1802 Membership Card Summary

Register Summary

D	8 bit	D register (Accumulator)	DF	1 bit	Data flag (ALU carry)
Rn	16 bit	1 of 16 registers (n=0-F)	P	4 bit	Program register select
X	4 bit	Data register select	T	8 bit	Saved X,P after interrupt
IE	1 bit	Interrupt Enable	Q	1 bit	Output flip-flop

Interrupt: Finish executing current instruction, then save X and P in T, then set P=1, X=2 and IE=0 (Inhibit). Execution continues at M(R1).

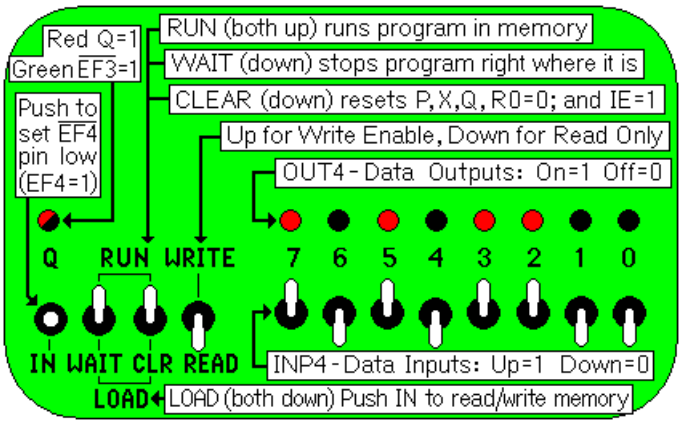
DMA: Finish executing current instruction, then use R0 as memory address for data transfer. DMA-IN writes data from bus to memory, or DMA-OUT reads data from memory to Bus. Then increment R0.

If simultaneous, DMA-IN has highest priority, then DMA-OUT, then Interrupt.

EF1-4: External 1-bit flags that can be tested by Branch instructions.

Q: External 1-bit output that can be set, reset, and tested by Branch and Skip instructions.

http://www.sunrise-ev.com/1802.htm



Register Operations

1n	INC	Rn	Rn=Rn+1
2n	DEC	Rn	Rn=Rn−1
60	IRX	Rx	Rx=Rx+1
8n	GLO	Rn	D=Rn(low8)
9n	GHI	Rn	D=Rn(high8)
An	PLO	Rn	Rn(low8)=D
Bn	PHI	Rn	Rn(high8)=D

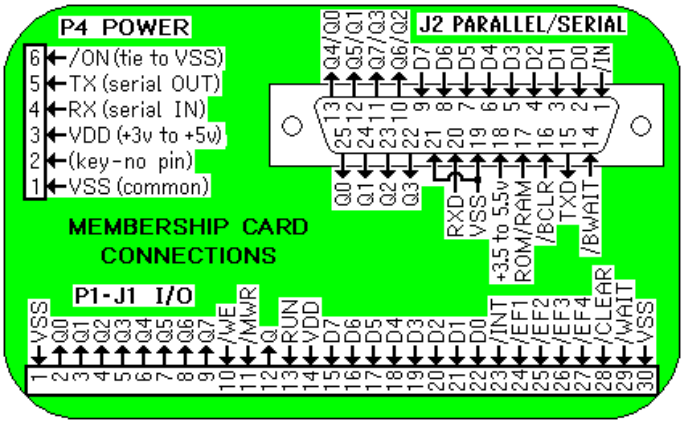
Memory Load / Store

0n	LDN	Rn	D=M(Rn) for n not 0
4n	LDA	Rn	D=M(Rn), then inc Rn
F0	LDX		D=M(Rx)
72	LDXA		D=M(Rx), then inc Rx
F8 nn	LDI	nn	D=M(Rp)
5n	STR	Rn	M(Rn)=D
73	STXD		M(Rx)=D, then dec Rx

Short Branch Instructions (nn is low byte of address on same page)

30 nn	BR	nn	branch to nn	38 nn	NBR	nn	no branch (skip nn)
31 nn	BQ	nn	branch if Q=1	39 nn	BNQ	nn	branch if Q=0
32 nn	BZ	nn	branch if D=0	3A nn	BNZ	nn	branch if D not 0
33 nn	BDF	nn	branch if DF=1,	3B nn	BNF	nn	branch if DF=0,
	or BPZ	nn	br. if positive or 0,		or BM	nn	branch if minus,
	or BGE	nn	br.if equal or greater		or BL	nn	branch if less than
34 nn	B1	nn	branch if EF1=1	3C nn	BN1	nn	branch if EF1=0
35 nn	B2	nn	branch if EF2=1	3D nn	BN2	nn	branch if EF2=0
36 nn	B3	nn	branch if EF3=1	3E nn	BN3	nn	branch if EF3=0
37 nn	B4	nn	branch if EF4=1	3F nn	BN4	nn	branch if EF4=0

name† dots=bus cycles opcode† †2nd byte †description



Long Branch & Skip (hhll=address high byte, low byte)

C0 hhll	LBR	hhll	branch to hhll	C8 hhll	NLBR	hhll	no br.(skip hhll)
C1 hhll	LBQ	hhll	branch if Q=1	C9 hhll	LBNQ	hhll	branch if Q=0
C2 hhll	LBZ	hhll	branch if D=0	CA hhll	LBNZ	hhll	branch if D not 0
C3 hhll	LBDF	hhll	branch if DF=1	CB hhll	LBNF	hhll	branch if DF=0
C4	NOP		no operation	CC	LSIE		skip 2 if IE=1
C5	LSNQ		skip 2 if Q=0	CD	LSQ		skip 2 if Q=1
C6	LSNZ		skip 2 if D not 0	CE	LSZ		skip 2 if D=0
C7	LSNF		skip 2 if DF=0	CF	LSDF		skip 2 if DF=1

