B1700 HARDWARE SPECIFICATIONS

DISTRIBUTION

DETROIT

D. Kosinski - CPM
R. Stackhouse - CPM
D. Bookwalter - CPM
C. Chamberlain - Tech Pub
J. Raker - Int'l
T. Allison - Int'l (2)
R. Gould - Int'l
H. Hayde - Int'l
J. Holton - Int'l
C. Nash - Int'l
E. Seif - Int'l
H. Nelson - BM&SS
J. Shifman - CSG
E. Rindels - BMG
W. Cunningham - BMG
E. Paulsen - TIO

PLYMOUTH

D. Calkins

DOWNINGTOWN

J. Berta

PAOLI

G. Smolnik

McLEAN

L. Guell
A. Kosla

PASADENA

D. Gale
J. Malson

MISSION VIEJO

T. Gibson

WESTLAKE

H. Smith
J. Dowling

GENNEVIELIERS

A. Isola

CUMBERNAULD

D. Wright

SENEFFE

J. Jeanty

GLENROTHES

B. Slattery

LIEGE

J. Wery
A. Beaujean

MIDDLESEX

W. Marshall

SANTA BARBARA

R. Bunker
Manufacturing (2)
Quality Assurance
TIO (2)
Software (30)
W. Keenan (5)
Hardware - J. Vigil
J. Trost
S. Bhagwan (2)

F. E. TRAINING DEV. GROUP

M. McCoy
3374 Rio Hondo Avenue
El Monte, California 91731

PATENT ATTORNEY

Mr. D. Bruce Prout
Christie, Parker & Hale
201 South Lake Avenue
Pasadena, California 91101

TIREMAN

F. Williams

P.S. #2201 6760

Burroughs Corporation

BUSINESS MACHINES GROUP
SMALL SYSTEMS PLANT

B1700 S-MEMORY PROCESSOR

PRODUCT SPECIFICATION

REVISIONS

REV LTR	REVISION ISSUE DATE	PAGES REVISED ADDED DELETED OR CHANGE OF CLASSIFICATION	PREPARED BY	APPROVED BY
F	1-8-76	Revised document primarily to reflect addition of S-Processor-2. Preface: Added information on S-Processor-2 Rel Specs: Added PS #2209 7968, SDS #2210 0143. Sec 2.0: Added M&E 2208 3141, 2208 3158, 2212 8276. Sec 3.1.11: Added PERR. Sec 3.3.10: Added reference to S-Proc-2. Added note on M-op fetch. Added note on addressing non-present memory. Sec 3.3.12: Added reference to S-Proc-2. Sec 3.3.14: Revised, expanded U-Register information. Sec 3.3.15: Revised eigth paragraph (bus interrupt). Added paragraph on C-Register bit reset. Sec 3.3.16, 3.3.17: Deleted references to MOVE, MOVE SCRATCHPAD. Sec 3.3.19, 3.3.20: Expanded to include operations from control panel. Sec 3.4.1: Exception (4) expanded to include SUM, DIFF; Exception (5) revised; Exception (6) deleted; Exception (7) renumbered to (6). Sec 3.4.19, 3.4.20, 3.4.21: Added notes reference reset of bits in C-Register. Sec 3.5: Deleted (branch taken) after BIAS.	WFK	CEWymore J Hale E Frost [illegible] 1-8-76
		Reorganized and copy edited entire document for entry into Product Specification data base. New section sequence is as follows. Asterisks indicate revisions referred to above. Was — Is Preface — 1.0* — 1.1* 2.0 — 1.2* — 1.3 1.0 — 2.0 3.0 — 3.0 — 3.1 3.3.10 — 3.1.1*	AAM	Al Wellkamm 1/8/75

Burroughs Corporation

BUSINESS MACHINES GROUP
SMALL SYSTEMS PLANT

B1700 S-MEMORY PROCESSOR

PRODUCT SPECIFICATION

REVISIONS

REV LTR	REVISION ISSUE DATE	PAGES REVISED ADDED DELETED OR CHANGE OF CLASSIFICATION	PREPARED BY	APPROVED BY
F	1-8-76 Cont.	Was → Is 3.3.11 → 3.1.2 3.3.9 → 3.1.3 3.3.8 → 3.1.4 3.3.1 → 3.1.5 3.3.2 → 3.1.6 3.3.3 → 3.1.7 3.3.4 → 3.1.8 3.3.5 → 3.1.9 3.3.6 → 3.1.10 3.1.9 → 3.1.11 3.3.7 → 3.1.12 3.3.15 → 3.1.13* → 3.1.13.1 → 3.1.13.2 3.3.12, 3.3.13 → 3.1.14* 3.3.14 → 3.1.15* 3.3.16 → 3.1.16* 3.3.17 → 3.1.17* 3.3.18 → 3.1.18 3.3.19 → 3.1.19* 3.3.20 → 3.1.20* 3.1 → 3.2 3.1.1 → 3.2.1 3.1.2 → 3.2.2 3.1.3 → 3.2.3 3.1.4 → 3.2.4 3.1.5 → 3.2.5 3.1.6 → 3.2.6 3.1.7 → 3.2.7 3.1.8 → 3.2.8 3.1.10, 3.1.11 → 3.2.9* 3.2 → 3.3 Numbering of Sec 3.4 and 3.5 remain unchanged.		

S-MEMORY PROCESSOR

PRODUCT SPECIFICATION

2201 6760

TABLE OF CONTENTS:

1.0 INTRODUCTION

1.1 PURPOSE

This specification defines the functional requirements of two S-Memory Processors intended for general purpose data processing. Environmental conditions, safety conditions, reliability parameters, power requirements, etc., are specified in P.S. #1913 1739, B1700 CENTRAL SYSTEMS.

Both processors have I/O bus interfaces for communication with the I/O subsystem, control panel interfaces for communication with the control panel, and memory-to-processor interfaces for communication with main memory (B1700 S-Memory).

S-Memory-Processor-1 can address up to 64K bytes (512K bits) of memory. S-Memory-Processor-2 can address up to 128K bytes (1024K bits).

Both processors are capable of operating with 2MHz as well as 4MHz clocks.

An appropriate processor adapter, either 2MHz or 4MHz, is required with S-Processor-1; S-Processor-2 requires no adapter but only a field adjustment of the clock rate.

1.2 PRODUCT IDENTIFICATION

2205 0967	S-MEMORY PROCESSOR-1
2208 3141	S-PROCESSOR-1 ADAPTER, 2MHz
2208 3158	S-PROCESSOR-1 ADAPTER, 4MHz
2212 8276	S-MEMORY PROCESSOR-2

1.3 RELATED SPECIFICATIONS

P.S. #	NAME
1904 5681	B1700 SYSTEM INDEX
1913 1739	B1700 CENTRAL SYSTEM
2204 8623	B1700 I/O BUS SUBSYSTEM
1913 1754	B1700 MEMORY SUBSYSTEM
2205 7376	B1700 S-MEMORY STORAGE CARD
2209 7968	B1700 CONTROL PANEL

S.D.S. #	NAME
2204 8656	B1700 S-MEMORY/S-PROCESSOR INTERFACE
2210 0143	B1700 CONTROL PANEL INTERFACE

2.0 GENERAL DESCRIPTION

The S-Memory Processor (or S-Processor) provides the combinatorial and arithmetic portion of the system along with other registers and hard storage that are appropriate for efficient operation.

The S-Processor provides hardware sensitivity to a set of low-level micro-functions which are used in a program string to fetch and execute instructions. This micro-program is contained in main memory (B1700 S-Memory).

Included in the S-Processor are registers and pseudo registers which are addressable by the individual micro-operators.

The registers are normally addressed by a 4-bit group (row) number and a 2-bit select (column) number as shown in Table 1.

Some of the registers listed, such as the Pseudo Sum Register, can serve only as source registers while others are capable of serving both as source and destination registers. Also, some of the registers listed are actually subregisters which, although parts of larger registers, can be individually addressed and manipulated.

