

E863

42-381 30 SHEETS 3 SQUARE
42-382 100 SHEETS 3 SQUARE

NATIONAL

E88B

F8AA

F884

5C	8D60	WRITE	EQU	*	Write a disk page.
5E	C604		BSR	SEEK	Seek the track
60	5A	REWRITE	LDAB	#4	Set max write tries
61	2602		DECB		have we tried 3 times?
63	3F00		BNE	L3	no, try again.
			SWI		yes, abort bad write.
			ØØ		
65	86CØ	L3	LDAA	#\$CØ	Set interface to write
67	8D 3A		BSR	SETUP	
69	8657		LDAA	#\$57	Write sector
6B	8D40		BSR	DOIT	
6D	81FD		CMPA	#\$FD	Was the write good?
6F	26EF		BNE	REWRITE	no, try again.
71	B6FFA3		LDAA	\$\$FFA3	Set interface to verify
74	8403		ANDA	#3	
76	ØD2Ø		BSR	SETUP1	
78	8677		LDAA	#\$77	Verify sector
7A	8D31		BSR	DOIT	
7C	81F9		CMPA	#\$F9	Was sector verified?
7E	26EØ		BNE	REWRITE	no, try rewriting.
80	39		RTS		yes, all done

81	ØØØØØØ	READ	EQU	*	Read a disk page.
8A	8D28		BSR	SEEK	Seek the track
86	C604		LDAB	#4	Set max read tries
88	5A	REREAD	DECB		have we tries 3 times?
89	2602		BNE	L4	no, go read
8B	3F00		SWI		yes, abort bad read.
			ØØ		
8D	8640	L4	LDAA	#\$40	set interface to read.
8F	8D0C		BSR	SETUP	
91	8677		LDAA	#\$77	read a sector
93	8D18		BSR	DOIT	
95	81FD		CMPA	#\$FD	Was read good?
97	26EF		BNE	REREAD	no, try again.
99	39		RTS		yes, all done

9A	ØØØØØØ	SETUP	EQU	*	Setup for disk operation
9D	BAFFA3		ORAA	\$\$FFA3	Combine operation and disk se
AØ	B7FFA3	SETUP1	STAA	\$\$FFA3	
A3	FEFFA2		LDX	\$\$FFA2	Clear all done flags
A6	FFFFA2		STX	\$\$FFA2	Clear DMA counter.
A9	39		RTS		return

AA	ØØØØØØ	DOIT	EQU	*	Do operation to disk
AD	B7FFA4		STAA	\$\$FFA4	
ØØ	B6FFA1	L5	LDAA	\$\$FFA1	Wait for operation done.
Ø3	2AEB		BPL	L5	
Ø5	B6FFA4		LDAA	\$\$FFA4	Get status
Ø8	84FD		ANDA	#\$FD	Mask off Data req Bit
ØA	39		RTS		and return
ØB	ØØØØØØ				

F8C5

SEEK EQU *

~~STAB~~ ~~\$FFA7~~ ~~09~~

BE	FFFC98	STX	TMPX	Save logical Sector
CE	0000	LDX	#0	
C4	FFFA0	STX	\$FFA0	Init PIA
C7	CEFFFF 09 DEX	LDX	#\$FFFF	
CA	FFFAZ	STX	\$FFAZ	to outputs
CD	CEZC16	LDX	#\$ZC16	
D0	FFFA0	STX	\$FFA0	
D3	F7FAZ	STAB	\$FFAZ	Set DMA Page
		LDAA	TMPX	Init X as base pointer
D6	B6 FC98	LDAA	TMPX	Set Disk Select
D9	49	ROLA		to top two bits of
DA	49	ROLA		Logical Sector.
DB	49	ROLA		
DC	16	TAB		
DD	8403	ANDA	#3	
DF	B7FFA3	STAA	\$\$\$ \$FFA3	And no DMA
E2	59	ROLB		Set Track # to
E3	C4 70	ANDB	#\$70	track field of
E5	B6 FC99	LDAA	TMPX+1	logical Sector.
E8	44	LSRA		
E9	44	LSRA		
EA	44	LSRA		
EB	44	LSRA		
EC	1B	ABA		
ED	43	COMA		
EE	B7FFA5	STAA	\$\$\$ \$FFA5	
FI	CE0006	LDX	#6	Set Count for Seek tries

