DCTRL.DOC

DISK CONTROLLER

^L

THIS FILE IS DCTRL.DOC --- DESCRIPTION OF DISK CONTROLLER

DISK CONTROLLER OPCODES I.E. PDP-10 I/O INSTRUCTION CODES

715	READ CMD	RC
716	READ MA	RM
717	READ DA	RA
720	READ ECC	RE
721	LOAD CMD	LC
722	LOAD MA	LM
723	LOAD DA	LA
724	LOAD ECC	LE

LOAD ECC

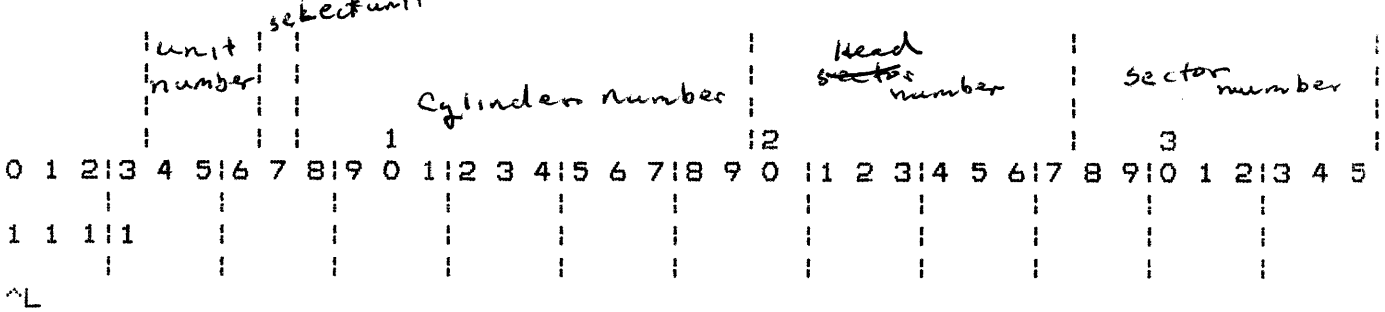
PRIMARILY CLEARS ERROR CORRECTING CODE LOGIC.
ALSO, DOES SPECIAL FUNCTION DEPENDING ON CONTENTS OF E.

CONTENTS OF E	FUNCTION
1	START A COMMAND
2	INITIALIZE CONTROLLER (SEND BEFORE STARTING CMD

LOAD DA, READ DA

LOAD AND READ DISK ADDRESS. SELECT UNIT.

BITS	CONTENTS
04-06	UNIT NUMBER
07	SELECT UNIT
	THIS BIT MUST BE CLEARED, THEN SET, TO SELECT UNIT
08-19	CYLINDER NUMBER
20-27	HEAD NUMBER
28-35	SECTOR NUMBER



^L

```
DETAIL FOR LOAD CMD:
BIT #   FUNCTION
=====   ========
15      USE SECTOR COUNTER
                FOR ROTATIONAL POSITION SENSING - TELLS CONTROLLER
                TO LISTEN TO SIGNALS FROM DRIVE TO TELL WHERE IT IS.
16      RELEASE
                RELEASE DRIVE FOR USE BY ANOTHER COMPUTER
17      RECALIBRATE
                SIDE EFFECT: RESETS SEEK ERROR
---------
18      FAULT CLEAR
                SIDE EFFECT: RESETS SEL UNIT FAULT
                USE BITS 17,18 WITH COMMAND=4
19      DATA STROBE LATE
20      DATA STROBE EARLY
---
21      SERVO OFFSET MINUS
22      SERVO OFFSET PLUS
                BITS 15-22 ARE 0 BY DEFAULT
                BITS 19-22 ARE FOR READING MARGINAL DATA
23      CMD FROM MEMORY
                COMMAND SENDS DATA FROM MEMORY TO DISK
                (USE WITH WRITE COMMANDS)
---
24      CMD 0
25      CMD 1
26      CMD 2
                BITS 24-26 ARE DISK CONTROLLER SEQUENCER START ADDRESS
------
27      32 BIT MODE
28      ***
29      ***
---
30      ANY ATTENTION INTERRUPT ENABLE
                ALLOWS INTERRUPT WHEN ANY UNIT IS AT ATTENTION
31      ***
32      DONE INTERRUPT ENABLE
---
33      ***
34      ***
35      ***

DETAIL FOR CMD (BITS 24-26)
        CMD     FN
        ---     --
        0       READ
        1       WRITE
        3       WRITE ALL               (SECTOR + FORMAT DATA)
        4       CONTROL FUNCTIONS       (RECAL, FAULT CLEAR)
        5       (WILL BE 'SEEK')
^L
```

```
DETAIL FOR READ CMD:
BIT #   FUNCTION
=====   ========
00      SELECT ERROR
                IF 0, SUCCESSFULLY TALKING TO A DISK UNIT
                IF 1, BITS 01-06 NOT VALID.
01      SEL UNIT WRITE PROTECTED
02      -SEL UNIT READY
                -READY LIGHT ON DRIVE
---
03      SEL UNIT ON CYLINDER
                SEEK COMPLETE
04      SEL UNIT SEEK ERROR
                DETECTED BY DRIVE
05      SEL UNIT FAULT
                DETECTED BY DRIVE
---
06      HEADER ECC ERR

07      HEADER COMPARE ERROR
                SECTOR COUNTER /= SECTOR HEADER
                (FOR USE WITH ROTATIONAL POSITION SENSING)
08      NOT BUSY
                DISK CONTROLLER SEQUENCER IS STOPPED
------
09      HARD ECC ERROR

10      SOFT ECC ERROR

11      READ OVERRUN ERROR
---
12      WRITE OVERRUN ERROR
13      SECTOR OVERRUN ERROR
14      INTERNAL PARITY ERROR
                BITS 6-14 ARE DETECTED BY CONTROLLER
---
15-17   (SAME AS LOAD CMD)
---------
18-20   (SAME AS LOAD CMD)
---
21-23   (SAME AS LOAD CMD)
---
24-26   (SAME AS LOAD CMD)
------
27      (SAME AS LOAD CMD)
28      ANY ERROR
                OR OF ALL ERROR BITS EXCEPT 14 (INTERNAL PARITY)
29      ANY ATTENTION
                AY UNIT IS AT ATTENTION
---
30      (SAME AS LOAD CMD)
^L
```

