytes A, B, or C is in error and has the same value as the identifier bytes, use Table 11 to determine the failing data byte. If the bad byte does not match the identifier byte, a multiple-byte error occurred and cannot be corrected.

   - If bytes A, B, and C are all bad bytes, byte A matches the identifier byte, and all 3 bytes (A, B, and C) are different, use Table 12 to determine the failing data byte. If 2 bytes are the same and 1 byte is different, a multiple-byte error occurred and cannot be corrected.

Table 11. Failing Data Byte (3 Bytes in Error)

Bad Byte	Failing Data Byte					
	Group 5	Group 4	Group 3	Group 2	Group 1	Group 0
Byte A	27		16	11		0
Byte B		22	17		6	1
Byte C	28	23	18	12	7	2

Table 12. Failing Data Byte (5 Bytes in Error)

Bad Byte			Failing Data Byte					
C	B	A	Group 5	Group 4	Group 3	Group 2	Group 1	Group 0
01	01	01	29	24	19	13	8	3
10	10	10						
11	11	11						
11	10	01	30	25	20	14	9	4
01	11	10						
10	01	11						
10	11	01	31	26	21	15	10	5
11	01	10						
01	10	11						

6. Use Table 13 to determine which bit in the byte is in error. The example shows that bit 57 is causing the error. The following conditions of the identifier byte bad bit(s) determine the bad bit(s):

Identifier Byte	Bit in Error (Table 13)
01	First bit in the byte
10	Second bit in the byte
11	Both bits in the byte

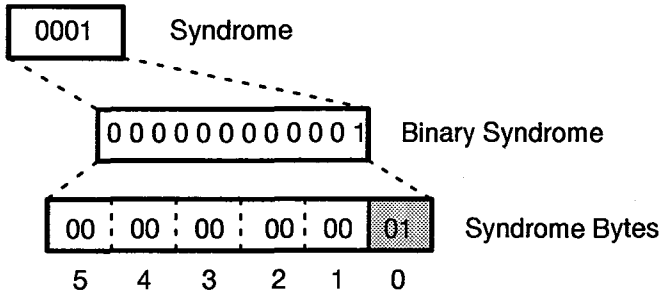
Table 13. Bits in the Byte

Byte Number	Bits in Byte		Byte Number	Bits in Byte		Byte Number	Bits in Byte		Byte Number	Bits in Byte	
	1st Bit	2nd Bit		1st Bit	2nd Bit		1st Bit	2nd Bit		1st Bit	2nd Bit
000	00	16	009	09	25	008	08	24	024	40	56
001	01	17	010	10	26	017	33	49	025	41	57
002	02	18	011	11	27	018	34	50	026	42	58
003	03	19	012	12	28	019	35	51	027	43	59
004	04	20	013	13	29	020	36	52	028	44	60
005	05	21	014	14	30	021	37	53	029	45	61
006	06	22	015	15	31	022	38	54	030	46	62
007	07	23	016	32	48	023	39	55	031	47	63

Example: Decoding a Failing Check Bit

If only 1 byte of a syndrome is in error, the failing bit(s) are check bit(s). The following example shows how to decode a syndrome for a failing check bit.

1. Decode the syndrome.



2. Use Table 14 to determine the single bad byte and to locate the failing check bit(s). In this example, the first bit in byte 0 is failing, which indicates that the failing check bit is 64.

Table 14. Failing Bit in a Checkbyte

Failing Byte	1st Bit	2nd Bit
0	64	67
1	65	68
2	66	69
3	70	73
4	71	74
5	72	75

Memory Overview

The communication protocol, or *handshaking*, that occurs between a CP module and CM module and between the options on the CM module uses **Valid** and **Resume** signals. Each transmitting option contains a counter. Each receiving option contains buffers. The transmitting option can send as many valid signals to the receiving option as there are buffers in that option. With each **Valid** signal sent, the counter in the transmitting option increments by one. When the reference advances to the next stage, the receiving option sends a **Resume** signal to the transmitting option. The counter in the transmitting option then decrements by one. Figure 19 illustrates the handshaking that occurs between options.

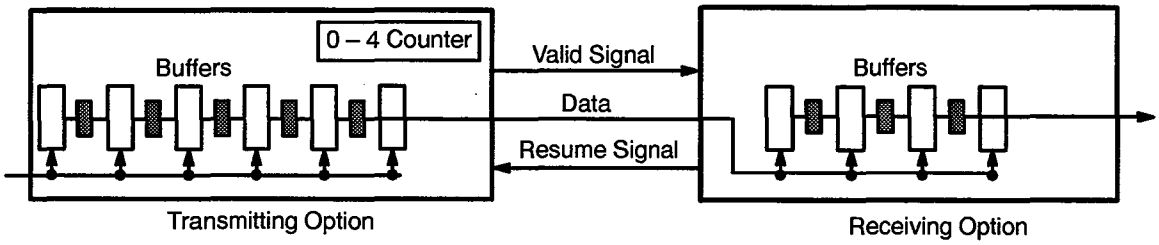


Figure 19. Handshaking between Module Options

CP-to-Memory Module Communications

Figure 20 shows communications between a CP and memory module during a memory reference. When a CPU references memory, it sends valid signals to memory followed by control signals and then address and data. Control information includes steering bits, a Write Reference signal, and a Reference Code. The steering bits direct data to the appropriate bank, the Write Reference signal tells memory whether the reference is a read or a write reference, and the Reference Code notifies memory of the type of reference coming from the CPU. During a read reference, the CPU also sends a Destination Code that gets buried into the write data field. This Destination Code is sent back to the CPU with the data to ensure that the reference gets to the appropriate place.

During a write operation, the CPU transfers data to memory in two data packets. Each data packet is 38 bits wide; two packets equal 1 data word. The data packets are sent in 2 clock periods (CPs). The first data packet is sent in CP 1, and the second data packet is sent in CP 2. During a read operation, the full 76-bit word is transferred to the CPU in 1 CP.

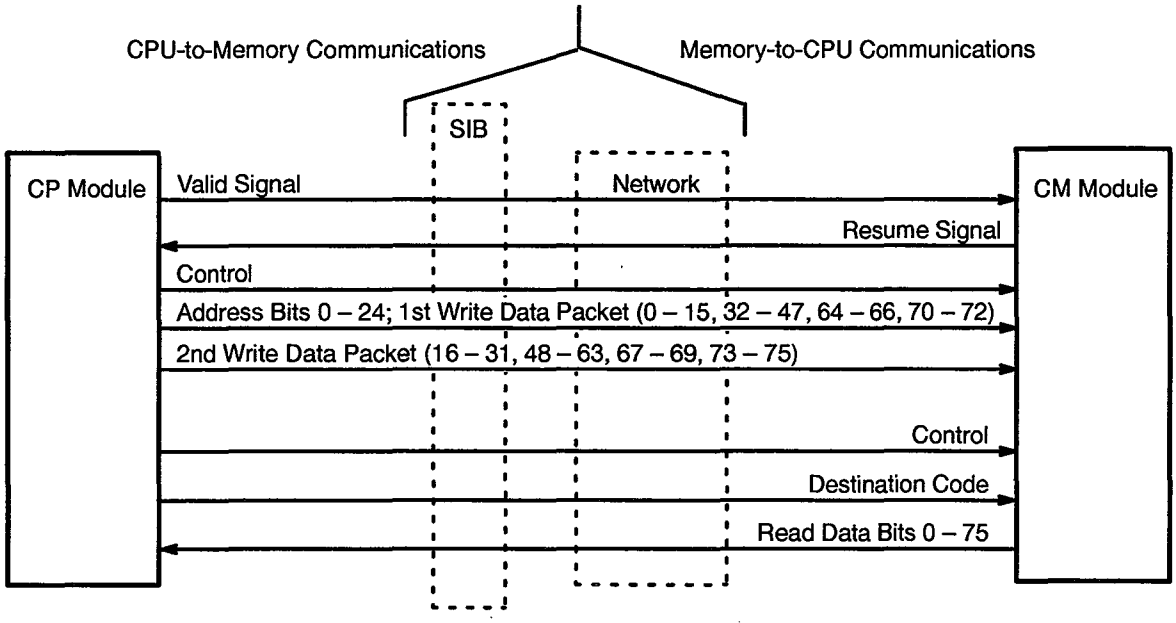


Figure 20. CPU-to-Memory Communications

Memory Module Operations

Control, address, and data pass through one of eight ZA connectors to arrive on the memory module. Each ZA connector is associated with one CPU access path. During a memory write operation, the ZA connectors distribute control, address, and data to the MF options. The MF options buffer the control, address, and data and send it to the MC options, which multiplex it and select its bank destination. The MA options steer the control, address, and data to the bank specified by the MC option.

On a read from memory, the ZA connectors distribute the control and address to the MF options. The MF options direct the address to the MC options that select the bank to address. The MA options steer the address to the bank specified by the MC option. Figure 28 is a block diagram of the memory module that shows the path through the options on the memory module. The following subsections explain the function of each option.

MF Option

The MF options are located next to the ZA connectors (refer to Figure 13 and Figure 14); two MF options are associated with one ZA connector and are grouped by processor path, two MF options per processor path. Each option handles 32 bits of the data word, 6 of the check bits, and one-half of the address bits, steering bits, and Reference Code. The

even-numbered option handles the upper bits (32 – 63 and check bits 70 – 75), and the odd-numbered option handles the lower bits (00 – 31 and check bits 64 – 69). Each MF option relays the data to 4 MC options during a memory write operation and outputs the data to a ZA connector during a memory read operation.