Table 2 summarizes the various conditions available by addressing particular pseudo source registers and actual registers; Figure 1 lists the micro-instructions and their variants; and Figure 2 is a diagram of the major registers.

Only the MAR(A), M, C, U, ML, and MIR registers are reset by a control panel CLEAR signal.

TABLE 1: M-PROCESSOR REGISTER SELECTION

		SELECT NUMBER 0	1	2	3
GROUP NUMBER	0	TA	FU	X	SUM
	1	TB	FT	Y	CMPX
	2	TC	FLC	T	CMPY
	3	TD	FLD	L	XANY
	4	TE	FLE	A	XEQY
	5	TF	FLF	M	MSKX
	6	CA	BICN	BR	MSKY
	7	CB	FLCN	LR	XORY
	8	LA	TOPM*	FA	DIFF
	9	LB	UNASSIGNED	FB	MAXS
	10	LC	UNASSIGNED	FL	MAXM
	11	LD	PERR*	TAS	U
	12	LE	XYCN	CP	MBR*
	13	LF	XYST	MSM*	DATA
	14	CC	INCN*	READ	CMND
	15	CD	CPU	WRIT	NULL

(*) TOPM, PERR, INCN, MSM, AND MBR are not physically present in the B1700 S-Memory Processor-1 or -2. When any of them is addressed as a source, a binary value of zero is returned. When addressed as destination, the data is lost. (See P.S. #1913 1747, M-MEMORY PROCESSOR, for meaning in the B1700 M-Memory Processors.)

TABLE 2: SUMMARY OF REGISTER CONDITIONS

BICN:	LSUY	CYF	CYD	CYL
XYCN:	MSBX	X=Y	X<Y	X>Y
XYST:	LSUX	INT	Y neq 0	X neq 0
FLCN:	FL=SFL	FL>SFL	FL<SFL	FL neq 0

CC:	CONTROL PANEL STATE LAMP FLIP-FLOP	TIMER INTERRUPT	I/O BUS INTERRUPT	CONTROL PNL INTERRUPT
CD:	MEMORY READ DATA PAR. ERR. INTERRUPT	MEMORY RSVD WRT/SWAP ADDR OUT OF BOUNDS OVERRIDE	MEMORY RSVD READ ADDR OUT OF BNDS INTERRUPT	MEMORY RSVD WRT/SWAP ADDR OUT OF BOUNDS INTERRUPT

Software

0 1 2 3

S ??

	MICRO NAME	MC 15	14	13	12	MD 11	10	9	8	ME 7	6	5	4	MF 3	2	1	0
1C	REGISTER MOVE	0	0	0	1	REG 1 GROUP SOURCE REGISTER				REG 1 SELECT			REG 2 SELECT	REG 2 GROUP SINK REGISTER			
2C	SCRATCHPAD MOVE	0	0	1	0	REGISTER GROUP SOURCE OR SINK				REG SELECT		MOV DIR	DPW 1/2	DOUBLE PAD WORD ADDRESS			
3C	4 BIT MANIPULATE	0	0	1	1	REGISTER GROUP, 4 BIT SOR & SNK				REG SEL	MANIPULATE VARIANTS			4 BIT MANIP. LITERAL			
4C	BIT TEST REL BRANCH FALSE	0	1	0	0	REGISTER GROUP, 4 BIT SOURCE				REG SEL	TESTBIT NUMBER		DSP SGN	RELATIVE BRANCH DISPLACEMENT MAG.			
5C	BIT TEST REL BRANCH TRUE	0	1	0	1	REGISTER GROUP 4 BIT SOURCE				REG SEL	TESTBIT NUMBER		DSP SGN	RELATIVE BRANCH DISPLACEMENT MAG.			
6C	SKIP WHEN	0	1	1	0	REGISTER GROUP 4 BIT SOR & SNK				REG SEL	SKIP TEST VARIANTS			4 BIT TEST MASK			
7C	READ/WRITE MEMORY	0	1	1	1	R/W VAR	COUNT FA/FL VARIANTS			DATAREG CODE		TW SGN		DATA TRANSFER WIDTH MAGNITUDE			
8C	MOVE 8 BIT LITERAL	1	0	0	0	REGISTER GROUP; REG SEL IS 2.				ENTIRE 8 BITS OF 8 BIT LITERAL							
9C	MOVE 24 BIT LITERAL	1	0	0	1	REGISTER GROUP; REL SEL IS 2.				8 MOST SIGNIFICANT BITS OF FULL 24 BIT LITERAL							
10C	SHIFT/ROTATE T REG	1	0	1	0	SINK REGISTER GROUP				SNK REG SELECT		S/R VAR		LEFT SHIFT/ROTATE COUNT			
11C	EXTRACT FROM T REG	1	0	1	1	RIGHT BIT POINTER FOR EXTRACTION FLD				SNK REG CODE				EXTRACTION FIELD WIDTH			
123C	BRANCH RELATIVE	1	1	0	DSP SGN	RELATIVE DISPLACEMENT MAGNITUDE											
145C	CALL RELATIVE	1	1	1	DSP SGN	RELATIVE CALLED ADDRESS MAGNITUDE											
4D	SHIFT/ROTATE X OR Y	0	0	0	0	0	1	0	0	S/R DIR VARIANT	X/Y VAR			LEFT OR RIGHT, X OR Y SHIFT/ROTATE COUNT			
5D	SHIFT X AND Y	0	0	0	0	0	1	0	1	S/R, DIR VARIANT				LEFT OR RIGHT X AND Y SHIFT/ROTATE COUNT			
6D	COUNT FA/FL	0	0	0	0	0	1	1	0	COUNT FA/FL VARIANTS				COUNT SCALAR MAGNITUDE			
7D	EXCHANGE DPW	0	0	0	0	0	1	1	1	SINK DPW ADDRESS				SOURCE DPW ADDRESS			
8D	SCRATCHPAD RELATE FA	0	0	0	0	1	0	0	0				DSP SGN	LEFT HALF PAD WORD ADDRESS			
9D	MONITOR	0	0	0	0	1	0	0	1	LITERAL OCCURRENCE IDENTIFIER							
2E	CASSETTE CONTROL	0	0	0	0	0	0	0	0	0	0	1	0	CASSETTE MANIP VARIANTS			
3E	BIAS	0	0	0	0	0	0	0	0	0	0	1	1	BIAS VARIANTS			TST FLG
6E	CARRY FF MANIPULATE	0	0	0	0	0	0	0	0	0	1	1	0	CYF CYD	CYF CYL	CYF 1	CYF 0
1F	HALT	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
3F	NORMALIZE X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
ZERO	NO OPERATION	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

VARIANTS	0 000	1 001	2 010	3 011	4 100	5 101	6 110	7 111
MOV DIR: 1/2 DPW:	P←R LEFT	R←P RIGHT						
MANIP. VARIANTS:	SET	AND	OR	EOR	INC	INC TEST	DEC	DEC TEST
DSP SIGN:	+	−						
DSP SIGN:	+	−						
SKIP TEST VARIANTS:	ANY CLR/	ALL CLR/	EQL CLR/	ALL CLR	ANY/ CLR/	ALL/ CLR/	EQL/ CLR	ALL/ CLR
R/W VAR:	READ	WRT						
CNT VAR:	NØP	FA↑		FA↑	FA↓	FA↑		FA↓
REG SEL: TW SIGN:	X +	Y −	FL↑ T	FL↓ L	FL↑		FL↑	FL↑
S/R VAR:	SHFT	ROT						
SINK REG CODE:	X	Y	T	L				
DSP SIGN:	+	−						
DSP SIGN:	+	−						
X/Y VAR:	X	Y						
S/R, DIR:	SFT←	SFT→	ROT←	ROT→				
S/R, DIR VARIANTS:	SFT← 1BIT	SFT→ 1BIT	RES.	RES.				
COUNT FA/FL VAR:	NØP	FA↑	FL↑	FA↑ FL↑	FA↑ FL↑	FA↑	FL↓	FA↓ FL↓
DSP SIGN:	+	−						
CASSETTE MANIP:	START TAPE	STOP@GAP	STOP ON X≠Y	UNDEF	UNDEF	UNDEF	UNDEF	UNDEF
TEST FLG: BIAS VAR:	TST/ UNIT	TEST F	S	FS	NØP	FCP	NØP	UNDEF

