

6800 MICROPROCESSOR

Instruction Set Summary

6800 SOFTWARE

CROSS ASSEMBLER, PL/W, SIMULATOR

6800 HARDWARE MODULES

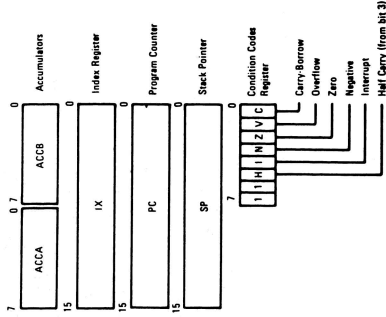
MPU, RAM, ROM, EROM PROGRAMMER, I/O, A/D

MICROCOMPUTER PERIPHERALS

CRT TERMINALS, CASSETTE

CUSTOM HARDWARE/SOFTWARE

HANDS—ON μ P WORKSHOPS



Programming Model Of The Microprocessing Unit



Corp.

902 N. 9th STREET
LAFAYETTE, IND. 47904
(317) 742-6802

ACCUMULATOR AND MEMORY OPERATIONS		IMMED		DIRECT		INDEX		EXTND		INHER		BOOLEAN/ARITHMETIC OPERATION (All register labels refer to contents)							
OPERATIONS	MEMORIC	OP	#	OP	#	OP	#	OP	#	OP	#	5	4	3	2	1	0	H	C
Add	ADDA	88	2	98	3	2	A8	5	2	B8	4	3							
Add	ADDB	C8	2	08	3	2	E8	5	2	F8	4	3							
Add Acmltrs	ABA																		
Add with Carry	ADCA	89	2	99	3	2	A9	5	2	B9	4	3	1B	2	1				
	ADCB	C9	2	09	3	2	E9	5	2	F9	4	3							
And	ANDA	94	2	94	3	2	A4	5	2	B4	4	3							
	ANDB	C4	2	04	3	2	E4	5	2	F4	4	3							
Bit Test	BITA	85	2	95	3	2	A5	5	2	B5	4	3							
	BITB	C5	2	05	3	2	E5	5	2	F5	4	3							
Clear	CLR						6F	7	2	7F	6	3	4F	2	1				
	CLRA												5F	2	1				
	CLRB																		
Compare	CPA	81	2	91	3	2	A1	5	2	B1	4	3							
	CPB	C1	2	01	3	2	E1	5	2	F1	4	3							
Compare Acmltrs Complement, 1's	CBA												11	2	1				
	COM						63	7	2	73	6	3							
	COMA												43	2	1				
	COMB												53	2	1				
Complement, 2's (Negate)	NEG						60	7	2	70	6	3							
	NEGA												40	2	1				
	NEGB												50	2	1				
Decimal Adjust, A	DAA												19	2	1				
Decrement	DEC																		
	DECA						6A	7	2	7A	6	3							
	DECB												4A	2	1				
Exclusive OR	EORA	88	2	98	3	2	A8	5	2	B8	4	3	5A	2	1				
	EORB	C8	2	08	3	2	E8	5	2	F8	4	3							
Increment	INC						6C	7	2	7C	6	3							
	INCA												4C	2	1				
	INCB												5C	2	1				

Converts Binary Add. of 8CD Characters into BCD Format

M-1→M

A-1→A

B-1→B

A→M→A

B→M→B

M+1→M

A+1→A

B+1→B

[illegible]

INDEX REGISTER AND STACK										IMMED		DIRECT		INDEX		EXTND		INHER		BOOLEAN/ARITHMETIC OPERATION									
POINTER OPERATIONS					MNEMONIC					OP	~	#	OP	~	#	OP	~	#	OP	~	#	H	I	N	Z	V	C		
Compare Index Reg					CPX	3	8C	3	3	9C	4	2	AC	6	2	BC	5	3											
Decrement Index Reg					DEX														09	4	1								
Decrement Stack Ptr					DES														34	4	1								
Increment Index Reg					INX														08	4	1								
Increment Stack Ptr					INS														31	4	1								
Load Index Reg					LDX	3	CE	3	3	DE	4	2	EE	6	2	FE	5	3											
Load Stack Ptr					LDS	3	8E	3	3	9E	4	2	AE	6	2	BE	5	3											
Store Index Reg					STX					DF	5	2	EF	7	2	FF	6	3											
Store Stack Ptr					STS					9F	5	2	AF	7	2	BF	6	3											
Indx Reg → Stack Ptr					TXS														35	4	1								
Stack Ptr → Indx Reg					TSX														30	4	1								