42 381 50 SHEETS 5 SQUARE
42 382 100 SHEETS 5 SQUARE
42 383 200 SHEETS 5 SQUARE
42 384 300 SHEETS 5 SQUARE
42 385 400 SHEETS 5 SQUARE
42 386 500 SHEETS 5 SQUARE
42 387 600 SHEETS 5 SQUARE
42 388 700 SHEETS 5 SQUARE
42 389 800 SHEETS 5 SQUARE
42 390 900 SHEETS 5 SQUARE
42 391 1000 SHEETS 5 SQUARE

~~0000~~
F8FF

RETRYS

F7	09	DEX		
F8	2602	BNE	L1	
FA	3F00	SWI	\$00	Trouble with seek
FC	BCFFAZ L1	CPX	\$FFA2	Clear Done flags
FF	B6FFA4 L2	LDAA	\$FFA4	Wait for drive ready
102	2AEB	BPL	L2	
104	B63B	LDAA	#\$3B	Read Track # from disk
106	B7FFA4	STAA	\$FFA4	
109	B6FFA1 L3	LDAA	\$FFA1	Operation Complete?
10C	0000	LDX	0000	Better retry, we haven't read any data
10E	48	LDAA	\$FFA4	Data Ready?
10F	0000 28FA	BVC	L3	no, try again.
111	F6FFA7	LDAB	\$FFA7	yes, grab track number
114	B6FFA1 L4	LDA A	\$FFA1	wait for operation complete
117	2AEB	BPL	L4	
	B6FFA4	LDAA	\$FFA4	
	0000	LDAA	\$FFA4	
	81FB	CMPA	#\$FB	F9
	26D9	BNE	RETRYS	

1110/1011

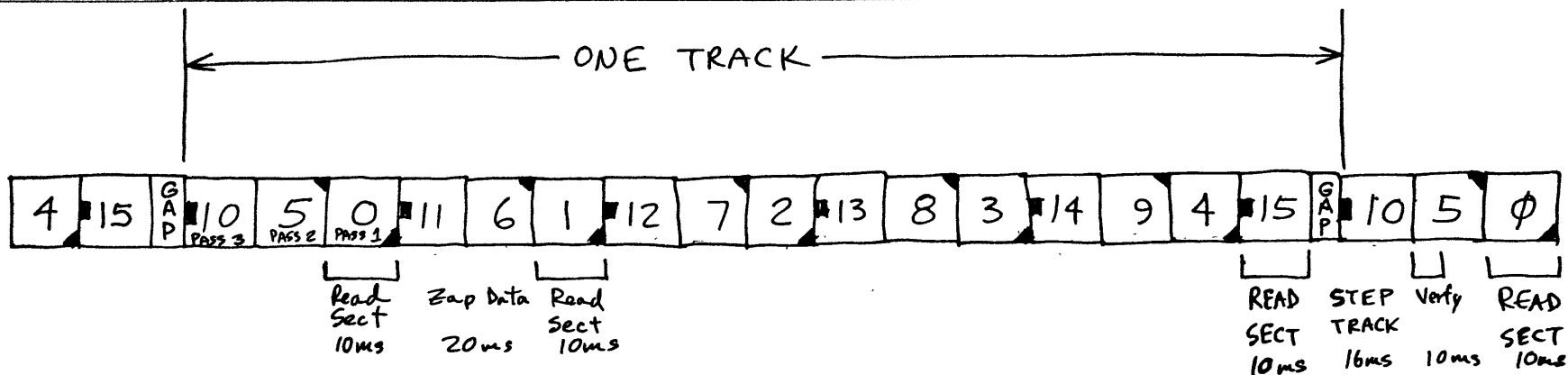
119/	B6FFA5	LDAA	\$FFA5	Are we on the right track
11C	11	CBA		
11D	27L4	BEQ	L6	yes, don't seek
11F	B7FFA7	STAA	\$FFA7	Set desired track
122	F7FFA5	STAB	\$FFA5	Set current track
125	BCFFA2	CPX	\$FFA2	Clear all done flags
*128	86E0	LDAA	#\$E0	Seek track
BDF8BA	8D841	JSR	DOIT	
26	84B9	ANDA	#\$B9	Was seek verified
2E	8199	CMPA	#\$99	
30	26C5	BNE	RETRYS	no, re try
32	B6FC99	L6	TMPX+1	yes, set sector register
35	840F	ANDA	#\$0F	
37	43	COMA		
138	B7FFA6	STAA	\$FFA6	
13B	39	RTS		and return
13C				