REGISTER SELECT

REGISTER GROUP	0	1	2	3
0	TA	FU	X	SUM
1	TB	FT	Y	CMPX
2	TC	FLC	T	CMPY
3	TD	FLD	L	XANY
4	TE	FLE	MAR(A)	XEOV
5	TF	FLF	M	MSKX
6	CA	BICN	BR	MSKY
7	CB	FLCN	LR	XORY
8	LA	RES.	FA	DIFF
9	LB	....	FB	MAXS
10	LC	....	FL	MAXM
11	LD	....	TAS	U
12	LE	XYCN	CP	RES.
13	LF	XYST	RES.	DATA
14	CC	RES.	READ	CMND
15	CD	CPU	WRIT	NULL

INDIVIDUAL BITS OF SOME 4-BIT REGISTERS HAVE SPECIAL MEANINGS AS NOTED BELOW:

NOTE: BITS ARE NUMBERED HERE ACCORDING TO THE HARDWARE CONVENTION:

LA	LB	LC	LD	LE	LF
3 0	3 0	3 0	3 0	3 0	3 0
MSB 23			L		LSB 00

BIT #	BICN	XYCN	XYST	FLCN
0	CYL	X>Y	X≠0	FL≠0
1	CYD	X<Y	Y≠0	FL<SFL
2	CYF	X=Y	INT OR	FL>SFL
3	LSUY	MSBX	LSUX	FL=SFL

INT OR = ANY 1 OR MORE OF: CC0 CC1 CC2 CD3

CC & CD

CC0 CONSOLE SOFT HALT INTERRUPT.
CC1 I/O BUS SERVICE REQUEST INTERPT.
CC2 REAL TIME CLOCK (100 MS) INTERPT.
CC3 STATE FLAG (DISPLAYED ON CONSOLE).
CD0 RESERVED
CD1 RESERVED
CD2 RESERVED
CD3 MEMORY READ DATA PARITY ERROR INTERRUPT.

FIGURE 1 S-PROCESSOR MICRO-INSTRUCTIONS AND VARIANTS

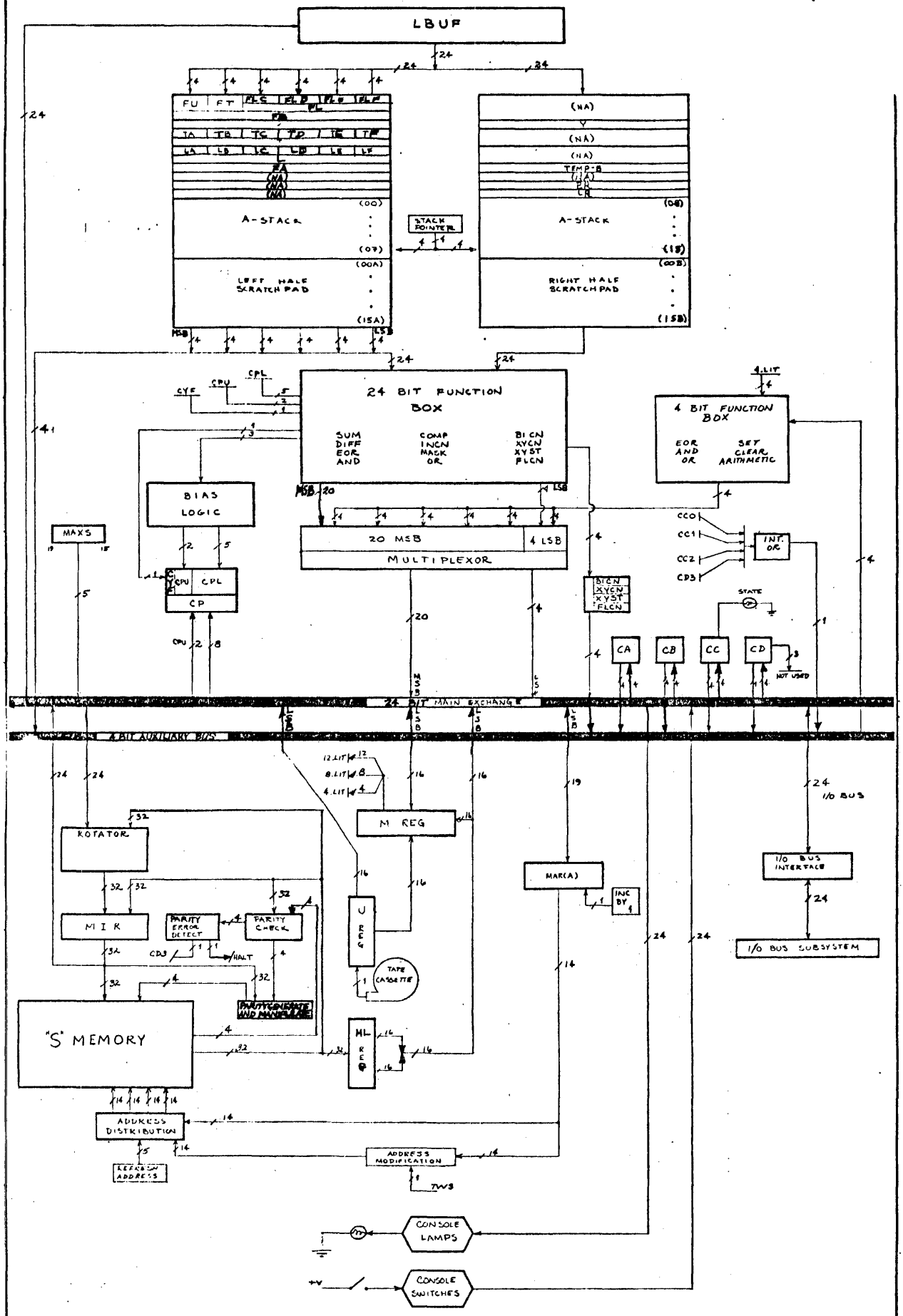


FIGURE 2 S-PROCESSOR REGISTERS

3.0 PRODUCT DESCRIPTION

3.1 PROCESSOR REGISTERS

3.1.1 MAR(A)

The MAR(A) Register is a micro-program address register capable of addressing micro-operators located in S-Memory. When used in this manner the lower-order 4 bits are ignored and the operator is fetched and stored in the M Register, bypassing the rotator (field isolation unit). [The MAR(A) Register is comprised of 19 bits in S-Processor-1 and 20 bits in S-Processor-2.]

Binary increments from 0 through 4095 (multiplied by 16) may be added to or subtracted from MAR(A) by means of a high-speed carry adder that facilitates micro-program branching. MAR(A) is automatically incremented by binary 16 during run mode. Wrap-around can occur and is permitted.

Note: The fetch of an M-op from the last (highest addressed) 16 bits of installed S-Memory is not permitted since the automatic fetch of the next M-op will in some cases address portions of memory that are not present which, in turn, may result in parity errors. In addition, the maximum allowable S-Memory is subject to change.

The MAR(A) Register addresses 16-bit micro-operators in S-Memory by assuming them to be located at bit boundary addresses exactly divisible by 16. These addresses are obtainable by ignoring the rightmost 4 bits of the register.

The MAR(A) Register is addressable as a source and as a sink. When addressed as a source, all bits are produced. However, the rightmost 4 bits are masked to zeros. When addressed as a sink, the rightmost 4 bits of the source are moved to the rightmost 4 bits of the MAR(A) Register, but are not significant.