Figure 21 is a block diagram of the MF option. The MF option provides a six-buffer relay between the CPU and memory and between memory and the CPU. Memory references arrive on the memory module from the processor and are latched in the MF options in a first-in-first-out order. The reference is advanced through the six buffers by a **Valid** signal from the CPU (CI option) and a **Resume** signal from the MF option. Memory references can be made as long as a buffer is available.

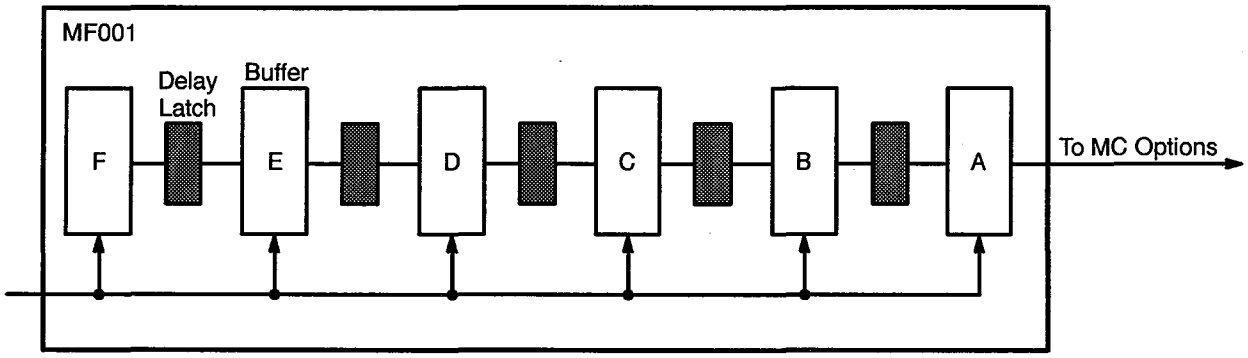


Figure 21. MF Option Block Diagram

MC Option

Each memory module contains 16 MC options. The MC options handle bank and processor access. The priority is handled on a first-come, first-serve basis. When a reference gains access to a bank or processor, that bank or processor gets last priority the next time it is accessed. The MC options have four input holding buffers that allow them to stack four memory references if the output request buffers are busy. The MC options receive 3 bank bits. Two of the bits determine which bank pair to address. The MC option sends the third bit to the MA options, which use it to determine the bank to address.

During a write reference, each MC option receives control, address, and data from four MF options, performs a 4 x 4 multiplex of the control, address, and data, and then sends it to four MA options. The MC options receive address and data in two packets; the first packet contains the lower data bits, and the second packet contains the upper data bits.

During a read reference, the MC option does a 4 x 4 multiplex of the data and Destination Code. This option also builds the bank address to return the data to the processor and builds the return path that designates which memory port was active.

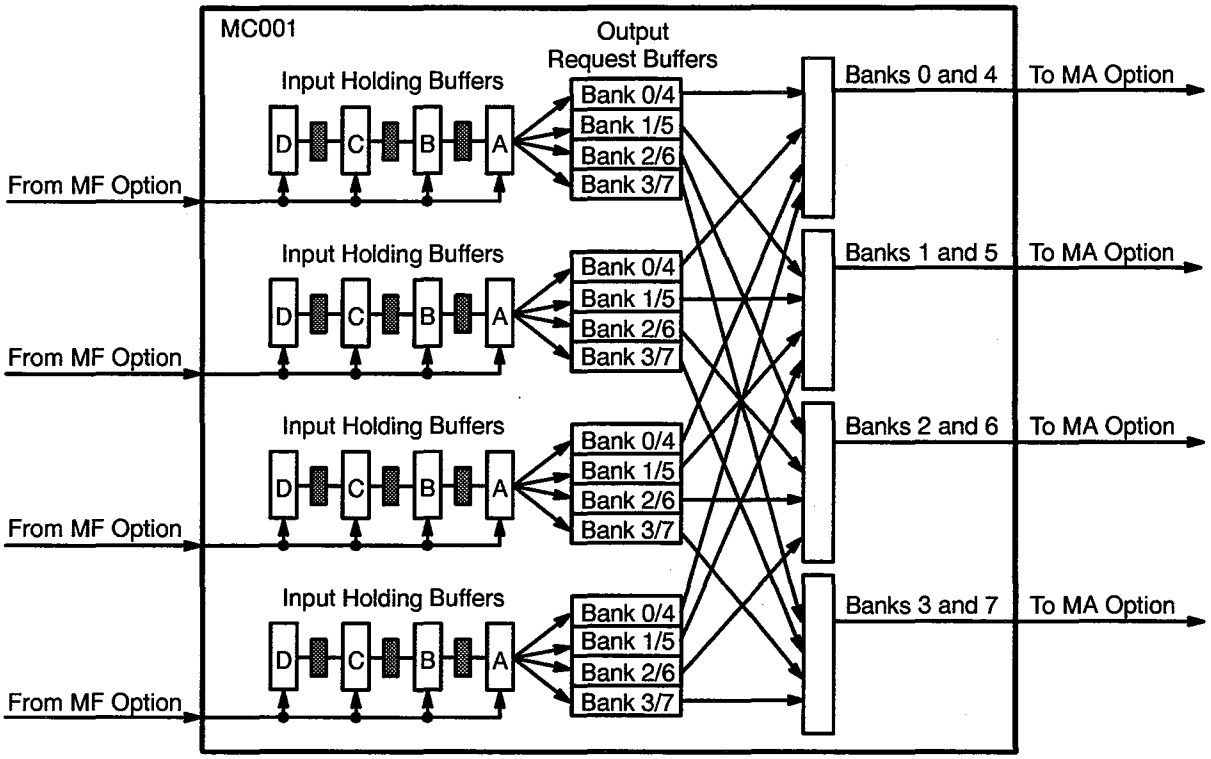


Figure 22. MC Option Block Diagram

MA Option

This option provides all the control signals necessary to access 2 banks of memory. It also provides a spare chip selection feature that allows the maintenance port to configure around a bad memory chip.

The MA option also receives 6 bits to control the memory access and memory cycle time. Access time can vary from 5 to 12 clock periods (CPs) and cycle time can be 2, 4, 6, 8, 10, or 11 CPs. This feature will be used with future memory chips to enable a faster access time and to avoid redesign of the module.

On a write reference, control, address, and data are gated into a 4-rank first-in-first-out buffer and then gated to one of two banks. Because the MA options work in pairs to access a bank pair, only 1 bank bit is required to steer the data to banks 0 through 3 or 4 through 7.

On a read reference, the MA options receive the control signals (including a Destination Code) and 20 data bits from the memory stack and gate them into an 8-rank first-in-first-out buffer. The control and data bits leave the MA options and enter the MC options that determine to which processor to send the bits.

The MA option also configures data around a bad memory chip. Because each memory chip is assigned 2 bits, the configuration automatically flaws both bits. Refer to "Spare Memory Chip Configuration" on page 24 for more information.

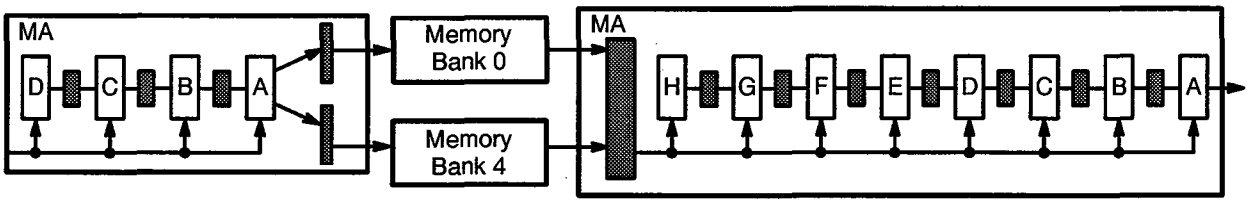


Figure 23. MA Option Block Diagram

Address Distribution

CRAY T90 series systems can use up to 35 address bits to determine which section, subsection, bank, and word in memory to address; however, not all address bits are used at this time (refer to "CRAY T94 Memory," "CRAY T916 Memory," and "CRAY T932 Memory" on pages 9, 12, and 16 respectively, for information on addressing).

The CP module examines the lower 3 address bits (0 through 2) to determine which memory section receives the data and address. The remaining address bits enter the network module through a connector. The network module examines bits 3 through 5 to determine which subsection receives the address and data (for CRAY T916 and CRAY T932 systems only). The remaining address bits enter the memory module that determines which bank and word receives the address and data.

As address bits are distributed through the CP, network, and memory modules, the modules strip off some of the address bits once they determine the section, subsection, bank, or word to address; then, some of the bits are replaced with Return Path bits that are used to steer the reference back to the CPU module that sent the reference. Figure 24 shows the distribution of address bits through the CP, network, and memory modules. It also illustrates which address bits are stripped off as they pass through the different modules.