50 SQUARE
100 SQUARE
150 SQUARE
200 SQUARE
250 SQUARE
300 SQUARE
350 SQUARE
400 SQUARE
450 SQUARE
500 SQUARE
550 SQUARE
600 SQUARE
650 SQUARE
700 SQUARE
750 SQUARE
800 SQUARE
850 SQUARE
900 SQUARE
950 SQUARE
1000 SQUARE

0010 -
1800 - \$A00



42-381 50 SHEETS 5 SQUARE
42-382 180 SHEETS 5 SQUARE
42-389 200 SHEETS 5 SQUARE



Wave Mate Interlace 3 sector format

INIT FFA2 ← FF
 FFA3 ← FF
 FFA0 ← 2E
 FFA1 ← 16
 RESTORE FFA3 ← 48 ⁴⁸
 FFA4 ← F0

READ FFA2 ← PAGE
 FFA3 ← 40 / 41 Disk
 FFA6 ← SECTOR
 FFA7 ← TRACK
 FFA4 ← E0 Return ~~FFA3~~ 99 & B9 Seek 98
 FFA4 ← 73 Return FD & FD Read

WRITE FFA2 ← PAGE
 FFA3 ← C0 / C1 Disk
 FFA6 ← SECTOR
 FFA7 ← TRACK
 FFA4 ← E0 Return 99 & B9 Seek
 FFA4 ← 53 Return FD & FD Write
 FFA4 ← 73 Return FD & FD Verify

FFA3 ← 0/1 no data
~~FFA3 ← 0/1 no data~~

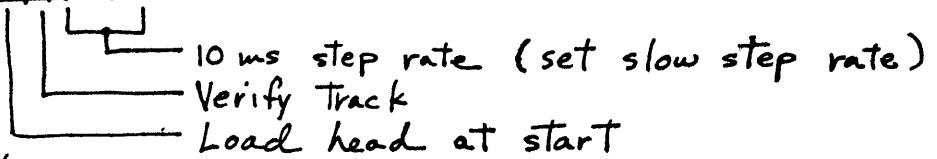
(WRITE ONLY)

F \emptyset

FFA4

1	1	1	1	0	0	0	0
---	---	---	---	---	---	---	---

 RESTORE (seek track zero)

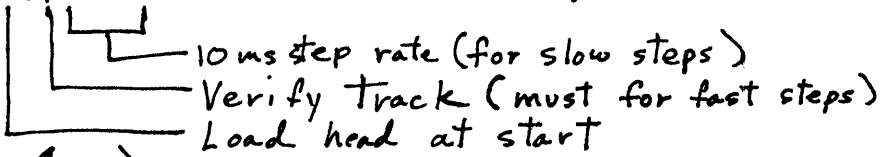


E \emptyset

FFA4

1	1	0	0	0	0	0	0
---	---	---	---	---	---	---	---

 SEEK (Data register contains Track)