The MAR(A) Register also serves to address S-Memory for the READ/WRITE micro-operator. In this case the address of the next micro-operator is temporarily stored in a scratchpad holding register and the address in FA is moved to the

MAR(A) Register. The most significant bits of FA are ignored. The least significant bits are not ignored and are used to address memory at any bit location. The data is passed through the rotator.

Note: When addressing S-Memory via the READ/WRITE micro-operator, data returned from non-present memory locations will be zeros with odd parity if the address in MAR(A) is in-bounds, and zeros with wrong parity if the address in MAR(A) is out-of-bounds. In addition, if maximum S-Memory is installed, data and parity from the extremes of memory (lowest addressed portion if addressed in the forward direction and highest addressed portion if addressed in the backward direction) will be bit-ORed with the zeros and odd parity bits, obtained because of the out-of-bounds condition. This, in some cases, may result in parity error.

3.1.2 M

The M Register (micro-register) is a 16-bit register used to hold the active micro-instruction (M-instruction or micro-operator or M-op). The state of this register is decoded to enable the different control signals within the processor to perform the operations called for by the M-Op. The M Register is broken into 4 fields for decoding that are structured such that 60 distinct M-ops can be decoded.

The M Register is addressable as a source and as a sink (destination). When used as a sink register, the source is bit-ORed with the upcoming M-op. Exception: In TAPE mode, the source is not bit-ORed with the upcoming M-op.

3.1.3 A STACK

The A Stack is 16 words deep by 24 bits wide, does not have automatic hard overflow, and operates as a push-down stack with a last-in, first-out type of structure. Using this stack, the microcoded routines operate in the normal software call-return type of programming. This allows for a highly shared microcode structure and reduces the associated memory requirements. Although the A Stack is not intended to be used as an operand stack, it has purposely been made 24 bits wide to allow limited

capabilities for operand storage.

Wrap around of the TAS pointer is provided. That is, 32 consecutive pops or 32 consecutive pushes will cause the TAS pointer to contain its original contents.

3.1.4 TAS

The TAS (Top of Stack) Register is a 24-bit register which is the top of the A Stack. The TAS Register is addressable as a source and as a sink. References to TAS result in the stack being automatically pushed or popped.

3.1.5 X AND Y

The X Register and the Y Register are 24-bit general purpose registers used primarily to hold and to act as a source for the two operands of the arithmetic and combinatorial unit. Each register is addressable as a source and as a sink.

Both registers, along with the L Register and the T Register, are capable of read/write operations with main memory.

Both registers are capable of the SHIFT/ROTATE operation. The X Register is capable of the NORMALIZE operation.

3.1.6 L

The L Register is a 24-bit general purpose register used typically to hold logical flags for the micro-program code. The L Register, as well as each 4-bit group of L (denoted as LA, LB, LC, LD, LE, and LF) is addressable as a source and as a sink.

Since the L Register is addressable in 4-bit groups, its contents are available for analysis and alteration via the 4-bit function box. MANIPULATE, SKIP and BIT-TEST-BRANCH instructions can operate on L data.

The L Register is one of 4 registers (X, Y, L, and T) capable of read/write operations with main memory.

3.1.7 T

The T Register is a 24-bit general purpose register used primarily for the interpretation of S-Language instructions. The T Register, as well as each 4-bit group of T denoted as TA, TB, TC, TD, TE, and TF, is addressable as a source and as a sink.

Since the T Register is addressable in 4-bit groups, its contents are available for analysis and alteration via the 4-bit function box. MANIPULATE, SKIP and BIT-TEST-BRANCH instructions can operate on T data.

The T Register is one of 4 registers (X, Y, L, and T) capable of read/write operations with main memory.

The T Register is also capable of the SHIFT/ROTATE and EXTRACT operations.

3.1.8 FA

The FA Register (Field Address) is a 24-bit register used primarily to hold an absolute bit address for main memory. It has the capability of directly addressing any bit in the memory starting at any point.

The FA Register is addressable as a source and as a sink.

The FA Register is capable of being counted up or down by a literal in a micro-instruction or by the value contained in CPL in order to facilitate iteration through a memory field. It can be incremented or decremented by a value in a left scratchpad word. It also has the capability of being loaded, stored, or swapped along with FB into a double scratchpad word.

Neither overflow nor underflow of FA is detected. The value of FA may go through its maximum value or its minimum value and wrap around.

3.1.9 FB

The FB Register is a 24-bit register which can be functionally divided into three portions: a 4-bit FU (Field Unit) Register, a 4-bit FT (Field Type) Register, and a 16-bit FL (Field Length) Register.

The FB Register, as well as each 4-bit portion of FB (denoted as FU, FT, FLC, FLD, FLE, and FLF) is addressable as a source and as a sink. In addition, the 16-bit portion comprised of FLC, FLD, FLE, and FLF and denoted as FL is also addressable as source and as a sink.

The FU Register holds the length of the unit which makes up a field in memory. The FT Register holds field type information while the FL Register holds the total length of the field. FL is capable of describing fields up to 65,636 bits and can be adjusted up or down by a literal in a micro-instruction or by the value contained in CPL in order to facilitate iteration through a memory field.

Note: Overflow of FL is not detected. The value of FL will go through its maximum value and wrap around. Underflow of FL is detected and will not wrap around. The value of zero is left in FL.

Since the FB Register is addressable in 4-bit groups, its contents are available for analysis and alteration via the 4-bit function box. MANIPULATE, SKIP and BIT-TEST-BRANCH instructions can operate on FB data.

FB has the capability of being swapped along with FA into a double scratchpad memory word.

FU and FL along with corresponding portions of the first cell of scratchpad are used to set the various conditions of FLCN (see Section 3.1.11, FLCN) and the various conditions of the CP register (see Section 3.4.16, BIAS).

3.1.10 SCRATCHPAD

A scratchpad memory of sixteen 48-bit words is provided to hold field descriptors during the iteration of operands. Some cells may be used to hold S-Language stack pointers

and other processor registers which are under constant manipulation.

The FU and FL portion of the FB Register and like portions of the first cell of the scratchpad are used to set the various conditions of FLCN (see Section 3.1.11, FLCN) and the various conditions of the CP register (see Section 3.4.16, BIAS).

3.1.11 FLCN

FLCN (field length condition) is a 4-bit pseudo register that holds the result of a static comparison of the FL portion of the FB register and the corresponding portion of the first scratchpad word. It is addressable as a source only.

FLCN:	FL=SFL	FL>SFL	FL<SFL	FL neq 0

3.1.12 BR AND LR

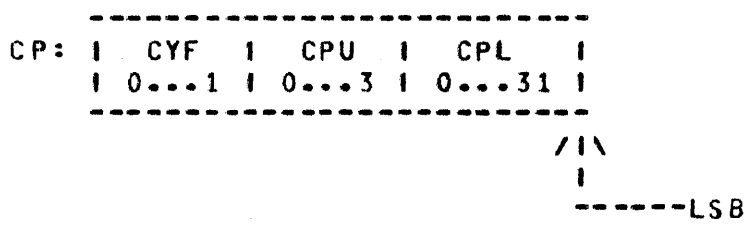
The LR and BR Registers are both 24-bit registers, addressable as sources and destinations.

3.1.13 C

The C (control) Register is a 24-bit register which is not addressable is an entity but which is functionally divided into one 8-bit section and four 4-bit sections.

3.1.13.1 CP (CYF, CPU, CPL)

The 8-bit section, addressed as CP, is comprised of the arithmetic unit carry flip-flop (CYF), the 2-bit unit control for the arithmetic unit (CPU) and the 5-bit variable data length control (CPL). CPU is addressable as a sink only.

