CRAY X-MP/2

Hardware Training Volume 2

HTV-0242

This document cannot be sent outside the United states without a Validated Export License.

VECTOR REGISTERS

SHARED REGISTERS

REAL-TIME/PROGRAMMABLE CLOCK

PERFORMANCE LOG

A REGISTERS FUNCTIONAL UNITS

ADD

MULTIPLY

S REGISTERS FUNCTIONAL UNITS

ADD

POP

FLOATING-POINT FUNCTIONAL UNITS

CONSTANT

ADD

RECIPROCAL

LOSP

LOSP OUTPUT SEQUENCE

LOSP TIMING DIAGRAM

HISP

HISP OUTPUT SEQUENCE

HISP INPUT SEQUENCE

VHISP

REQUIREMENTS FOR COURSE COMPLETION

At the end of each course, an evaluation is written which asks whether the student completed the course. Due to the nature of the courses in Hardware Training, some courses are referred to as theory only; others are theory-lab courses. The determining factors of whether a student receives a Certificate of Completion or Certificate of Attendance are listed as follows:

Theory Only Courses

Theory only courses are courses which do not contain a scheduled period of lab instruction.

a) All exercises (in-class and take-home) are completed and handed in to the instructor.
b) Should a student miss an exercise or test due to absence, it is the responsibility of the student to make up the missing exercise.
c) Students pass with a minimum overall average of 70%.
d) Students attend all class sessions unless an excused absence is authorized by the instructor and/or supervisor.
e) Students are not tardy more than three times unless authorized by the instructor.
f) Students are responsible for informing the instructor for the reason of absence or tardiness.

Theory-lab Courses

Theory-lab courses are courses which do contain a scheduled period of lab instruction.

a) All of the above.
b) Student completes all prerequisites prior to class unless a variance is authorized by a system Supervisor or the Training Manager.
c) Student passes the pretest, when given, prior to the start of the instructional material.
d) A final written exam is passed by the student with a minimal grade of 70%.
e) Student passes the practical lab sessions and the final practical exam.
f) All lab check-off sheets for skills-based training are signed by both the student and instructor and are handed in to the instructor.

If the qualifications or standards as listed above are not met, the student will receive a Certificate of Attendance rather than a Certificate of Completion.

ACKNOWLEDGEMENTS

The technical information in this volume was obtained from Engineering and other Hardware Training volumes.

The following individuals have provided technical updates:

Rod Anderson
Dennis Brown
Ginny Fetter
Gary Kroening
Rick Slick
John Spiegel

COURSE DESCRIPTION

CRAY X-MP Hardware Maintenance

Duration: 9 weeks

Prerequisite: Cray Systems Introduction.

Deals with the theory of the CRAY X-MP design combined with system lab practical sessions. Emphasizes the theory and practical use of off-line diagnostics for troubleshooting a CRAY X-MP failure. Includes a final practical test and final theory test.

MATERIALS REQUIRED

The following materials will be required to complete CRAY X-MP Hardware Maintenance:

- CMQ0105010 CRAY X-MP CAL Version 1 Reference Card
- HM-1010B CRAY X-MP/IOS Based Diagnostic Reference Manual
- HMM-0607 CRAY X-MP/22, 24 Chassis Map (Size C)
- HMM-0801 CRAY X-MP/216 Chassis Map (Size C)
- HMM-0805 CRAY X-MP/216 Block Diagram (Size C)
- CID0100000 CRAY X-MP/216 Instruction Timing Diagrams
- CSM0110000 CRAY X-MP/22, 24, 28 and 216 Hardware Reference Manual
- CCW0115000 CRAY X-MP/14se Chassis Map (Size C)
- HR-3005 CRAY X-MP Computer Systems Functional Description Manual
- HTV-0242 CRAY X-MP Hardware Training Volume; Volume 2
- HTV-0243 CRAY X-MP Hardware Training Volume; Volume 3
- HQ-1005 CRAY X-MP Diagnostic Ready Reference Guide

The following materials will be used and will remain in the classroom:

- Booolean
- Wire Tabs
- Diagnostics
- A Practical Guide to UNIX System V (by Mark G. Sobell)

SKILLS LIST

CRAY X-MP PRACTICAL SKILLS LIST

NAME: _____ DATE: _____

HOW A	HOW B	HOW C	*LOCATOR*	LEVEL
			1. MG's, compressors, mainframe, PDU, breakers	
			2. Voltage sense wires, power indicator lights & regulator sense wires	
			3. Thermostats, thermistors & fault indicator lights on PDU	
			4. Power supply voltage indicators - LED's	
			5. Cheater switch on PDU	
			6. Halon system alarms, sensors, panel and disarm panic buttons	
			7. CRAY X-MP, IOS, and SSD (if available) maintenance panels	
			8. Identify random drop cables (purpose and connection based on label)	
			9. Find a cable, given only its purpose and connections	

			ADJUSTMENTS	
			1. PDU voltage for normal operation	
			2. High, low voltage margins	
			3. Fast, slow, super slow and normal clocks	
			4. Scanner high temperature controls	

			PROCEDURES	
			1. Power on PDU	
			2. Power on CRAY X-MP	
			3. Power On/Off Micro MCU (A:HDOFF)	
			4. Column power cycle (any column)	
			5. Cable Micro MCU to CRAY X-MP	
			6. Cable IOS to CRAY X-MP	
			7. Power on and allow connect of Remote Maintenance facility	
			8. Take and evaluate compressor readings	

			MANUALS AND BOOKS	LEVEL
			1. Daily log book	
			2. Swap log	
			3. Diagnostic listings	
			4. HMM-1010	
			5. Quick Reference Guide	
			6. PMI/PMP HM1003	
			7. Maintenance aids	

			SOFTWARE BASICS	
			1. Review monitors (MTA, MTI, AMP 2 and/or AMP 4)	
			2. Review modes	
			3. Run system	
			4. Command buffers	
			5. Run CPUM0 or :B01	
			6. CMOS	
			7. Micro MCU	
			8. Bring up Error Logger	
			9. Show different table in Error Logger e.g. 0 - CPU 0 error information 1 - CPU 1 error information	

			TROUBLESHOOTING BUGS	
			1. Write Code, Scope Loop	
			2. Scoping	
			3. Scope Charts, Training Aids	

COMMENTS: _____

_____

KEY:

	A	B	C
HOW	Shown or Video	Lecture or Theory	Performed

LEVEL Excellent - 5 Average - 3 Unsatisfactory - 1
Above Average - 4 Below Average - 2 Not Apply - N/A

Student's Signature _____

Instructor's Signature _____

If you sign this sheet, you are responsible on site for this task.

MF02W04A

BLOCK DIAGRAM ANALYSIS

OBJECTIVES

With the aid of student reference material, upon completion of the course, a student should be capable of:

I. Describing the overall specifications of the memory including size, speed, data format, and address format.

II. Determining the storage module(s) associated with a work or bit of data, given its memory address.

III. Describing interleaving and its effect on memory speed.

IV. Interpreting memory parity error information.

V. Describing the main characteristics of all the register types in the X-MP.

VI. Describing how, when, and where instructions are loaded into the CPU.

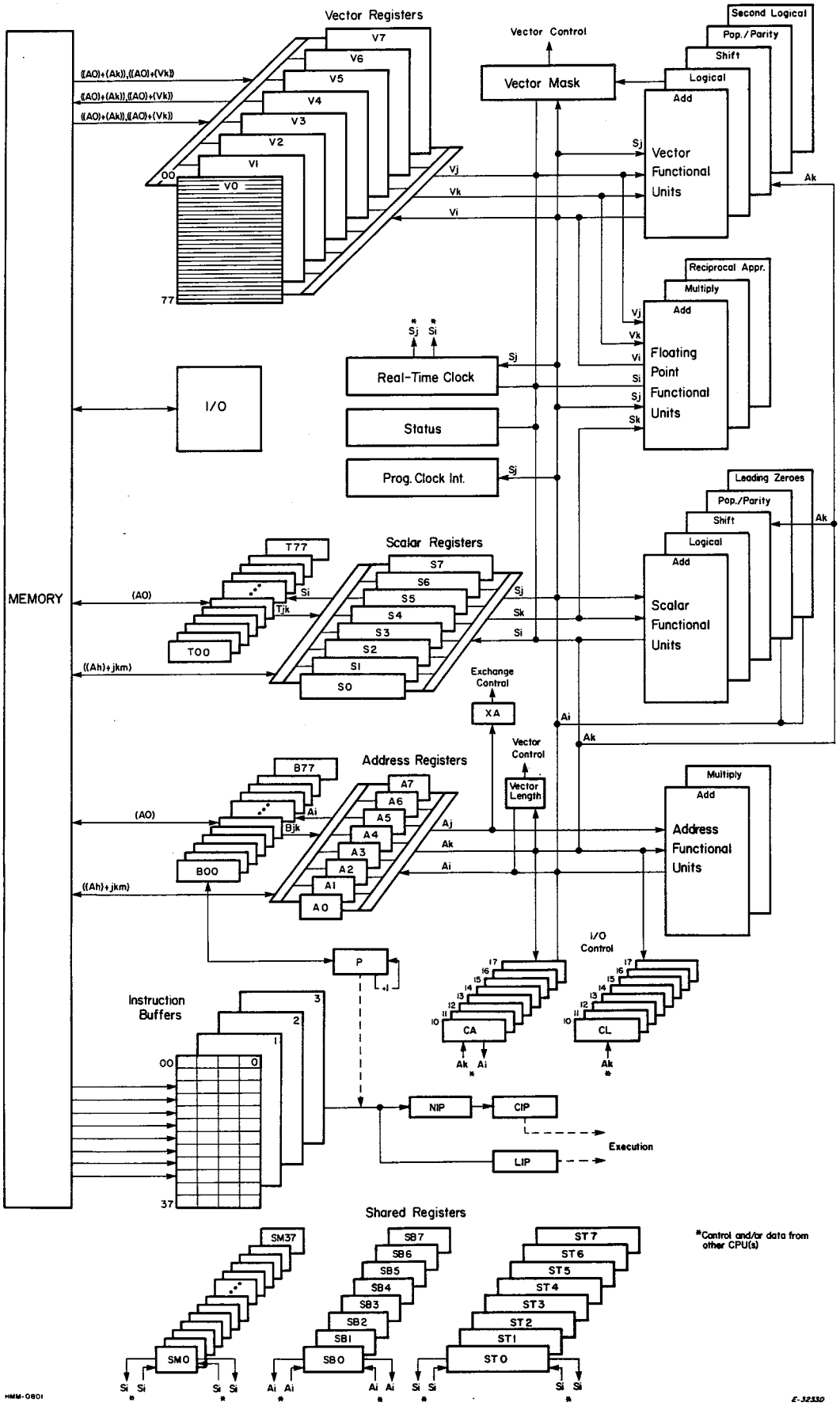
VII. Describing the flow of instructions from memory to the time of issue.

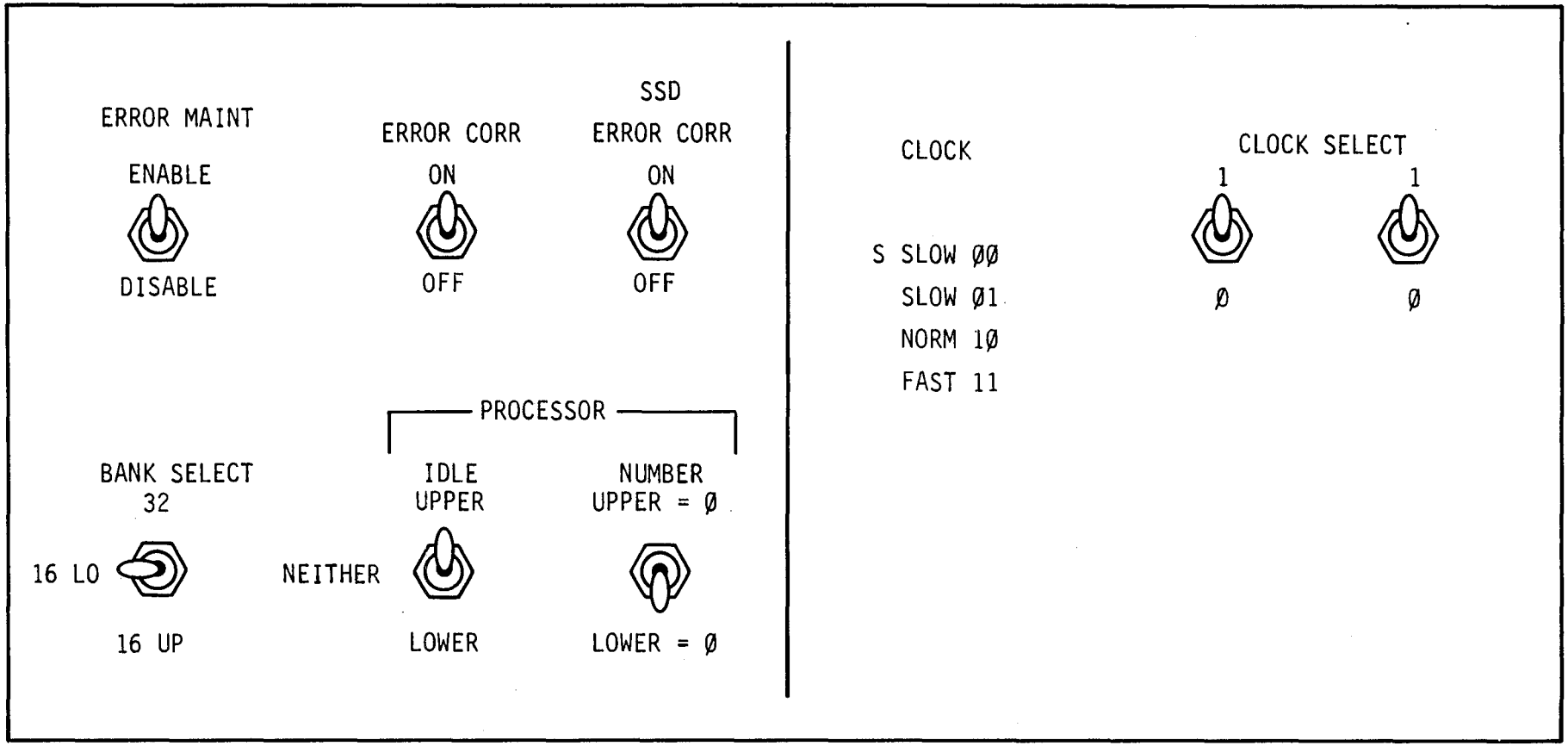
VIII. Describing the method that Cray uses for swapping control from one program to another.

IX. Generating the addresses associated with all types of memory references.

X. Determining the conditions that would delay the issuing of any given instruction.

CRAY X-MP BLOCK DIAGRAM





CRAY X-MP SWITCH PANEL

REFRIGERATION AND POWER CONSIDERATION

I. Cray X-MP

A. SN101-114 X-MP System Configuration

1. Power: All systems must have 3 MG's; 2 running, 1 standby.
   a. This will handle worst case 4 meg. word X-MP, 4 process or IOP, and 32 meg. word SSD.
2. Refrigeration: All systems
   a. 1-50 ton compressor for X-MP.
   b. 1-25 ton compressor for IOP and SSD.
   c. Overseas -- 2-50 ton compressors.

B. SN115 X-MP System Configuration

1. Power: The following need 3 MG's; 2 running, 1 standby. The rest 2 MG's; 1 running, 1 standby.

a.	4M X-MP	4Pr. IOP	32M SSD
b.	4M X-MP	4Pr. IOP	16M SSD
c.	4M X-MP	4Pr. IOP	08M SSD
d.	4M X-MP	3Pr. IOP	32M SSD
e.	4M X-MP	3Pr. IOP	16M SSD
f.	4M X-MP	3Pr. IOP	08M SSD
g.	2M X-MP	4Pr. IOP	32M SSD
h.	2M X-MP	4Pr. IOP	16M SSD
i.	2M X-MP	3Pr. IOP	32M SSD

2. Refrigeration: All systems
   a. 1-50 ton compressor for system. Possibly an added 25 tons.
   b. Overseas -- probably 2-50 ton compressors.

II. Cray-1 S

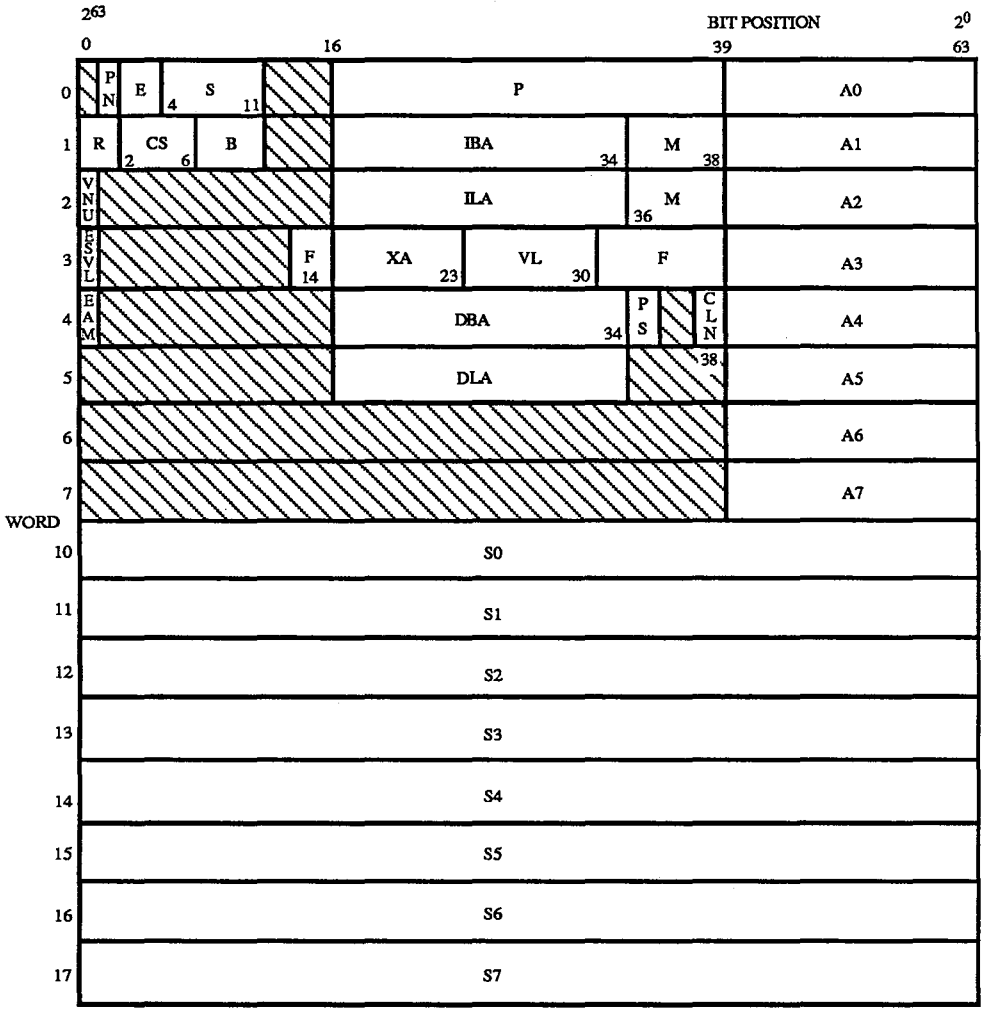
A. Cray-1 S System Configuration

1. Power:
   a. 3 MG's with IOP and/or SSD; 2 running, 1 in standby.
   b. 2MG's for Cray-1 S; 1 running, 1 in standby.
2. Refrigeration:
   a. 1-50 ton for Cray-1 S.
   b. 1-25 ton for IOP and/or SSD.

III. Cray-1 M

A. Cray-1 M System Configuration

1. Power:
   a. 1 MG with IOP and/or SSD; 1 running, standby optional.
2. Refrigeration:
   a. 1-50 ton with IOP and/or SSD.



READ OUT AREA

WORD	BIT	FIELD
0	2^{62}	PN - Processor Number
0	2^{61} - 2^{60}	E - Error Type
0	2^{59} - 2^{52}	S - Syndrome Bits
1	2^{63} - 2^{62}	R - Read Mode
1	2^{61} - 2^{57}	CS - Chip Select Bits
1	2^{56} - 2^{52}	B - Bank Address

STATUS AREA

WORD	BIT	FIELD
0	2^{24} - 2^{47}	P - Program Address Register
1	2^{29} - 2^{47}	IBA - INSTRUCTION BASE ADDRESS
2	2^{29} - 2^{47}	ILA - INSTRUCTION LIMIT ADDRESS
3	2^{40} - 2^{47}	XA - EXCHANGE ADDRESS
3	2^{33} - 2^{39}	VL - Vector Length
4	2^{29} - 2^{47}	DBA - Data Base Address
5	2^{29} - 2^{47}	DLA - Data Limit Address

WORD	BIT	FIELD
1&2	2^{24} - 2^{28}	M - Mode Register
1	2^{24}	IMM - Interrupt Monitor Mode
1	2^{25}	SEI - Select for External Interrupt
1	2^{26}	BDM - Bidirectional Memory Access
1	2^{27}	FPS - Floating Point Error Status
1	2^{28}	WS - Waiting on Semaphore
2	2^{24}	MM - Monitor Mode
2	2^{25}	IUM - Interrupt on Uncorrectable Memory Error
2	2^{26}	IFP - Interrupt on Floating Point Error
2	2^{27}	ICM - Interrupt on Correctable Memory Error
2	2^{28}	IOR - Interrupt on Operand Range Error
2	2^{63}	VNU - Vector Not Used
3	2^{63}	ESVL - Enable Second Vector Logical
4	2^{63}	EAM - Enhance Addressing Mode

FLAGS

3	2^{49}	ICP - Interrupt from Internal CPU
3	2^{48}	DL - Deadlock
3	2^{32}	PCI - Program Clock
3	2^{31}	MCU - MCU Interrupt (MIOP)
3	2^{30}	FPE - Floating Point Error
3	2^{29}	ORE - Operand Range Error
3	2^{28}	PRE - Program Range Error
3	2^{27}	ME - Memory Error
3	2^{26}	IOI - I/O Interrupt
3	2^{25}	EEX - Error Exit (000)
3	2^{24}	NEX - Normal Exit (004)
4	2^{28}	PS - Program State
4	2^{24} - 2^{25}	CLN - Cluster Number Register

A-0929E

CRAY X-MP EXCHANGE PACKAGE

Main sections of the X-MP that are used in memory operation are as follows:

1. Memory addressing - The address path as it moves from the CPU to each of the memory modules.

2. Memory control - The section and bank selection logic that decodes the low order 5 bits of address and selects one of the possible 32 banks. Prior to bank selection the memory ports are checked for the following conflicts:
   1. Bank busy conflict
   2. Simultaneous bank conflict
   3. Section conflict

3. Input data path and check bit generation - The data path from CPU or I/O section to the memory modules. This path includes the check bit generation logic.

4. Memory storage - Where the data is stored during a write operation and where its read out of during a read operation.

5. Output data and error correction - The data path from memory to the CPU or I/O section. This data path includes the error correction logic.

6. MCU and exchange status - The module responsible for generating status information, address and syndrome, if a memory error should occur.

X2002S0417

X-MP MEMORY CONFIGURATIONS

MODEL	SERIAL NUMBERS	CHIP SIZE AND TYPE	MODULE TYPE	NUMBER OF BANKS	NUMBER OF COLUMNS	ADDRESS FORMAT		
						C.S.	INTERNAL	BK/SEC
X-MP/12	301 -UP	16K MOS	3YA	16	6	20-18	17----4	3--0
X-MP/14	301 -UP	16K MOS	3YA	32	6	21-19	18----5	4--0
X-MP/14	301 -UP	64K MOS	3YM	16	6	21-20	19----4	3--0
X-MP/18	301 -UP	64K MOS	3YM	32	6	22-21	20----5	4--0
X-MP/22	101-109 111-114	4K ECL	3ZM	16	12	20-16	15----4	3--0
X-MP/24	101-109 111-114	4K ECL	3ZM	32	12	21-17	16----5	4--0
X-MP/22	110, 115 -UP	16K ECL	3ZV	16	6	20-18	17----4	3--0
X-MP/24	110, 115-UP	16K ECL	3YM	32	6	21-19	18----5	4--0
X-MP/24	401 -UP	64K MOS	3YM	16	8	21-20	19----4	3--0
X-MP/28	401 -UP	64K MOS	3YM	32	8	22-21	20----5	4--0
X-MP/216	401 -UP	64K MOS	3YM	32	8	22-21	20----5	23,4--0
X-MP/48	200 -UP	16K ECL	3ZV	64	12	23-21	20----6	5--0
X-MP/416	218 -UP	64K ECL	3ZU	64	12	23 22	21----6	5--0

XV204S23M

The SECDED error processing scheme is based on error detection and correction codes devised by R. W. Hamming.† An 8-bit check byte is appended to the 64-bit data word before the data is written in memory. The 8 check bits are each generated as even parity bits for a specific group of data bits. The Error Correction Matrix shows the bits of the data word used to determine the state of each check bit. An X in the horizontal row indicates that data bit contributes to the generation of that check bit. Thus, check bit 0 is the bit making group parity even for the group of 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} through 2^{55}.

The 8 check bits and the data word are stored in memory at the same location. When read from memory, the same 64-bit matrix is used to generate a new set of check bits, which are compared with the old check bits. The resulting 8 comparison bits are called syndrome†† bits (S bits). The states of these S bits are all symptoms of any error that occurred (1 = no compare). If all syndrome bits are 0, no memory error is assumed.

Any change of state of a single bit in memory causes an odd number of syndrome bits to be set to 1. A double error (an error in two bits) appears as an even number of syndrome bits set to 1.

The matrix is designed so that:

1. If all syndrome bits are 0, an error did not occur.

2. If only one syndrome bit is a 1, the associated check bit is in error.

3. If more than one syndrome bit is 1 and the parity of all syndrome bits (S0 - S7) is even, then a double error occurred within the data bits or check bits.

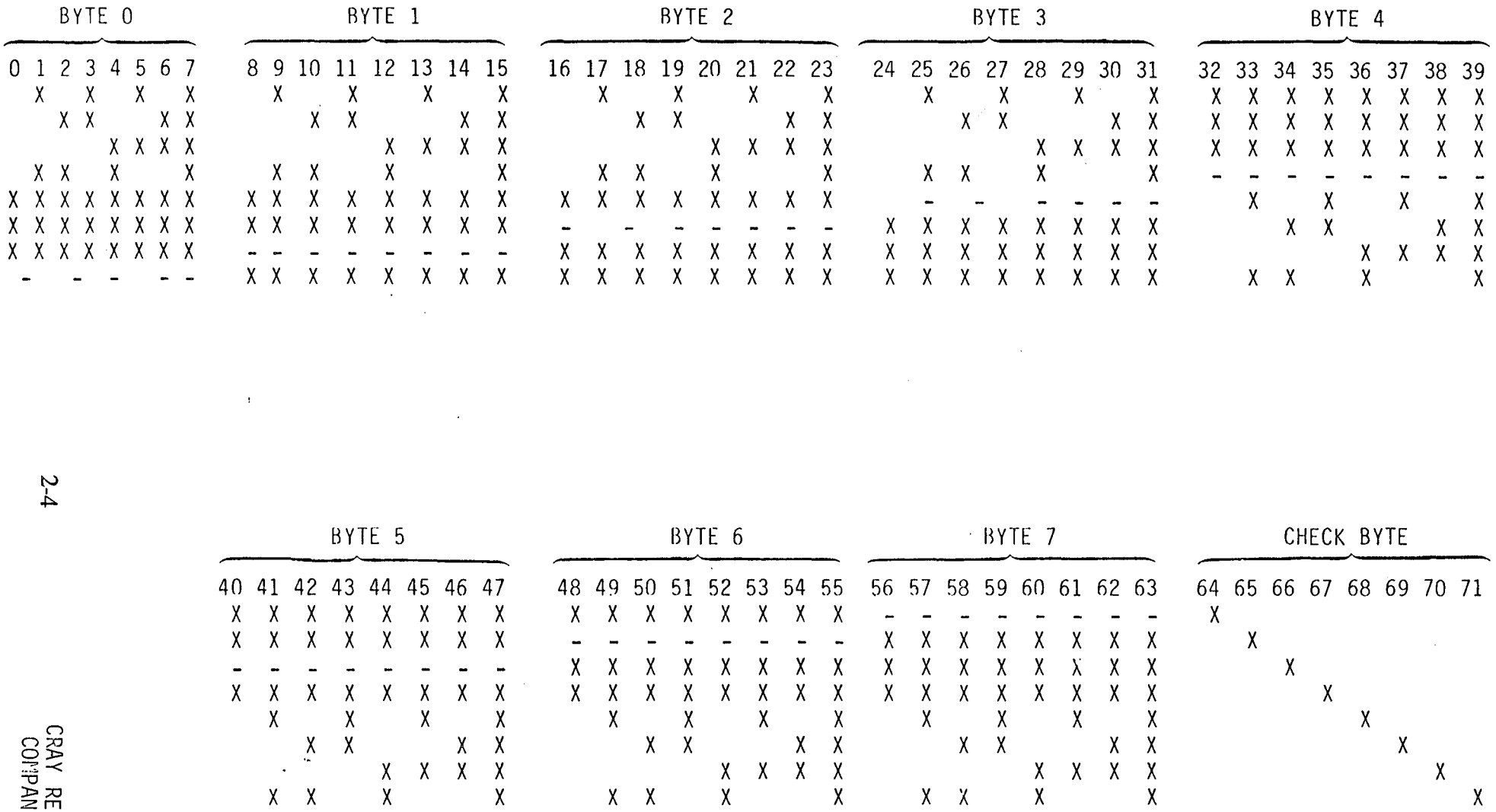
4. If more than one syndrome bit is a 1 and the parity of all syndrome bits (S0 - S7) is odd, and there are 3 ones and a zero in one half of the syndrome code, then a single and correctable error occurred. The syndrome bits can be decoded to identify the 8-bit byte containing the error as well as the altered bit in that byte.

5. If more than two errors occurred, the results are unpredictable.

---

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

†† Syndrome: Any set of characteristics regarded as identifying a certain type, condition, etc. Websters New World Dictionary.



ERROR CORRECTION CODE

A-0339

0-OK	40-69	100-70	140-D	200-71	240-D	300-D	340-24
1-64	41-D	101-D	141-M	201-D	241-M	301-M	341-D
2-65	42-D	102-D	142-M	202-D	242-M	302-M	342-D
3-D	43-M	103-M	143-D	203-M	243-D	303-D	343-27
4-66	44-D	104-D	144-M	204-D	244-M	304-M	344-D
5-D	45-M	105-M	145-D	205-M	245-D	305-D	345-29
6-D	46-M	106-M	146-D	206-M	246-D	306-D	346-30
7-32	47-D	107-D	147-38	207-D	247-34	307-36	347-D
10-67	50-D	110-D	150-M	210-D	250-M	310-M	350-D
11-D	51-M	111-M	151-M	211-M	251-D	311-D	351-25
12-D	52-M	112-M	152-D	212-M	252-D	312-D	352-26
13-40	53-D	113-D	153-46	213-D	253-42	313-44	353-D
14-D	54-M	114-M	154-D	214-M	254-D	314-D	354-28
15-48	55-D	115-D	155-54	215-D	255-50	315-52	355-D
16-56	56-D	116-D	156-62	216-D	256-58	316-60	356-D
17-D	57-M	117-M	157-D	217-M	257-D	317-D	357-31
20-68	60-D	120-D	160-00	220-D	260-08	320-16	360-D
21-D	61-M	121-M	161-D	221-M	261-D	321-D	361-M
22-D	62-M	122-M	162-D	222-M	262-D	322-D	362-M
23-M	63-D	123-D	163-03	223-D	263-11	323-19	363-D
24-D	64-M	124-M	164-D	224-M	264-D	324-D	364-M
25-M	65-D	125-D	165-05	225-D	265-13	325-21	365-D
26-M	66-D	126-D	166-06	226-D	266-14	326-22	366-D
27-D	67-35	127-37	167-D	227-33	267-D	327-D	367-39
30-D	70-M	130-M	170-D	230-M	270-D	330-D	370-M
31-M	71-D	131-D	171-01	231-D	271-09	331-17	371-D
32-M	72-D	132-D	172-02	232-D	272-10	332-18	372-D
33-D	73-43	133-45	173-D	233-41	273-D	333-D	373-47
34-M	74-D	134-D	174-04	234-D	274-12	334-20	374-D
35-D	75-51	135-53	175-D	235-49	275-D	335-D	375-55
36-D	76-59	136-61	176-D	236-57	276-D	336-D	376-63
37-M	77-D	137-D	177-07	237-D	277-15	337-23	377-D

D - DOUBLE ERROR
M - MULTIPLE ERROR

X2002C0101

MEMORY REFERENCE INSTRUCTIONS

SCALAR REFERENCE

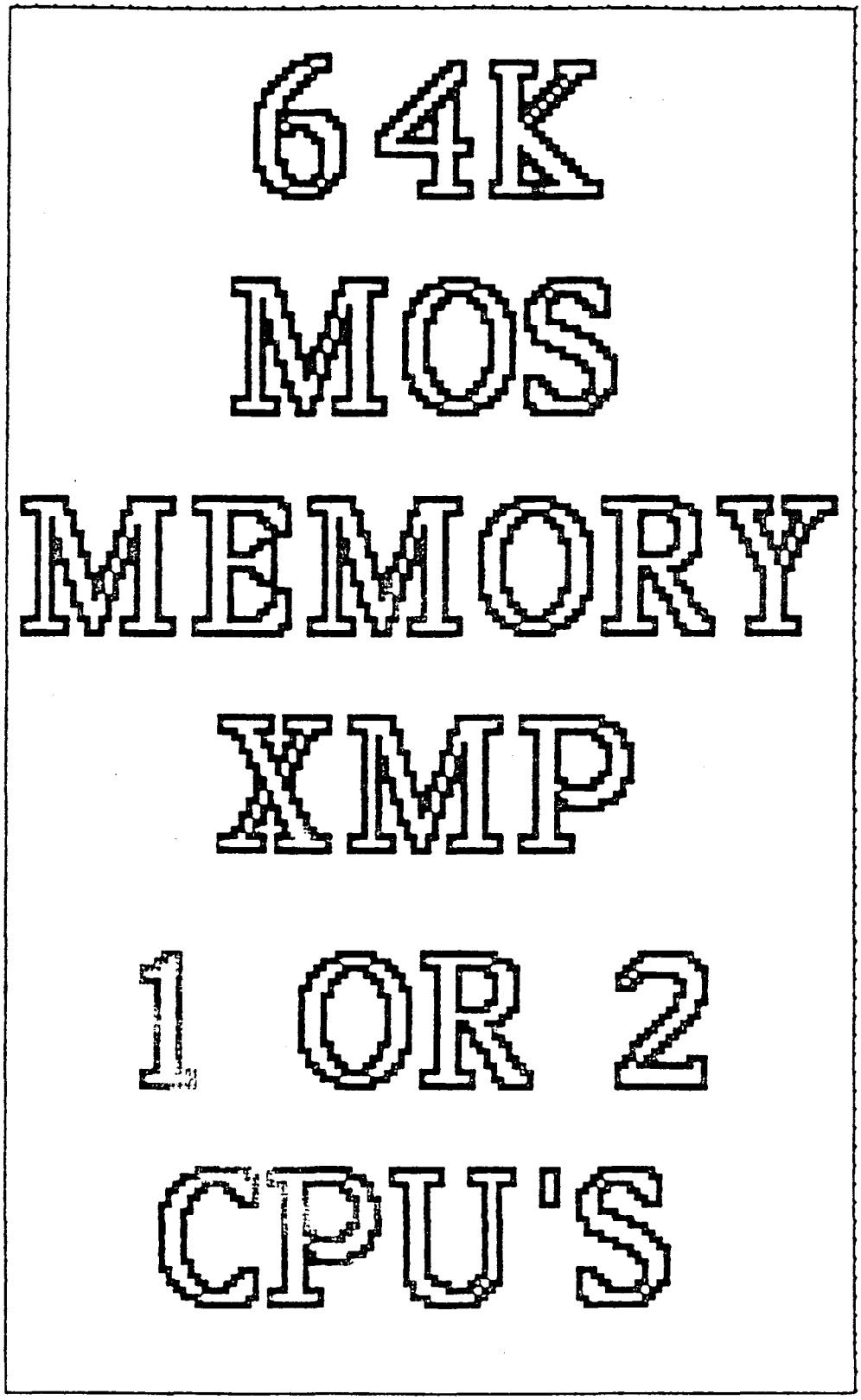
10hijkm	-	Read from memory address specified by ((Ah) + exp) to Ai (Ao = o)
100ijkm	-	Read from memory address specified by exp to Ai
10hi000	-	Read from memory address specified by (Ah) to Ai
11hijkm	-	Write (Ai) to memory address specified by ((Ah) + exp) (Ao = o)
110ijkm	-	Write (Ai) to memory address specified by exp
11hi000	-	Write (Ai) to memory address specified by (Ah)
12hijkm	-	Read from memory address specified by ((Ah) + exp) to Si (Ao = o)
120ijkm	-	Read from memory address specified by exp to Si
12hi000	-	Read from memory address specified by (Ah) to Si
13hijkm	-	Write (Si) to memory address specified by ((Ah) + exp) (Ao = o)
130ijkm	-	Write (Si) to memory address specified by exp
13hi000	-	Write (Si) to memory address specified by (Ah)

B/T REGISTER BLOCK REFERENCE

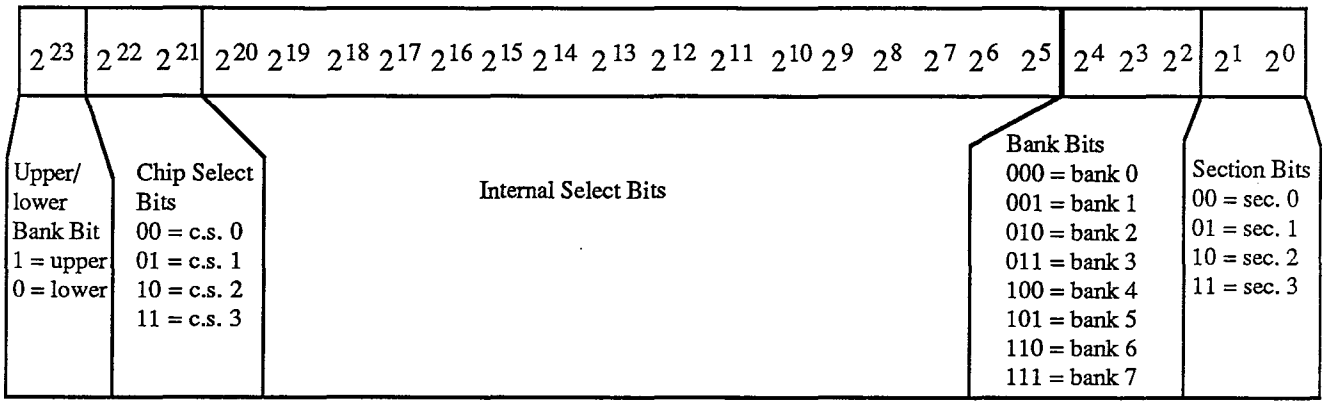
034ijk	-	Read (Ai) words from STARTING memory address (Ao) to Bjk
035ijk	-	Write (Ai) words to STARTING memory address (Ao) from Bjk
036ijk	-	Read (Ai) words from STARTING memory address (Ao) to Tjk
037ijk	-	Write (Ai) words to STARTING memory address (Ao) from Tjk

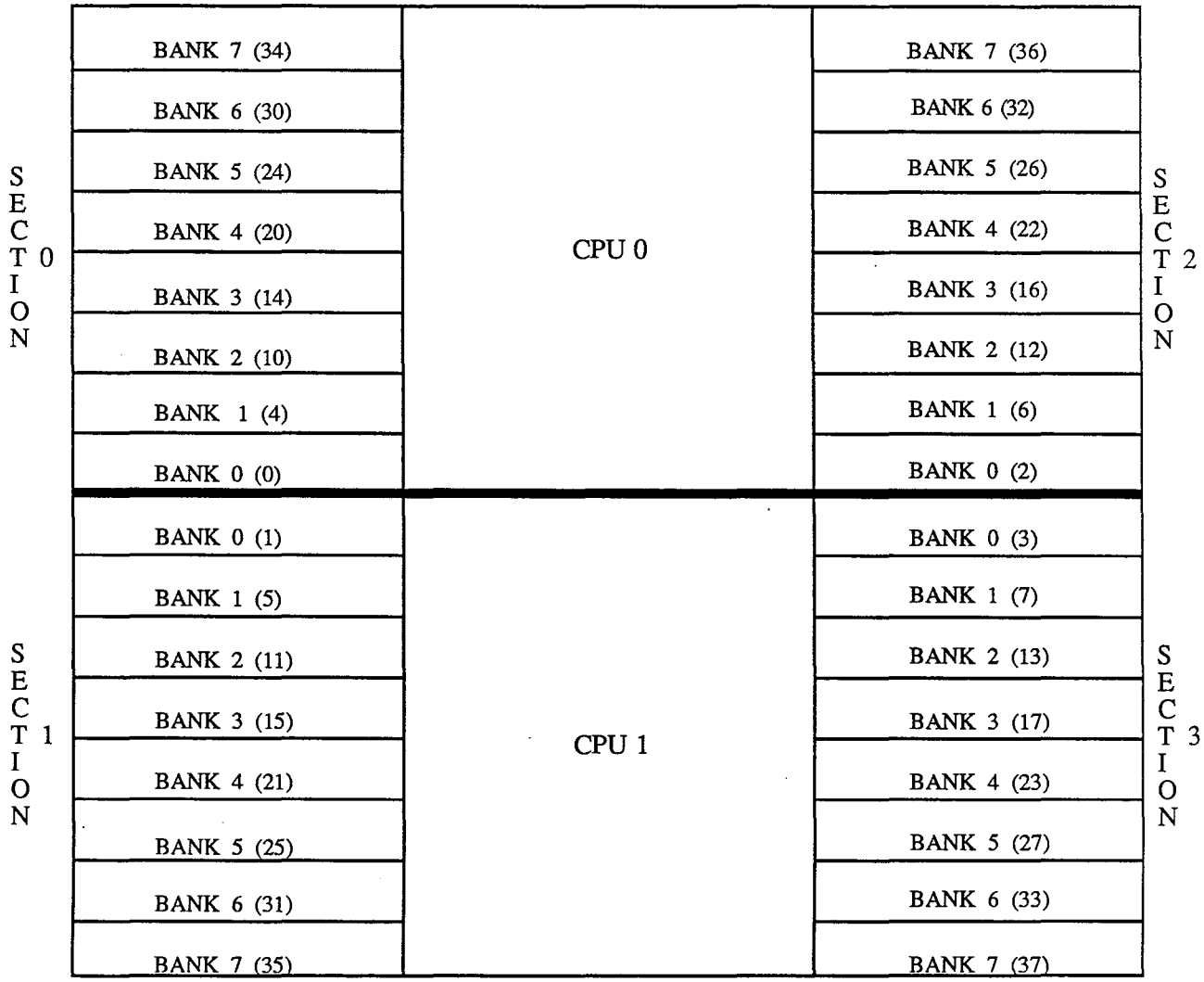
VECTOR REGISTER BLOCK REFERENCE

176i0k	-	Read (VL) words into Vi from memory address specified by (Ao) incremented by (Ak)
176i00	-	Read (VL) words into Vi from memory address specified by (Ao) incremented by 1
176ilk	-	Read (VL) words from memory address specified by ((Ao) + (Vk)) to Vi (Gather)
1770jk	-	Write (VL) words from Vj to memory address specified by (Ao) incremented by (Ak)
1770j0	-	Write (VL) words from Vj to memory address specified by (Ao) incremented by 1
177ljk	-	Write (VL) words from Vj to memory address specified by ((Ao) + (Vk)) (Scatter)

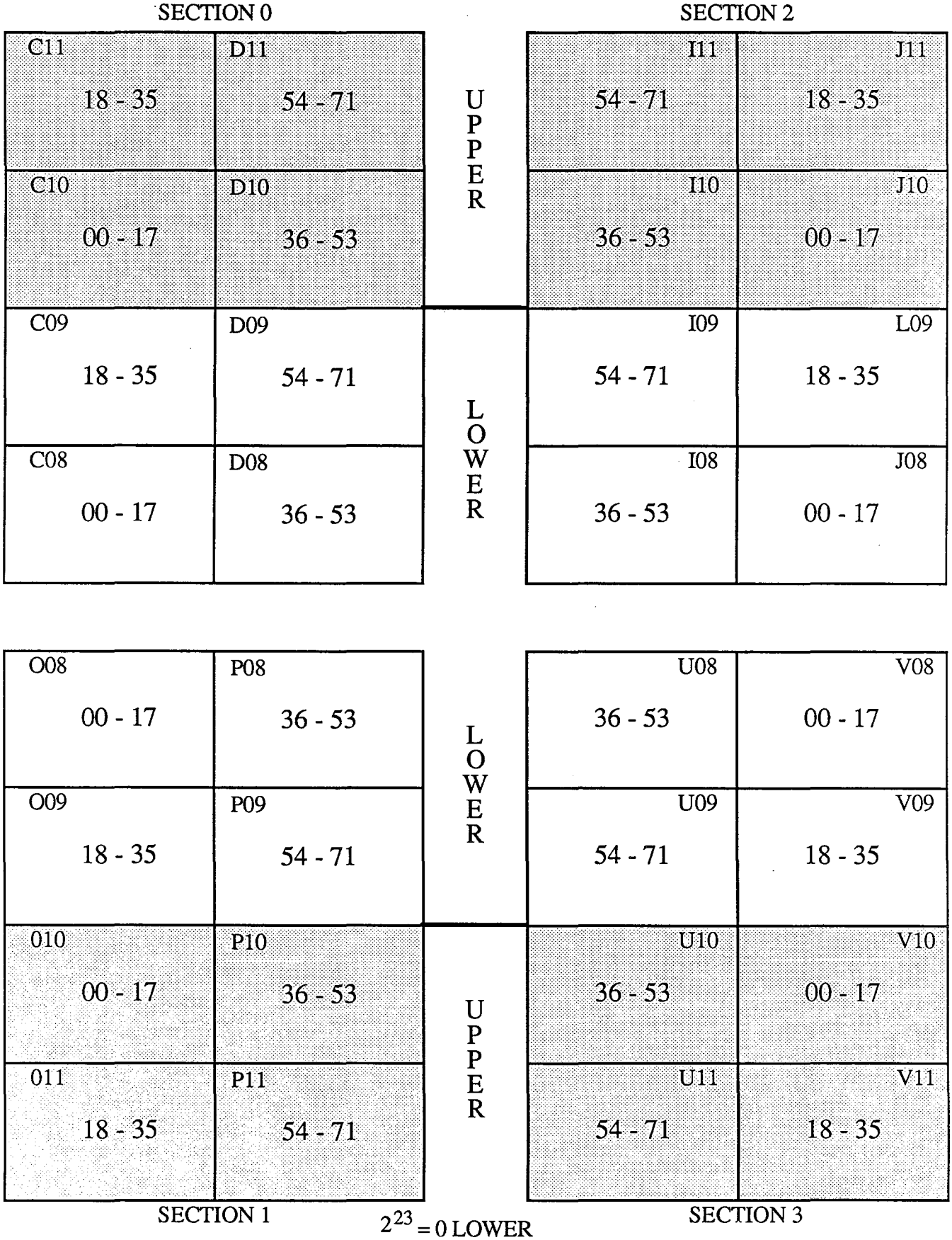

64K
MOS
MEMORY
XMP
1 OR 2
CPU'S

16 MILLION WORD MEMORY ADDRESSING





Hardware Trng.
A-5869 R.W.A.

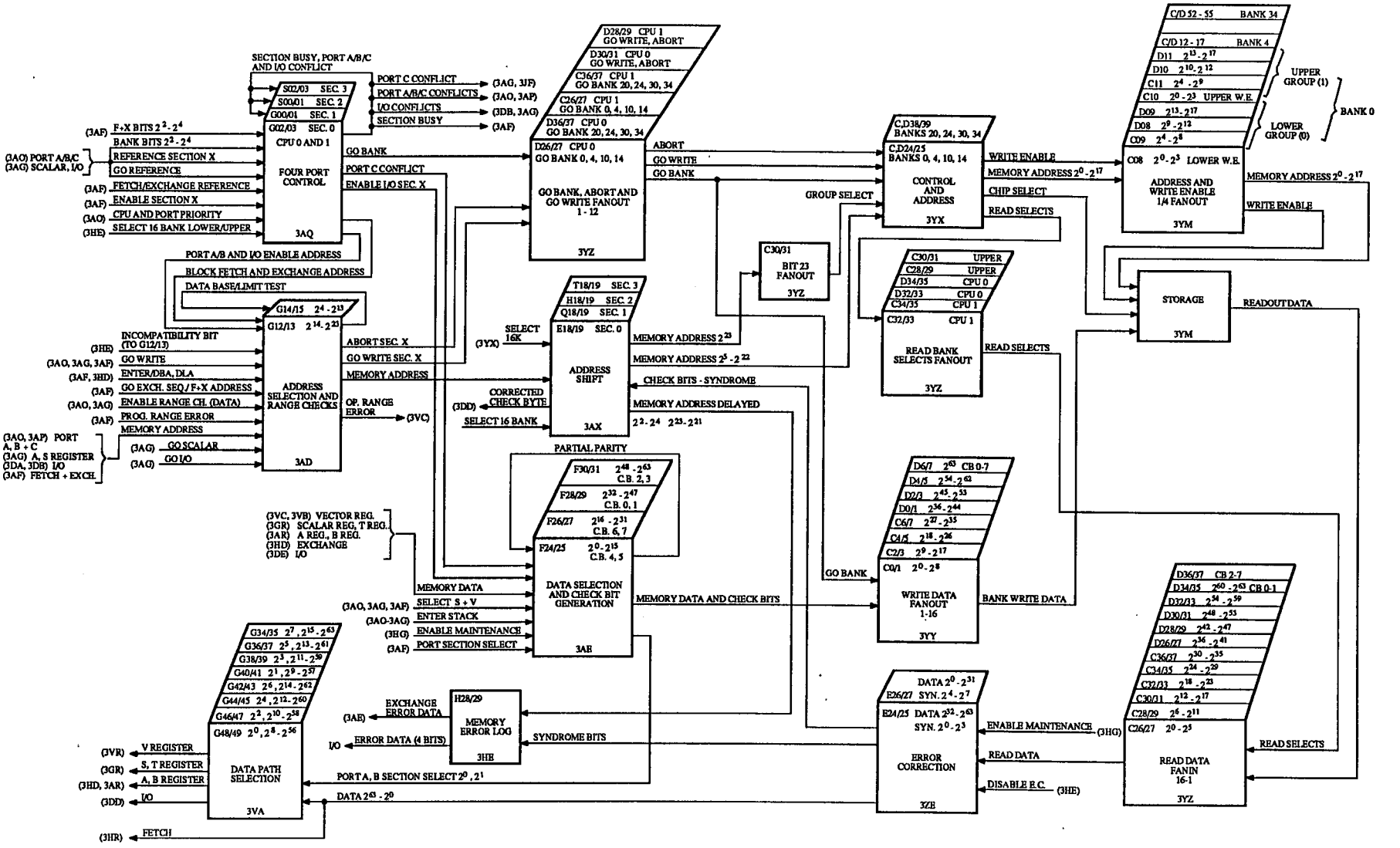


2^{23} = 0 LOWER

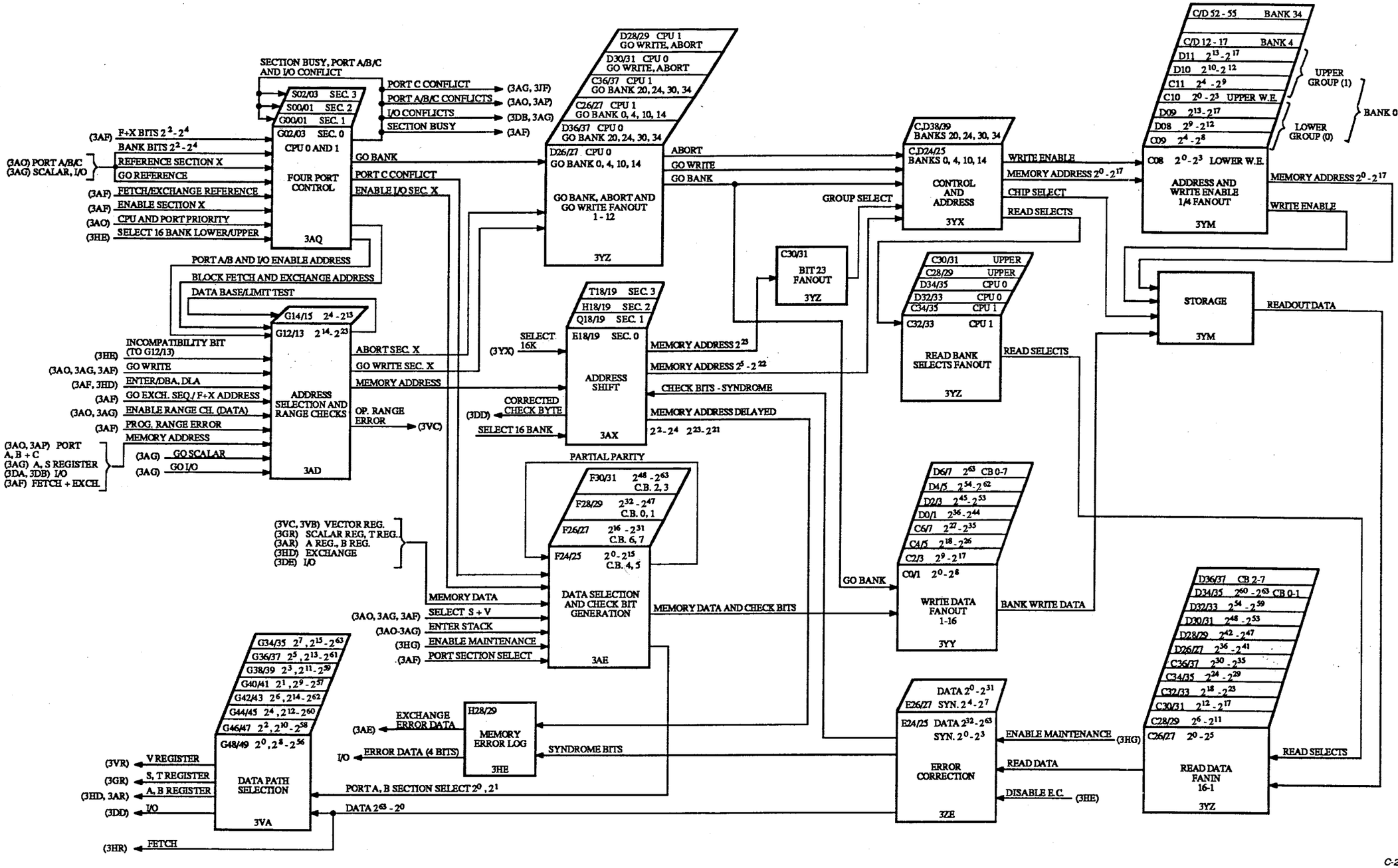
2^{23} = 1 UPPER

Hardware Trng.
A-5868 R.W.A.

CRAY X-MP 216 BIT LAYOUT

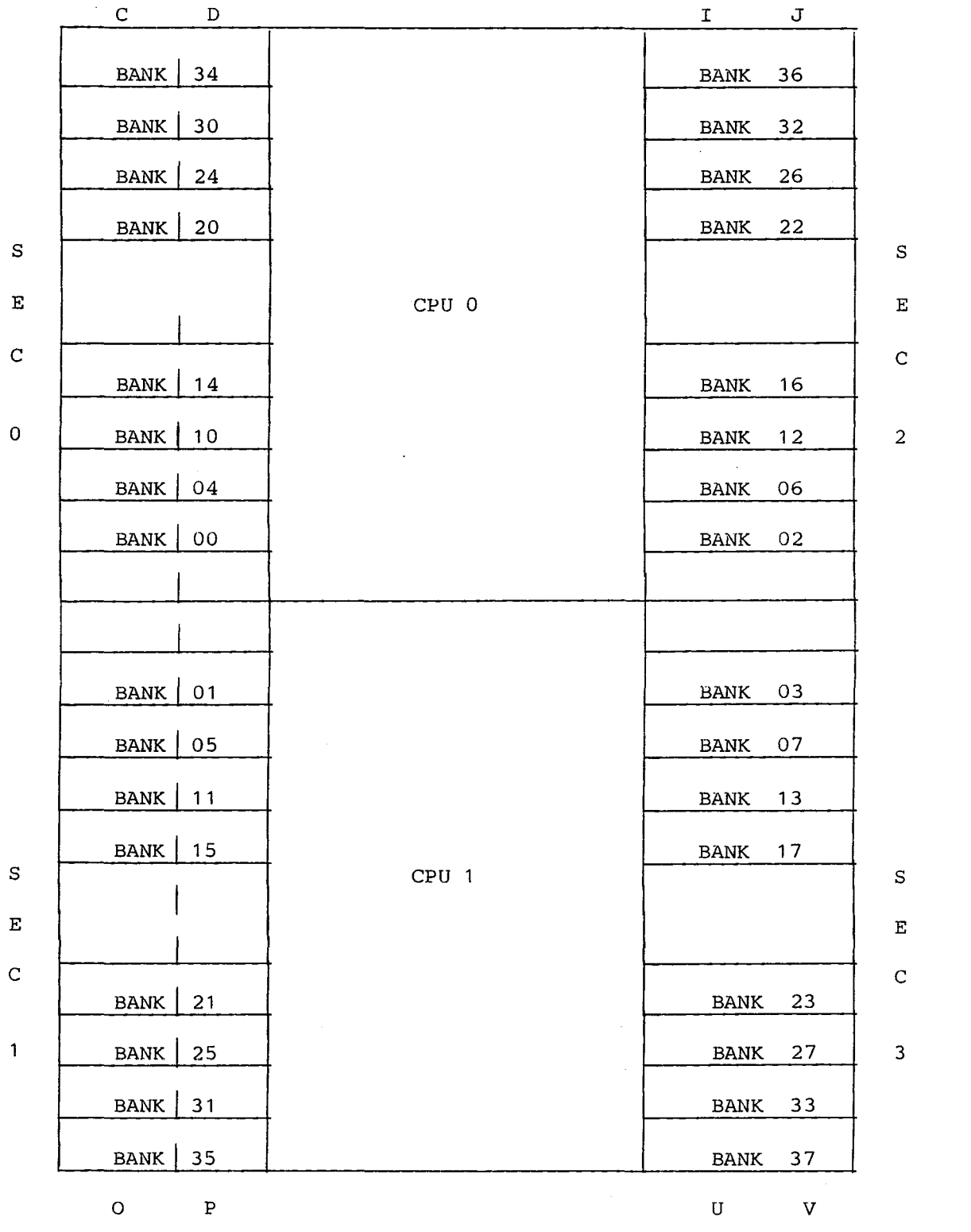


CRAY X-MP/2 16 MILLION WORD MEMORY BLOCK DIAGRAM
(CPU 0 AND SECTION 0)

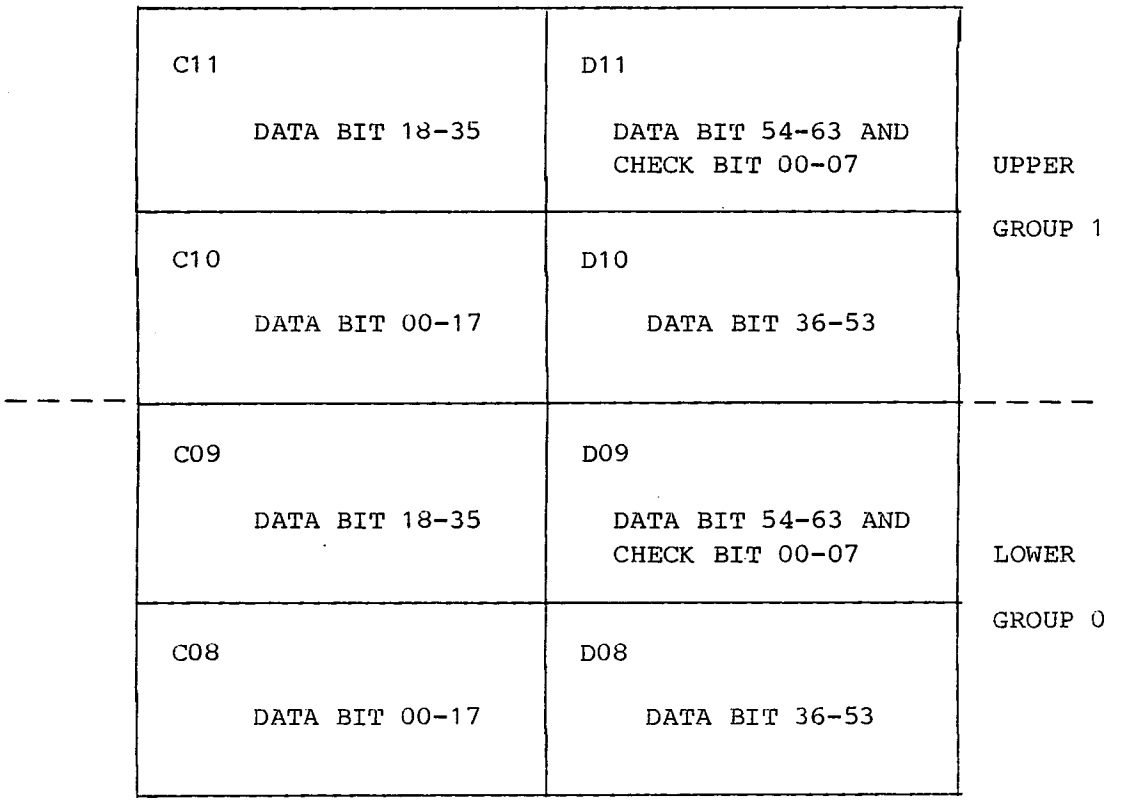


C-2994A

CRAY X-MP/2 16 MILLION WORD MEMORY BLOCK DIAGRAM
(CPU 0 AND SECTION 0)

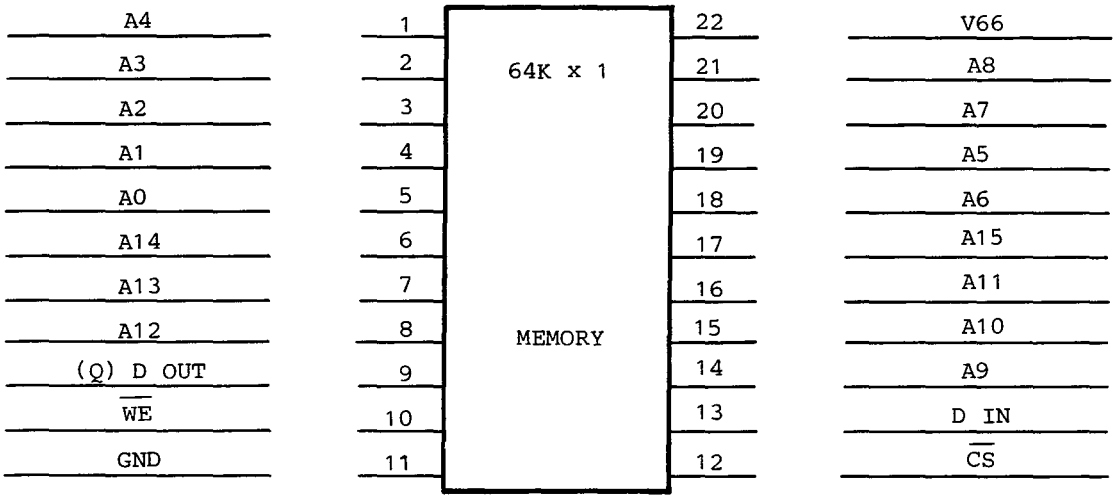

C
D
I
J
BANK 34
BANK 30
BANK 24
BANK 20
BANK 36
BANK 32
BANK 26
BANK 22
SECC 0
SECC 2
CPU 0
BANK 14
BANK 10
BANK 04
BANK 00
BANK 16
BANK 12
BANK 06
BANK 02
BANK 01
BANK 05
BANK 11
BANK 15
BANK 03
BANK 07
BANK 13
BANK 17
SECC 1
SECC 3
CPU 1
BANK 21
BANK 25
BANK 31
BANK 35
BANK 23
BANK 27
BANK 33
BANK 37
O
P
U
V

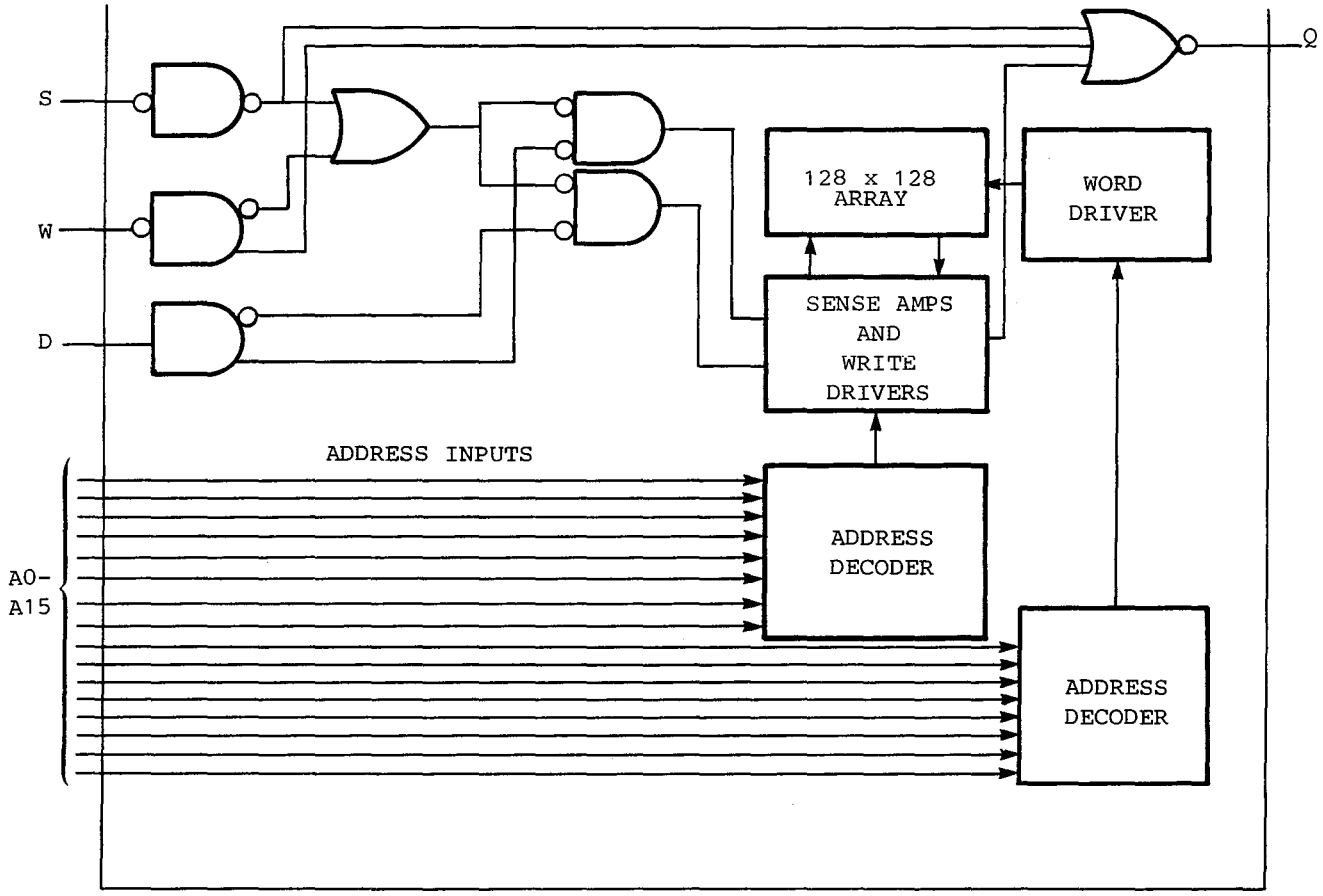
X4802C0601



X4802C0602

SECTION 0 BANK 0





MACRO DEFINITION

Q = D IN: DCD(A-0)KS + WE

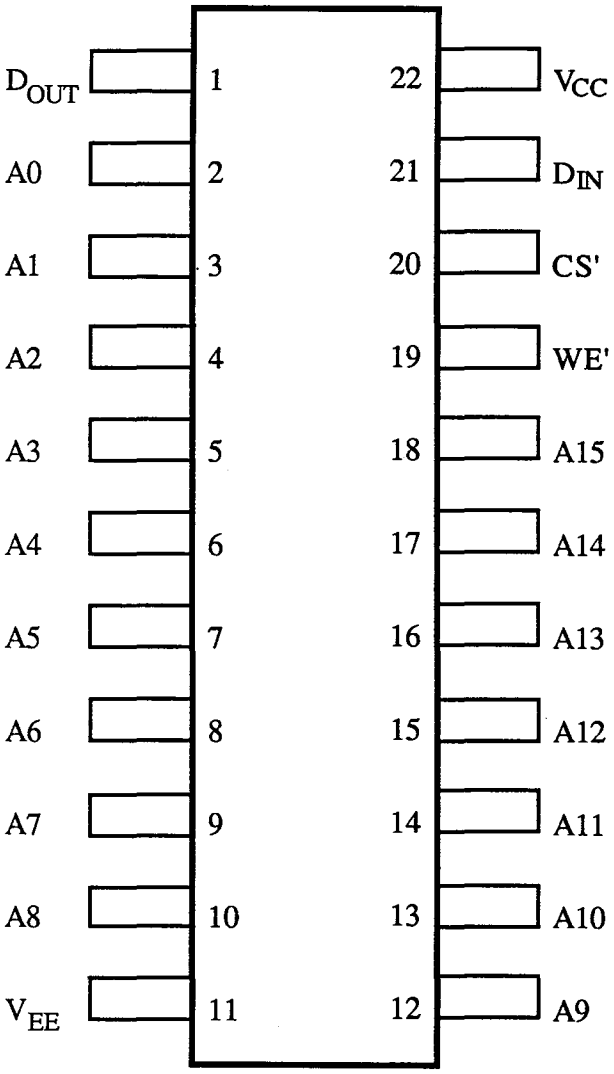
Q = D IN: DCD(A15 A14 A13 A12 A11 A10 A9 A8 A7 A6 A5 A4 A3 A2 A1 A0)/CS + WE

9 = 13 17 6 7 8 16 15 14 21 20 18 19 1 2 3 4 5 12 10

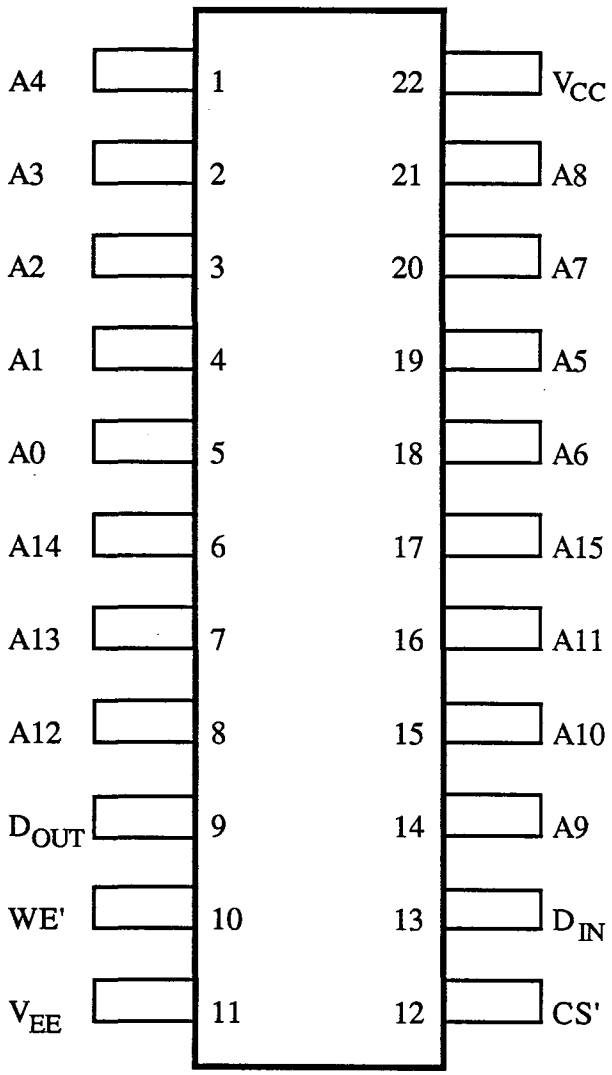
POWER DISSIPATION:	-5.2 V.	-5.2 V.	LOADING
	780 MW.	14 MW.	794 MWT

NAME: 64K x 1 MEMORY (STATIC MOS RAM)

A-2988



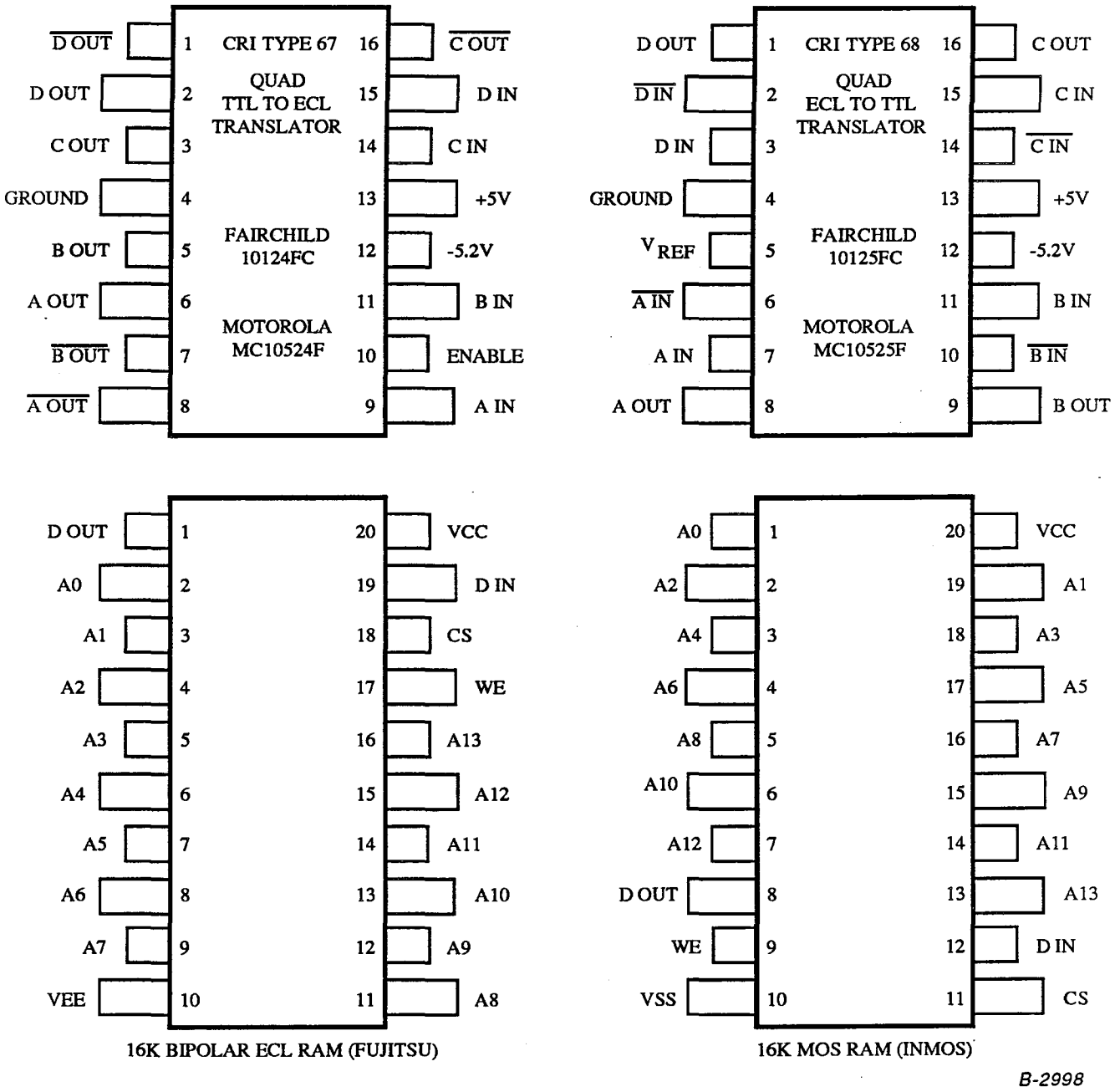
64K x 1 MEMORY
(ECL RAM)



64K x 1 MEMORY
(MOS RAM)

Hardware Trng.
A-6359 G.K.F.