31 -ACTIVE
        SEQUENCER STOPPED AND FIFO EMPTY (ON INPUT TO MEMORY)
32 (SAME AS LOAD CMD)
---
33 READ COMPARE ERROR
        CONTROLLER COMPARES MEMORY TO DISK (NOT IMPLEMENTED)
34 TIMOUT ERROR
        *** NOT IMPLEMENTED ***
35 MEM PAR ERR

BYTE PACKING BY HARDWARE:
        1ST 4 BYTES GO TO (OR COME FROM) WORD N, BITS 0-31. NEXT BYTE
        GETS SPLIT 4 BITS TO WORD N, BITS 32-35, AND 4 BITS TO WORD N+1,
        BITS 32-35. LAST 4 BYTES TO WORD N+1, BITS 0-31.

32 BIT MODE: DATA TAKEN FROM HIGH ORDER 4 BYTES ONLY.

^L

READ FIFO PROM

-32 BIT MODE

PROM ADDRESS	LOAD NUMBER	FULL
0	1	0
1	2	0
2	3	0
3	4	0
4	4	1
5	0	0
6	1	0
7	2	0
10	3	0
11	5	1
12	0	0

NOTE: NEW ADDRESS = ADDRESS + 1
EXCEPT 12--> 0

32 BIT MODE

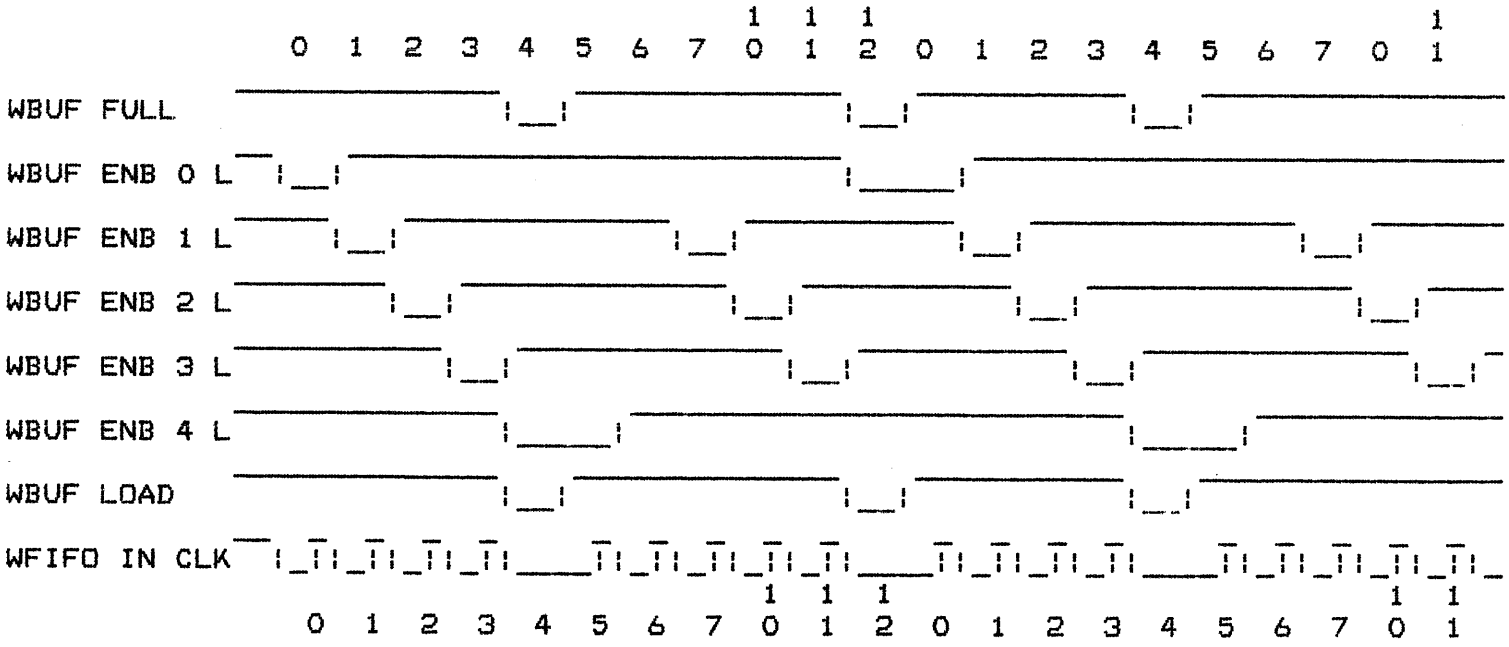
PROM ADDRESS	LOAD NUMBER	FULL
20	1	0
21	2	0
22	3	0
23	4	1
24	0	0
25	1	0
26	2	0
27	3	0
30	4	1
31	0	0

NOTE: NEW ADDRESS = ADDRESS +1
EXCEPT 31--> 20

^L

WRITE FIFO PROM

-32 BIT MODE			32 BIT MODE		
PROM ADDRESS	SEL NUMBER	WBUF FULL	PROM ADDRESS	SEL NUMBER	WBUF FULL
0	0	1	20	0	1
1	1	1	21	1	1
2	2	1	22	2	1
3	3	1	23	3	1
4	4	0	24	0	0
5	4	1	25	0	1
6	0	1	26	1	1
7	1	1	27	2	1
10	2	1	30	3	1
11	3	1	31	0	0
12	0	0			
NOTE: NEW ADDRESS = ADDRESS + 1 EXCEPT 12--> 0			NOTE: NEW ADDRESS = ADDRESS +1 EXCEPT 31--> 20		



NOTE:
THE FIFO IS CLOCK ON THE RISING EDGE OF WFIFO IN CLK.