JUMP AND BRANCH																						
OPERATIONS					MNEMONIC					RELATIVE		INDEX		EXTND		INHER		BRANCH TEST				
5	4	3	2	1	0																	
H	I	N	Z	V	C																	
						BRA	20	4	2											None		
						BCC	24	4	2											C = 0		
						BCS	25	4	2											C = 1		
						BEQ	27	4	2											Z = 1		
						BGE	2C	4	2											N < V = 0		
						BGT	2E	4	2											Z + (N < V) = 0		
						BHI	22	4	2											C + Z = 0		
						BLE	2F	4	2											Z + (N < V) = 1		
						BLS	23	4	2											C + Z = 1		
						BLT	2D	4	2											N < V = 1		
						BMI	28	4	2											N = 1		
						BNE	26	4	2											Z = 0		
						BVC	28	4	2											V = 0		
						BVS	29	4	2											V = 1		
						BPL	2A	4	2											N = 0		
						BSR	8D	8	2													
						Jump																
						JMP																
						Jump To Subroutine																

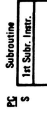
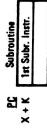
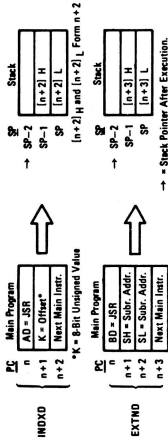
} See Special Operations

OPERATION CODE, NUMERICAL LISTING:

OP	MNEM	MODE	~	#	OP	MNEM	MODE	~	#	OP	MNEM	MODE	~	#	OP	MNEM	MODE	~	#	OP	MNEM	MODE	~	#
01	NOP	INNER	2	1	30	TSX	INNER	4	1	60	NEG	INDXD	7	2	80	SUBA	EXTND	4	3	D7	STAB	DIR	4	2
06	TAP	REL	2	1	31	INS	REL	4	1	63	COM	REL	8	2	81	CMPA	REL	8	2	D8	EORB	REL	4	3
07	TPA	REL	2	1	32	PULA	REL	4	1	64	LSR	IMMED	3	3	82	SBCA	IMMED	3	3	D9	ADCB	REL	4	3
08	INX	REL	4	1	33	PULB	REL	4	1	66	ROR	DIR	3	2	84	ANOA	DIR	3	2	DA	ORAB	REL	4	3
09	DEX	REL	4	1	34	DES	REL	4	1	67	ASR	REL	3	2	85	BITA	REL	3	2	DB	ADDB	REL	4	3
0A	CLV	REL	2	1	35	TXS	REL	4	1	68	ASL	REL	3	2	86	LDAA	REL	3	2	DE	LDX	REL	4	2
0B	SEV	REL	2	1	36	PSHA	REL	4	1	69	ROL	REL	7	2	87	STAA	REL	7	2	DF	STX	DIR	5	2
0C	CLC	REL	2	1	37	PSHB	REL	4	1	6A	DEC	REL	7	2	88	EORA	REL	3	2	E0	SUBB	INDXD	5	2
0D	SEC	REL	2	1	38	RTS	REL	5	1	6C	INC	REL	7	2	89	ADCA	REL	3	2	E1	CMPB	REL	4	3
0E	CLI	REL	2	1	39	RTI	REL	10	1	6D	TST	REL	7	2	8A	ORAA	REL	4	3	E2	SBCB	REL	4	3
0F	SEI	REL	2	1	3E	WAI	REL	9	1	6E	JMP	REL	4	2	8B	ADDA	REL	4	2	E3	ANDB	REL	5	2
10	SBA	REL	2	1	3F	SWI	REL	12	1	6F	CLR	INDXD	7	2	8C	CPX	REL	3	2	E4	ANDB	REL	5	2
11	CBA	REL	2	1	40	NEGA	REL	2	1	70	NEG	EXTND	6	3	8D	JSR	REL	9	3	E5	BITB	REL	5	2
16	TAB	REL	2	1	43	COMA	REL	2	1	73	COM	REL	6	3	8E	LDS	REL	5	3	E6	LDAB	REL	5	2
17	TBA	REL	2	1	44	LSRA	REL	2	1	74	LSR	REL	6	3	8F	STS	EXTND	6	3	E7	STAB	REL	6	2
19	DAA	REL	2	1	46	RORA	REL	2	1	76	ROR	REL	6	3	90	SUBB	IMMED	2	2	E8	EORB	REL	5	2
18	ABA	INNER	2	1	47	ASRA	REL	2	1	77	ASR	REL	6	3	91	ORAA	REL	4	2	E9	ADCB	REL	5	2
20	BRA	REL	4	2	48	ASLA	REL	2	1	78	ASL	REL	6	3	9A	ADCA	REL	4	2	EA	ORAB	REL	5	2
22	BHI	REL	4	2	49	ROLA	REL	2	1	79	ROL	REL	6	3	9C	CPX	REL	4	2	EB	ADDB	REL	5	2
23	BLS	REL	4	2	4A	DECA	REL	2	1	7A	DEC	REL	6	3	9E	LDS	REL	4	2	EE	LDX	REL	6	2
24	BCC	REL	4	2	4C	INCA	REL	2	1	7C	INC	REL	6	3	A2	SBCA	REL	5	2	EF	STX	INDXD	7	2
25	BCS	REL	4	2	4D	TSTA	REL	2	1	7D	TST	REL	6	3	A3	ANDA	REL	5	2	F0	SUBB	EXTND	4	3
26	BNE	REL	4	2	4F	CLRA	REL	2	1	7E	JMP	REL	3	3	A5	BITA	REL	5	2	F1	CMPB	REL	4	3
27	BEQ	REL	4	2	50	NEGB	REL	2	1	7F	CLR	EXTND	6	3	A6	LDAA	REL	5	2	F2	SBCB	REL	4	3
28	BVC	REL	4	2	53	COMB	REL	2	1	80	SUBA	IMMED	2	2	A7	STAA	REL	6	2	F4	ANDB	REL	4	3
29	BVS	REL	4	2	54	LSRB	REL	2	1	81	CMPA	REL	2	2	A8	EORA	REL	5	2	F5	BITB	REL	4	3
2A	BPL	REL	4	2	56	RORB	REL	2	1	82	SBCA	REL	2	2	CE	LDX	IMMED	3	3	F6	LDAB	REL	4	3
2B	BMI	REL	4	2	57	ASRB	REL	2	1	84	ANDA	REL	2	2	DE	SUBB	DIR	3	2	F7	STAB	REL	5	3
2C	BGE	REL	4	2	58	ASLB	REL	2	1	85	BITA	REL	2	2	D1	CMPB	REL	5	2	F8	EORB	REL	4	3
2D	BLT	REL	4	2	59	ROLB	REL	2	1	86	LDAA	REL	2	2	D2	SBCB	REL	6	2	F9	ADCB	REL	4	3
2E	BGT	REL	4	2	5A	DECB	REL	2	1	88	EORA	REL	2	2	D4	ANDB	REL	8	2	FA	ORAB	REL	4	3
2F	BLE	REL	4	2	5C	INCB	REL	2	1	89	ADCA	REL	2	2	D5	BITB	REL	6	2	F8	ADDB	REL	4	3
					5D	SCB	REL	2	1	8A	ORAA	REL	4	2	AF	STS	INDXD	7	2	FE	LDX	REL	5	3
					5F	CLRB	INNER	2	1	8B	ADDA	IMMED	2	2						FF	STX	EXTND	8	3