CRAY X-MP/-1, -2, -4 MEMORY CHIPS



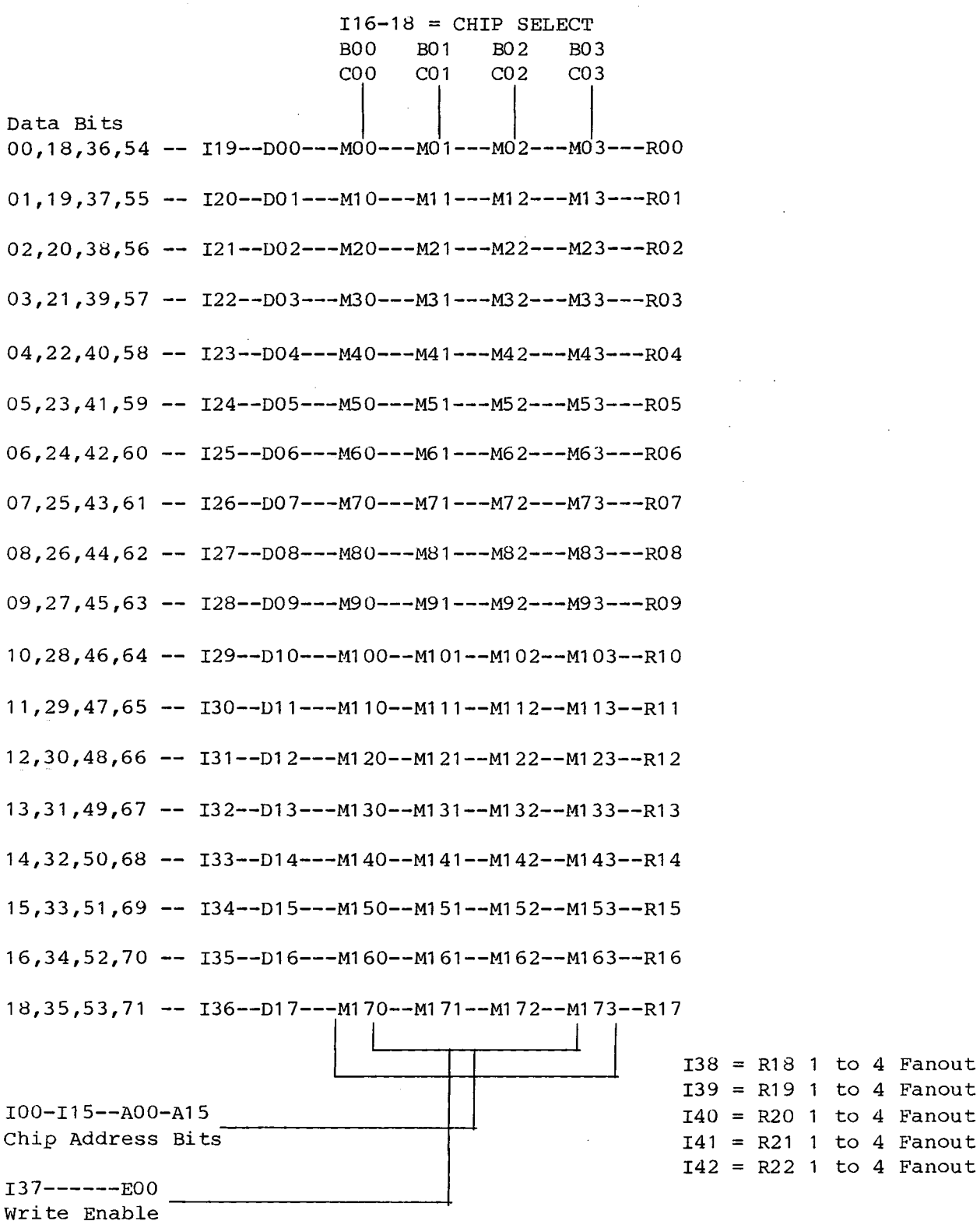
16K BIPOLAR ECL RAM (FUJITSU)

16K MOS RAM (INMOS)

B-2998

CRAY X-MP/-1, -2, -4 MEMORY CHIPS

3YM MODULE CHIP CHART



X4802C0603

Lower and upper refer to bit 23. If address bit 23 = 0 it's Lower Memory and if bit 23 = 1 it's Upper Memory.

DATA BITS	BANK (Octal)	LOWER MEMORY	UPPER MEMORY
00-17	0	C08	C10
18-35	0	C09	C11
36-53	0	D08	D10
54-71	0	D09	D11
00-17	1	O08	O10
18-35	1	O09	O11
36-53	1	P08	P10
54-71	1	P09	P11
00-17	2	J08	J10
18-35	2	J09	J11
36-53	2	I08	I10
54-71	2	I09	I11
00-17	3	V08	V10
18-35	3	V09	V11
36-53	3	U08	U10
54-71	3	U09	U11
00-17	4	C12	C14
18-35	4	C13	C15
36-53	4	D12	D14
54-71	4	D13	D15
00-17	5	O12	O14
18-35	5	O13	O15
36-53	5	P12	P14
54-71	5	P13	P15
00-17	6	J12	J14
18-35	6	J13	J15
36-53	6	I12	I14
54-71	6	I13	I15
00-17	7	V12	V14
18-35	7	V13	V15
36-53	7	U12	U14
54-71	7	U13	U15

X4802C0604

3YM MEMORY LOCATOR

DATA BITS	BANK (Octal)	LOWER MEMORY	UPPER MEMORY
00-17	10	C16	C18
18-35	10	C17	C19
36-53	10	D16	D18
54-71	10	D17	D19
00-17	11	O16	O18
18-35	11	O17	O19
36-53	11	P16	P18
54-71	11	P18	P19
00-17	12	J16	J18
18-35	12	J17	J19
36-53	12	I16	I18
54-71	12	I17	I19
00-17	13	V16	V18
18-35	13	V17	V19
36-53	13	U16	U18
54-71	13	U17	U19
00-17	14	C20	C22
18-35	14	C21	C23
36-53	14	D20	D22
54-71	14	D21	D23
00-17	15	O20	O22
18-35	15	O21	O23
36-53	15	P20	P22
54-71	15	P21	P23
00-17	16	J20	J22
18-35	16	J21	J23
36-53	16	I20	I22
54-71	16	I21	I23
00-17	17	V20	V22
18-35	17	V21	V23
36-53	17	U20	U22
54-71	17	U21	U23

X4802C0605

3YM MEMORY LOCATOR

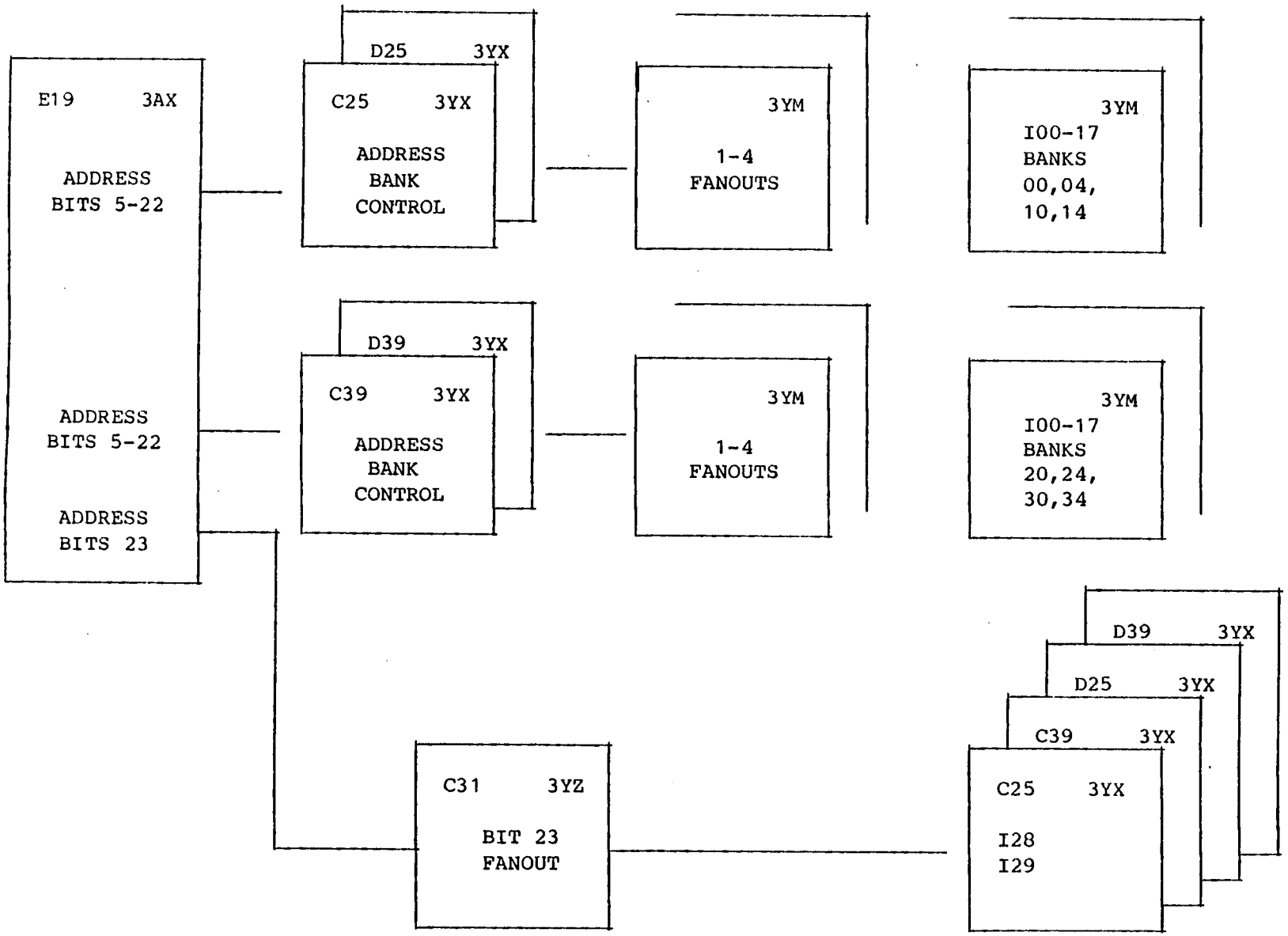
DATA BITS	BANK (Octal)	LOWER MEMORY	UPPER MEMORY
00-17	20	C40	C42
18-35	20	C41	C43
36-53	20	D40	D42
54-71	20	D41	D43
00-17	21	O40	O42
18-35	21	O41	O43
36-53	21	P40	P42
54-71	21	P41	P43
00-17	22	J40	J42
18-35	22	J41	J43
36-53	22	I40	I42
54-71	22	I41	I43
00-17	23	V40	V42
18-35	23	V41	V43
36-53	23	U40	U42
54-71	23	U41	U43
00-17	24	C44	C46
18-35	24	C45	C47
36-53	24	D44	D46
54-71	24	D45	D47
00-17	25	O44	O46
18-35	25	O45	O47
36-53	25	P44	P46
54-71	25	P45	P47
00-17	26	J44	J46
18-35	26	J45	J47
36-53	26	I44	I46
54-71	26	I45	I47
00-17	27	V44	V46
18-35	27	V45	V47
36-53	27	U44	U46
54-71	27	U45	U47

X4802C0606

3YM MEMORY LOCATOR

DATA BITS	BANK (Octal)	LOWER MEMORY	UPPER MEMORY
00-17	30	C48	C50
18-35	30	C49	C51
36-53	30	D48	D50
54-71	30	D49	D51
00-17	31	O48	O50
18-35	31	O49	O51
36-53	31	P48	P50
54-71	31	P49	P51
00-17	32	J48	J50
18-35	32	J49	J51
36-53	32	I48	I50
54-71	32	I49	I51
00-17	33	V48	V50
18-35	33	V49	V51
36-53	33	U48	U50
54-71	33	U49	U51
00-17	34	C52	C54
18-35	34	C53	C55
36-53	34	D52	D54
54-71	34	D53	D55
00-17	35	O52	O54
18-35	35	O53	O55
36-53	35	P52	P54
54-71	35	P53	P55
00-17	36	J52	J54
18-35	36	J53	J55
36-53	36	I52	I54
54-71	36	I53	I55
00-17	37	V52	V54
18-35	37	V53	V55
36-53	37	U52	U54
54-71	37	U53	U55

X4802C0607


E19 3AX
ADDRESS BITS 5-22
ADDRESS BITS 5-22
ADDRESS BITS 23
D25 3YX
C25 3YX
ADDRESS BANK CONTROL
3YM
1-4 FANOUTS
3YM
I00-17 BANKS 00,04, 10,14
D39 3YX
C39 3YX
ADDRESS BANK CONTROL
3YM
1-4 FANOUTS
3YM
I00-17 BANKS 20,24, 30,34
C31 3YZ
BIT 23 FANOUT
D39 3YX
D25 3YX
C39 3YX
C25 3YX
I28
I29

X4802C0608

LOC.	TERM	DESCRIPTION	LOC.	TERM	DESCRIPTION
C27	R12	CPU 1 GO BANK 00, 10	D27	R12	CPU 0 GO BANK 00, 10
C27	R13	CPU 1 GO BANK 04, 14	D27	R13	CPU 0 GO BANK 04, 14
C27	R14	CPU 1 GO BANK 10, 14	D27	R14	CPU 0 GO BANK 10, 14
C27	R15	CPU 1 GO BANK 00, 04	D27	R15	CPU 0 GO BANK 00, 04
C29	R12	CPU 0 GO READ BANK 20, 24 + 30, 34 UPPER	D29	R12	CPU 1 GO WRITE
C29	R13	CPU 1 GO READ BANK 20, 24 + 30, 34 UPPER	D29	R13	CPU 1 ABORT
C29	R14	CPU 1 GO READ BANK 00, 04 + 10, 14 UPPER	D29	R14	NOT USED
C29	R15	NOT USED	D29	R15	NOT USED
C31	R12	CPU 0 ADDRESS BIT 23	D31	R12	CPU 0 GO WRITE
C31	R13	CPU 1 ADDRESS BIT 23	D31	R13	CPU 0 ABORT
C31	R14	CPU 0 GO READ BANK 00, 04 + 10, 14 UPPER	D31	R14	NOT USED
C31	R15	NOT USED	D31	R15	NOT USED
C33	R12	CPU 1 GO READ BANK 00, 10	D33	R12	CPU 0 GO READ BANK 00, 10
C33	R13	CPU 1 GO READ BANK 04, 14	D33	R13	CPU 0 GO READ BANK 04, 1
C33	R14	CPU 1 GO READ BANK 10, 14	D33	R14	CPU 0 GO READ BANK 10, 14
C33	R15	NOT USED	D33	R15	NOT USED
C35	R12	CPU 1 GO READ BANK 20, 30	D35	R12	CPU 0 GO READ BANK 20, 30
C35	R13	CPU 1 GO READ BANK 24, 34	D35	R13	CPU 0 GO READ BANK 24, 34
C35	R14	CPU 1 GO READ BANK 30, 34	D35	R14	CPU 0 GO READ BANK 30, 34
C35	R15	NOT USED	D35	R15	NOT USED
C37	R12	CPU 1 GO BANK 20, 30	D37	R12	CPU 0 GO BANK 20, 30
C37	R13	CPU 1 GO BANK 24, 34	D37	R13	CPU 0 GO BANK 24, 34
C37	R14	CPU 1 GO BANK 30, 34	D37	R14	CPU 0 GO BANK 30, 34
C37	R15	NOT USED	D37	R15	CPU 0 GO BANK 20, 24

X4802C0609

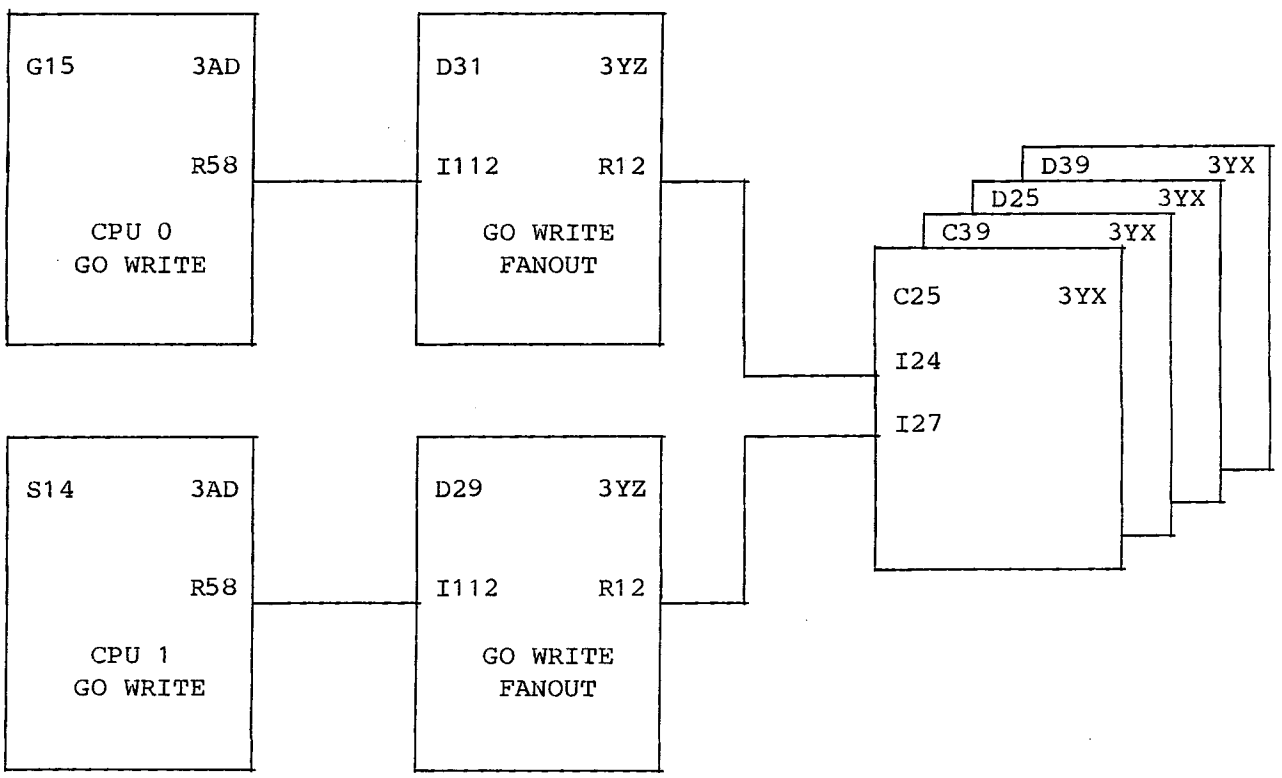
1-4 FANOUTS - 3YM MODULES - SECTION 0
ADDRESS AND WRITE ENABLE FANOUTS

			I38	I39	I40	I41	I42		I38	I39	I40	I41	I42
B		C23	04	05	06	07	08	D23	13	14	15	16	17
A	14	C22	00	01	02	03	—	D22	09	10	11	12	WE
N		C21	04	05	06	07	08	D21	13	14	15	16	17
K		C20	00	01	02	03	—	D20	09	10	11	12	WE
B		C19	04	05	06	07	08	D19	13	14	15	16	17
A	10	C18	00	01	02	03	—	D18	09	10	11	12	WE
N		C17	04	05	06	07	08	D17	13	14	15	16	17
K		C16	00	01	02	03	—	D16	09	10	11	12	WE
B		C15	04	05	06	07	08	D15	13	14	15	16	17
A	4	C14	00	01	02	03	WE	D14	09	10	11	12	—
N		C13	04	05	06	07	08	D13	13	14	15	16	17
K		C12	00	01	02	03	WE	D12	09	10	11	12	—
B		C11	04	05	06	07	08	D11	13	14	15	16	17
A	0	C10	00	01	02	03	WE	D10	09	10	11	12	—
N		C09	04	05	06	07	08	D09	13	14	15	16	17
K		C08	00	01	02	03	WE	D08	09	10	11	12	—

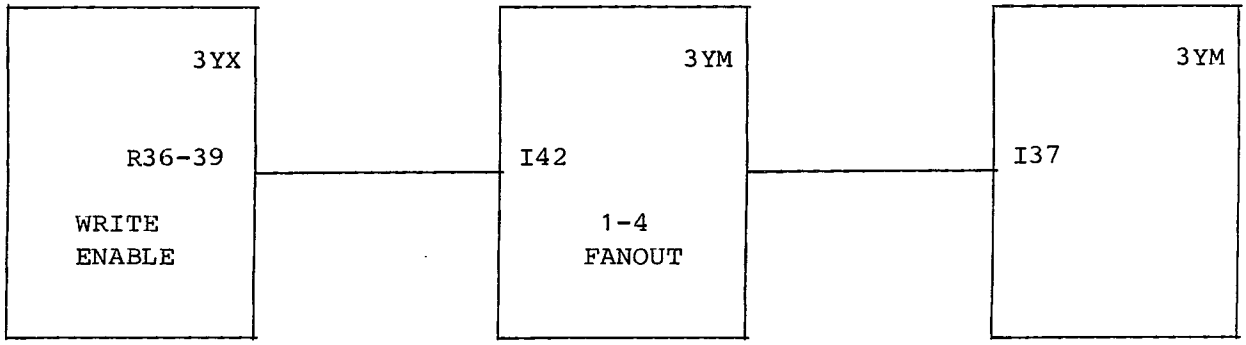
X4802C0610

Location

D38/39	-	Address Bits 2^{14} - 2^{21} for Banks 20, 24, 30, 34
	-	Chip Select Enables for Banks 30 or 34
	-	Upper or Lower Write Enables for Bank 30 or 34
	-	Go Read Bank 30 or 34
	-	Go Read Upper
C38/39	-	Address Bits 2^{5} - 2^{13} for Banks 20, 24, 30, 34
	-	Chip Select Enables for Banks 20 or 24
	-	Upper or Lower Write Enables for Bank 20 or 24
	-	Go Read Bank 20 or 24
	-	Go Read Upper
D24/25	-	Address Bits 2^{14} - 2^{21} for Banks 0, 4, 10, 14
	-	Chip Select Enables for Bank 10 or 14
	-	Upper or Lower Write Enables for Bank 10 or 14
	-	Go Read Bank 10 or 14
	-	Go Read Upper
C24/25	-	Address Bits 2^{5} - 2^{13} for Banks 0, 4, 10, 14
	-	Chip Select Enables for Bank 0 or 4
	-	Upper or Lower Write Enables for Bank 0 or 4
	-	Go Read Upper


G15
3AD
R58
CPU 0
GO WRITE
D31
3YZ
I112
R12
GO WRITE
FANOUT
S14
3AD
R58
CPU 1
GO WRITE
D29
3YZ
I112
R12
GO WRITE
FANOUT
D39
3YX
D25
3YX
C39
3YX
C25
3YX
I24
I27
X4802C0611

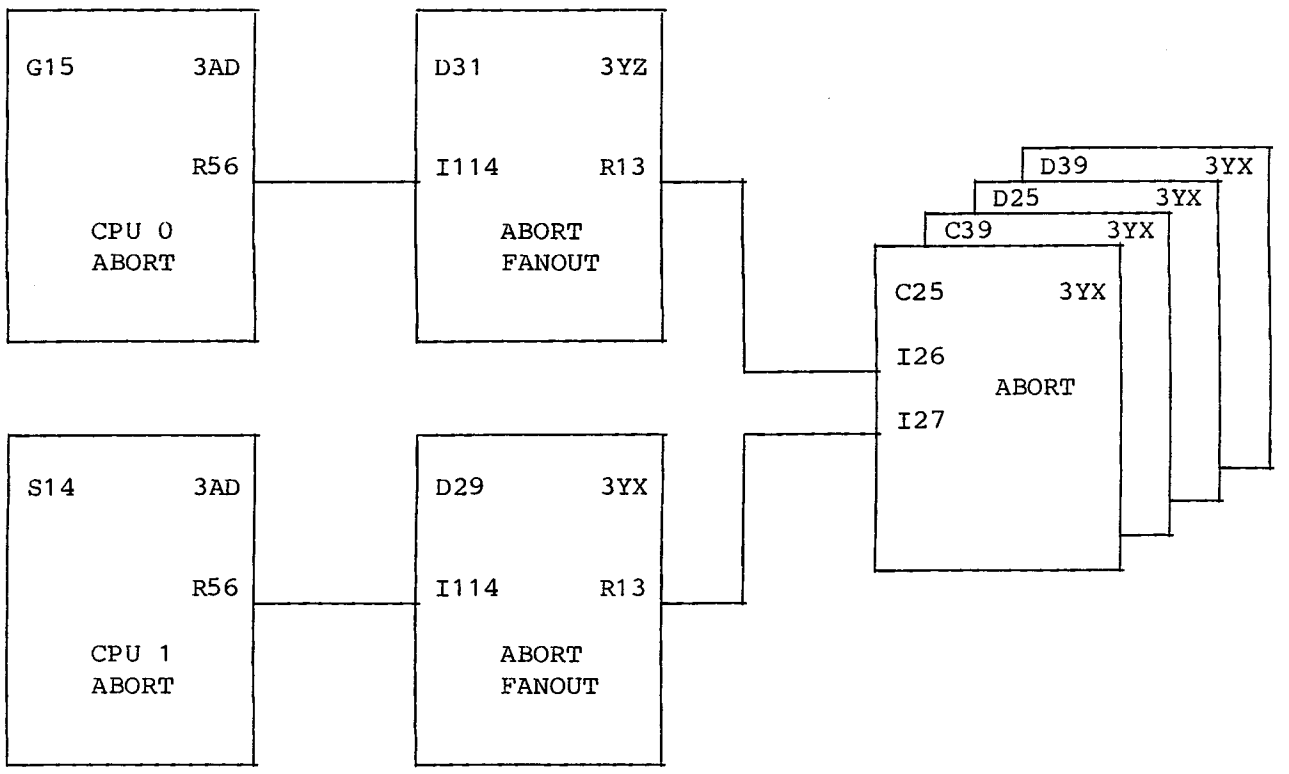
GO WRITE



C	D
20, 24	30, 34
BANK 0,4	10, 14
BANK 1,5	11, 15
21, 25	31, 35
O	P

X4802C0612

WRITE ENABLE



X4802C0613

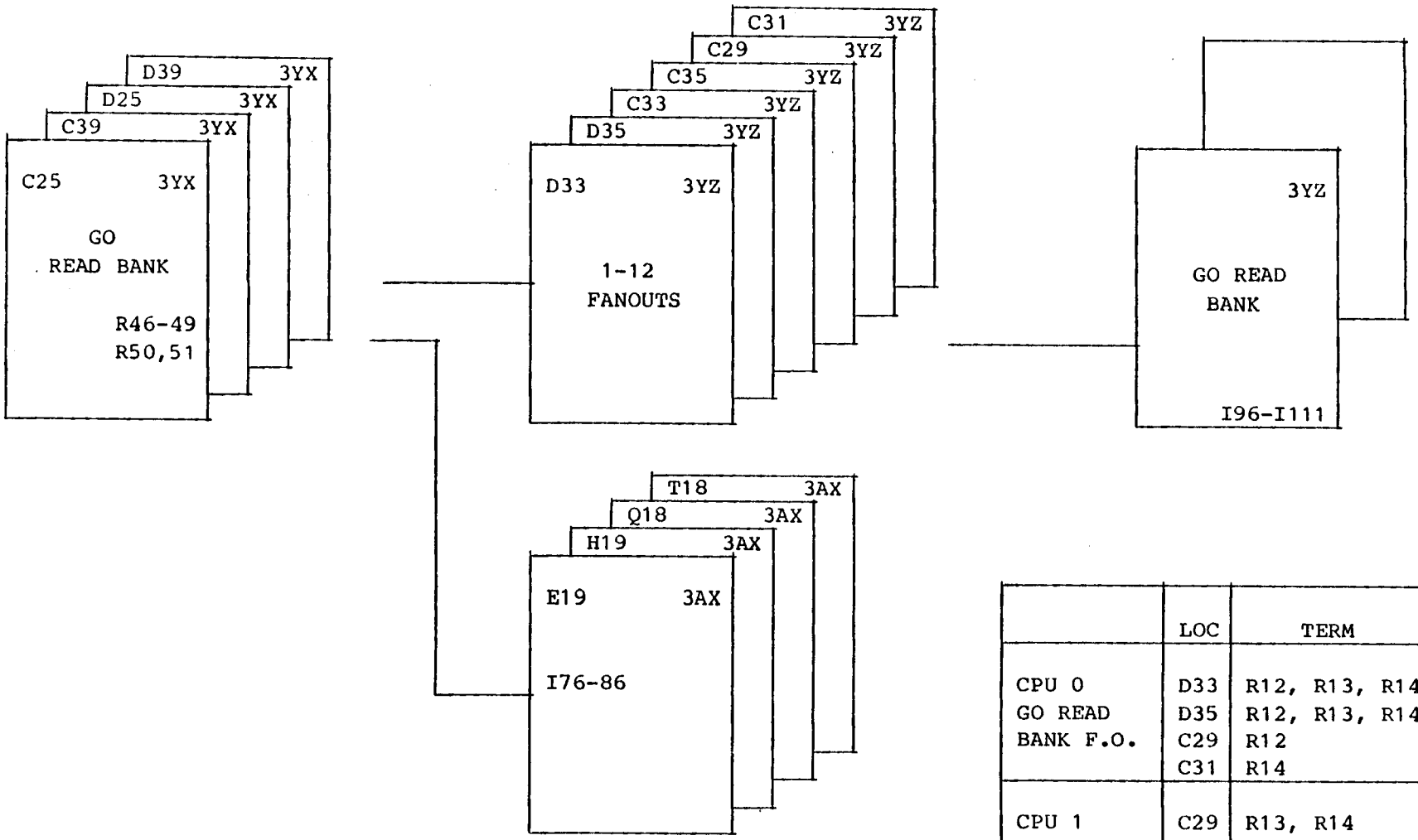
ABORT

HTV-0242

3YY BIT PLACEMENT - SECTION 0

LOCATION	DESCRIPTION
C01	WRITE DATA BIT 00-08
C03	WRITE DATA BIT 09-17
C05	WRITE DATA BIT 18-26
C07	WRITE DATA BIT 27-35
D01	WRITE DATA BIT 36-44
D03	WRITE DATA BIT 45-53
D05	WRITE DATA BIT 54-62
D07	WRITE DATA BIT 63 AND CHECK BIT 00-07

X4802C0614



	LOC	TERM
CPU 0	D33	R12, R13, R14
GO READ	D35	R12, R13, R14
BANK F.O.	C29	R12
	C31	R14
CPU 1	C29	R13, R14
GO READ	C33	R12, R13, R14
BANK F.O.	C35	R12, R13, R14

X4802C0615

GO READ BANK

3YZ BIT PLACEMENT - SECTION 0

LOCATION	DESCRIPTION
C27	READ DATA BIT 00 - 05
C29	READ DATA BITS 06 - 11
C31	READ DATA BITS 12 - 17
C33	READ DATA BITS 18 - 23
C35	READ DATA BITS 24 - 29
C37	READ DATA BITS 30 - 35
D27	READ DATA BITS 36 - 41
D29	READ DATA BITS 42 - 47
D31	READ DATA BITS 48 - 53
D33	READ DATA BITS 54 - 59
D35	READ DATA BITS 60 - 63 AND CHECK BITS 00 - 01
D37	READ DATA CHECK BITS 02 - 07

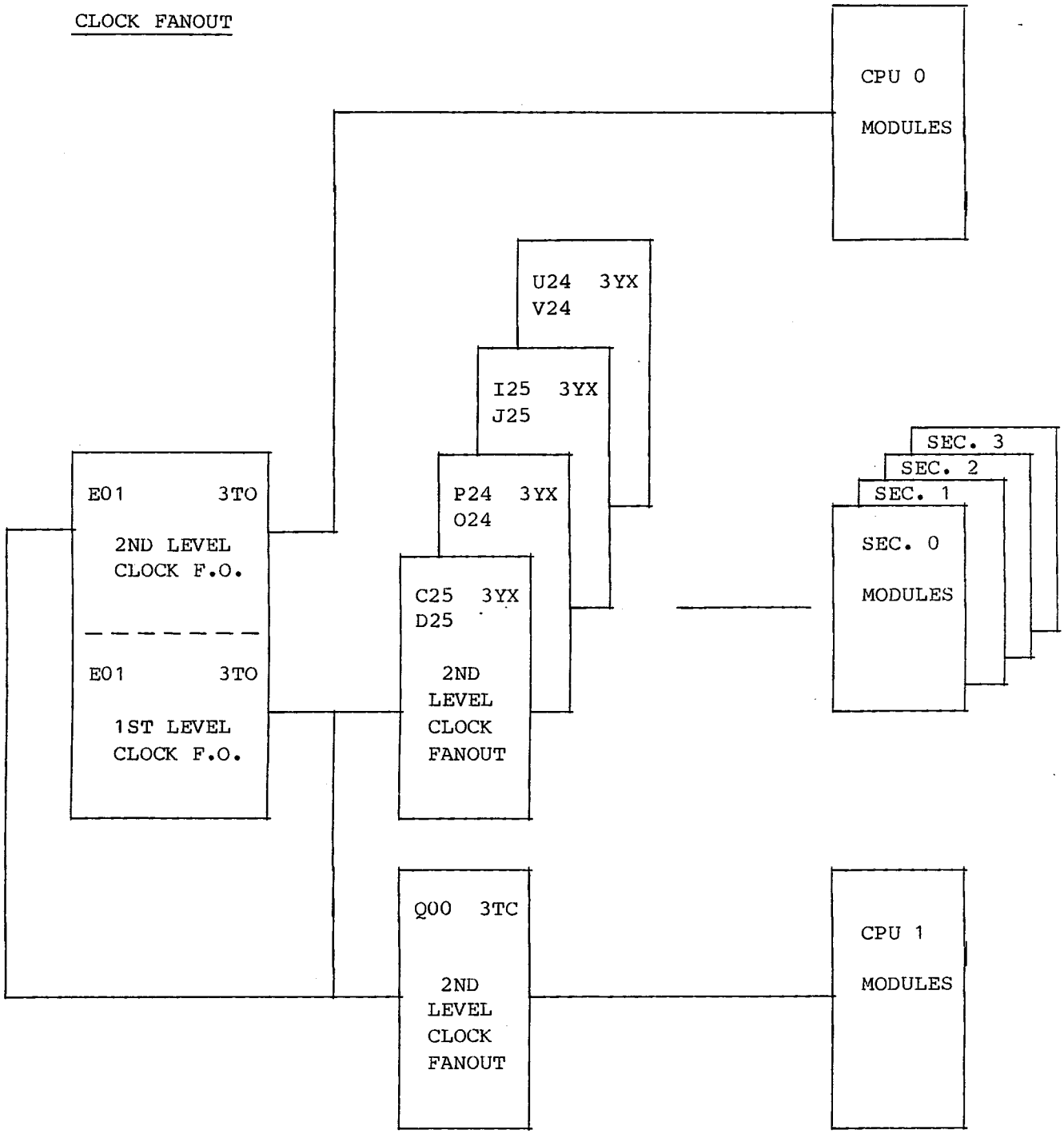
X4802C0616

3ZE MODULE BIT ASSIGNMENTS

LOC.	I00-I31	I32-I63	CPU	I64-70
E21	DATA BIT 32-63	DATA BIT 00-31	1	CHECK BIT 4-6 AND 0-3
E23	DATA BIT 00-31	DATA BIT 32-63	1	CHECK BIT 0-2 AND 4-7
E25	DATA BIT 32-63	DATA BIT 00-31	0	CHECK BIT 4-6 AND 0-3
E27	DATA BIT 00-31	DATA BIT 32-63	0	CHECK BIT 0-2 AND 4-7
Q20	DATA BIT 00-31	DATA BIT 32-63	0	CHECK BIT 0-2 AND 4-7
Q22	DATA BIT 32-63	DATA BIT 00-31	0	CHECK BIT 4-6 AND 0-3
Q24	DATA BIT 00-31	DATA BIT 32-63	1	CHECK BIT 0-2 AND 4-7
Q26	DATA BIT 32-63	DATA BIT 00-31	1	CHECK BIT 4-6 AND 0-3
H21	DATA BIT 32-63	DATA BIT 00-31	1	CHECK BIT 4-6 AND 0-3
H23	DATA BIT 00-31	DATA BIT 32-63	1	CHECK BIT 0-2 AND 4-7
H25	DATA BIT 32-63	DATA BIT 00-31	0	CHECK BIT 4-6 AND 0-3
H27	DATA BIT 00-31	DATA BIT 32-63	0	CHECK BIT 0-2 AND 4-7
T20	DATA BIT 00-31	DATA BIT 32-63	0	CHECK BIT 0-2 AND 4-7
T22	DATA BIT 32-63	DATA BIT 00-31	0	CHECK BIT 4-6 AND 0-3
T24	DATA BIT 00-31	DATA BIT 32-63	1	CHECK BIT 0-2 AND 4-7
T26	DATA BIT 32-63	DATA BIT 00-31	1	CHECK BIT 4-6 AND 0-3

X4802C0617

CLOCK FANOUT



X4802C0618

HTV-0242

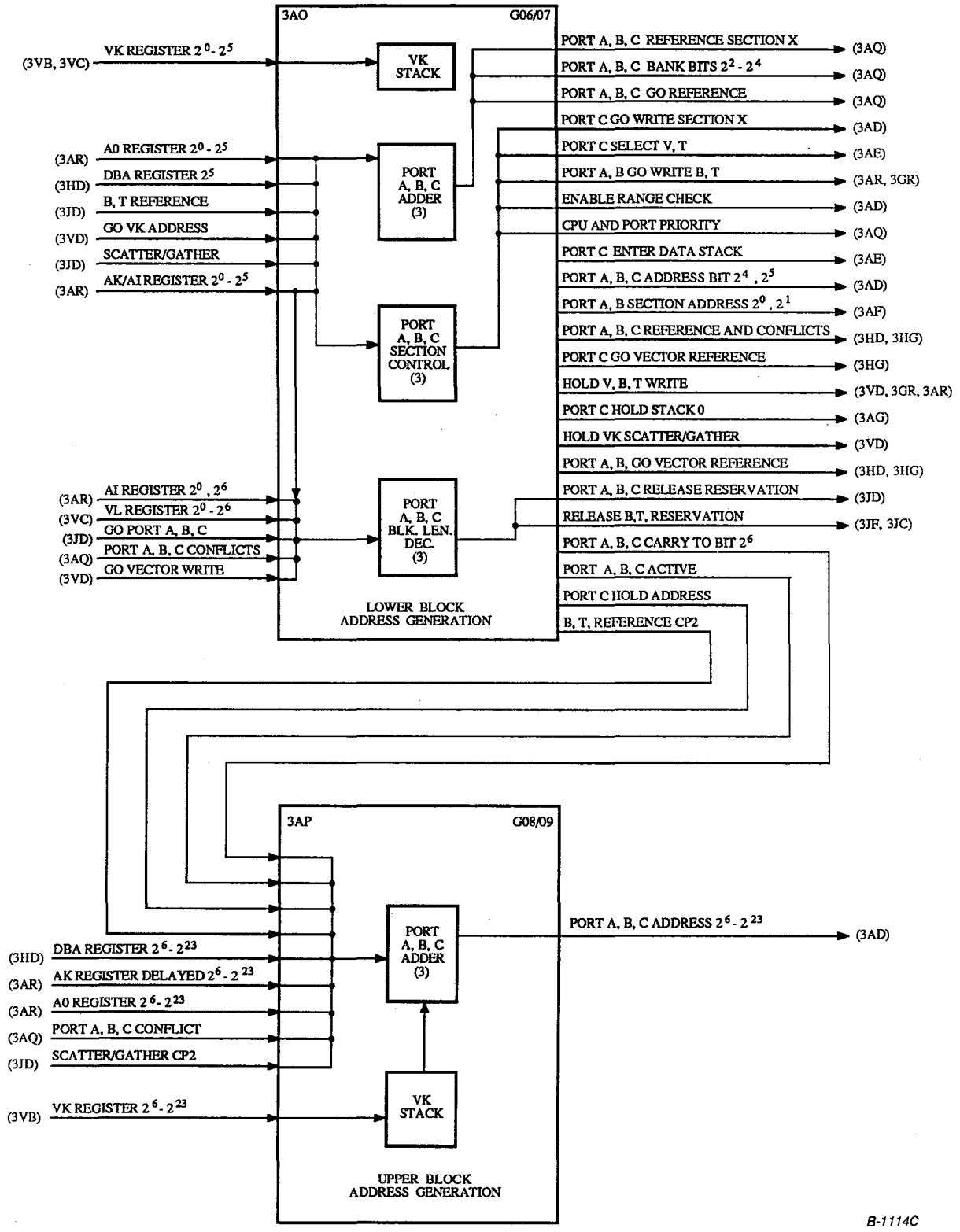
4 CLOCK PERIOD CYCLE

SOURCE			DESTINATION		
TYPE	LOC.	TERM	TYPE	LOC.	TERM
3YX	C25	R58	3HR	F05	I155
3YX	C25	R58	3HR	F07	I155
3YX	D25	R58	3HA	F09	I89
3YX	D25	R58	3AO	G07	I49
3YX	I25	R58	3AQ	G01	I102
3YX	I25	R58	3AQ	G03	I102
3YX	J25	R58	3HE	G05	I76
3YX	J25	R58	3AF	H29	I115
3YX	O24	R58	3HR	R04	I155
3YX	O24	R58	3HR	R06	I155
3YX	P24	R58	3HA	R08	I89
3YX	P24	R58	3AO	S06	I49
3YX	U24	R58	3AQ	S00	I102
3YX	U24	R58	3AQ	S02	I102
3YX	V24	R58	3AF	S04	I76
3YX	V24	R58	3HE	T28	I115

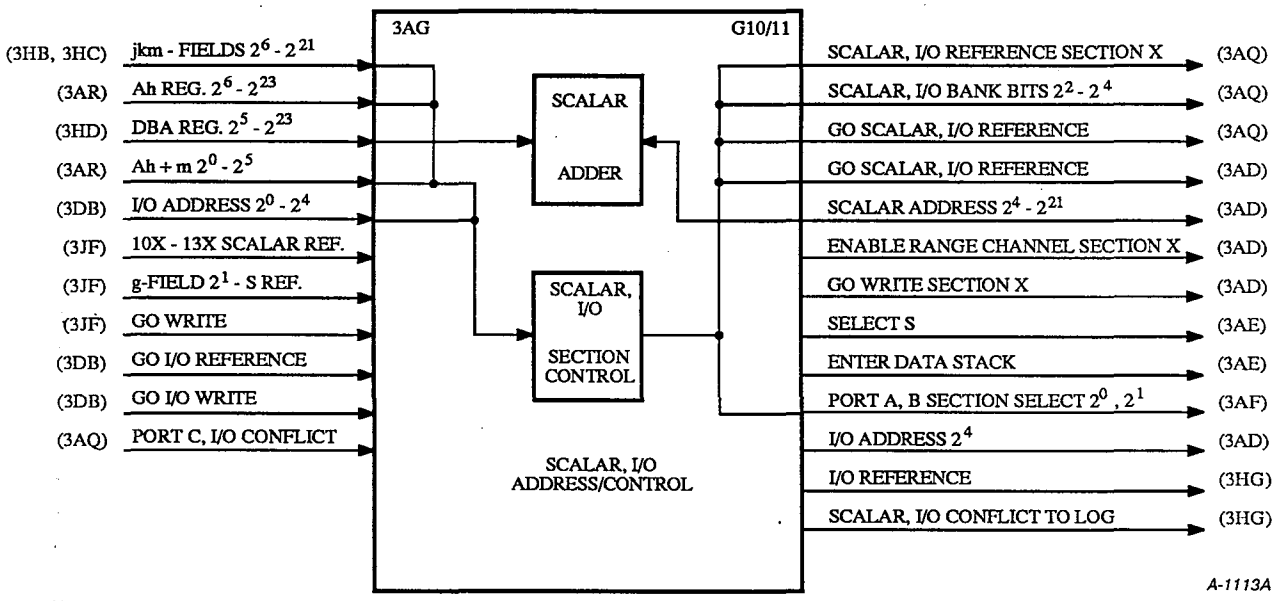
CYCLE SELECT

TYPE	LOC.	TERM	TYPE	LOC.	TERM
3YX	D25	R56	3AQ	G03	I103
3YX	I25	R56	3AQ	G01	I103
3YX	P24	R56	3AQ	S02	I103
3YX	U24	R56	3AQ	S00	I103

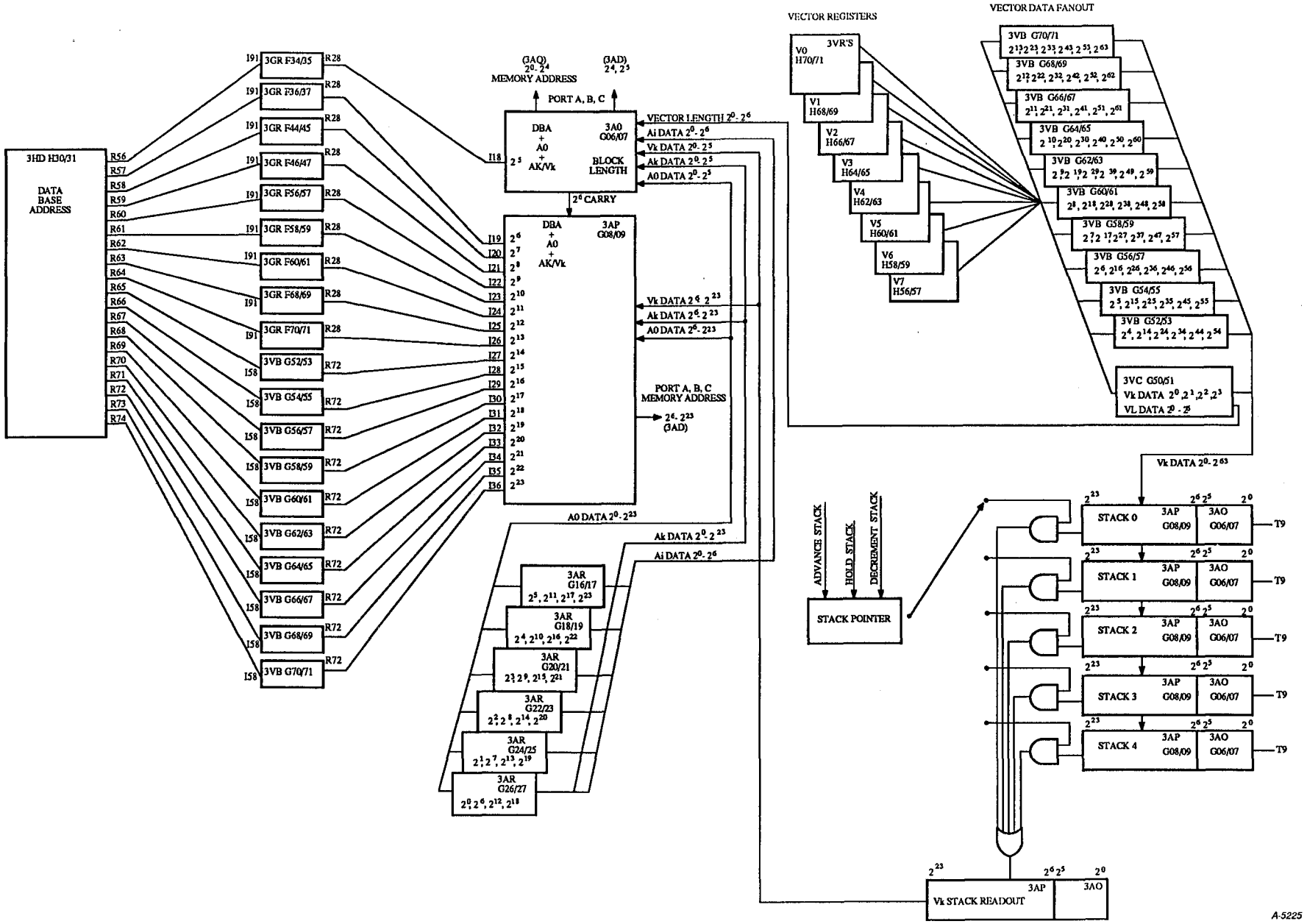
X4802C0619



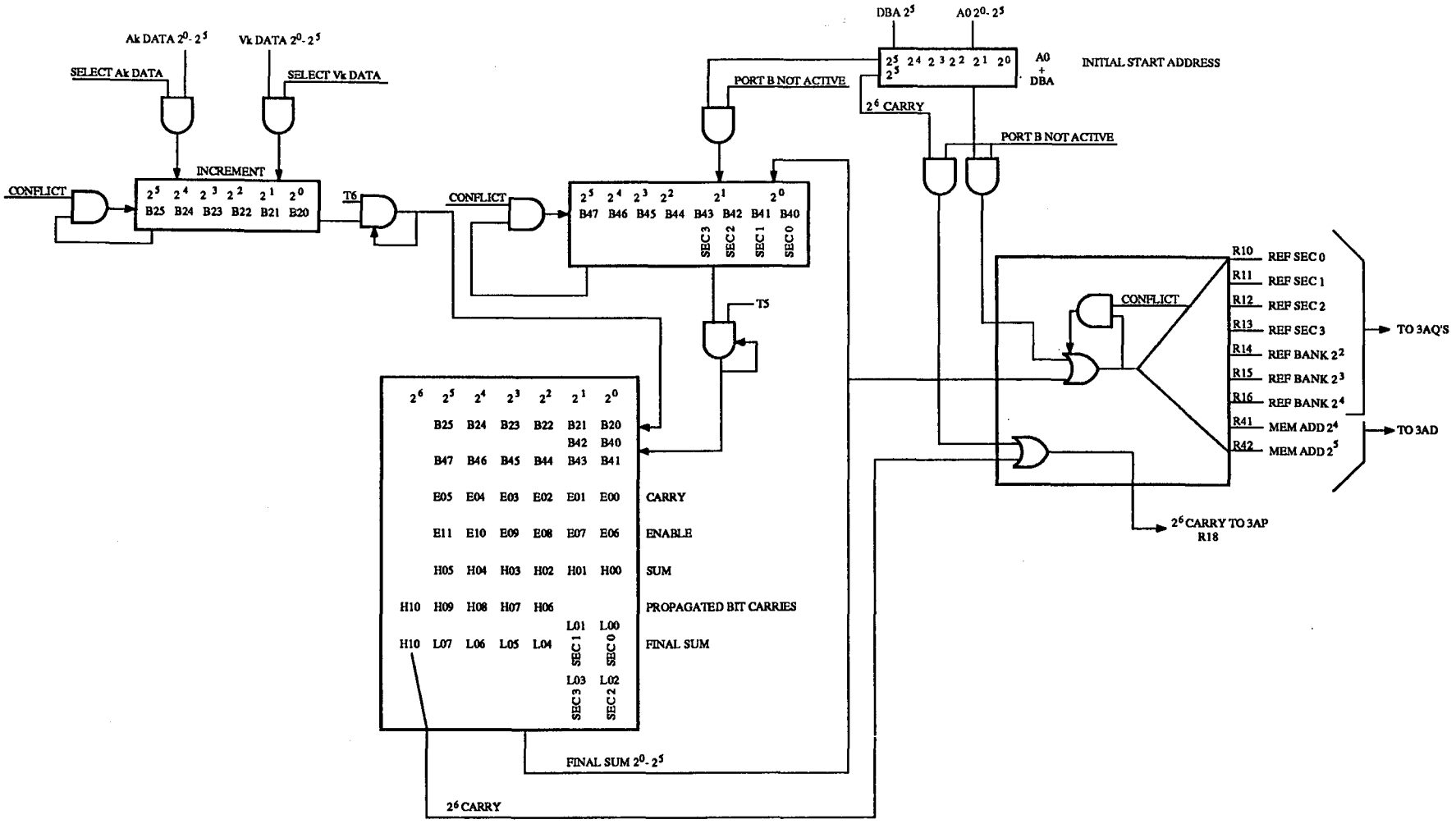
CRAY X-MP/2 MEMORY BLOCK TRANSFER ADDRESS GENERATION (CPU 0)



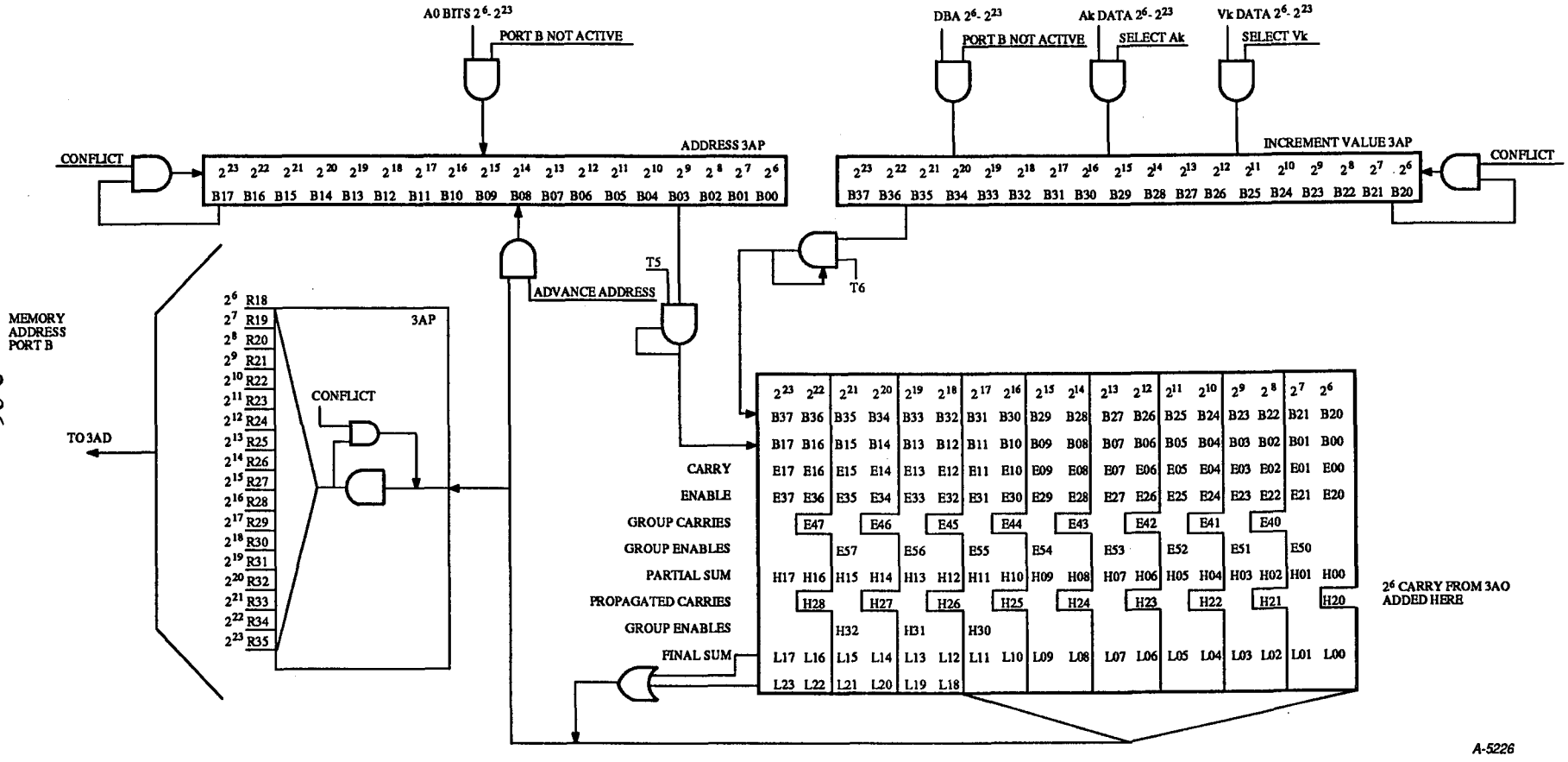
SCALAR, I/O MEMORY ADDRESS GENERATION
CRAY X-MP CPU-0