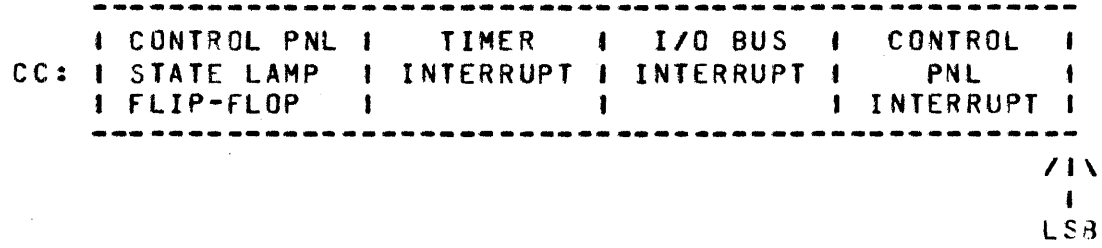


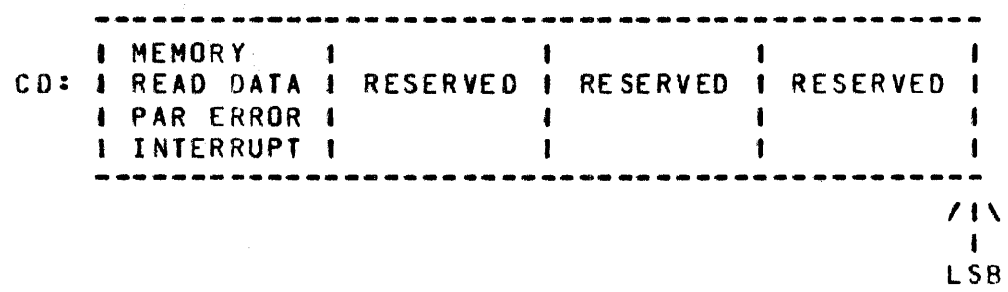
3.1.13.2 CA, CB, CC, CD

The remaining 16-bits of the C Register are addressable in 4-bit groups as CA, CB, CC and CD. Their contents are available for analysis and alteration via the 4-bit function box. MANIPULATE, SKIP and BIT-TEST-BRANCH instructions are applicable.

The 4-bit groups designated as CA and CB have no special functional assignment and are available as general purpose 4-bit storage registers.

The two 4-bit registers designated as CC and CD are used for the storage of various processor states and conditions as shown below:





The control panel state lamp flip-flop when true will cause the STATE lamp to light.

The timer interrupt signal is developed from the primary power frequency, field adjustable for either 50Hz or 60Hz. The interrupt signal is received and is used to set the appropriate CC bit once every 100 milliseconds.

The I/O Bus interrupt level is derived from the various I/O controls connected to the processor's I/O Bus. The level is the result of a service request by one or more controls and is used to set the interrupt bit every clock time.

The control panel interrupt level is derived from the on position of the control panel's INTERRUPT switch. The level from the switch is used to set the interrupt bit every clock time.

The memory read data parity error interrupt is set as a result of a parity error signal received from the interface to main memory.

No reaction occurs as a result of any interrupt until the micro-program tests the interrupt bit.

A micro-instruction or the control panel CLEAR pushbutton is capable of resetting a bit in the C Register. The bit being reset will be false for at least one clock period following the reset regardless of the continued existence of the condition to set the bit (e.g., control panel or service request interrupts). Any test micro-operator executed in this clock period will find the bit false. If the condition does not continue to exist beyond the reset time, a failure to set the bit may occur (e.g., timer interrupt).

3.1.14 MAXS, MAXM

MAXS is a 24-bit pseudo register that can be field-adjusted to give the actual size of the main memory installed in the system. It is addressable as a source only. MAXS has 4K-byte resolution. The least significant 15 bits and the most significant 5 (S-Processor-1) or 4 (S-Processor-2) bits of the register are always zeroes. MAXM, also a 24-bit pseudo register, always has a value of zero for the S-Processors.

3.1.15 U

The U Register is a 16-bit register used primarily to accumulate the bit-by-bit input from the control panel's tape cassette. The U Register is addressable as a source register only.

In RUN mode, if data is not yet available in the register, the micro-operator will be delayed. Data not accepted in time will be lost.

In TAPE mode, the register's contents are automatically moved to the M Register for execution except when executing micro-operators that reference the U Register as a source. In these cases the source data is moved directly from the U Register to the destination and will not be treated as an instruction to be executed. The next instruction from the tape will follow this data. For MOVE 24-BIT LITERAL, the 8 bits of the literal in the M Register are also moved to the destination. For CONDITIONAL BRANCH, the A Register may be changed but the next micro-operator read from the tape will also be executed.

Note: In RUN mode, the U Register may not be addressed after the issuance of a CASSETTE STOP micro.

3.1.16 DATA

DATA is a 24-bit pseudo register that can act as a source or as a destination. It is used to transfer data to and from the I/O Bus. When it is used as a source, the

processor generates the RC (RESPONSE COMPLETE) signal to the interface and accepts the 24-bits of data from the bus. When used as a destination, the processor generates the RC signal to the interface and the data from the designated source to the bus.

3.1.17 CMND

CMND (Command) is a 24-bit pseudo register that can act as a destination only. It is used to transfer commands to devices on the I/O Bus. The processor generates the CA signal to the interface and moves the data (or the command) from the designated source to the bus.

3.1.18 NULL

NULL is a 24-bit pseudo register that can act as a source and as a destination. When addressed as a source, all zeros are supplied to the destination. When addressed as a destination, the source data is not accepted. However, NULL is useful as a destination in order to pop the TAS Register without affecting other registers.

3.1.19 READ

READ is a 24-bit pseudo register that is not permitted to be addressed either as a source or as a destination in the S-Processors. It is used in conjunction with a memory read operation from the control panel. (See P.S. #1913 1747 for meaning in the M-Processors.)

3.1.20 WRIT

WRIT is not permitted to be addressed as a source or as a destination, but is used in conjunction with a memory write operation from the control panel.

3.2 24-BIT FUNCTION BOX

The 24-bit "function box" is composed of a 24-bit arithmetic unit and a 24-bit combinatorial unit. It has as data inputs the contents of the X and Y Registers and the carry flip-flop (CYF). It also uses the CPU and CPL portions of the C Register.

All results from the combinatorial section are generated immediately and are continuously available to the micro-programmer. A move to one of the input registers or an alteration of a value in the CP portion of the C Register immediately generates a new result. The results are available to the next micro-instruction and are accessed by moving the contents of a result register to a destination register or by testing one of the 4-bit condition registers.

The results are most of the commonly used functions between two operands. These include the And, Or, Exclusive-Or, sum, carry-out, difference, and borrow functions and the set of equal-to, greater-than, and less-than relationals. The results of the unary operations of complementation and masking are also available.

The results of the arithmetic unit are under control of the CPU and the CPL Registers as follows.

CPU	UNIT TYPE	POSSIBLE CPL VALUES	DATA TYPE
00	1-bit operands	1 to 24	Binary
00	4-bit operands	4,8,12,16,20,or 24	4-bit binary
10	Undefined		
11	Undefined		

For valid arithmetic operations, the operand length (as specified by CPL) must be an exact multiple of the length of the unit specified by CPU.

The contents of each of the registers described in the following subsections are immediately available to the micro-programmer.

3.2.1 SUM

SUM is a pseudo register equal to the sum of the X, Y, and CYF Registers (X + Y + CYF). Zero bits in the more significant bit positions of the 24-bit result are produced when the length as determined by CPL (5-bit variable data length control) is less than 24. Results are not defined for CPL values 25 through 31. The carry-out level is generated from the bit position of the output specified by CPL. If CPL = 0, the carry-out level is equal to CYF. If CPL = 1, the carry-out level is generated from the rightmost bit of X, Y and CYF. See Section 3.1.13.1, CP (CYF, CPU, CPL).

If CPU (2-bit arithmetic unit control) = 00, the binary sum is produced. See Section 3.1.13.1, CP (CYF, CPU, CPL).

If CPU = 01, the decimal sum is produced by considering the X and Y inputs to be comprised of six 4-bit units. Results are not defined for non-Binary Coded Decimal (BCD) units. CPL must be a multiple of four.