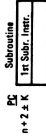
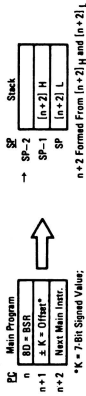
SPECIAL OPERATIONS

JSR, JUMP TO SUBROUTINE:



(S Formed From SH and SL)

BSR, BRANCH TO SUBROUTINE:



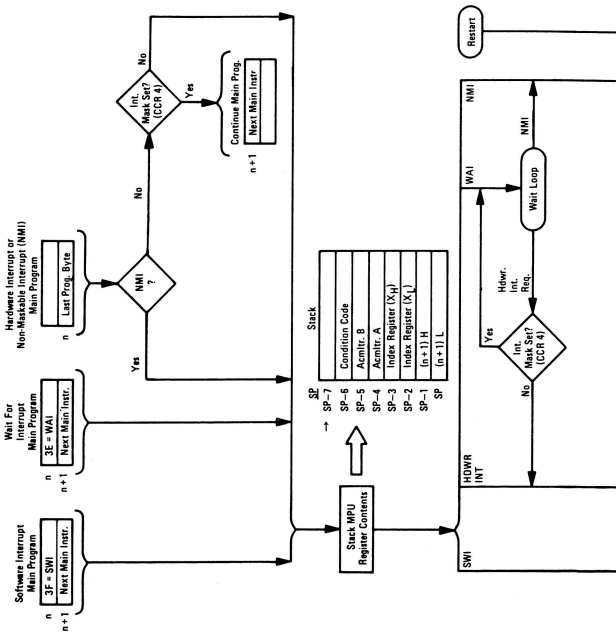
JMP, JUMP:

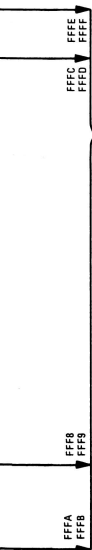


RTS, RETURN FROM SUBROUTINE:



INTERRUPTS:

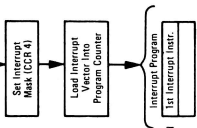




Interrupt Memory Assignment¹

FFF8	MS
Constant, Hdware	LS
FFF9	MS
Constant, Hdware	LS
FFFA	MS
Software	LS
FFFB	MS
Software	LS
FFFC	MS
Non-Maskable Int.	LS
FFFD	MS
Non-Maskable Int.	LS
FFFE	MS
Restart	LS
FFFF	MS
Restart	LS

First Instr.
Addr. Formed
By Fetching
2-Bytes From
Per. Mem.
Assign.