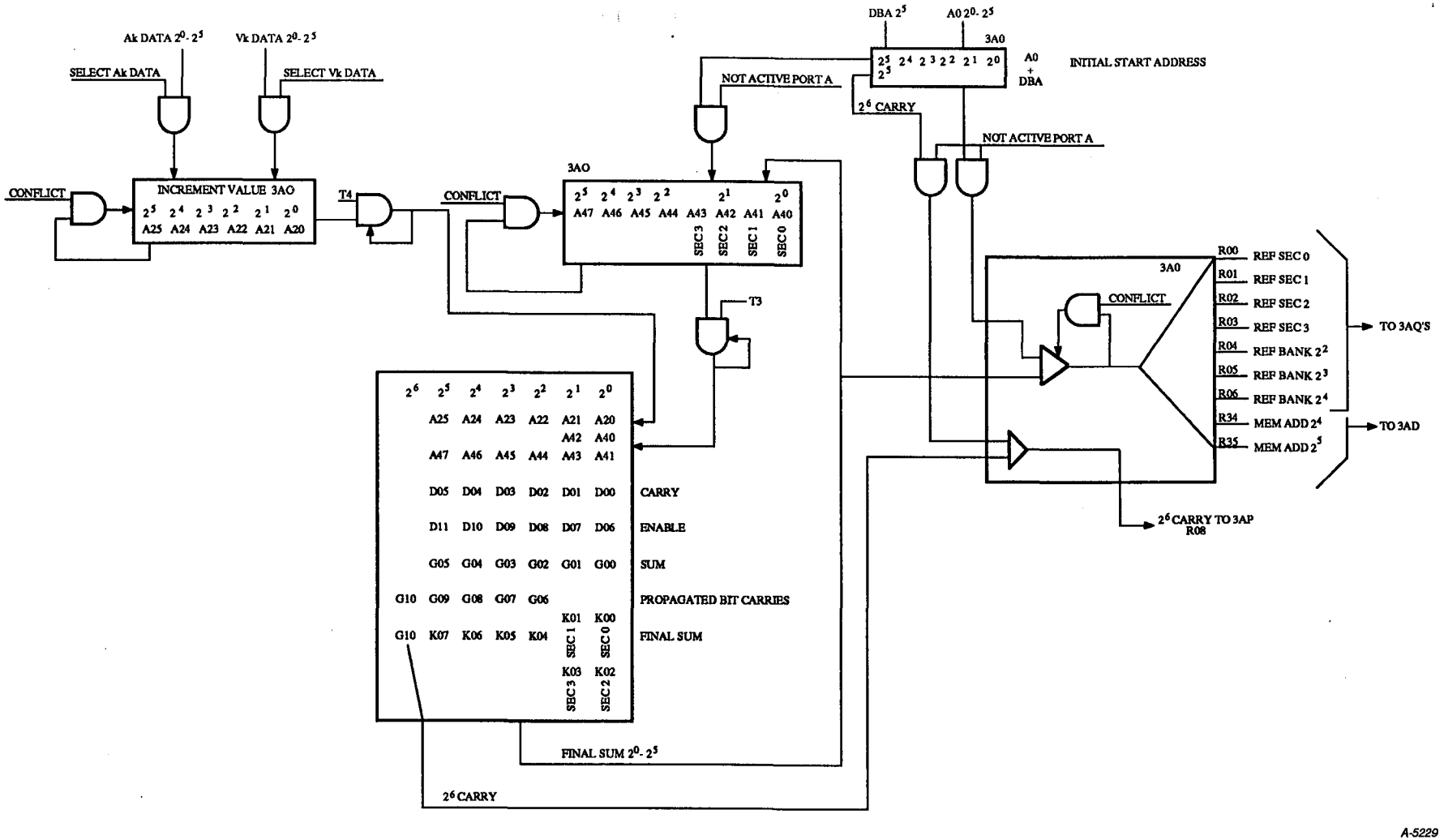
CRAY X-MP BLOCK ADDRESSING



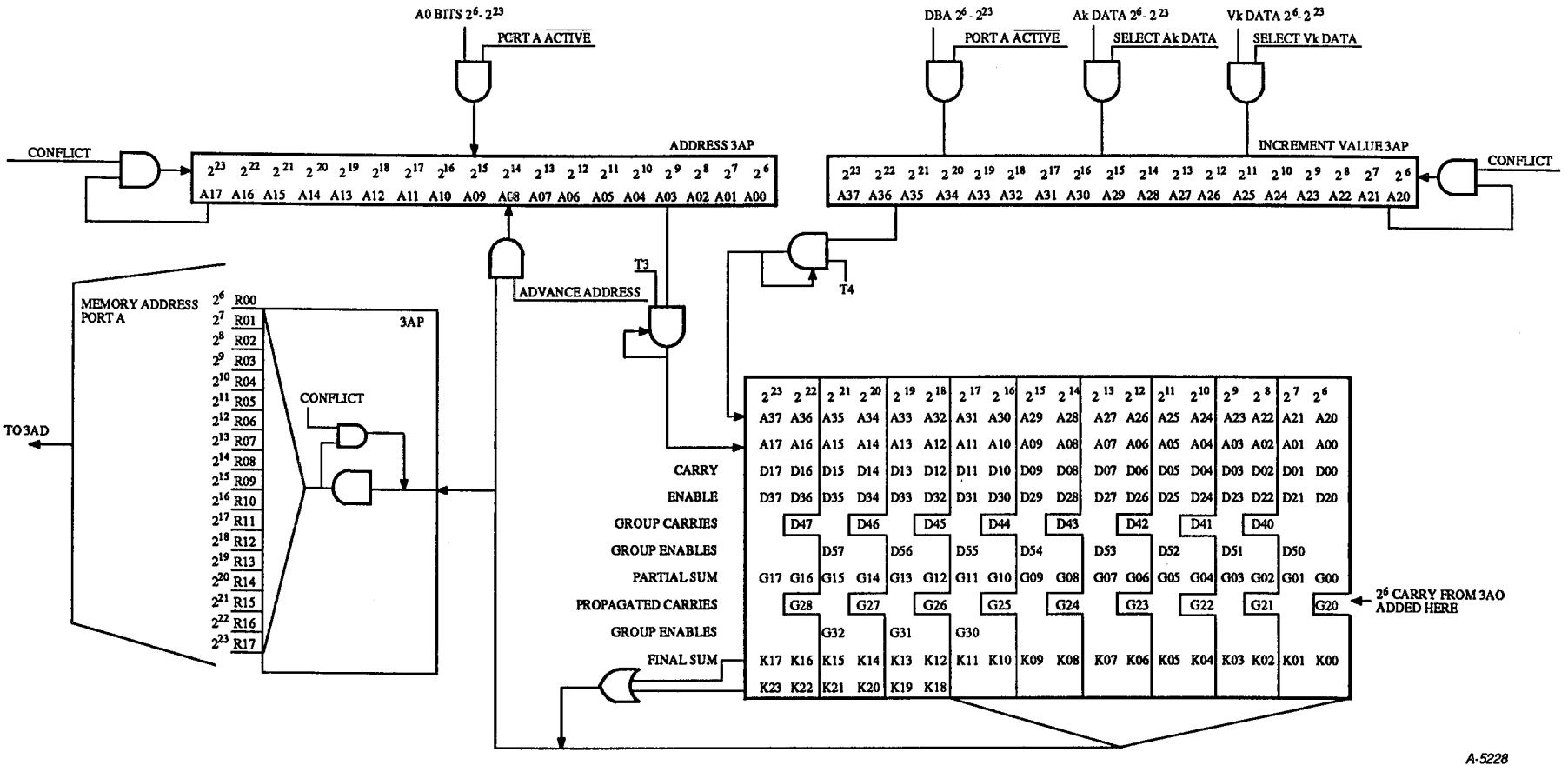
CRAY X-MP 3AO PORT B ADDRESS



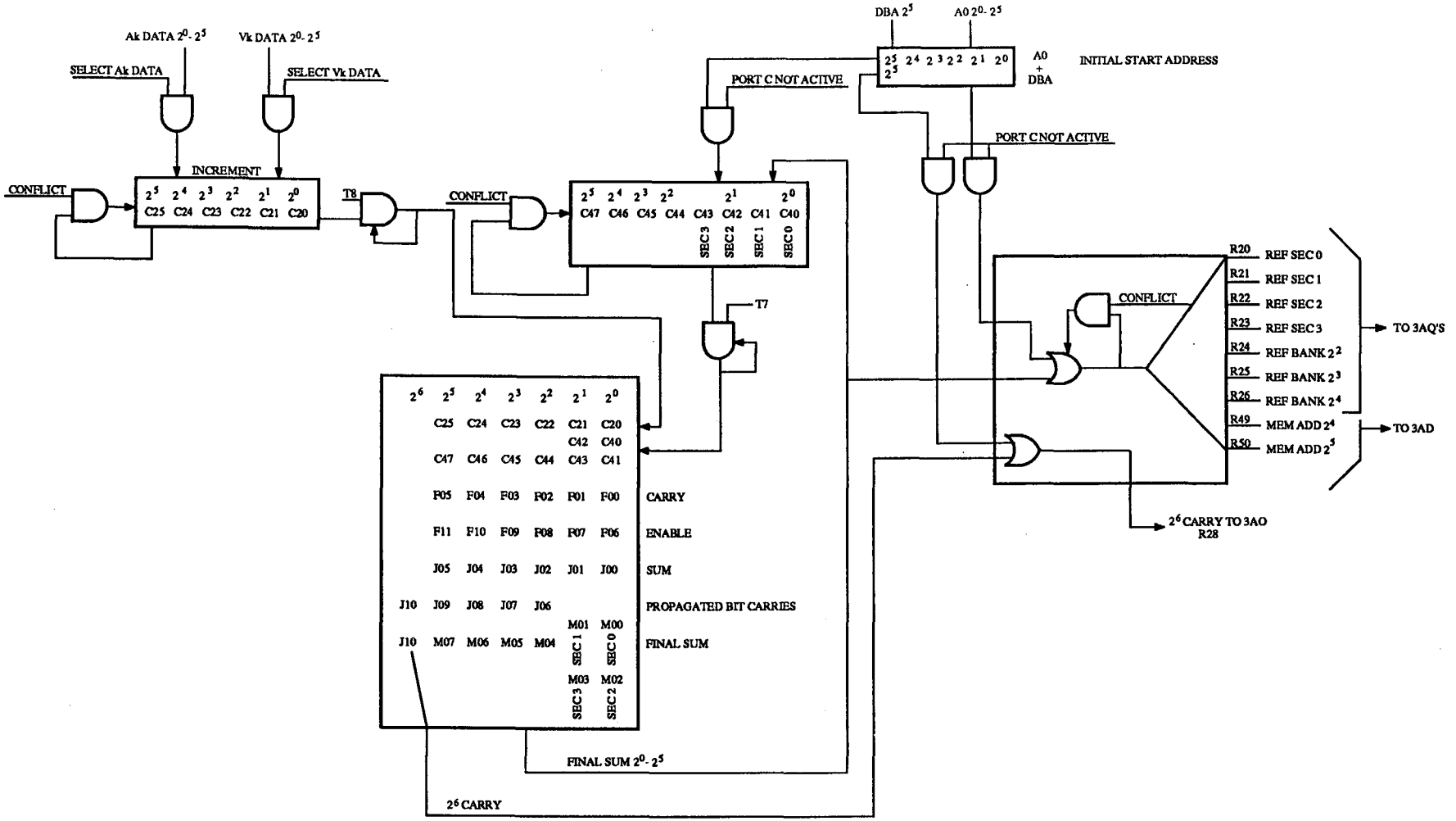
CRAY X-MP 3AP PORT B ADDRESS



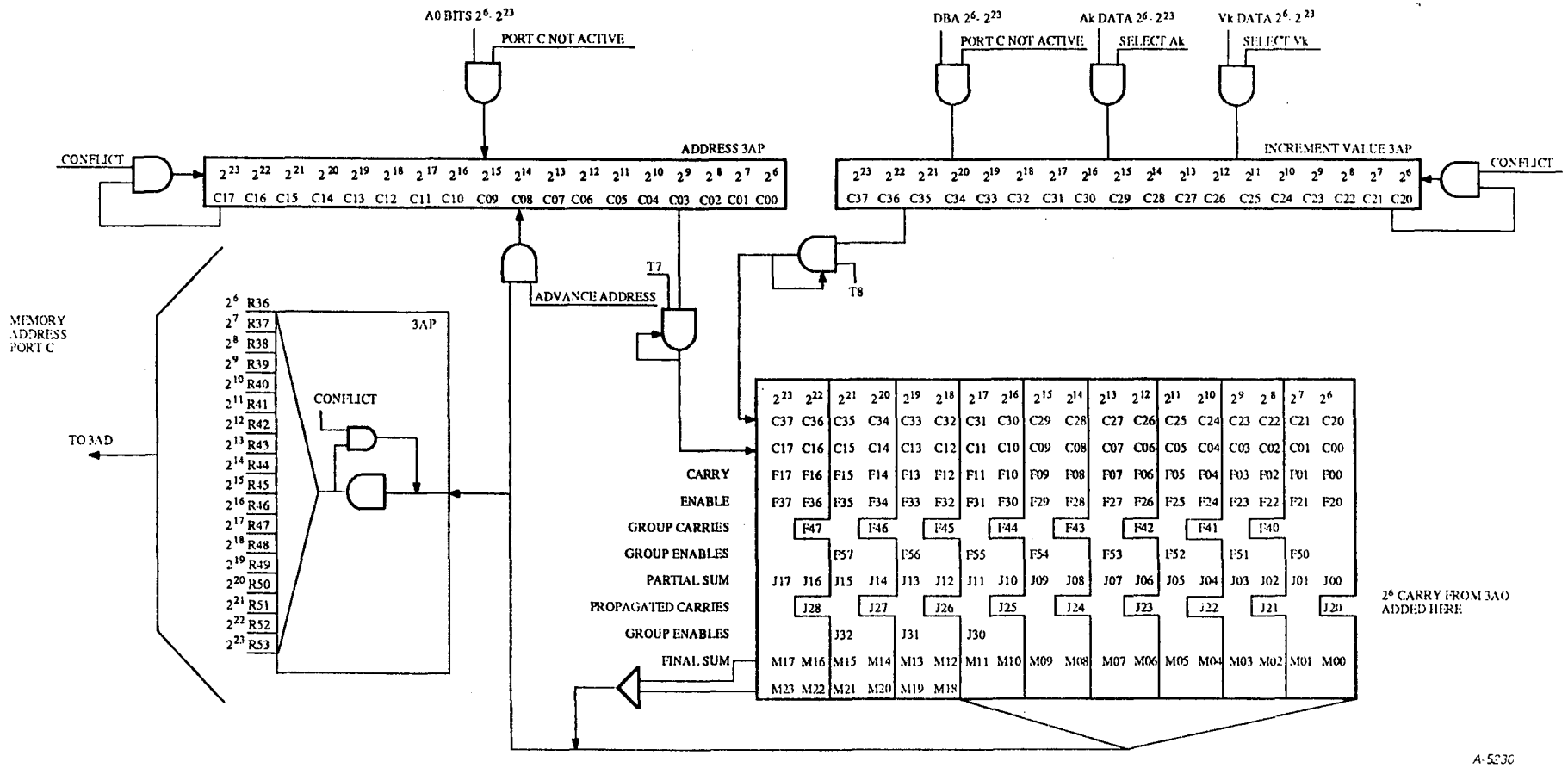
CRAY X-MP 3AO PORT A ADDRESS



CRAY X-MP 3AP PORT A ADDRESS

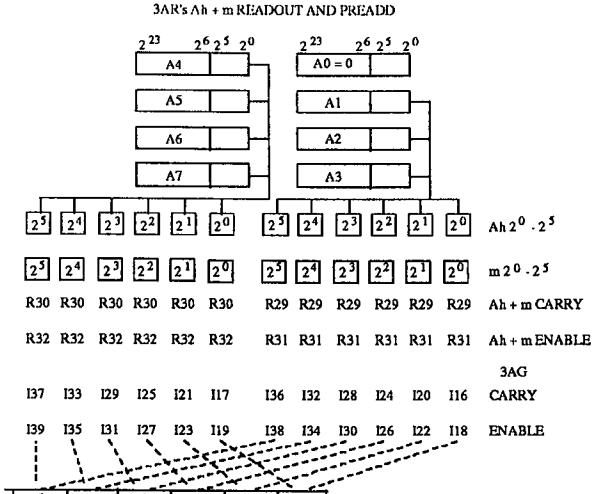


CRAY X-MP 3AO PORT C ADDRESS



CRAY X-MP 3AP PORT C ADDRESS

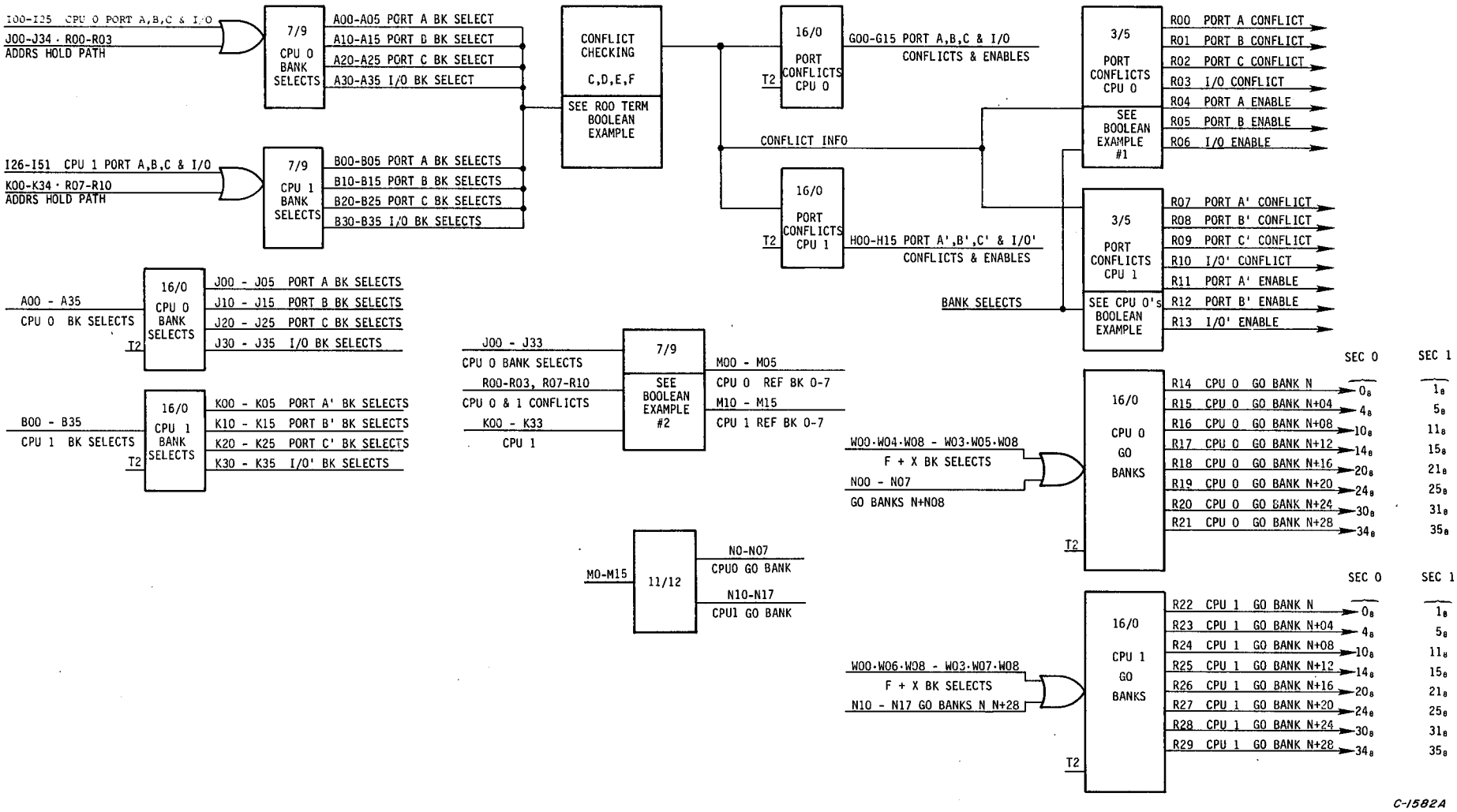
SCALAR MEMORY ADDRESSING



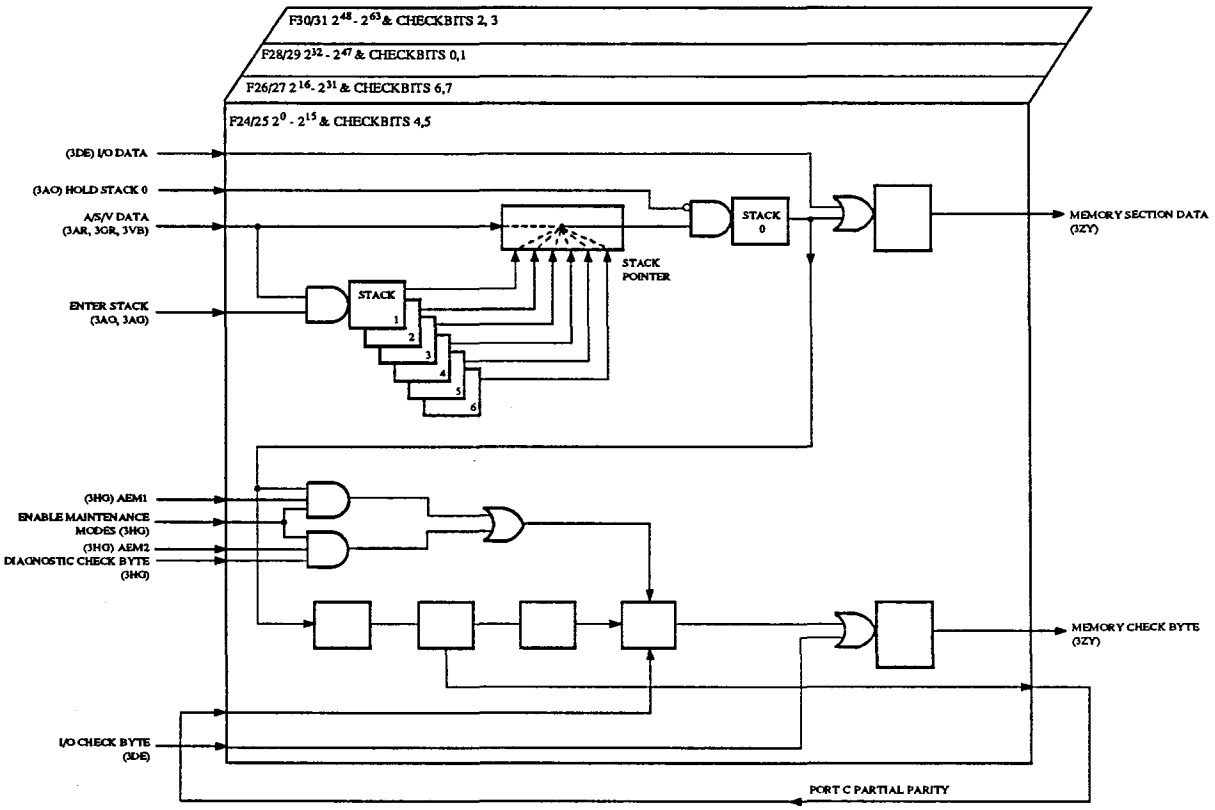
2^{23}	2^{22}	2^{21}	2^{20}	2^{19}	2^{18}	2^{17}	2^{16}	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}	
																		B05	B04	B03	B02	B01	B00	SUM
																	B12	B11	B09	B08	B07	B06		BIT CARRIES
																				B10				
																			R06	R05	R04	$\overline{2^1}$	$\overline{2^0}$	SEC. 0 SELECT 2^0 2^1 R00
																			R07			$\overline{2^1}$	2^0	SEC. 1 SELECT 2^0 2^1 R01
																						2^1	$\overline{2^0}$	SEC. 2 SELECT 2^0 2^1 R02
																						2^1	2^0	SEC. 3 SELECT 2^0 2^1 R03
																								2^2 - 2^4 = R04 - R07 BANK SELECTS
A15	A15	A15	A14	A13	A12	A11	A10	A09	A08	A07	A06	A05	A04	A03	A02	A01	A00							JKM FROM 3HC, 3HB
I76	I75	I74	I73	I72	I71	I70	I69	I68	I67	I66	I65	I64	I63	I62	I61	I60	I59	I58						DBA FROM 3GR's AND 3VB's
I57	I56	I55	I54	I53	I52	I51	I50	I49	I48	I47	I46	I45	I44	I43	I42	I41	I40							Ah FROM 3AR's
*D36	*D34	*D32	*D30	*D28	*D26	*D24	*D22	*D20	*D18	*D16	*D14	*D12	*D 10	*D08	*D06	*D04	*D02	*D00						CARRIES Ah + JKM + DBA
D37	D35	D33	D31	D29	D27	D25	D23	D21	D19	D17	D15	D13	D11	D09	D07	D05	D03	D01						SUMS
		D47		D46		D45		D44		D43		D42		D41		D40								Ah + jkm BIT CARRIES
	D58		D57		D56		D55		D54		D53		D52		D51		D50							Ah + jkm SUMS
E21										E20														Ah + jkm PROPAGATED CARRIES
	E22																							Ah + jkm ENABLE BITS 2^{14} - 2^{22}
E18	E17	E16	E15	E14	E13	E12	E11	E10	E09	E08	E07	E06	E05	E04	E03	E02	E01	E00						Ah + jkm + DBA BIT CARRIES
																		F01	F00					Ah + jkm + DBA SUM 2^4 , 2^5
F19	F18	F17	F16	F15	F14	F13	F12	F11	F10	F09	F08	F07	F06	F05	F04	F03	F02							Ah + jkm + DBA ENABLE BITS
G17	G16	G15	G14	G13	G12	G11	G10	G09	G08	G07	G06	G05	G04	G03	G02	G01	G00							Ah + jkm + DBA CARRY BITS
	H04			H03			H02			H01			H00											GROUP ENABLES
J05	J06		J04			J03			J02			J01			J00									PROPAGATED GROUP CARRIES
K19	K18	K17	K16	K15	K14	K13	K12	K11	K10	K09	K08	K07	K06	K05	K04	K03	K02	K01	K00					PARTIAL SUM
L19	L18	L17	L16	L15	L14	L13	L12	L11	L10	L09	L08	L07	L06	L05	L04	L03	L02	L01	L00					LATCHED PARTIAL SUM HELD FOR SCALAR CONFLICTS EAM MODIFICATION ON 2^{22} AND 2^{23}
M05			M04			M03			M02			M01			M00									LATCHED PROPAGATED GROUP CARRIES HELD FOR SCALAR CONFLICTS
				M09			M08			M07			M06											LATCHED GROUP ENABLES HELD FOR SCALAR CONFLICTS
R39	R38	R37	R36	R35	R34	R33	R32	R31	R30	R29	R28	R27	R26	R25	R24	R23	R22	R21	R20					FINAL SUM TO 3AD

E19

A-4831
XVC01C36M



3AQ MODULE DIAGRAM
MEMORY SECTION CONTROL



MEMORY DATA SELECTION AND CHECK BYTE GENERATION

A, S, or V memory data enters the 3AE module and is put on stack 1. Depending on the value of the stack pointer and the condition of the Hold Stack 0 Signal, it may also be put on stack 0 at the same time. When the Hold Stack 0 signal is dropped, subsequent data entering the 3AE is put only on stack 1, with stack 1's previous data moving to stack 2, and the stack pointer being incremented. The data on stack 0 is held there by a controlled clock. This process continues with incoming data being put on stack 1, previous data being moved to the next highest stack, and the stack pointer being incremented until the Hold Stack 0 signal is again raised. At this point, stack 0's data is sent off to memory, the data on the highest stack (as indicated by the stack pointer) is moved to stack 0, and new data (if there is any) is put on stack 1, with other data holding (or advancing), and the stack pointer decrementing (or holding) accordingly. This process continues until all of the stacks are empty, and the stack pointer is once again at 0.

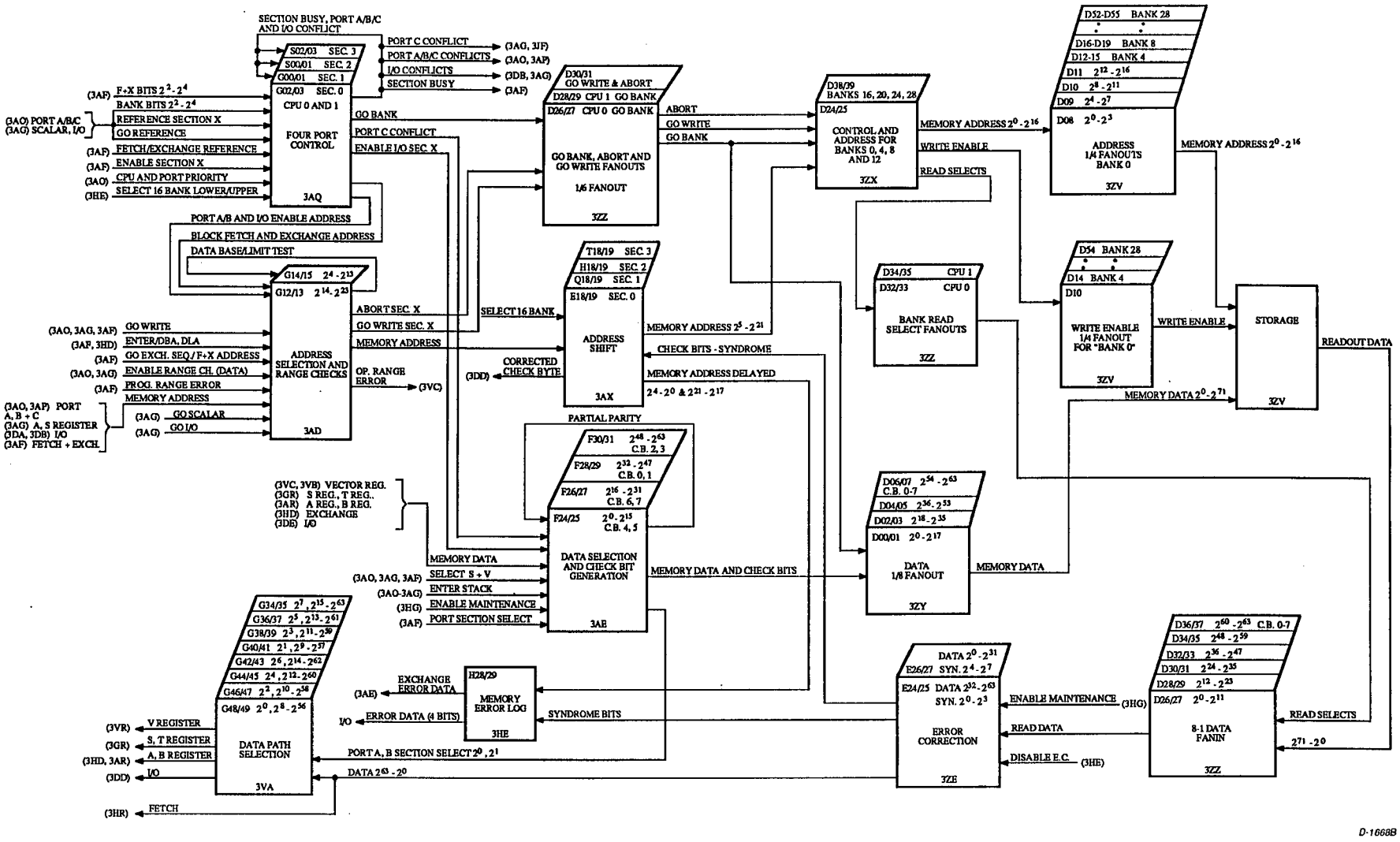
As the data enters stack 0, a copy is also sent elsewhere on the module for check byte generation. A partial parity check is sent off board to the other 3AE's at TS 3/5, and information from them is received at TS 11/13, just in time for the final check byte to be generated and sent off with the memory data at TS 16/0. In the event one of the two maintenance modes was selected, the check byte is generated either from the upper 2 bits of each parcel (AEM1 mode), or from a separate diagnostic check byte read in from the S1 register via a 001511 monitor instruction (AEM2 mode).

C-3660

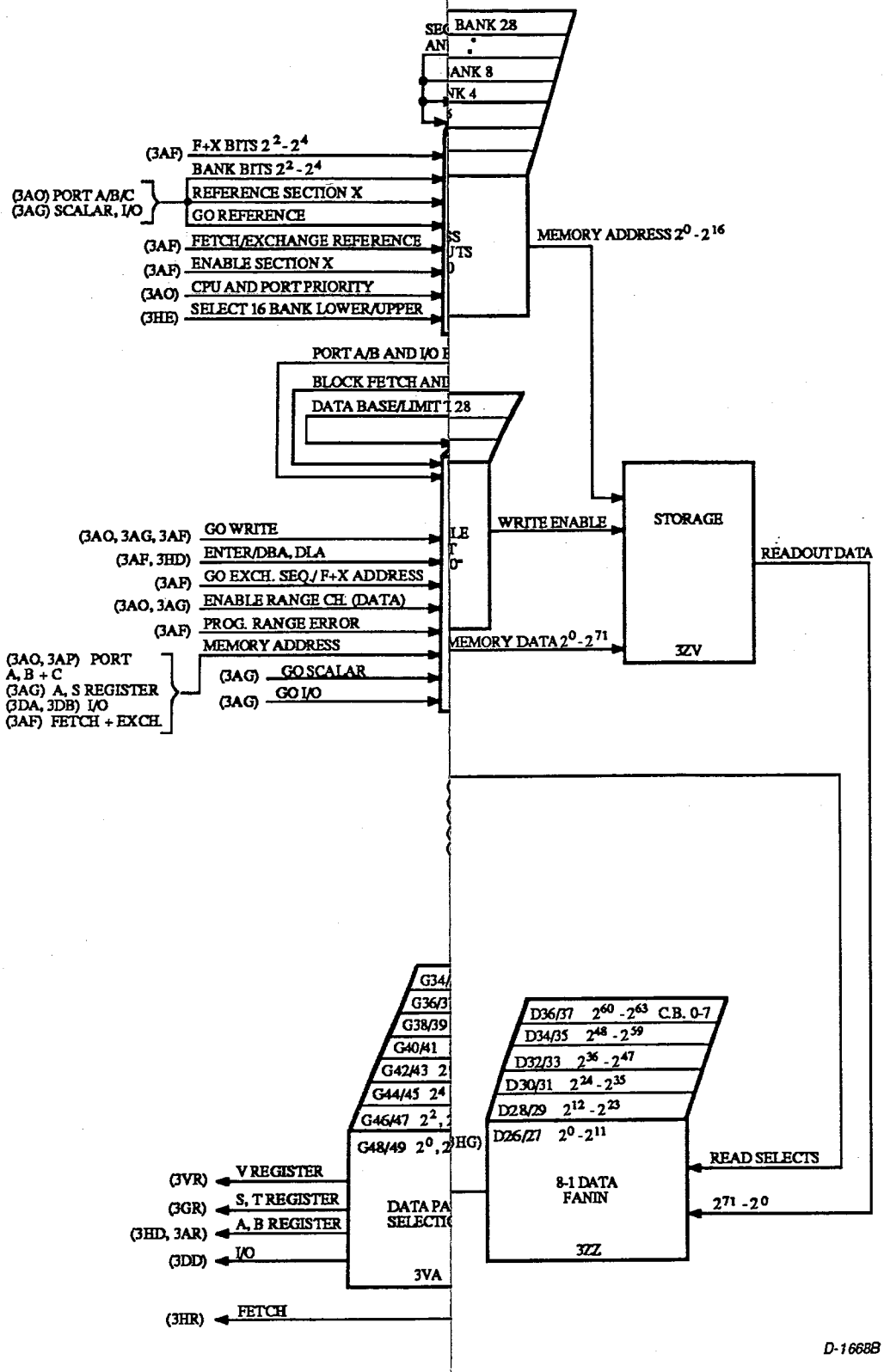
3AE MODULE BLOCK DIAGRAM

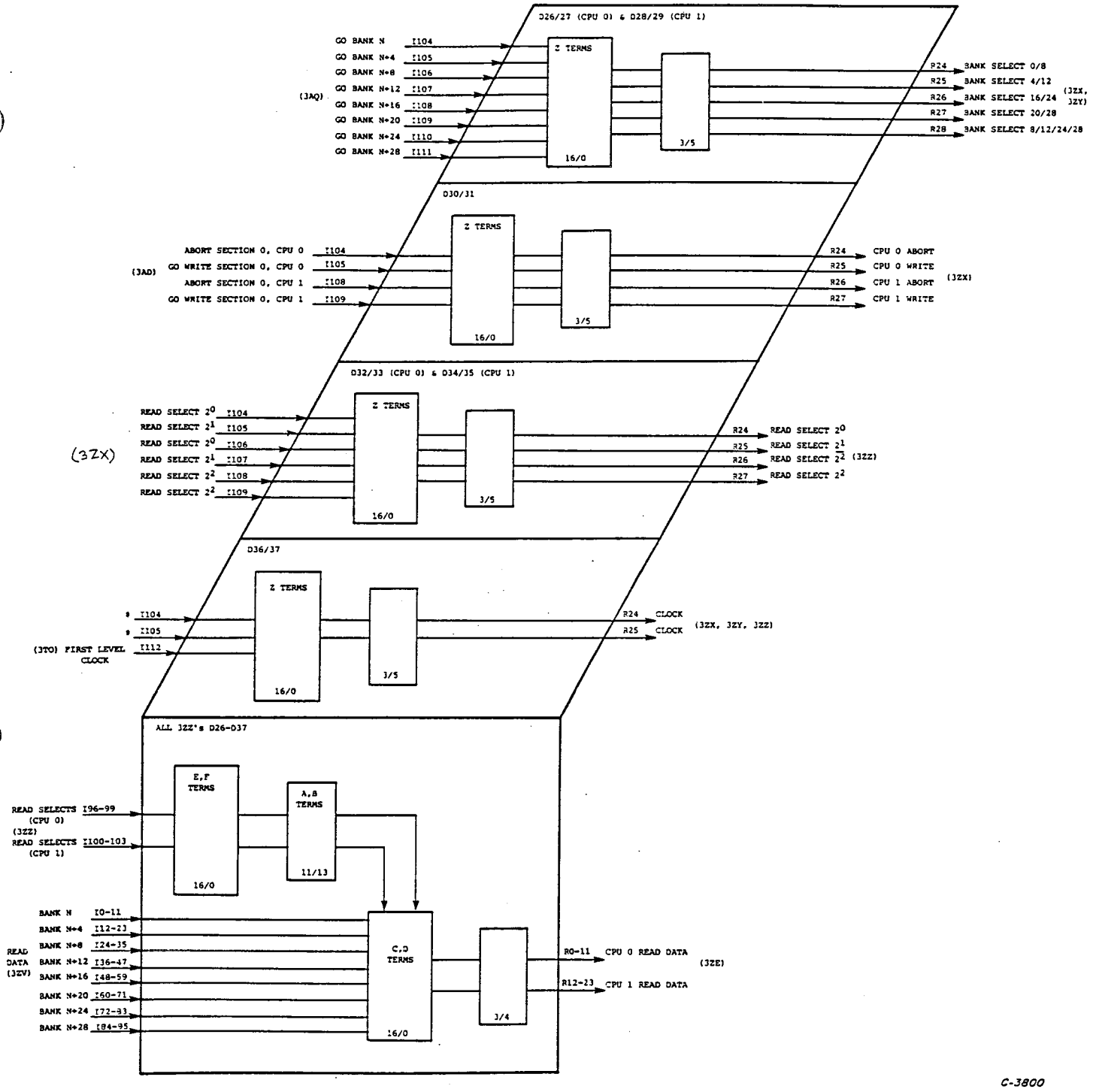
CRAY X-MP / 22,24
6 COLUMN
SECTION

ECL MEMORY CHIPS
(16K)



CRAY X-MP MEMORY BLOCK DIAGRAM
(CPU 0 AND SECTION 0) (6 COLUMN)


BANK 28
BANK 8
BANK 4
(3AF) F+X BITS 2^2 - 2^4
BANK BITS 2^2 - 2^4
(3AO) PORT A/B/C
(3AG) SCALAR, I/O
REFERENCE SECTION X
GO REFERENCE
(3AF) FETCH/EXCHANGE REFERENCE
(3AF) ENABLE SECTION X
(3AO) CPU AND PORT PRIORITY
(3HE) SELECT 16 BANK LOWER/UPPER
MEMORY ADDRESS 2^0 - 2^16
PORT A/B AND I/O
BLOCK FETCH AND
DATA BASE/LIMIT
28
(3AO, 3AG, 3AF) GO WRITE
(3AF, 3HD) ENTER/DBA, DLA
(3AF) GO EXCH. SEQ./ F+X ADDRESS
(3AO, 3AG) ENABLE RANGE CH. (DATA)
(3AF) PROG. RANGE ERROR
MEMORY ADDRESS
WRITE ENABLE
STORAGE
READOUT DATA
MEMORY DATA 2^0 - 2^71
3ZV
(3AO, 3AP) PORT A, B + C
(3AG) A, S REGISTER
(3DA, 3DB) I/O
(3AF) FETCH + EXCH.
(3AG) GO SCALAR
(3AG) GO I/O
G34/
G36/3
G38/39
G40/41
G42/43 2
G44/45 2^4
G46/47 2^2, 2
G48/49 2^0, 2^ HG)
D36/37 2^60 - 2^63 C.B. 0-7
D34/35 2^48 - 2^59
D32/33 2^36 - 2^47
D30/31 2^24 - 2^35
D28/29 2^12 - 2^23
D26/27 2^0 - 2^11
READ SELECTS
8-1 DATA FANIN
2^71 - 2^0
3ZZ
(3VR) V REGISTER
(3GR) S, T REGISTER
(3HD, 3AR) A, B REGISTER
(3DD) I/O
DATA PA
SELECTI
3VA
(3HR) FETCH
D-1668B



CRAY X-MP/2 3ZZ MODULE BLOCK DIAGRAM

Each of the 3ZZ Modules performs 2 main functions: Receiving memory data from each of the 8 banks within a particular section, and then selecting the data from one of them to pass on to the 3ZE Module for error correction; and fanning out various control signals depending on the location of the module. These control signals include the Go banks, Aborts, Go writes, Read selects and the clock to coordinate memory activities. The fanout is 1/6, with 6 copies of each of the R-terms (R24-R28) being generated.

BANK	COL.	BITS 0-17	18-35	36-53	54-71
0 1 2 3	D P I U	8	9	10	11
4 5 6 7	D P I U	12	13	14	15
10 11 12 13	D P I U	16	17	18	19
14 15 16 17	D P I U	20	21	22	23
20 21 22 23	D P I U	40	41	42	43
24 25 26 27	D P I U	44	45	46	47
30 31 32 33	D P I U	48	49	50	51
34 35 36 37	D P I U	52	53	54	55

< MODULE < LOCATIONS <

D	COL	I
34		36
30		32
24		26
20		22
14		16
10		12
	BANKS	
4		6
0		2
1		3
5		7
11		13
15		17
21		23
25		27
31		33
35		37

P	COL	U

ADDRESS TEST POINT CHART

ABSOLUTE ADDRS BIT	BANK ADDRS BIT	BANK N+0 TEST POINT	BOOLEAN	BANK N+4 TEST POINT	BOOLEAN	BANK N+8(10_8) TEST POINT	BOOLEAN	BANK N+12(14_8) TEST POINT	BOOLEAN
2^5	0	B71	R00	B72	R17	A04	R34	A10	R51
2^6	1	B69	R01	B70	R18	A07	R35	A08	R52
2^7	2	B67	R02	B65	R19	A21	R36	A11	R53
2^8	3	B53	R03	B54	R20	A20	R37	A19	R54
2^9	4	B06	R04	B07	R21	A61	R38	A60	R55
2^{10}	5	B04	R05	B05	R22	A68	R39	A69	R56
2^{11}	6	B17	R06	B02	R23	A71	R40	A70	R57
2^{12}	7	B03	R07	B01	R24	A72	R41	A67	R58
2^{13}	8	D72	R08	D71	R25	C02	R42	C01	R59
2^{14}	9	D69	R09	D70	R26	C03	R43	C06	R60
2^{15}	10	D65	R10	D62	R27	C12	R44	C10	R61
2^{16}	11	D53	R11	D54	R28	C20	R45	C13	R62
2^{17}	12	D12	R12	D20	R29	C64	R46	C66	R63
2^{18}	13	D04	R13	D05	R30	C67	R47	C68	R64
2^{19}	14	D01	R14	D21	R31	C69	R48	C71	R65
2^{20}	15	D19	R15	D22	R32	C72	R49	C70	R66
2^{21}	16	B39	R16	B23	R33	B22	R50	A52	R67
WRITE ENABLE		B51	R72	B21	R73	C21	R74	C51	R75

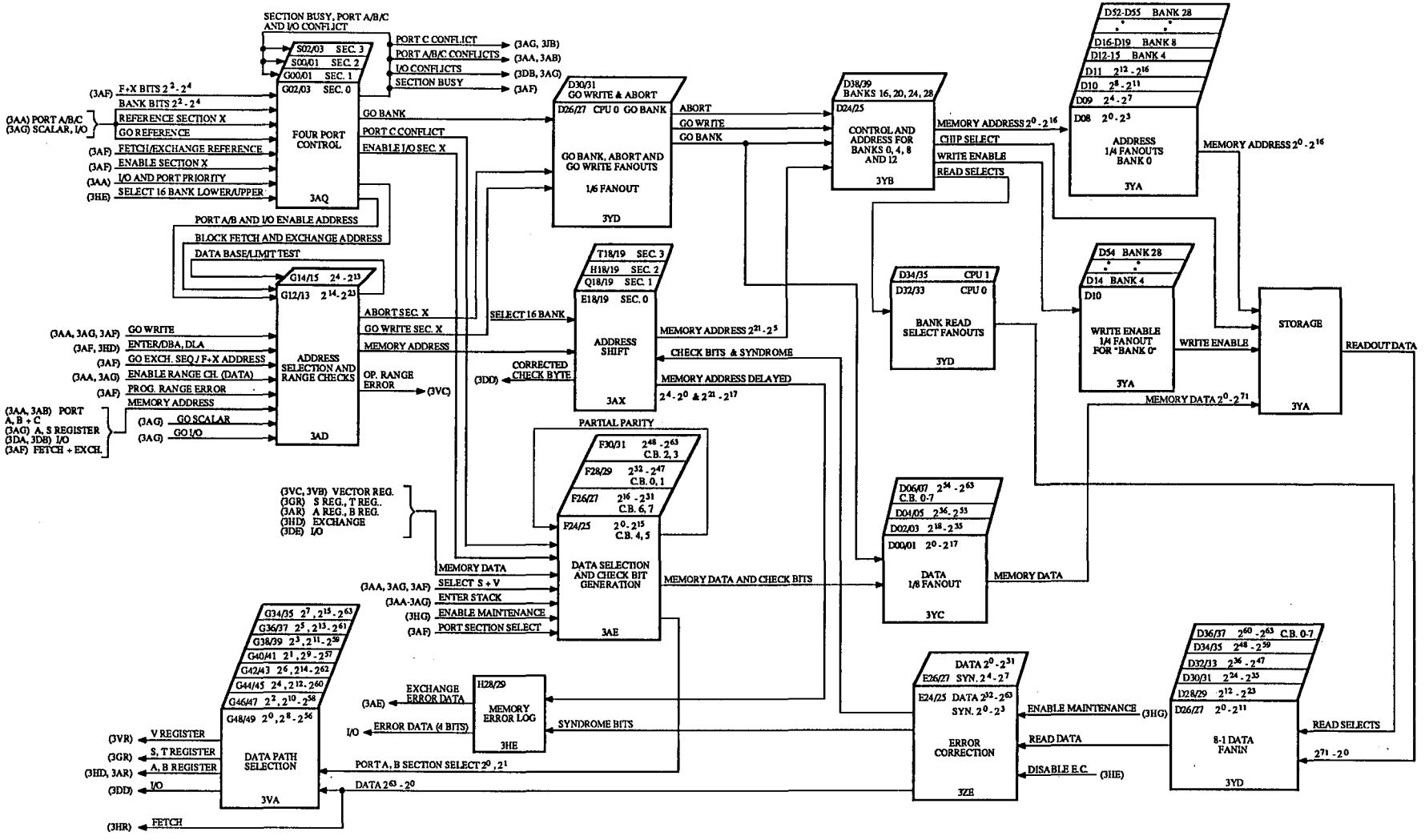
ADDRESS BIT FANOUTS

MODULES @ D8, 12, 16, 20, 40, 44, 48, 52
MODULES @ 9, 13, 17, 21, 41, 45, 49, 53
MODULES @ 10, 14, 18, 22, 42, 46, 50, 54
MODULES @ 11, 15, 19, 23, 43, 47, 51, 55

FANOUT BANK ADDRESS BIT 0 - 3 TO IT'S OWN BANK
FANOUT BANK ADDRESS 4 - 7 TO IT'S OWN BANK
FANOUT BANK ADDRESS 8 & 11 W.E. TO IT'S OWN BANK
FANOUT BANK ADDRESS 12 - 16 TO IT'S OWN BANK

CRAY X-MP / 11,12,14
MOSS MEMORY
SECTION

(16K CHIPS)



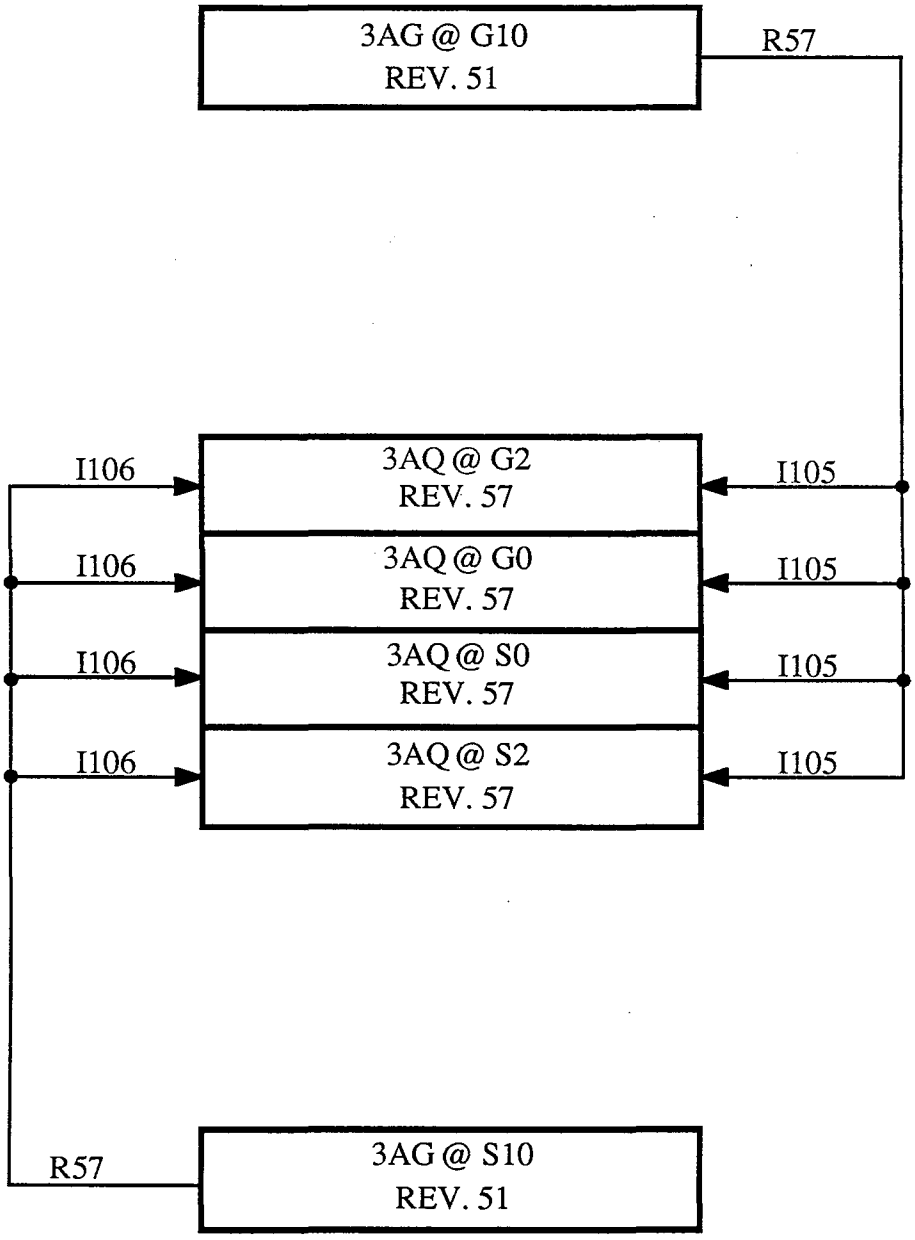
CRAY X-MP/1 MEMORY BLOCK DIAGRAM
(CPU 0 AND SECTION 0)

"I/O LOCKOUT CIRCUIT"

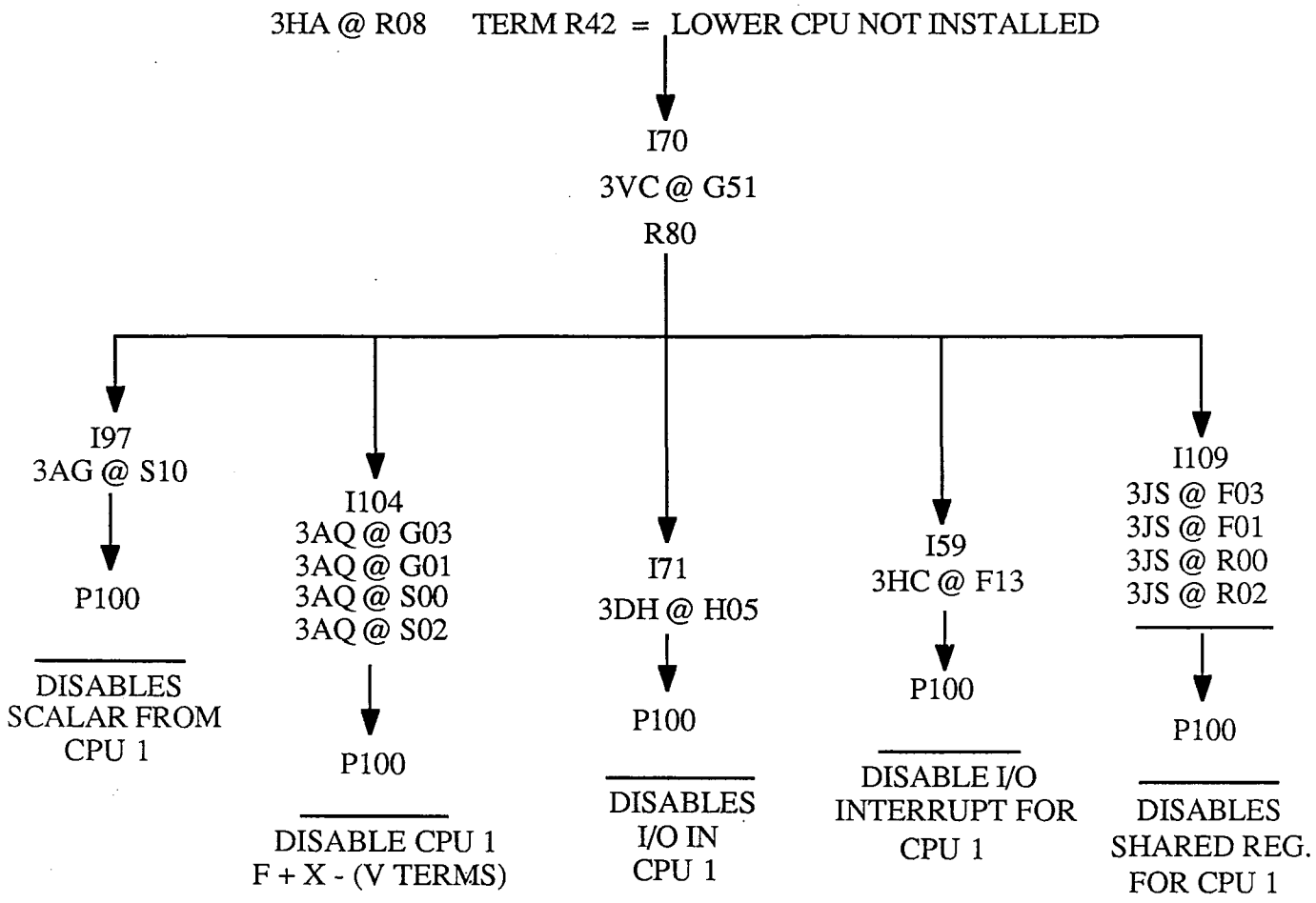
Description: When an I/O reference encounters a conflict, a counter on the 3AG module begins to count and continues to count for as long as the conflict lasts. When the counter reaches a count of 100_8, a signal called "I/O Lockout" is sent to each of the 3AQ's. The 3AQ's then bring up conflict on ports A, B, C and a short time later the I/O reference is allowed to proceed. As soon as the I/O reference proceeds (I/O conflict drops) the "I/O Lockout" signal from 3AG goes away and normal port operation once again proceeds.

XV2S07M

"I/O LOCKOUT CIRCUIT"

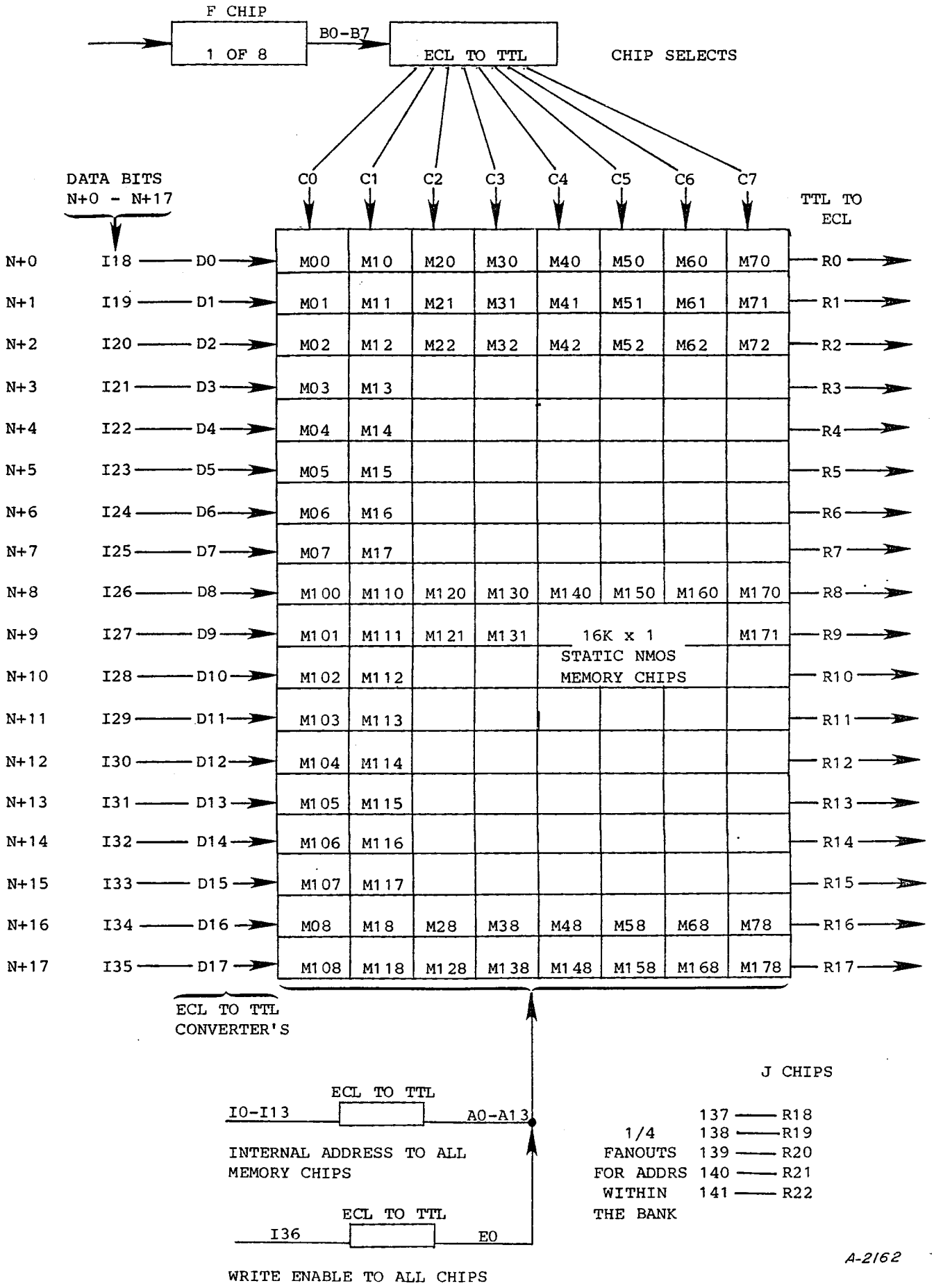


A-4938
XV2S08M



NO LOWER CPU INSTALLED

A-4939
XV2S09M


F CHIP
1 OF 8
B0-B7
ECL TO TTL
CHIP SELECTS
DATA BITS
N+0 - N+17
C0
C1
C2
C3
C4
C5
C6
C7
TTL TO
ECL
N+0
I18
D0
M00
M10
M20
M30
M40
M50
M60
M70
R0
N+1
I19
D1
M01
M11
M21
M31
M41
M51
M61
M71
R1
N+2
I20
D2
M02
M12
M22
M32
M42
M52
M62
M72
R2
N+3
I21
D3
M03
M13
R3
N+4
I22
D4
M04
M14
R4
N+5
I23
D5
M05
M15
R5
N+6
I24
D6
M06
M16
R6
N+7
I25
D7
M07
M17
R7
N+8
I26
D8
M100
M110
M120
M130
M140
M150
M160
M170
R8
N+9
I27
D9
M101
M111
M121
M131
16K x 1
STATIC NMOS
MEMORY CHIPS
M171
R9
N+10
I28
D10
M102
M112
R10
N+11
I29
D11
M103
M113
R11
N+12
I30
D12
M104
M114
R12
N+13
I31
D13
M105
M115
R13
N+14
I32
D14
M106
M116
R14
N+15
I33
D15
M107
M117
R15
N+16
I34
D16
M08
M18
M28
M38
M48
M58
M68
M78
R16
N+17
I35
D17
M108
M118
M128
M138
M148
M158
M168
M178
R17
ECL TO TTL
CONVERTER'S
ECL TO TTL
I0-I13
A0-A13
INTERNAL ADDRESS TO ALL
MEMORY CHIPS
J CHIPS
1/4
FANOUTS
FOR ADDRS
WITHIN
THE BANK
137
138
139
140
141
R18
R19
R20
R21
R22
ECL TO TTL
I36
E0
WRITE ENABLE TO ALL CHIPS
A-2162

CRAY X-MP/22,24
12 COLUMN
SECTION

Below is a brief description of the modules used in the Memory Addressing section.

3AD Module

The 3AD module selects the memory address from the I/O or CPU section and performs operand range error checks, then passes it on to the 3AX module.

If an operand range error or a program range error, from the 3AF module, is detected, an abort section signal will be generated.

The 3AD module also generates the go write to the selected section during the write operation.

The 3AD module also holds the data base address and data limit address.

3AX Module

The 3AX module passes bits 2^5-2^{21} of the memory address to the 3ZA module, for 32 bank operation. For 16 bank operation the memory address bits 2^4-2^{20} are shifted left one bit to make it 2^5-2^{21} and then passed to the 3ZA module.

The 3AX also delays the address bits and sends bit 2^0-2^4 and 2^{17}-2^{21}, no internal chip select bits, to the 3AE module, for memory error logging.

During an I/O high speed and very high speed memory read operation the 3AX receives the check bits and syndrome character and corrects the check byte before sending it to the 3DD module.

3ZA Module

The 3ZA module is a Memory Storage Module and also does 1 to 18 address fanout for address bits 2^5-2^{21}, to the selected bank.

Each 3ZA module handles one address bit and is test pointed on the output term.

X2002S0419

Below is a brief description of the modules used in the memory control section.

3AQ Module

The 3AQ modules are responsible for resolving memory access conflicts. The three types of memory access conflicts that can occur are: bank busy, simultaneous bank and section busy.

Once a memory access conflict is resolved or if none exists the 3AQ module will generate the go bank signal.

3ZF Module

The 3ZF module is a Memory Storage Module, with control fanout on it. It accomplishes 1 to 36 fanout for go read and go bank signals, and 1 to 8 fanout for go fetch, go write and abort signals.

3ZS Module

The 3ZS module is a Memory Storage Module with bank control and write enable fanout (1-18) on it.

When the 3ZS module receives the abort signal, it disables the go read and write enable signals which aborts the memory reference.

The go fetch signal comes into the 3ZS modules to determine which CPU receives the data.

X2002S0418

Below is a brief description of the modules used in the Memory Input data path and check bit generation.

3AE Module

The 3AE modules select the memory data from the I/O or CPU section and does the check bit generation on it.

Each 3AE modules handles one parcel of the memory word and generates two of the check bits.

3ZD Module

The 3ZD modules is a Memory Storage Module, that also does 1-18 data fanout.

Each 3ZD module handles eight bits of data or eight check bits.

X2002S0420

Below is a brief description of the modules involved in the Memory Output data and Error Correction.

3ZR Modules

The 3ZR modules are Memory storage modules, that also accomplishes 8 to 1 fan in of the Read Data from the Memory modules.

Each 3ZR module handles Four (4) bits of data or check bits.

3ZE Modules

The 3ZE modules perform Error Correction on the Data Read from Memory and generates the corrected data and syndrome character.

The type of Error Correction is SECDED (Single Error Correction Double Error Detection).

Each 3ZE module receives all 64 data bits and 8 check bits to generate the 4 syndrome bits and the 32 data bits it's responsible for.

Note: The 3ZE modules feed the Instruction Buffer (3HR modules) directly.

3VA Module

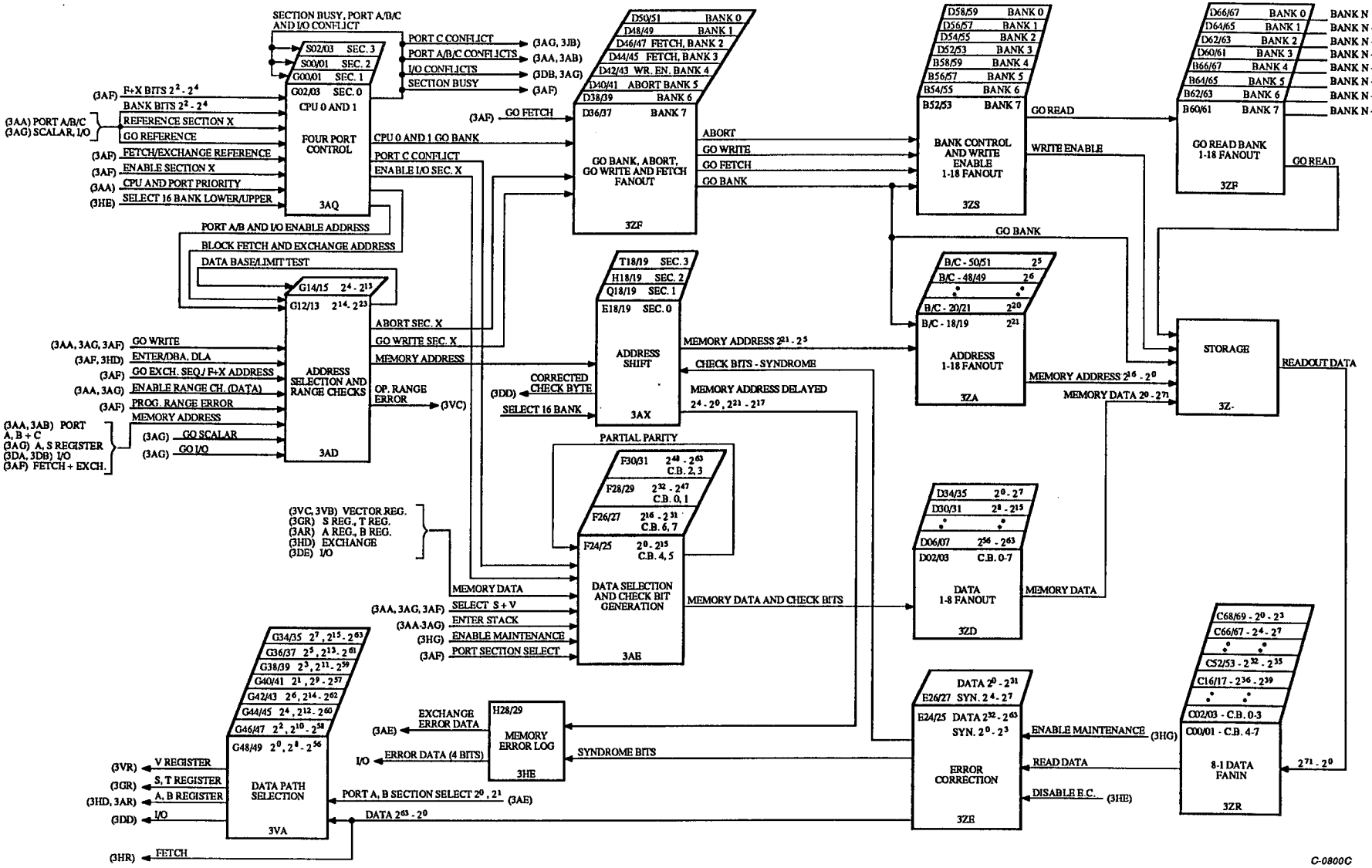
The 3VA module directs the Output data to the selected section of the CPU or I/O.

Each 3VA module handles eight bits of nonconsecutive data.

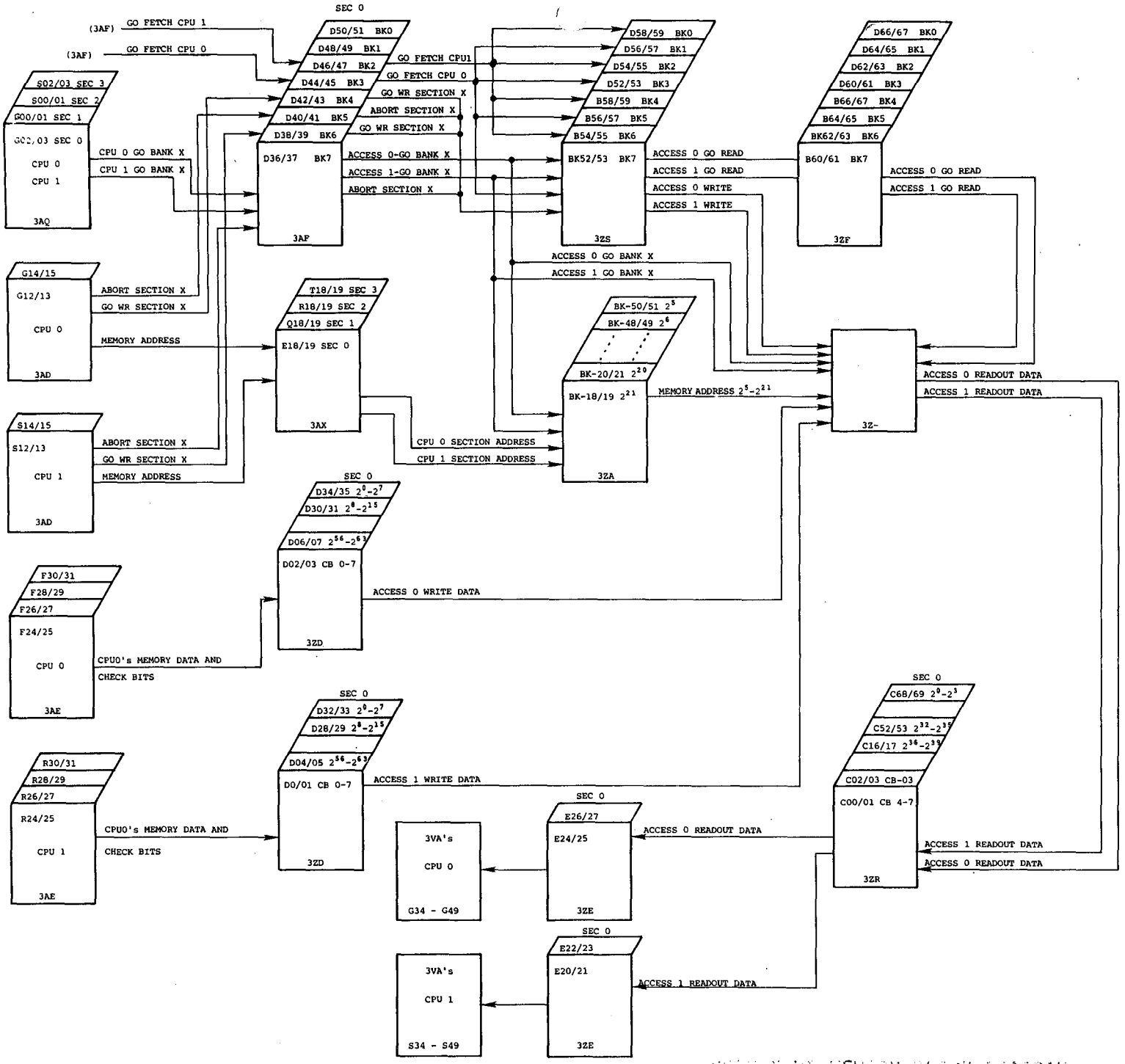
3HE Module

The 3HE module is responsible for generating status information, address (no internal chip) and syndrome to the Error Logger and Exchange Package for a Memory error.