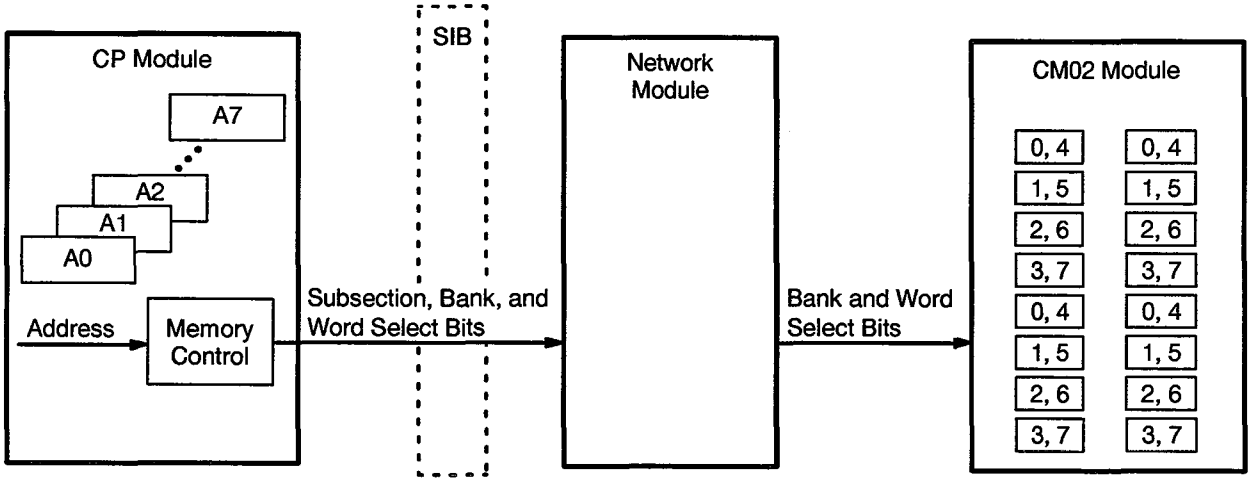


Figure 24. Address Distribution

When address bits arrive on the memory module, the odd-numbered MF options receive address bits 0 through 11 and the even-numbered MF options receive address bits 12 through 24. Figure 25 is a block diagram that shows the flow of address bits through the memory module. As the address bits flow through each option, the options strip off the bits that are no longer needed to determine which word in memory to address. The MA option requires 21 address bits to determine the word select.

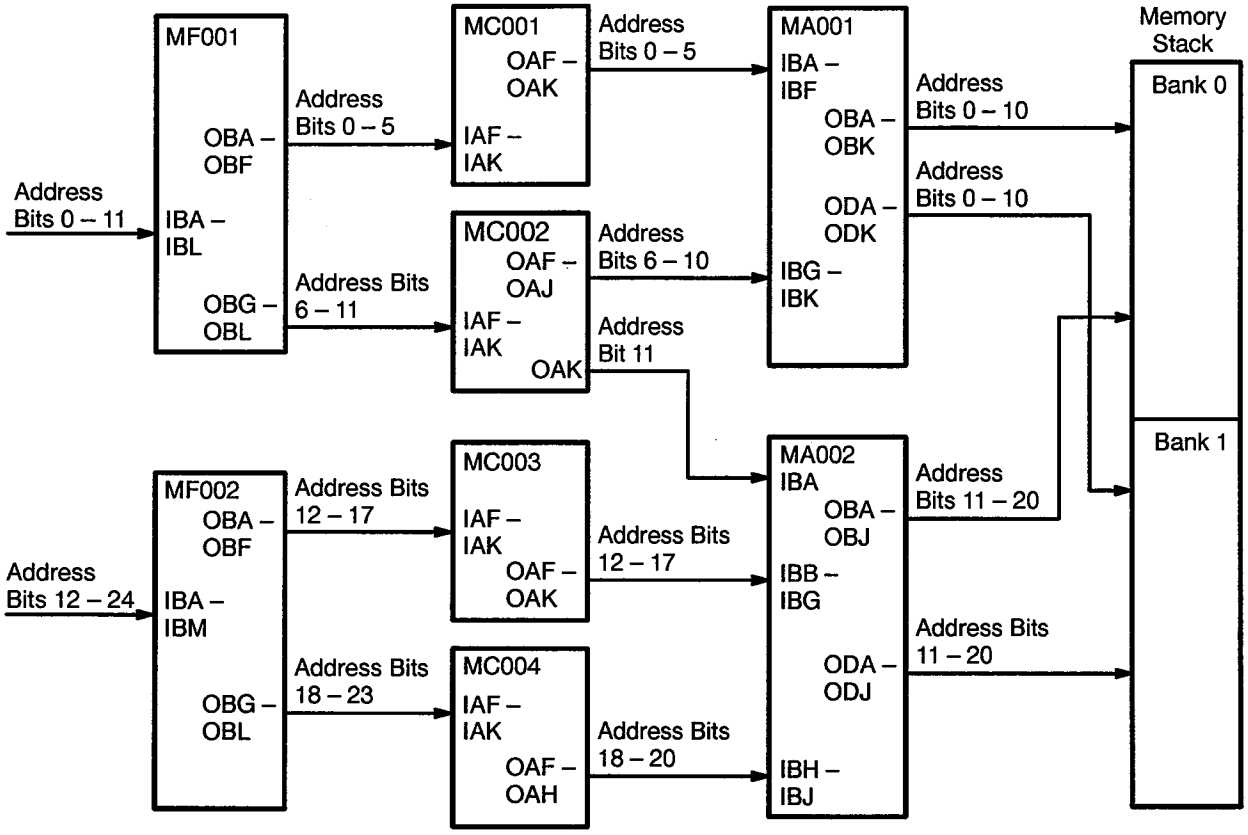


Figure 25. Address Distribution Block Diagram (Memory Module)

Memory Write Operation: A Block Diagram Description

Refer to the block diagram, Figure 29, when reading the following information. When a CPU writes data into memory, it sends a Valid signal followed by control signals and then address bits and data. A data word is 76 bits: 64 data bits and 12 error-correction bits. The write reference arrives on the memory module through one of eight ZA connectors that are associated with a CPU access path. The ZA connectors pass the reference to the MF options, which pass the data to the MC and then to the MA options before they arrive on the memory stack.

Write Control Signals

Figure 29 shows the control signals associated with a memory write operation; however, the figure illustrates only the options associated with a write operation for the lower data bits being sent from processor 0. Two MF options, eight MC options, and four MA options are required to send a complete data word to memory.

A write reference uses three control signals: steering bits, a **Write Reference** signal, and **Reference Type** bits. The steering bits include **Bank Bits 2, 0, 3,** and **1**, and **Subsection Bits 1, 0, 2,** and **3**. The bank bits are used by the MC and MA options to steer the address and data to the appropriate bank in memory. The subsection bits pass through only the memory module options; these bits were used in the network module to determine which memory section to address.

The **Write Reference** signal informs each memory module option that the reference is a write reference. This signal is needed because data is written to memory in packet form, with each option receiving two data packets. The first data packet contains the lower data bits, and the second data packet contains the upper data bits. If the **Write Reference** signal is equal to 1, the reference is a write reference and the memory module options can expect to receive two data packets. If the **Write Reference** signal is equal to 0, then the reference is a read reference and the options can expect to receive one data packet containing all 76 bits.

The **Reference Type** bits are 4 bits used by the MA options to notify memory of the type of reference coming from the CPU. Table 15 lists the terms associated with the **Reference Type** bits and decodes the bits by CPU function.

Write reference control signals pass through the MF options and enter the MC options. Table 16 lists the control and data bits that leave each MF option. Two MF options are needed to direct all the control signals from one processor to memory.

Table 15. MA Option Reference Type Bit Decode

Function	ICD	ICC	ICB	ICA
Abort	0	0	0	0
CPU or I/O read	0	0	0	1
CPU write	0	0	1	0
Not valid function	0	0	1	1
Set bad bit code	0	1	0	0
Not valid function	0	1	0	1
Read bad bit code	0	1	1	0
Not valid function	0	1	1	1
I/O write	1	X	X	X

Note: X = either a 1 or a 0.

Table 16. MF Option Bit Assignments (Write)

Processor 0	
MF001	MF 002
Processor 1	
MF 003	MF 004
Processor 2	
MF 005	MF 006
Processor 3	
MF 007	MF 008
Processor 4	
MF 009	MF 010
Processor 5	
MF 011	MF 012
Processor 6	
MF 013	MF 014
Processor 7	
MF 015	MF 016
Data 0 – 31, 64 – 69	Data 32 – 63, 70 – 75
Address bits 0 – 11	Address bits 12 – 23
Reference type 0 – 01	Reference type 02 – 03
Subsection bits 0 – 01	Subsection bits 02 – 03
Bank bits 00, 02	Bank bits 01, 03
Write reference bit for MC000 – MC004	Write reference bit for MC005 – MC008
Valid signal for MC001 – MC004	Valid signal for MC005 – MC008

The MC options use the Bank Bits (**Bank Bit 0, 1, 2**) to steer the address and data to the appropriate bank in memory. The MC options need only **Bank Bit 2** during a memory write operation because this bit determines which of two bank groups receives the address and data: the upper four banks or the lower four banks. Once the MC options determine which bank to address, they drop the bank bits and create **Return Path** bits 0 and 1. The **Return Path** is a 2-bit code that is forced to either a 0 or a 1 to designate which processor the reference came from. The MA options use the **Return Path** bits to notify the processor that the reference arrived on the memory module.

The other write reference control signals pass through the MC options and enter the MA options. Table 17 lists the control, address, and data bits that leave each MC option. Eight MC options handle the control, address, and data for four processors. Options MC001 through MC008 handle control, address, and data from processors 0 through 3; options MC009 through MC016 handle control, address, and data from processors 4 through 7.