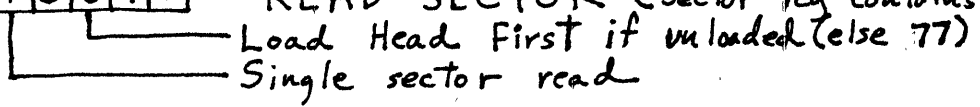


73 (77)

FFA4

0	1	1	1	0	0	1	1
---	---	---	---	---	---	---	---

 READ SECTOR (Sector reg contains Sector)



53 (57)

FFA4

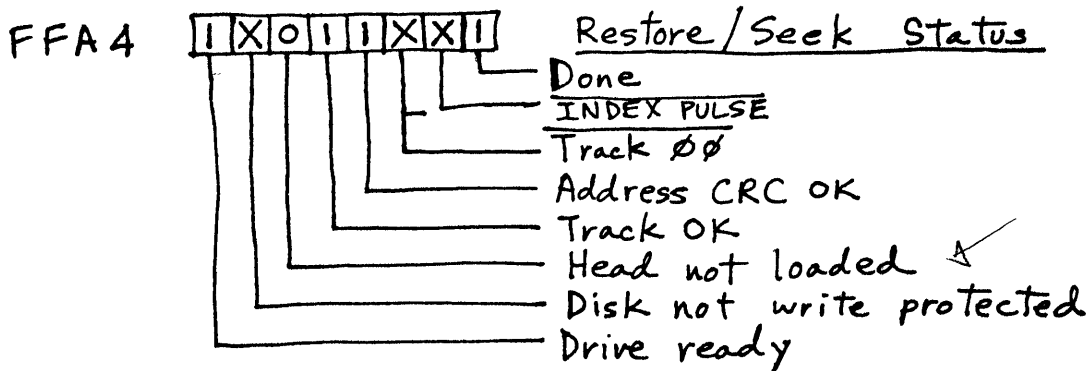
0	1	0	1	0	0	1	1
---	---	---	---	---	---	---	---

 WRITE SECTOR (sector reg contains Sector)