If CPU = 10 or 11, the sum is undefined.

3.2.2 DIFF

DIFF is a pseudo register equal to the difference of the X, Y, and CYF Registers (X - Y - CYF). Zero bits appear in the more significant bit positions of the 24-bit result when the length as determined by CPL is less than 24. Results are not defined for CPL values 25 through 31. The borrow-out level, generated from the static comparison of all 24-bits of X & Y, is true if X < Y or if X = Y and CYF is true.

If CPU = 00, the binary difference is produced.

If CPU = 01, the decimal difference is produced by considering the X & Y inputs to be comprised of six 4-bit units. Results are not defined for non-BCD units. CPL must be a multiple of four.

If CPU = 10 or 11, the difference is undefined.

A negative result is in 2's complement form in the binary case and in 10's complement form in the decimal case.

3.2.3 XANY, XORY, XEOY

XANY (X and Y), XORY (X or Y), and XEOY (X exclusive-or Y) are 24-bit pseudo registers that hold the results of the specified logical functions. Zero bits are produced in the more significant bit positions of the 24-bit result when the length as determined by CPL is less than 24. Results are not defined for CPL values 25 through 31.

3.2.4 CMPX, CMPY

CMPX (complement of X) and CMPY (complement of Y) are 24-bit pseudo registers that hold the 1's complement of the specified registers. Zero bits are produced in the more significant positions of the 24-bit result when the length as determined by CPL is less than 24. CPL values 25 through 31 have undefined results.

3.2.5 MSKX, MSKY

MSKX, (mask of X) and MSKY (mask of Y) are 24-bit pseudo registers that hold the mask of the appropriate register (X or Y). Beginning with LSB of X or Y, the number of bit positions included in the mask is determined by the value of CPL. Zero bits are produced in the more significant bit positions of the 24-bit result when the length as determined by CPL is less that 24. Results are not defined for CPL values 25 through 31.

3.2.6 BICN

BICN (binary conditions) is a 4-bit pseudo register that holds the following binary conditions, considered as a 4-bit group, and addressable as a source only:

NOTE: CYF is also addressed by the SET CYF M-Instruction as well as being available in the (8-bit) group addressed as CP.

```
          -------------------------
BICN:     | LSUY | CYF | CYD | CYL |
          -------------------------
             |      |     |     |
             |      |     |     Carry Out Level
             |      |     Borrow-Out Level
             |      Carry Flip-Flop
             Least Significant Unit of Y
```

The carry-out level is a function of X, Y, CPL, and CPU. See Section 3.2.1, SUM. The borrow-out level is a function of X, Y, and CYF. See Section 3.2.2, DIFF.

LSUY is true if the least significant unit of Y is equal to 1 and CPU = 00, or if the least significant unit of Y is equal to 1001 and CPU = 00.

3.2.7 XYCN

XYCN (XY condition) is a 4-bit pseudo register that holds the following relational conditions, considered as a 4-bit group and addressable as a source only:

```
          -------------------------
XYCN:     | MSBX | X=Y | X<Y | X>Y |
          -------------------------
```

MSBX is true if the bit in X referenced by CPL is 1. CPL = 1 references the rightmost bit of X while CPL = 24 references the leftmost bit. MSBX = 0 if CPL = 0.

The relational results are based on the binary value of all 24-bits of X and Y.

3.2.8 XYST

XYST (XY states) is a 4-bit pseudo register that holds the following relational conditions, considered as a 4-bit group and addressable as a source only:

XYST:	LSUX	INT	Y neq 0	X neq 0

LSUX is true when the least significant unit of X is equal to 1 and CPU = 00, or when the least significant unit of X is equal to 1001 and CPU = is not equal to 00.

The relational results are based on the binary value of all 24 bits of X or Y.

INT is true if any of the following conditions as reflected in (INCN), CC, and CD are true. (See Section 3.2.9, INCN and 3.1.13.2, CA, CB, CC, CD.)

1. Timer Interrupt CC(1)
2. Bus I/O Interrupt CC(2)
3. Control Panel Interrupt CC(3)
4. Memory Parity Error Interrupt CD(0)

Software #'s

3.2.9 INCN, PERR

INCN (interrupt conditions) and PERR (Parity error) are 4-bit pseudo registers which are not physically present in the S-Processors. When either is addressed as a source, a binary value of zero is yielded; when addressed as destinations, the data is lost.

3.3 4-BIT FUNCTION BOX

The 4-bit function box (4-bit arithmetic and combinatorial section of the processor) can accept, as one of its inputs, the contents of any of the following 4-bit registers and pseudo registers.

TA	TB	TC	TD	TE	TF
LA	LB	LC	LD	LE	LF
FU	FT	FLC	FLD	FLE	FLF
CA	CB	CC	CD		
CA	CB	CC	CD		

Outputs include the result of most of the commonly used functions between two operands; for example: Set, And, Or, Exclusive-Or, Binary Sum and Difference (both modulo 16). Outputs are directed back to the source register.

The sum and difference output can be tested for overflow and underflow respectively and, based on the test, a skip of one instruction can be made.

The 4-bit function box also provides for the selective testing of one of the bits of a four-bit group and relative branching based on the result of the test. A skip of one instruction based on the result of testing on a combination of up to four bits in the group is also provided.

BICN, XYCN, XYST, and FLCN are not actually registers but can be sourced as if they were. They can be changed only as a result of changing the condition which they reflect.

3.4 M-INSTRUCTIONS

3.4.1 REGISTER MOVE

FORMAT:

OP CODE 0001	SOURCE REGISTER GROUP # 0 -15	SOURCE REGISTER GROUP # 0...3	DESTINATION REGISTER SELECT # 0...3	DESTINATION REGISTER GROUP # 0...15

Move the contents of the source register to the destination register. If the source register is smaller than the destination register, data are right justified with left (most significant) zero bits supplied. If the source register is larger than the destination register, data are truncated from the left.

The contents of the source register are unchanged unless it is also the destination register.

Exceptions:

1. When M is used as a destination register in run or step mode, the operation is changed to a bit-OR which modifies the next micro-operation. It does not modify the instruction as stored in the memory. In tape mode, no bit-OR takes place.

2. CPU, WRIT, READ, and CMND are excluded as source registers.

3. BICN, FLCN, XYCN, XYST, INCN, READ, WRIT, SUM, CMPX, CMPY, XANY, XEQY, MSKX, MSKY, XORY, DIFF, MAXS, MAXM, and U are excluded as destination registers.

4. When DATA, SUM, or DIFF is designated as a source, CMND and DATA are excluded as destinations.

5. U is excluded as a source in STEP mode but is permitted as a source in RUN or TAPE mode. When U is used as a source, TAS, MAR(A) and M registers are excluded as destinations.

6. When A, M, CP, or DATA is designated as a source, all 4-bit registers are prohibited as destinations.

3.4.2 SCRATCHPAD MOVE

FORMAT:

OP CODE	RGSTR GRP #	RGSTR SLCT#	DIRECTION	SCRTCHPD WORD	SCRTCHPD WORD
0010	0...15	0...3	0 TO SCRTCHPD 1 FROM SCRTCHPD	0-LFT WRD 1-RT WRD	ADDRESS 0...15

Move the contents of the register (SCRATCHPAD) to SCRATCHPAD (register). If the move is between registers of unequal lengths, the data is right justified with left (most significant) zero bits supplied or with data truncated from the left, whichever is appropriate.

The contents of the source register are unchanged.

Exceptions:

1. When M is used as a destination register, the operation is changed to a bit-OR which modifies the next micro-operation. It does not modify the instruction as stored in the memory.

2. CPU, WRIT, READ, and CMND are excluded as source registers.

3. BICN, FLCN, XYCN, XYST, INCN, READ, WRIT, SUM, CMPX, CMPY, XANY, XORY, XEOY, MSKX, MSKY, DIFF, MAXS, MAXM and U are excluded as destination registers.