Table 17. MC Option Bit Assignments (Write)

Write Reference	Lower Bits				Upper Bits			
Processors 0 – 3	MC001	MC002	MC003	MC004	MC005	MC006	MC007	MC008
Processors 4 – 7	MC009	MC010	MC011	MC012	MC013	MC014	MC015	MC016
1st data packet 2nd data packet	00 – 04 16 – 20	05 – 09 21 – 25	10 – 14 26 – 30	15, 64 – 66 31, 67 – 69	32 – 36 48 – 52	37 – 41 53 – 57	42 – 46 58 – 62	47, 70 – 72 63, 73 – 75
Address bit	00 – 05	06 – 11	12 – 17	18 – 20	00 – 05	06 – 11	12 – 17	18 – 20
Ref type	00 – 01	02 – 03	00 – 01	02 – 03	00 – 01	02 – 03	00 – 01	02 – 03
Subsection bit	00		01				03	
Bank		02		02		02		02
Return bit	00		01		00		01	
Write ref bit	x		x		x		x	

The 16 MA options handle control, address, and data for four processors. Options MA001 through MA016 handle control, address, and data for processors 0 through 3; options MA017 through MA032 handle control, address, and data for processors 4 through 7. Table 18 lists the control, address, and data received by each MA option.

Table 18. MA Option Bit Assignments (Write)

	Processor 0 – 3				Processor 4 – 7			
Bank 0/4	MA001	MA002	MA009	MA010	MA017	MA018	MA025	MA026
Bank 1/5	MA003	MA004	MA011	MA012	MA019	MA020	MA027	MA028
Bank 2/6	MA005	MA006	MA013	MA014	MA021	MA022	MA029	MA030
Bank 3/7	MA007	MA008	MA015	MA016	MA023	MA024	MA031	MA032
Write Reference	**Lower Bits**		**Upper Bits**		**Lower Bits**		**Upper Bits**	
Write data	00 – 09 16 – 25	10 – 15 26 – 31 64 – 69	32 – 41 48 – 57	42 – 47 58 – 63 70 – 75	00 – 09 16 – 25	10 – 15 26 – 31 64 – 69	32 – 41 48 – 57	42 – 47 58 – 63 70 – 75
Address	00 – 11	12 – 20	00 – 11	12 – 20	00 – 11	12 – 20	00 – 11	12 – 20
Ref type	00 – 03	00 – 03	00 – 03	00 – 03	00 – 03	00 – 03	00 – 03	00 – 03
Subsection bit	00	01	02	03	00	01	02	03
Bank bit	02	02	02	02	02	02	02	02
Return path bit	00	01	00	01	00	01	00	01

The MA options generate three control signals that enter the memory stack: **Chip Select**, **Write Enable**, and **Clock**. The **Chip Select** signal and the **Write Enable** signal must be present for a write reference to complete. The **Clock** signal is a copy of the clock that is sent to the memory stack.

Write Data Path

Figure 31 shows the read and write data path for processors 0 through 3, and Figure 32 shows the read and write data path for processors 4 through 7. Refer to these illustrations for detailed descriptions of the data flow between options on a memory module.

Data enters the MF options in packet form; the lower bits (0 through 31 and 64 through 69) arrive on the odd-numbered MF options in 1 clock period, and the upper bits (32 through 63 and 70 through 75) arrive on the even-numbered MF options in the next clock period. Each MF option sends data to four MC options. The MC options distribute the data to the MA options, which direct it to the appropriate memory bank. The MC options create a **Special** (**Sp**) bit. This **Sp** bit is used by the MA options to configure around a bad memory chip. Refer to "Spare Memory Chip Configuration" on page 24 for more information. Figure 26 shows the write data path of the lower bits for processor 0.

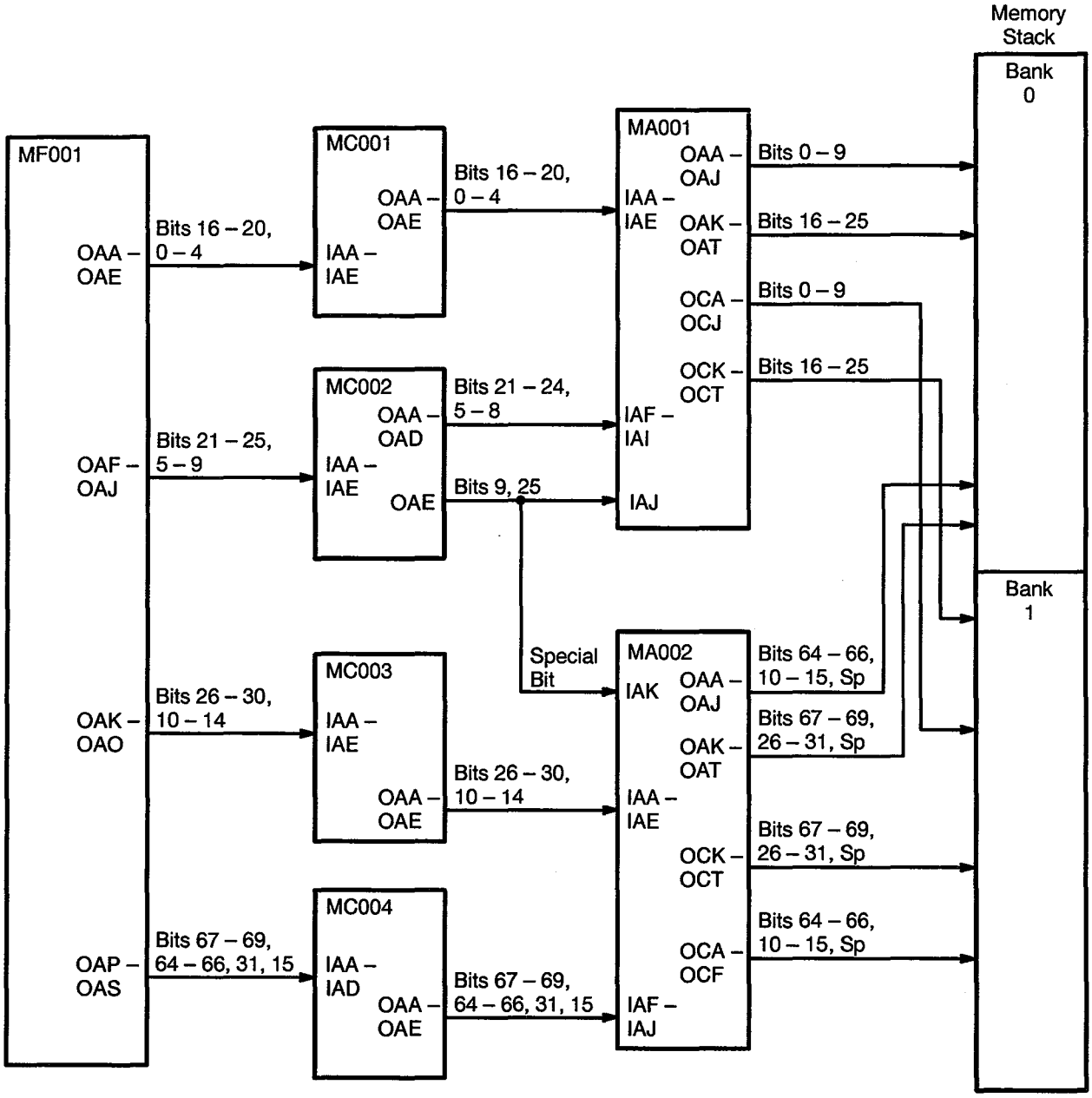


Figure 26. Write Data Flow for Processor 0

Memory Read Operation: A Block Diagram Description

Refer to the memory read block diagram, Figure 30, when reading the following information. When a CPU reads data from memory, it sends a Valid signal followed by control signals to the memory module. The Valid and control signals arrive on the memory module through one of eight ZA connectors that are associated with a CPU access port. These signals flow through the MF, MC, and MA options and steer the read reference to the designated location in the memory stack to retrieve the data.

Data leaves the memory stack and enters the MA options. The MA options send the data and control signals back to the CPU through the MC and MF options. In addition, the MA options generate a Destination Code that is sent back to the CPU as part of the read reference informing the CPU where to put the data.

Read Control Signals

Figure 30 shows the control signals associated with a memory read operation; however, the figure illustrates only the options associated with a read reference for the lower data bits in banks 0 and 4 from processors 0 through 3.

Three control signals are associated with a memory read operation: Subsection Bits, Bank Bits, and a **Destination Code**. The Subsection Bits (**Subsection Bits 0, 1, 2, 3**) flow through the options on the memory module and are used by the network module to determine which subsection the data came from. The Bank Bits (**Bank Bit 0, 1, 2**) are used by the MA and MC options to determine which bank the data came from. The CPU generates the **Destination Code,** which gets buried into the write data field on a read reference. When the MA option detects a read reference it checks the write data field and sends the Destination Code back to the CPU with the return data. The CPU uses the Destination Code to determine what to do with the data.

When the MA option detects a write reference, it forces several bits of the destination code. Table 19 lists the **Destination Code** bits generated by the MA option. There are 7 bits of destination code that the MA option uses. These bits are used by the CPU to determine that the reference actually completed. The CPU counts the number of references that were sent out and then checks this count against how many references completed. This information is sent to the JA options on the CP module to determine whether or not memory is quiet or if the CPU should generate a hold issue condition. Figure 33 and Figure 34 are block diagrams showing the path the destination code takes from the MA options on the memory module to the JA options on the CP module.