^L

MICRO CODE DEFINITIONS

```
        PROM A   UM 0-7
        PROM B   UM 8-15
        PROM C   UM 16-19

UM  0   IR SEL READ CLK - NORMALLY THE DISK CONTROLLER DERIVES ITS INTERNAL
                CLOCK FROM THE 103 MS SERVO CLOCK THE DRIVE SUPPLIES. DURING A
                READ, WITHIN A GOOD DATA FIELD, THE CONTROLLER USES THE READ
                DATA CLOCK FROM THE DRIVE.

UM  1   IR BYTE CLK - STRETCHES MICROINSTRUCTION UNTIL END OF ONE BYTE TIME-
                UNTIL THE BITCNT REGISTER MOD 8 = 0.
UM  2   IR WAIT 0 - CONDITIONS THAT DELAY EXECUTION OF MICROINSTRUCTION. ACTION
    3   IR WAIT 1    HAPPEN ON FINAL CLOCK.
    4   IR WAIT 2
        0 -
        1 - WAITED (2 CYCLES)          [BYTE CLOCK + DOUBLE = 1.6 USEC]
        2 - ON CYLINDER (SYNC)
        3 - SECTOR START   [WAIT UNTIL TRAILING EDGE OF SECTOR PULSE. IF SECTOR
                            COUNTER OPTION PROVIDED, WAIT UNTIL CORRECT SECTOR
                            NUMBER ALSO.]
        4 - SYNC BYTE DETECT  [THE PREAMBLE CONSISTS OF A LONG STRING OF 0'S
                               FOLLOWED BY A SINGLE ONE. THIS WAITS UNTIL
                               DETECTING THE ONE. ON THE NEXT CLOCK THE RSH
                               WILL CONTAIN THE FIRST DATA BIT
        5 - INDEX (SYNC)
        6 - BIT CNT=SECTOR SIZE-1 BYTE
        7 - BIT CNT=ECC MODULUS-1
UM  5   IR FCN 0
    6   IR FCN 1
    7   IR FCN 2
    8   IR FCN 3
        0 -                          10 - SET DONE
        1 - CLR RSH                  11 - SET DONE IF NO ECC
        2 - CLR ECC                  12 - TEST HEADER
        3 - RESET ATTENTION          13 - TEST HEADER ECC
        4 -                          14 - TEST ECC SOFT
        5 -                          15 - SET ECC HARD
        6 -                          16 - TEST INDEX DONE
        7 -                          17 - SAMPLE ECC
UM  9   IR CHECK SECTRO OVERRUN
UM 10   -CYLINDER TAG
UM 11   -HEAD TAG
UM 12   IR TAG ENA
UM 13   IR READ GATE
UM 14   IR WRITE GATE
UM 15   IR DATA SEL 0
   16   IR DATA SEL 1
        0 - WRITE SYNC (END OF BYTE)   AND  ECC FEEDBACK
        1 - WRITE 0                    AND  ECC READ DATA
        2 - WRITE DATA                 AND  ECC WRITE DATA
        3 - WRITE ECC                  AND  ECC HOLD
UM 17   DATA FIELD
UM 18   TAKE BYTE
UM 19   ADDRESS MARK ENABLE
```

```
        WRITE - 100
        READ ALL - 200
        WRITE ALL - 300
        RECALIBRATE - 400
        SEEK - 500
        SEND HEAD - 600



                        WRITE - 100

WRITE - DO IMPLIED SEEK; WAIT FOR SECTOR, CHECK HEADER; TRANSFER 576 BYTES DATA
        AND 4 BYTES OF ECC.

        ;SEND CYLINDER
100     CYL-TAG, BYTE-CLK, WAIT [ON-CYLINDER]   $  ;WAIT FOR PREV SYNC
101     CYL-TAG, BYTE-CLK, TAG-ENB, WAIT [2] $
102     CYL-TAG, BYTE-CLK,WAIT [ON-CYLINDER] $

        ;SEND HEAD SELECT

103     HEAD-TAG, BYTE-CLK $
104     HEAD-TAG, TAG-ENB, BYTE-CLK, WAIT [2] $
105     HEAD-TAG, BYTE-CLK $

        ;ASSERT CONTROL TAG

106     BYTE-CLK $ SETUP BUS
107     TAG-ENB, BYTE-CLK, WAIT [2] $  ;DELAY FOR MINI-SEEK
                                       ;START
110     TAG-ENB, BYTE-CLK, WAIT [ON-CYLINDER] $

        ;WAIT FOR SECTOR

111     TAG-ENB, BYTE-CLK, WAIT [SECTOR-START] $

        ;DELAY 12 BYTES TO READ GATE - AVOIDS SEEING GARBAGE

112     TAG-ENB, BYTE-CLK, WAIT [2] $
113     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $
114     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $
115     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $
116     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $
117     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $

        ;ASSERT READ-GATE, WAIT 10 USEC FOR GOOD DATA
```

```
120     TAG-ENB, CSO, READ-GATE, BYTE-CLK, WAIT [2] $
121     TAG-ENB, CSO, READ-GATE, BYTE-CLK, WAIT [2] $
122     TAG-ENB, CSO, READ-GATE, BYTE-CLK, WAIT [2] $
123     TAG-ENB, CSO, READ-GATE, BYTE-CLK, WAIT [2] $
124     TAG-ENB, CSO, READ-GATE, BYTE-CLK, WAIT [2] $
125     TAG-ENB, CSO, READ-GATE, BYTE-CLK, WAIT [2] $
126     TAG-ENB, CSO, READ-GATE, BYTE-CLK, FCN [CLR-RSH] $
127     READ-CLK, TAG-ENB, CSO, READ-GATE, BIT-CLK, WAIT [SYN-BYTE],
        FCN [CLR-ECC] $

        CHECK HEADER FOR VALID POSITION

130     READ-CLK, TAG-ENB, CSO, READ-GATE, BYTE-CLK, ECC [READ-DATA] $
131     READ-CLK, TAG-ENB, CSO, READ-GATE, BYTE-CLK, ECC [READ-DATA] $
132     READ-CLK, TAG-ENB, CSO, READ-GATE, BYTE-CLK, ECC [READ-DATA] $
133     READ-CLK, TAG-ENB, CSO, READ-GATE, BYTE-CLK, ECC [READ-DATA] $

        ;CHECK HEAD ECC

134     READ-CLK, TAG-ENB, CSO, READ-GATE, BYTE-CLK, ECC [READ-DATA] $
135     READ-CLK, TAG-ENB, CSO, READ-GATE, BYTE-CLK, ECC [READ-DATA] $
136     READ-CLK, TAG-ENB, CSO, READ-GATE, BYTE-CLK, ECC [READ-DATA] $
137     READ-CLK, TAG-ENB, CSO, READ-GATE, BYTE-CLK, ECC [READ-DATA] $
140     TAG-ENB, CSO, BIT-CLK, ECC [HOLD], WAIT [2], FCN [TEST-HEADER-ECC] $
        ; DROP READ GATE, TEST HEADER ECC
141     TAG-ENB, CSO, BIT-CLK, WAIT [2] $ ;4 CLOCKS BEFORE WRITE

        ;WRITE A GAPE OF 16 ZERO BYTES FOLLOWED BY SYNC
        ;THE READ LOGIC USES 12 BYTES FOR VFO RELOCK

142     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-0], WAIT [2]  $
143     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-0], WAIT [2]  $
144     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-0], WAIT [2]  $
145     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-0], WAIT [2]  $
146     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-0], WAIT [2]  $
147     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-0], WAIT [2]  $
150     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-0], WAIT [2]  $
151     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-0], WAIT [2]  $
152     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-SYNC],
        FCN [CLR-ECC], TAKE-BYTE $  ;GET FIRST DATA BYTE INTO SHIFT REGISTER

        ;WRITE DATA FIELD
153     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-DATA],
        DATA-FIELD, TAKE-BYTE, WAIT [SECTOR-SIZE-1] $
154     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-DATA],
        DATA-FIELD $  ;LAST BYTE, DON'T TAKE N + 1

        ;WRITE ECC
```

```
155     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-ECC] $
156     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-ECC] $
157     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-ECC] $
160     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-ECC] $

        ;WRITE A FEW TRAILING 0'S

161     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-0] $
162     TAG-ENB, CSO, BYTE-CLK, WRITE-GATE, ECC [WRITE-0] $
163     WRITE-GATE, BYTE-CLK $  ;DROP CONTROL TAG
164     BYTE-CLK, FCN [SET-DONE] $
```