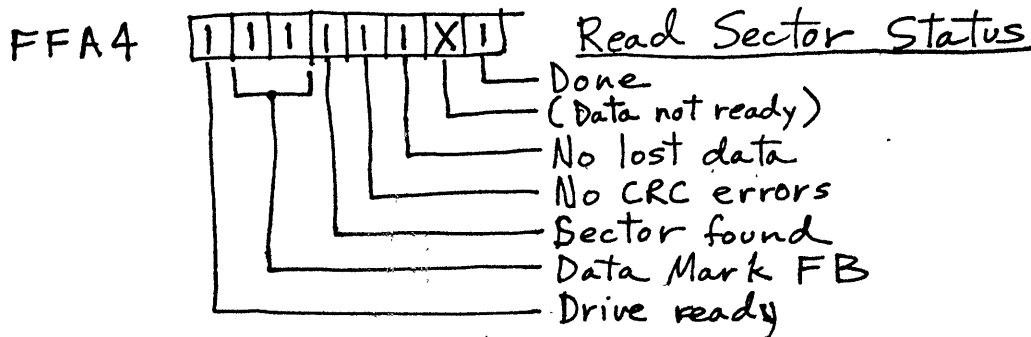


STATUS coming back

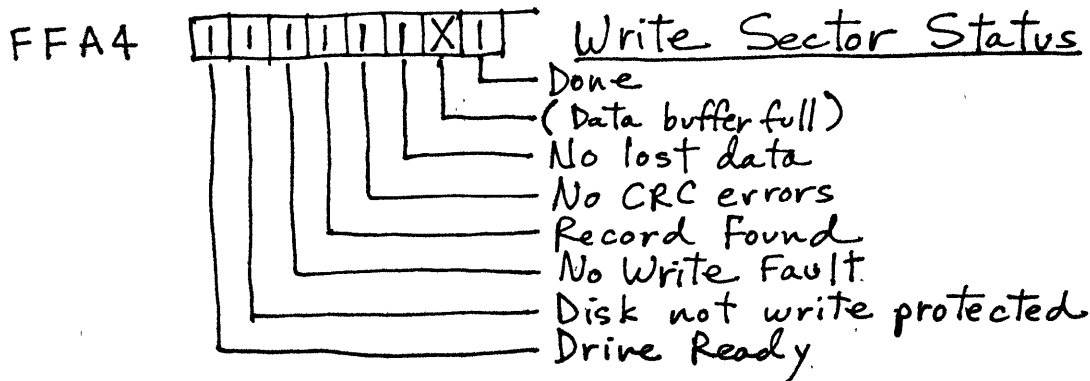
99 AND B9



FD AND FD

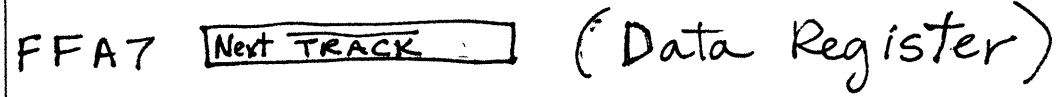
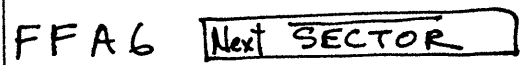
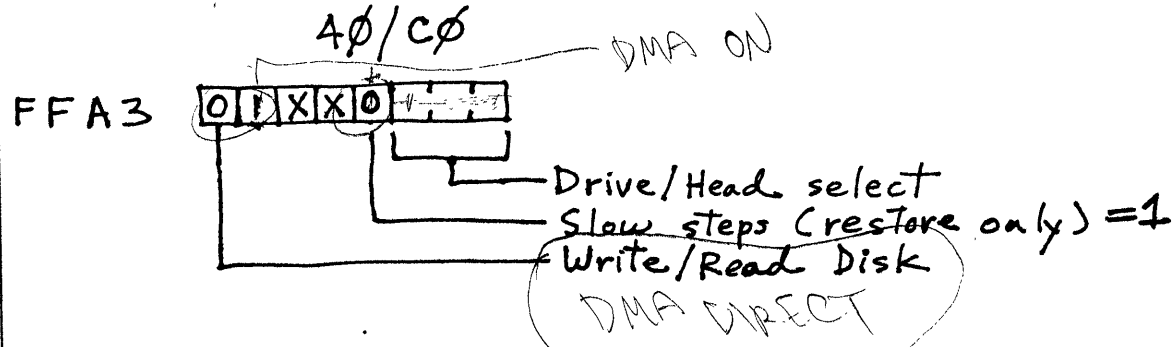
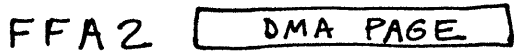
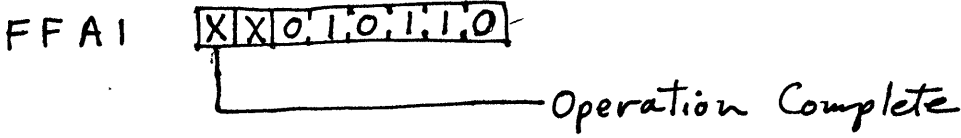
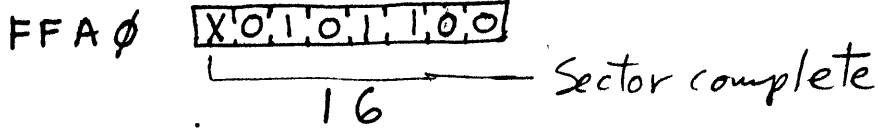