Logic, Arithmetic, and Shift (result in D, carry/borrow/shift in DF)

F1	OR	D=D or M(Rx)	F9 nn	ORI	nn D=D or nn
F2	AND	D=D and M(Rx)	FA nn	ANI	nn D=D and nn
F3	XOR	D=D xor M(Rx)	FB nn	XRI	nn D=D xor nn
F4	ADD	D=D+M(Rx)	FC nn	ADI	nn D=D+nn
74	ADC	D=D+M(Rx)+DF	7C nn	ADCI	nn D=D+nn+DF
F5	SD	D=M(Rx)−D	FD nn	SDI	nn D=nn−D
75	SDB	D=M(Rx)−D−not DF	7D nn	SDBI	nn D=nn−D−not DF
F7	SM	D=D−M(Rx)	FF nn	SMI	nn D=D−nn
77	SMB	D=D−M(Rx)−not DF	7F nn	SMBI	nn D=D−nn−not DF

only add, subtract, and shift instructions change DF

Hex	Binary	Dec
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

Hex-Binary-Decimal Conversion

MSD / LSD	0x 000	1x 001	2x 010	3x 011	4x 100	5x 101	6x 110	7x 111
x0 0000	NUL	DLE	(sp)	0	@	P	`	p
x1 0001	SOH	DC1	!	1	A	Q	a	q
x2 0010	STX	DC2	"	2	B	R	b	r
x3 0011	ETX	DC3	#	3	C	S	c	s
x4 0100	EOT	DC4	$	4	D	T	d	t
x5 0101	ENQ	NAK	%	5	E	U	e	u
x6 0110	ACK	SYN	&	6	F	V	f	v
x7 0111	BEL	ETB	'	7	G	W	g	w
x8 1000	BS	CAN	(	8	H	X	h	x
x9 1001	HT	EM	)	9	I	Y	i	y
xA 1010	LF	SUB	*	:	J	Z	j	z
xB 1011	VT	ESC	+	;	K	[	k	{
xC 1100	FF	FS	,	<	L	\	l	|
xD 1101	CR	GS	-	=	M	]	m	}
xE 1110	SO	RS	.	>	N	^	n	~
xF 1111	SI	US	/	?	O	_	o	DEL

Hex-Binary-ASCII Conversion

Logic, Arithmetic, and Shift Instructions (continued)

F6	SHR	shift D right; D(msb)=0, DF=D(lsb)	76 SHRC	shift D right with carry; or RSHR D(msb)=DF, DF=D(lsb)
FE	SHL	shift D left; D(lsb)=0, DF=D(msb)	7E SHLC	shift D left with carry; or RSHL D(lsb)=DF, DF=D(msb)

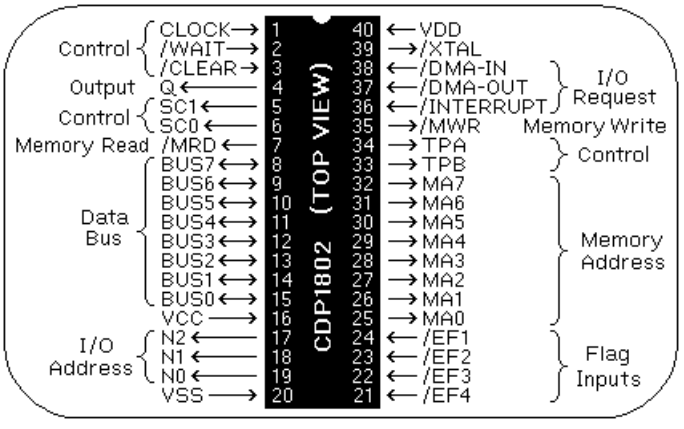
Control Instructions

00	IDL	wait for INT or DMA	78 SAV	M(Rx)=T
68		reserved for 1804/5/6	7A REQ	reset Q=0
70	RET	X,P=M(Rx); inc Rx; IE=1	7B SEQ	set Q=1
71	DIS	X,P=M(Rx); inc Rx; IE=0	Dn SEP	n set P=n
79	MARK	T=X,P; M(R2)=X,P; X=P; dec R2	En SEX	n set X=n

Input/Output Instructions (with N output on pins N0, N1, and N2)

61	OUT1	bus=M(Rx), N=1; inc Rx	69	INP1	M(Rx)=D=bus, N=1
62	OUT2	bus=M(Rx), N=2; inc Rx	6A	INP2	M(Rx)=D=bus, N=2
63	OUT3	bus=M(Rx), N=3; inc Rx	6B	INP3	M(Rx)=D=bus, N=3
64	OUT4	bus=M(Rx), N=4; inc Rx	6C	INP4	M(Rx)=D=bus, N=4
65	OUT5	bus=M(Rx), N=5; inc Rx	6D	INP5	M(Rx)=D=bus, N=5
66	OUT6	bus=M(Rx), N=6; inc Rx	6E	INP6	M(Rx)=D=bus, N=6
67	OUT7	bus=M(Rx), N=7; inc Rx	6F	INP7	M(Rx)=D=bus, N=7

Add: DF=1 if carry. Subtract: DF=0 if borrow (negative).



Membership Card Summary: Keep the left half in the manual, in case you need to make a copy later.

Cut out the right half, fold, and put in the Altoids case as a quick reference (and so board won't short to case).