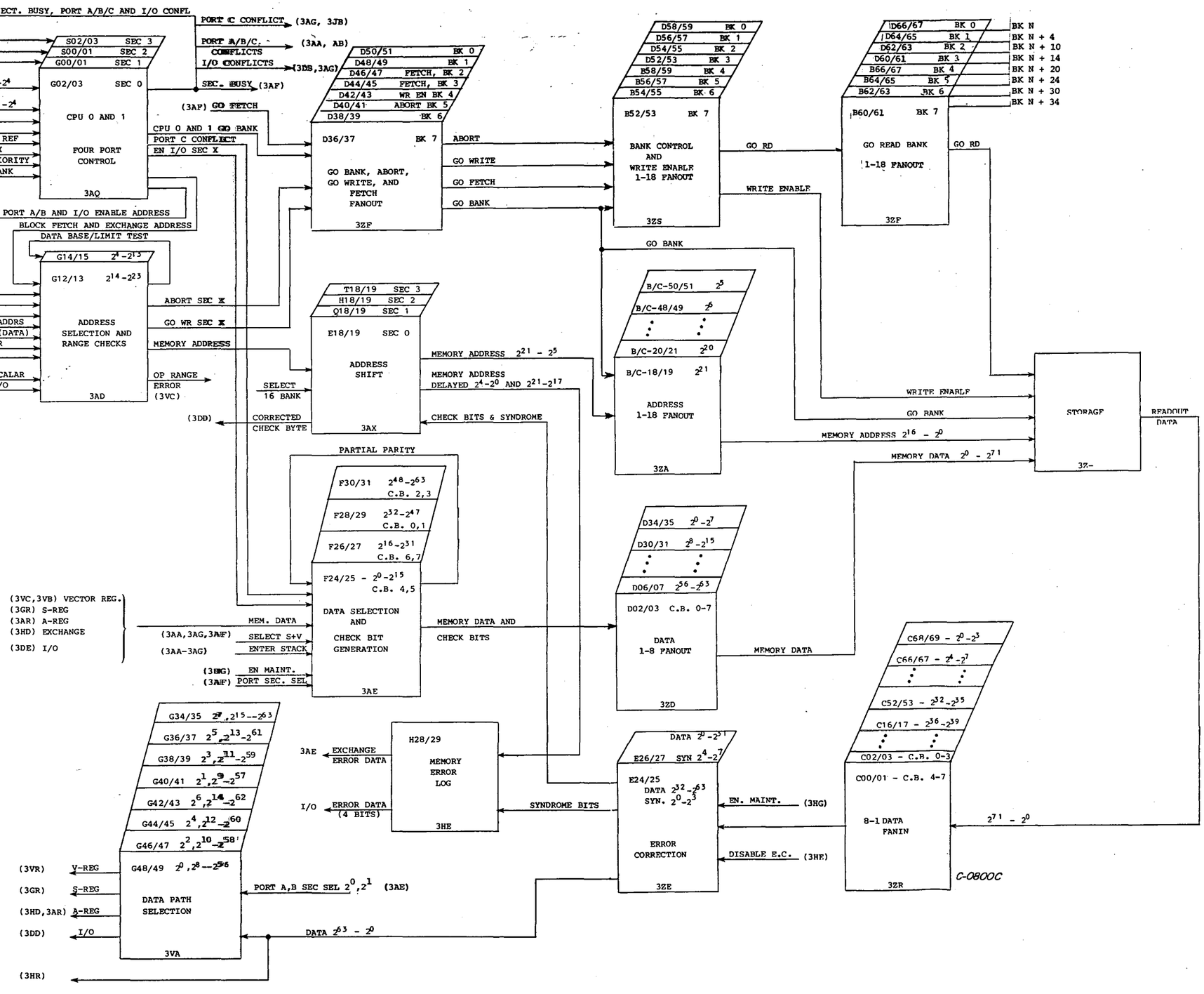
X2002S0422



CRAY X-MP MEMORY BLOCK DIAGRAM (CPU 0 AND SECTION 0)



D-1508



CRAY X-MP memory block diagram (CPU 0 and section 0)

BANK	0	1	2	3	4	5	6	7	10	11	12	13	14	15	16	17	20	21	22	23	24	25	26	27	30	31	32	33	34	35	36	37
CHASSIS	D	P	I	U	D	P	I	U	C	O	J	V	C	O	J	V	B	N	K	W	B	N	K	W	A	M	L	X	A	M	L	X
BITS																																
0-3	70	70	70	70	68	68	68	68	70	70	70	70	68	68	68	68	70	70	70	70	68	68	68	68	70	70	70	70	68	68	68	68
4-7	66	--	--	66	64	--	--	64	66	--	--	66	64	--	--	64	66	--	--	66	64	--	--	64	66	--	--	66	64	--	--	64
8-11	62	--	--	62	60	--	--	60	62	--	--	62	60	--	--	60	62	--	--	62	60	--	--	60	62	--	--	62	60	--	--	60
12-15	58	--	--	58	56	--	--	56	58	--	--	58	56	--	--	56	58	--	--	58	56	--	--	56	58	--	--	58	56	--	--	56
16-19	54	--	--	54	52	--	--	52	54	--	--	54	52	--	--	52	54	--	--	54	52	--	--	52	54	--	--	54	52	--	--	52
20-23	50	--	--	50	48	--	--	48	50	--	--	50	48	--	--	48	50	--	--	50	48	--	--	48	50	--	--	50	48	--	--	48
24-27	46	--	--	46	44	--	--	44	46	--	--	46	44	--	--	44	46	--	--	46	44	--	--	44	46	--	--	46	44	--	--	44
28-31	42	--	--	42	40	--	--	40	42	--	--	42	40	--	--	40	42	--	--	42	40	--	--	40	42	--	--	42	40	--	--	40
32-35	38	--	--	38	36	--	--	36	38	--	--	38	36	--	--	36	38	--	--	38	36	--	--	36	38	--	--	38	36	--	--	36
36-39	34	--	--	34	32	--	--	32	34	--	--	34	32	--	--	32	34	--	--	34	32	--	--	32	34	--	--	34	32	--	--	32
40-43	30	--	--	30	28	--	--	28	30	--	--	30	28	--	--	28	30	--	--	30	28	--	--	28	30	--	--	30	28	--	--	28
44-47	26	--	--	26	24	--	--	24	26	--	--	26	24	--	--	24	26	--	--	26	24	--	--	24	26	--	--	26	24	--	--	24
48-51	22	--	--	22	20	--	--	20	22	--	--	22	20	--	--	20	22	--	--	22	20	--	--	20	22	--	--	22	20	--	--	20
52-55	18	--	--	18	16	--	--	16	18	--	--	18	16	--	--	16	18	--	--	18	16	--	--	16	18	--	--	18	16	--	--	16
56-59	14	--	--	14	12	--	--	12	14	--	--	14	12	--	--	12	14	--	--	14	12	--	--	12	14	--	--	14	12	--	--	12
60-63	10	--	--	10	8	--	--	8	10	--	--	10	8	--	--	8	10	--	--	10	8	--	--	8	10	--	--	10	8	--	--	8
CB0-3	6	--	--	6	4	--	--	4	6	--	--	6	4	--	--	4	6	--	--	6	4	--	--	4	6	--	--	6	4	--	--	4
CB4-7	2	--	--	2	0	--	--	0	2	--	--	2	0	--	--	0	2	--	--	2	0	--	--	0	2	--	--	2	0	--	--	0

B-0602

CRAY X-MP MEMORY LAY-OUT

1:18 ADDRESS BIT FANOUTS - 3ZA MODULES

Bit	3ZA	3ZA
2^{16}	C18	B18
2^{15}	C20	B20
2^{14}	C22	B22
2^{13}	C24	B24
2^{12}	C26	B26
2^{11}	C28	B28
2^{10}	C30	B30
2^{9}	C32	B32
2^{8}	C34	B34
2^{7}	C36	B36
2^{6}	C38	B38
2^{5}	C40	B40
2^{4}	C42	B42
2^{3}	C44	B44
2^{2}	C46	B46
2^{1}	C48	B48
2^{0}	C50	B50

BANKS IN SECTION 0

	R10	R11	R12	R13	R10	R11	R12	R13
T.P.	$\overline{A63}$	$\overline{C72}$	D47	C15	$\overline{A63}$	$\overline{C72}$	D47	C15
		feeds 3Z- @				feeds 3Z- @		
	BK0	BK1	BK2	BK3	BK4	BK5	BK6	BK7
	DO2	D00	C02	C00	B02	B00	A02	A00
	D06	D04	C06	C04	B06	B04	A06	A04
	D10	D08	C10	C08	B10	B08	A10	A08
	D14	D12	C14	C12	B14	B12	A14	A12
	D18	D16	C18	C16	B18	B16	A18	A16
	D22	D20	C22	C20	B22	B20	A22	A20
	D26	D24	C26	C24	B26	B24	A26	A24
	D30	D28	C30	C28	B30	B28	A30	A28
	D34	D32	C34	C32	B34	B32	A34	A32
	D38	D36	C38	C36	B38	B36	A38	A36
	D42	D40	C42	C40	B42	B40	A42	A40
	D46	D44	C46	C44	B46	B44	A46	A44
	D50	D48	C50	C48	B50	B48	A50	A48
	D54	D52	C54	C52	B54	B52	A54	A52
	D58	D56	C58	C56	B58	B56	A60	A56
	D62	D60	C62	C60	B62	B60	A64	A60
	D66	D64	C66	C64	B66	B64	A66	A64
	D70	D68	C70	C68	B70	B68	A70	A68
Absolute BK	0	4	10	14	20	24	30	34

XV2S01M

8 TO 1 FANIN OF MEMORY READ DATA

BIT POSITION	CB 4-7	CB 0-3	2^{60}-2^{63}	2^{56}-2^{59}	2^{52}-2^{55}	2^{48}-2^{51}	2^{44}-2^{47}	2^{40}-2^{43}	2^{36}-2^{39}
3ZR LOCATION	C01	C03	C05	C07	C09	C11	C13	C15	C17
FROM	A01	A05	A09	A13	A17	A21	A25	A29	A33
	A03	A07	A11	A15	A19	A23	A27	A31	A35
	B01	B05	B09	B13	B17	B21	B25	B29	B33
	B03	B07	B11	B15	B19	B23	B27	B31	B35
	C01	C05	C09	C13	C17	C21	C25	C29	C33
	C03	C07	C11	C15	C19	C23	C27	C31	C35
	D01	D05	D09	D13	D17	D21	D25	D29	D33
	D03	D07	D11	D15	D19	D23	D27	D31	D35

BIT POSITION	2^{32}-2^{35}	2^{28}-2^{31}	2^{24}-2^{27}	2^{20}-2^{23}	2^{16}-2^{19}	2^{12}-2^{15}	2^{8}-2^{11}	2^{4}-2^{7}	2^{0}-2^{3}
3ZR LOCATION	C53	C55	C57	C59	C61	C63	C65	C67	C69
FROM	A37	A41	A45	A49	A53	A57	A61	A65	A69
	A39	A43	A47	A51	A55	A59	A63	A67	A71
	B37	B41	B45	B49	B53	B57	B61	B65	B69
	B39	B43	B47	B51	B55	B59	B63	B67	B71
	C37	C41	C45	C49	C53	C57	C61	C65	C69
	C39	C43	C47	C51	C55	C59	C63	C67	C71
	D37	D41	D45	D49	D53	D57	D61	D65	D69
	D39	D43	D47	D51	D55	D59	D63	D67	D71

3ZD

1 TO 8 WRITE DATA FANOUT

MODULE LOCATION	D02/D03								D06/D07								D10/D11							
BIT POSITION	CB 7	CB 6	CB 5	CB 4	CB 3	CB 2	CB 1	CB 0	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}
R TERMS	27 17	26 16	25 15	24 14	23 13	22 12	21 11	20 10	27 17	26 16	25 15	24 14	23 13	22 12	21 11	20 10	27 17	26 16	25 15	24 14	23 13	22 12	21 11	20 10
FEEDS	B01 A01	B01 A01	B01 A01	B01 A01	B05 A05	B05 A05	B05 A05	B05 A05	B09 A09	B09 A09	B09 A09	B09 A09	B13 A13	B13 A13	B13 A13	B13 A13	B17 A17	B17 A17	B17 A17	B17 A17	B21 A21	B21 A21	B21 A21	B21 A21
	B03 A03	B03 A03	B03 A03	B03 A03	B07 A07	B07 A07	B07 A07	B07 A07	B11 A11	B11 A11	B11 A11	B11 A11	B15 A15	B15 A15	B15 A15	B15 A15	B19 A19	B19 A19	B19 A19	B19 A19	B23 A23	B23 A23	B23 A23	B23 A23
	C01	C01	C01	C01	C05	C05	C05	C05	C09	C09	C09	C09	C13	C13	C13	C13	C17	C17	C17	C17	C21	C21	C21	C21
	C03	C03	C03	C03	C07	C07	C07	C07	C11	C11	C11	C11	C15	C15	C15	C15	C19	C19	C19	C19	C23	C23	C23	C23
	D01	D01	D01	D01	D05	D05	D05	D05	D09	D09	D09	D09	D13	D13	D13	D13	D17	D17	D17	D17	D21	D21	D21	D21
	D03	D03	D03	D03	D07	D07	D07	D07	D11	D11	D11	D11	D15	D15	D15	D15	D19	D19	D19	D19	D23	D23	D23	D23

MODULE LOCATION	D14/D15								D18/D19								D22/D23							
BIT POSITION	2^{47}	2^{26}	2^{45}	2^{44}	2^{43}	2^{42}	2^{41}	2^{40}	2^{39}	2^{38}	2^{37}	2^{36}	2^{35}	2^{34}	2^{33}	2^{32}	2^{31}	2^{30}	2^{29}	2^{28}	2^{27}	2^{26}	2^{25}	2^{24}
R TERMS	27 17	26 16	25 15	24 14	23 13	22 12	21 11	20 10	27 17	26 16	25 15	24 14	23 13	22 12	21 11	20 10	27 17	26 16	25 15	24 14	23 13	22 12	21 11	20 10
FEEDS	B25 A25	B25 A25	B25 A25	B25 A25	B29 A29	B29 A29	B29 A29	B29 A29	B33 A33	B33 A33	B33 A33	B33 A33	B37 A37	B37 A37	B37 A37	B37 A37	B41 A41	B41 A41	B41 A41	B41 A41	B45 A45	B45 A45	B45 A45	B45 A45
	B27 A27	B27 A27	B27 A27	B27 A27	B31 A31	B31 A31	B31 A31	B31 A31	B35 A35	B35 A35	B35 A35	B35 A35	B39 A39	B39 A39	B39 A39	B39 A39	B43 A43	B43 A43	B43 A43	B43 A43	B47 A47	B47 A47	B47 A47	B47 A47
	C25	C25	C25	C25	C29	C29	C29	C29	C33	C33	C33	C33	C37	C37	C37	C37	C41	C41	C41	C41	C45	C45	C45	C45
	C27	C27	C27	C27	C31	C31	C31	C31	C35	C35	C35	C35	C39	C39	C39	C39	C43	C43	C43	C43	C47	C47	C47	C47
	D25	D25	D25	D25	D29	D29	D29	D29	D33	D33	D33	D33	D37	D37	D37	D37	D41	D41	D41	D41	D45	D45	D45	D45
	D27	D27	D27	D27	D31	D31	D31	D31	D35	D35	D35	D35	D39	D39	D39	D39	D43	D43	D43	D43	D47	D47	D47	D47

MODULE LOCATION	D26/D27								D30/D31								D34/D35							
BIT POSITION	2^{23}	2^{22}	2^{21}	2^{20}	2^{19}	2^{18}	2^{17}	2^{16}	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}
TERMS	27 17	26 16	25 15	24 14	23 13	22 12	21 11	20 10	27 17	26 16	25 15	24 14	23 13	22 12	21 11	20 10	27 17	26 16	25 15	24 14	23 13	22 12	21 11	20 10
FEEDS	B49 A49	B49 A49	B49 A49	B49 A49	B53 A53	B53 A53	B53 A53	B53 A53	B57 A57	B57 A57	B57 A57	B47 A57	B61 A61	B61 A61	B61 A61	B61 A61	B65 A65	B65 A65	B65 A65	B65 A65	B69 A69	B69 A69	B69 A69	B69 A69
	B51 A51	B51 A51	B51 A51	B51 A51	B55 A55	B55 A55	B55 A55	B55 A55	B59 A59	B59 A59	B59 A59	B59 A59	B63 A63	B63 A63	B63 A63	B63 A63	B67 A67	B67 A67	B67 A67	B67 A67	B17 A17	B71 A71	B71 A71	B71 A71
	C49	C49	C49	C49	C53	C53	C53	C53	C57	C57	C57	C57	C61	C61	C61	C61	C65	C65	C65	C65	C69	C69	C69	C69
	C51	C51	C51	C51	C55	C55	C55	C55	C59	C59	C59	C59	C63	C63	C63	C63	C67	C67	C67	C67	C71	C71	C71	C71
	D49	D49	D49	D49	D53	D53	D53	D53	D57	D57	D57	D57	D61	D61	D61	D61	D65	D65	D65	D65	D69	D69	D69	D69
	D51	D51	D51	D51	D55	D55	D55	D55	D59	D59	D59	D59	D63	D63	D63	D63	D67	D67	D67	D67	D71	D71	D71	D71

TWO

FETCH OPERATION OVERVIEW

A. Means of loading an Instruction Buffer.

B. Four Fetch's (Memory References) per Fetch Operation.

1. Eight words fetched from memory per clock period.
2. 32 words total in 4 clock periods.

C. Fetch Timing

1. Branch out of buffer is 16 clock periods for ECL Memory 32 Bank and 18 clock periods for 16 Bank.
2. Branch out of buffer is 19 clock periods for MOS Memory 32 Bank and 25 clock periods for 16 Bank.

MODULES USED

3AF MODULE

This is the Fetch Control Module. It will control the addressing of memory and all the control signals needed for the Fetch Operation. (1/CPU)

3HR MODULE

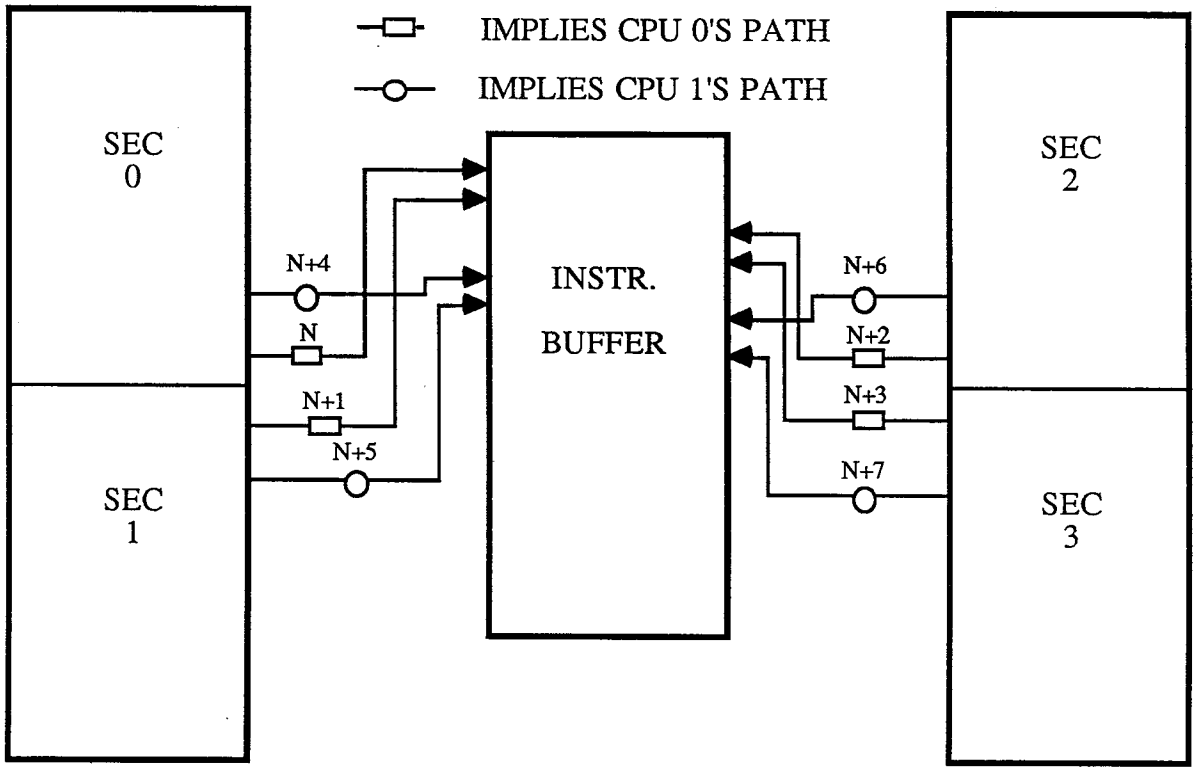
This is the Instruction Buffer Module. Each one will handle 16 bits of the memory word. Each module contains one parcel of all four instruction buffers, for both CPU's. (4 used by both CPU's)

3HA MODULE

This is the Program Counter Module. It contains the Program Counter Register (P-Reg.), Instruction Buffer Address Registers (IBAR), Buffer Counter, and the IBAR and P-Reg. Comparator. (1/CPU)

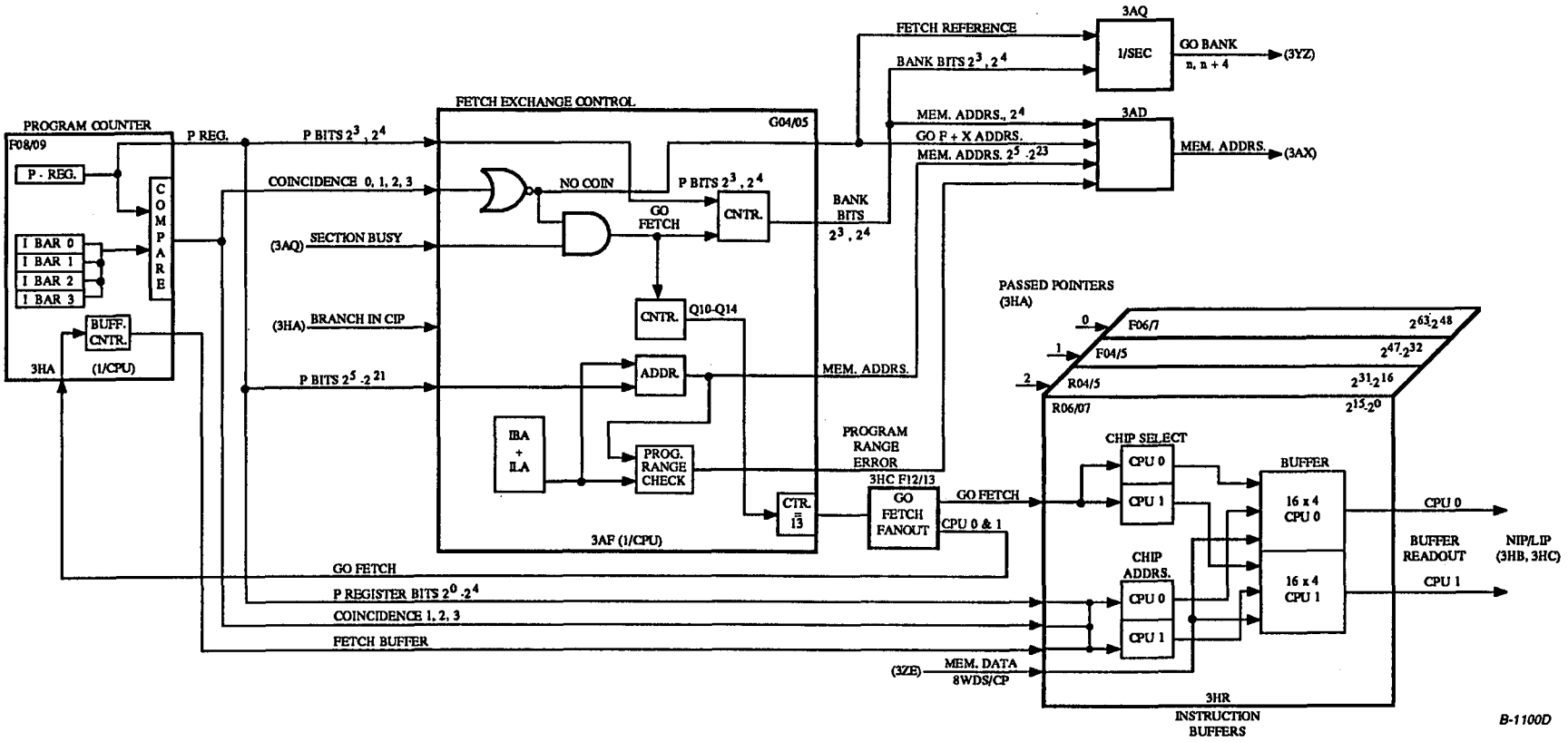
FETCH DATA FLOW DIAGRAM

Below is found a simplified diagram showing how the data is loaded from memory to the Instruction Buffer, during a Fetch Operation.

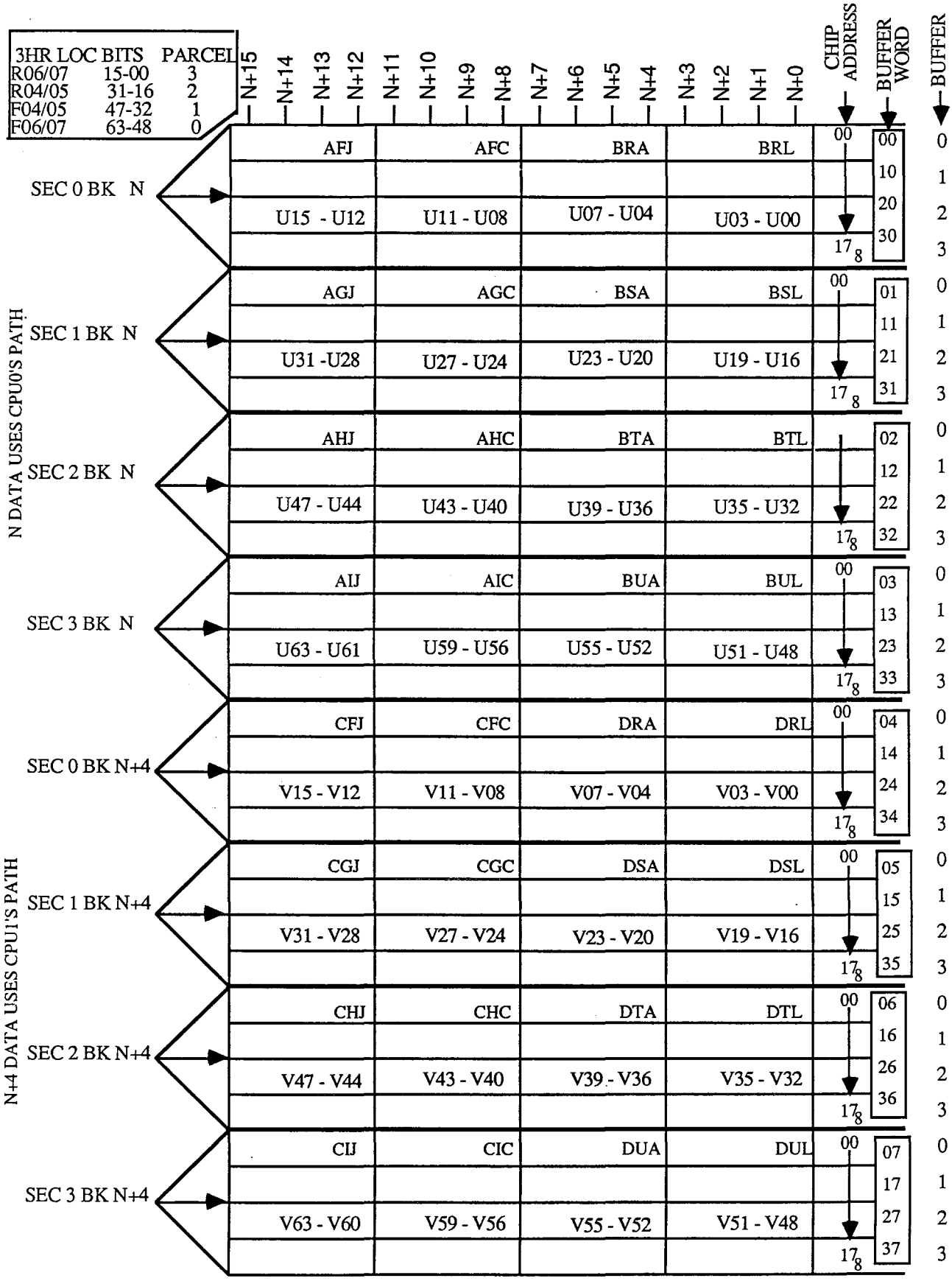


NOTE: N CAN EQUAL 0, 10, 20, OR 30.

Hardware Trng.
A-5899 R.W.A.

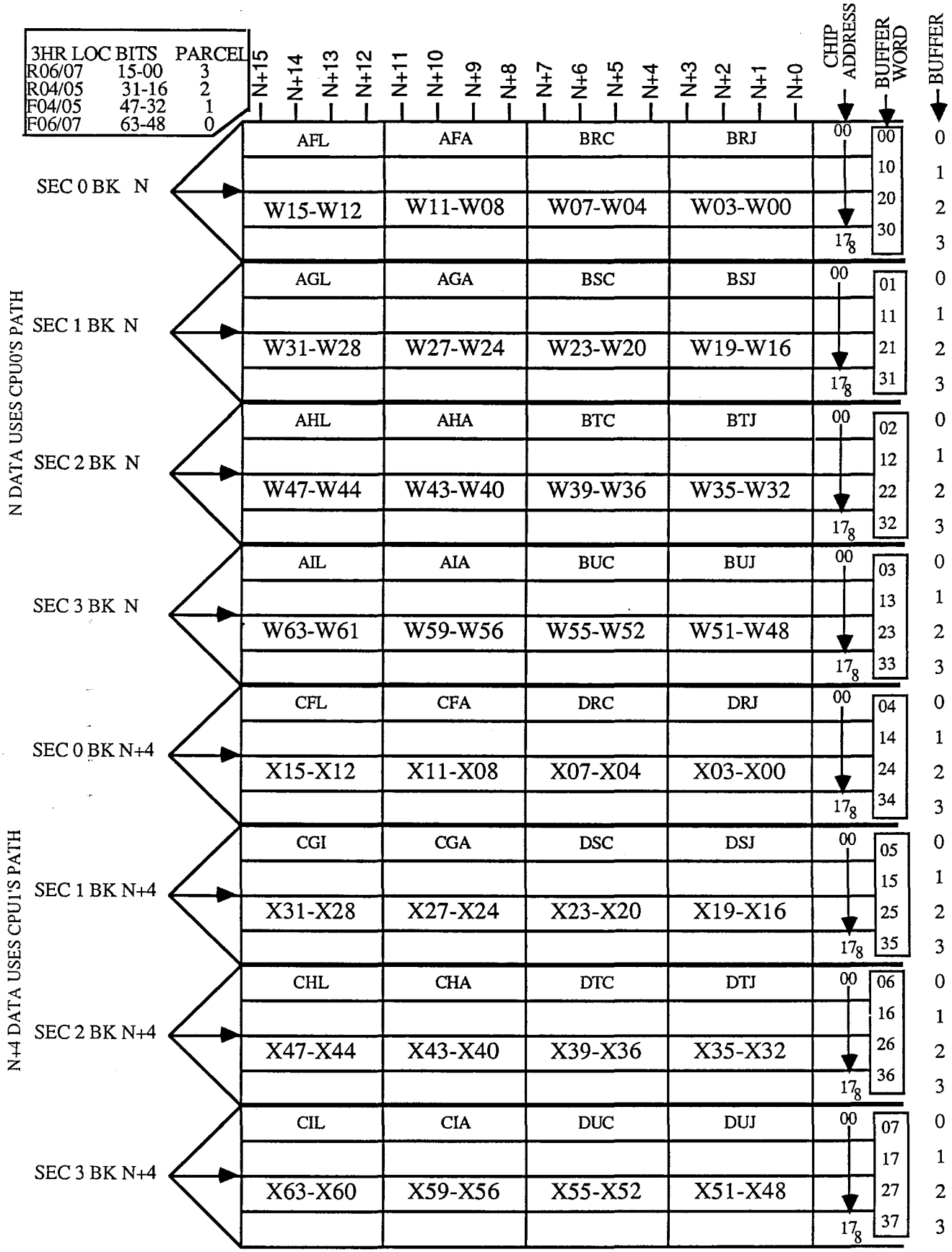


CRAY X-MP FETCH OPERATION CONTROL (CPU 0 32 BANK)



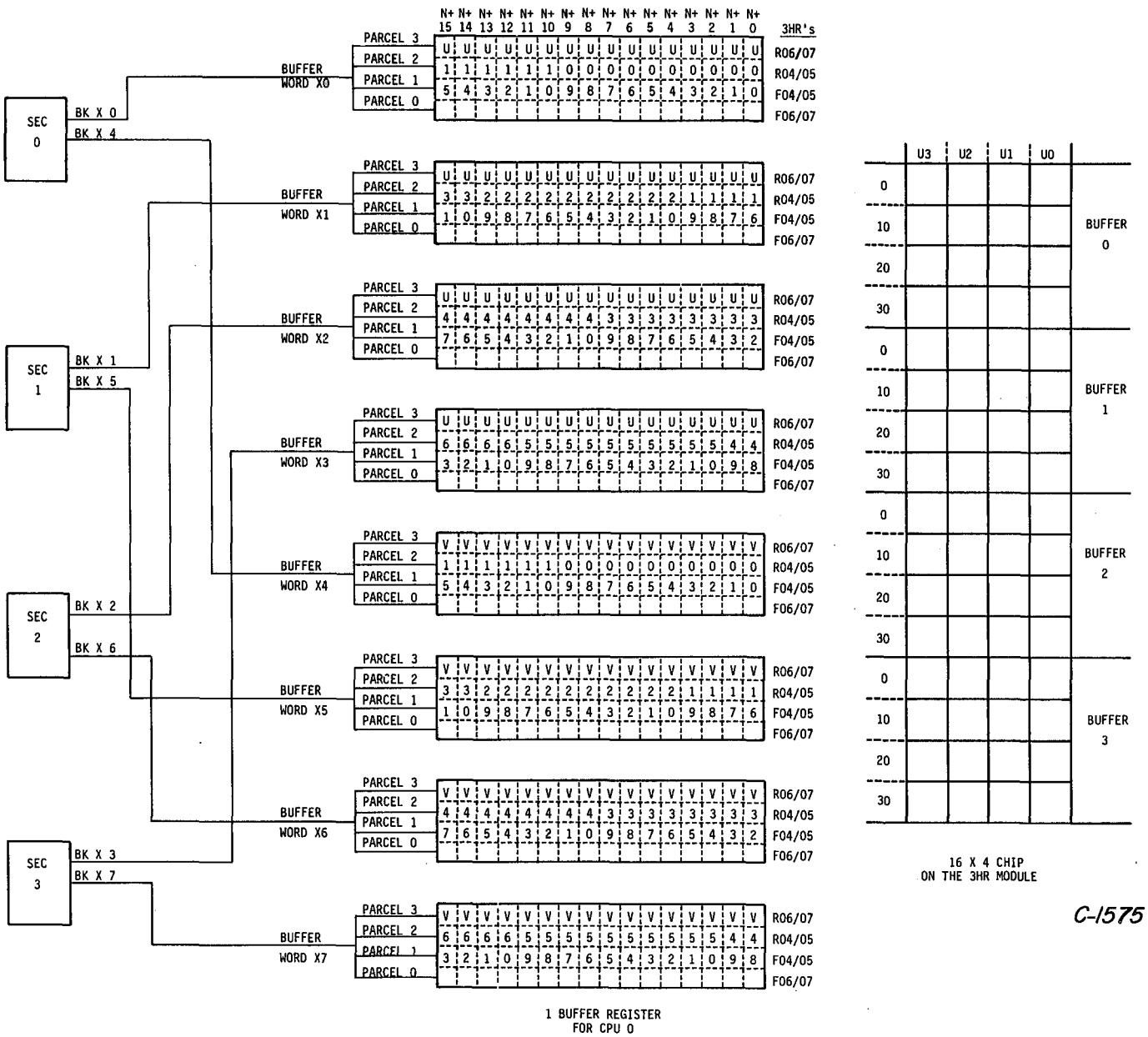
3HR MODULE-INSTRUCTION BUFFER LAYOUT (CPU0)

rwa



3HR MODULE-INSTRUCTION BUFFER LAYOUT (CPU1)

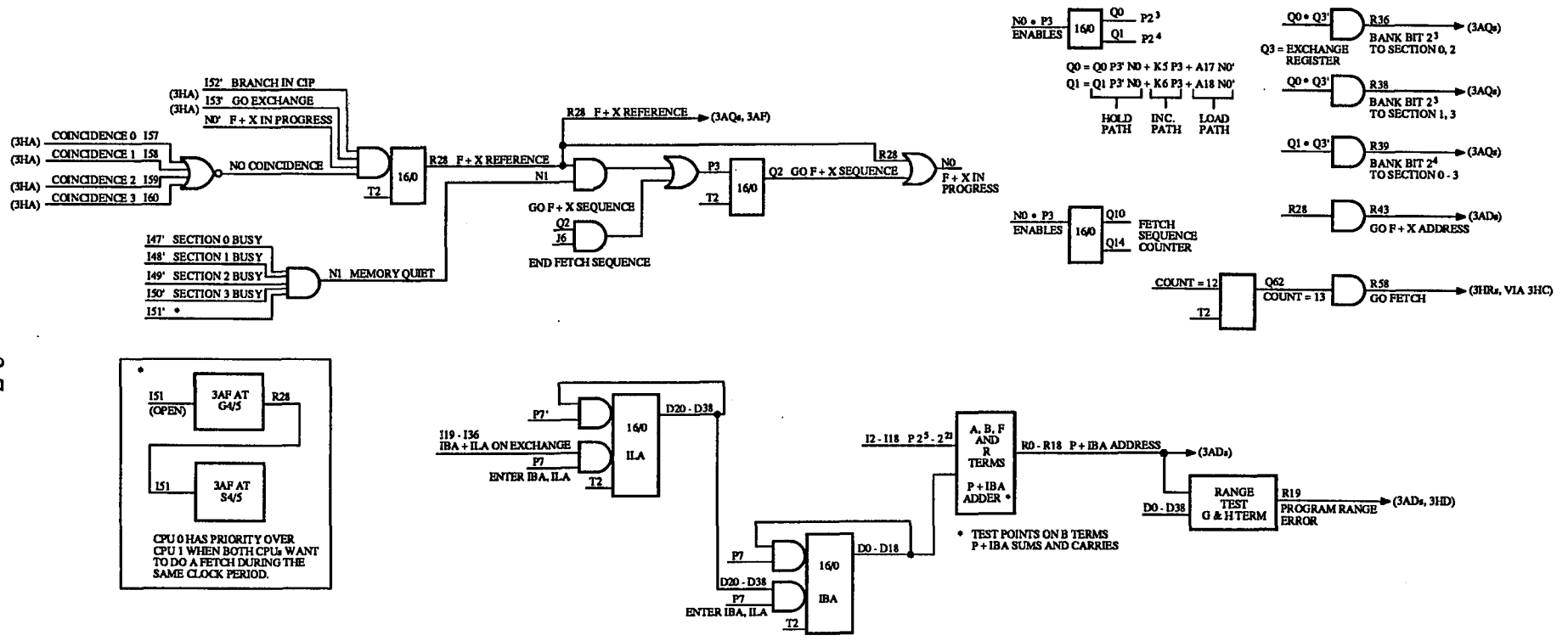
rwa



NOTE: PARCEL 3 = BITS 00-15
PARCEL 2 = BITS 16-31
PARCEL 1 = BITS 32-47
PARCEL 0 = BITS 48-63

NOTE: FOR CPU 1, THE TERMS WILL BE A W INSTEAD OF A U AND A T INSTEAD OF A V.

FETCH D A TRANSFER



CRAY X-MP 3AF MODULE FETCH CONTROL

A/B REGISTER RESERVATIONS:

1) A registers are reserved as RESULT registers, but NOT as operand registers.

2) All B registers are reserved during block transfers (034-035), since you can start with any B register and wrap around.

3) On B register memory references, Port A is reserved for a read (034), and Port C is reserved for a write (035).

A/B REGISTER HOLD ISSUE CONDITIONS:

1) When the A register needed for a result is reserved as a result.

2) On an A register memory reference (10X-11X), if ANY one of Ports A, B, C is busy; or if there is a 10X-13X in CP2 and CP3 and a conflict; or if Ah was busy during the previous CP.

3) On a B register block transfer (034-035), if A0 is busy; or if there is a scalar memory reference (10X-13X) in CP 0, 1, 2, or 3; or, for an 034 instruction, if Port A is busy; or, for an 035 instruction, if Port C is busy.

4) On an 01hijkm instruction, if Ah was busy during the previous CP.

5) On a 005, 007, 024-025, or 034-035 instruction, if the B registers are reserved.

6) On a jump instruction (010-013), if A0 was busy in any of the 3 previous CP's.

7) For a 033 followed by a 027 instruction, or a 025 followed by a 024 instruction, hold issue for 1 CP.

8) On an 005 instruction, hold issue for 2 CP's.

9) If the second parcel of a two-parcel instruction is in a different buffer (2 CP delay), or not in any buffer.

S/T REGISTER RESERVATIONS:

1) S registers are reserved as RESULT registers, but NOT as operand registers.

2) All T registers are reserved during block transfers (036-037), since you can start with any T register and wrap around.

3) On T register memory references, Port B is reserved for a read (036), and Port C is reserved for a write (037).

S/T REGISTER HOLD ISSUE CONDITIONS:

1) When the S register needed for a result is reserved as a result.

2) When the Floating Point Functional Unit needed is reserved by the Vector registers.

3) On an S register memory reference (12X-13X), if ANY one of Ports A, B, C is busy; or if there is a 10X-13X in CP2 and CP3 and a conflict; or if Ah was busy during the previous CP.

4) On a T register block transfer (036-037), if A0 is busy; or if there is a scalar memory reference (10X-13X) in CP 0, 1, 2, or 3; or, for an 036 instruction, if Port B is busy; or, for an 037 instruction, if Port C is busy.

5) On an 074-075, or 036-037 instruction, if the T registers are reserved.

6) On a single shift instruction (052-053), if S0 is busy.

7) On a jump instruction (014-017), if S0 was busy in ANY of the 3 previous CP's.

8) For an 056-061 followed by an 052-055 instruction, or an 075 followed by an 074 instruction, hold issue for 1 CP.

9) If the second parcel of a two-parcel instruction is in a different buffer (2 CP delay), or NOT in any buffer.

VECTOR REGISTER RESERVATIONS:

1) V registers are reserved either as RESULT registers or as OPERAND registers. A register that is reserved as a RESULT register may be used as an OPERAND register in subsequent instructions (CHAINING).

2) All functional units used by vectors are reserved.

3) On a memory read (176), Port B or A is reserved, and on a memory write (177), Port C is reserved.

NOTE: On functional unit instructions,
OPERAND registers are reserved for VL + 3 CP's
Functional Units are reserved for VL + 4 CP's
RESULT registers are reserved for VL + FU + 6 CP's

VECTOR REGISTER HOLD ISSUE CONDITIONS:

1) When the V register needed for an operand is reserved as an operand.

2) When the V register needed for a result is reserved as either an operand or a result.

3) When the Vector F.U. or Floating Point F.U. is reserved.

4) When the S-register needed is reserved.

5) On a 076-077 instruction, if the V register needed is reserved as either an operand or a result.

6) On a memory read (176), if Ports A and B are both busy; and on a memory write (177), if Port C is busy.

7) On a 176-177 instruction, if there is a scalar reference (10X-13X) in CP 0, 1, 2, or 3, or if A0 or Ak is reserved.

8) On a Gather/Scatter instruction (176i1k or 1771jk), if there was another Gather/Scatter in the previous CP, or one in process.

9) For an 0030j0 or 073i00, if there is an 0030j0 or 073i00 in process, hold issue for 1 CP; if there is a 14X or 175 in process, hold issue for (VL) + 5 CP's.

MODULES USED

3HR MODULE

This is the Instruction Buffer Module; each one will handle 16 bits of the memory word. Each module contains one parcel of all four Instruction Buffers. It also contains the Readout Register that presents the instruction to NIP/LIP. (4/CPU)

3HA MODULE

This is the Program Counter and Branch Control Module which will control which word is put out on the Instruction Buffer Readout Register and it also sends out the command to select the correct parcel to be read into NIP/LIP. (1/CPU)

3HB MODULE

The NIP/LIP upper eight bits are contained on this module, which are the G, H, and upper bit of I Fields of the instruction. This module also decodes two parcel instructions so it can help generate the blank NIP signal, and it also is decoding for a 000 or 004 instruction to initiate an Exchange Operation. Along with that the CPU Master Control and Interrupt Signals are fed into this module to also start an exchange. (1/CPU)

3HC MODULE

The NIP/LIP lower eight bits are contained on this module. It will also contain a copy of the CIP data for these bits. (1/CPU)

3JA MODULE

This is the CIP Data Fanout Module. It will be used as a secondary source to assist the other issue control modules for extra copies of G, H, I, J, and K Fields. (1/CPU)

3JF MODULE

This is the Address Register Issue Control Module, which controls the issuing of the A Register Instructions.

3JC MODULE

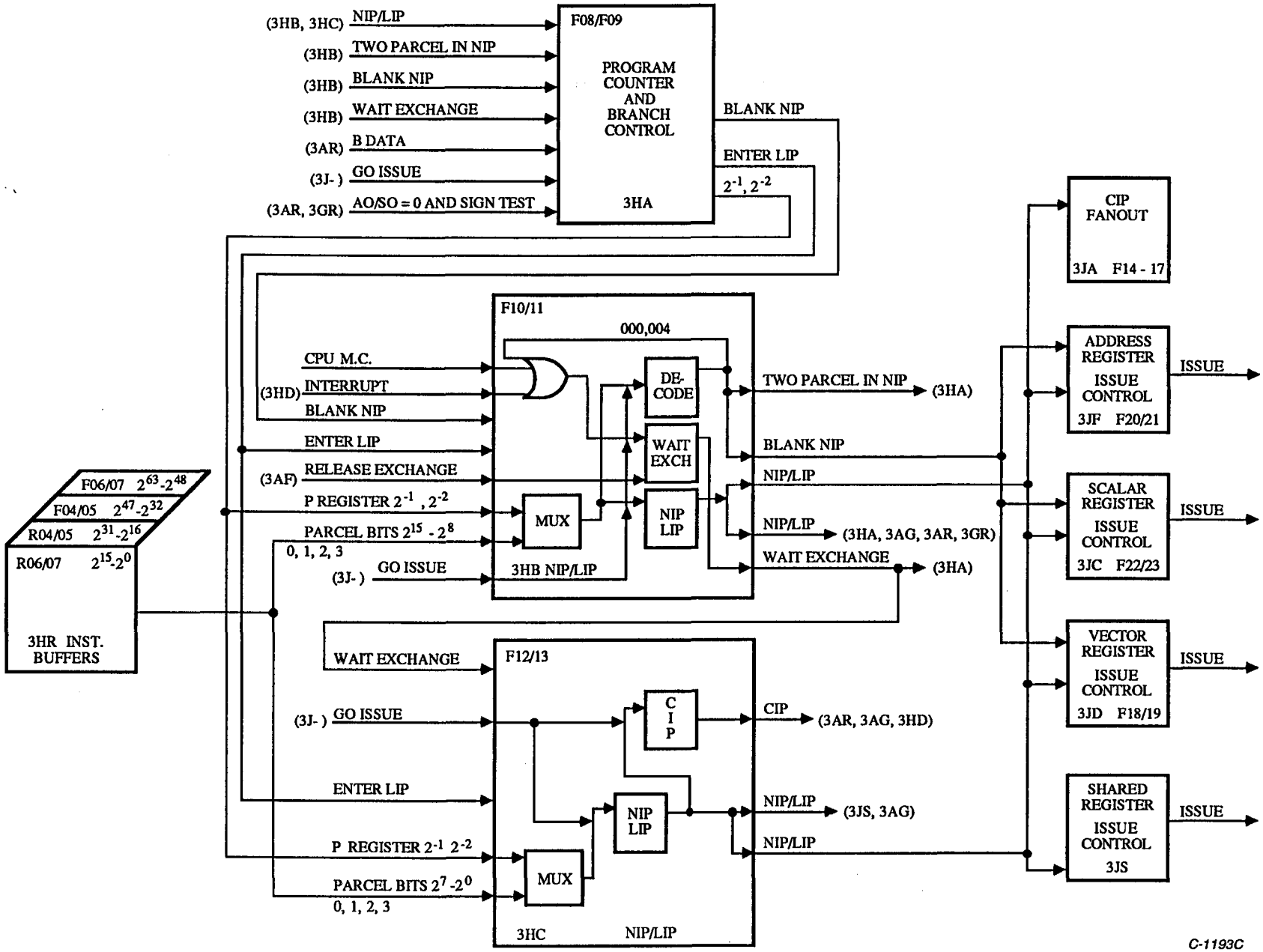
This is the Scalar Register Issue Control Module, which controls the issuing of the S Register instructions.

3JD Module

This is the Vector Register Issue Control Module, which controls the issuing of the V Register Instructions.

3JS MODULE

This is the Shared Register Issue Control Module, which controls the issuing of the Shared Register Instructions.



CRAY X-MP ISSUE (CPU 0)

ADDRESS REGISTERS AND CONTROL

The A register block diagram contains four different module types. They are the 3JA, 3JF, 3AB, and the 3AR modules. Below is a brief description of each modules function.

3JA MODULE

The 3JA module is responsible for CIP fanout. It holds a copy of the current instruction to be issued. It then generates several copies to the 3AR and 3GR modules.

3JF MODULE

The 3JF module is the CIP issue control module for the A register type instructions. The 3JF is responsible for A register reservation control, conflict checking, and A register issue control.

The 3JF module decodes the instruction and determines if the instruction can issue. It does so by comparing what registers the instruction wants to use against what registers may already be reserved. The 3JF also checks for entering data conflict and holds issue if it detects any.

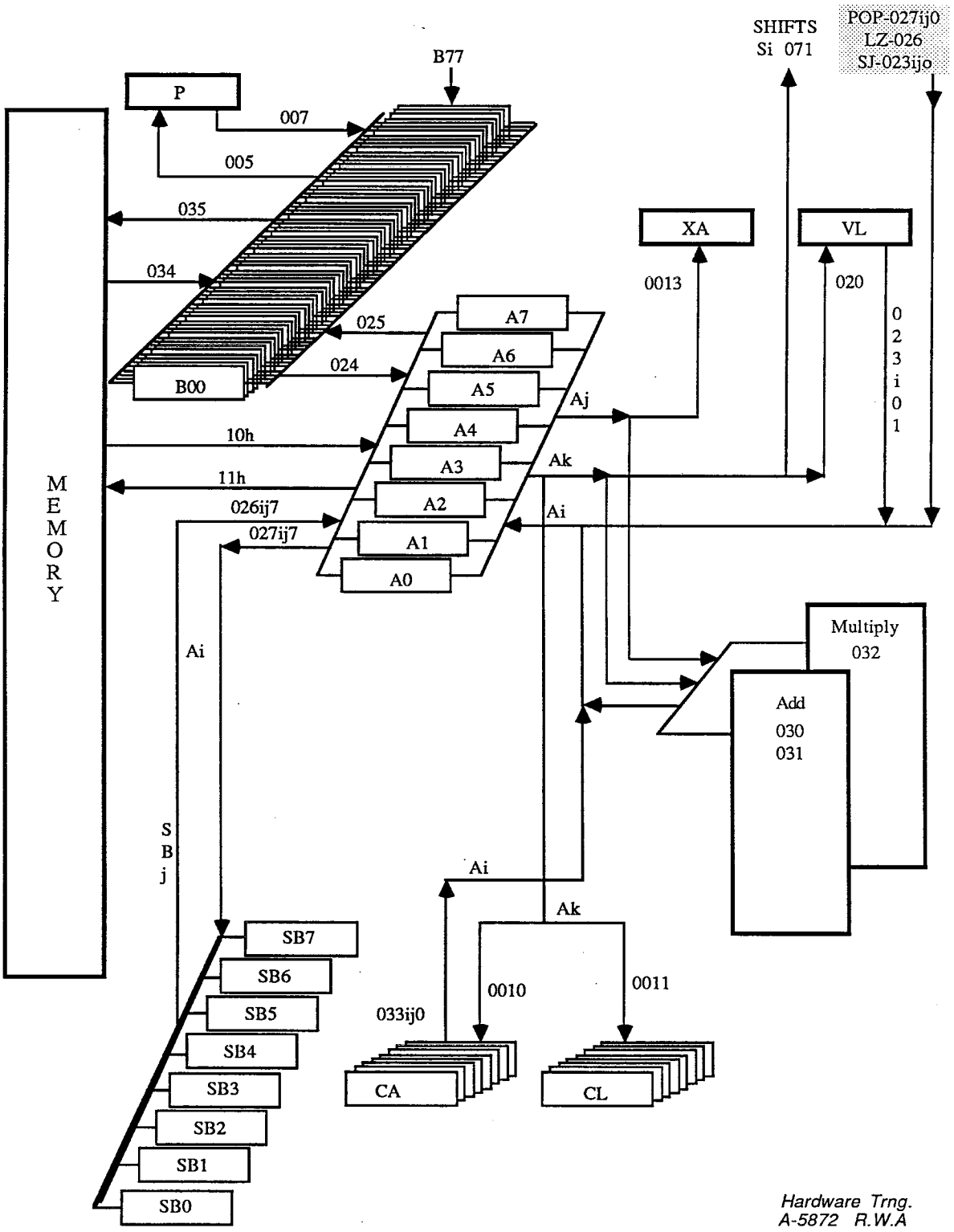
3AB MODULE

The 3AB fans out the source bits and the bypass bits from the 3JF on their way to the 3AR's.

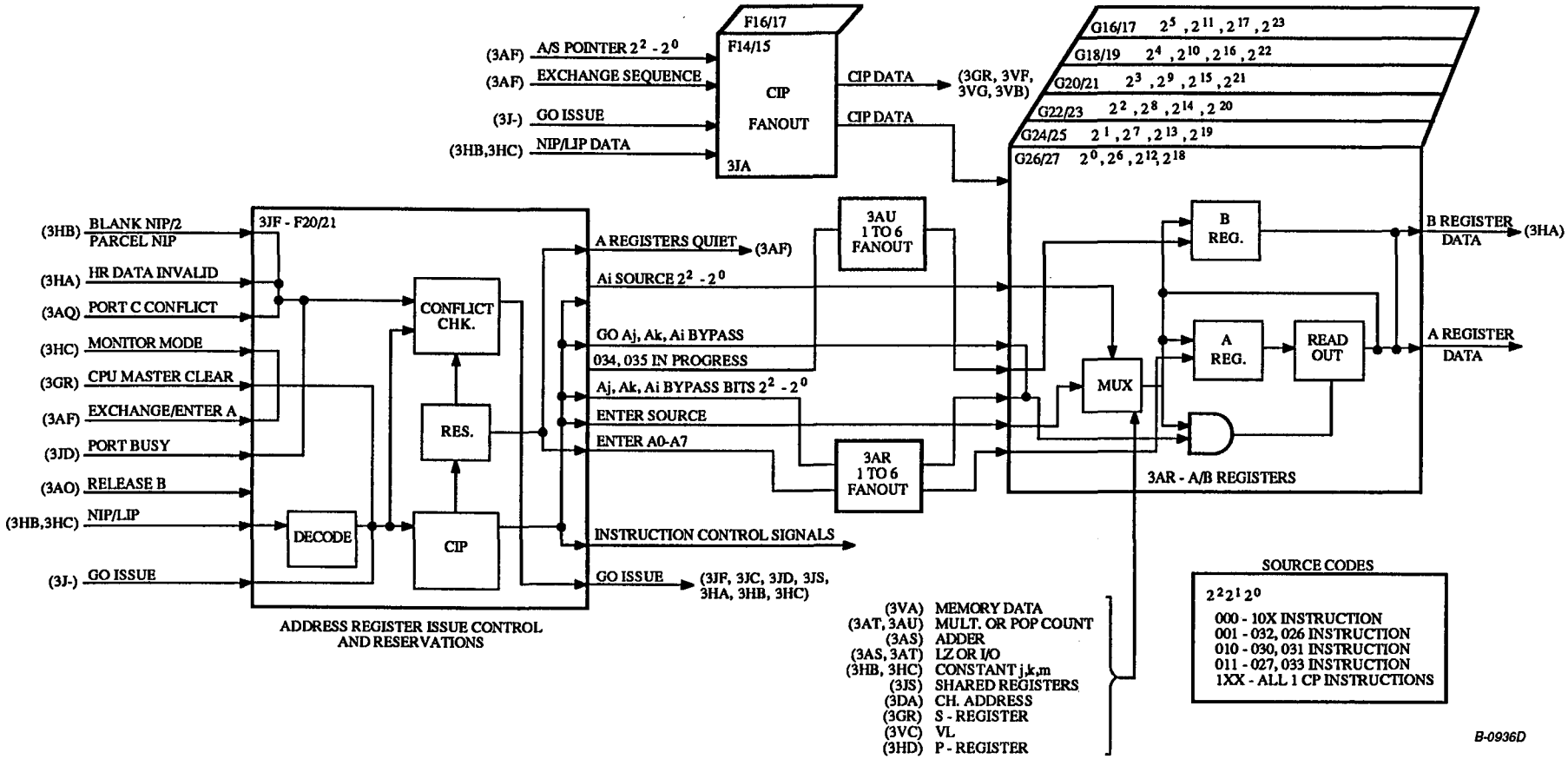
3AR MODULE

The 3AR modules are the A registers. Each 3AR holds bits n, n+6, n+12, and n+18 for all eight A regiters and all 64 B registers. The 3AR also has a multiplexed input path and three readout registers, one each for Ai, Aj, and Ak.

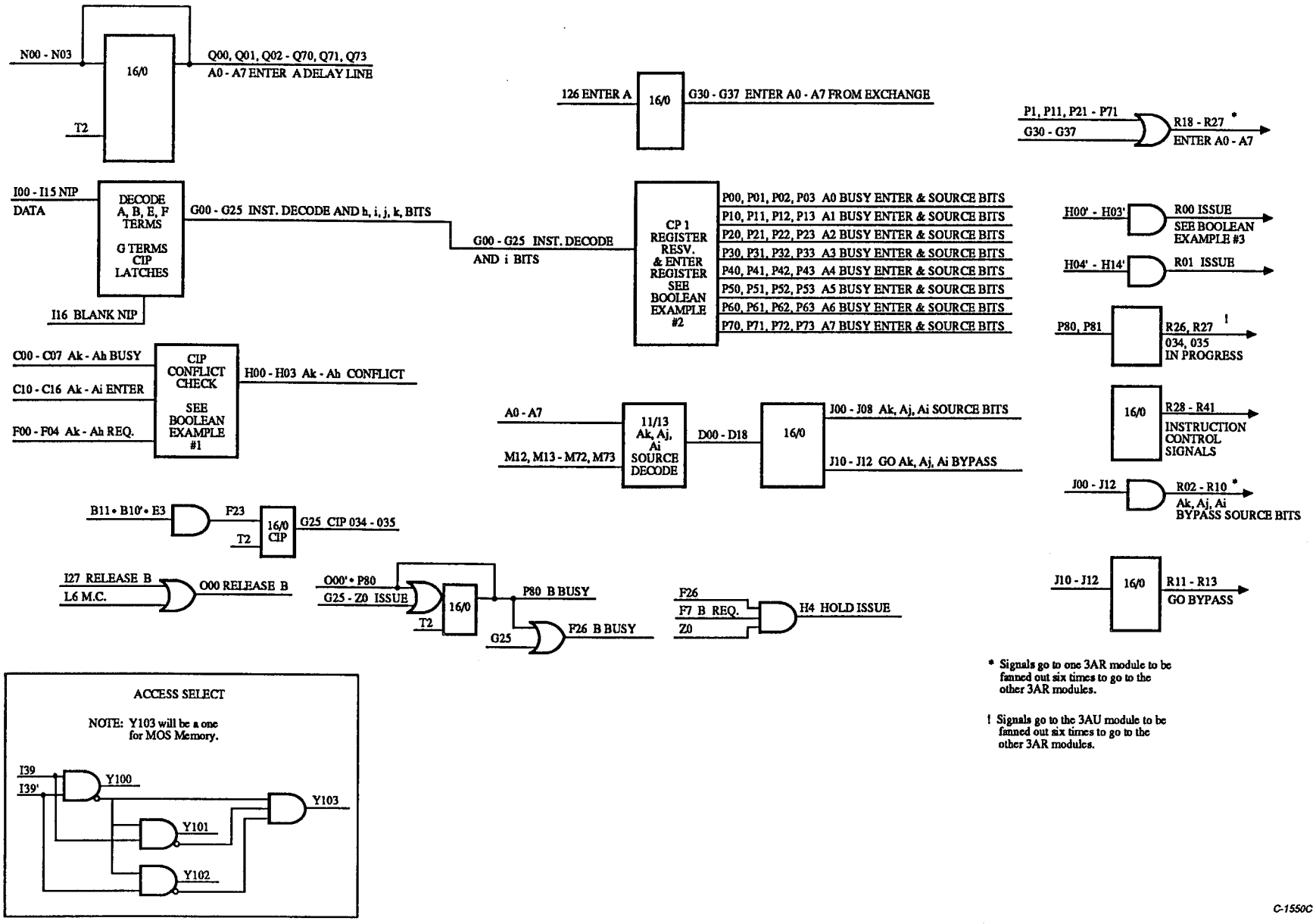
X2005S0203



CRAY X-MP A REGISTERS



CRAY X-MP ADDRESS REGISTERS AND CONTROL BLOCK DIAGRAM (CPU 0)



* Signals go to one 3AR module to be fanned out six times to go to the other 3AR modules.

! Signals go to the 3AU module to be fanned out six times to go to the other 3AR modules.

C-1550C

CRAY X-MP 3JF MODULE

EXAMPLE OF CONFLICT CHECK

```
Boolean  H00' = C10  C0'  C4'  Z0'  F0'  #  #  H0'  F1'  #  C14 ; T2        AK CONFLICT
         H00  = (F0 + F1 + H0) . (Z0 + H0) . (C0 . C4) . (C10' + C14') ; T2
```

AK CONFLICT = (AK REQ + AK CONFLICT) . (ISSUE + AK CONFLICT) . (AK BUSY) . $\overline{\text{(AK ENTER)}}$

```
Boolean  F00 = E0  B8'  E9  +  A15'  E3  B11'     AK REQUIRED
```

AK REQ = (g = 00X . $\overline{\text{i BIT 2}}$. h = xx1) + $\overline{\text{h = x3}}$.g = 03x.$\overline{\text{h BIT 2}}$) = 0010-0013 INST + 030-032 INST

```
Boolean  F01 = E0  E10  +  E16  B12  +  E7  E9  +  E18  B11'                AK REQUIRED
```

AK REQ = (g = 00x.h = xx2) + (05x, 07x, 17x.h = 6,7) + (q = 07x.h = x1) + (15x.$\overline{\text{h BIT 2}}$)

= 002 INST + 056,057,076,077,176,177 INST + 071 INST + 150 - 153 INST

```
Boolean  C00 = M10  A1  +  M20  A2  +  M30  A3  +  B2     AK BUSY
```

AK BUSY = (A1 BUSY . k = x1) + (A2 BUSY . k = x2) + (A3 BUSY . k = x3) = (k BIT 2)

```
Boolean  C04 = M40  A0  +  M50  A1  +  M60  A2  +  M70  A3  +  B2'  AK BUSY
```

AK BUSY = (A4 BUSY . k = x0) + (A5 BUSY . k = x1) + (A6 BUSY . k = x2) + (A7 BUSY . k = x3) + $\overline{\text{(k BIT 2)}}$

```
Boolean  C10 = M11  A1  +  M21  A2  +  M31  A3  +  B2        AK ENTER
```

AK ENTER = (A1 ENTER . k = x1) + (A2 ENTER . k = x2) + (A3 ENTER . k = x3) + (k BIT 2)

```
Boolean  C14 = M41  A0  +  M51  A1  +  M61  A1  +  M71  A3  +  B2'          AK BUSY
```

AK ENTER = (A4 ENTER . k = x0) + (A5 ENTER . k = x1) + (A6 ENTER . k = x2) + (A7 ENTER . k = x3) + $\overline{\text{(k BIT 2)}}$

3JF MODULE - BOOLEAN EXAMPLE #2

REGISTER RESERVATION EXAMPLE

```
Boolean  P10'  =  S1'  Z0'  +  M11  +  M10' ; T2
         P10   =  (S1 + Z0)  .  M11'  .  M10 ; T2
```

A1 BUSY = (A1 BUSY + ISSUE) . $\overline{\text{A1 ENTER}}$. A1 BUSY : LATCHED

```
Boolean  S01  =  #  P10
```

A1 BUSY = A1 BUSY

```
Boolean  M10  =  K1  +  S1  L6'
```

A1 BUSY = SET A1 BUSY + (A1 BUSY . $\overline{\text{MASTER CLEAR}}$)

ENTER "A" EXAMPLE

```
Boolean  M11  =  K1  S1'  K12  +  Q10  S1
```

ENTER A1 = (ENTER A1 SOURCE . $\overline{\text{A1 BUSY}}$. A1 IS SOURCE CP1) + (A1 IS SOURCE CP2 . A1 BUSY)

X2005S0204

3JF MODULE - BOOLEAN EXAMPLE #3

ISSUE EXAMPLE

Boolean R00' = H0 + H1 + H2 + H3

R00 = H0' . H1' . H2' . H3'

Issue = $\overline{\text{AK CONFLICT}}$. $\overline{\text{AJ CONFLICT}}$. $\overline{\text{AI CONFLICT}}$. $\overline{\text{AH CONFLICT}}$

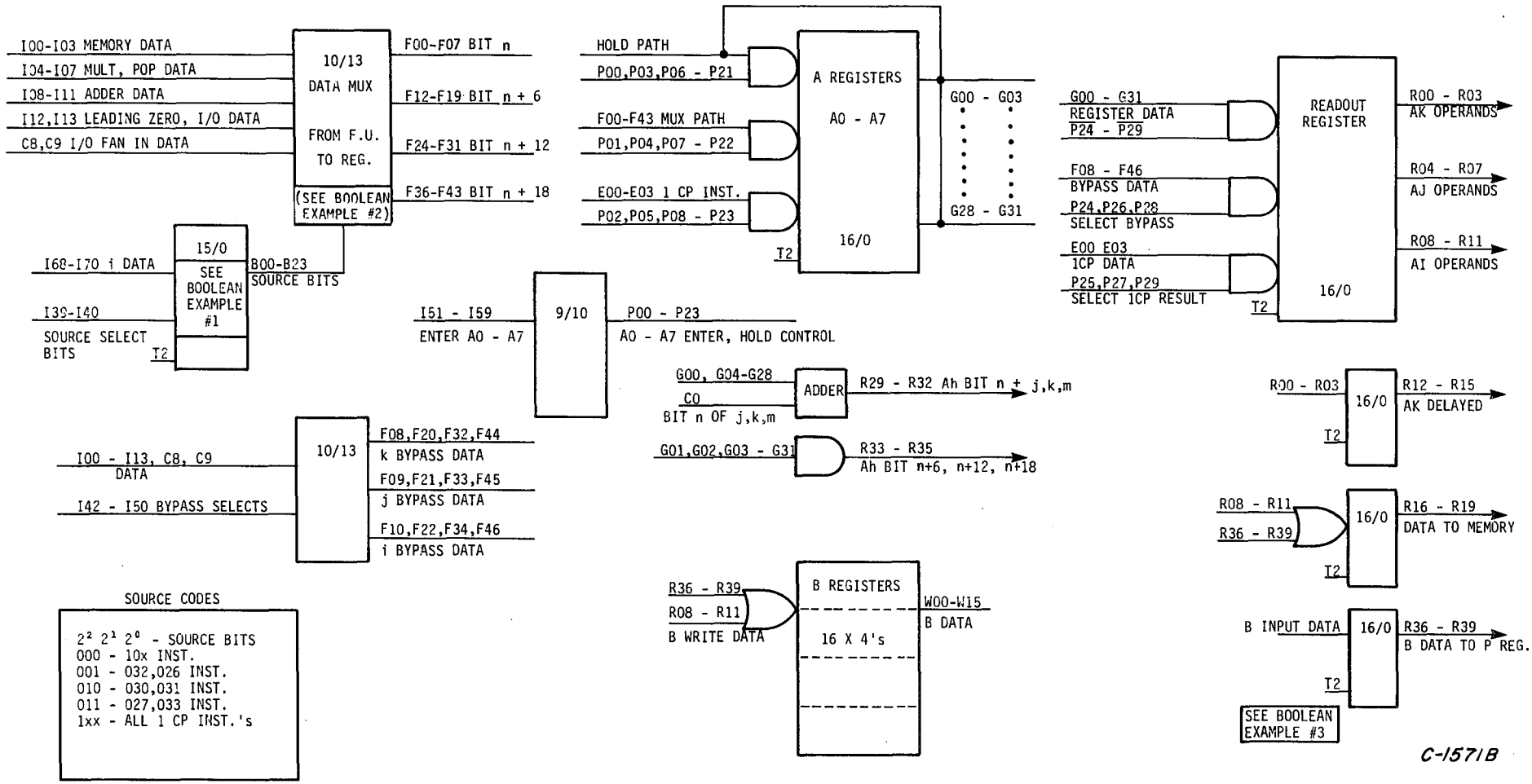
Boolean R01' = H4 + H5 H6 + H9 + H10 H14

R01 = H4' . (H5' + H6') . H9' . (H10' + H14')

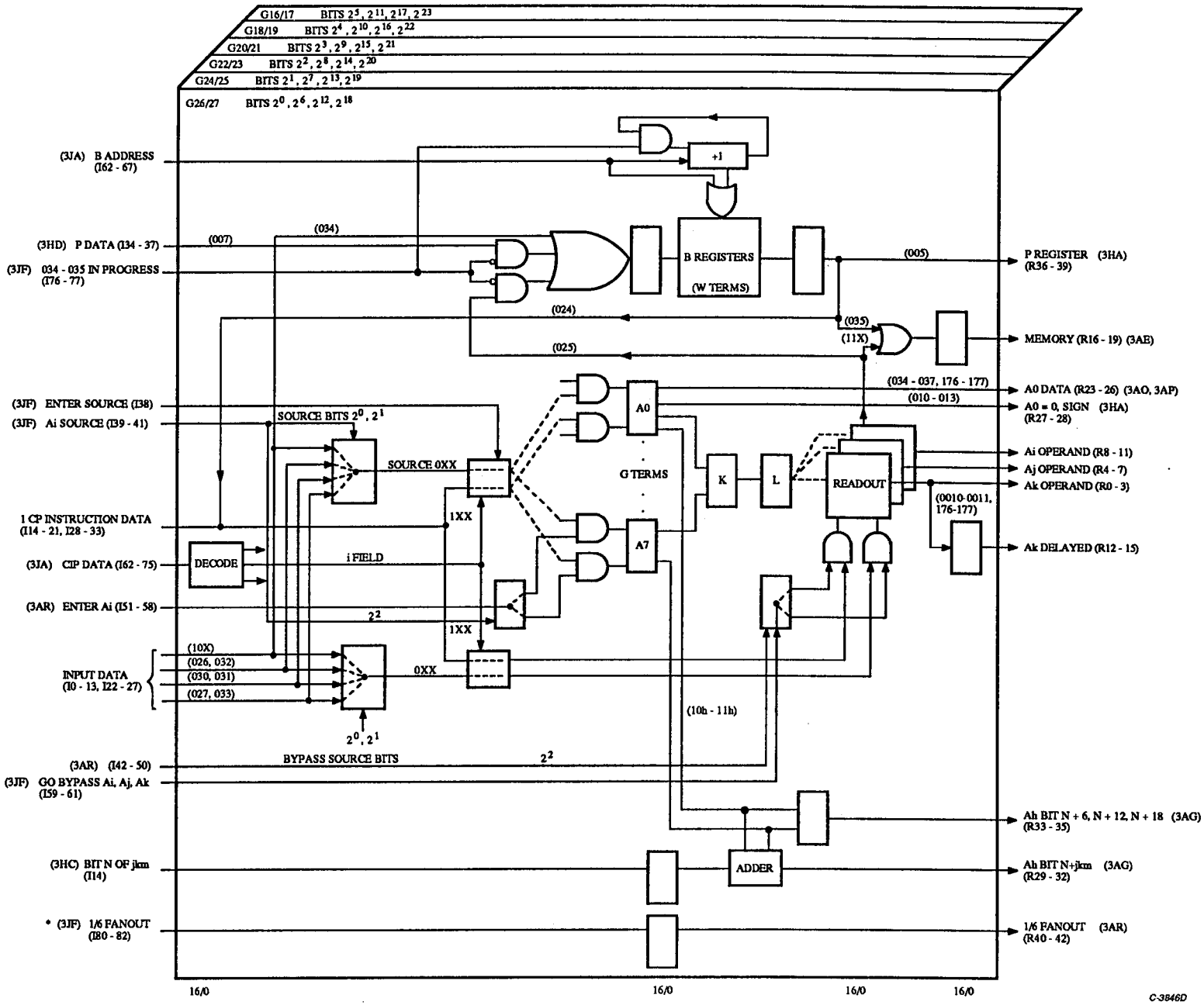
ISSUE = $\overline{\text{HOLD ISSUE}}$. ($\overline{\text{TPSF + HRDT}}$) . ($\overline{\text{0027xx IN CIP}}$) .

($\overline{\text{100-137 IN CIP}}$ + $\overline{\text{100-137 IN CP1 ; CP2 AND CONFLICT}}$)

X2005S0202



CRAY X-MP 3AR MODULE



* NOTE: I80 - I82 are used by the Enter Ai signal and the Go Bypass Source bits.