4. M as a source results in a transfer of 24 zeroes.

5. U is excluded as a source in STEP mode.

3.4.3 XCH F WITH DOUBLEPAD WORD

	OP CODE 0000 0111	DESTINATION 48-BIT SCRATCHPAD WORD 0...15	SOURCE 48-BIT SCRATCHPAD WORD 0...15
FORMAT:			

Move the contents of the FA and FB registers to a holding register. Move the contents of the left and right word of the source scratchpad word to the FA and FB Register respectively. Move the contents of the holding register to the left and right word of the destination scratchpad word.

3.4.4 MOVE 8-BIT LITERAL

	OP CODE 1000	DESTINATION REGISTER GROUP # 0...15	LITERAL 0...255
FORMAT:			

Move the 8-bit literal given in the instruction to the destination register. If the move is between registers of unequal lengths, the data is right justified with left (most significant) zero bits supplied. The register select number is assumed to be 2.

Exceptions:

READ and WRIT are excluded as destination registers.

3.4.5 MOVE 24-BIT LITERAL

	OP CODE 1001	DESTINATION REGISTER GROUP # 0...15	24-BIT LITERAL 0...MAX
FORMAT:			

Move the 24-bit literal given in the instruction to the destination register. If the move is between registers of unequal lengths, the literal is truncated from the left. The register select number is assumed to be 2.

Exceptions:

M, CP, READ, and WRIT are excluded as destination registers.

3.4.6 READ/WRITE MEMORY

	OP CODE 0111	DIRECTION 0 to RGSTR 1 TO MEMORY	COUNT VRNTS 0...7	RGSTR # 00 = X 01 = Y 10 = T 11 = L	FIELD DIRECTION 0 - POSITIVE 1 - NEGATIVE	MEMORY FIELD LENGTH 0...26
FORMAT:						

Move the register's (memory's) contents to the memory (register). If the value of the memory field length is less than 24, the data from memory is right justified with left (most significant) zero bits supplied while the data from the register is truncated from the left.

The contents of the source is unchanged.

Register FA contains the bit address of the memory field while the memory field direction sign and memory field length is given in the instruction.

If the value of the memory field length as given in the instruction is zero, the value in CPL is used.

Memory field length values (or CPL values if MFL = 0) of 25 and 26 are truncated to the value 24. When used on a write operation, the value 25 and 26 cause odd and even parity respectively to be written into memory regardless of the parity of tne read data.

For a description of the count variants, see Section 3.4.7, COUNT FA/FL.

3.4.7 COUNT FA/FL

FORMAT:

OP CODE	COUNT VARIANTS	LITERAL
0000 0110	0...7	0...31

Increment (decrement) binarily the designated register(s) by the value of the literal contained in the instruction or by the value of CPL if the value of the literal is zero.

Neither overflow nor underflow of FA is detected. The value of FA may go through its maximum value or its minimum value and wrap around.

Overflow of FL is not detected. The value of FL will go through its maximum value and wrap around. Underflow of FL is detected and will not wrap around. The value of zero is left in FL.

Literal values (or CPL values if LIT = 0) of 25 through 31 are truncated to the value 24.

Count variants are as follows:

V = 000 No count
    001 Count FA Up
    010 Count FL Up
    011 Count FA Up and FL Down
    100 Count FA Down and FL Up
    101 Count FA Down
    110 Count FL Down
    111 Count FA Down and FL Down

3.4.8 SCRATCHPAD RELATE FA

FORMAT:	OP CODE	RESERVED	SIGN OFF SPAD WORD	LEFT SCRATCHPAD WORD ADDRESS
	0000 1000	000	0-POSITIVE 1-NEGATIVE	0...15

Replace the contents of the FA Register by the binary sum of the FA Register and specified scratchpad register.

Neither overflow nor underflow of FA is detected. The value of FA may go through its maximum value or its minimum value and wrap around.

3.4.9 EXTRACT FROM REGISTER T

FORMAT:	OP CODE	ROTATE BIT COUNT	DESTINATION REGISTER	EXTRACT BIT COUNT
	1011	0...24	00 - X 01 - Y 10 - T 11 - L	0...24

Rotate Register T left by the number of bits specified and then extract from the right the number of bits specified. Move this result to the destination register supplying left most (most significant) zero bits if the extract

count is less than 24.

The contents of the source register are unchanged unless it is also the destination register.

A rotate value of 24 is equivalent to 0.

3.4.10 SHIFT/ROTATE REGISTER T LEFT

FORMAT:

OP CODE 1010	DSTNATN REGISTR GROUP # 0...15	DSTNATN REGISTR SELECT# 0...2	S/R VARIANT 0 - SHIFT 1 - ROTATE	S/R BIT COUNT 0...24

SHIFT (ROTATE) Register T left by the number of bits specified and then move the 24-bit result to the destination register. If the move is between registers of equal lengths, the data is right justified with data truncated from the left.

The contents of the source register are unchanged unless the source register is also the destination register.

Zero fill on the right and truncation on the left occurs for the SHIFT operation.

If the value of the SHIFT/ROTATE count as given in the instruction is zero, the value given in CPL is used.

Exceptions:

1. When M is used as a destination register, the operation is changed to a bit-or which modifies the next micro-operation. It does not modify the instruction as stored in the memory.

2. BICN, FLCN, XYCN, XYST, INCN, READ, and WRIT.

3.4.11 SHIFT/ROTATE REGISTER X/Y LEFT/RIGHT

FORMAT:

OP CODE	S/R VARIANT	L/R VARIANT	X/Y VARIANT	S/R BIT COUNT
0000 0100	0-SHIFT 1-ROTATE	0-LEFT 1-RIGHT	0-X REG 1-Y REG	0...24

SHIFT (ROTATE) Register X or Register Y left (right) by the number of bits specified.

Zero fill on the right and truncation on the left occurs for the left shift. Zero fill on the left and truncation on the right occurs for the right shift.

3.4.12 SHIFT REGISTERS XY LEFT/RIGHT

FORMAT:

OP CODE	SHIFT VARIANT=	L/R VARIANT	SHIFT BIT COUNT=
0000 0101	0	0-LEFT 1-RIGHT	000001

ONLY

SHIFT the concatenated X and Y registers left (right) by one bit. The register X is the leftmost (more significant) half of the concatenated 48-bit XY register.

ONLY

Zero fill on the right and truncation on the left occurs for the left shift. Zero fill on the left and truncation on the right occurs for the right shift.

If the value of the shift count as given in the instruction is not 1, an undefined result will occur.

COP OUT

3.4.13 NORMALIZE X

FORMAT:

OP CODE	
0000 0000 0000 0011	

SHIFT the X Register left while counting FL down, until FL=0 or until the bit in X referenced by CPL = 1. Zeros are shifted into the rightmost end of X.

CPL = 1 references the rightmost bit of X while CPL = 24 references the leftmost bit of X. CPL is undefined.

3.4.14 CALL

FORMAT:

OP CODE	DISPLACEMENT SIGN	DISPLACEMENT VALUE
111	0=POSITIVE 1=NEGATIVE	0...4095

Push the address of the next in-line micro-instruction into the A Stack and then fetch the next micro-instruction from the location obtained by adding the signed displacement value given in the instruction to the address of the next in-line micro-instruction.

A displacement value indicates the number of 16-bit words.

Note: When the A Address is stored in the A Stack, it is multiplied by 16 and stored as a bit address.

Note: Exit is accomplished by employing the REGISTER MOVE instruction with the TAS as the source register and MAR(A) as the destination register.

3.4.15 BRANCH

FORMAT:

OP CODE	DISPLACEMENT SIGN	DISPLACEMENT VALUE
110	0=POSITIVE 1=NEGATIVE	0...4095

Fetch the next micro-instruction from the location obtained by adding the signed displacement value given in the instruction to the address of the micro-instruction next-in-line.

A displacement value indicates the number of 16-bit words.