(completion)

Table 19. MA Option Destination Code Forced Bits

Function	OFZ	OFY	OFX	OFW	OFV	OFU	OFT
Read/abort	IAG	IAF	IAE	IAD	IAC	IAB	IAA
I/O write	0	0	0	ICC	ICB	ICA	0
Processor write	0	0	0	0	0	0	1
Set bad bit (Reconfigure)	0	0	0	0	0	1	1

Data leaves the memory stack and enters the MA options. The MA options also receive two control signals from the MC options: the **Reference Type** and the **Return Path** bit.

The **Reference Type** bits are 4 bits used by the MA options to notify memory of the type of reference coming from the CPU. Table 15 lists the terms associated with the **Reference Type** bits and decodes the bits by CPU function.

The **Return Path** is a 2-bit code that steers the reference back to the CPU module that sent the reference. Read reference control signals pass through the MA options and enter the MC options. Table 20 lists the control and data that leave each MA option.

Table 20. MA Option Bit Assignments (Read)

	Processor 0 – 3				**Processor 4 – 7**			
Bank 0/4	MA001	MA002	MA009	MA010	MA017	MA018	MA025	MA026
Bank 1/5	MA003	MA004	MA011	MA012	MA019	MA020	MA027	MA028
Bank 2/6	MA005	MA006	MA013	MA014	MA021	MA022	MA029	MA030
Bank 3/7	MA007	MA008	MA015	MA016	MA023	MA024	MA031	MA032
Read Reference	**Lower Bits**		**Upper Bits**		**Lower Bits**		**Upper Bits**	
Read data	00 – 15 64 – 66	16 – 31 67 – 69	32 – 47 70 – 72	48 – 63 73 – 75	00 – 15 64 – 66	16 – 31 67 – 69	32 – 47 70 – 72	48 – 63 73 – 75
Destination code	00 – 06	--	07 – 13	--	00 – 06	--	07 – 13	--
Return bit	00	01	00	01	00	01	00	01
Subsection bit	00	01	02	03	00	01	02	03
Bank bit	--	02	--	--	--	02	--	--

The MC options use the **Return Path** bits to determine which bank or processor the reference came from. Then, the MC options drop these bits and convert them to Bank Bits. The Bank Bits inform the CPU which memory bank the data came from.

During a read reference, MC001, MC002, MC009, and MC010 report errors. These options have forced bank bits on them that determine in which bank the error occurred. **Bank Bit 2** is used to determine whether the error occurred in the upper or lower banks. Table 21 lists the bank bits used for error correction in options MC001, MC002, MC009, and MC010.

Table 21. Bank Bits for Error Correction

MC001 MC009	MC002 MC010	
Bank Bit 1	Bank Bit 0	Bank Number
Forced 0 (IEN)	Forced 0 (IEN)	Bank 0/4
Forced 0 (IFN)	Forced 1 (IFN)	Bank 1/5
Forced 1 (IGN)	Forced 0 (IGN)	Bank 2/6
Forced 1 (IHN)	Forced 1 (IHN)	Bank 3/7

Control signals leave the MC options and enter the MF options. Table 22 lists the data and control signals that leave the MC options. Table 23 lists the data and control signals that enter the MF options. The MF options send the data and control signals back to the CPU through one of eight CPU access ports.

Table 22. MC Option Bit Assignments (Read)

Processor 0 – 3	MC001	MC002	MC003	MC004	MC005	MC006	MC007	MC008
Processor 4 – 7	MC009	MC010	MC011	MC012	MC013	MC014	MC015	MC016
Read reference	Lower Bits				Upper Bits			
1st data packet 2nd data packet	00 – 09	10 – 15 64 – 66	16 – 25	26 – 31 67 – 69	32 – 41	42 – 47 70 – 72	48 – 57	58 – 63 73 – 75
Destination code	00 – 01	02 – 03	04 – 05	06	07 – 08	09 – 10	11 – 12	13
Subsection bit	00	– –	01	– –	02	– –	03	– –
Bank bit	00	01 – 02	03	04 – 05	– –	06	– –	– –

Table 23. MF Option Bit Assignments (Read)

Processor 0	
MF001	MF 002
Processor 1	
MF 003	MF 004
Processor 2	
MF 005	MF 006
Processor 3	
MF 007	MF 008
Processor 4	
MF 009	MF 010
Processor 5	
MF 011	MF 012
Processor 6	
MF 013	MF 014
Processor 7	
MF 015	MF 016
Read Data 0 – 31, 64 – 69	**Read Data** 32 – 63, 70 – 75
Destination code bits 0 – 6	Destination code bits 7 – 13
Subsection bits 0 – 1	Subsection bits 02 – 03
Bank bits 0 – 2	Bank bits 3 – 6

Read Data Path

Figure 31 shows the read and write data path for processors 0 through 3, and Figure 32 shows the read and write data paths for processors 4 through 7. Refer to these illustrations for detailed descriptions of the data flow between options on a memory module.

Data leaves the memory stack as a 76-bit data word and enters the MA options. The **Special Bit (Sp)** is the spare bit configured in the memory stack when a bad memory chip exists. If this bit is used, the MA options shift the bits to bypass the bad chip during a write operation and shift them back during a read operation. Refer to "Spare Memory Chip Configuration" on page 24 for more information. Data leaves the MA options and flows through the MC and MF options before it leaves the memory module through the ZA connectors. Figure 27 shows the read data path of the lower bits for processor 0.