^L

READ ALL - 200

```
READ ALL - THIS IS A CROCK BECAUSE OF LACK OF WORD COUNTER
         - JUST READS UNTIL END OF SECTOR

        ;SEND CYLINDER

200     CYL-TAG, BYTE-CLK, WAIT [ON-CYLINDER] $
201     CYL-TAG, TAG-ENB, BYTE-CLK, WAIT [2] $
202     CYL-TAG, BYTE-CLK, WAIT [ON-CYLINDER] $

        ;SEND HEAD

203     HEAD-TAG, BYTE-CLK $
204     HEAD-TAG, TAG-ENB, BYTE-CLK, WAIT [2] $
205     HEAD-TAG, BYTE-CLK $

        ;CONTROL TAG, WATCH OUT FOR SERVO OFFSET  SEEK

206     BYTE-CLK $
207     TAG-ENB, BYTE-CLK, WAIT [2] $
210     TAG-ENB, BYTE-CLK, WAIT [ON-CYLINDER] $

        ;WAIT FOR SECTRO - THIS IS TRICKY IF WE AREN'T
        ;USING SECTOR COUNTER MODE

211     TAG-ENB, BYTE-CLK, WAIT [SECTOR-START] $

        ;DELAY 12 BYTES TO READ GATE

212     TAG-ENB, BYTE-CLK, WAIT [2] $
213     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $
214     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $
215     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $
216     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $
217     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $

        ;ASSERT READ GATE, WAIT 10 USEC FOR GOOD DATA

220     TAG-ENB, CSO, READ-GATE, BYTE-CLK, WAIT [2] $
221     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $
222     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $
223     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $
224     TAG-ENB, CSO, BYTE-CLK, WAIT [2] $
225     TAG-ENB, CSO, BYTE-CLK, WAIT [2], CLR RSH $
226     TAG-ENB, CSO, READ-GATE, BYTE-CLK, FCN [CLR-ECC] $  ;RESET BIT COUNT

        ;READ CONTINUOUSLY TO END OF SECTOR

227     READ-CLK, TAG-ENB, CSO, BYTE-CLK, READ-GATE, DATA-FIELD,
        WAIT [ECC-MODULUS] $  ;WAIT FOR LONG TIME, STOP BY SECTOR OVERRUN
230     BYTE-CLK, FCN [SET-DONE] $  ;FOR THE HELL OF IT
```