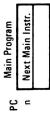
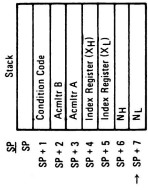


NOTE: MS = Most Significant Address Byte;

LS = Least Significant Address Byte;

1: Memory location responding to the indicated addresses should be reserved for interrupt vectors

RTI, RETURN FROM INTERRUPT:



HEXADECIMAL AND DECIMAL CONVERSION

HOW TO USE THE TABLES

CONVERSION TO DECIMAL: Find the decimal weights for corresponding hexadecimal characters beginning with the least significant character. The sum of the decimal weight is the decimal value of the hexadecimal number.

CONVERSION TO HEXADECIMAL: Find the highest decimal value in the table which is lower than or equal to the decimal number to be converted. The corresponding hexadecimal character is the most significant character. Subtract the decimal value found from the decimal number to be converted. With the difference, repeat the process to find subsequent hexadecimal characters.

HEXADECIMAL AND DECIMAL CONVERSION												
15		BYTE		8	7	BYTE		0				
15		CHAR	12	11	CHAR	8	7	CHAR	4	3	CHAR	0
HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	HEX	DEC	DEC
0	0	0	0	0	0	0	0	0	0	0	0	0
1	4	095	1	256	1	16	1	1	1	1	1	1
2	8	192	2	512	2	32	2	32	2	2	2	2
3	12	288	3	768	3	48	3	48	3	3	3	3
4	16	384	4	1	024	4	64	4	64	4	4	4
5	20	480	5	1	280	5	80	5	80	5	5	5
6	24	576	6	1	536	6	96	6	96	6	6	6
7	28	672	7	1	792	7	112	7	112	7	7	7
8	32	768	8	2	048	8	128	8	128	8	8	8
9	36	864	9	2	304	9	144	9	144	9	9	9
A	40	960	A	2	560	A	160	A	160	A	10	10
B	45	056	B	2	816	B	176	B	176	B	11	11
C	49	152	C	3	072	C	192	C	192	C	12	12
D	53	248	D	3	328	D	208	D	208	D	13	13
E	57	344	E	3	584	E	224	E	224	E	14	14
F	61	440	F	3	840	F	240	F	240	F	15	15

POWERS OF TWO		2 ⁿ		n	
1	0	1	0	1	0
2	1	2	1	2	1
4	2	4	2	4	2
8	3	8	3	8	3
16	4	16	4	16	4
32	5	32	5	32	5
64	6	64	6	64	6
128	7	128	7	128	7
256	8	256	8	256	8
512	9	512	9	512	9
1,024	10	1,024	10	1,024	10
2,048	11	2,048	11	2,048	11
4,096	12	4,096	12	4,096	12
8,192	13	8,192	13	8,192	13
16,384	14	16,384	14	16,384	14
32,768	15	32,768	15	32,768	15
65,536	16	65,536	16	65,536	16
131,072	17	131,072	17	131,072	17
262,144	18	262,144	18	262,144	18
524,288	19	524,288	19	524,288	19
1,048,576	20	1,048,576	20	1,048,576	20

ASCII CHARACTER SET (7-BIT CODE)									
M.S. CHAR	0	1	2	3	4	5	6	7	
L.S. CHAR	000	001	010	011	100	101	110	111	
0	NUL	DLE	SP	␣	@	P	\	p	
0000									
1	SOH	DC1	!	1	A	Q	a	q	
0001									
2	STX	DC2	"	2	B	R	b	r	
0010									
3	ETX	DC3	#	3	C	S	c	s	
0011									
4	EOT	DC4	\$	4	D	T	d	t	
0100									
5	ENQ	NAK	%	5	E	U	e	u	
0101									
6	ACK	SYN	&	6	F	V	f	v	
0110									
7	BEL	ETB	.	7	G	W	g	w	
0111									
8	BS	CAN	(	8	H	X	h	x	
1000									
9	HT	EM	)	9	I	Y	i	y	
1001									
A	LF	SUB	*	:	J	Z	j	z	
1010									
B	VT	ESC	+	;	K	[	k		
1011									
C	FF	FS	,	<	L	\	l	l	
1100									
D	CR	GS	-	=	M	]	m		
1101									
E	SO	RS	.	>	N	^	n	~	
1110									
F	SI	US	/	?	O	_	o		
1111									DEL