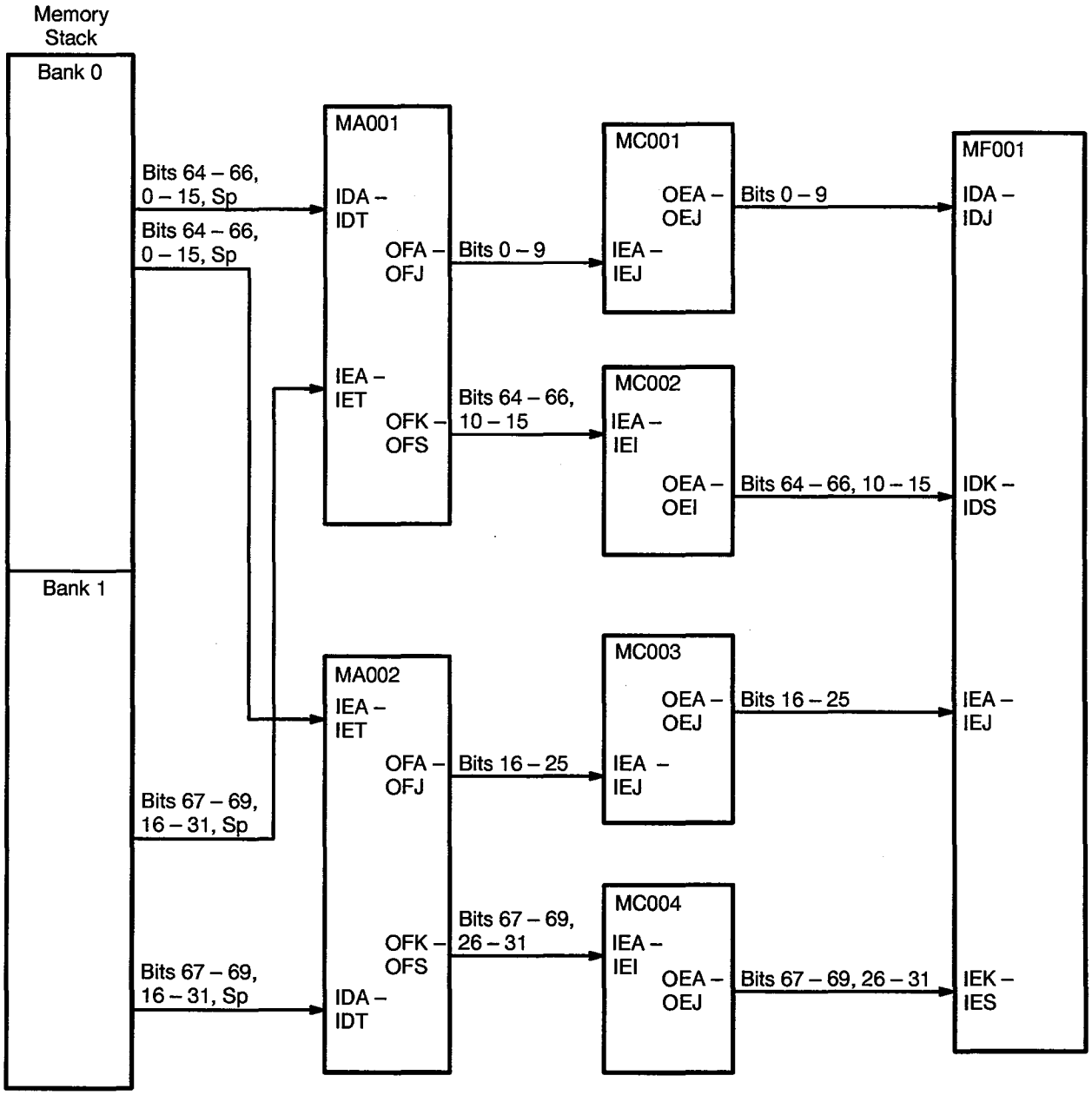


Figure 27. Read Data Flow for Processor 0

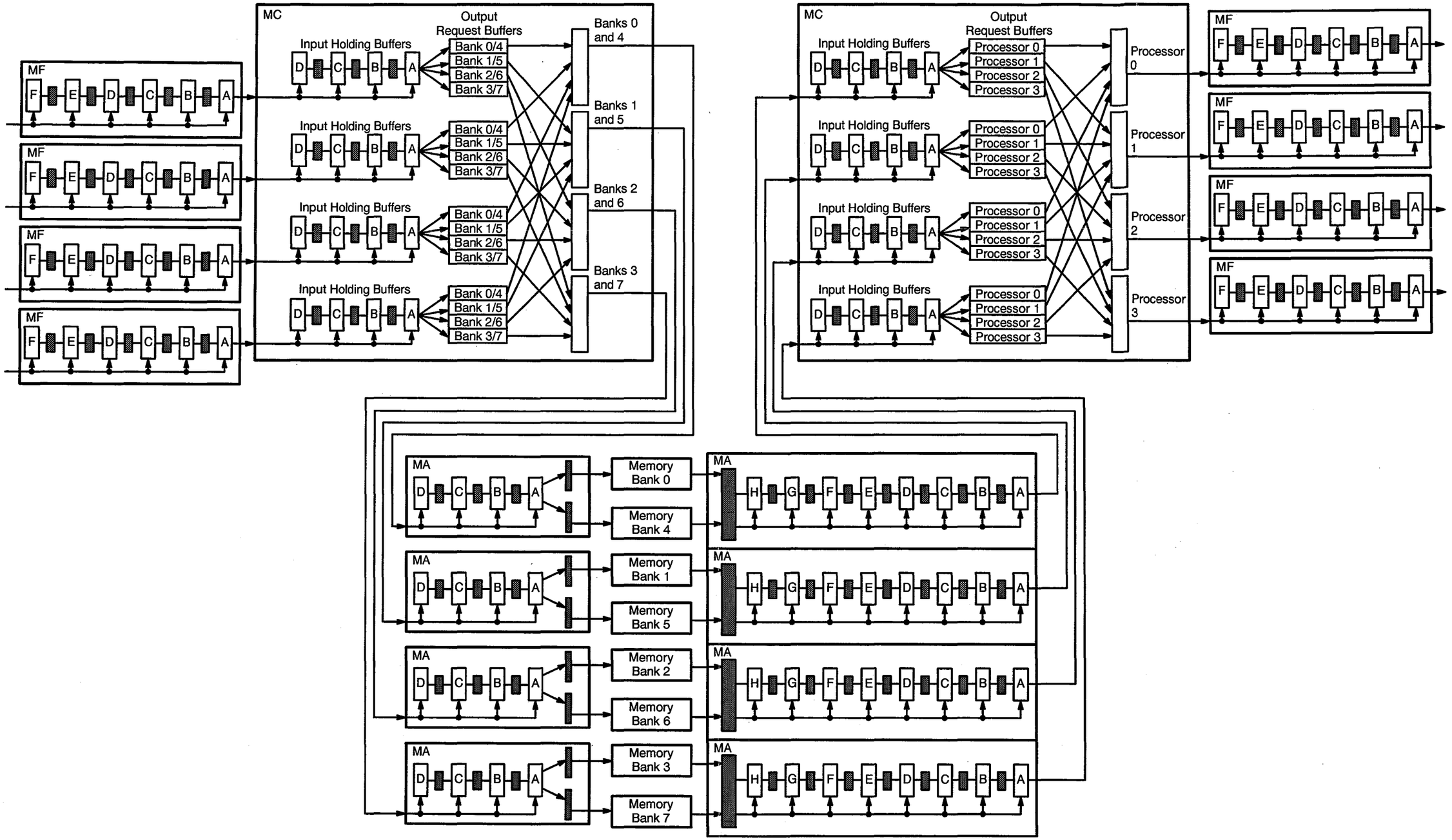


Figure 28. CM02 Module Block Diagram

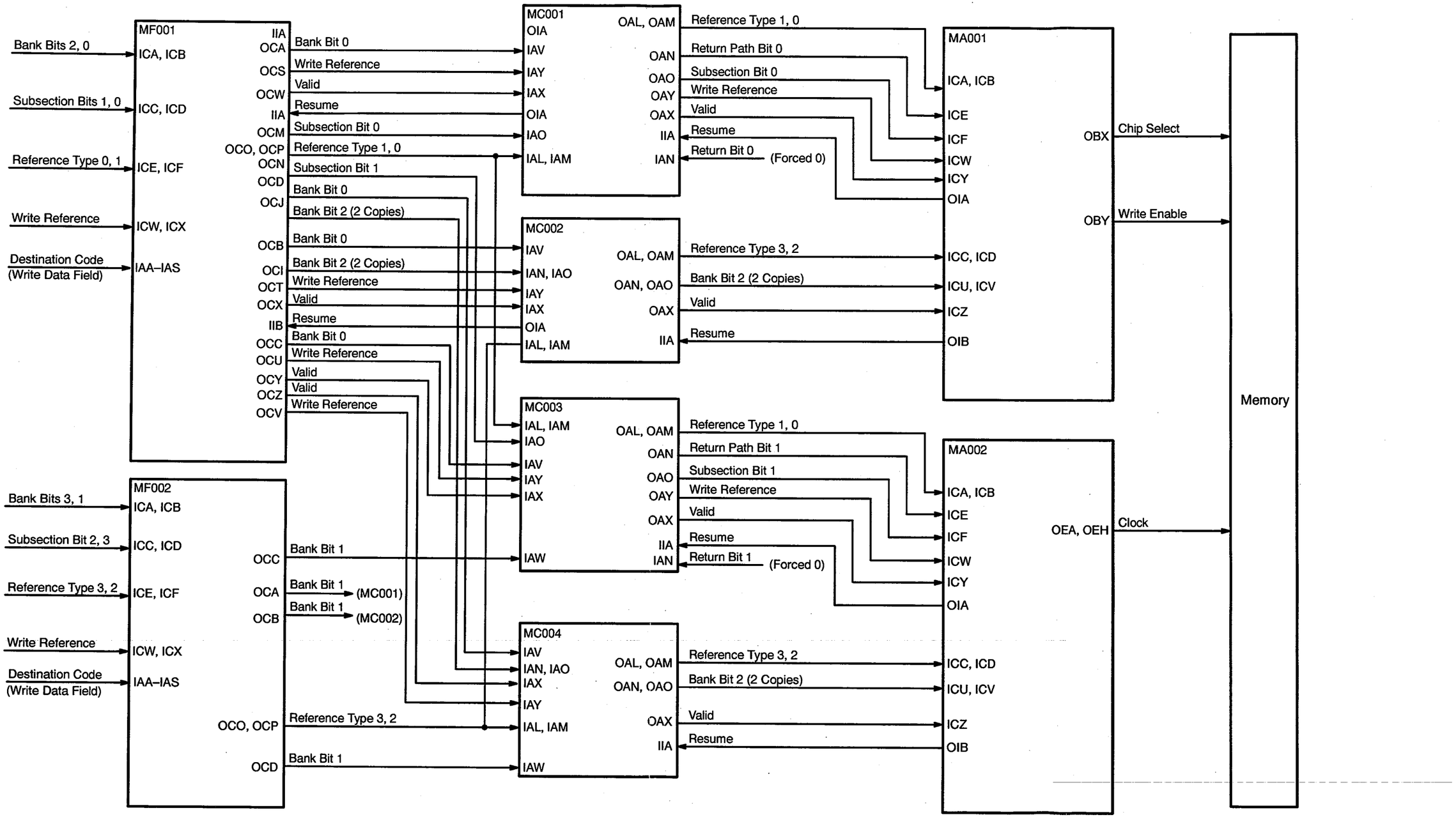


Figure 29. Memory Write Operation Control Signals (Lower Bits, Processor 0)