^L

WRITE ALL - 300

WRITE ALL - IMPLICIT SEEK, THEN WRITE BYTES FROM ONE INDEX PULSE TO THE NEXT

```
        ;SEND CYLINDER

300     CYL-TAG, BYTE-CLK, WAIT [ON-CYLINDER] $
301     CYL-TAG, BYTE-CLK, TAG-ENB, WAIT [2] $
302     CYL-TAG, BYTE-CLK, WAIT [ON-CYLINDER] $

        ;SEND HEAD

303     HEAD-TAG, BYTE-CLK $
304     HEAD-TAG, BYTE-CLK, TAG-ENB, WAIT [2] $
305     HEAD-TAG, BYTE-CLK $

        ;CONTROL TAG

306     BYTE-CLK $
307     TAG-ENB, BYTE-CLK, WAIT [2] $
310     TAG-ENB, BYTE-CLK, WAIT [ON-CYLINDER] $
311     TAG-ENB, BYTE-CLK, WAIT [2] $  ;WAIT FOR A WHILE BEFORE WRITING
312     TAG-ENB, BYTE-CLK, WAIT [2] $
313     TAG-ENB, BYTE-CLK, WAIT [2] $

        ;WRITE DATA TO NEXT INDEX

314     TAG-ENB, BYTE-CLK, WAIT [INDEX], FCN [CLR-ECC] $
315     TAG-ENB, BYTE-CLK, WRITE-GATE, ECC [WRITE-0], TAKE-BYTE $
        ;WRITE BYTE OF 0'S, INIT PIPELINE
316     TAG-ENB, BYTE-CLK, WRITE-GATE, ECC [WRITE-DATA],

317     TAG-ENB, BYTE-CLK, WRITE-GATE, ECC [WRITE-DATA],
        TAKE-BYTE, DATA-FIELD,WAIT [INDEX] $
320     BYTE-CLK, FCN [SET-DONE] $
```

^L

RECALIBRATE - 400

RECALIBRATE OR OTHER DRIVE OP. RESET ATTENTION

```
400     BYTE-CLK, WAIT [2] $
401     TAG-ENB, BYTE-CLK, WAIT [2] $  ;SEND IT FOR A WHILE
402     TAG-ENB, BYTE-CLK, WAIT [2] $  ;SEND IT FOR A WHILE
403     TAG-ENB, BYTE-CLK, WAIT [2] $  ;SEND IT FOR A WHILE
404     TAG-ENB, BYTE-CLK, WAIT [2] $  ;SEND IT FOR A WHILE
405     TAG-ENB, BYTE-CLK, WAIT [2] $  ;SEND IT FOR A WHILE
406     TAG-ENB, BYTE-CLK, WAIT [2] $  ;SEND IT FOR A WHILE
407     TAG-ENB, BYTE-CLK, READ-GATE, WAIT [2] $
        ;SOME DRIVES LIKE READ GATE TO RESET ATTENTION
410     TAG-ENB, BYTE-CLK, FCN [RESET-ATTENTION] $
411     TAG-ENB, BYTE-CLK, WAIT [2] $  ;SEND IT FOR A WHILE
412     TAG-ENB, BYTE-CLK, WAIT [2] $  ;SEND IT FOR A WHILE
413     TAG-ENB, BYTE-CLK, WAIT [2] $  ;SEND IT FOR A WHILE
414     TAG-ENB, BYTE-CLK, WAIT [2] $  ;SEND IT FOR A WHILE
415     BYTE-CLK, FCN [SET-DONE] $
```

^L

SEEK - 500

```
SEEK - INITIATE SEEK

        ;SEND CYLINDER

500     CYL-TAG, BYTE-CLK, WAIT [ON-CYLINDER] $  ;PREVIOUS SEEK
501     CYL-TAG, BYTE-CLK, TAG-ENB, WAIT [2] $
502     CYL-TAG, BYTE-CLK, FCN [SET-DONE] $
```

SEND HEAD - 600

```
SEND HEAD FOR THE HELL OF IT

600     HEAD-TAG, BYTE-CLK, WAIT [ON-CYLINDER] $
601     HEAD-TAG, BYTE-CLK, TAG-ENB, WAIT [2]
602     HEAD-TAG, BYTE-CLK, FCN [SET-DONE] $
```

^L

TAG BUS DECODE

BUS	TAG 1 IN CYLINDER ADDRESS	TAG 2 IN HEAD ADDRESS	TAG 3 IN CONTROL ADDRESS
BIT 0	0	0	WRITE GATE
BIT 1	1	1	READ GATE
BIT 2	2	2	SERVO OFFSET PLUS
BIT 3	3		SERVO OFFSET MINUS
BIT 4	4		FAULT CLEAR
BIT 5	5		AM ENABLE
BIT 6	6		RTZ
BIT 7	7		DATA STROBE EARLY
BIT 8	8		DATA STROBE LATE
BIT 9	9		RELEASE

^L

SIGNAL LINES

I. ADDRESS AND CONTROL TAG FUNCTIONS (RECEIVED BY THE UNIT)

1. CYLINDER ADDRESS (TAG 1)
   STROBES THE TEN BUS LINES USED TO CARRY THE CYLINDER ADDRESS TO THE DRIVE.

2. HEAD SELECT (TAG 2)
   STROBES THE HEAD ADDRESS THAT WILL BE SELECTED BY BITS 0 - 2.

3. CONTROL SELECT (TAG 3)
   THIS SIGNAL ACTS AS AN ENABLE AND MUST BE TRUE FOR THE ENTIRE CONTROL OPERATION.
   1. WRITE GATE (BIT 0)
      THE WRITE GATE LINE ENABLES THE WRITE DRIVE.
   2. READ GATE (BIT 1)
      ENABLING OF THE READ GATE, ENABLES DIGITAL READ DATA ON THE TRANSMISSION LINES. THE LEADING EDGE OF READ GATE TRIGGERS THE READ CHAIN TO SYNCRONIZE ON AN ALL ZEROS PATTERN.
   3. SERVO OFFSET PLUS (BIT 2)
      WHEN THIS SIGNAL IS TRUE, NO PHYSICAL MOVEMENT OF THE HEADS IS PERFORMED IN THE DRIVE, USED ONLY TO MEET TIMING REQUIREMENTS OF SMD DRIVE FAMILY.
   4. SERVO OFFSET MINUS (BIT 3)
      WHEN THIS SIGNAL IS TRUE, NO PHYSICAL MOVEMENT OF THE HEADS IS PERFORMED IN THE DRIVE, USED ONLY TO MEET TIMING REQUIREMENTS OF SMD DRIVE FAMILY.
   5. FAULT CLEAR (BIT 4)
      A PULSE, 100 NS MINIMUM, SENT TO THE MMD WILL CLEAR THE FAULT FLIP-FLOP IF THE FAULT CONDITION NO LONGER EXITS.
   6. AM ENABLE (BIT 5)
      THE AM (ADDRESS MARK) ENABLE LINE, IN CONJUNCTION WITH WRITE GATE OR READ GATE, ALLOWS THE WRITING OR RECOVERING OF ADDRESS MARKS. WHEN AM ENABLE IS TRUE WHILE WRITE GATE IS TRUE, THE WRITER STOPS TOGGLING AND ERASES THE DATA, CREATING AN ADDRESS MARK. WRITE FAULT DETECTION IN THE UNIT IS INHIBITED BY THIS SIGNAL.