CRAY X-MP 3AR MODULE BLOCK DIAGRAM

3AR MODULE BOOLEAN EXAMPLE #1

```
Boolean     B00 = MUX (A10  B00) : DCD (A0) ; T2**A0 SOURCE 2**0
            B00 = A10 • A0 + B00 Ā0 ; T2
```

A0 SOURCE BIT 0 = (SOURCE SELECT BIT 0 • ENTER SOURCE A0) + (A0 SOURCE BIT 0 • $\overline{\text{ENTER SOURCE A0}}$)

3AR BOOLEAN EXAMPLE #2

```
Boolean     F00 = MUX (I0  I4  I8  I12) : DCD (B01  B00)/#     **A0 RESULT
            F00 = (I0 • B01 • B00) + (I4 • B01 • B00) + (I8 • B01 • B00) + (I12 • B01 • B00)
```

A0 RESULT = (MEMORY DATA • $\overline{\text{A0 SOURCE BIT 1}}$ • $\overline{\text{A0 SOURCE BIT 0}}$) + (MULT. DATA • $\overline{\text{A0 SOURCE BIT 1}}$ • A0 SOURCE BIT 0) + (ADDER DATA • A0 SOURCE BIT 1 • $\overline{\text{A0 SOURCE BIT 0}}$) + (LEADING ZERO DATA • A0 SOURCE BIT 1 • A0 SOURCE BIT 0)

C-1571

3AR BOOLEAN EXAMPLE #3

```
Boolean   R36  =  Y0  P34'  P35'  +  F11  P34  +  I34  P35  ;  T2  *** B DATA
```

B DATA = (DATA FROM 16X4's • $\overline{\text{GO WRITE B}}$ • $\overline{\text{007 INST}}$) + (MEMORY DATA • GO WRITE B) + (P REG DATA • 007 INST)

C-1571

MODULE LOCATION	BIT POSITION			
	n	n+6	n+12	n+18
G26/27	0	6	12	18
G24/25	1	7	13	19
G22/23	2	8	14	20
G20/21	3	9	15	21
G18/19	4	10	16	22
G16/17	5	11	17	23

A-REGISTER

	TEST POINT	BOOLEAN TERM	TEST POINT	BOOLEAN TERM	TEST POINT	BOOLEAN TERM	TEST POINT	BOOLEAN TERM
A0	$\overline{A21}$	$\overline{G00}$	$\overline{A20}$	$\overline{G01}$	$\overline{D72}$	$\overline{G02}$	$\overline{D71}$	$\overline{G03}$
A1	$\overline{A06}$	$\overline{G04}$	A27	$\overline{G05}$	D70	$\overline{G06}$	D68	$\overline{G07}$
A2	$\overline{A39}$	$\overline{G08}$	A33	$\overline{G09}$	D55	$\overline{G10}$	D57	$\overline{G11}$
A3	$\overline{A38}$	$\overline{G12}$	A31	$\overline{G13}$	D48	$\overline{G14}$	D46	$\overline{G15}$
A4	$\overline{B38}$	G16	B49	G17	C21	G18	C30	G19
A5	$\overline{B51}$	G20	B68	G21	C17	G22	C19	G23
A6	$\overline{B52}$	G24	B39	G25	C08	G26	C16	G27
A7	$\overline{B50}$	G28	B69	G29	C06	G30	C05	G31

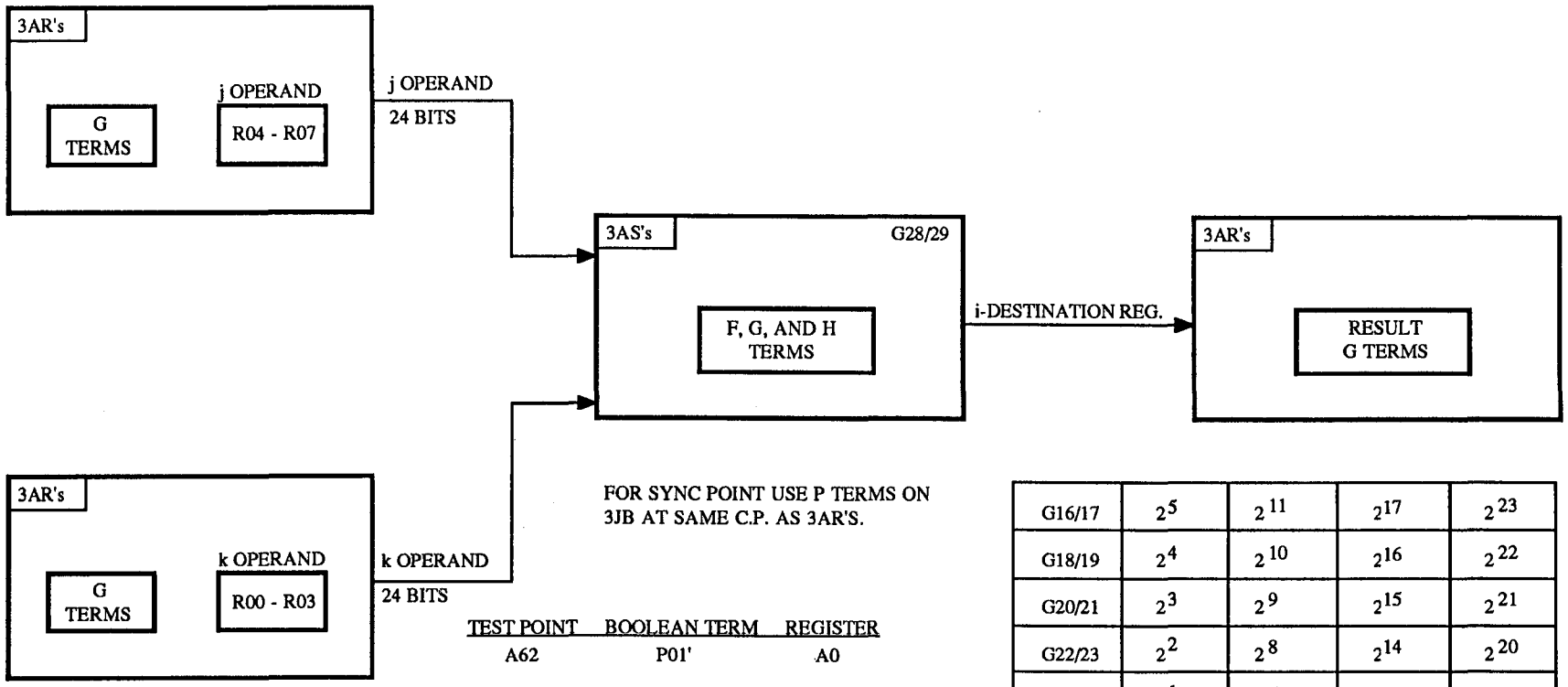
B-REGISTERS

BOOLEAN TERMS ON 16X4 REGISTER CHIPS (TYPE R)

00_8 - 17_8	W0	W1	W2	W3
20_8 - 37_8	W4	W5	W6	W7
40_8 - 57_8	W8	W9	W10	W11
60_8 - 77_8	W12	W13	W14	W15

A-5408

CRAY X-MP A/B REGISTER CONFIGURATION AND TEST POINTS
CPU - 0 (3AR)



FOR SYNC POINT USE P TERMS ON 3JB AT SAME C.P. AS 3AR'S.

TEST POINT	BOOLEAN TERM	REGISTER
A62	P01'	A0
A63	P11'	A1
A48	P21'	A2
A52	P31'	A3
A35	P41'	A4
A38	P51'	A5
A18	P61'	A6
A40	P71'	A7

G16/17	2^5	2^{11}	2^{17}	2^{23}
G18/19	2^4	2^{10}	2^{16}	2^{22}
G20/21	2^3	2^9	2^{15}	2^{21}
G22/23	2^2	2^8	2^{14}	2^{20}
G24/25	2^1	2^7	2^{13}	2^{19}
G26/27	2^0	2^6	2^{12}	2^{18}

3AR MODULE AND BIT LOCATION. THIS IS FOR CPU 0. FOR CPU 1 USE S LOCATIONS.

A-2265

CRAY X-MP ADDRESS REGISTERS

SCALAR REGISTER AND CONTROL

The S register control block diagram includes the 3JA, 3JC, 3VB, and the 3GR modules. Below is a brief description of the modules involved in the block diagram.

3JA MODULE

The 3JA module is responsible for CIP fanout. It holds a copy of the current instruction to be issued. It then generates several copies to the 3AR and the 3GR modules.

3JC MODULE

The 3JC module is the CIP and issue control module for S register type instructions. The 3JC is responsible for S register reservation control, conflict checking and S register issue control. The 3JC module decodes the instruction and determines if the instruction can issue. It does so by comparing what registers the instruction wants to use against what register may already be reserved. The 3JC also checks for entering data conflicts and holds issue if any is detected.

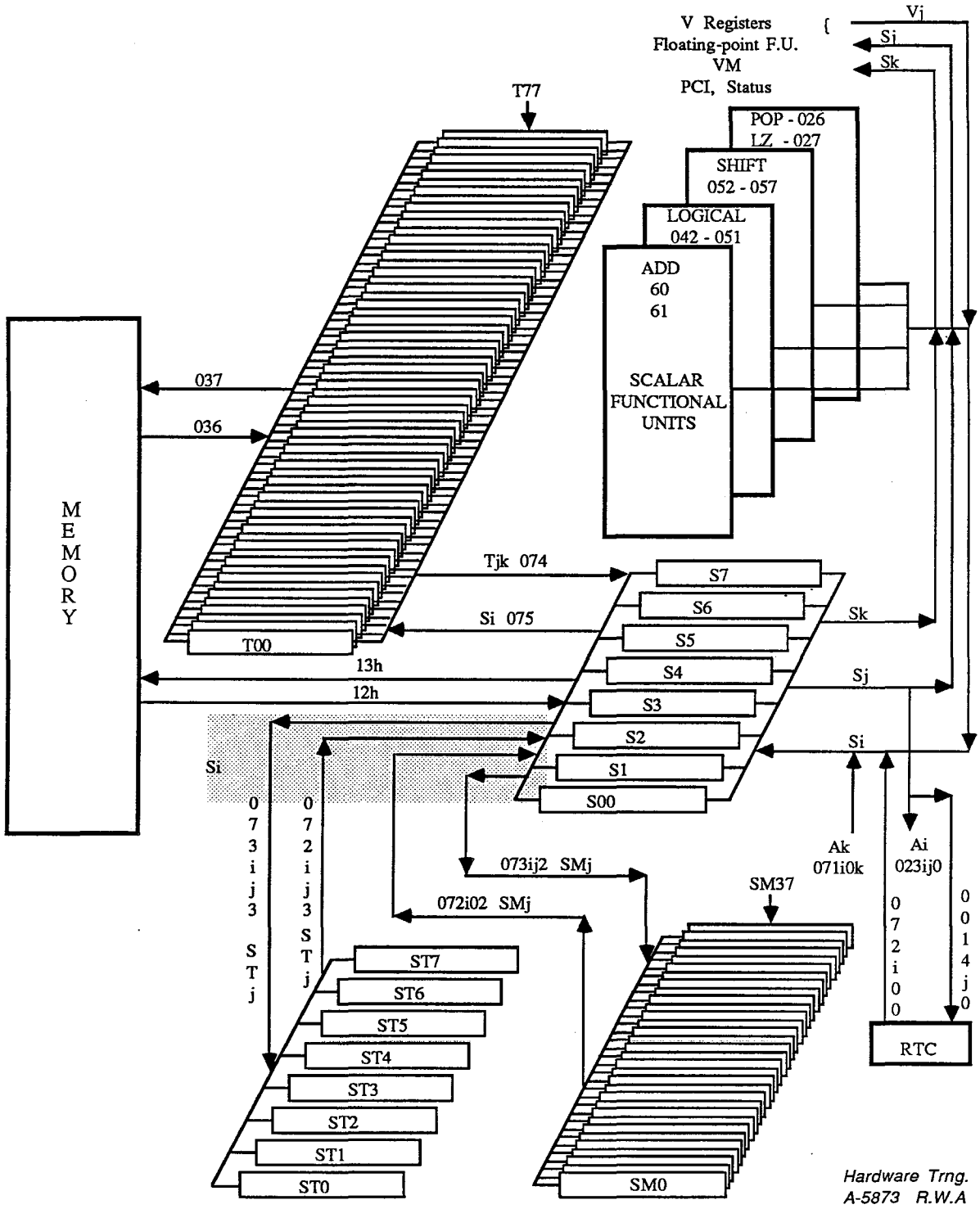
3VA MODULE

The 3VA modules fan out the Enter S0-S7 signals from the 3JC module to all the 3GR modules.

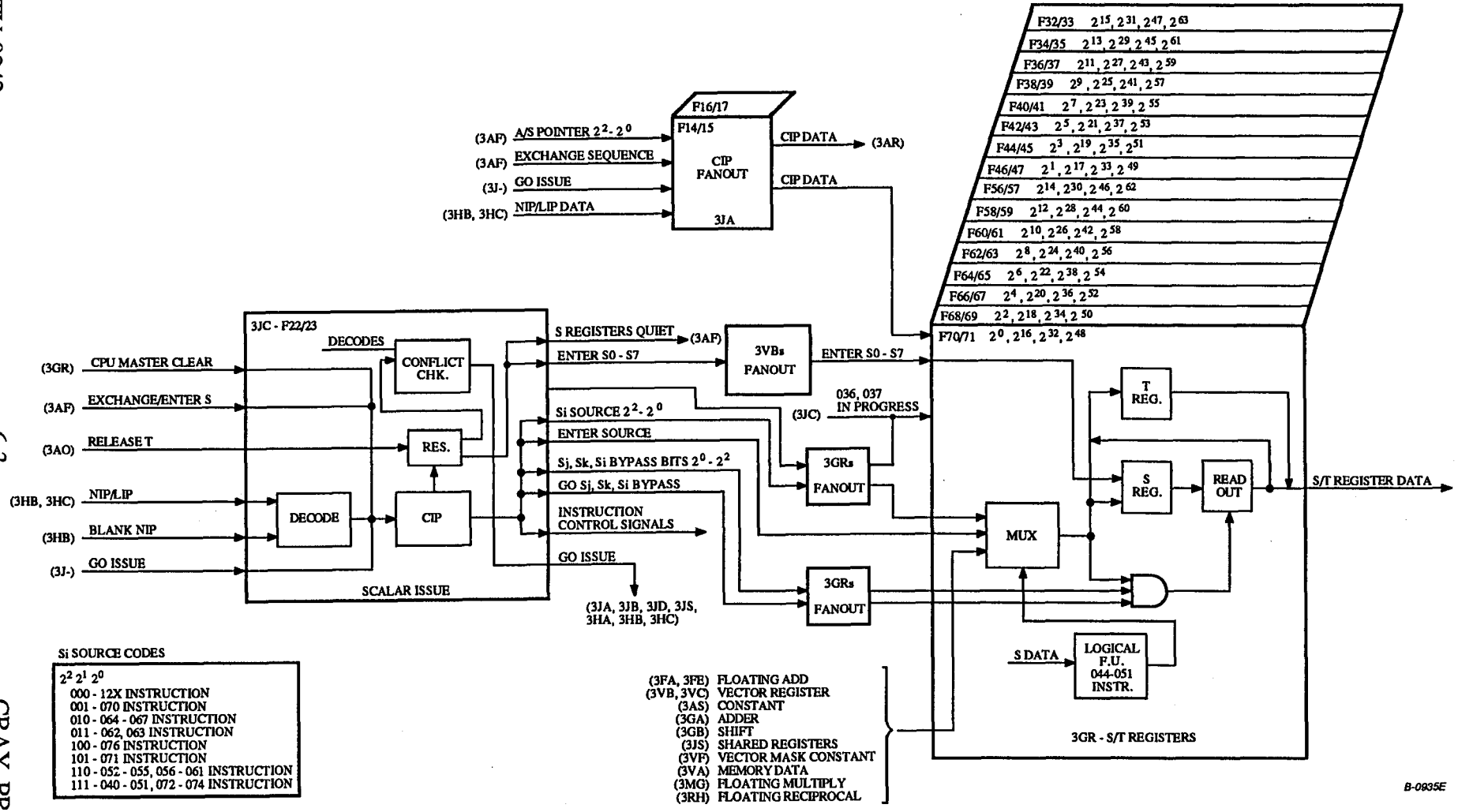
3GR MODULE

The 3GR modules are the S registers. Each 3GR module holds bits n, n+16, n+23, and n+48 for all eight S registers and all 64 T registers. The 3GR's have a multiplexed input path and three readout registers; one each for Si, Sj, and Sk.

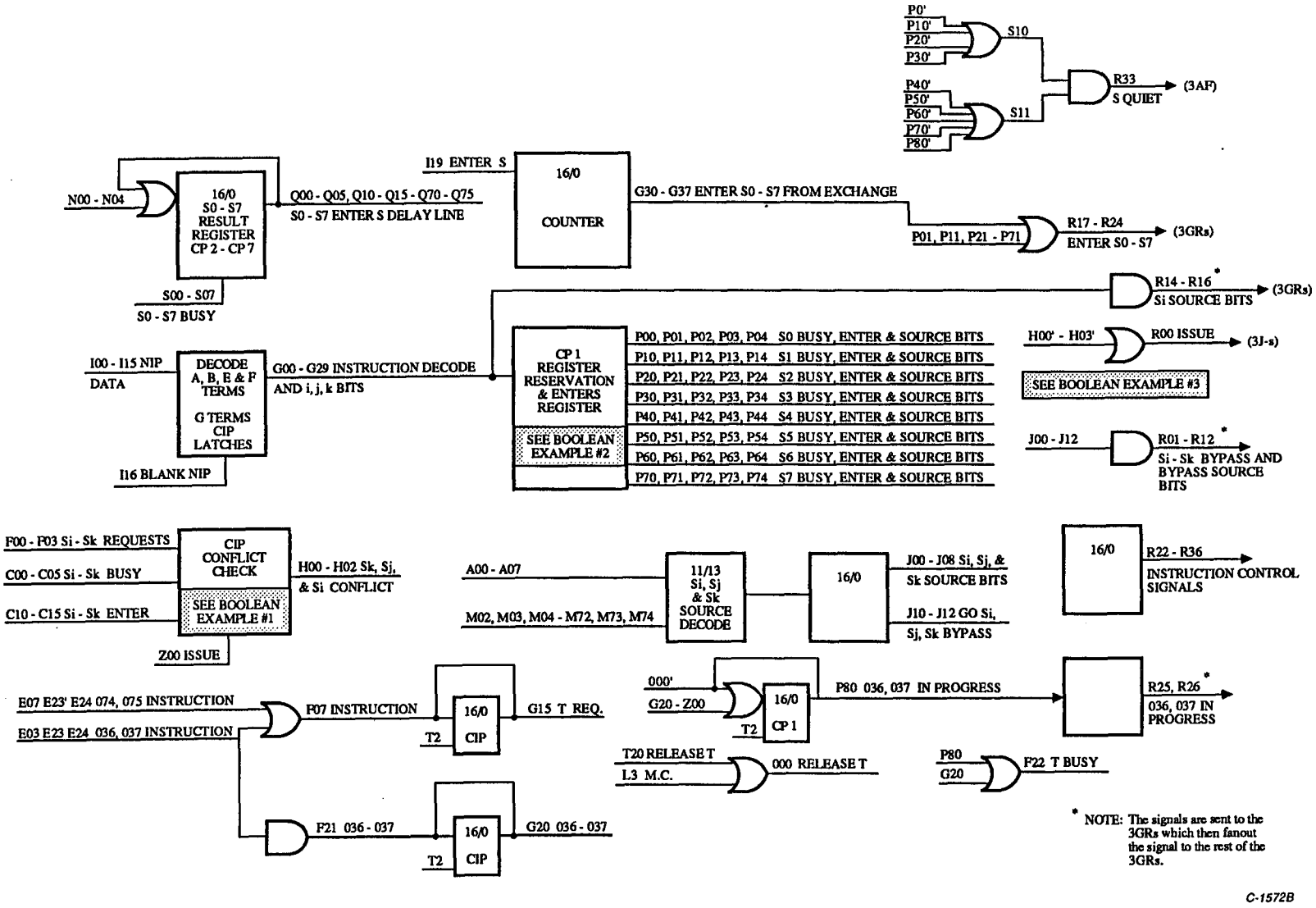
The 3GR's also contain the S register logical functional unit.



CRAY X-MP S REGISTERS



CRAY X-MP SCALAR REGISTERS AND CONTROL BLOCK DIAGRAM (CPU 0)



CRAY X-MP 3JC MODULE S REGISTER ISSUE CONTROL

EXAMPLE OF CONFLICT CHECKS

```
Boolean   H00' = C10 C0' C3' Z0' F0' # # H0' # # C13 ; T2 SK CONFLICT
          H00  = (F0 + H0) • (Z0 + H0) • (C3 • C0) • (C10' + C13') ; T2
```

SK CONFLICT = (SK REQ + SK CONFLICT) • (ISSUE + SK CONFLICT) • (SK BUSY) • $\overline{\text{(SK ENTER)}}$; T2

```
Boolean   F00 = E4 E24 + E5 E24' E23' + E6     SK REQUIRED
```

SK REQUIRED = (g = 04x • h BIT 2) + (g = 05x • $\overline{\text{h BIT 2}}$ • $\overline{\text{h BIT 1}}$) + (g = 06x)

SK REQUIRED = 044 - 047 INST + 050 - 051 INST + 060 - 067 INST

```
Boolean   C00 = M10 A1 + M20 A2 + M30 A3 + B2     SK BUSY
```

SK BUSY = (S1 BUSY • k = x1) + (S2 BUSY • k = x2) = (S3 BUSY • k = x3) + (k BIT 2)

```
C03 = M40 A0 + M50 A1 + M60 A2 + M70 A3 + B2'     SK BUSY
```

SK BUSY = (S4 BUSY • k = x0) + (S5 BUSY • k = x1) + (S6 BUSY • k = x2) + (S7 BUSY • k = x3) + $\overline{\text{(k BIT 2)}}$

```
C10 = M11 A1 + M21 A2 + M31 A3 + B2     SK ENTER
```

SK ENTER = (S1 ENTER • k = x1) + (S2 ENTER • k = x2) + (S3 ENTER • k = x3) + (k BIT 2)

```
C13 = M41 A0 + M51 A1 + M61 A2 + M71 A3 + B2'     SK ENTER
```

SK ENTER = (S4 ENTER • k =x0) + (S5 ENTER • k = x1) + (S6 ENTER • k = x2) + S7 ENTER • k = x3) + $\overline{\text{(k BIT 2)}}$

3JC MODULE EXAMPLE 2

REGISTER RESERVATION EXAMPLE

```
Boolean   P10' = S1' Z0' + M11 + M10' ; T2     S1 BUSY
          P10  = (S1 + Z0) • M11' • M10 : T2
```

S1 BUSY = (S1 BUSY + ISSUE) • $\overline{\text{(S1 ENTER)}}$ (S1 BUSY)

```
Boolean   S01 = # P10                          S1 BUSY
```

S1 BUSY = S1 BUSY

```
Boolean   M10 = K1 + S1 L3'                    S1 BUSY
```

S1 BUSY = SET S1 BUSY + (S1 BUSY • $\overline{\text{MASTER CLEAR}}$)

ENTER "A" EXAMPLE

```
Boolean   M11 = K13 K1 S1' + # Q10 S1          S1 ENTER
```

S1 ENTER = (S1 IS SOURCE CP1 • ENTER S1 SOURCE • $\overline{\text{S1 BUSY}}$) + (S1 IS SOURCE CP2 • S1 BUSY)

C-1572

3JC MODULE EXAMPLE 3

ISSUE EXAMPLE

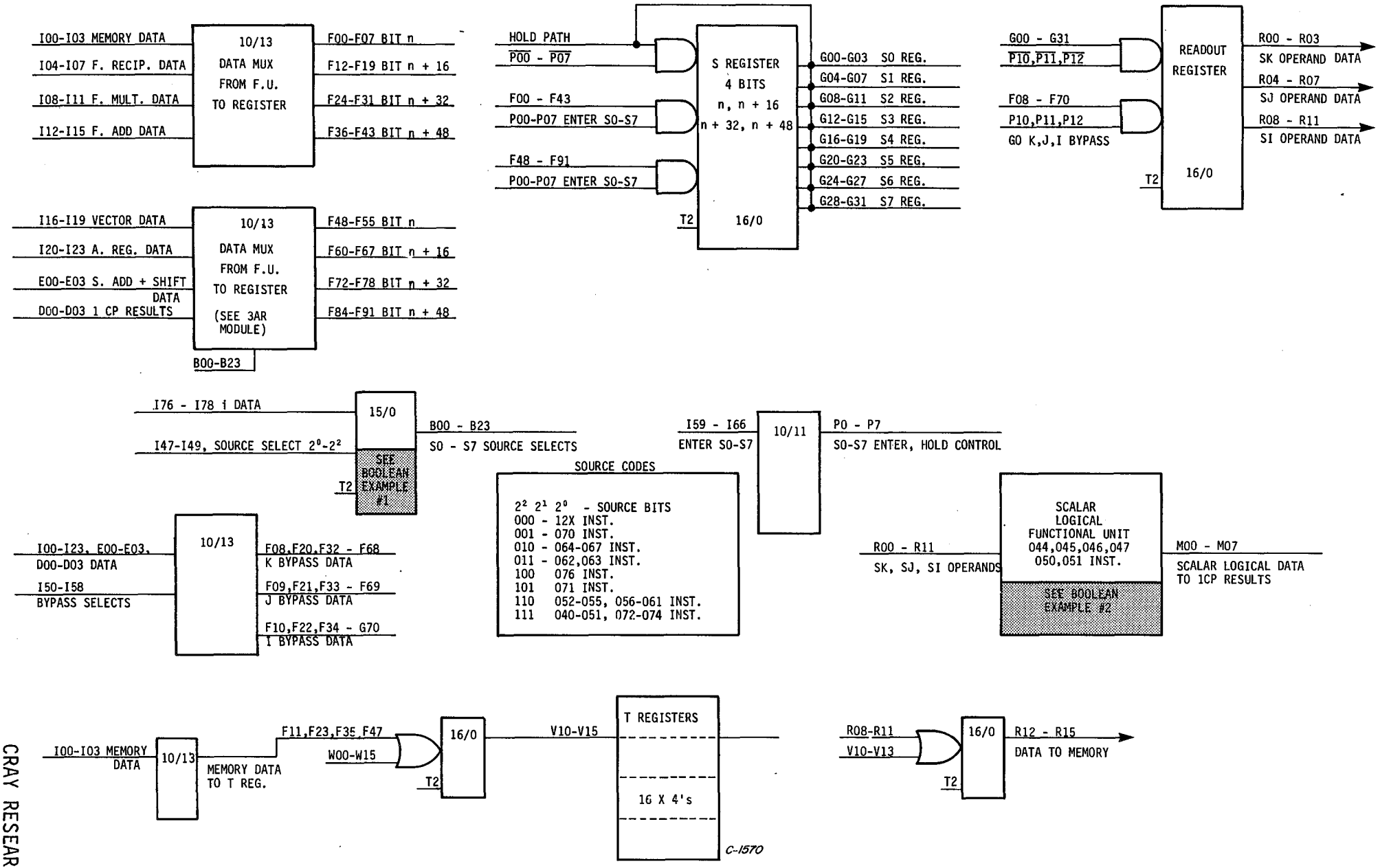
```
Boolean   R00' = H0 + H1 + H2 + H3        ISSUE
          R00  = H0'  H1'  H2'  H3'
```

ISSUE = $\overline{\text{SK CONFLICT}} \cdot \overline{\text{SJ CONFLICT}} \cdot \overline{\text{SI CONFLICT}} \cdot \overline{\text{HOLD ISSUE}}$

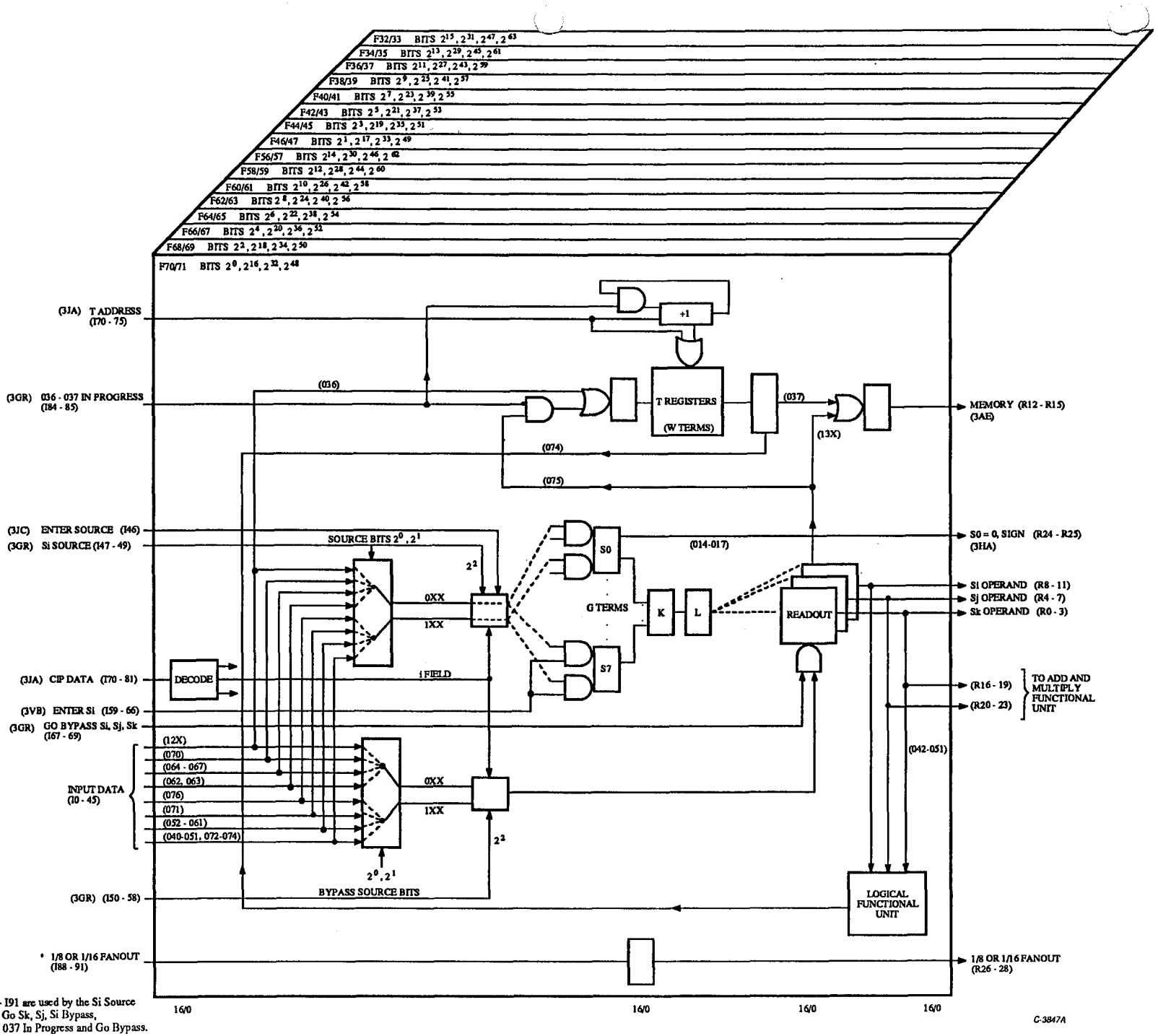
```
Boolean   H03  = F9  0  F8  F4  F7  Z0  #  #  F22  #  Z0'  ;  T2     HOLD ISSUE
          H03  = F22  F7  Z0  +  F4  Z0  +  F8  Z0  +  F9  Z0'  ;  T2
```

HOLD ISSUE = (T BUSY • T REQ • ISSUE) + (S0 HOLD • ISSUE) + (ACCESS CONFLICT • ISSUE) + (CONTINUE HOLD • $\overline{\text{ISSUE}}$)

C-1572



3GR MODULE



* NOTE: I88 - I91 are used by the Si Source bits, Go Sk, Sj, Si Bypass, 036, 037 In Progress and Go Bypass.

CRAY X-MP 3GR MODULE BLOCK DIAGRAM

3GR MODULE BOOLEAN EXAMPLE #2

Boolean M00 = R4' R0' H15 + R4 R0' H17 + R4' R0 H18 SCALAR LOGICAL

LOGICAL RESULT BIT M = $\overline{\text{SJ BIT n}}$ • $\overline{\text{SK BIT n}}$ • 047 INST + SJ BIT n • $\overline{\text{SK BIT n}}$ • 045,046,051 INST + $\overline{\text{SJ BIT n}}$ •
SK BIT n • 046,051 INST

Boolean M04 = R4 R0 H16 + R8 R0' H19

LOGICAL RESULT BIT M = SJ BIT n • SK BIT n • 044,047,050,051 INST + SI BIT n $\overline{\text{SK BIT n}}$ 050 INST.

FOR EXAMPLE 047 INST.

047 RESULT BIT M = $\overline{\text{SJ BIT n}}$ • $\overline{\text{SK BIT n}}$ • 047 INST. + SJ BIT n • SK BIT n • 047 INST.

047 RESULT BIT = (M00 = R4' R0' H15) + (M04 = R4 R0 R16)

047 INST. = LOGICAL EQUIVALENCE

C-1570

3GR MODULE BOOLEAN EXAMPLE #1

```
Boolean   B00   =  MUX (A10  B00) : DCD (A0) ; T2   SO SOURCE 2**0
```

SO SOURCE BIT 0 = SOURCE SELECT BIT 0 • ENTER SO SOURCE + SO SOURCE BIT 0 • $\overline{\text{ENTER SO SOURCE}}$

C-1570

MODULE LOCATION	BIT POSITION			
	n	n+16	n+32	n+48
F70/71	0	16	32	48
F68/69	2	18	34	50
F66/67	4	20	36	52
F64/65	6	22	38	54
F62/63	8	24	40	56
F60/61	10	26	42	58
F58/59	12	28	44	60
F56/57	14	30	46	62
F46/47	1	17	33	49
F44/45	3	19	35	51
F42/43	5	21	37	53
F40/41	7	23	39	55
F38/39	9	25	41	57
F36/37	11	27	43	59
F34/35	13	29	45	61
F32/33	15	31	47	63

S-REGISTER

	TEST POINT	BOOLEAN TERM	TEST POINT	BOOLEAN TERM	TEST POINT	BOOLEAN TERM	TEST POINT	BOOLEAN TERM
S0	$\overline{B30}$	G00	$\overline{B34}$	G01	$\overline{D29}$	G02	$\overline{D27}$	G03
S1	$\overline{A55}$	G04	$\overline{A42}$	G05	$\overline{C38}$	G06	$\overline{C36}$	G07
S2	$\overline{B46}$	G08	$\overline{B42}$	G09	$\overline{D46}$	G10	$\overline{D42}$	G11
S3	$\overline{A38}$	G12	$\overline{A36}$	G13	$\overline{C31}$	G14	$\overline{C33}$	G15
S4	$\overline{B55}$	G16	$\overline{B48}$	G17	$\overline{D50}$	G18	$\overline{D48}$	G19
S5	$\overline{A29}$	G20	$\overline{A31}$	G21	$\overline{C25}$	G22	$\overline{C27}$	G23
S6	$\overline{B64}$	$\overline{G24}$	$\overline{B53}$	$\overline{G25}$	$\overline{D62}$	$\overline{G26}$	$\overline{D51}$	$\overline{G27}$
S7	$\overline{A12}$	$\overline{G28}$	$\overline{A35}$	G29	$\overline{C11}$	G30	C12	G31

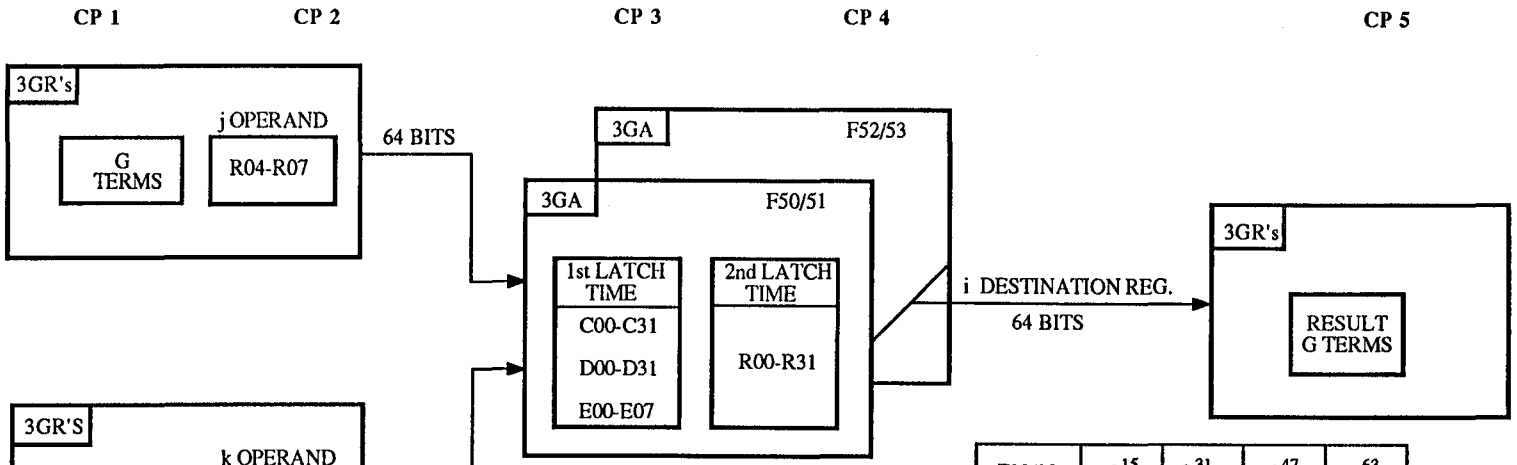
T-REGISTER

BOOLEAN TERMS ON 16X4 REGISTER CHIPS (TYPE R)

00_8 - 17_8	W0	W1	W2	W3
20_8 - 37_8	W4	W5	W6	W7
40_8 - 57_8	W8	W9	W10	W11
60_8 - 77_8	W12	W13	W14	W15

A-5409

CRAY X-MP S/T REGISTER CONFIGURATION AND TEST POINTS
CPU - 0 (3GR)



For sync point use P terms on 3JC at same clock period as first latch time on 3GA.

TEST POINT	BOOLEAN TERM	REGISTER
A02'	P01'	S0
A10	P11'	S1
A08	P21'	S2
A18	P31'	S3
A44	P41'	S4
A72	P51'	S5
A68	P61'	S6
A59	P71'	S7

060ijk INSTRUCTION

061ijk INSTRUCTION

F32/33	2^{15}	2^{31}	2^{47}	2^{63}
F34/35	2^{13}	2^{29}	2^{45}	2^{61}
F36/37	2^{11}	2^{27}	2^{43}	2^{59}
F38/39	2^{9}	2^{25}	2^{41}	2^{57}
F40/41	2^{7}	2^{23}	2^{39}	2^{55}
F42/43	2^{5}	2^{21}	2^{37}	2^{53}
F44/45	2^{3}	2^{19}	2^{35}	2^{51}
F46/47	2^{1}	2^{17}	2^{33}	2^{49}
F56/57	2^{14}	2^{30}	2^{46}	2^{62}
F58/59	2^{12}	2^{28}	2^{44}	2^{60}
F60/61	2^{10}	2^{26}	2^{42}	2^{58}
F62/63	2^{8}	2^{24}	2^{40}	2^{56}
F64/65	2^{6}	2^{22}	2^{38}	2^{54}
F66/67	2^{4}	2^{20}	2^{36}	2^{52}
F68/69	2^{2}	2^{18}	2^{34}	2^{50}
F70/71	2^{0}	2^{16}	2^{32}	2^{48}

3GR Module and bit locations. This is for CPU 0. For CPU 1 use R locations.

A-2266

EXCHANGE OPERATION OVERVIEW

A. Method of transferring control from one program to another program

B. Conditions that cause an Exchange to take place

1. Deadstart Sequence via the MCU

2. Interrupt flags

3. Program exit
   a. 000 Instruction - Error Exit
   b. 004 Instruction - Normal Exit

C. Conditions that would cause an Exchange to be delayed

1. NIP Register contains a valid instruction, but not a 0034jk instruction that has the semaphore specified by the jk field equal to a one (hold issue condition)

2. S, V OR A Register busy

3. Memory not quiet (any section busy)

D. Exchange Timing

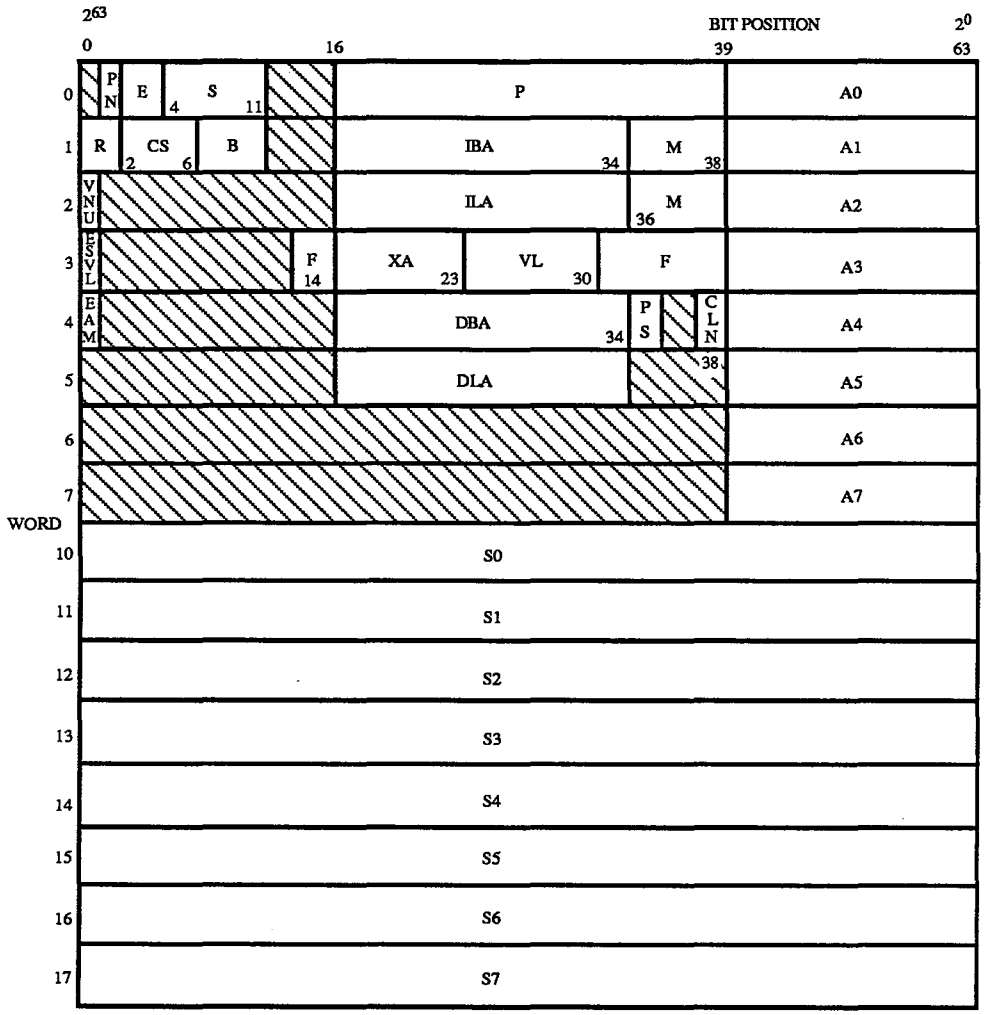
1. For 32 Bank, ECL Memory: 40 CP's
2. For 16 Bank, ECL Memory: 42 CP's
   NOTE: 21 CP's for the Exchange
   + 3 CP's to avoid conflicts
   +16 CP's for fetch on a 32 Bank or 18 CP'S for 16 Bank.
   40 CP's total for a 32 Bank or 42 CP'S for 16 bank machine.
3. For 32 Bank, MOS Memory: 51 CP's
4. For 16 Bank, MOS Memory: 57 CP's

NOTE: After every Exchange, a Fetch operation will take place.

MOS MEMORY					ECL MEMORY				
	READ		WRITE			READ		WRITE	
CP'S	BANK			BANK	CP'S	BANK			BANK
1	0	A0			1	0	A0		
2	1	A1			2	1	A1		
3	2	A2			3	2	A2		
4	3	A3			4	3	A3		
5	4	A4			5	4	A4		
6	5	A5			6	5	A5	A0	0
7	6	A6			7	6	A6	A1	1
8	7	A7			8	7	A7	A2	2
9	10	S0			9	10	S0	A3	3
10	11	S1	A0	0	10	11	S1	A4	4
11	12	S2	A1	1	11	12	S2	A5	5
12	13	S3	A2	2	12	13	S3	A6	6
13	14	S4	A3	3	13	14	S4	A7	7
14	15	S5	A4	4	14	15	S5	S0	10
15	16	S6	A5	5	15	16	S6	S1	11
16	17	S7	A6	6	16	17	S7	S2	12
17			A7	7	17			S3	13
18			S0	10	18			S4	14
19			S1	11	19			S5	15
20			S2	12	20			S6	16
21			S3	13	21			S7	17
22			S4	14					
23			S5	15					
24			S6	16					
25			S7	17					

MOS MEMORY VS. ECL MEMORY
EXCHANGE TIMING COMPARISON CHART

Hardware Trng.
XV203W14M R.W.A.



READ OUT AREA

WORD	BIT	FIELD
0	2^{62}	PN - Processor Number
0	2^{61} - 2^{60}	E - Error Type
0	2^{59} - 2^{52}	S - Syndrome Bits
1	2^{63} - 2^{62}	R - Read Mode
1	2^{61} - 2^{57}	CS - Chip Select Bits
1	2^{56} - 2^{52}	B - Bank Address

STATUS AREA

WORD	BIT	FIELD
0	2^{24} - 2^{47}	P - Program Address Register
1	2^{29} - 2^{47}	IBA - INSTRUCTION BASE ADDRESS
2	2^{29} - 2^{47}	ILA - INSTRUCTION LIMIT ADDRESS
3	2^{40} - 2^{47}	XA - EXCHANGE ADDRESS
3	2^{33} - 2^{39}	VL - Vector Length
4	2^{29} - 2^{47}	DBA - Data Base Address
5	2^{29} - 2^{47}	DLA - Data Limit Address

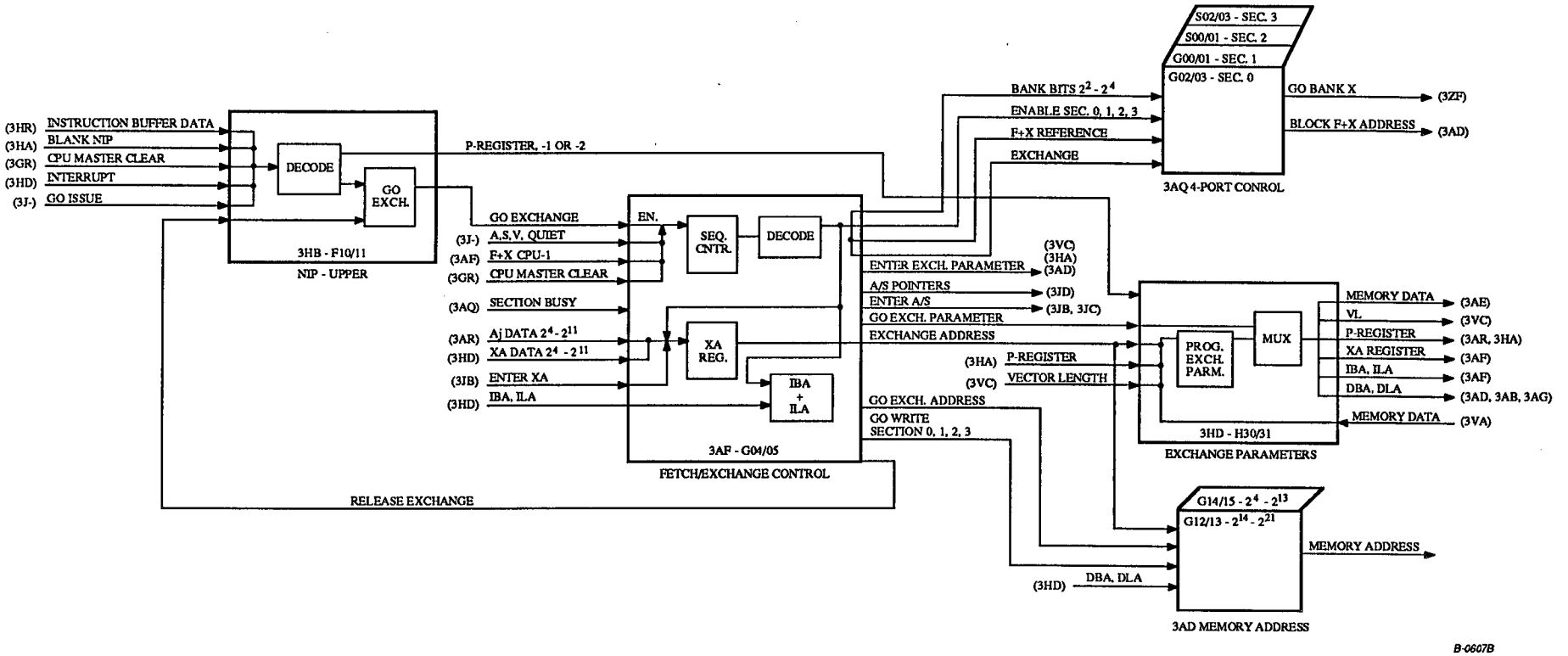
WORD	BIT	FIELD
1&2	2^{24} - 2^{28}	M - Mode Register
1	2^{24}	IMM - Interrupt Monitor Mode
1	2^{25}	SEI - Select for External Interrupt
1	2^{26}	BDM - Bidirectional Memory Access
1	2^{27}	FPS - Floating Point Error Status
1	2^{28}	WS - Waiting on Semaphore
2	2^{24}	MM - Monitor Mode
2	2^{25}	IUM - Interrupt on Uncorrectable Memory Error
2	2^{26}	IFP - Interrupt on Floating Point Error
2	2^{27}	ICM - Interrupt on Correctable Memory Error
2	2^{28}	IOR - Interrupt on Operand Range Error
2	2^{63}	VNU - Vector Not Used
3	2^{63}	ESVL - Enable Second Vector Logical
4	2^{63}	EAM - Enhance Addressing Mode

FLAGS

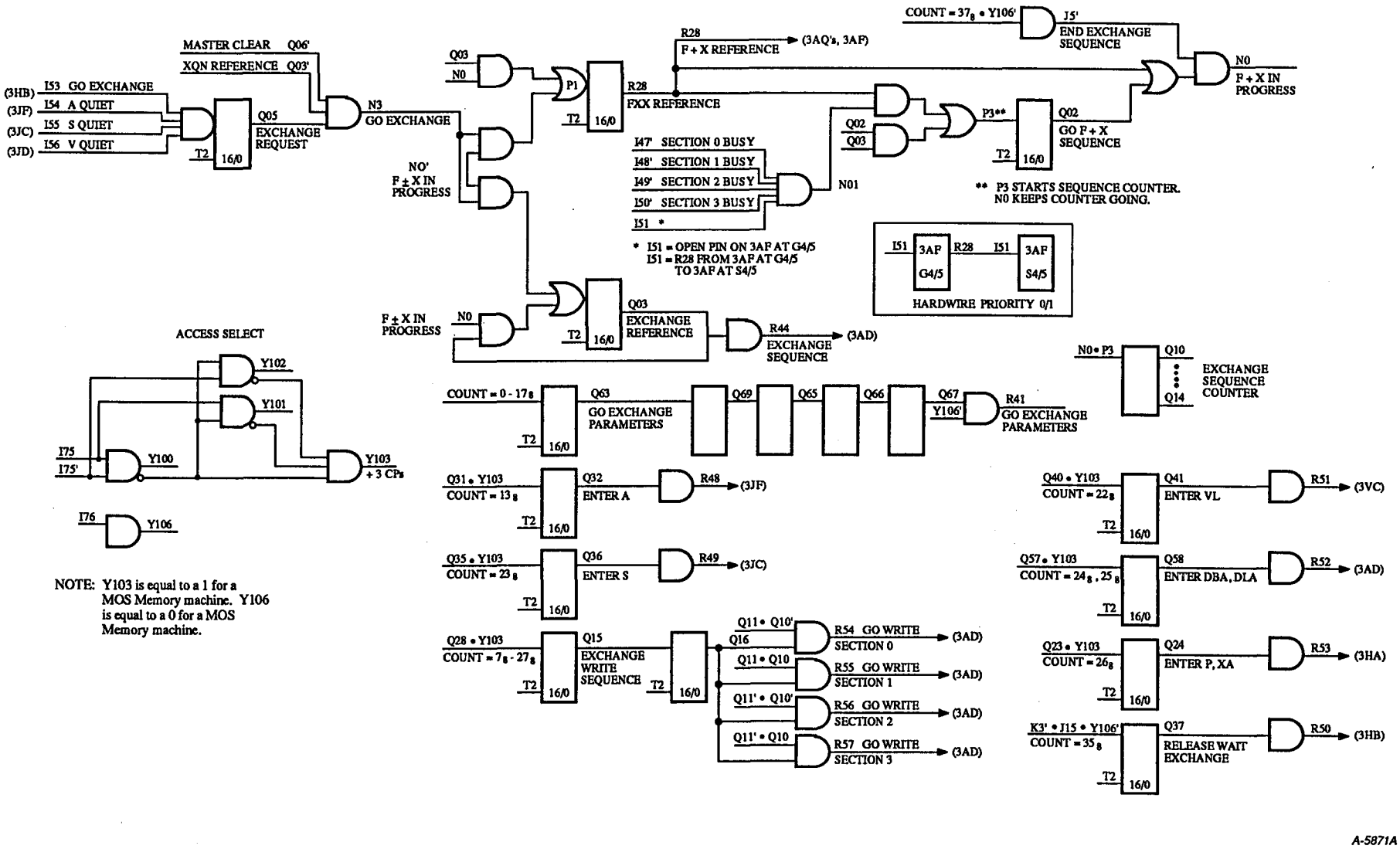
3	2^{49}	ICP - Interrupt from Internal CPU
3	2^{48}	DL - Deadlock
3	2^{32}	PCI - Program Clock
3	2^{31}	MCU - MCU Interrupt (MIOP)
3	2^{30}	FPE - Floating Point Error
3	2^{29}	ORE - Operand Range Error
3	2^{28}	PRE - Program Range Error
3	2^{27}	ME - Memory Error
3	2^{26}	IOI - I/O Interrupt
3	2^{25}	EEX - Error Exit (000)
3	2^{24}	NEX - Normal Exit (004)
4	2^{28}	PS - Program State
4	2^{24} - 2^{25}	CLN - Cluster Number Register

A-0929E

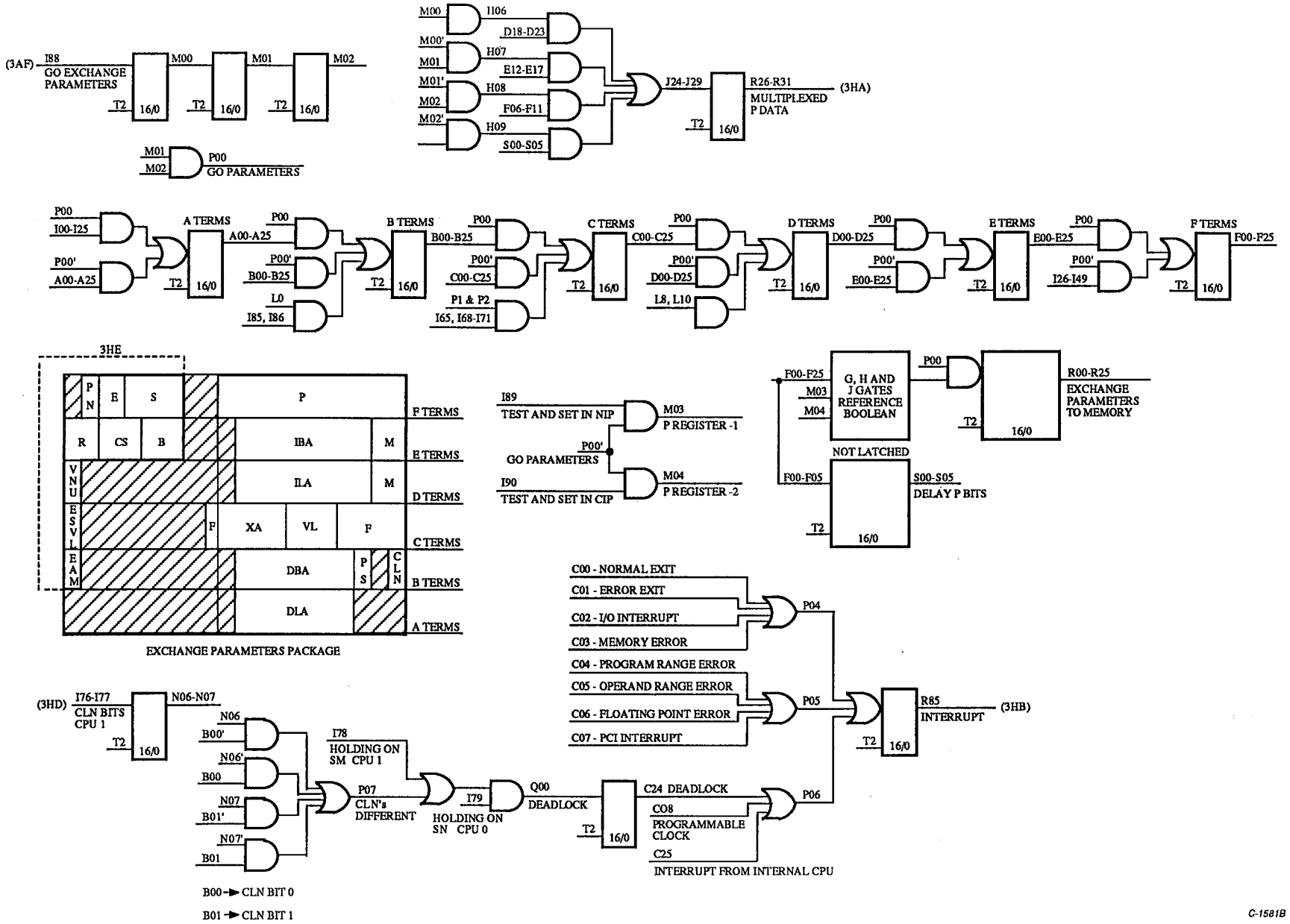
CRAY X-MP EXCHANGE PACKAGE



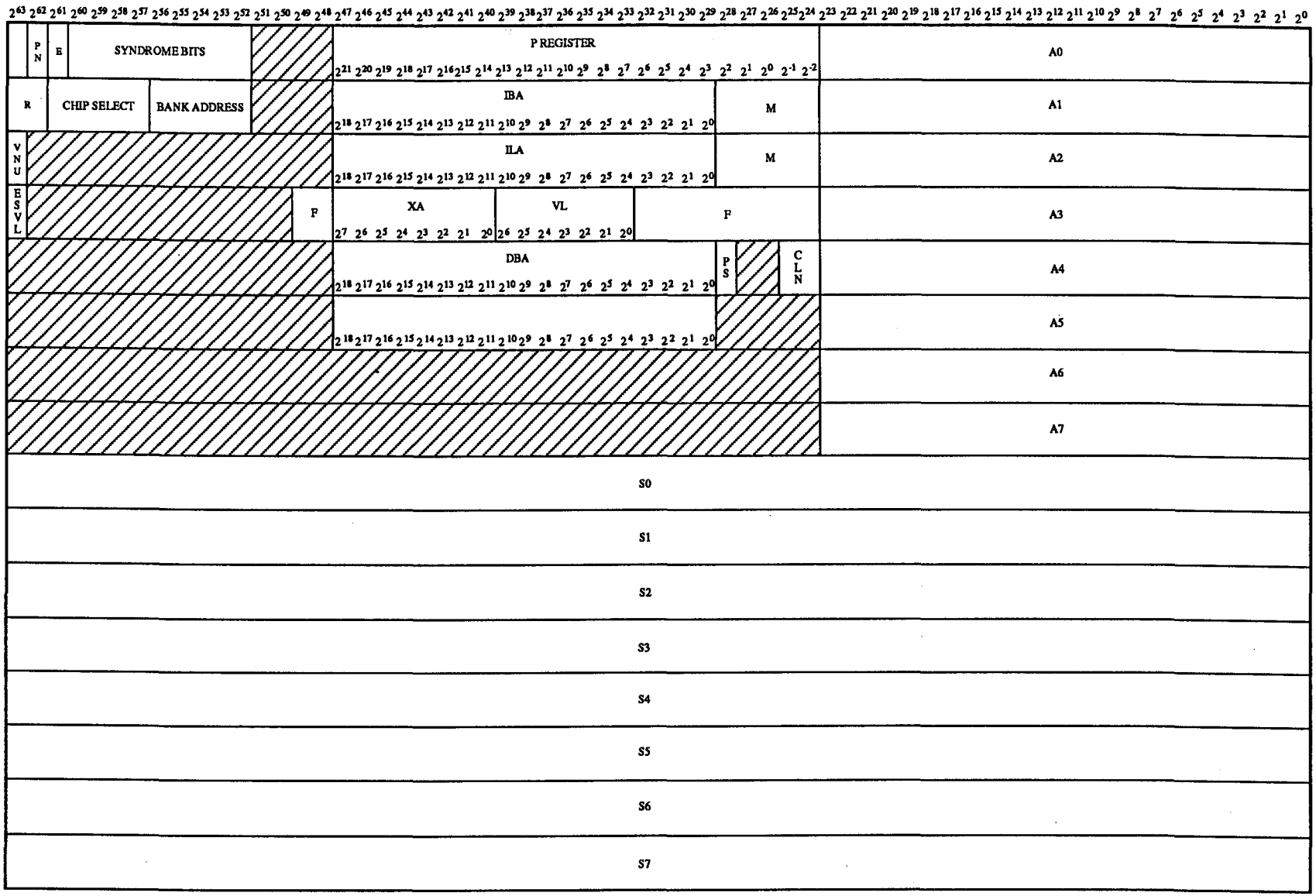
CRAY X-MP EXCHANGE SEQUENCE CONTROL (CPU 0)



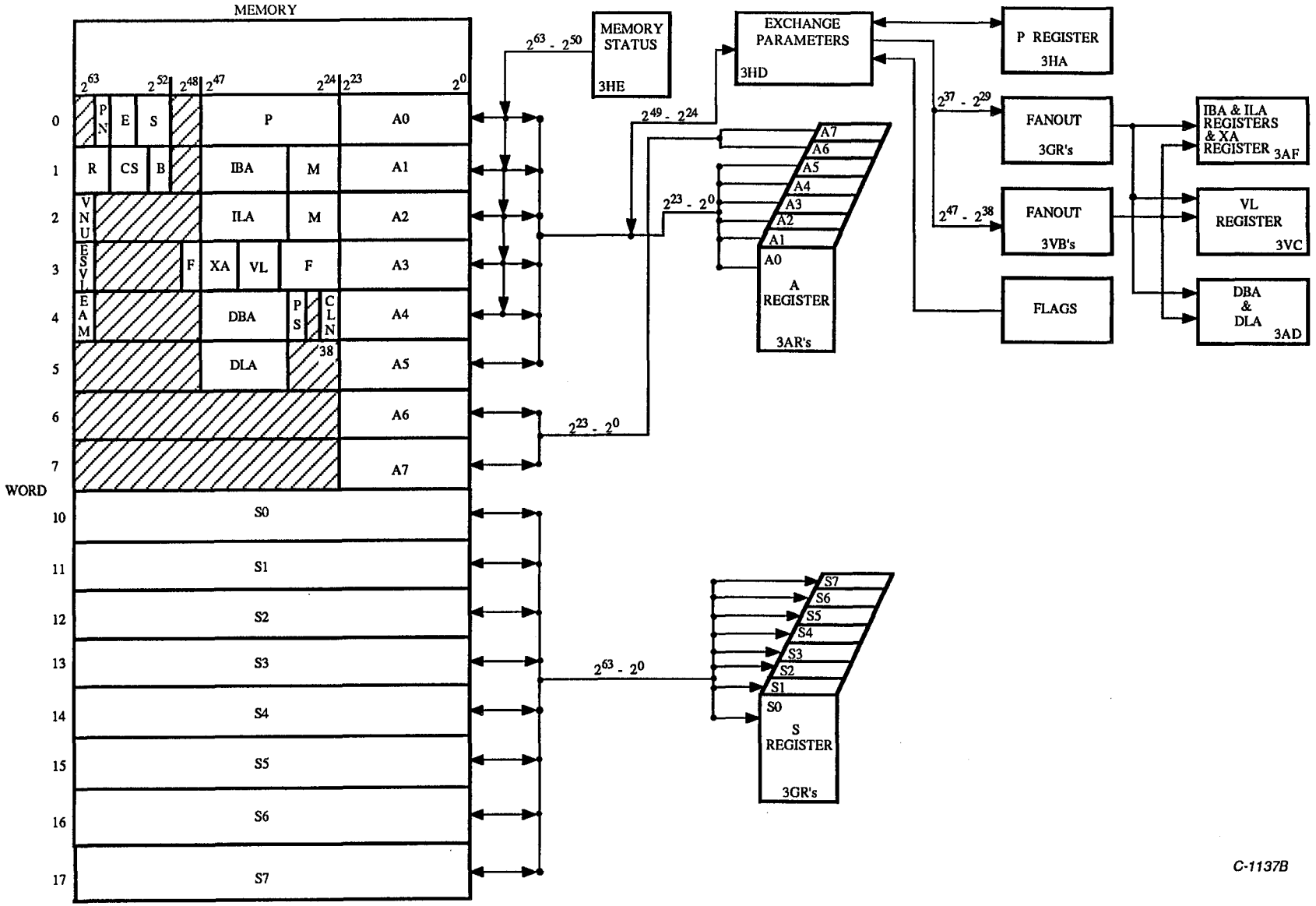
CRAY X-MP 3AF MODULE EXCHANGE CONTROL
(FOR CRAY X-MP/216 WITH MOS MEMORY)