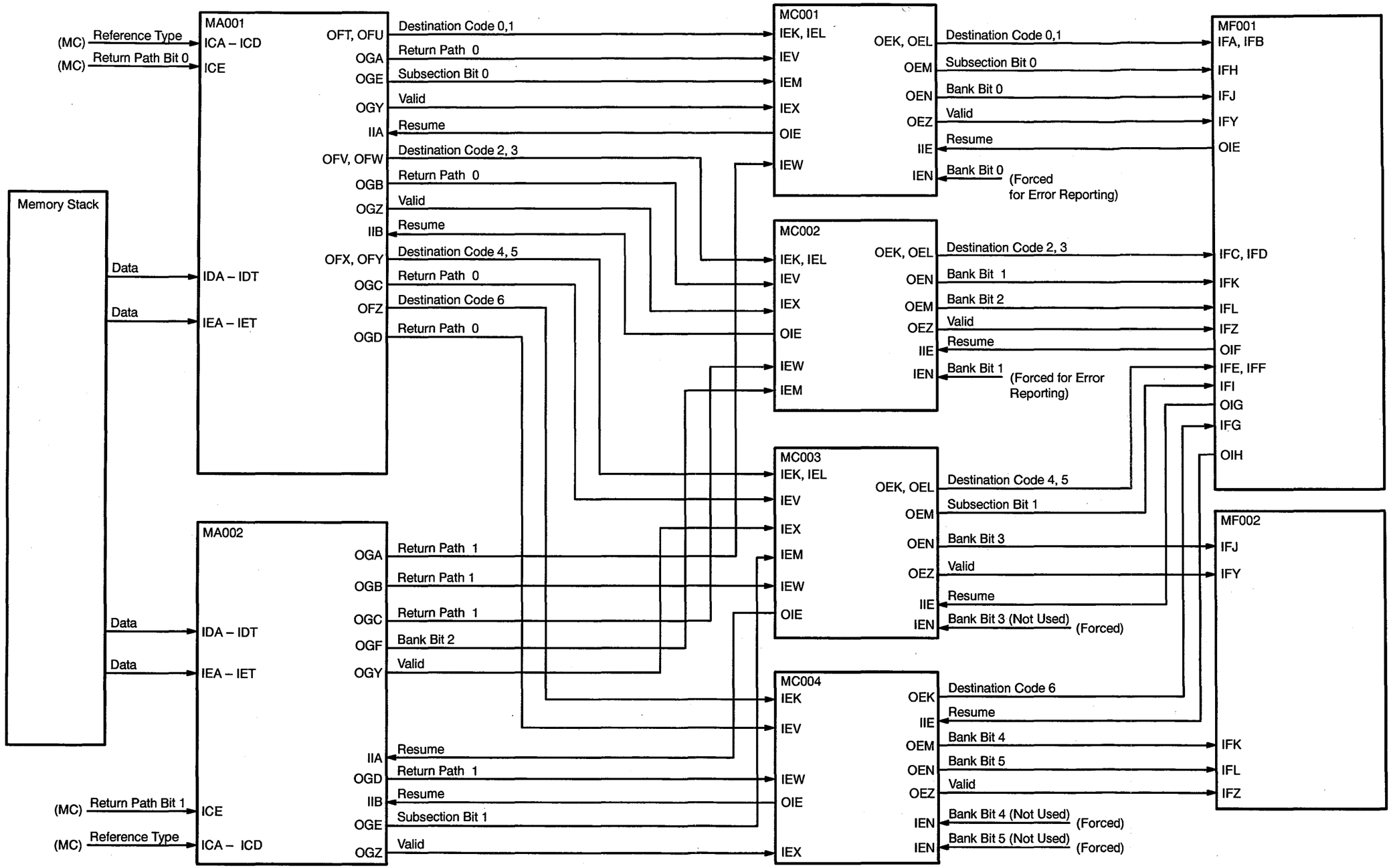


Figure 30. Memory Read Operation Control Signals (Lower Bits, Processor 0 Bank 0/4)