   NOTE: IF ADDRESS MARK IS NOT USED, BIT 5 MUST BE HELD INACTIVE DURING CONTROL SELECT FUNCTIONS.

7. RTZ (BIT 6)
   A PULSE, 250 NS MINIMUM, 1.0 MS MAXIMUM, SENT TO THE MMD WILL CAUSE THE ACTUATOR TO SEEK TRACK 0, RESET THE HEAD REGISTER AND CLEAR THE SEEK ERROR FLIP-FLOP.
   THIS SEEK IS SIGNIFICANTLY LONGER THAN ANORMAL SEEK TO TRACK 0, AND SHOULD ONLY BE USED FOR RECALIBRATION, NOT DATA ACQUISITION.
8. DATA STROBE EARLY (BIT 7)
   WHEN THIS LINE IS TRUE, THE MMD PLO DATA SEPARATOR WILL STROBE THE DATA AT A TIME EARLIER THAN NOMINAL. NORMAL STROBE TIMING WILL BE RETURNED WHEN THE LINE IS FALSE.
9. DATA STROBE LATE (BIT 8)
   WHEN THIS LINE IS TRUE, THE MMD PLO DATA SEPARATOR WILL STROBE THE DATA AT A TIME LATER THAN NOMINAL. NORMAL STROBE TIMING WILL BE RETURNED WHEN THE LINE IS FALSE.

NOTE: THE DATA STROBE SIGNALS ARE INTENTED TO BE AN AID IN RECOVERING MARGINAL DATA. THE DATA STROBE POSITION RETURNS TO NOMINAL WHEN THE RESPECTIVE SIGNALS GO FALSE. A CARRIAGE OFFSET WILL RESULT IN A LOSS ON CYLINDER AND SEEK END FOR A PERIOD OF 2 MS. DATA SHALL NOT BE WRITTEN WHILE IN THE OFFSET MODE.

10. RELEASE (BIT 9) (DUAL CHANNEL ONLY)

II. INDIVIVUAL LINES

1. SECTOR MARK
   THE SECTOR MARK IS DERIVED FROM THE SERVO TRACK.
   TIMING INTEGRITY IS MAINTAINED THROUGHOUT SEEK OPERATIONS.

2. FAULT
   WHEN THIS IS TRUE, A FAULT CONDITION EXISTS IN THE MMD. THE FOLLOWING TYPES OF FAULTS MAY BE DETECTED BY THE MMD; DC VOLTAGE FAULT, HEAD SELECT FAULT, WRITE FAULT, WRITE OR READ WHILE OFF CYLINDER, AND WRITE GATE DURING A READ OPERATION. A FAULT CONDITION WILL IMMEDIATELY INHIBIT THE WRITER TO PREVENT DATA DESTRUCTION. THE DC VOLTAGE FAULT INDICATES A BELOW NORMAL VOLTAGE FROM THE POSITIVE OR NEGATIVE POWER SUPPLIES. THE HEAD SELECT FAULT INDICATES THAT MORE THAN ONE HEAD IS SELECTED. THE WRITE FAULT INDICATES LOW (OR THE ABSENCE OF) WRITE CURRENT AS WELL AS THE ABSENCE OF WRITE DATA.
   THIS LINE MAY BE CLEARED BY CONTROL SELECT, OR FAULT CLEAR ON THE OPERATOR PANEL, OR MASTER FAULT CLEAR ON THE FAULT CARD (PROVIDING THE FAULT NO LONGER EXISTS). FAULTS ARE STORED IN INDIVIDUAL FLIP-FLOPS AS A MAINTENANCE AID, AND MAY BE CLEARED ONLY BY POWERING DOWN DC POWER OR CLEARING THE FAULT BY MEANS OF THE SWITCH ON THE FAULT CARD.

3. SEEK ERROR
   WHEN THIS LINE IS TRUE, A SEEK ERROR HAS OCCURRED. THE ERROR MAY ONLY BE CLEARED BY PERFORMING A RTZ. THIS SIGNAL INDICATES THAT THE CARRIAGE HAS MOVED TO A POSITION OUTSIDE THE RECORDING FIELD.

4. ON CYLINDER
   THIS STATUS INDICATES THE SERVO HAS POSITION THE HEADS OVER A TRACK. THE SATAUS IS CLEARED WITH ANY SEEK INSTRUCTION CAUSING CARRIAGE MOVEMENT, OR A ZERO-TRACK SEEK. A CARRIAGE OFFSET WILL RESULT IN LOSS OF ON CYLINDER FOR A PERIOD OF 2.75 ms "(NOMIAL). FO A ZERO TRACK SEEK, ON CYLINDER DROPS FOR 150 us MAX. FOR 80/160 MB.

5. INDEX
   THIS SIGNAL OCCURS ONCE PER REVOLUTION, AND ITS LEADING EDGE IS CONSIDERED THE LEADING EDGE OF THE SECTOR ZERO, TYPICALLY 2.5 us. TIMING INTEGRITY IS RETAINED THROUGHOUT SEEK OPERATIONS.