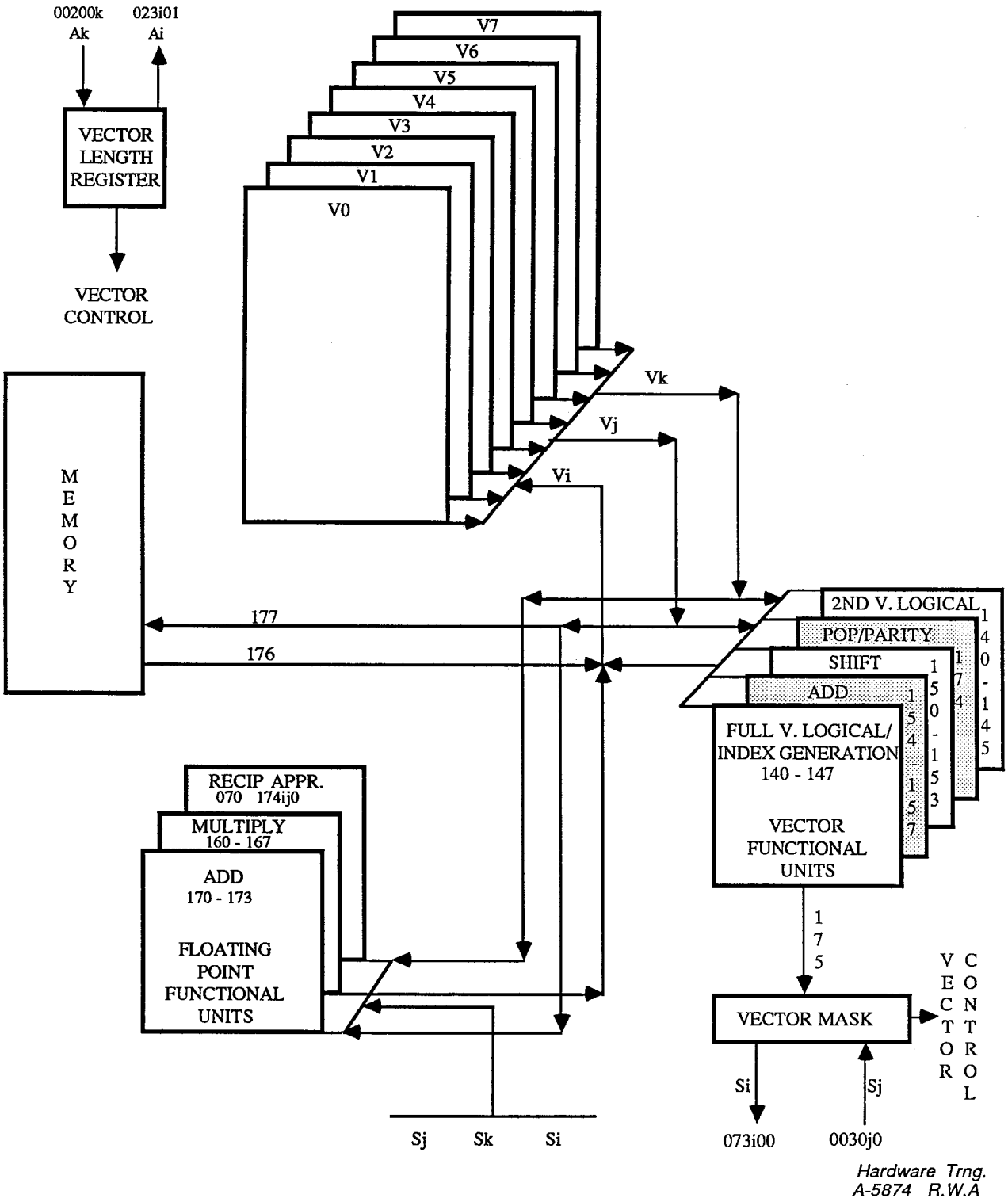
CRAY X-MP 3HD MODULE EXCHANGE REFERENCE



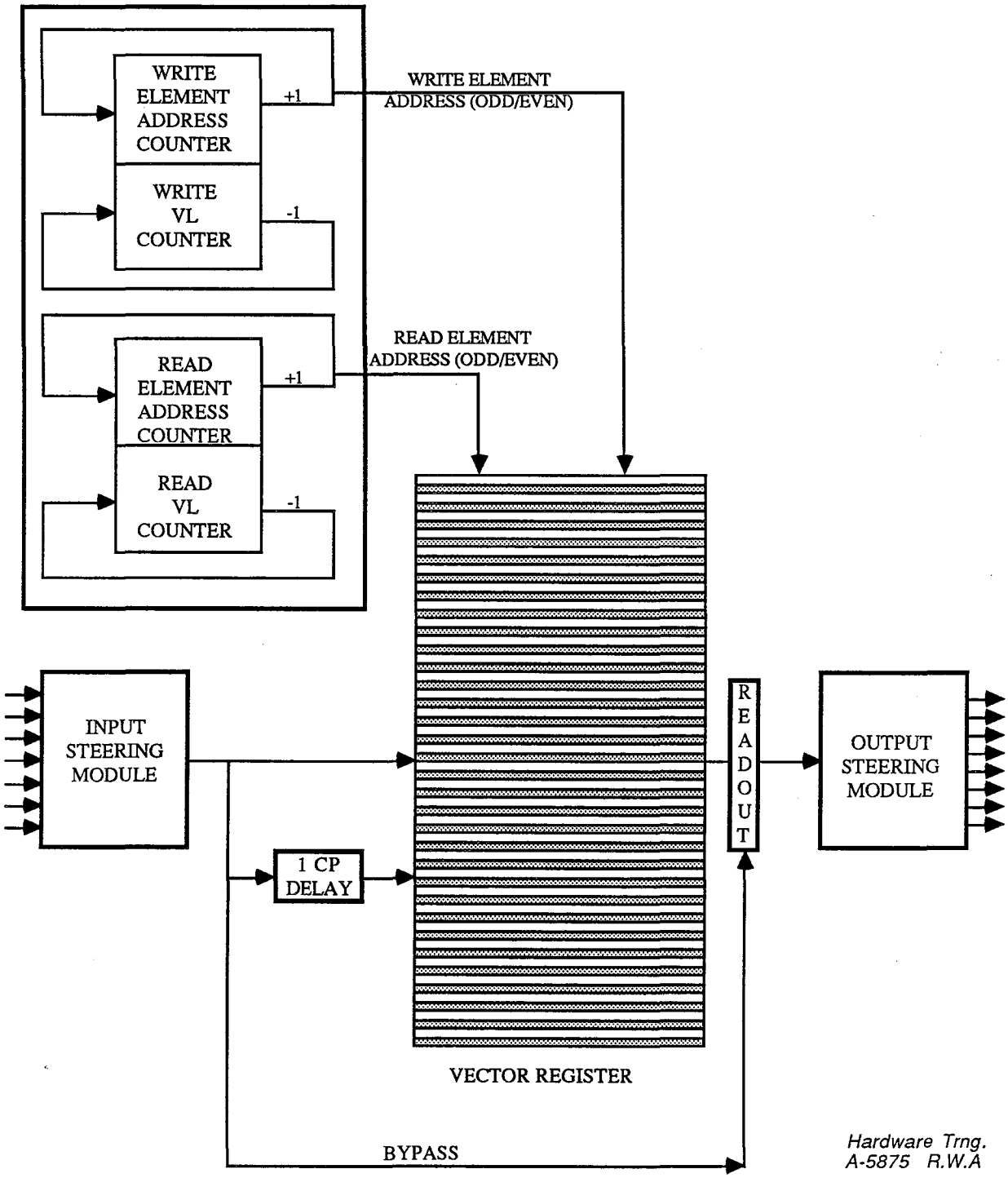
EXCHANGE PACKAGE



CRAY X-MP EXCHANGE DATA

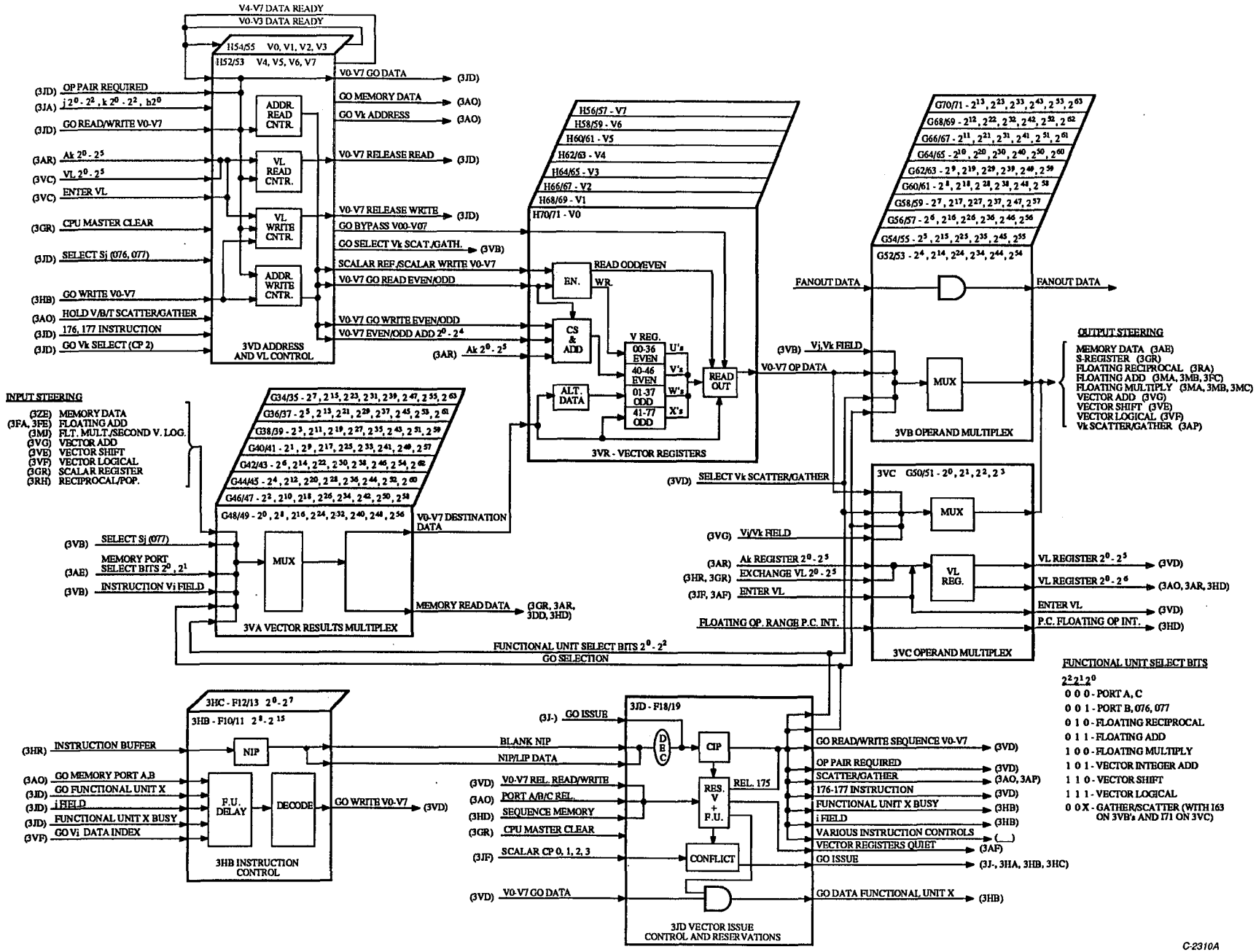


CRAY X-MP VECTOR REGISTER

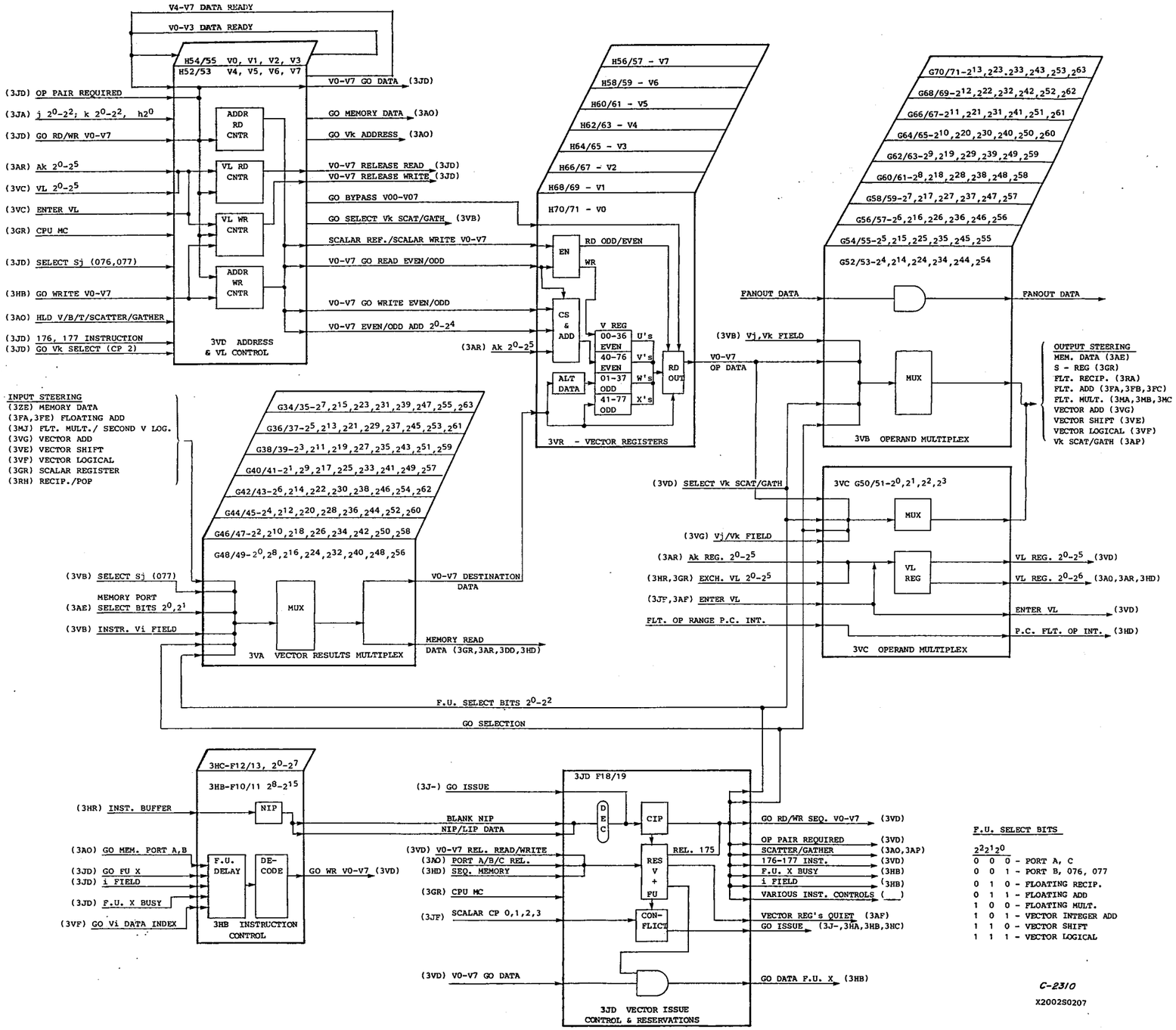


CRAY X-MP VECTOR REGISTER DATA FLOW

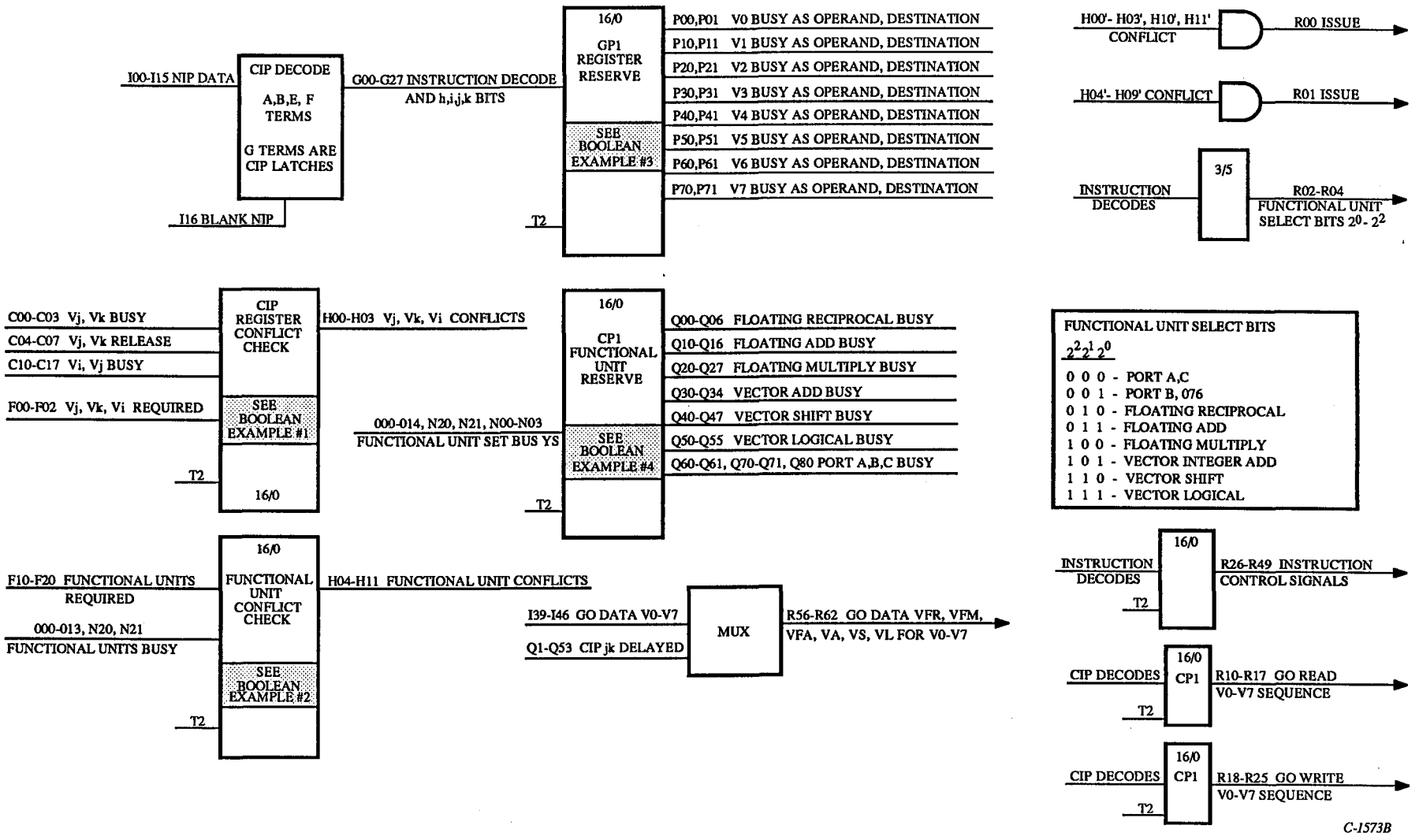
00200k	-	Transmit (Ak) to VL register.
002000	-	Transmit 1 to Vl register.
023i01	-	Transmit (VL) to Ai.
076ijk	-	Transmit (Vj, element (Ak)) to Si.
077ijk	-	Transmit (Sj) to Vi element (Ak).
077iok	-	Clear Vi element (Ak)
176i0k	-	Read (VL) words to Vi from (Ao) incremented by (Ak).
176i00	-	Read (VL) words to Vi from (Ao) incremented by 1.
176i1k	-	Read (VL) words to Vi using ((Ao) + (Vk)).
1770jk	-	Store (VL) words from Vj to (Ao) increment by (Ak).
1770j0	-	Store (VL) words from Vj to (Ao) incremented by 1.
1771jk	-	Store (VL) words from Vj using ((Ao) + (Vk)).
155ijk	-	Integer sums of (Vj) and (Vk) to Vi.



CRAY X-MP VECTOR REGISTER AND CONTROL BLOCK DIAGRAM (CPU 0)



CRAY X-MP2 VECTOR REGISTER AND CONTROL
BLOCK DIAGRAM (CPU 0)



CRAY X-MP 3JD MODULE

3JD MODULE BOOLEAN EXAMPLE #1

REGISTER CONFLICT

```
Boolean   H00' = C4 C0' C2' Z0' F0' # # H0' # # C6 ; T2     VK CONFLICT
          H00  = (F0 + H0) • (Z0 + H0) • (C2 • C0) • (C4' + C6') ; T2
```

VK CONFLICT = (VK REQ + VK CONFLICT) • (ISSUE + VK CONFLICT) • (VK BUSY) • $\overline{\text{(VK RELEASE)}}$

3JD MODULE BOOLEAN EXAMPLE #2

FU CONFLICT (FLOATING POINT)

```
Boolean   H04 = N20 01 0Ø 07 X0 # F10 F21 # F11 F12 ; T2      VFU BUSY
          H04 = X0 F10 F21 + 07 F10 F21 + 0Ø F10 + 01 F11 + N20 F12
```

VFU BUSY = (FR BUSY FOR VP • SELECT FR • VP INST) + (SCALAR FR • SELECT FR • VP INST)
+ (FR BUSY • SELECT FR) + (FA BUSY • SELECT FA) + (FM BUSY • SELECT FM)

MEMORY CONFLICT

```
Boolean     H07 = F16 010 011 + F17 010 + F18 011 ; T2     MEMORY PORT BUSY
```

MEMORY PORT BUSY = (SELECT PORT A + B • PORT A BUSY • PORT B BUSY) + (SELECT PORT A • PORT A BUSY) + (SELECT PORT B • PORT B BUSY)

C-1573

3JD MODULE BOOLEAN EXAMPLE #3

```
Boolean   P00' = P00' Z0' + M00' + D00 ; T2          V0 JK BUSY
          P00  = (P00 + Z0) • M00 • D00' ; T2
```

V0 JK BUSY = (V0 JK BUSY + ISSUE) • V0 JK set BUSY • $\overline{\text{RELEASE JK V0}}$

```
Boolean   M00  = K0 L2' + K10 K42' L2' + P0          V0 JK BUSY
```

V0 JK set BUSY = (Vk REQ • k = 0 • VIJ CONFLICT) + (Vj REQ • j = 0 • $\overline{\text{076,077 INST}}$ • $\overline{\text{VIJ CONFLICT}}$) + V0 JK BUSY hold.

```
Boolean   P01' = P01' Z0' + M01' + D10 ; T2          V0 I BUSY
          P01  = (P01 + Z0) • M01 • D10' ; T2
```

```
Boolean   M01  = K20 K42' L2' + P1                   V0 I BUSY
```

V0 I set BUSY = Vi REQ • i = 0 • $\overline{\text{076,077 INST}}$ • $\overline{\text{VIJ CONFLICT}}$ + V0 I BUSY hold

C-1573

3JD MODULE BOOLEAN EXAMPLE #4

FU BUSY

```
Boolean   Q10' = Q10 Z0' + 01' ; T2                    FA BUSY
          Q10  = (Q10 + Z0) • 01 ; T2
```

FA BUSY held = (FA BUSY held + ISSUE) • FA set BUSY

```
Boolean   001  = G19' K33 L3' + V1' S1                 FA BUSY
```

FA SET BUSY = $\overline{\text{VFU REQ. (SCALAR)}}$ • FA SELECTED • $\overline{\text{VFU CONFLICT}}$ + RELEASE FA • FA BUSY held)

```
Boolean   S01' = Q10'                                  FA BUSY
```

MEMORY PORT BUSY

```
Boolean   Q60' = Q60' Z0' + # N0' + 015 ; T2           PORT A BUSY
          Q60  = (Q60 + Z0) • N0 • 015' ; T2
```

PORT A BUSY held = (PORT A BUSY + ISSUE) • PORT A set BUSY • $\overline{\text{RELEASE PORT A}}$

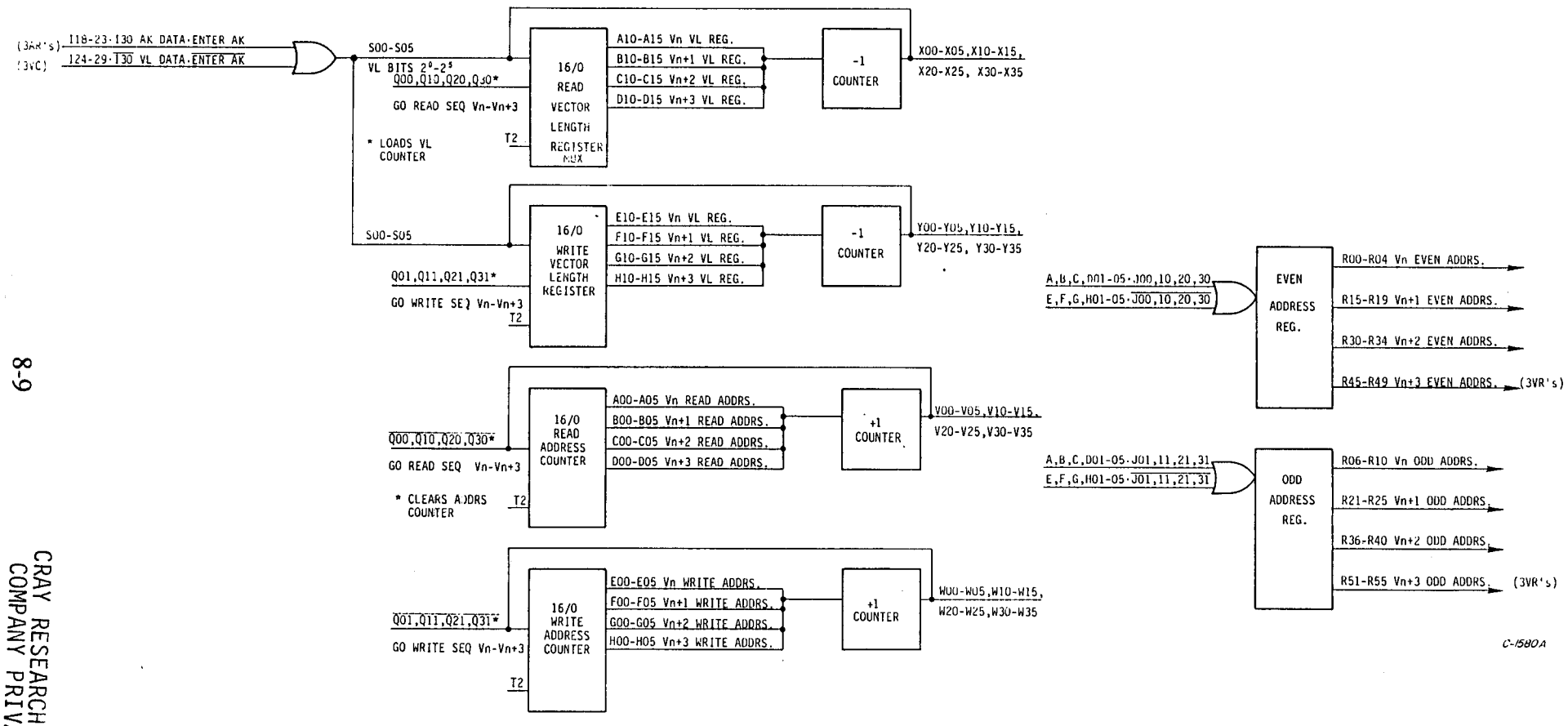
```
Boolean   N00  = 070 G22 G9' L4' + K40 G9' L4' + Q60
```

PORT A set BUSY = (PORT B BUSY • 176,177 INST • $\overline{\text{10X-13X INST}}$ • $\overline{\text{MEMORY CONFLICT}}$) + B MEM.
REF. • $\overline{\text{10X-13X INST}}$ • $\overline{\text{MEMORY CONFLICT}}$) + PORT A BUSY held

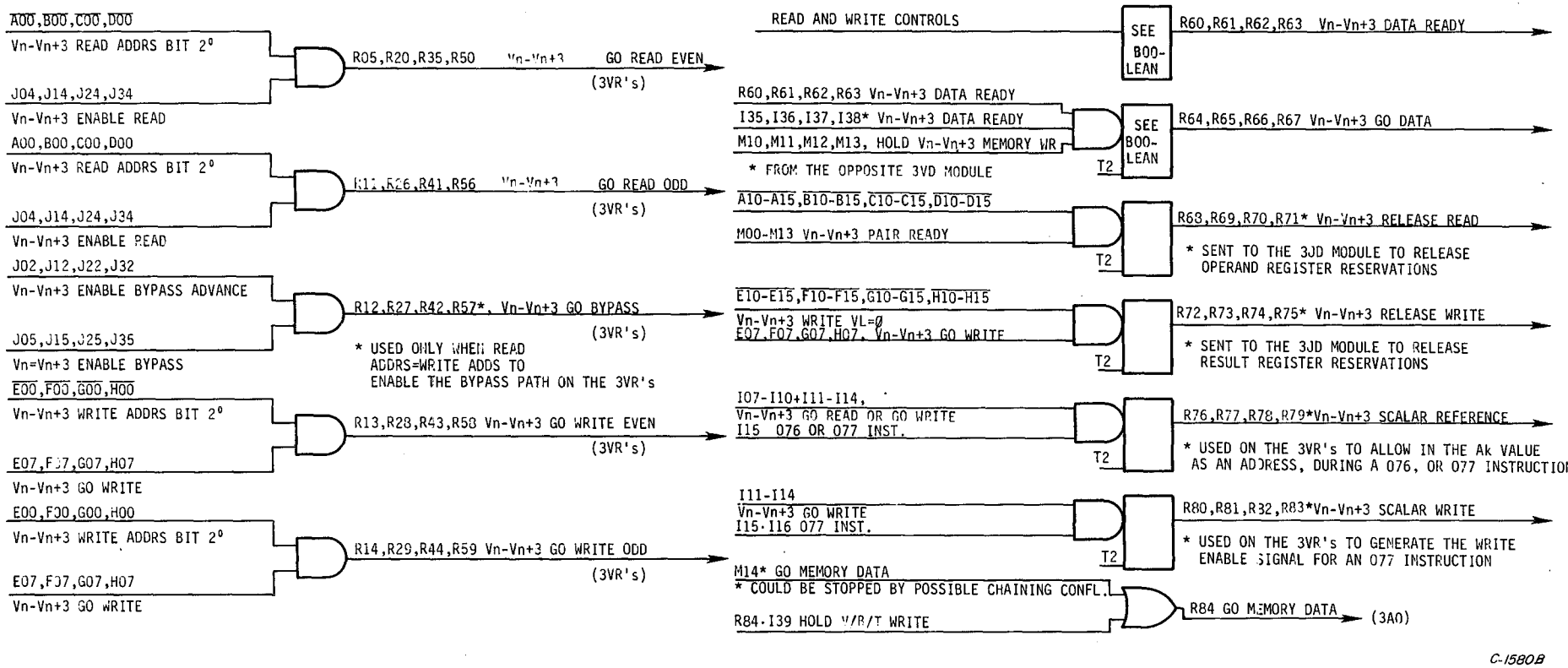
```
Boolean   015  = I35 + # # L6                         RELEASE PORT A
```

RELEASE PORT A = RELEASE PORT A + M.C.

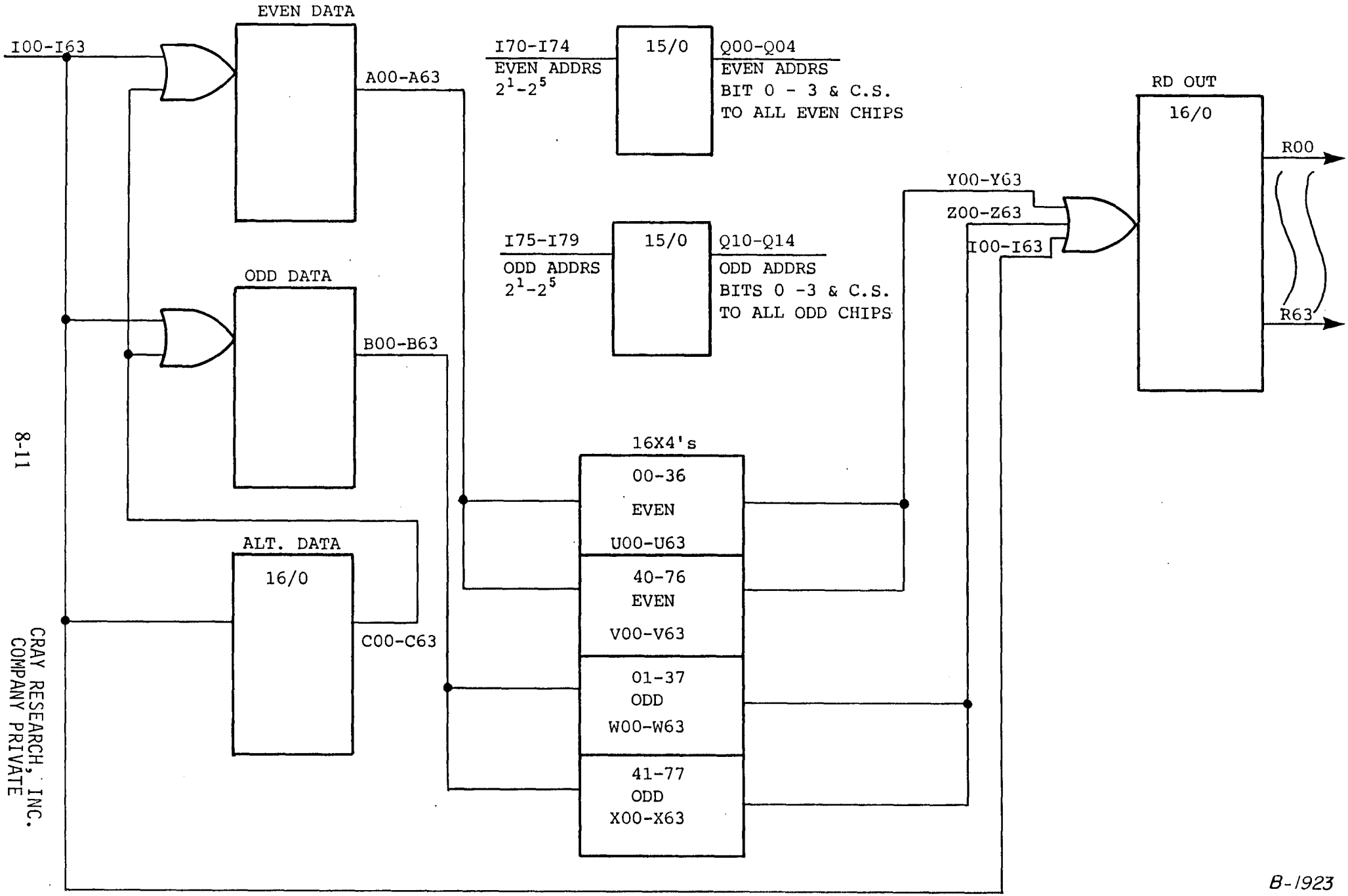
C-1573



3VD MODULE (SHEET 1 of 2)



CRAY X-MP 3VD MODULE



B-1923

CRAY X-MP 3VR MODULE

BIT POSITION

VECTOR REGISTER ELEMENT	2^0	2^1	2^2	2^3	2^4	2^5	2^6	→	2^{63}
00_8 - 36_8 (EVEN)	U0	U1	U2	U3	U4	U5	U6	→	U63
40_8 - 76_8 (EVEN)	V0	V1	V2	V3	V4	V5	V6	→	V63
01_8 - 37_8 (ODD)	W0	W1	W2	W3	W4	W5	W6	→	W63
41_8 - 77_8 (ODD)	X0	X1	X2	X3	X4	X5	X6	→	X63

X-MP VECTOR DATA
CPU - 0 (3VR)

	V0	V1	V2	V3	V4	V5	V6	V7
X-MP 4	G66/67	G64/65	G62/63	G60/61	G58/59	G56/57	G54/55	G52/53
X-MP 2	H70/71	H68/69	H66/67	H64/65	H62/63	H60/61	H58/59	H56/57

BIT	T.P.	TERM	BIT	T.P.	TERM	BIT	T.P.	TERM	BIT	T.P.	TERM
2^0	B72	R00	2^{16}	B02	R16	2^{32}	D72	R32	2^{48}	D05	R48
2^1	B69	R01	2^{17}	B01	R17	2^{33}	D68	R33	2^{49}	D04	R49
2^2	A01	R02	2^{18}	A64	R18	2^{34}	C02	R34	2^{50}	C50	R50
2^3	A02	R03	2^{19}	A70	R19	2^{35}	C01	R35	2^{51}	C52	R51
2^4	B64	R04	2^{20}	B09	R20	2^{36}	D67	R36	2^{52}	D06	R52
2^5	B68	R05	2^{21}	B23	R21	2^{37}	D65	R37	2^{53}	D07	R53
2^6	A07	R06	2^{22}	A62	R22	2^{38}	C03	R38	2^{54}	C49	R54
2^7	A05	R07	2^{23}	A60	R23	2^{39}	C21	R39	2^{55}	C48	R55
2^8	B67	R08	2^{24}	B71	R24	2^{40}	D70	R40	2^{56}	D02	R56
2^9	B70	R09	2^{25}	B04	R25	2^{41}	D69	R41	2^{57}	D01	R57
2^{10}	A22	R10	2^{26}	A71	R26	2^{42}	C25	R42	2^{58}	C38	R58
2^{11}	A21	R11	2^{27}	A66	R27	2^{43}	C22	R43	2^{59}	C72	R59
2^{12}	B66	R12	2^{28}	B05	R28	2^{44}	D71	R44	2^{60}	D03	R60
2^{13}	B10	R13	2^{29}	A53	R29	2^{45}	D66	R45	2^{61}	D08	R61
2^{14}	A04	R14	2^{30}	A72	R30	2^{46}	C23	R46	2^{62}	C71	R62
2^{15}	A25	R15	2^{31}	A69	R31	2^{47}	C24	R47	2^{63}	C47	R63

SYNC.	TEST POINT	BOOLEAN TERM
WRITE EVEN	$\overline{B27}$	Q06
WRITE ODD	$\overline{C58}$	Q16

SYNC. ON WRITE EVEN FOR EVEN NUMBERED ELEMENTS AND WRITE ODD FOR ODD NUMBERED ELEMENTS ON THE MODULE THE DATA IS GOING TO ENTER. SYNC IS 1 CP AHEAD OF DATA.

3VR REGISTER CHIP LOCATION CHART

BITS	EVEN ELEMENTS 0-36	EVEN ELEMENTS 40-76	ODD ELEMENTS 1-37	ODD ELEMENTS 41-77
	U	V	W	X
0-3	B-SL	B-UL	A-GA	A-IA
4-7	B-SK	B-UK	A-GB	A-IB
8-11	B-SJ	B-UJ	A-GC	A-IC
12-15	B-SI	B-UI	A-GD	A-ID
16-19	B-SA	B-UA	A-GL	A-IL
20-23	B-SB	B-UB	A-GK	A-IK
24-27	B-SC	B-UC	A-GJ	A-IJ
28-31	B-SD	B-UD	A-GI	A-II
32-35	D-SL	D-UL	C-GA	C-IA
36-39	D-SK	D-UK	C-GB	C-IB
40-43	D-SJ	D-UJ	C-GC	C-IC
44-47	D-SI	D-UI	C-GD	C-ID
48-51	D-SA	D-UA	C-GL	C-IL
52-55	D-SB	D-UB	C-GK	C-IK
56-59	D-SC	D-UC	C-GJ	C-IJ
60-63	D-SD	D-UD	C-GI	C-II

X2008S0403

TEST POINTS FOR DATA - FROM FLOATING ADD
RECIPROCAL, PORT A OF MEMORY OR PORT B OF MEMORY

	V0 DATA		V1 DATA		V2 DATA		V3 DATA		V4 DATA		V5 DATA		V6 DATA		V7 DATA	
	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM
BIT N	$\overline{A29}$	G00	$\overline{A33}$	G01	$\overline{A31}$	G02	$\overline{A35}$	G03	B50	$\overline{G04}$	B48	$\overline{G05}$	B42	$\overline{G06}$	B46	$\overline{G07}$
BIT N+8	$\overline{A40}$	G10	$\overline{A41}$	G11	$\overline{A45}$	G12	$\overline{A42}$	G13	B37	$\overline{G14}$	B38	$\overline{G15}$	B35	$\overline{G16}$	B32	$\overline{G17}$
BIT N+16	$\overline{C23}$	G20	$\overline{C25}$	G21	$\overline{C38}$	G22	$\overline{C40}$	G23	D59	$\overline{G24}$	D48	$\overline{G25}$	D42	$\overline{G26}$	D38	$\overline{G27}$
BIT N+24	$\overline{C20}$	G30	$\overline{C27}$	G31	$\overline{C51}$	G32	$\overline{C50}$	G33	D29	$\overline{G34}$	D27	$\overline{G35}$	D26	$\overline{G36}$	D25	$\overline{G37}$
BIT N+32	$\overline{A52}$	G40	$\overline{A55}$	G41	$\overline{A62}$	G42	$\overline{A63}$	G43	B14	$\overline{G44}$	B16	$\overline{G45}$	B27	$\overline{G46}$	B23	$\overline{G47}$
BIT N+40	$\overline{A67}$	G50	$\overline{A59}$	G51	$\overline{A66}$	G52	$\overline{A69}$	G53	B06	$\overline{G54}$	B12	$\overline{G55}$	B09	$\overline{G56}$	B10	$\overline{G57}$
BIT N+48	$\overline{C65}$	G60	$\overline{C66}$	G61	$\overline{C68}$	G62	$\overline{C59}$	G63	D09	$\overline{G64}$	D13	$\overline{G65}$	D12	$\overline{G66}$	D17	$\overline{G67}$
BIT N+56	$\overline{C63}$	G70	$\overline{C61}$	G71	$\overline{C70}$	G72	$\overline{C72}$	G73	D35	$\overline{G74}$	D03	$\overline{G75}$	D06	$\overline{G76}$	D05	$\overline{G77}$

TEST POINTS FOR DATA - FROM FLOATING MULTIPLY,
VECTOR ADD, VECTOR SHIFT OR VECTOR LOGICAL

	TEST POINT	BOOL. TERM													TEST POINT	BOOL. TERM
BIT N	$\overline{A26}$	H00	$\overline{A25}$	H01	$\overline{A19}$	H02	$\overline{A27}$	H03	B65	$\overline{H04}$	B55	$\overline{H05}$	B52	$\overline{H06}$	B29	$\overline{H07}$
BIT N+8	$\overline{A13}$	H10	$\overline{A07}$	H11	$\overline{A43}$	H12	$\overline{A46}$	H13	B33	$\overline{H14}$	B36	$\overline{H15}$	B31	$\overline{H16}$	B28	$\overline{H17}$
BIT N+16	$\overline{C33}$	H20	$\overline{C29}$	H21	$\overline{C34}$	H22	$\overline{C42}$	H23	D33	$\overline{H24}$	D34	$\overline{H25}$	D30	$\overline{H26}$	D58	$\overline{H27}$
BIT N+24	$\overline{C47}$	H30	$\overline{C31}$	H31	$\overline{C55}$	H32	$\overline{C57}$	H33	D66	$\overline{H34}$	D40	$\overline{H35}$	D23	$\overline{H36}$	D14	$\overline{H37}$
BIT N+32	$\overline{A50}$	H40	$\overline{A47}$	H41	$\overline{A65}$	H42	$\overline{A61}$	H43	B13	$\overline{H44}$	B25	$\overline{H45}$	B19	$\overline{H46}$	B18	$\overline{H47}$
BIT N+40	$\overline{A58}$	H50	$\overline{A57}$	H51	$\overline{A72}$	H52	$\overline{A70}$	H53	B51	$\overline{H54}$	B08	$\overline{H55}$	B14	$\overline{H56}$	B05	$\overline{H57}$
BIT N+48	$\overline{C44}$	H60	$\overline{C46}$	H61	$\overline{C35}$	H62	$\overline{C48}$	H63	D46	$\overline{H64}$	D44	$\overline{H65}$	D04	$\overline{H66}$	D10	$\overline{H67}$
BIT N+56	$\overline{C19}$	H70	$\overline{C52}$	H71	$\overline{C41}$	H72	$\overline{C67}$	H73	D18	$\overline{H74}$	D08	$\overline{H75}$	D16	$\overline{H76}$	D31	$\overline{H77}$
	V0		V1		V2		V3		V4		V5		V6		V7	

SYNC. ON WRITE EVEN FOR EVEN ELEMENTS AND WRITE ODD FOR ODD ELEMENTS
ON THE 3VR MODULE THE DATA IS GOING TO.

WRITE EVEN T.P. $\overline{B27}$ BOOL. TERM Q06
WRITE ODD T.P. C58 BOOL. TERM Q16

CPU Ø 3VA LOCATIONS X-MP 2	X-MP 4	BIT N	BIT N+8	BIT N+16	BIT N+24	BIT N+32	BIT N+40	BIT N+48	BIT N+56
G48/49	H44/45	2^0	2^8	2^{16}	2^{24}	2^{32}	2^{40}	2^{48}	2^{56}
G46/47	H42/43	2^2	2^{10}	2^{18}	2^{26}	2^{34}	2^{42}	2^{50}	2^{58}
G44/45	H40/41	2^4	2^{12}	2^{20}	2^{28}	2^{36}	2^{44}	2^{52}	2^{60}
G42/43	H38/39	2^6	2^{14}	2^{22}	2^{30}	2^{38}	2^{46}	2^{54}	2^{62}
G40/41	H36/37	2^1	2^9	2^{17}	2^{25}	2^{33}	2^{41}	2^{49}	2^{57}
G38/39	H34/35	2^3	2^{11}	2^{19}	2^{27}	2^{35}	2^{43}	2^{51}	2^{59}
G36/37	H32/33	2^5	2^{13}	2^{21}	2^{29}	2^{37}	2^{45}	2^{53}	2^{61}
G34/35	H30/31	2^7	2^{15}	2^{23}	2^{31}	2^{39}	2^{47}	2^{55}	2^{63}

MODULE LOCATIONS AND BIT POSITIONS ARE THE SAME FOR THE OTHER CPU's
CHASSIS DESIGNATOR ARE DIFFERENT.

3VA TEST POINT CHART

C-1949A

X4101S0103

		Vj DATA MEMORY / Vk DATA VECTOR LOG.		Vj DATA Si / Vj DATA VECTOR LOG.		Vj DATA FLT. RECIP. / Vj DATA VECTOR SHIFT		Vj DATA FLT. ADD / Vk DATA VECTOR ADD		Vk DATA FLT. ADD / Vj DATA VECTOR ADD		Vj DATA FLT. MULTIPLY / Vk DATA FLT. MULTIPLY		SELECT Vk SCATTER/ GATHER	
		TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM
	BIT N	$\overline{B04}$	G00	$\overline{B03}$	G01	$\overline{B10}$	G02	$\overline{B08}$	G03	$\overline{B18}$	G04	$\overline{B16}$	G05		
		B40	G11	B48	G10	B36	G09	B35	G08	B33	G07	B31	G06	$\overline{B21}$	G72
	BIT N+10	$\overline{A70}$	G12	$\overline{A68}$	G13	$\overline{A63}$	G14	$\overline{A61}$	G15	$\overline{A50}$	G16	$\overline{A52}$	G17		
		A36	G23	A34	G22	A32	G21	A38	G70	A29	G19	A27	G18	$\overline{A39}$	G73
V0-V3	N+20	$\overline{D48}$	G24	$\overline{D46}$	G25	$\overline{D36}$	G26	$\overline{D38}$	G27	$\overline{D29}$	G28	$\overline{D27}$	G29		
		D04	G35	D03	G34	D12	G33	D10	G32	D20	G31	D18	G30		
OPERAND DATA	N+30	$\overline{C29}$	G36	$\overline{C27}$	G37	$\overline{C36}$	G38	$\overline{C31}$	G39	$\overline{C38}$	G40	$\overline{C40}$	G41		
		C70	G47	C72	G46	C63	G45	C65	G44	C52	G43	C46	G42		
	N+40	$\overline{B57}$	G48	$\overline{B50}$	G49	$\overline{B65}$	G50	$\overline{B63}$	G51	$\overline{B70}$	G52	$\overline{B72}$	G53		
		D50	G59	D52	G58	D67	G57	D72	G56	D65	G55	D59	G54		
	N+50	$\overline{A31}$	G60	$\overline{A16}$	G61	$\overline{A15}$	G62	$\overline{A18}$	G63	$\overline{A06}$	G64	$\overline{A08}$	G65		
		C03	G71	C04	G70	C05	G69	C06	G68	C12	G67	C10	G66		

		TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM	TEST POINT	BOOL. TERM
	BIT N	$\overline{B06}$	H00	$\overline{B20}$	H01	$\overline{B25}$	H02	$\overline{B23}$	H03	$\overline{B14}$	H04	$\overline{B12}$	H05		
		B43	H11	B38	H10	B45	H09	B27	H08	B47	H07	B29	H06	$\overline{B24}$	H72
	BIT N+10	$\overline{A72}$	H12	$\overline{A67}$	H13	$\overline{A65}$	H14	$\overline{A59}$	H15	$\overline{A55}$	H16	$\overline{A57}$	H17		
		A40	H23	A44	H22	A48	H21	A46	H20	A43	H19	A42	H18	$\overline{A26}$	H73
V4-V7	N+20	$\overline{D42}$	H24	$\overline{D44}$	H25	$\overline{D40}$	H26	$\overline{D35}$	H27	$\overline{D33}$	H28	$\overline{D31}$	H29		
		D06	H35	D07	H34	D14	H33	D16	H32	D23	H31	D25	H30		
OPERAND DATA	N+30	$\overline{C23}$	H36	$\overline{C33}$	H37	$\overline{C35}$	H38	$\overline{C44}$	H39	$\overline{C42}$	H40	$\overline{C48}$	H41		
		C68	H47	C67	H46	C55	H45	C59	H44	C57	H43	C50	H42		
	N+40	$\overline{B52}$	H48	$\overline{B55}$	H49	$\overline{B59}$	H50	$\overline{B61}$	H51	$\overline{B67}$	H52	$\overline{B68}$	H53		
		D63	H59	D55	H58	D68	H57	D70	H56	D60	H55	D57	H54		
	N+50	$\overline{A23}$	H60	$\overline{A25}$	H61	$\overline{A13}$	H62	$\overline{A20}$	H63	$\overline{A10}$	H64	$\overline{A11}$	H65		
		C18	H71	C20	H70	C07	H69	C08	H68	C14	H67	C16	H66		

SYNC ON READ EVEN FOR EVEN ELEMENTS DATA & READ ODD FOR ODD ELEMENTS DATA ON THE 3VR MODULES THAT ARE THE OPERANDS FOR THE FUNCTION. SYNC IS 2CP's AHEAD OF THE DATA.

READ EVEN T.P. $\overline{A58}$ BOOLEAN TERM Q05
READ ODD T.P. C40 BOOLEAN TERM Q15

CPU 0

3VB LOCATIONS		BIT N	BIT N+10	BIT N+20	BIT N+30	BIT N+40	BIT N+50
X-MP 2	X-MP 4						
G52/53	H48/49	2^{4}	2^{14}	2^{24}	2^{34}	2^{44}	2^{54}
G54/55	H50/51	2^{5}	2^{15}	2^{25}	2^{35}	2^{45}	2^{55}
G56/57	H52/53	2^{6}	2^{16}	2^{26}	2^{36}	2^{46}	2^{56}
G58/59	H54/55	2^{7}	2^{17}	2^{27}	2^{37}	2^{47}	2^{57}
G60/61	H56/57	2^{8}	2^{18}	2^{28}	2^{38}	2^{48}	2^{58}
G62/63	H58/59	2^{9}	2^{19}	2^{29}	2^{39}	2^{49}	2^{59}
G64/65	H60/61	2^{10}	2^{20}	2^{30}	2^{40}	2^{50}	2^{60}
G66/67	H62/63	2^{11}	2^{21}	2^{31}	2^{41}	2^{51}	2^{61}
G68/69	H64/65	2^{12}	2^{22}	2^{32}	2^{42}	2^{52}	2^{62}
G70/71	H66/67	2^{13}	2^{23}	2^{33}	2^{43}	2^{53}	2^{63}

MODULE LOCATIONS AND BIT POSITIONS ARE THE SAME FOR THE OTHER CPU's CHASSIS DESIGNATE OR ARE DIFFERENT

B-1140C

X4101S0104

NOTE:
** BITS 2^{3}-2^{0} ARE ON THE 3VC MODULE BUT ARE NOT TEST POINTED.

X-MP 3VB MODULE TEST POINTS FOR
VECTOR DATA TO THE FUNCTIONAL UNITS

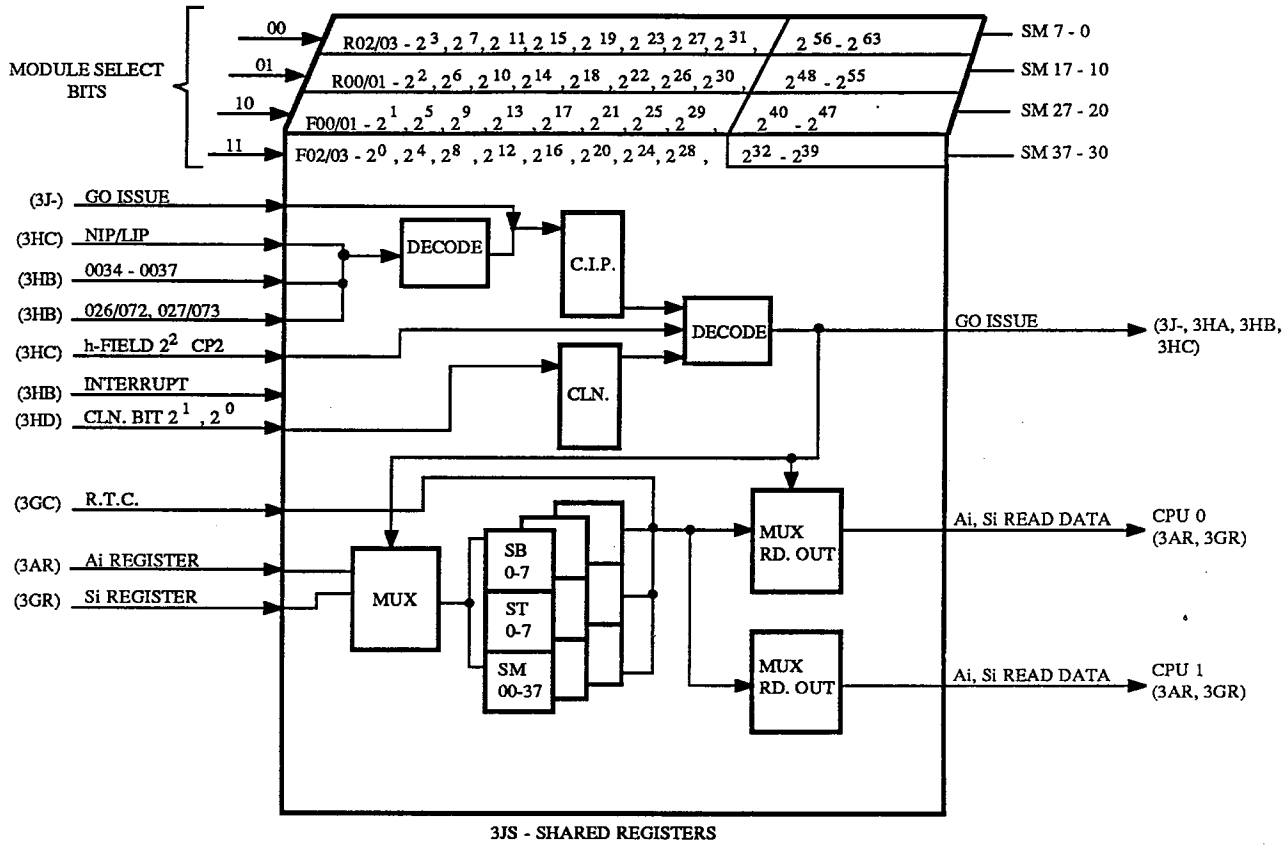
SHARED REGISTER INSTRUCTIONS:

SB/ST

026ij7	Ai	SBj	TRANSMIT (SBj) TO Ai	(READ SB)
027ij7	SBj	Ai	TRANSMIT (Ai) TO SBj	(WRITE SB)
072ij3	Si	STj	TRANSMIT (STj) TO Si	(READ ST)
073ij3	STj	Si	TRANSMIT (Si) TO STj	(WRITE ST)

SM

0034jk	SMjk	1, TS	TEST & SET, SMjk	$0 \leq jk \leq 31_{10}$
0036jk	SMjk	0	CLEAR SMjk	$0 \leq jk \leq 31_{10}$
0037jk	SMjk	1	SET SMjk	$0 \leq jk \leq 31_{10}$
072i02	Si	SM	TRANSMIT (SM) TO Si	(READ SEMAPHORES)
073i02	SM	Si	TRANSMIT (Si) TO SM	(LOAD SEMAPHORES)



* All inputs and outputs, excluding R.T.C., are for both CPU 0 and CPU 1. There are three sets of SB, ST, and SM's of which one set is selected by CLN.

B-0609D

CRAY X-MP SHARED REGISTERS AND CONTROL

3JS MODULE
SHARED AND SEMAPHORE REGISTER
TEST POINTS

BITS ON MODULE				BIT POSITION	CLUSTER #1		CLUSTER #2		CLUSTER #3		
					TERM	T.P.	TERM	T.P.	TERM	T.P.	
56	48	40	32	SM7,17,27,37 n	G00	B31	G08	B29	G16	A06	
57	49	41	33	. . . . n+1	G01	B25	G09	B27	G17	A08	
58	50	42	34	. . . . n+2	G02	B72	G10	B70	G18	A12	
59	51	43	34	. . . . n+3	G03	B68	G11	B67	G19	A14	
60	52	44	36	. . . . n+4	G04	B65	G12	B63	G20	A35	} LATCHES
61	53	45	37	. . . . n+5	G05	B59	G13	B61	G21	A37	
62	54	46	38	. . . . n+6	G06	B55	G14	B50	G22	A38	
63	55	47	39	SM0,10,20,30 n+7	G07	B57	G15	B52	G23	A40	

MODULE LOCATIONS
R2 R0 F0 F2

R2 R0 F0 F2

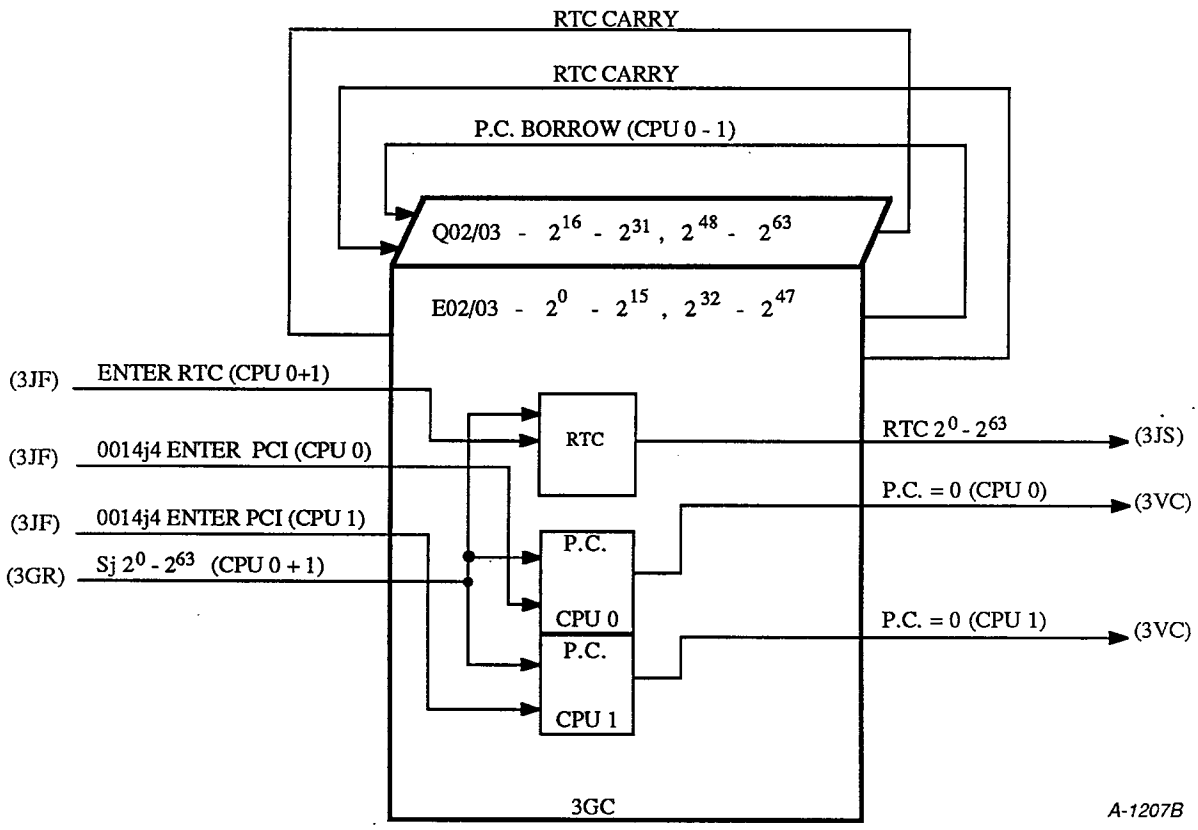
SHARED/SM READ OUT REGISTER

BITS ON MODULE				*BIT POSITION	CPU0 T.P.'s		CPU1 T.P.'s		CLUSTER # 1	CLUSTER #2	CLUSTER #3	
					TERM	T.P.	TERM	T.P.				
3	2	1	0	n	R06	C53	R22	C58	S00	S16	S32	
7	6	5	4	n+4	R07	C54	R23	C55	S01	S17	S33	
11	10	9	8	n+8	R08	C48	R24	C49	S02	S18	S34	
15	14	13	12	n+12	R09	C45	R25	C47	S03	S19	S35	
19	18	17	16	n+16	R10	D04	R26	D02	S04	S20	S36	
23	22	21	20	n+20	R11	D22	R27	D15	S05	S21	S37	
27	26	25	24	n+24	R12	D34	R28	D18	S06	S22	S38	
31	30	29	28	n+28	R13	D20	R29	D32	S07	S23	S39	} 16 X 4'S
56	48	40	32	n	R14	C43	R30	C42	S08	S24	S40	
57	49	41	33	n+1	R15	C44	R31	C46	S09	S25	S41	
58	50	42	34	n+2	R16	C36	R32	C35	S10	S26	S42	
59	51	43	35	n+3	R17	C37	R33	C34	S11	S27	S43	
60	52	44	36	n+4	R18	D25	R34	D21	S12	S28	S44	
61	53	45	37	n+5	R19	D23	R35	D27	S13	S29	S45	
62	54	46	38	n+6	R20	D39	R36	D35	S14	S30	S46	
63	55	47	39	n+7	R21	D41	R37	D30	S15	S31	S47	

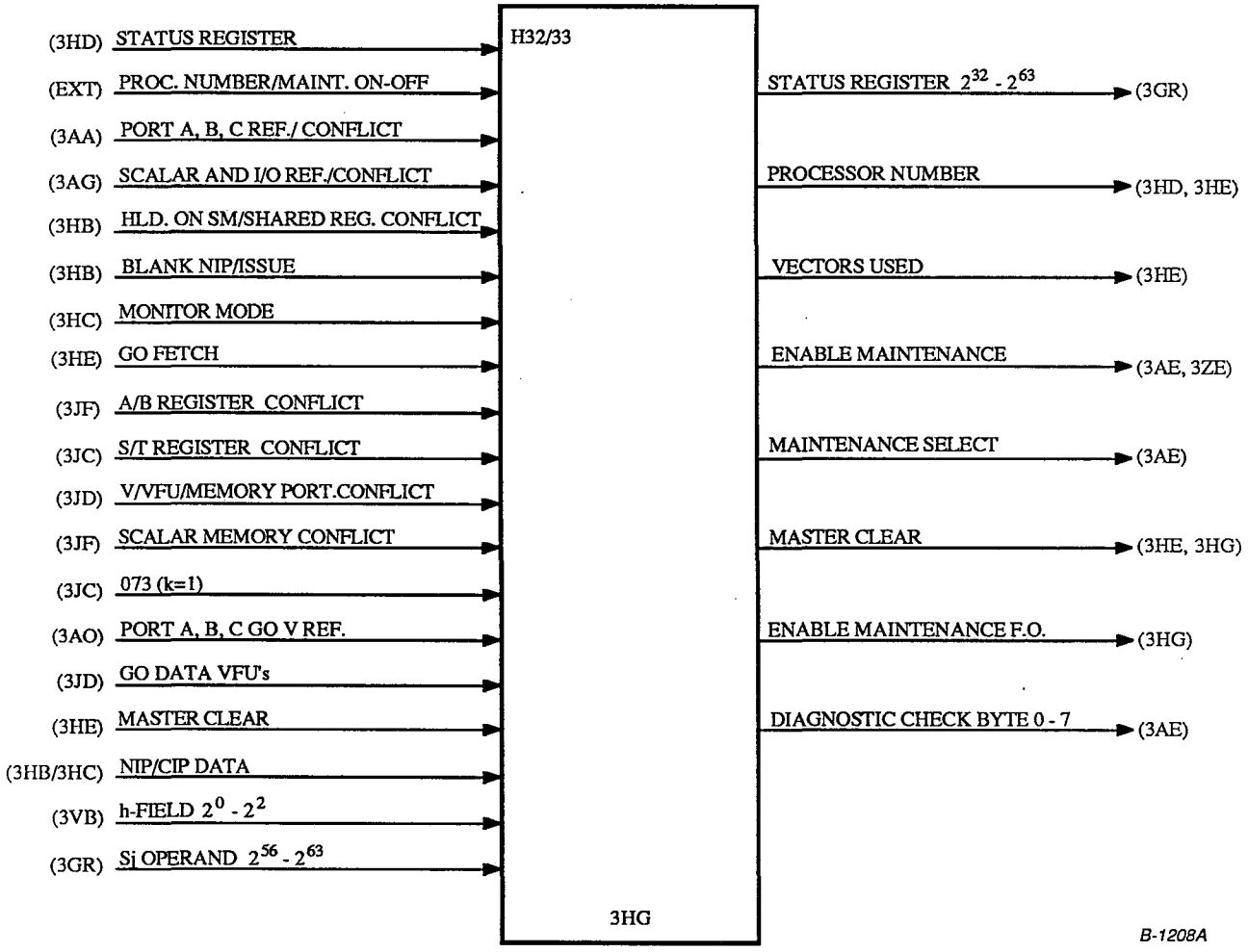
* The bit position layout on the 3JS module is every 4th bit through 2^{31} then consecutive bit positions through 2^{63}.

B-1914

THE READ OUT REGISTER IS USED FOR THE REAL TIME CLOCK DATA ALSO.



CRAY X-MP REAL TIME/PROGRAMMABLE CLOCK



CRAY X-MP PERFORMANCE LOG

PURPOSE: To monitor certain hardware related events during the execution of a user program in order to evaluate relative performance.

DESCRIPTION: The performance monitor consists of eight 46-bit counters which can be selected to monitor one of four groups of 8 performance related hardware events in the CRAY X-MP. The maximum increment of any counter is 3 per clock period. The 46-bit counters can be expected to monitor a time interval of about 62 hours before being reset.

The performance events are monitored only while in non-monitor mode. The instructions to select the monitoring group and read the counter values are executed in monitor mode.

X2011X0405

0015j0 - Monitor Mode, Select Performance Monitor

This instruction selects one of four groups of hardware events which will be monitored during the non-monitor mode intervals. Each group consists of eight counters, each 46 bits wide.

The lower order 2 bits of the j field in this instruction select the group. This instruction will also clear all 8 performance counters.

During each clock period in non-monitor mode, the 8 performance counters will advance their counts according to the total number of selected events which will occur. Entering Monitor Mode disables advancing of the performance counters.

The following list shows the performance counters selected for Performance Monitor j, $0 \leq j \leq 3$.

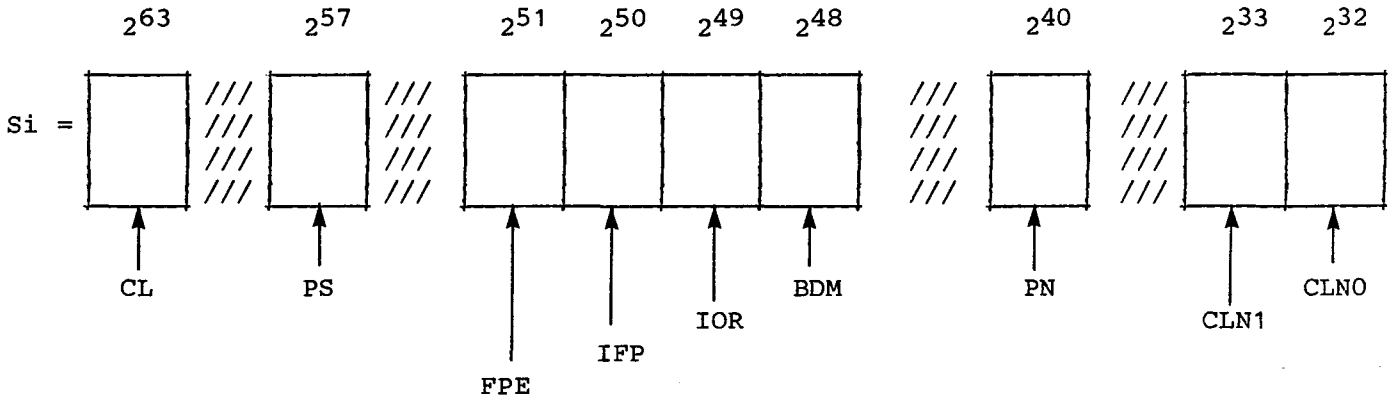
X2011S0406

PERFORMANCE MONITOR (j)	PERFORMANCE COUNTER	DESCRIPTION	INCREMENT PER CP
0	0	# of instructions ISSUED	+1
	1	# of CP's HOLDING ISSUE	+1
	2	# of FETCHES	+1
	3	# of I/O REFERENCES	+1
	4	# of CPU REFERENCES	+3 max
	5	# of F.A. OPERATIONS	+1
	6	# of F.M. OPERATIONS	+1
	7	# of F.R. OPERATIONS	+1
1	0	SEMAPHORES	+1
	1	SHARED REGISTERS	+1
	2	A REGISTERS	+1
	3	S REGISTERS	+1
	4	V REGISTERS	+1
	5	V FUNCTIONAL UNITS	+1
	6	SCALAR MEMORY	+1
	7	BLOCK MEMORY	
2	0	# of FETCHES	+1
	1	# of SCALAR REFERENCES	+1
	2	# of SCALAR CONFLICTS	+1
	3	# of I/O REFERENCES	+1
	4	# of I/O CONFLICTS	+1
	5	# of BLOCK REFERENCES	+3 max
	6	# of BLOCK CONFLICTS	+3 max
	7	# of VECTOR MEMORY REFERENCE	+3 max
3	0	# of 000 - 017 INSTRUCTIONS	+1
	1	# of 020 - 137 INSTRUCTIONS	+1
	2	# of 140 - 157,175 INSTRUCTIONS	+1
	3	# of 160 - 174 INSTRUCTIONS	+1
	4	# of 176, 177 INSTRUCTIONS	+1
	5	# of V. INTEGER OPERATIONS	+3 max
	6	# of FLOATING-POINT OPERATIONS	+3 max
	7	# of VECTOR MEMORY REFERENCES	+3 max

X2011S0407

073i01 - Read Status --> Si

This instruction returns the following status to the high-order bits of Si



(CL)	Clustered, CLN = 0	--> Si Bit 63
(PS)	Program State	--> Si Bit 57
(FPE)	Floating Point Error Occurred	--> Si Bit 51
(IFP)	Floating Point Interrupt Enabled	--> Si Bit 50
(IOR)	Operand Range Interrupt Enabled	--> Si Bit 49
(BDM)	Bidirectional Memory Enabled	--> Si Bit 48
*(PN)	Processor Number	--> Si Bit 40
*(CLN1)	Cluster Number Bit 1	--> Si Bit 33
*(CLN0)	Cluster Number Bit 0	--> Si Bit 32

* NOTE: These bit positions return the value of zero if executed in non-monitor mode.

073i11 - MONITOR MODE, COUNTER

This instruction returns either the upper bits or the lower bits of a performance counter to the upper bits of Si.

Entering monitor mode clears the performance counter pointer and disables advancing of all performance counters while in Monitor Mode. After execution of each 073i11 instruction, the performance counter pointer is advanced by one. The even values of the performance counter pointer select the lower order bits of a performance counter to be read into Si as follows:

Performance Counter Bits 2^0 - 2^{25} Si Bits 2^{32} - 2^{57}

The odd values of the pointer select the upper bits of a performance counter to be read into Si as follows:

Performance Counter Bits 2^{26} - 2^{45} Si Bits 2^{38} - 2^{57}

NOTE: There must be a 2 clock period delay between sequential 073i11 instructions.

A sequence for reading the performance counter is as follows:

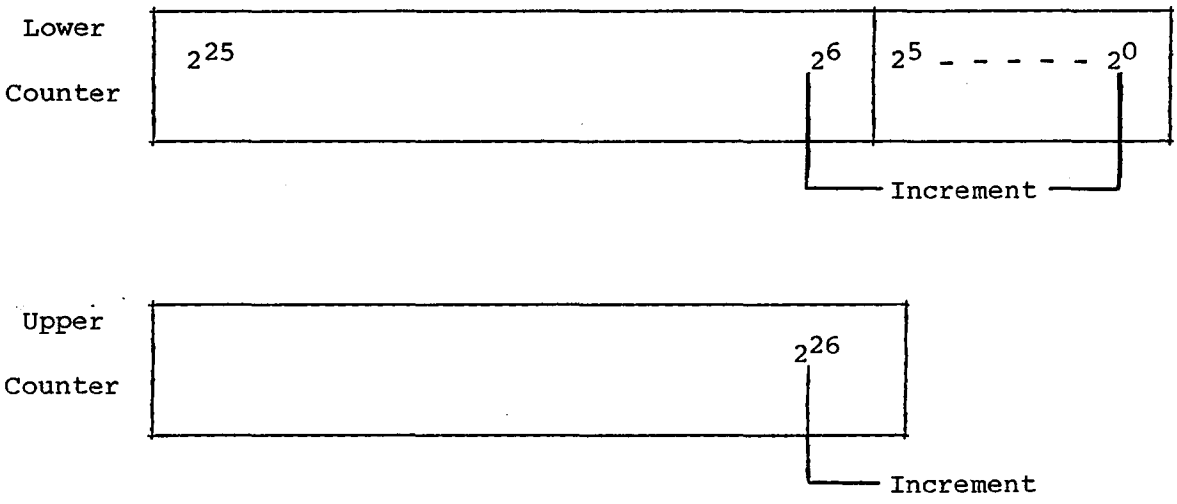
```
        Monitor Mode
c       073i11              Lower Bits Counter 0      Si
        [2 cp delay
c       073i11              Upper Bits Counter 0      Si
        [2 cp delay
c       073i11              Lower Bits Counter 1      Si
        [2 cp delay
c       073i11              Upper Bits Counter 1      Si
```

X2011S0409

073i21 - Monitor Mode, Maintenance (Increment Counter)

This is a hardware maintenance instruction which will increment a performance counter while in Monitor Mode. The even values of the performance counter pointer select Bit 2^0 and 2^6 of the counter to be incremented by one each time this instruction is executed. The odd values of the pointer select Bit 2^{26} to be incremented. The pointer can be advanced from even to odd and to the next counter via the 073i11 instruction.

NOTE: There must be a one clock period delay between sequential 073i21 instructions.



SECDED MAINTENANCE MODE FUNCTIONS

PURPOSE: - To allow verification of checkbit memory storage.

- To allow verification of checkbit generation.

- To allow verification of error detection and correction.

DESCRIPTION:

Additional logic has been added to the modules involved with SECDED to ease the checkout of these modules in specialized maintenance modes.

The instructions utilized for SECDED checkout are as follows:

INSTRUCTIONS

073x31 - Monitor Mode, Clear Modes

This instruction clears all 3 possible SECDED maintenance modes; ZEM1, AEM1, AEM2.

c
001501 - Monitor Mode, Set ZEM1 Mode

c Sets the 3ZE Maintenance Mode 1. This Mode enables the 8 memory checkbits c to replace certain data bits on any subsequent memory read.

c The data bits are replaced as follows:

c Check Bit 2^0 replaces Data Bit 2^{46}
c Check Bit 2^1 replaces Data Bit 2^{47}
c Check Bit 2^2 replaces Data Bit 2^{62}
c Check Bit 2^3 replaces Data Bit 2^{63}
c Check Bit 2^4 replaces Data Bit 2^{14}
c Check Bit 2^5 replaces Data Bit 2^{15}
c Check Bit 2^6 replaces Data Bit 2^{30}
c Check Bit 2^7 replaces Data Bit 2^{31}

NOTE: This mode can be enabled only when the maintenance mode switch is on.

c 001511 - Monitor Mode, S1 --> Diagnostic Checkbyte

This instruction loads the diagnostic checkbyte with the highest order 8 bits of S1 as follows:

c S1 Bit 2^{56} --> Diagnostic Checkbyte Bit 2^{0}
c S1 Bit 2^{57} --> Diagnostic Checkbyte Bit 2^{1}
c S1 Bit 2^{58} --> Diagnostic Checkbyte Bit 2^{2}
c S1 Bit 2^{59} --> Diagnostic Checkbyte Bit 2^{3}
c S1 Bit 2^{60} --> Diagnostic Checkbyte Bit 2^{4}
c S1 Bit 2^{61} --> Diagnostic Checkbyte Bit 2^{5}
c S1 Bit 2^{62} --> Diagnostic Checkbyte Bit 2^{6}
c S1 Bit 2^{63} --> Diagnostic Checkbyte Bit 2^{7}

001521 - Monitor Mode, Set AEM1 Mode

Sets the 3AE Maintenance Mode 1. This mode enables certain data bits to replace the 8 check bits being written into memory for any subsequent CPU write to memory.

NOTE: Any I/O Write to memory is not affected by this maintenance mode.

The check bits are replaced as follows:

Data Bit 2^{46} replaces Check Bit 2^{0}
Data Bit 2^{47} replaces Check Bit 2^{1}
Data Bit 2^{62} replaces Check Bit 2^{2}
Data Bit 2^{63} replaces Check Bit 2^{3}
Data Bit 2^{14} replaces Check Bit 2^{4}
Data Bit 2^{15} replaces Check Bit 2^{5}
Data Bit 2^{30} replaces Check Bit 2^{6}
Data Bit 2^{31} replaces Check Bit 2^{7}

NOTE: This mode can be enabled only when the maintenance mode switch is on.