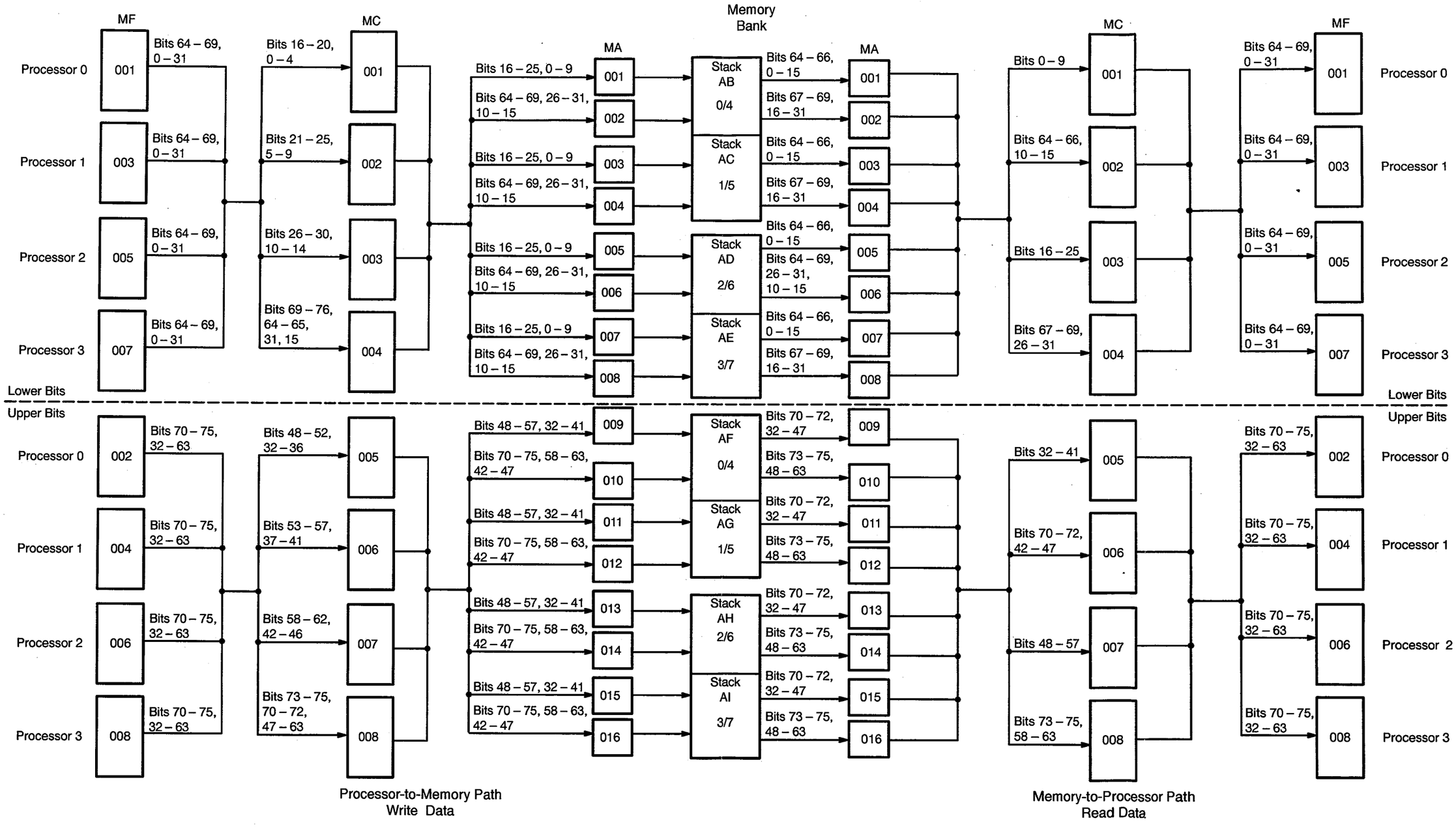


Figure 31. Read and Write Data Path for Processors 0 – 3 Section N

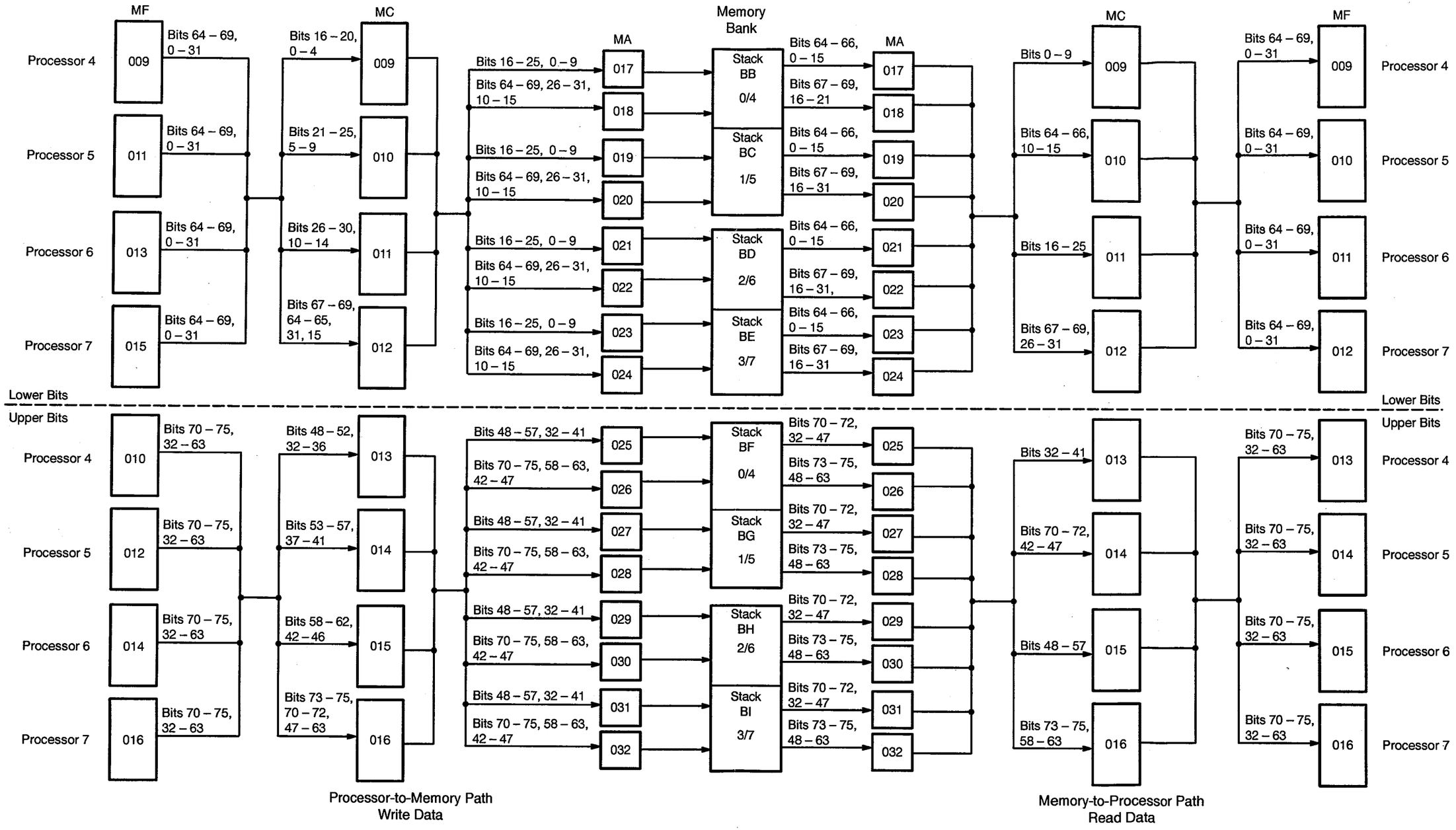


Figure 32. Read and Write Data Path for Processors 4 – 7 Section N + 1

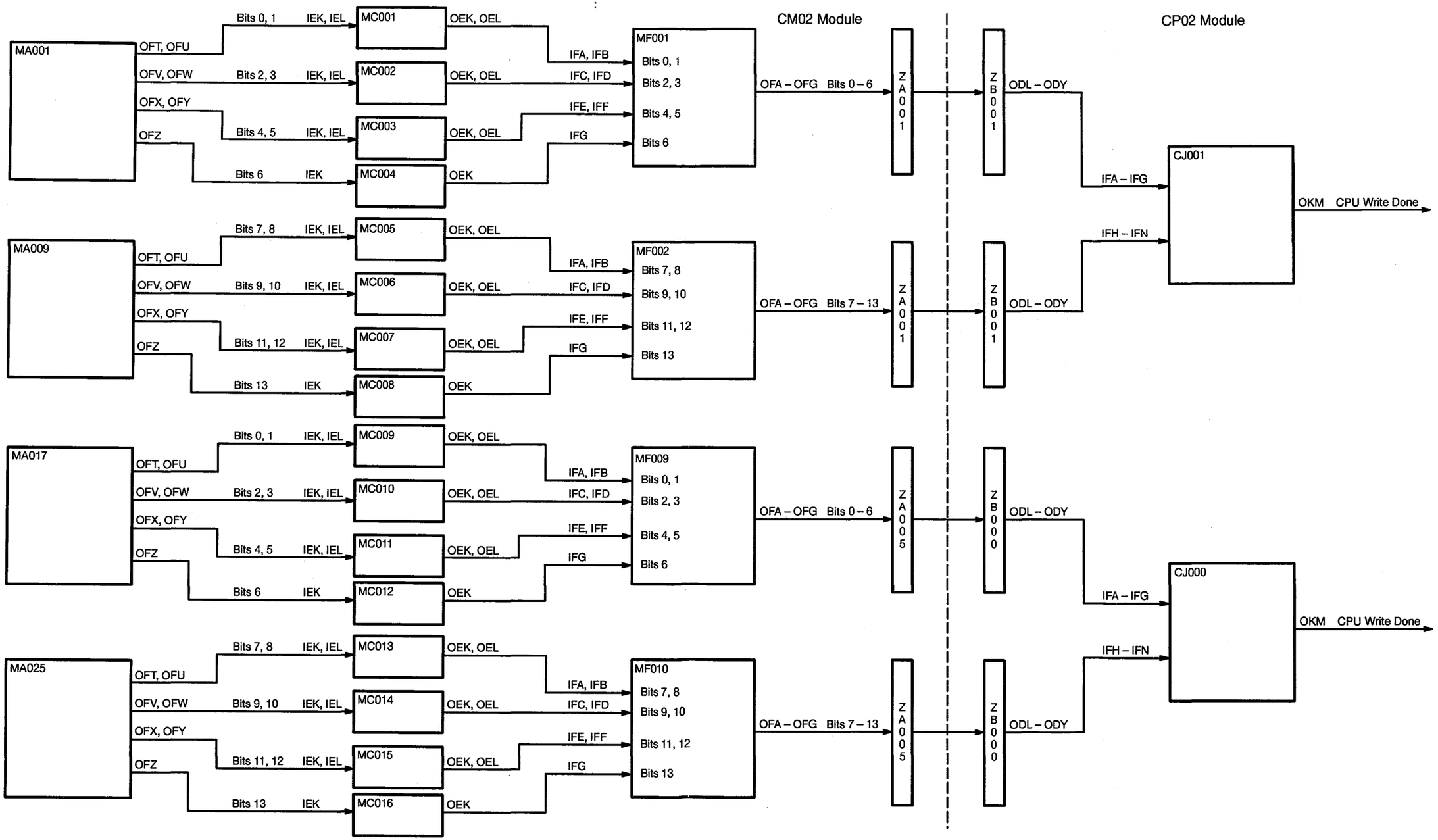


Figure 33. Memory Module Destination Code (Section 0/1)

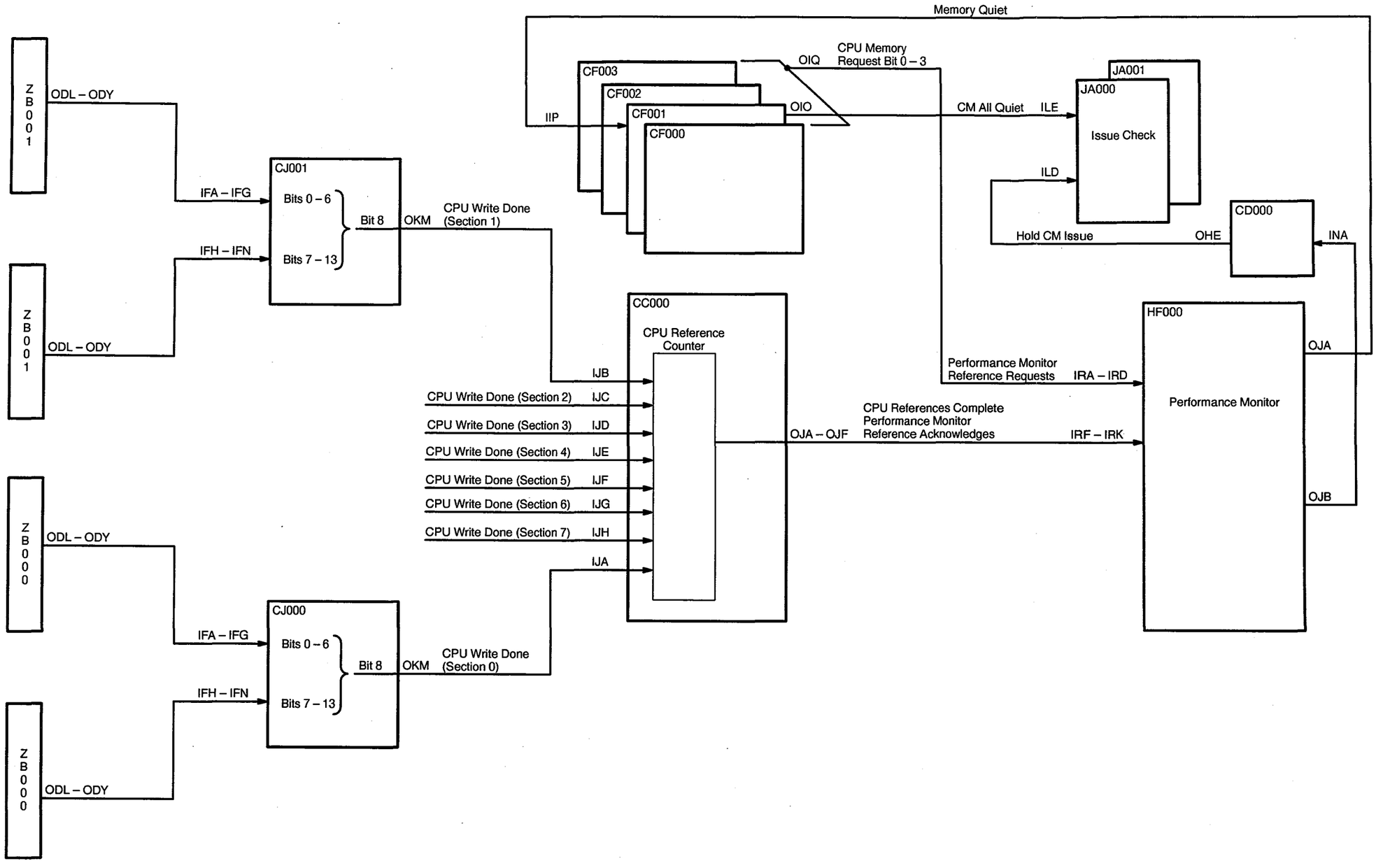


Figure 34. CPU Distribution of Destination Code from Memory Module

Reader Comment Form

Title: CM02 Memory Module
Preliminary Information

Number: HTM-xxx-x
December 1994

Your feedback on this publication will help us provide better documentation in the future. Please take a moment to answer the few questions below.

For what purpose did you primarily use this document?

_____Troubleshooting
_____Tutorial or introduction
_____Reference information
_____Classroom use
_____Other - please explain _____

Using a scale from 1 (poor) to 10 (excellent), please rate this document on the following criteria and explain your ratings:

_____Accuracy _____
_____Organization _____
_____Readability _____
_____Physical qualities (binding, printing, page layout) _____
_____Amount of diagrams and photos _____
_____Quality of diagrams and photos _____

Completeness (Check one)

_____Too much information _____

_____Too little information _____

_____Just the right amount of information

Your comments help Hardware Publications and Training improve the quality and usefulness of your publications. Please use the space provided below to share your comments with us. When possible, please give specific page and paragraph references. We will respond to your comments in writing within 48 hours.

NAME _____
JOB TITLE_____
FIRM_____
ADDRESS_____
CITY_____STATE_____ZIP_____
DATE_____

[or attach your business card]

CRAY RESEARCH, INC.