6. UNIT READY
   WHEN TRUE, AND THE UNIT IS SELECTED, THIS LINE INDICATES THAT THE UNIT IS UP TO SPEED, AND THE HEADS ARE OVER THE RECORDING TRACKS AND NO FAULT CONDITION EXITS WITHIN THE MMD. THESE THREE LINES CONTAIN ENCODED INFORMATION WHICH IS DECODED IN THE DRIVE TO PRODUCE THE TAG FUNCTION.

7. OPEN CABLE
   DISABLES THE INTERFACE IN THE EVENT THAT THE "A" INTERFACE CABLE IS DISCONNECTED OR CONTROLLER POWER IS LOST.

8. UNIT SELECT TAG
   THIS SIGNAL GATES THE DESIRED LOGIC NUMBER INTO THE LOGIC NUMBER COMPARE CIRCUIT. THE UNIT WILL BE SELECTED INTERNALLY 600 ns (MAXIUM) AFTER THE LEADING EDGE OF THIS SIGNAL.

   NOTE: THIS FUNCTION MUST BE EDGE TRIGGERED.

9. UNIT SELECT (0, 1, 2, 3)
   THESE FOUR LINES ARE BINARY CODED TO SELECT THE LOGICAL NUMBER 1 TO 16 MMDs. THE UNIT NUMBER (0 THROUGH 15) IS SELECTABLE BY MEANS OF SWITCHES LOCATED ON A CARD IN THE LOGIC CHASSIS.

10. ADDRESS MARK FOUND

11. SEEK END
12. UNIT SELECTED
13. WRITE PROTECTED
14. POWER SEQUENCING

III. DATA AND CLOCK LINES
1. WRITE DATA
2. SERVO CLOCK
3. READ DATA
4. READ CLOCK
5. WRITE CLOCK

^L

DRIVE FAULT

1	VOLTAGE FAULT	
2		
3		
4		
5		
6	MPU FAULT	START UP OR FIRST SEEK
7	I/O TEST	START UP OR FIRST SEEK
8		
9		
10	MARGINAL MOTOR START	START UP OR FIRST SEEK
11		
12		
13		
14		
15		
16		
17		
18		
19		
20		
21		
22		

^L

DISK CONTROLLER CONNECTORS

CABLE A DISK Not CFDA CFDB

CONTROLLER					DRIVE	
SET CYLINDER	*	C4Z01, C4Y01	---------->		TAG 1	1, 31
SET HEAD ADDRESS	*	C4Z02, C4Y02	---------->		TAG 2	2, 32
CONTROL SELECT	*	C4Z03, C4Y03	---------->		TAG 3	3, 33
BUS 0	*	C4Z04, C4Y04	---------->		BIT 0	4, 34
BUS 1	*	C4Z05, C4Y05	---------->		BIT 1	5, 35
BUS 2	*	C4Z06, C4Y06	---------->		BIT 2	6, 36
BUS 3	*	C4Z07, C4Y07	---------->		BIT 3	7, 37
BUS 4	*	C4Z08, C4Y08	---------->		BIT 4	8, 38
BUS 5	*	C4Z09, C4Y09	---------->		BIT 5	9, 39
BUS 6	*	C4Z10, C4Y10	---------->		BIT 6	10, 40
BUS 7	*	C4Z11, C4Y11	---------->		BIT 7	11, 41
BUS 8	*	C4Z12, C4Y12	---------->		BIT 8	12, 42
BUS 9	*	C4Z13, C4Y13	---------->		BIT 9	13, 43
DEVICE DISABLE		C4Z14, C4Y14	---------->		OPEN CABLE DETECTOR	14, 44
FAULT	*	C4Z15, C4Y15	<----------		FAULT	15, 45
SEEK ERROR	*	C4Z16, C4Y16	<----------		SEEK ERROR	16, 46
ON-CYLINDER	*	C4Z17, C4Y17	<----------		ON CYLINDER	17, 47
INDEX	*	C4Z18, C4Y18	<----------		INDEX	18, 48
UNIT READY	*	C4Z19, C4Y19	<----------		UNIT READY	19, 49
ADDRESS MARK	*	C4Z20, C4Y20	<----------		ADDRESS MARK FOUND	20, 50
NC	#	C4Z21, C4Y21	<----------		BUSY	21, 51
SELECT ENABLE		C4Z22, C4Y22	---------->		UNIT SELECT TAG	22, 52
DEV SEL 0		C4Z23, C4Y23	---------->		UNIT SELECT 0	23, 53
DEV SEL 1		C4Z24, C4Y24	---------->		UNIT SELECT 1	24, 54
SECTOR MARK	*	C4Z25, C4Y25	<----------		SECTOR	25, 55
DEV SEL 2		C4Z26, C4Y26	---------->		UNIT SELECT 2	26, 56
DEV SEL 3		C4Z27, C4Y27	---------->		UNIT SELECT 3	27, 57
WRITE PROTECTED	*	C4Z28, C4Y28	<----------		WRITE PROTECTED	28, 58
NC		C4Z29	---------->		POWER SEQUENCE PICK	29
NC		C4Y29	---------->		POWER SEQUENCE HOLD	59
NC		C4Z30, C4Y30			NOT USE (SPARE)	30, 60

NOTES:

\# DUAL CHANNEL UNITS ONLY.

* GATED BY UNIT SELECTED.