001531 - Monitor Mode, Set AEM2 Mode

Sets the 3AE Maintenance Mode 2. This mode enables the diagnostic checkbyte to replace the 8 check bits being written into memory for any subsequent CPU write to memory.

NOTE: Any I/O Write to memory is not affected by this maintenance mode.

The check bits are replaced as follows:

Diagnostic Checkbyte Bit 2^0 replaces Check Bit 2^0
Diagnostic Checkbyte Bit 2^1 replaces Check Bit 2^1
Diagnostic Checkbyte Bit 2^2 replaces Check Bit 2^2
Diagnostic Checkbyte Bit 2^3 replaces Check Bit 2^3
Diagnostic Checkbyte Bit 2^4 replaces Check Bit 2^4
Diagnostic Checkbyte Bit 2^5 replaces Check Bit 2^5
Diagnostic Checkbyte Bit 2^6 replaces Check Bit 2^6
Diagnostic Checkbyte Bit 2^7 replaces Check Bit 2^7

NOTE: This mode can be enabled only when the maintenance mode switch is on.

MAINTENANCE MODE SWITCH

The Maintenance Mode Switch enables the following hardware maintenance instructions which then can be executed in Monitor Mode:

c 1 001501 - Set ZE Maint. Mode 1: Checkbyte --> Data Field.

c 2 001511 - Load S1 --> Diagnostic Checkbyte.

3 001521 - Set AE Maint. Mode 1: Data Bits --> Checkbyte.

4 001531 - Set AE Maint. Mode 2: Diag. Checkbyte --> Checkbyte.

5 073x31 - Clear all Error Correct Maintenance Modes (NOTE: Master clear also clears all error correct maintenance modes).

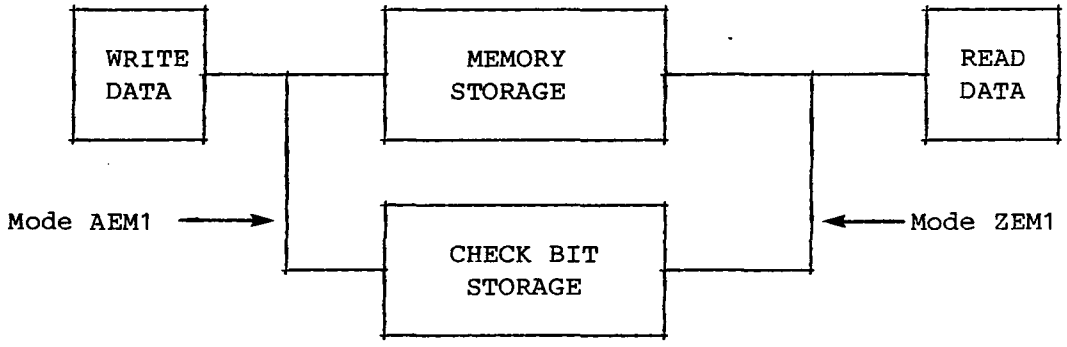
X2011S0413

USAGE

Since the program instructions for any test are written to memory through the I/O port, a fetch must not be allowed to take place while in maintenance mode ZEM1 is in effect. If a fetch is allowed to take place with this mode in effect, the checkbits will replace certain bits in the instructions, rendering the test useless. Therefore, this maintenance mode must be cleared before a fetch is executed. (One alternative would be to use the complementary maintenance mode AEM1 to initially rewrite the check bits of the test instructions, but this may get messy).

VERIFICATION OF CHECK BIT STORAGE

Two of the maintenance modes (AEM1, ZEM1) are used to verify the check bit storage modules. The mode AEM1 enables certain data bits to replace the 8 check bits being written into memory and the Mode ZEM1 enables those 8 check bits to replace certain data bits on a memory read. The maintenance modes AEM1 and ZEM1 are complementary with respect to check bit and data bit positions.

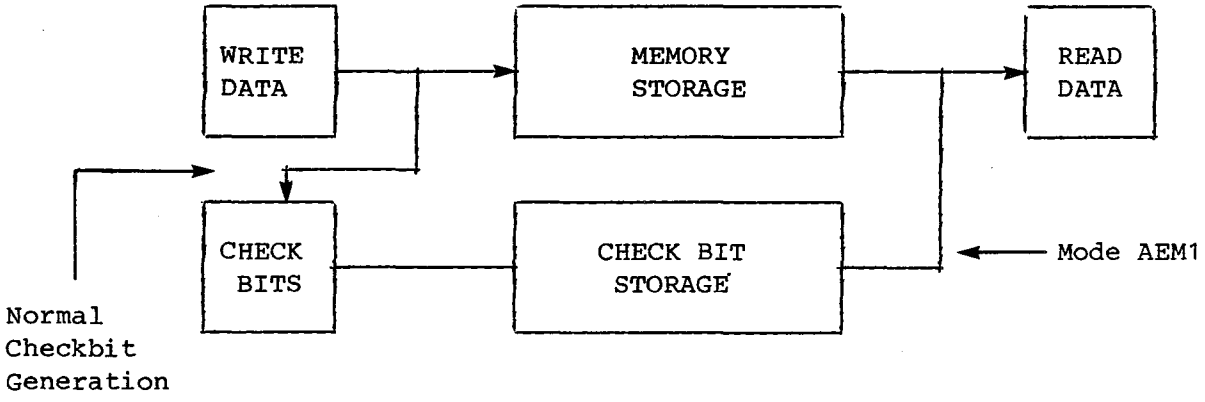


Modes AEM1, ZEM1

VERIFICATION OF CHECK BIT GENERATION

The maintenance mode ZEM1 can be used to verify the correct generation of check bits for a given word of data. With mode ZEM1 set, the 8 check bits replace certain data bits during a read from memory. The test program can then extract these check bits and verify their correctness.

NOTE: Error correction would normally be disabled during this test.

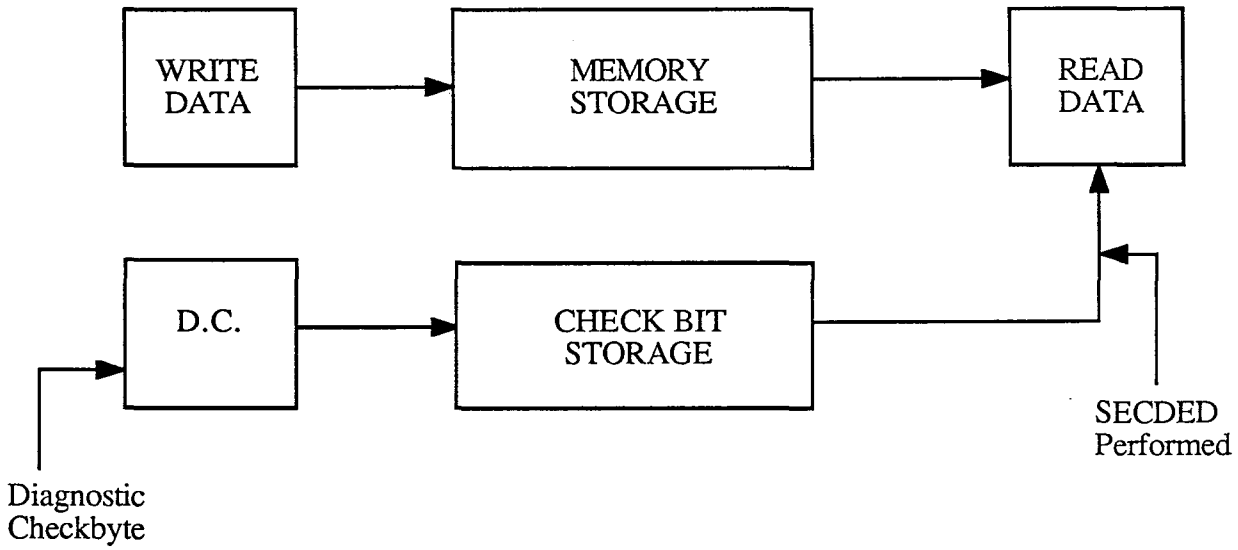


Mode ZEM1

VERIFICATION OF ERROR DETECTION AND CORRECTION

The maintenance mode AEM2 can be used to verify the correct operation of the error detection and correction circuitry by allowing a diagnostic checkbyte to be written to memory for a given data word.

The diagnostic checkbyte can be initially loaded with the upper order bits of S0 via the 001511 instruction. This diagnostic checkbyte will then be written into memory in place of the normal checkbits on any subsequent CPU write to memory. With error correction enabled, a subsequent read of the memory location containing the diagnostic checkbyte will allow different paths within SECDED to be checked out.



Mode AEM2

ADDRESS FUNCTIONAL UNITS

OBJECTIVES

With the aid of student reference material, upon completion of the course a student should be capable of:

I. Determining the failing module type, module location, and chip, given the failing instruction or operation and the data error.

II. Describing the main functions and differences of each of the following module types:

3AS for Address Adder
3AU and 3AT for Address Multiply

III. Defining with respect to time and use, the address and control signals.

IV. Tracing operands through the adder generating all the boolean terms.

V. Diagnostic applications:

Address Add - 1ADD, 1ARA
Address Mult - 1ARM, 1AMT

FUNCTIONAL UNIT TIMES

	CRAY 1S	CRAY X-MP
Address Add	1	1
Address Multiply	5	3
Scalar Add	2	2
Scalar Pop Count	3	3
Scalar Leading Zero	2	2
Scalar Single Shift	1	1
Scalar Double Shift	2	2
Scalar Logical	1	1
Vector Add	2	2
Vector Logical	1	1
Vector Pop Count	5	4
Vector Single Shift	3	2
Vector Double Right Shift	3	2
Vector Double Left Shift	3	3
Floating Add	5	5
Floating Multiply	6	6
Floating Reciprocal	13	13
Second Vector Logical	N/A	3

ADDRESS ADDERS (Module Involved)

3AS MODULE

The 3AS module receives the Ak and Aj operand bits 2^0 - 2^{23} from the 3AR when a 030ijk or 031ijk instruction is issued. The 3AS will first complement the Ak operand if a 031ijk instruction is issued. It will then add the Aj and Ak operand, incorporating a 2-bit look ahead add philosophy. The result takes one clock period in the functional unit and is then sent back to the 3AR module. The 3AS module is also used for Scalar Leading Zero Count, and a 033 instruction, and the floating-point constants.

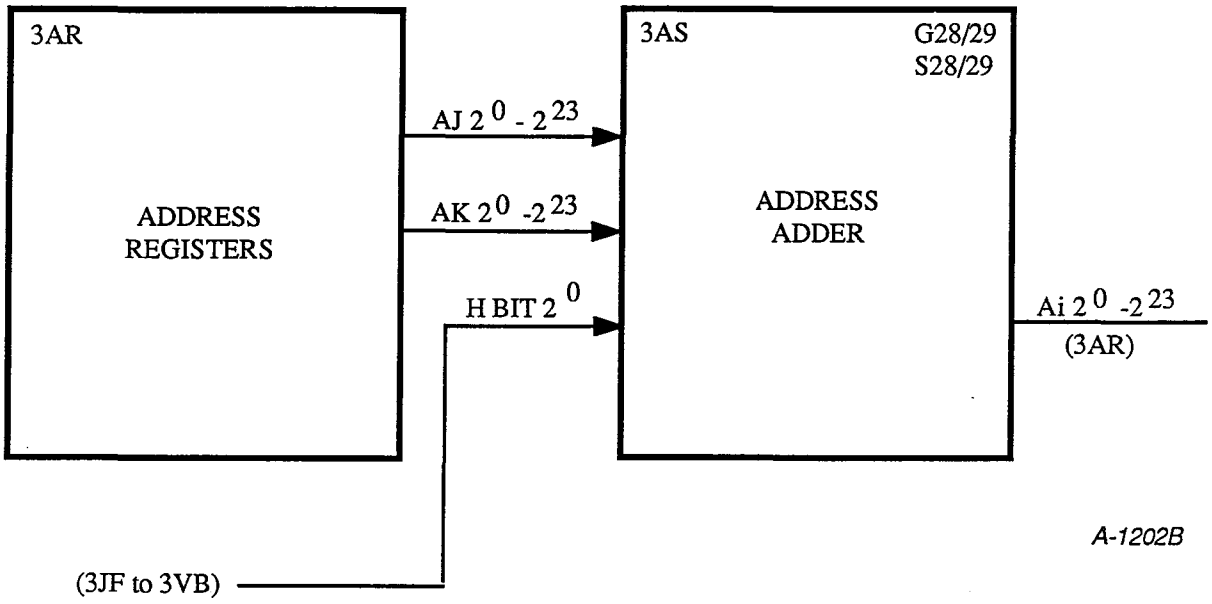
ADDRESS REGISTER

FUNCTIONAL UNIT

INSTRUCTIONS

ADDRESS ADD/SUBTRACT

030IJK	-	Integer sum of AJ and AK to AI
030I0K	-	Transmit AK to AI
030I00	-	Transmit 1 to AI
030IJ0	-	Integer sum of AJ and 1 to AI
031IJK	-	Integer difference AJ minus AK to AI
031I00	-	Transmit -1 to AI
031I0K	-	Transmit the negative of AK to AI
031IJ0	-	Integer difference of AJ minus 1 to AI

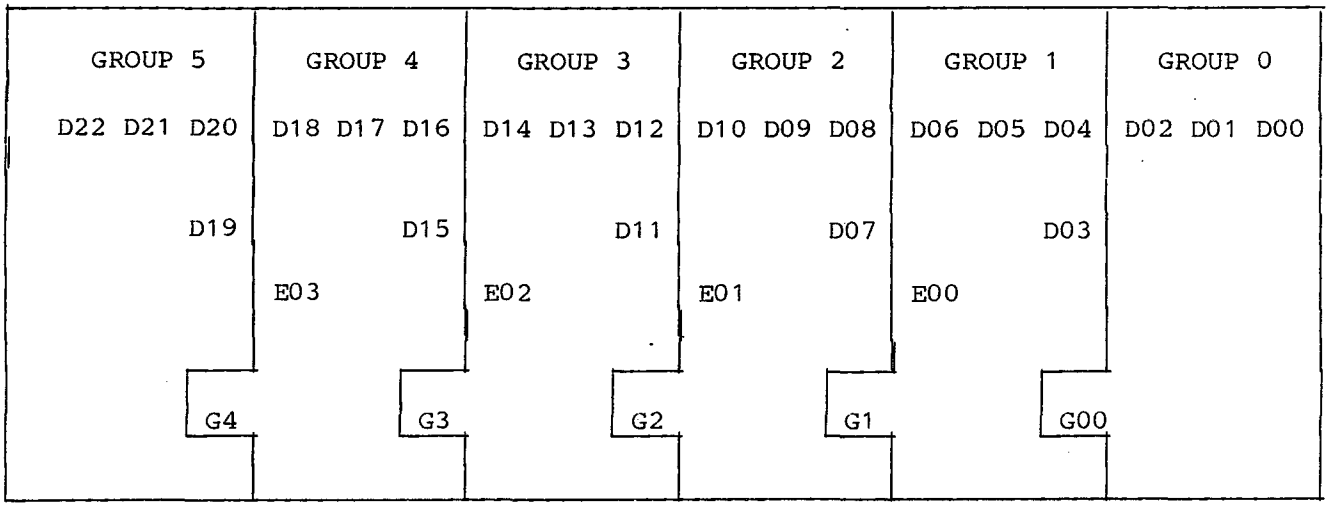


CRAY X-MP/2 ADDRESS ADDER BLOCK DIAGRAM

2-BIT LOOK-AHEAD ADD

0	0	1
0	1	1
ENABLE	SUM	CARRY/ENABLE

PROPAGATED GROUP CARRIES



D00-D02, D04-D06, D08-D10, D12-D14, D16-D18, D20-D22 - CARRIES WITHIN THE GROUP
D03, D07, D11, D15, D19 - CARRY OUTSIDE OF THE GROUP
E00, E01, E02, E03 - ALL BITS WITHIN THE GROUP ARE NOT ENABLES

G04 = D03 E00 E01 E02 E03 + D07 E01 E02 E03 + D11 E02 E03 + D15 E03 + D19

G03 = D03 E00 E01 E02 + D07 E01 E02 + D11 E02 + D15

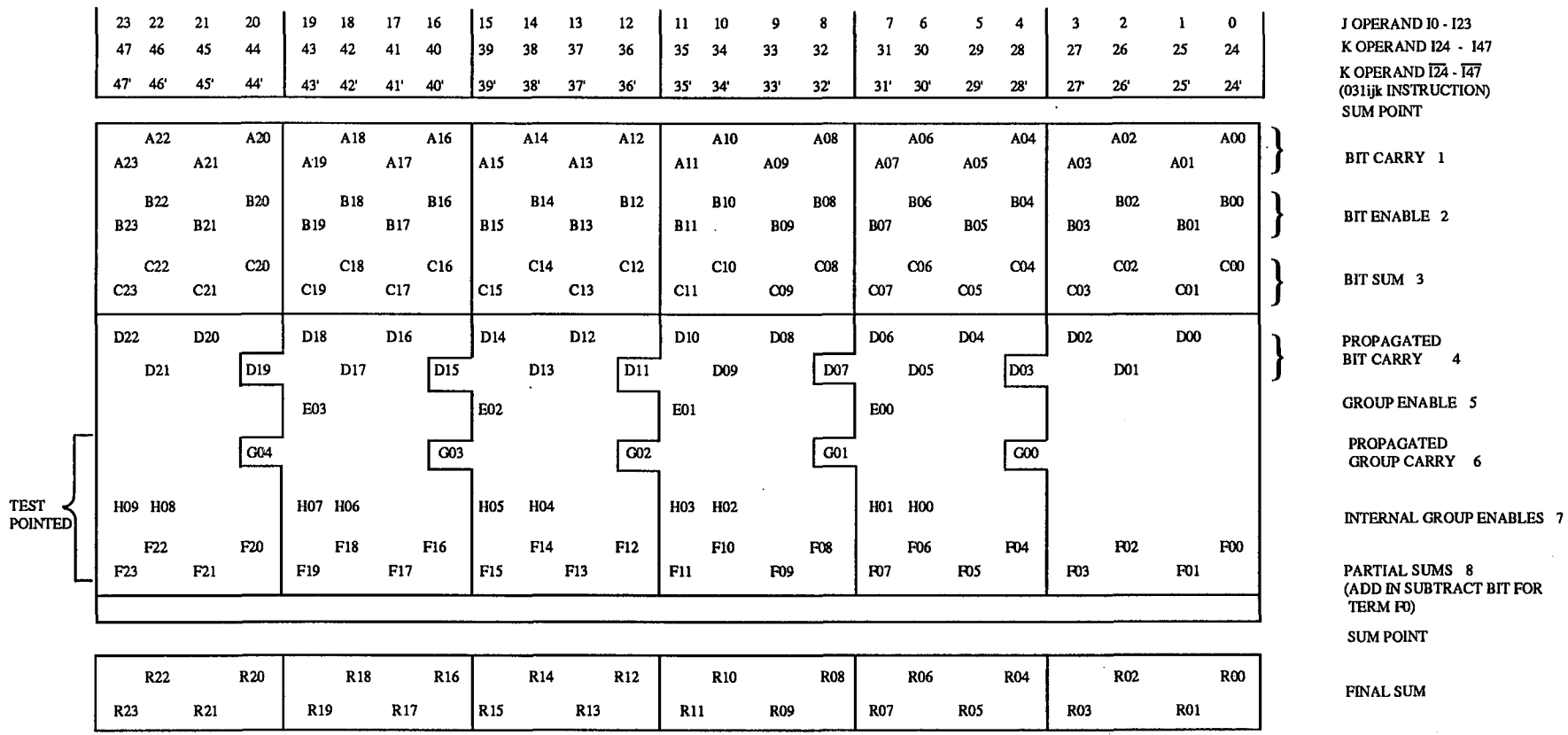
G02 = D03 E00 E01 + D07 E01 + D11

G01 = D03 E00 + D07

G00 = D03

ADDRESS ADDER 3AS MODULE

X2005C0101



1 Carry - Both input bits equal one

2 Enable - Both input bits equal zero

3 Sum - One but not both input bits equal a one

4 Bit Carry - Carry into the next higher bit position (looks at 4 bit group)

5 All bits within group are not enable

6 A carry propagated out of the highest bit position of the group

7 Bits below this bit are sum within

8 Half add between bit sum and bit carry; F0 term includes the addition of the subtract bit for 2's complement addition.

$$I93 = Q3 = M0 = \text{Subtract } (H2^0)$$
$$\frac{\text{T.P.}}{\text{C14}}$$

B-1042C

3AS MODULE ADDRESS ADDER

PROPAGATE CARRY GENERATE

TYPICAL PROPAGATE CARRY EQUATION:

D D03 = A3 A2 A1 A0 M0 # $\overline{B3}$ $\overline{B2}$ $\overline{B1}$ $\overline{B0}$ 0 #

REWRITE D03 EQUATION

C5 C4 C3 C2 C1 C0 E5 E4 E3 E2 E1 G

D03=A3 A2 A1 A0 M0 # $\overline{B3}$ $\overline{B2}$ $\overline{B1}$ $\overline{B0}$ 0 #

D03=#0 $\overline{B0}$ $\overline{B1}$ $\overline{B2}$ $\overline{B3}$ + M0 $\overline{B0}$ $\overline{B1}$ $\overline{B2}$ $\overline{B3}$ + A0 $\overline{B1}$ $\overline{B2}$ $\overline{B3}$

+ A1 $\overline{B2}$ $\overline{B3}$ + A2 $\overline{B3}$ + A3 #

LATCHED PROPAGATE CARRY GENERATE

TYPICAL PROPAGATE CARRY EQUATION:

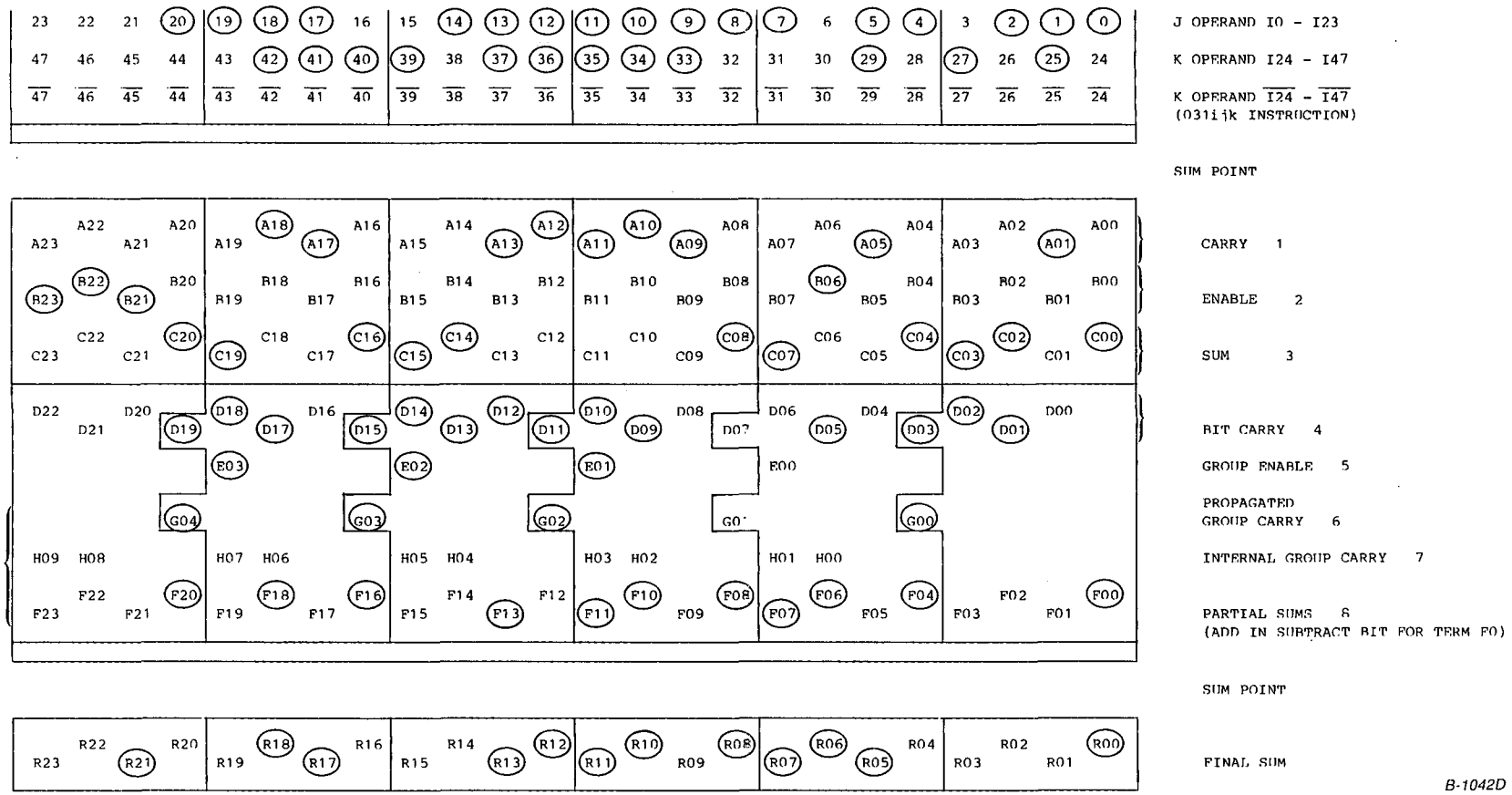
N * J00 = J0 M0 00 H0 $\overline{W0}$ # # $\overline{V4}$ V4 W0; T2

REWRITE J00 EQUATION

C4 C3 C2 C1 C0 E4 E3 E2 E1 F G

J00 =$\overline{J0}$ M0 0 0 H0 $\overline{W0}$ # # $\overline{V4}$ V4 W0; T2

J00 =H0 $\overline{V4}$ # # $\overline{W0}$ + 0 # # $\overline{W0}$ + 0 # $\overline{W0}$ + M0 $\overline{W0}$ V4 + J0 W0; T2



1 CARRY - BOTH INPUT BITS EQUAL ONE

2 ENABLE - BOTH INPUT BITS EQUAL ZERO

3 SUM - ONE BUT NOT BOTH INPUT BITS EQUAL A ONE

4 BIT CARRY - CARRY INTO THE NEXT HIGHER BIT POSITION (LOOKS AT 4 BIT GROUP)

5 ALL BITS WITHIN GROUP ARE NOT ENABLE

6 A CARRY PROPAGATED OUT OF THE HIGHEST BIT POSITION OF THE GROUP

7 BITS BELOW THIS BIT ARE SUM WITHIN

8 HALF ADD BETWEEN BIT SUM AND BIT CARRY; F0 TERM INCLUDES THE ADDITION OF THE SUBTRACT BIT FOR 2's COMPLEMENT ADDITION

$$I93 = \underset{\overline{C14}}{\overset{Q3}{\overline{T.P.}}} = M0 = \text{SUBTRACT } (H2^0)$$

$$\begin{array}{rl} 07477667 & \longrightarrow Aj \\ +\ 01737052 & \longrightarrow Ak \\ \hline 11436741 & \end{array}$$

3AS MODULE ADDRESS ADDER

ADDRESS MULTIPLY (Modules Involved)

3AT MODULE

The 3AT module receives the Ak and Aj bits 2^0 - 2^{15} from the 3AR when a 032ijk instruction is issued. The 3AT will first generate the Logical products on Ak and Aj bits, then sum the multiply pyramid using 3-bit adds. If there is a carry into bit position 2^{16} or 2^{17} the 3AT will send these bit positions to the 3AU. The results take 3CP's in the functional unit and are then sent back to the 3AR module. The 3AT will output bits 2^0 - 2^{15} of the summation to the 3AR.

3AU MODULE

The 3AU module operates the same as the 3AT module. However, the 3AU handles bits 2^0 - 2^{23} from the 3AR. The 3AU also looks for the 2^{16} and 2^{17} bits which could propagate from the 3AT and adds them into the pyramid summation. The 3AU will output bits 2^{16} - 2^{23} of the summation to the 3AR.

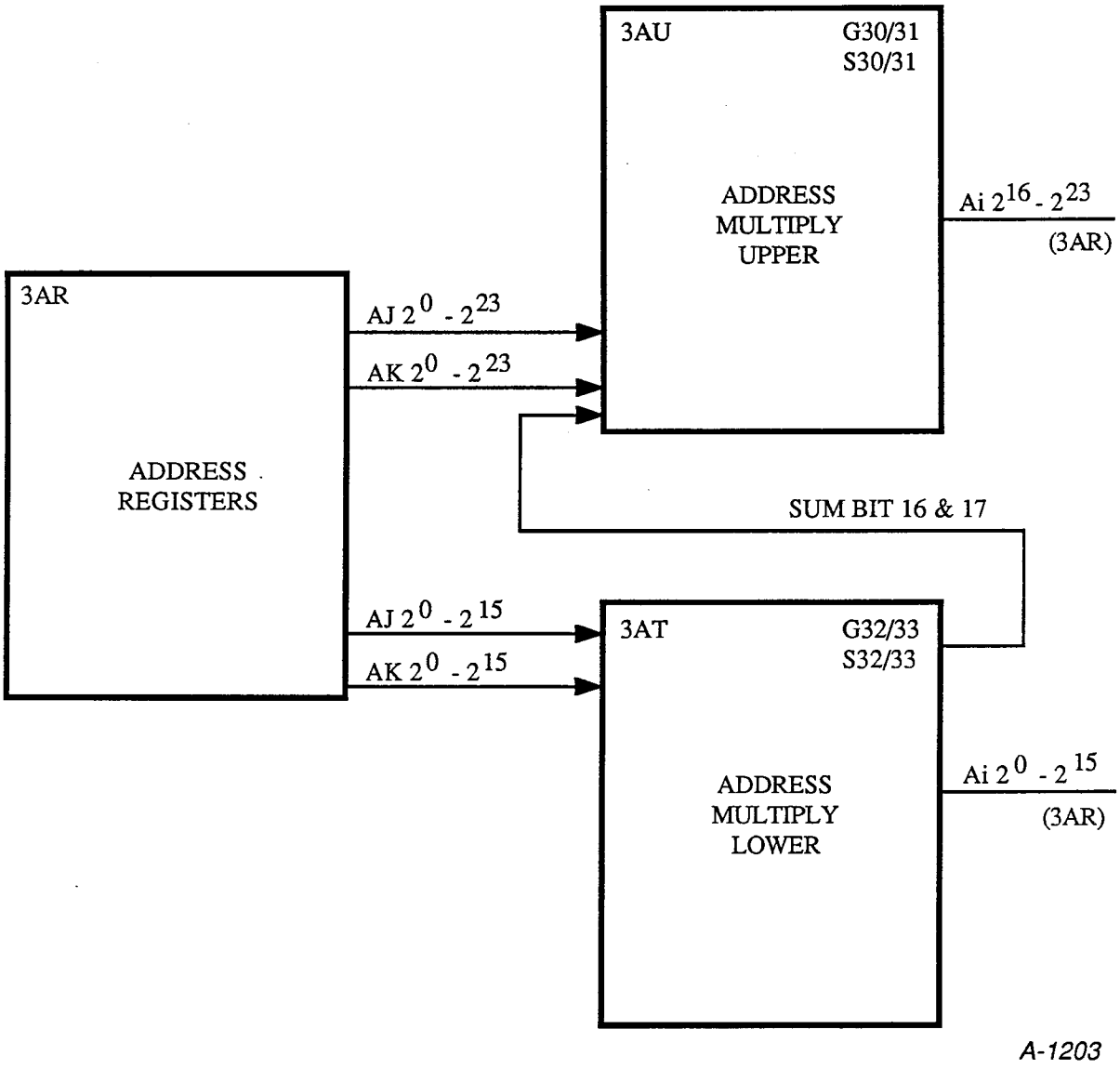
ADDRESS REGISTER FUNCTIONAL UNIT INSTRUCTIONS

ADDRESS MULTIPLY

032IJK - Integer product of AJ times AK to AI

032IJ0 - Transmit AJ to AI

032I0K - Zero fill AI



CRAY X-MP ADDRESS MULTIPLY BLOCK DIAGRAM

032ijk Instruction (A_j) x (A_k) = Ai using the Multiply F.U.

24 bit (A_j) times 24 bit (A_k) equals 24 bit answer in A_i

The answer is the lower 24 bits of the product sum. The upper 24 bits are truncated.

No overflow is detected.

```
                  1 0 1 0 1 0
                  1 0 1 1 1 1
                  ------------
                  1 0 1 0 1 0
                1 0 1 0 1 0
              1 0 1 0 1 0
            1 0 1 0 1 0
          0 0 0 0 0 0
        1 0 1 0 1 0
      ---------------------------   Sum Point
      1 1 1 1 0 1 1 0 1 1 0
```

6 Bit binary Multiply

```
                  1 0 1 0 1 0
                  1 0 1 1 1 1
                  ------------
                  1 0 1 0 1 0
                  0 1 0 1 0
                  1 0 1 0
                  0 1 0
   Truncated      0 0
                  0
      ---------------------------   Sum Point
                  1 1 0 1 1 0
```

6 Bit binary Multiply

The 3 Bit Add:

0		1	0	0		1	0	1		1	
0		0	1	0		0	1	1		1	
0		0	0	1		1	1	0		1	
0		1	1	1		10	10	10		11	
Enable		Sum				Carry/Enable				Carry/Sum	

3 Bit Adds do not employ lookahead on carries.

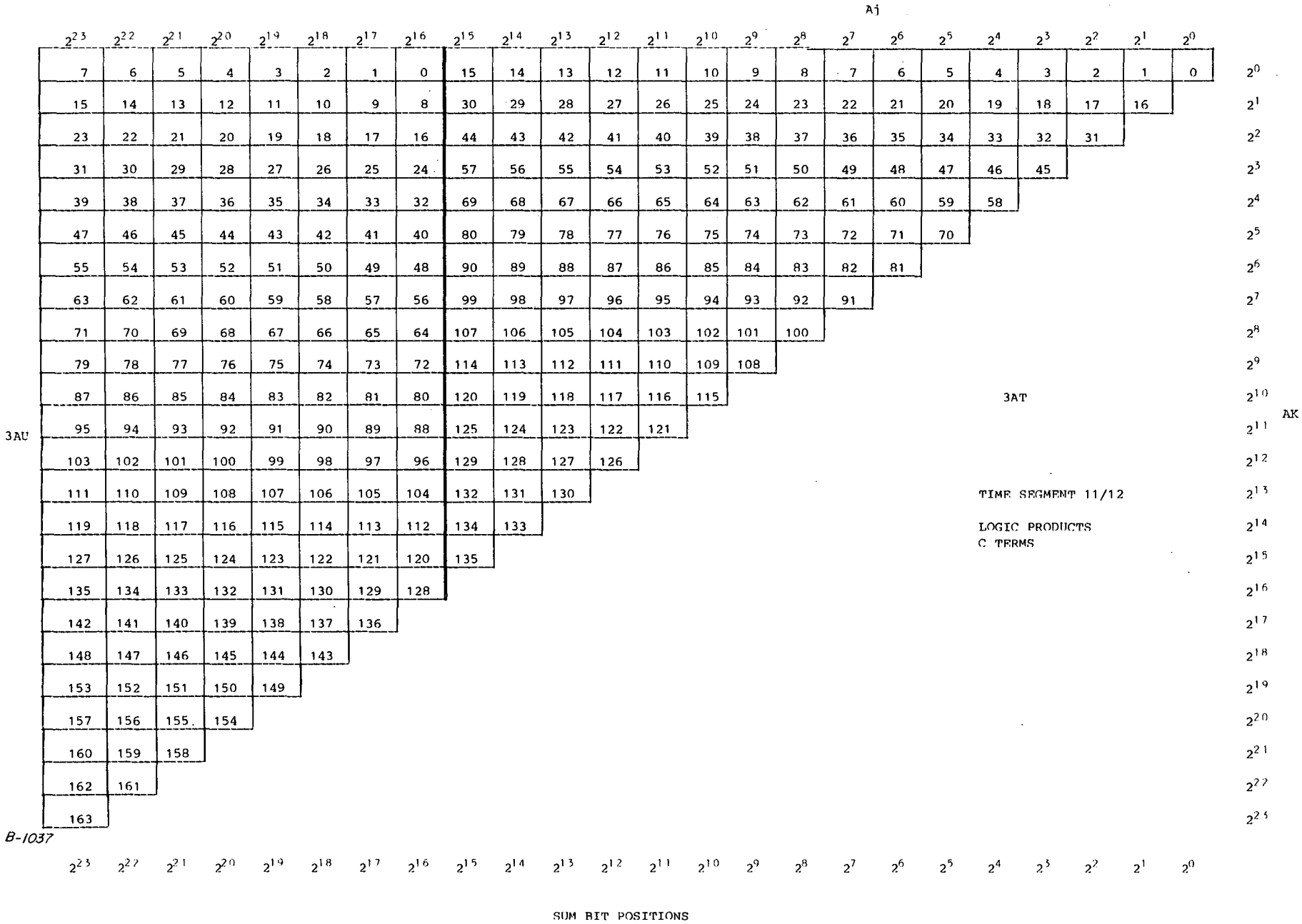
Typical 3 Bit Add Boolean Equations:

Sum

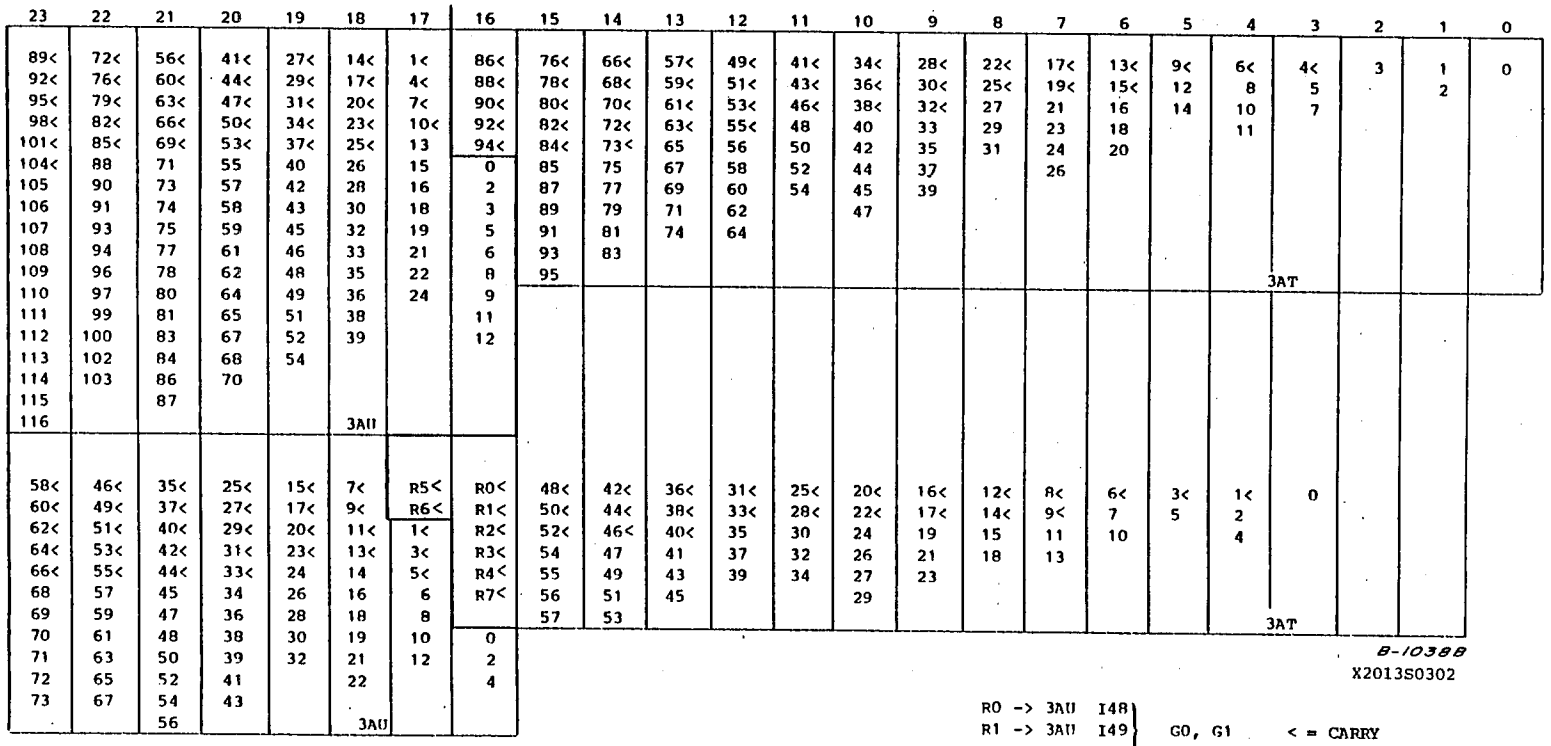
T07 * D03 = Sum (C02, C17, C31); T2

Carry

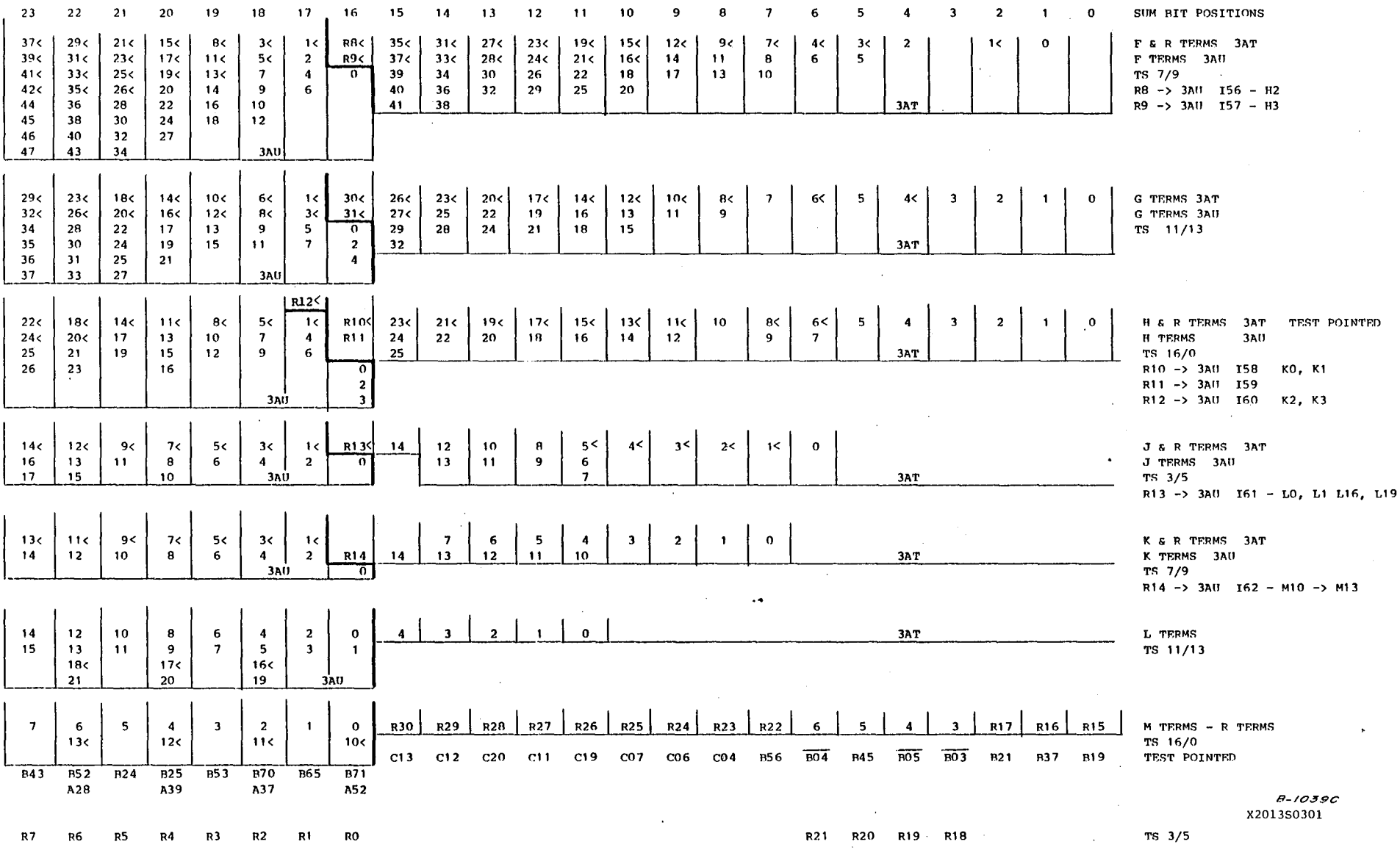
* D04 = Carry (C02, C17, C31); T2



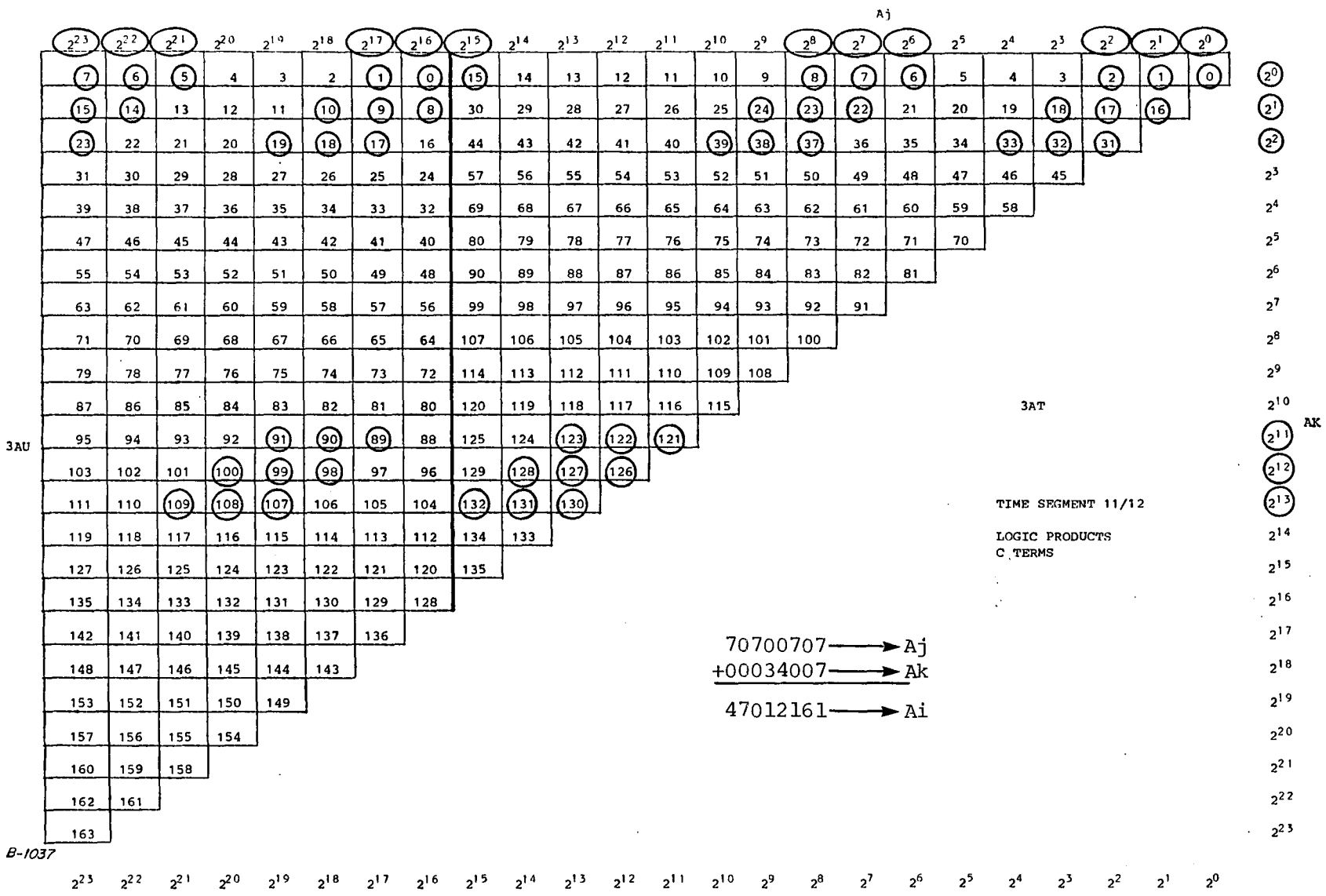
ADDRESS MULTIPLY PYRAMID



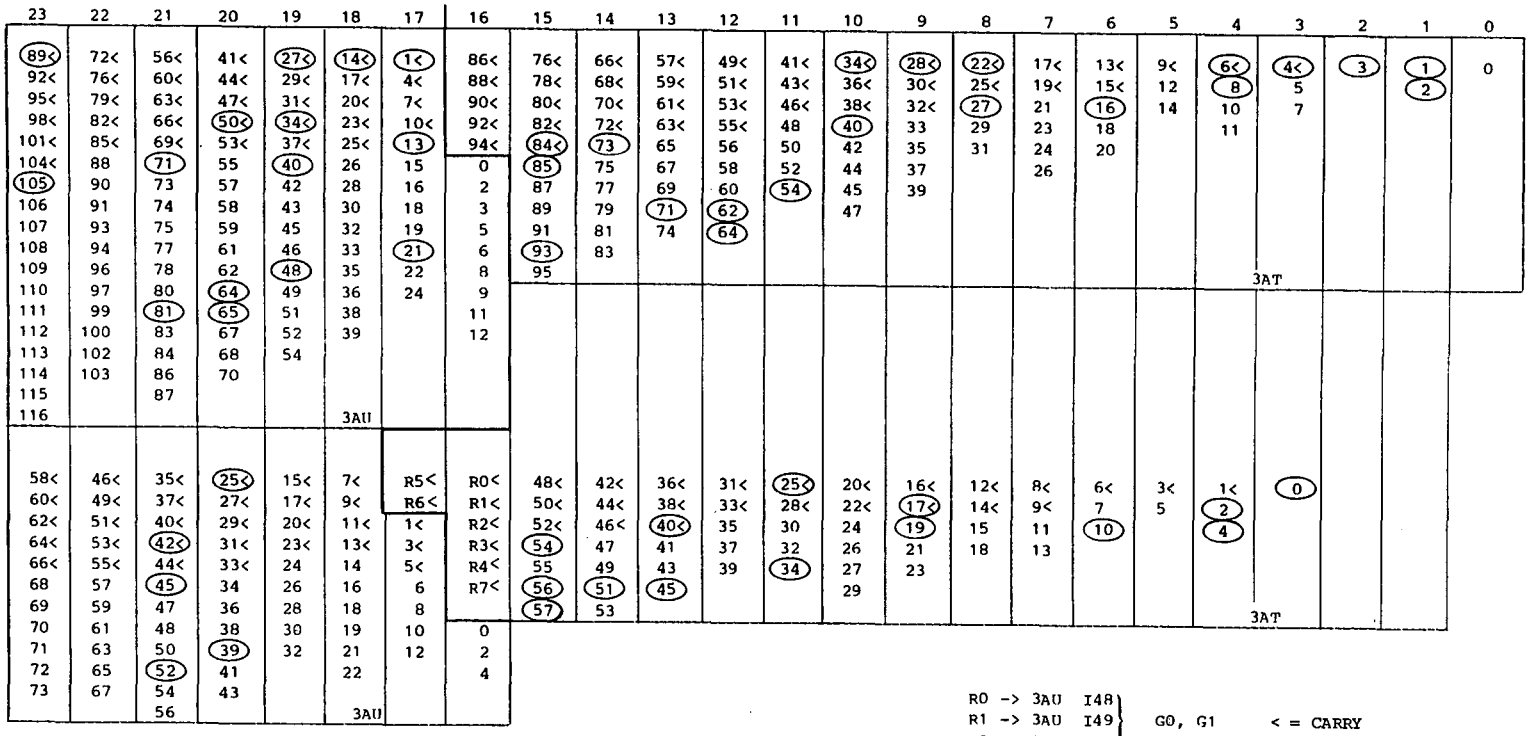
ADDRESS MULTIPLY PYRARMID



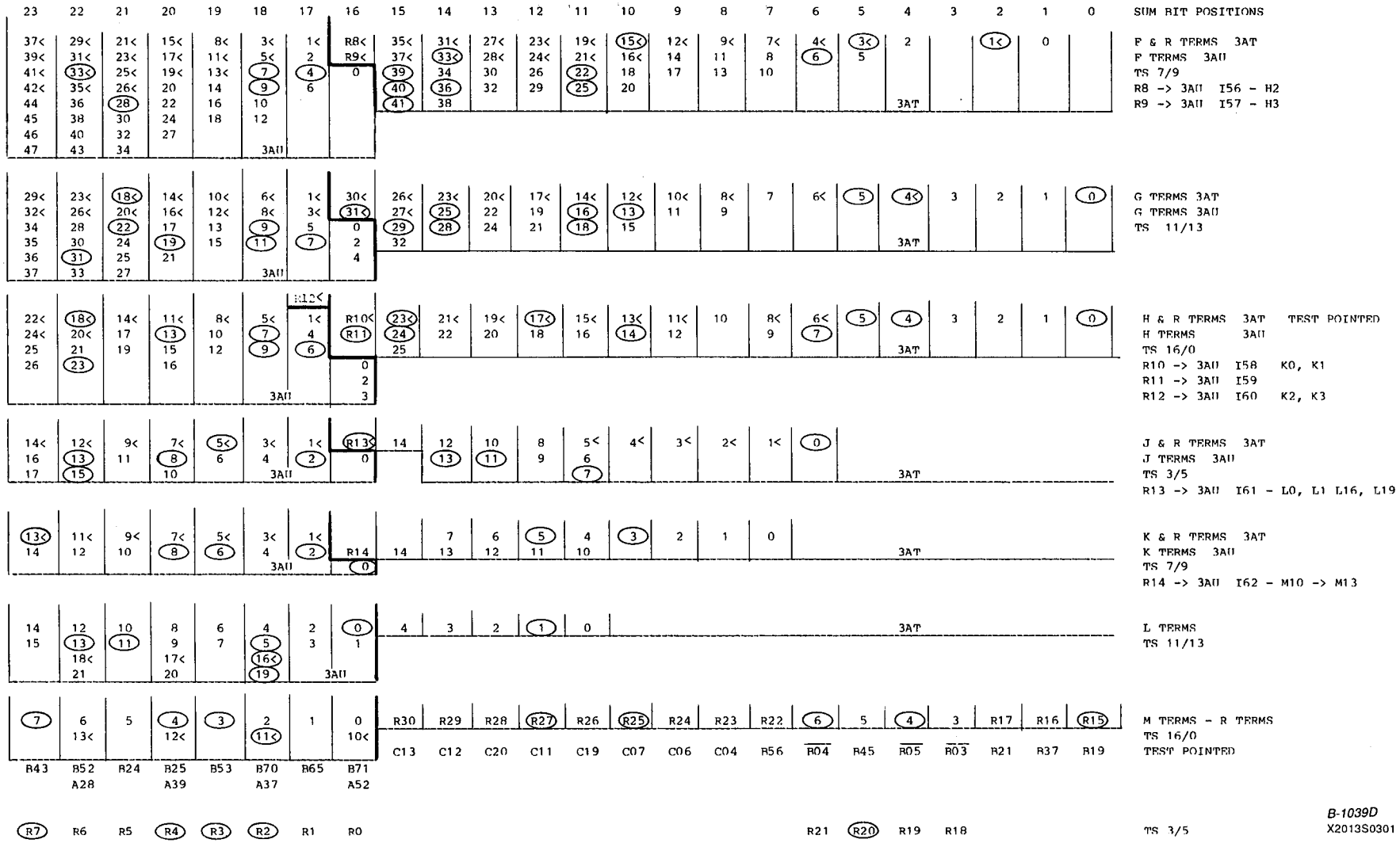
ADDRESS MULTIPLY PYRAMID XMP



ADDRESS MULTIPLY PYRAMID



ADDRESS MULTIPLY PYRAMID



B-1039D
X2013S0301

ADDRESS MULTIPLY PYRAMID

SCALAR FUNCTIONAL UNITS

OBJECTIVES

With the aid of student reference material, upon completion of the course a student should be capable of:

I. Determining the failing module type, module location and chip, given the failing instruction or operation and the data error.

II. Describing the main functions and differences of each of the following modules associated with the scalar functional units:

3GA for the Scalar Add
3GA, 3AT and 3AS for the Scalar Pop Count and Leading Zero Count
3GB for the Scalar Single and Double Shifts

III. Defining with respect to time and use, the address and control signals.

IV. Tracing operands through the Scalar Functional Unit.

V. Diagnostic Application:

3SAR, 3SRA	-	Scalar Add
3SIS, 3ARI	-	Scalar Pop/LZ
3SRS	-	Scalar Shift
3SVC, 3SRL	-	Scalar Logical

SCALAR LOGICAL

044IJK - Logical project of (SJ) and (SK) to SI

044IJ0 - Sign bit of (SJ) to SI

045IJK - Logical product of (SJ) and $(\overline{SK})$ to SI

045IJ0 - (SJ) with sign bit cleared to SI

046IJK - Logical difference of (SJ) and (SK) to SI

046IJ0 - Toggle sign bit of (SJ), enter in SI

047IJK - Logical equivalence of (SJ) and (SK) to SI

047I0K - Transmit ones compliment of (SK) to SI

047IJ0 - Logical equivalence of (SJ) and sign bit to SI

047I00 - Enter ones compliment of sign bit to SI

050IJK - Scalar merge

050IJ0 - Merge of (SI) and sign bit of (SJ) to SI

051IJK - Logical sum of (SJ) and (SK) to SI

051I0K - Transmit (SK) to SI

051IJ0 - Logical sum of (SJ) and sign bit to SI

051I00 - Enter sign bit to SI

SCALAR ADD (Module Involved)

3GA MODULE

The 3GA module receives two 64-bit operands, Sj and Sk, when a 060 or 061 instruction is issued. Twos complement arithmetic is performed on the k operand when a 061 instruction is issued. The operands are summed on the 3GA module. The results are then sent to the 3GR modules when the control term Go Result for the 060, 061 instruction is present. If Go Result is not present, zeros are sent to the 3GR. Because the Scalar Add F.U. shares the same inputs on the 3GR as the Scalar Shift F.U., the 3GA module is also used for the initial sums for the Scalar Pop Count and Leading Zero F.U.

SCALAR REGISTER

FUNCTIONAL UNIT

INSTRUCTIONS

SCALAR ADD/SUBJECT

060iJK - Integer sum of (Sj) and (Sk) to Si

060i0k - Transmit (Sk) to Si

060iJ0 - Integer sum of (2^{63}) and (Sj) to Si

061iJK - Integer difference of (Sj) minus (Sk) to Si

061i0K - Transmit the negative of (Sk) to Si

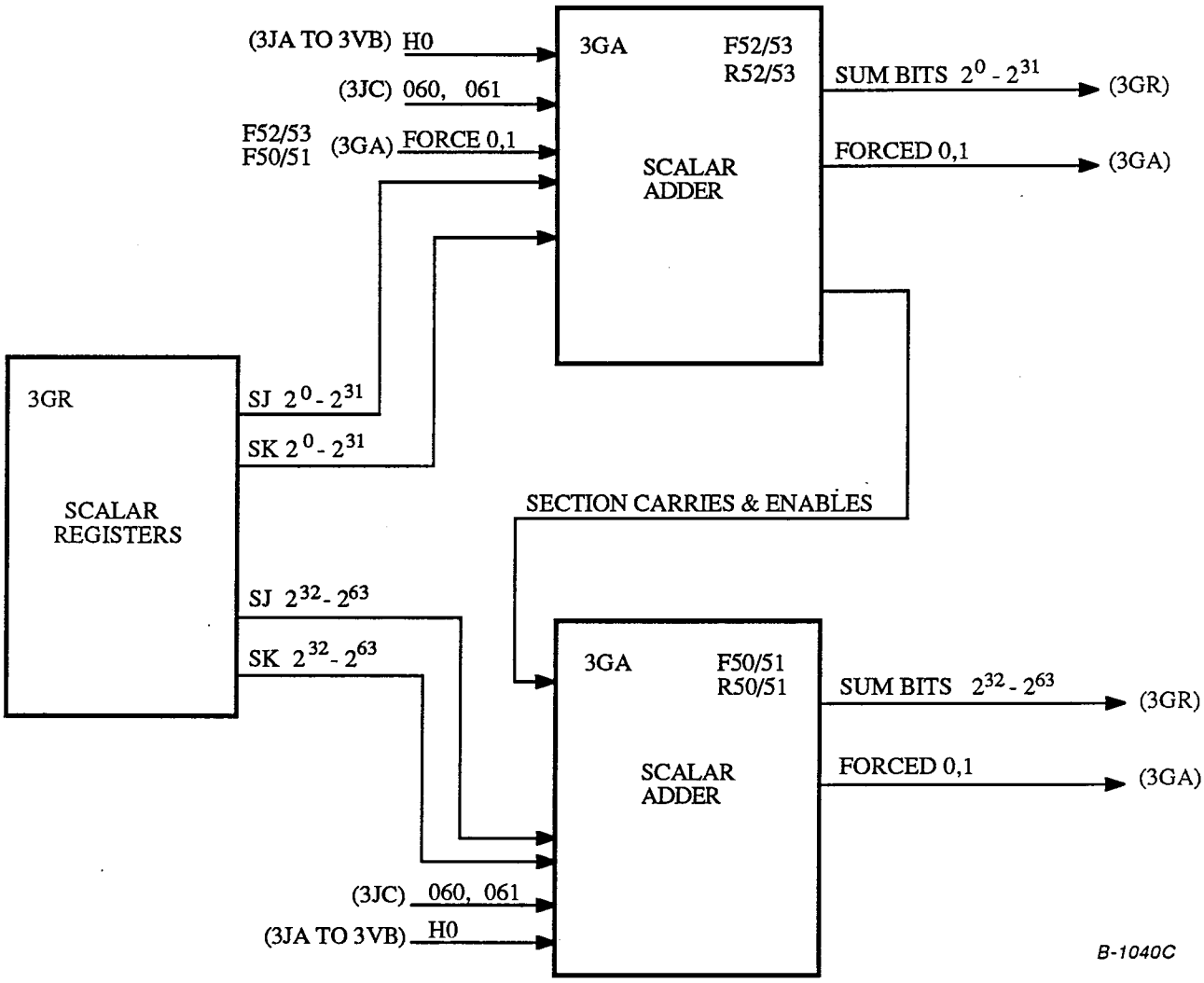
061iJ0 - Integer difference of (Sj) and (2^{63}) to Si

C.A.L. FORMAT

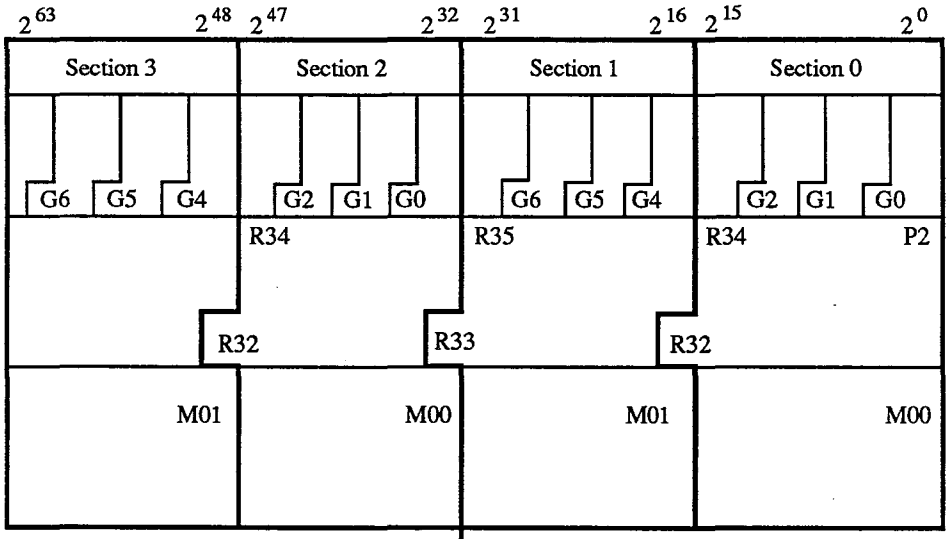
Si Sj + Sk

Si Sj - Sk

Si -Sk



CRAY X-MP SCALAR ADDER



T.P. $\overline{B61}$ Q02 = Subtract

R32 = Section 2 Carry

R34 = Section 2 Enable

I64 = R32 (3GA) F53 Section 0 Carry

I65 = R33 (3GA) F53 Section 1 Carry

I66 = R34 (3GA) F53 Section 0 Enable

I67 = R35 (3GA) F53 Section 1 Enable

M00 = [P2 + I64 + I65] [I66 + I64 + I65] [I67 + I65]

M00 = P2 I66 I67 + I64 I67 + I65

M01 = P2 I66 I67 R34 + I64 I67 R34 + I65 R34 + R32

R32 = Section 0 Carry

R34 = Section 0 Enable

I64 = R57 (3GA) F53 Forced 0

I65 = R57 (3GA) F51 Forced 0

I66 = R58 (3GA) F53 Forced 1

I67 = R58 (3GA) F51 Forced 1

P2 = Mode h0; Subtract

M00 = [P2 + I64 + I65] [I66 + I64 + I65] [I67 + I65]

M00 = P2

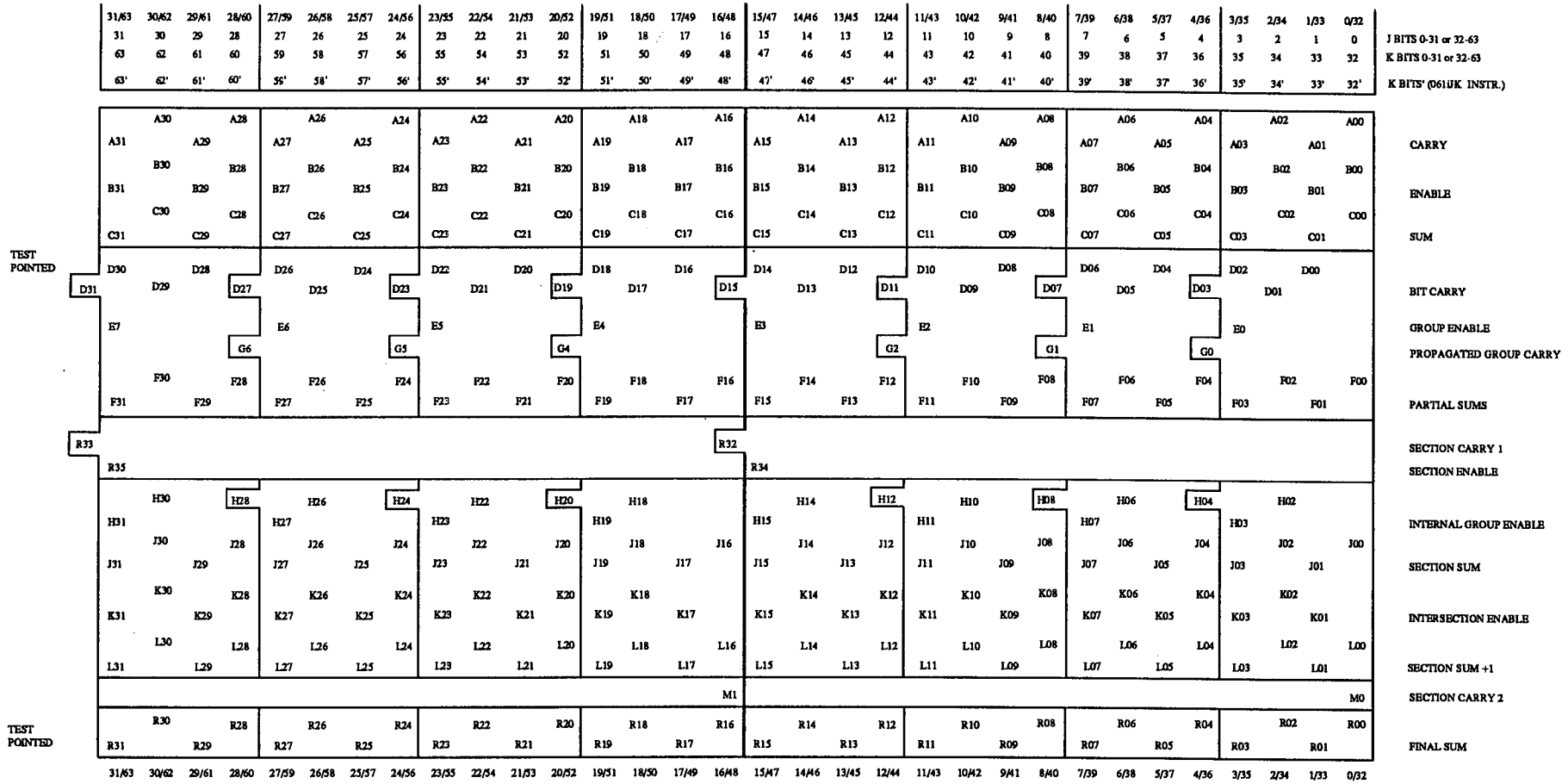
M01 = P2 I66 I67 R34 + I64 I67 R34 + I65 R34 + R32

M01 = P2 R34 + R32

A-1584A

CRAY X-MP PROPAGATED ENABLES AND SECTION CARRIES

SCALAR ADD FUNCTIONAL UNIT



1* Section Carry R32 & R33 → 3GA @ F50/51 as I64 & I65
Section Enable R34 & R35 → 3GA @ F50/51 as I66 & I67
R32-R35 are not outputted on 3GA @ F50/51

Bits 0-31 3GA @ F52/53
Bits 32-63 3GA @ F50/51

2* M0 on 3GA @ F52/53 = Subtract
M1 on 3GA @ F52/53 = Section 0 Carry or Subtract Instruction & Section 0 Enable

M0 on 3GA @ F50/51 = Section 1 Carry or Section 0 Carry with Section 1 Enable or Subtract Bit and Section 0 & 1 Enable
M1 on 3GA @ F50/51 = Section 2 Carry or Section 1 Carry with Section 2 Enable or Section 0 Carry with Section 1 & 2 Enable or Subtract Bit with Section 0, 1, 2 Enables

C-1043H

CRAY X-MP/2 - 3GA SCALAR ADD

SCALAR POP COUNT (Modules Involved)

3GA MODULE

The Scalar Pop Count operand uses the same input as the Scalar Add when a 026 instruction is issued. The 1-bits in the Sj operand are partially totalled using 3 levels of 3-bit adds in the 3GA module. The partial sums are then sent to the 3AT for completion of the pop count.

3AT MODULE

The 3AT completes the summation and sends the results to the 3AR module. If the k-field of the 026 instruction equals zero, all 7 bits of the result are sent to Ai. If the k-field of the instruction equals one, only bit 2^0 is sent to the Ai register for even parity. The 7-bit result corresponds to a 100 octal or all bits in the S register set.