FD AND FD



CB
CA
DB
DA

2C




```

1: *
2: * RESTORE DRIVE (RESET AND INITIALIZE)
3: *
4: * CAUSES CURRENT OPERATION TO BE ABORTED AND
5: * A RESTORE OPERATION TO START.
6: *
7: *
049C 8: RESTOREI EQU *
049C C627 9: LDAB #27 ABORT CURRENT OPERATION
049E F7FFA4 10: STAB $FFA4
04A1 CE0000 11: LDX #0 INITIALIZE DISK FIA
04A4 FFFFA0 12: STX $FFA0
04A7 CEFFFF 13: LDX #FFFF
04AA FFFFA2 14: STX $FFA2
04AB CE2C16 15: LDX #2C16
04B0 FFFFA0 16: STX $FFA0
04B3 7FFFA3 17: CLR $f1a3
18: *
04B6 B6FFA3 19: restore LDAA $f1a3 SELECT SLOW STEP RATE
04B7 8A08 20: ORAA #8
04BE B7FFA3 21: STAA $FFA3
04C0 86F5 22: LDAA #6F5
04C3 B7FFA4 23: STAA $FFA4 START RESTORE AT 10 MS
04C3 39 24: RTS
25: *
26: * SEEK
27: *
28: * SEEKS TO DESIRED TRACK
29: * A-TRACK NUMBER
30: *
31: *
04C4 32: SEEK EQU *
04C4 43 33: COMB
04C5 BDC4C9 34: JSR WAIT WAIT FOR CURRENT OPERATION COMPLETE
04C6 B7FFA7 35: STAB $FFA7 STOP TRACK MS
04C8 B6FFA3 36: LDAA $f1a3 select fast step rate
04CE 84F7 8A08 37: ANDA #17
04D0 B7FFA3 38: STAA $f1a3
04D3 86E5 39: LDAA #6E5 SEEK TRACK
04D5 B7FFA4 40: STAA $FFA4
04D8 39 41: RTS
42: *
43: * WAIT
44: *
45: * WAITS FOR CURRENT OPERATION DONE AND DRIVE READY
46: *
47: *
04D9 48: WAIT EQU *
04D9 F6FFA4 49: LDAB $FFA4 1000000
04DC CA7E 50: ORAB #7E
04DE 5C 51: INCB
04DF 26F8 52: BNE WAIT
04E1 F6FFA4 53: LDAB $FFA4
04E4 39 54: RTS

```

```

1: *
2: *      READ A SECTOR
3: *
4: *      READS A SECTOR INTO MEMORY
5: *      STARTING AT DMA
6: *      A=Sector number
7: *
8: *
9: 04E5 reads EQU *
10: 04E5 C612      ldab #dma/256
11: 04E7 F7FFA2    reads1 stab $ffa2 set DMA page
12: 04EA 43        reads2 coma
13: 04EB B7FFA6    staa $ffa6 set sector number
14: 04ED B6FFA3    LDAA $FFA3 SET DMA AND READ
15: 04F1 840F      ANDA #0F
16: 04F3 8A40      ORAA #40
17: 04F5 B7FFA3    STAA $FFA3
18: 04F8 C673      LDAB #73 GET READ COMMAND
19: 04FA 2015      BRA WRITC JUMP
20: *
21: *      WRITE A SECTOR
22: *
23: *      WRITES A SECTOR FROM MEMORY
24: *      STARTING AT DMA
25: *      A=Sector number
26: *
27: *
28: 04FC WRITES EQU *
29: 04FC C612      ldab #dma/256
30: 04FE F7FFA2    writes1 stab $ffa2 set DMA page
31: 0501 43        writes2 coma
32: 0502 B7FFA6    staa $ffa6 set sector number
33: 0503 B6FFA3    LDAA $FFA3 SET DMA AND WRITE
34: 0505 840F      ANDA #0F
35: 050A 8A00      ORAA #00
36: 050C B7FFA3    STAA $FFA3
37: 050F C653      LDAB #53 GET WRITE COMMAND
38: 0511 WRITC EQU *
39: 0511 B6FFA2    LDAA $FFA2 CLEAR DONE FLAG
40: 0514 F7FFA4    STAB $FFA4 START DISK
41: *
42: 0517 WRITL EQU *
43: 0517 E6FFA1    LDAA $FFA1 OPERATION COMPLETE?
44: 051A 2B0A      EMI DONE YES, EXIT
45: 051C E6FFA0    LDAA $FFA0 NO, SECTOR COMPLETE?
46: 051F 2AF6      EPL WRITL NO, LOOP
47: 0521 7CFFA2    INC $FFA2 YES, INC DMA PAGE
48: 0524 20F1      BRA WRITL LOOP
49: 0526 DONE EQU *
50: 0526 B6FFA4    LDAA $FFA4 GET STATUS
51: 0528 39        RTS AND RETURN.