3.4.16 BIAS

FORMAT:

OP CODE	VARIANTS	TEST CPL NEQ 0 FLAG
0000 0000 0011	0...7	0 - NO TEST 1 - TEST CPL RESULT

Set CPU to the value 1 if the value of FU is 4 or 8 and to 0 otherwise, unless V = 2. If V = 2, CPU is undefined.

Set the value of CPL to the smallest of the values denoted on each line in the following table.

V	VALUES
0	FU
1	24 and FL
2	24 and SFL
3	24 and FL and SFL
4	CPL
5	24 and CPL and FL
6	CPL
7	not defined

If test flag equals 1 and final value of CPL is not zero, the next 16-bit micro-instruction is skipped.

3.4.17 SET CYF

FORMAT:

OP CODE	VARIANTS
0000 0000 0110	1,2,4,8

Set the carry flip-flop as specified by the variants.

V = 1 Set CYF to 0
2 Set CYF to 1
4 Set CYF to CYL
8 Set CYF to CYD

Note CYD = (X<Y) + (X=Y)CYF.

3.4.18 4-BIT MANIPULATE

FORMAT:

OP CODE	REGISTER GROUP #	REGISTER SELECT #	VARIANTS	LITERAL
0011	0...15	0...1	0...7	0...15

Perform the operation specified by the variants on the designated register.

V = 0 Set the register to the value of the literal.
1 Set the register to the logical And of the register and literal.
2 Set the register to the logical Or of the register and literal.
3 Set the register to the logical Exclusive-Or of the register and literal.
4 Set the register to the binary sum (modulo 16) of the register and literal.
5 Set the register to the binary sum (modulo 16) of the register and literal, and skip the next M-Instruction if a carry is produced.

6 Set the register to the binary difference (modulo 16) of the register and literal.

7 Set the register to the binary difference (modulo 16) of the register and literal, and skip the next M-Instruction if a borrow is produced.

Exception

BICN, FLCN, XYCN, XYST, and CPU, when specified as operand registers, are not changed as a result of this operation. However, the carry and borrow outputs are produced and a skip can result.

3.4.19 BIT TEST BRANCH FALSE

FORMAT:

OP CODE	REG GROUP #	REGISTER SELECT #	REG BIT #	DSPLCMNT SIGN	DSPLCMNT VALUE
0100	0...15	0...1	0...3	0-POS 1-NEG	0...15

Test the designated bit within the specified register and branch relative to the next instruction by the signed displacement value if the bit is zero. If the bit is one, a displacement value of zero is assumed and control passes to the next in-line M-instruction. A displacement value indicates the number of 16-bit words from the next in-line instruction.

See Section 3.1.13.2 for information on the reset of bits in the C Register.

3.4.20 BIT TEST BRANCH TRUE

FORMAT:

OP CODE	RGSTR GROUP #	RGSTR SELECT #	RGSTR BIT #	DSPLCMNT SIGN	DSPLCMNT VALUE
0101	0...15	0...1	0...3	0-POSITIVE 1-NEGATIVE	0...15

Test the designated bit within the specified register and branch relative to the next instruction by the signed

displacement value if the bit is one. If the bit is zero, a displacement value of zero is assumed and control passes to the next in-line M-instruction. A displacement value indicates the number of 16-bit words from the next in-line instruction.

See Section 3.1.14.2, CA, CB, CC, CD, for information on the reset of bits in the C Register.

3.4.21 SKIP WHEN

	OP CODE	REGISTER GROUP #	REGISTER SELECT #	VARIANTS	MASK
FORMAT:	0110	0...15	0...1	0...7	0...15

Test only the bits in the register that are referenced by the "1" bits in the mask, ignoring all others unless V = 2 or V = 6. If so, compare all bits for an equal condition. Then perform the action as specified below.

V = 0 If any of the referenced bits is a "1", skip the next M-instruction.

1 If all of the referenced bits are "1", skip the next M-instruction.

2 If the register is equal to the mask, skip the next M-instruction.

3 Same as V = 1, but also clear the referenced bits to zero without affecting the non-referenced bits.

4 If any of the referenced bits is a "1", do not skip the next M-instruction.

5 If all of the referenced bits are "1" do not skip the next M-instruction.

6 If the register is equal to the mask, do not skip the next instruction.

7 Same as V = 4, but also clear the referenced bits to zero without affecting the non-referenced bits.

Note: If the mask equals 0000, the "ANY" result is false. The skip is not made for V = 0 and is made for V = 4. If the mask equals 0000, the "ALL" result is true. The skip is made for V = 1 and V = 3 and is not made for V = 5 and V = 7.

Exceptions:

1. BICN, FLCN, XYCN, and XYST cannot be cleared with V = 3 or 7. However, they can be tested.

2. See Section 3.1.13.2 for information on the reset of bits in the C Register.

3.4.22 CASSETTE CONTROL

FORMAT:	OP CODE	VARIANTS	RESERVED
	0000 0000 0010	0...7	FLAG BIT 0...1

Perform the indicated operation on the tape cassette.

V = 0 Start Tape
1 Stop Tape (The processor also halts if it is in TAPE mode.)
2 Stop Tape if X ~~not~~ Y (The processor also halts regardless of mode.)
3 Reserved
4 Reserved
5 Reserved
6 Reserved
7 Reserved

Note: All Stop Tape variants cause the tape to halt in the next available gap.

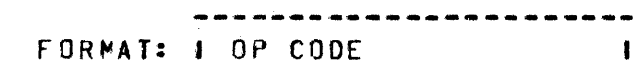
3.4.23 HALT

FORMAT:	OP CODE
	0000 0000 0000 0001

Stop execution of the micro-instructions. The next micro to be executed is fetched and stored in the M Register and the MAR(A) register points to the next following micro.

3.4.24 NO OPERATION

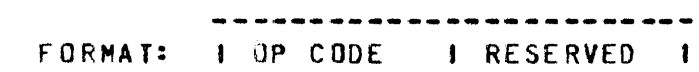
FORMAT:



Skip to the next sequential instruction.

3.4.25 NO OPERATION

FORMAT:



Skip to the next sequential instruction.

3.5 M-INSTRUCTION TIMING

M-instruction	Clocks
REGISTER MOVE	2
(BCD source -- SUM, DIFF, BICN with CPU = 01)	3
(MAR(A) as destination)	4
(BCD source and MAR(A) destination)	5
(U as source)	Many
SCRATCHPAD MOVE	2
(BCD source -- SUM, DIFF, BICN with CPU = 01)	3
(MAR(A) as destination)	4
SWAP F WITH DOUBLEPAD WORD	10
~~MOVE~~ 8-BIT LITERAL	2
(MAR(A) as destination)	4
~~MOVE~~ 24-BIT LITERAL	6
(TAPE mode)	Many
WRITE MEMORY	8
READ MEMORY	8
COUNT FA/FL	4
SCRATCHPAD RELATE FA	4
EXTRACT FROM REGISTER T	3
SHIFT/ROTATE REGISTER T LEFT	3
(MAR(A) as destination)	5
SHIFT/ROTATE REGISTER X/Y LEFT/RIGHT	3
SHIFT REGISTER XY LEFT/RIGHT	6
NORMALIZE	7/bit shifted
(if FL goes to zero)	7/bit shifted plus 2
(if MSBX = 1)	4
CALL	5
BRANCH	4
BIAS	2
(branch taken)	4
SET CYF	2
4-BIT MANIPULATE	2
(skip taken)	4
BIT TEST BRANCH FALSE	2
(branch taken)	4
(BCD source -- BICN with CPU = 01)	3
(BCD source and branch taken)	5

M-instructions	Clocks
BIT TEST BRANCH TRUE	2
(branch taken)	4
(BCD source -- BICN with CPU = 01)	3
(BCD source and branch taken)	5
SKIP WHEN	2
(branch taken)	4
(BCD source -- BICN with CPU = 01)	3
(BCD source and branch taken)	5
CASSETTE CONTROL	2
HALT	2
NOOP	2