THE SCALAR POPULATION COUNT FUNCTIONAL UNIT

026IJ0 - Population Count of (SJ) to AI

Scalar population count instruction.
Counts the number of 1 bits in the 64 bit J operand, and places this count in AI register (lower 7 bits)

026IJ1 - Population Count Parity of (SJ) to AI

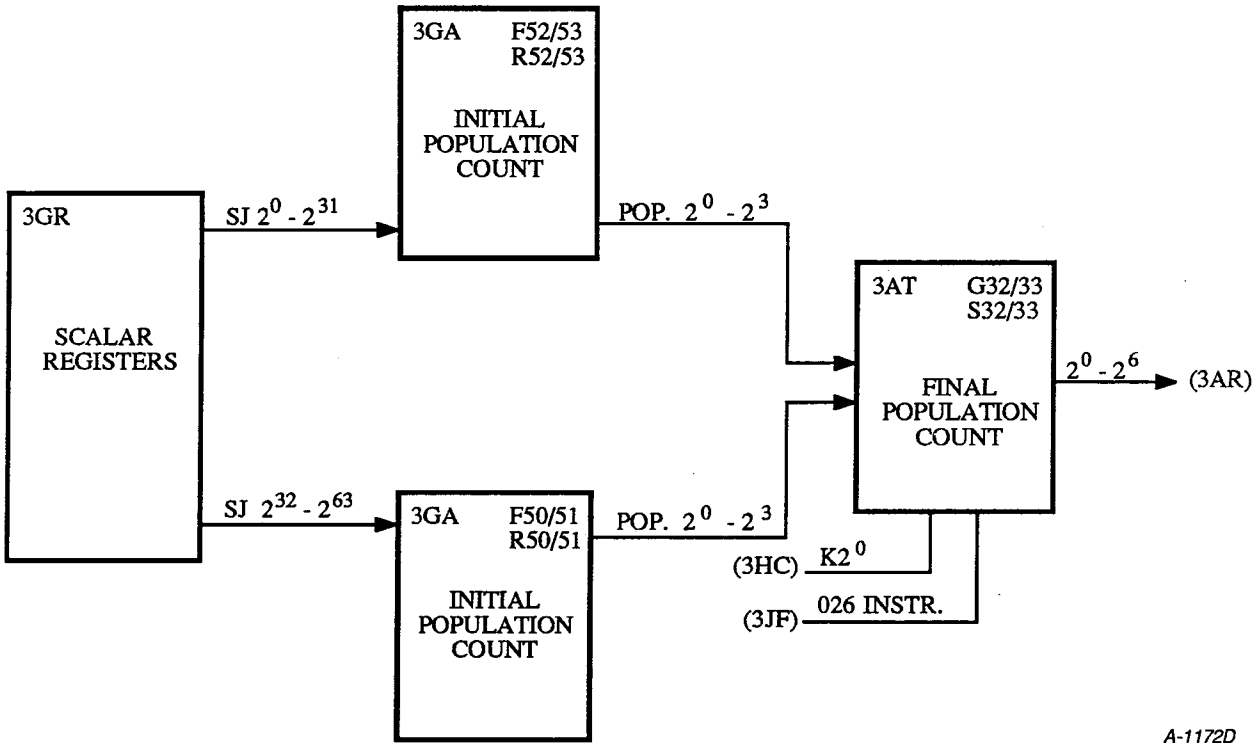
If the K field of this instruction contains a one, only the 2^0 bit of the answer is sent.

C.A.L. FORMAT

AI	PSJ	Scalar Population Count
AI	QSJ	Scalar Parity

DIAGNOSTIC APPLICATION

3SIS



CRAY X-MP/2 SCALAR POPULATION COUNT
BLOCK DIAGRAM

3AT MODULE BOOLEAN EXAMPLE

Boolean R15 = Z0 P0 + H0 P0' ; T2

DATA BIT 2^0 = [POP DATA • 026] + [MULT DATA • $\overline{026}$] ; T2

Boolean R16' = Z1' P0 + H1' P0' + P0 I78 ; T2

R16 = [Z1 + P0'] [H1 + P0] [P0' + I78'] ; T2

DATA BIT 2^1 = [POP DATA + $\overline{026}$] [MULT DATA + 026] [$\overline{026}$ + $\overline{\text{K BIT } 2^0}$] ; T2

Boolean R17' = Z2' P0 + H2' P0' + P0 I78 ; T2

R17 = [Z2 + P0'] [H2 + P0] [P0' + I78'] ; T2

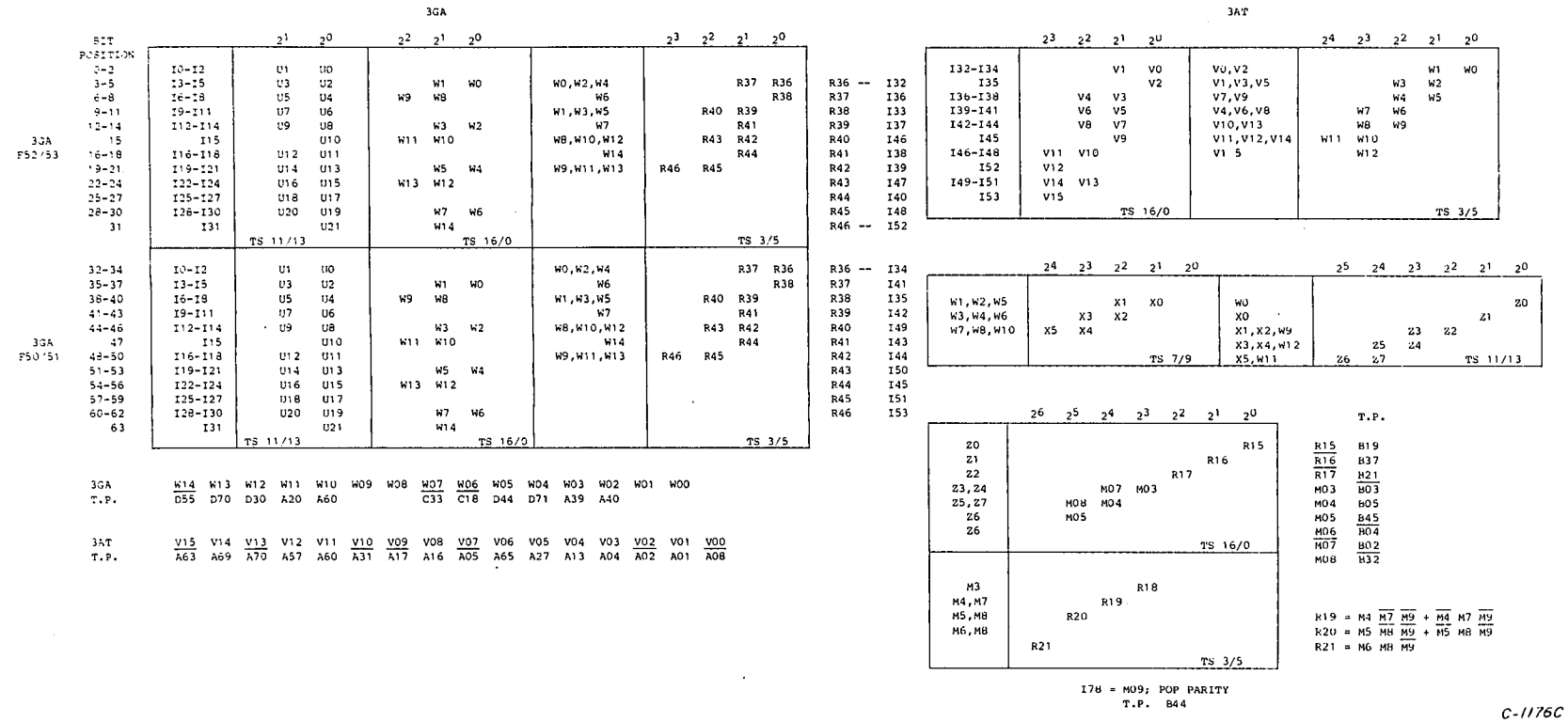
DATA BIT 2^2 = [POP DATA + $\overline{026}$] [MULT DATA + 026] [$\overline{026}$ + $\overline{\text{K BIT } 2^0}$] ; T2

Boolean R18 = M3 M9'

DATA BIT 2^3 = [POP OR MULT DATA • $\overline{\text{K BIT } 2^0}$]

A-1601

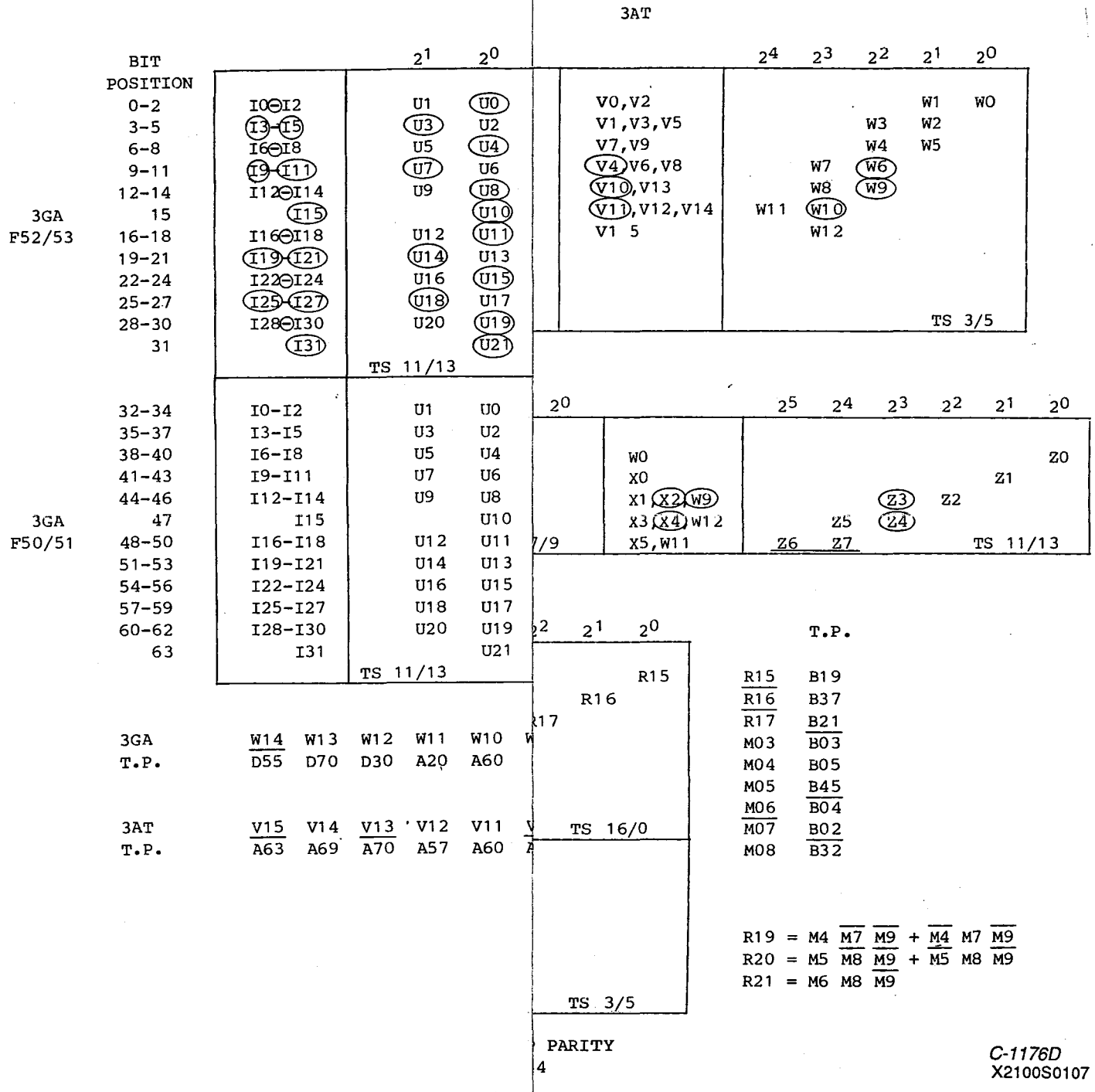
SCALAR POP COUNT



C-1176C

X2100S0107

SCALAR POP COUNT FUNCTION UNIT


3AT
BIT POSITION
3GA F52/53
0-2
3-5
6-8
9-11
12-14
15
16-18
19-21
22-24
25-27
28-30
31
I0⊖I2
I3-I5
I6⊖I8
I9-I11
I12⊖I14
I15
I16⊖I18
I19-I21
I22⊖I24
I25-I27
I28⊖I30
I31
U1 U0
U3 U2
U5 U4
U7 U6
U9 U8
U10
U12 U11
U14 U13
U16 U15
U18 U17
U20 U19
U21
TS 11/13
V0,V2
V1,V3,V5
V7,V9
V4,V6,V8
V10,V13
V11,V12,V14
V1 5
W1 W0
W3 W2
W4 W5
W7 W6
W8 W9
W11 W10
W12
TS 3/5
3GA F50/51
32-34
35-37
38-40
41-43
44-46
47
48-50
51-53
54-56
57-59
60-62
63
I0-I2
I3-I5
I6-I8
I9-I11
I12-I14
I15
I16-I18
I19-I21
I22-I24
I25-I27
I28-I30
I31
U1 U0
U3 U2
U5 U4
U7 U6
U9 U8
U10
U12 U11
U14 U13
U16 U15
U18 U17
U20 U19
U21
TS 11/13
W0
X0
X1,X2,W9
X3,X4,W12
X5,W11
Z0
Z1
Z3 Z2
Z5 Z4
Z6 Z7
TS 11/13
3GA T.P.
W14 W13 W12 W11 W10
D55 D70 D30 A20 A60
3AT T.P.
V15 V14 V13 V12 V11
A63 A69 A70 A57 A60
R15
R16
R17
TS 16/0
TS 3/5
T.P.
R15 B19
R16 B37
R17 B21
M03 B03
M04 B05
M05 B45
M06 B04
M07 B02
M08 B32
R19 = M4 M7 M9 + M4 M7 M9
R20 = M5 M8 M9 + M5 M8 M9
R21 = M6 M8 M9
PARITY
C-1176D
X2100S0107

SCALAR LEADING ZERO COUNT (Modules Involved)

3GA Module

The 3GA module inputs the Sj operand when a 027 instruction is issued. The Scalar leading zeroes Sj operand uses the same input path as the Scalar Add. The Sj operand is divided into 4-bit groups. The number of leading zeroes are counted within each group. The 3GA then sends its group counts to the 3AT and 3AS modules. The 3GA at F50/51 counts the leading zeroes for bit locations 2^0 - 2^{31}. The 3GA at F52/53 counts the leading zeroes for bit locations 2^{32} - 2^{63}.

3AT Module

If all groups from the 3GAs are zeroes, the 3AT module will output bit 2^6 to an A register. This register indicates 100 octal for a total of 64 decimal leading zeroes.

3AS Module

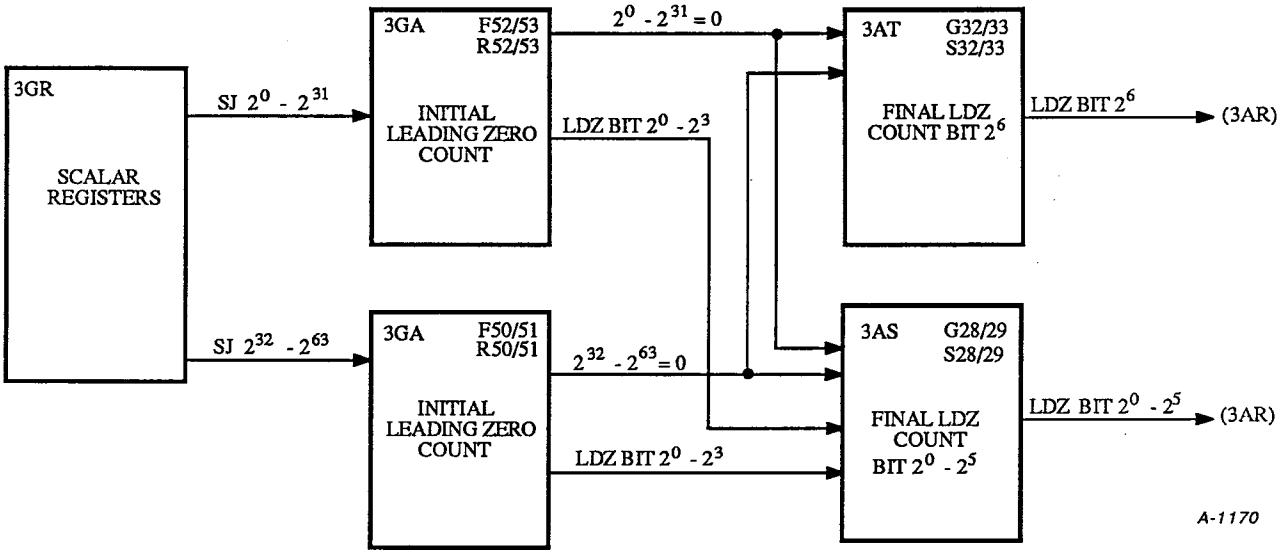
The 3AS will output bits 2^0 - 2^5 to an A register whenever the leading zero count is less than 100 octal. The 3AS will total the number of zeroes from the most significant bits which were sent to the 3AS by the 3GAs. The most significant bits are handled on the 3GA at F50/51.

XV203S05A

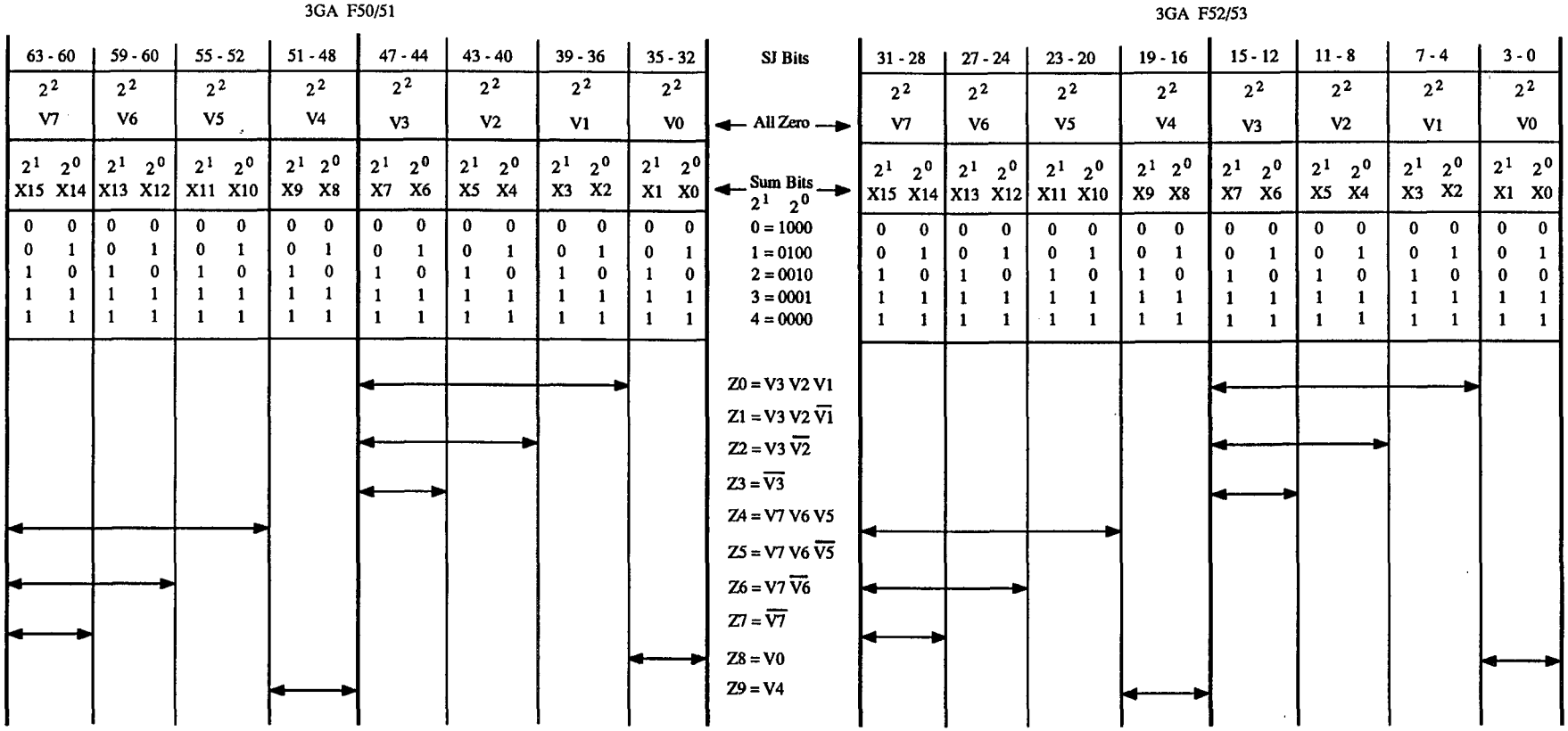
027IJ0	Scalar Leading Zero Count Instruction. Counts to the first 1 bit in the 64 bit J operand and places the count in the AI register (lower 7 bits)

C.A.L. FORMAT

AI	ZSJ	Scalar Leading Zero



CRAY X-MP SCALAR LEADING ZERO



Boolean X15' = #A31 + 0 A29 + A30 T2
X15 = [A31' • A30'] ; T2

X15 = [Data Bit N = 2^2] [Data Bit N + 2^1 ; T2]
Count = 2 or 3

Boolean X14' = #A31 + A30' A29 ; T2
X14 = [A31'] [A30 + A29'] ; T2

X14 = [Data Bit N + 2^2] [Data Bit N + 2^1 + Data bit N + 2^0] ; T2
Count = 1 or 3

Example of one four bit group

Boolean Terms	2^{63} A31	2^{62} A30	2^{61} A29	2^{60} A28	2^2 V7	2^1 X15	2^0 X14
No Leading Zero	1	0	0	0	0	0	0
1 Leading Zero	0	1	0	0	0	0	1
2 Leading Zero	0	0	1	0	0	1	0
3 Leadiong Zero	0	0	0	1	0	1	1
4 Leading Zero	0	0	0	0	1	1	1

B-1327E

CRAY X-MP 3GA SCALAR LEADING ZERO

3GA F50/51 BITS 2^{63} - 2^{48}

LEADING ZERO COUNT	01	02	03	04	05	06	07	10	11	12	13	14	15	16	17	20	
SUM BIT 0	x		x		x		x		x		x		x		x		R52
SUM BIT 1		x	x			x	x			x	x			x	x		R53
SUM BIT 2				x	x	x	x					x	x	x	x		R54
SUM BIT 3								x	x	x	x	x	x	x	x		R55
ALL ZERO																x	R56

3GA F50/51 BITS 2^{47} - 2^{32}

21	22	23	24	25	26	27	30	31	32	33	34	35	36	37	40	
x		x		x		x		x		x		x		x		R47
	x	x			x	x			x	x			x	x		R48
			x	x	x	x					x	x	x	x		R49
							x	x	x	x	x	x	x	x		R50
															x	R51

3GA F52/53 BITS 2^{31} - 2^{16}

LEADING ZERO COUNT	41	42	43	44	45	46	47	50	51	52	53	54	55	56	57	60	
SUM BIT 0	x		x		x		x		x		x		x	x	x		R52
SUM BIT 1		x	x			x	x			x	x			x	x		R53
SUM BIT 2				x	x	x	x					x	x		x		R54
SUM BIT 3								x	x	x	x	x	x	x	x		R55
ALL ZERO																x	R56

3GA F52/53 BITS 2^{15} - 2^{0}

61	62	63	64	65	66	67	70	71	72	73	74	75	76	77	100	
x		x		x		x		x		x		x		x		R47
	x	x			x	x			x	x			x	x		R48
			x	x	x	x					x	x	x	x		R49
							x	x	x	x	x	x	x	x		R50
															x	R51

3GA F50/51 TO 3AS G28/29

R52 - I48
R53 - I49
R54 - I50
R55 - I51
R56 - I52
R56 - I57 ON 3AT G32/33

R47 - I53
R48 - I54
R49 - I55
R50 - I56
R51 - I57
R51 - I56 ON 3AT G32/33

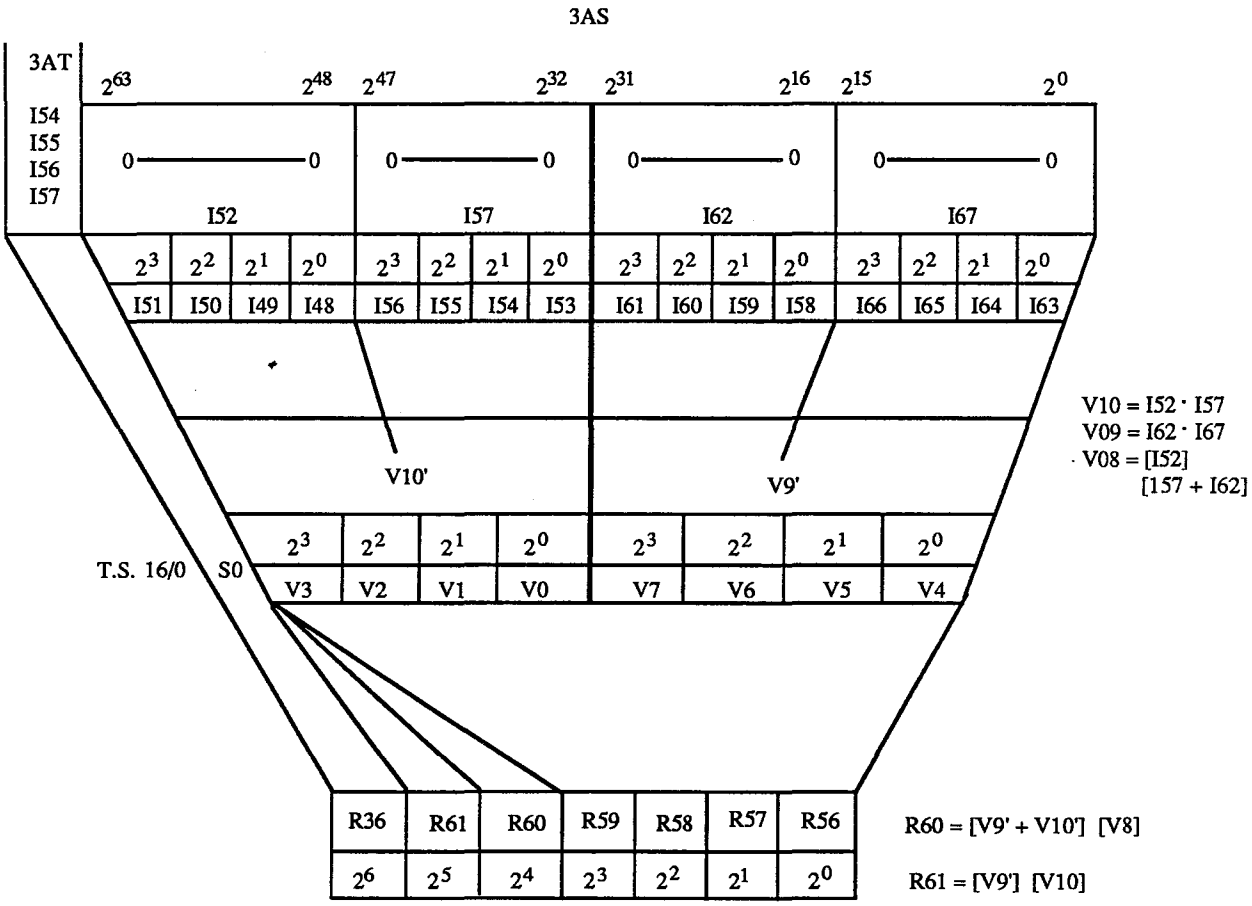
3GA F52/53 TO 3AS G28/29

R52 - I58
R53 - I59
R54 - I60
R55 - I61
R56 - I62
R56 - I55 ON 3AT G32/33

R47 - I63
R48 - I64
R49 - I65
R50 - I66
R51 - I67
R51 - I54 ON 3AT G32/33

A-5407

CRAY X-MP 3GA SCALAR LEADING ZERO



CONTROL

I52 - Take I terms I48 - I51 if you do not have I52.
If you do have I52 take I term I56 - I53 and make V0 - V3.

I62 - Take I terms I58 - I61 if you do not have I62.
If you do have I62 take I term I63 - I66 and make V4-V7.

V10' - If you do not have V10 make R56 - R59 from V0 - V3.
V10' count < 40

V9' - If you do have V9· V10 make R56 - R59 from V4 - V7.
V9' V10 count > 40 < 100

R60 - Count is $\geqq 20_8$ - 37_8, $\geqq 60_8$ - 77_8

R61 - Count is 40_8 - 57_8 or 60_8 - 77_8

R36 - Count is 100_8

B-1602E

CRAY X-MP/2 3AS AND 3AT MODULE DIAGRAM

3AT MODULE BOOLEAN EXAMPLE

```
Boolean      S00' = P1  I54'  I55'  I56'  I57'  #  #  #  #  #  #
             S00' = I57' + I56' + I55' + I54' + P1
```

S00 = [I57] [I56] [I55] [I54] [P1']

I57 = R56 3GA @ F51 BITS 48 - 63 = 0
I56 = R51 3GA @ F51 BITS 47 - 32 = 0
I55 = R56 3GA @ F53 BITS 31 - 16 = 0
I54 = R51 3GA @ F53 BITS 15 - 0 = 0

3AS MODULE BOOLEAN EXAMPLE

I52 = R56 3GA @ F51 BITS 48 - 63 = 0
I57 = R51 3GA @ F51 BITS 32 - 47 = 0
I62 = R56 3GA @ F53 BITS 16 - 31 = 0
I67 = R51 3GA @ F53 BITS 0 - 15 = 0

```
Boolean      V09' = I67' N2' + N2' I62' : T2
             V09 = [I67 + N2 ] [N2  + I62] : T2
```

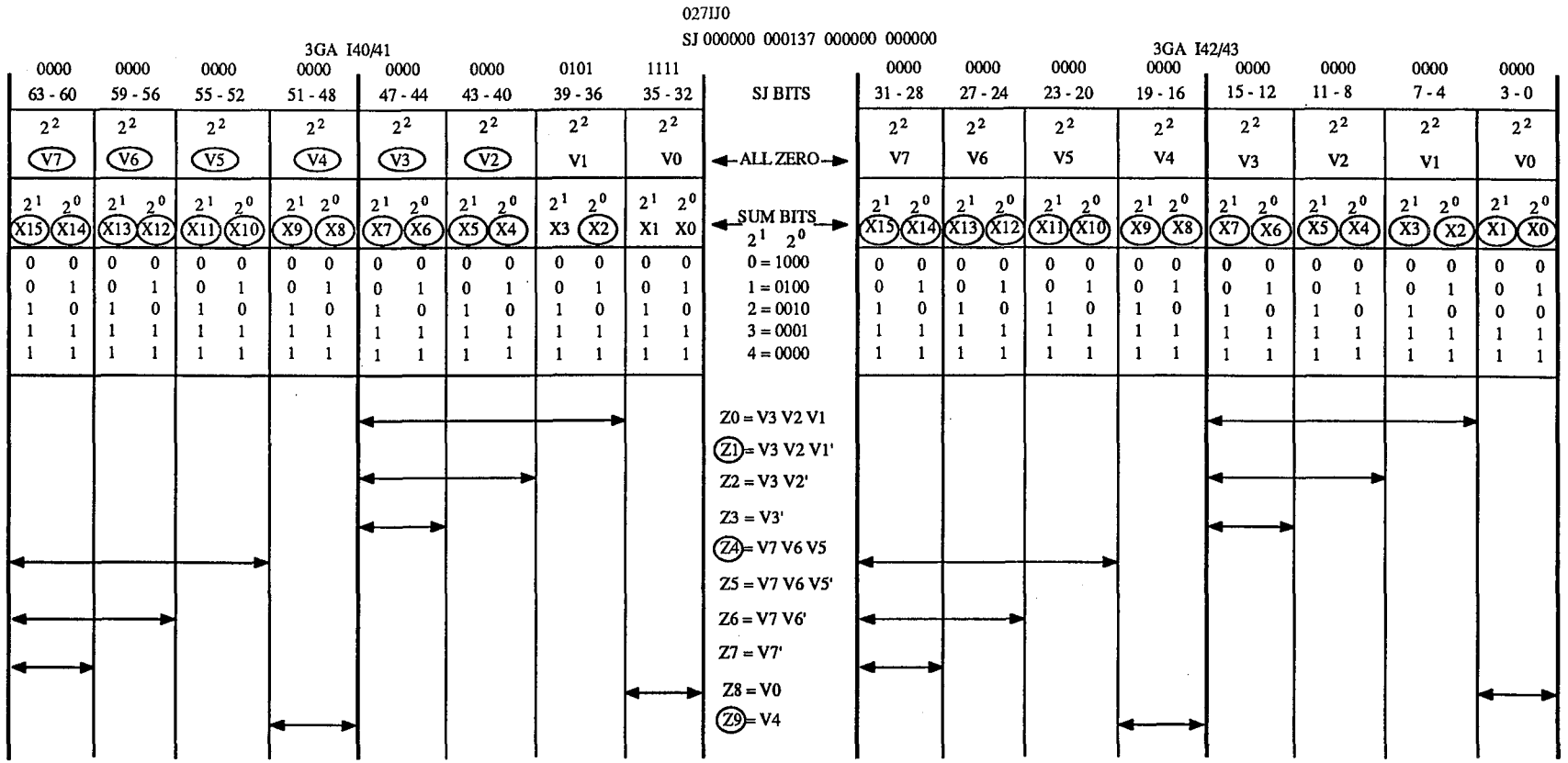
V09 = [BITS 2^0 - 2^{15} = 0's] [BITS 2^{31} - 2^{16} = 0's]

```
Boolean      V10 = I52 I57 + N2
```

V10 = [BITS 2^{48} - 2^{63} = 0] [BITS 2^{32} - 2^{47} = 0]

A-1600

SCALAR LEADING ZERO



Boolean X15' = #A31 + 0 A29 + A30 ; T2
X15 = (A31' · A30') ; T2

$X15 = \overline{(DATA\ BIT\ N + 2^2)}\ \overline{(DATA\ BIT\ N + 2^1}$; T2)
COUNT = 2 OR 3

Boolean X14' = #A31 + A30' A29 ; T2
X14 = (A31') (A30 + A29') ; T2

$X14 = \overline{(DATA\ BIT\ N\ ^2 2}\ )\ (DATA\ BIT\ N + 2^1 + \overline{DATA\ BIT\ N + 2^0})$; T2
COUNT = 1 OR 3

EXAMPLE OF ONE FOUR BIT GROUP

BOOLEAN TERMS	2^{63} A31	2^{62} A30	2^{61} A29	2^{60} A28	2^2 V7	2^1 X15	2^0 X14
NO LEADING ZERO	1	0	0	0	0	0	0
1 LEADING ZERO	0	1	0	0	0	0	1
2 LEADING ZERO	0	0	1	0	0	1	0
3 LEADING ZERO	0	0	0	1	0	1	1
4 LEADING ZERO	0	0	0	0	1	0	0

A-5943

3GA SCALAR LEADING ZERO

3GA F50/51 BITS 2^{63} - 2^{48}

LEADING ZERO COUNT	01	02	03	04	05	06	07	10	11	12	13	14	15	16	17	20	
SUM BIT 0	X		X		X		X		X		X		X		X		R52
SUM BIT 1		X	X			X	X			X	X			X	X		R53
SUM BIT 2				X	X	X	X					X	X	X	X		R54
SUM BIT 3								X	X	X	X	X	X	X	X		R55
ALL ZERO																X	(R56)

3GA F50/51 BITS 2^{47} - 2^{32}

	21	22	23	24	25	26	27	30	31	32	33	34	35	36	37	40	
SUM BIT 0	X		X		X		X		X		X		X		X		(R47)
SUM BIT 1		X	X			X	X			X	X			X	X		R48
SUM BIT 2				X	X	X	X					X	X	X	X		R49
SUM BIT 3								X	X	X	X	X	X	X	X		(R50)
ALL ZERO																X	R51

3GA F52/53 BITS 2^{31} - 2^{16}

LEADING ZERO COUNT	41	42	43	44	45	46	47	50	51	52	53	54	55	56	57	60	
SUM BIT 0	X		X		X		X		X		X		X	X	X		R52
SUM BIT 1		X	X			X	X			X	X			X	X		R53
SUM BIT 2				X	X	X	X					X	X		X		R54
SUM BIT 3								X	X	X	X	X	X	X	X		R55
ALL ZERO																X	(R56)

3GA F52/53 BITS 2^{15} - 2^{0}

	61	62	63	64	65	66	67	70	71	72	73	74	75	76	77	100	
SUM BIT 0	X		X		X		X		X		X		X		X		R47
SUM BIT 1		X	X			X	X			X	X			X	X		R48
SUM BIT 2				X	X	X	X					X	X	X	X		R49
SUM BIT 3								X	X	X	X	X	X	X	X		R50
ALL ZERO																X	(R51)

3GA F50/51 TO 3AS G28/29

R52 - I48
R53 - I49
R54 - I50
R55 - I51
R56 - I52
R56 - I57 ON 3AT G32/33

R47 - I53
R48 - I54
R49 - I55
R50 - I56
R51 - I57
R51 - I56 ON 3AT G32/33

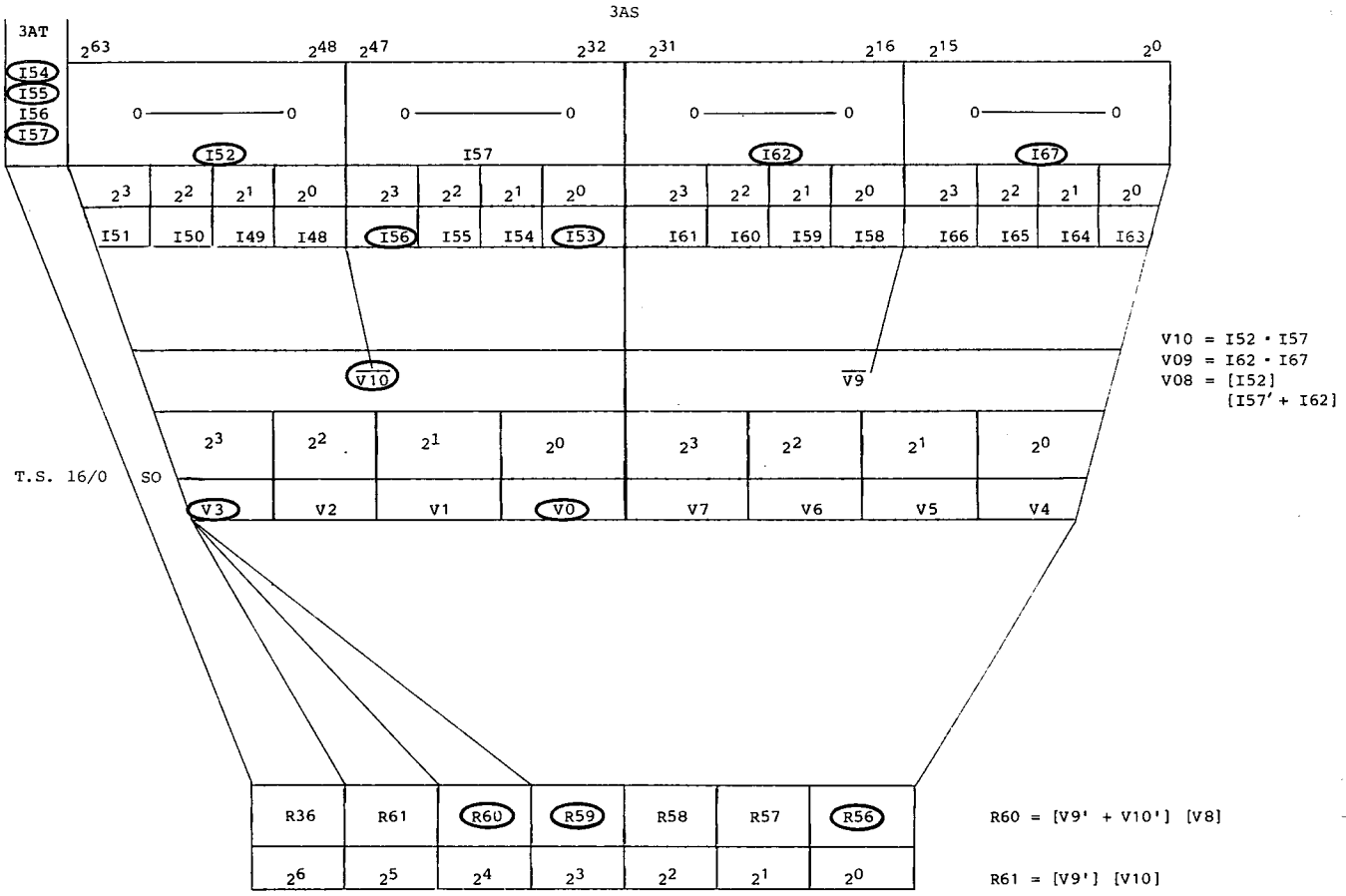
3GA F52/53 TO 3AS G28/29

R52 - I58
R53 - I59
R54 - I60
R55 - I61
R56 - I62
R56 - I55 ON 3AT G32/33

R47 - I63
R48 - I64
R49 - I65
R50 - I66
R51 - I67
R51 - I54 ON 3AT G32/33

X2001C0302

3GA SCALAR LEADING ZERO



CONTROL

I52 - TAKE I TERMS I48 - I51 IF YOU DO NOT HAVE I52
IF YOU DO HAVE I52 TAKE I TERM I56 - I53 AND MAKE V0-V3.

I62 - TAKE I TERMS I58 - I61 IF YOU DO NOT HAVE I62
IF YOU DO HAVE I62 TAKE I TERM I63 - I66 AND MAKE V4-V7.

$\overline{V10}$ - IF YOU DO NOT HAVE V10 MAKE R56 - R59 FROM V0-V3. $\overline{V10}$ COUNT < 40

$\overline{V9}$ - IF YOU DO NOT HAVE V9 • V10 MAKE R56 - R59 FROM V4-V7. $\overline{V9}$ V10 COUNT > 40 < 100

R60 - COUNT IS $\geqq 20_8$-37_8, $\geqq 60_8$-77_8

R61 - COUNT IS 40_8-57_8 or 60_8-77_8

R36 - COUNT IS 100_8

B-1602D
X4815S0710

X-MP 3AS AND 3AT MODULE DIAGRAM

EXAMPLE NO. 1
HTV-0242

SCALAR SINGLE/DOUBLE SHIFT (Module Involved)

3GB MODULE

The 3GB modules perform the Scalar Shifts for both the single and double shift operations. The 052ijk - 055ijk instructions cause the single right or left operations to be performed. The 056ijk and 057ijk cause double right or left operations to be performed.

All Scalar Shifts are left end off and zero filled. To perform a single right shift, the twos complement of the number of places to right shift must be supplied in the jk field of the instruction.

The Scalar double shift count is supplied by the Ak field in the instruction. Any bits set in Ak from 2^7 - 2^{23} will zero the results. For a right double shift, the twos complement of the number of places to right shift is calculated on the 3GB module.

The 3GB at F48/49 handles all the odd number bits of Sj or Si. The 3GB at F54/55 handles all the even bits of Sj or Si. There is no cross-over of bits between the 3GB modules. A shift of one place on the module is equivalent to a shift of two places in the register.

The two 3GB modules output their results to the 3GR modules where the final result is ORed together.

The shift count is in the JK field of the

052 IJK
Result to S0

054 IJK
Result to S_I

Instructions.

C.A.L. FORMAT

SI	SI < JK	Scalar single left shift to SI
S0	SI < JK	Scalar single left shift to S0

DIAGNOSTIC APPLICATION

3SRS

3SRR

THE SINGLE RIGHT SHIFT

The shift count is

100_8 - JK field of the

053 IJK
result to S0

055 IJK
result to S_I

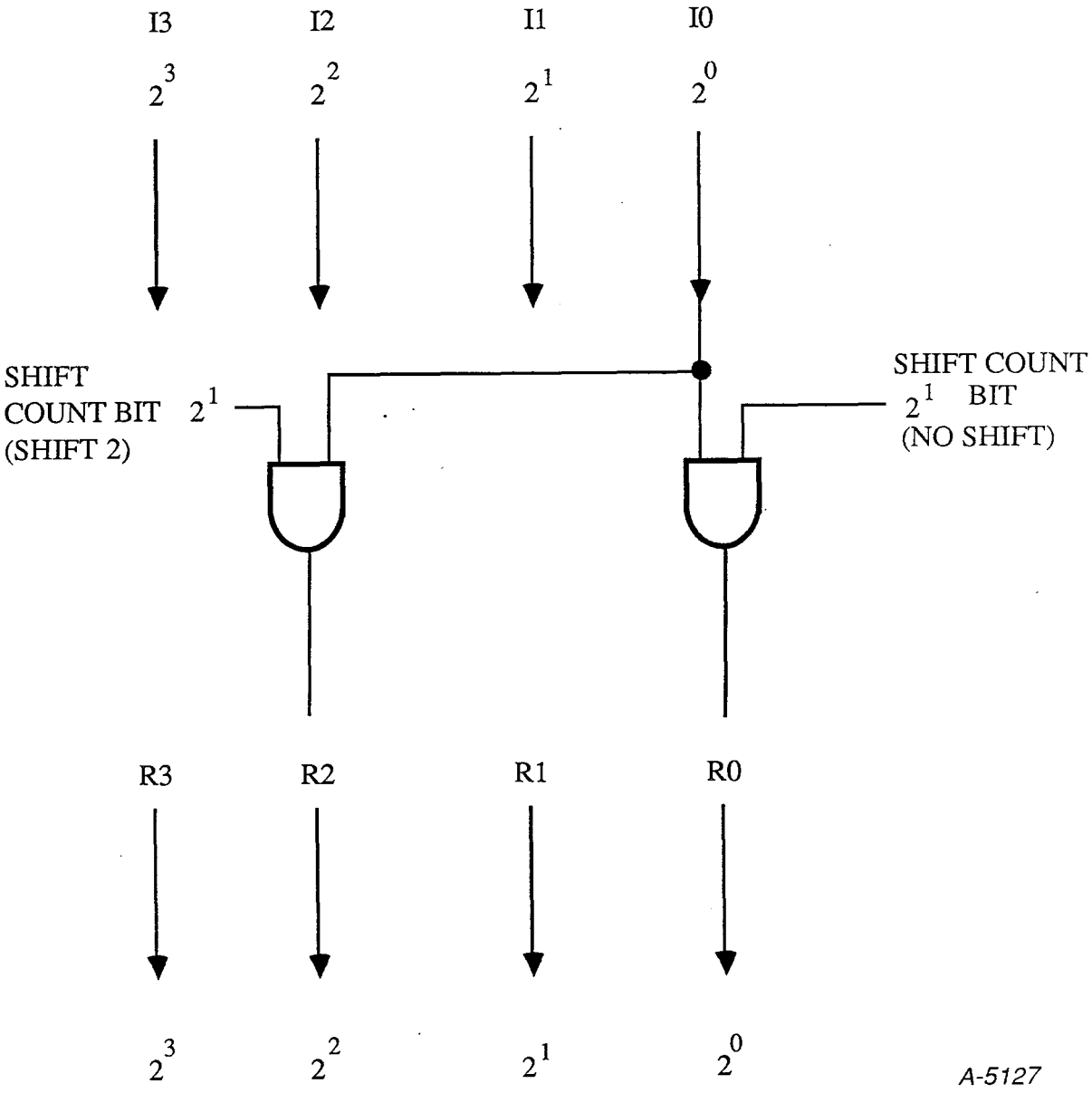
Instructions.

C.A.L. FORMAT

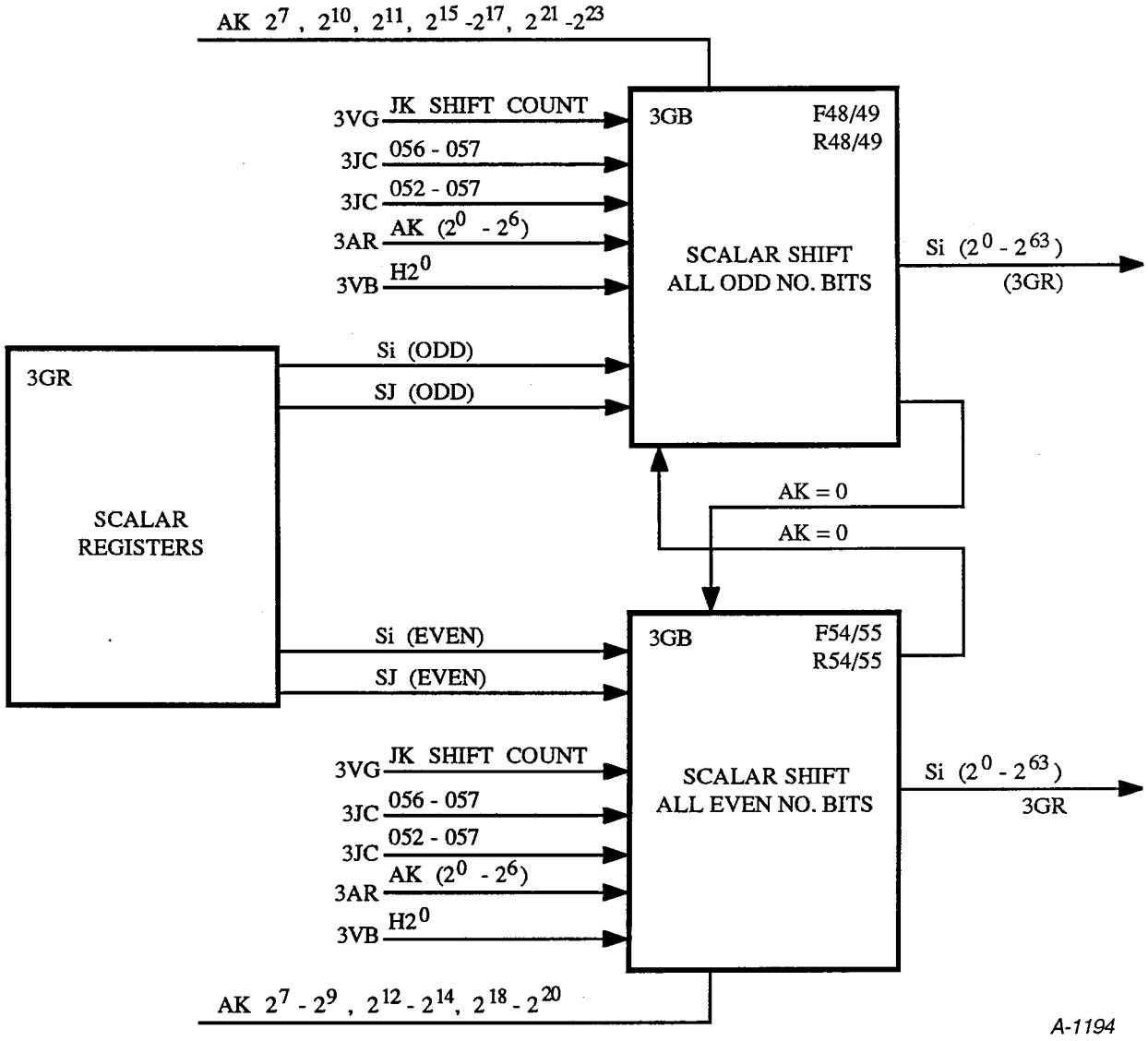
S0	SI > JK	Scalar right shift to S0
SI	SI > JK	Scalar right shift to SI

BASIC SHIFT

OPERATION

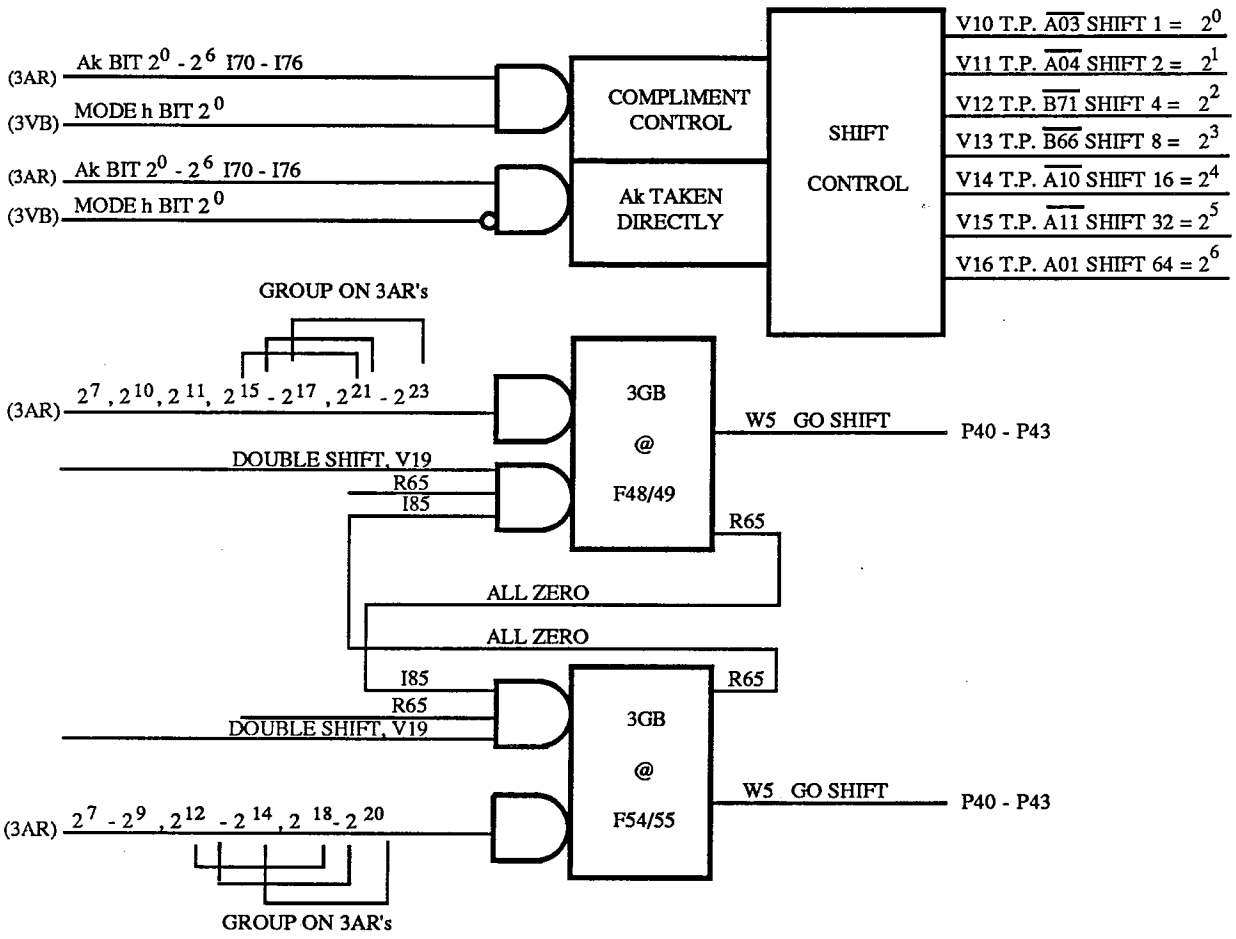


A-5127



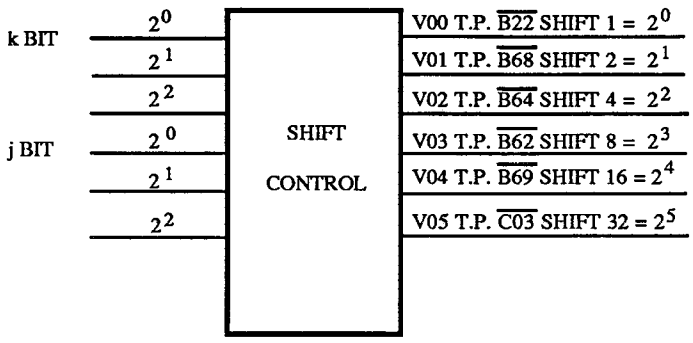
CRAY X-MP SCALAR SHIFT

SCALAR DOUBLE SHIFT COUNT
(3GB)



Ak 2^7 - 2^{23} EQUAL ZERO ALLOW THE OUTPUT TO THE 3GR's

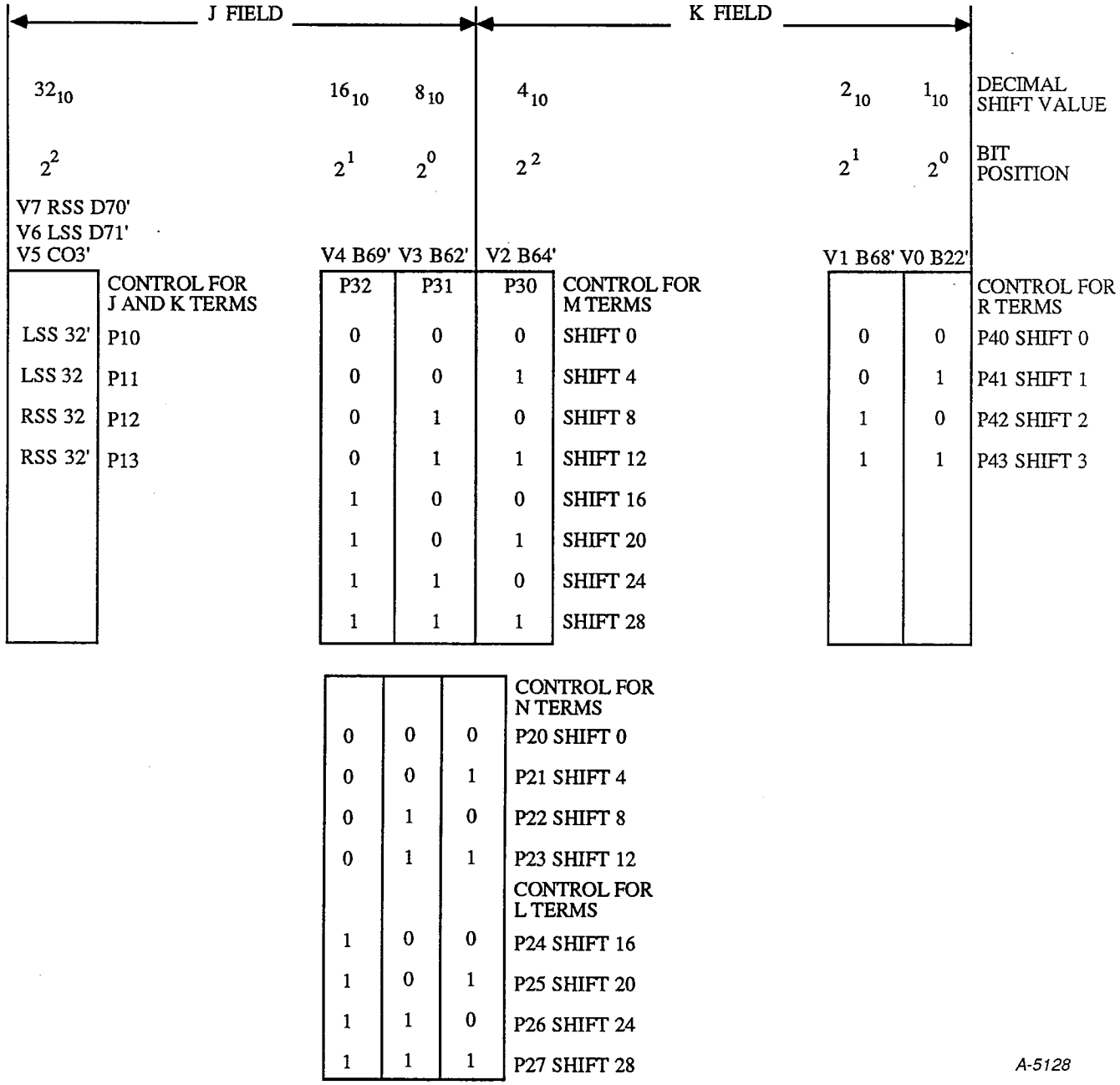
SCALAR SINGLE SHIFT COUNT
(3GB)



CRAY X-MP SCALAR DOUBLE/SINGLE SHIFT COUNT
(3GB)

B-1591A

1. SCALAR SINGLE SHIFT INSTRUCTIONS 052 i j k - 055 i j k
2. DECODE j AND k FIELD FOR SHIFT COUNT.
3. RIGHT SHIFT COUNTS MUST BE PUT IN 2's COMPLIMENT FORM.



SCALAR SHIFT CONTROL

62 60 58 56 54 52 50 48 46 44 42 40 38 36 34 32 30 28 26 24 22 20 18 16 14 12 10 8 6 4 2 0 3GB @ F54/55

63 61 59 57 55 53 51 49 47 45 43 41 39 37 35 33 31 29 27 25 23 21 19 17 15 13 11 9 7 5 3 1 3GB @ F48/49

31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 21 11 10 9 8 7 6 5 4 3 2 1 0 I TERMS

LEFT SINGLE SHIFT:
I0-I31 -- K0-K31 SHIFT $\overline{32}$
I0-I15 -- K16-K31 SHIFT 32

RIGHT SINGLE SHIFT:
I16-I31 -- K0-K15; I0-I15 -- J0-J15 SHIFT $\underline{32}$
I16-I31 -- J0-J15 SHIFT $\underline{32}$

K31 K30 K29 K28 K27 K26 K25 K24 K23 K22 K21 K20 K19 K18 K17 K16 | K15 K14 K13 K12 K11 K10 K9 K8 K7 K6 K5 K4 K3 K2 K1 K0 | J15 J14 J13 J12 J11 J10 J9 J8 J7 J6 J5 J4 J3 J2 J1 J0 SHIFT BITS

TS 10/13

N7 N6 N5 N4 N3 N2 N1 N0 M46 M44 M42 M40 M38 M36 M34 M32 M30 M28 M26 M24 M22 M20 M18 M16 M14 M12 M10 M8 M6 M4 M2 M0 L12 L10 SHIFT 0

N7 N6 N5 N4 N3 N2 N1 N0 M46 M44 M42 M40 M38 M36 M34 M32 M30 M28 M26 M24 M22 M20 M18 M16 M14 M12 M10 M8 M6 M4 M2 M0 L12 L10 SHIFT 4

N7 N6 N5 N4 N3 N2 N1 N0 M46 M44 M42 M40 M38 M36 M34 M32 M30 M28 M26 M24 M22 M20 M18 M16 M14 M12 M10 M8 M6 M4 M2 M0 L12 L10 SHIFT 8

N7 N6 N5 N4 N3 N2 N1 N0 M46 M44 M42 M40 M38 M36 M34 M32 M30 M28 M26 M24 M22 M20 M18 M16 M14 M12 M10 M8 M6 M4 M2 M0 L12 L10 SHIFT 12

M47 M45 M43 M41 M39 M37 M35 M33 M31 M29 M27 M25 M23 M21 M19 M17 M15 M13 M11 M9 M7 M5 M3 M1 L13 L11 L7 L6 L5 L4 L3 L2 L1 L0 SHIFT 16

M47 M45 M43 M41 M39 M37 M35 M33 M31 M29 M27 M25 M23 M21 M19 M17 M15 M13 M11 M9 M7 M5 M3 M1 L13 L11 L7 L6 L5 L4 L3 L2 L1 L0 SHIFT 20

M47 M45 M43 M41 M39 M37 M35 M33 M31 M29 M27 M25 M23 M21 M19 M17 M15 M13 M11 M9 M7 M5 M3 M1 L13 L11 L7 L6 L5 L4 L3 L2 L1 L0 SHIFT 24

M47 M45 M43 M41 M39 M37 M35 M33 M31 M29 M27 M25 M23 M21 M19 M17 M15 M13 M11 M9 M7 M5 M3 M1 L13 L11 L7 L6 L5 L4 L3 L2 L1 L0 SHIFT 28

M47 M45 M43 M41 M39 M37 M35 M33 M31 M29 M27 M25 M23 M21 M19 M17 M15 M13 M11 M9 M7 M5 M3 M1 L13 L11 L7 L6 L5 L4 L3 L2 L1 L0

N7 N6 N5 N4 N3 N2 N1 N0 M46 M44 M42 M40 M38 M36 M34 M32 M30 M28 M26 M24 M22 M20 M18 M16 M14 M12 M10 M8 M6 M4 M2 M0 L12 L10

TS 16/0 SHIFT BITS 2^1 2^0

R62 R60 R58 R56 R54 R52 R50 R48 R46 R44 R42 R40 R38 R36 R34 R32 R30 R28 R26 R24 R22 R20 R18 R16 R14 R12 R10 R8 R6 R4 R2 R0 SHIFT 0

2^{62} 2^{60} 2^{58} 2^{56} 2^{54} 2^{52} 2^{50} 2^{48} 2^{46} 2^{44} 2^{42} 2^{40} 2^{38} 2^{36} 2^{34} 2^{32} 2^{30} 2^{28} 2^{26} 2^{24} 2^{22} 2^{20} 2^{18} 2^{16} 2^{14} 2^{12} 2^{10} 2^{8} 2^{6} 2^{4} 2^{2} 2^{0} F54/55

2^{63} 2^{61} 2^{59} 2^{57} 2^{55} 2^{53} 2^{51} 2^{49} 2^{47} 2^{45} 2^{43} 2^{41} 2^{39} 2^{37} 2^{35} 2^{33} 2^{31} 2^{29} 2^{27} 2^{25} 2^{23} 2^{21} 2^{19} 2^{17} 2^{15} 2^{13} 2^{11} 2^{9} 2^{7} 2^{5} 2^{3} 2^{1} F48/49

R63 R61 R59 R57 R55 R53 R51 R49 R47 R45 R43 R41 R39 R37 R35 R33 R31 R29 R27 R25 R23 R21 R19 R17 R15 R13 R11 R9 R7 R5 R3 R1 R64 SHIFT 1

2^{63} 2^{61} 2^{59} 2^{57} 2^{55} 2^{53} 2^{51} 2^{49} 2^{47} 2^{45} 2^{43} 2^{41} 2^{39} 2^{37} 2^{35} 2^{33} 2^{31} 2^{29} 2^{27} 2^{25} 2^{23} 2^{21} 2^{19} 2^{17} 2^{15} 2^{13} 2^{11} 2^{9} 2^{7} 2^{5} 2^{3} 2^{1} 2^{-01} F54/55

2^{62} 2^{60} 2^{58} 2^{56} 2^{54} 2^{52} 2^{50} 2^{48} 2^{46} 2^{44} 2^{42} 2^{40} 2^{38} 2^{36} 2^{34} 2^{32} 2^{30} 2^{28} 2^{26} 2^{24} 2^{22} 2^{20} 2^{18} 2^{16} 2^{14} 2^{12} 2^{10} 2^{8} 2^{6} 2^{4} 2^{2} 2^{0} F48/49

R62 R60 R58 R56 R54 R52 R50 R48 R46 R44 R42 R40 R38 R36 R34 R32 R30 R28 R26 R24 R22 R20 R18 R16 R14 R12 R10 R8 R6 R4 R2 R0 SHIFT 2

2^{62} 2^{60} 2^{58} 2^{56} 2^{54} 2^{52} 2^{50} 2^{48} 2^{46} 2^{44} 2^{42} 2^{40} 2^{38} 2^{36} 2^{34} 2^{32} 2^{30} 2^{28} 2^{26} 2^{24} 2^{22} 2^{20} 2^{18} 2^{16} 2^{14} 2^{12} 2^{10} 2^{8} 2^{6} 2^{4} 2^{2} 2^{0} F54/55

2^{63} 2^{61} 2^{59} 2^{57} 2^{55} 2^{53} 2^{51} 2^{49} 2^{47} 2^{45} 2^{43} 2^{41} 2^{39} 2^{37} 2^{35} 2^{33} 2^{31} 2^{29} 2^{27} 2^{25} 2^{23} 2^{21} 2^{19} 2^{17} 2^{15} 2^{13} 2^{11} 2^{9} 2^{7} 2^{5} 2^{3} 2^{1} F48/49

R63 R61 R59 R57 R55 R53 R51 R49 R47 R45 R43 R41 R39 R37 R35 R33 R31 R29 R27 R25 R23 R21 R19 R17 R15 R13 R11 R9 R7 R5 R3 R1 R64 SHIFT 3

2^{63} 2^{61} 2^{59} 2^{57} 2^{55} 2^{53} 2^{51} 2^{49} 2^{47} 2^{45} 2^{43} 2^{41} 2^{39} 2^{37} 2^{35} 2^{33} 2^{31} 2^{29} 2^{27} 2^{25} 2^{23} 2^{21} 2^{19} 2^{17} 2^{15} 2^{13} 2^{11} 2^{9} 2^{7} 2^{5} 2^{3} 2^{1} 2^{-01} F54/55

2^{62} 2^{60} 2^{58} 2^{56} 2^{54} 2^{52} 2^{50} 2^{48} 2^{46} 2^{44} 2^{42} 2^{40} 2^{38} 2^{36} 2^{34} 2^{32} 2^{30} 2^{28} 2^{26} 2^{24} 2^{22} 2^{20} 2^{18} 2^{16} 2^{14} 2^{12} 2^{10} 2^{8} 2^{6} 2^{4} 2^{2} 2^{0} F48/49

R15 R14 R13 R12 R11 R10 R9 R8 R7 R6 R5 R4 R3 R2 R1 R0 TEST POINT

B5 B6 B14 B13 B12 B7 B15 B11 B2 B10 B18 B9 B8 B3 B4 B1

R31 R30 R29 R28 R27 R26 R25 R24 R23 R22 R21 R20 R19 R18 R17 R16 TEST POINT

$\overline{A53}$ $\overline{A69}$ $\overline{A47}$ $\overline{A52}$ $\overline{A16}$ $\overline{A50}$ $\overline{A21}$ $\overline{A19}$ $\overline{A64}$ $\overline{A67}$ $\overline{A68}$ $\overline{A51}$ $\overline{A70}$ $\overline{A71}$ $\overline{A65}$ $\overline{A66}$

R47 R46 R45 R44 R43 R42 R41 R40 R39 R38 R37 R36 R35 R34 R33 R32 TEST POINT

$\overline{C27}$ $\overline{C65}$ $\overline{C63}$ $\overline{C64}$ $\overline{C23}$ $\overline{C2}$ $\overline{C1}$ $\overline{C21}$ $\overline{C25}$ $\overline{C68}$ $\overline{C66}$ $\overline{C69}$ $\overline{C67}$ $\overline{C71}$ $\overline{C70}$ $\overline{C72}$

R65 R64 R63 R62 R61 R60 R59 R58 R57 R56 R55 R54 R53 R52 R51 R50 R49 R48 TEST POINT

D4 $\overline{A72}$ D64 D35 D58 D53 D34 D32 D47 D39 D48 D2 C49 D66 D69 D6 D5 D1

D-1553D

CRAY X-MP/2 SCALAR SHIFT
SINGLE LEFT/RIGHT SHIFT