```

```
1: *
2: *      READ A SECTOR
3: *
4: *      X=Logical disk sector
5: *      E=DMA page
6: *
7: read  equ      *
OBAS C612 8:      ldab   #dma/256
OBAS F7FFA2 9: readi  stab   #ifa2  Set DMA page
OBAD B00ED2 10:     jsr    seeki  Seek to a logical track
OBEO B004D9 11:     jsr    wait   Wait for seek done
OBEB B004EA 12:     jsr    reads2 Read the sector
OBEE 849C 13:     anda   #*7c
OBEE 819C 14:     cmpa   #*7c
OBEB 2601 15:     bne    readerr
OBEC 39 16:     rts     Return if no errors
17: *
18: readerr equ *
OBED B0068A 19:     jsr    crif   Else report error and try again
OBED C2033B 20:     idx   #reader
OBED 589C 21:     eora   #*7c
OBED B00243 22:     jsr    errep
OBED B00AAA 23:     jsr    sectrk report sector and track
OBED DE08 24:     idx   t
OBED F6FFA2 25:     ldab   #ifa2
OBED 20D8 26:     bra    readi
27: *
28: *      SEEK LOGICAL TRACK
29: *
30: *      X=Logical disk sector
31: *      returns A=sector
32: *
33: seeki  equ      *
OBED2 DF08 34:     stx   t
OBED4 9609 35:     ldaa  t+1  Get sector number
OBED6 840F 36:     anda  #*f
OBED8 36 37:     psha
OBED7 9609 38:     ldaa  t+1  Get track number
OBED8 D608 39:     ldab  t
OBEDB 34 40:     isrb
OBEDC 46 41:     rora
OBEDF 34 42:     isrb
OBEE0 46 43:     rora
OBEE1 34 44:     isrb
OBEE2 46 45:     rora
OBEE3 44 46:     isra
OBEE4 B004C4 47:     jsr    seeki  Seek to the track
OBEE7 32 48:     puia.  Restore sector number to A
OBEE8 39 49:     rts     Return
```

C680
for
GET
VIEW

Status of seek in B

```
1: *  
2: *      WRITE A SECTOR  
3: *  
4: *      X=Logical disk sector  
5: *      B=DMA Page  
6: *  
7: write  equ      *  
8:      ldab      #dma/256  
9: writei stab      $ffa2  Set DMA page  
10:      jsr      seeki   Seek to a logical track  
11:      jsr      wait    Wait for seek done  
12:      jsr      writes2 Write the sector  
13:      anda     ##fc  
14:      cmpa     ##fc  
15:      bne     writerr  
16:      rts      Return if no errors  
17: *  
18: writerr equ      *  
19:      jsr      crlf    Else report error and try again  
20:      idx      #writer  
21:      eora     ##fc  
22:      jsr      errep  
23:      jsr      sectrk  
24:      idx      t  
25:      ldab     $ffa2  
26:      bra     writei  
27: *  
28: RESTORER equ      *  
29:      jsr      restorei  
30:      jmp      resterr
```

OBE9
OBE9 C&12
OBE9 F7FFA2
OBE9 ED0BD2
OBF1 ED04D9
OBF4 ED0301
OBF7 84FC
OBF9 81FC
OBF8 2&01
OBF8 37

OBEF
OBEF BD068A
OC01 CE03C9
OC04 83FC
OC06 ED0243
OC09 ED09AA
OC0C DE08
OC0E F&FFA2
OC11 20D8
OC13
OC13 BD047C
OC16 7E0B8D