^L

DISK CONTROLLER CONNECTORS

CABLE B

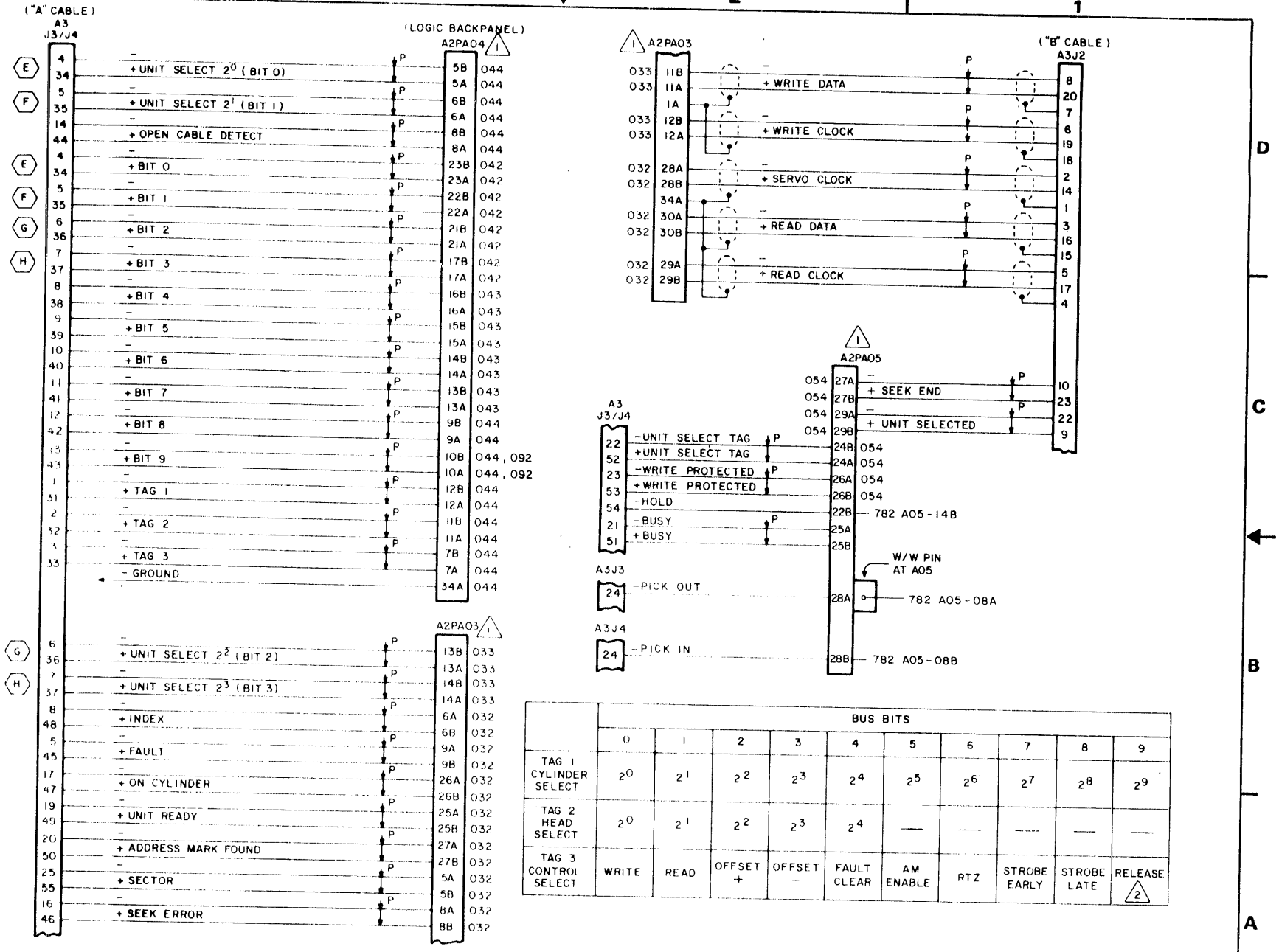
Disk not CFDA CFDB

DRIVE 0 SERVO CLOCK	C5Z02, C5Y01	<----------	SERVO CLOCK	2, 14
DRIVE 0 READ DATA	C5Z03, C5Y03	<----------	READ DATA	3, 16
DRIVE 0 READ CLOCK	C5Z05, C5Y04	<----------	READ CLOCK	5, 17
DRIVE 0 WRITE CLOCK	C5Z06, C5Y06	---------->	WRITE CLOCK	6, 19
DRIVE 0 WRITE DATA	C5Z08, C5Y07	---------->	WRITE DATA	8, 20
DRIVE 0 SELECTED	C5Y09, C5Z09	<----------	UNIT SELECTED	22, 9
DRIVE 0 SEEK END	C5Z10, C5Y10	<----------	SEEK END	10, 23
DRIVE 0 INDEX	C5Z12, C5Y11	<----------	INDEX	12, 24
DRIVE 0 SECTOR	C5Z13, C5Y13	<----------	SECTOR	13, 26
IO GND	C5Z01		GROUND	1
	C5Z04			4
	C5Z07			7
	C5Z11			11
	C5Y02			15
	C5Y05			18
	C5Y08			21
	C5Y12			25

NOTE:
1. 26 CONDUCTOR FLAT CABLE. MAXIMUM - 50 FT.
2. NO SIGNALS GATED BY UNIT SELECTED.

DRIVE 1 SERVO CLOCK	C5Z25, C5Y24	<----------	SERVO CLOCK	2, 14
DRIVE 1 READ DATA	C5Z26, C5Y26	<----------	READ DATA	3, 16
DRIVE 1 READ CLOCK	C5Z28, C5Y27	<----------	READ CLOCK	5, 17
DRIVE 1 WRITE CLOCK	C5Z29, C5Y29	---------->	WRITE CLOCK	6, 19
DRIVE 1 WRITE DATA	C5Z31, C5Y30	---------->	WRITE DATA	8, 20
DRIVE 1 SELECTED	C5Y32, C5Z32	<----------	UNIT SELECTED	22, 9
DRIVE 1 SEEK END	C5Z33, C5Y33	<----------	SEEK END	10, 23
DRIVE 1 INDEX	C5Z35, C5Y34	<----------	INDEX	12, 24
DRIVE 1 SECTOR	C5Z36, C5Y36	<----------	SECTOR	13, 26
IO GND	C5Z01		GROUND	1
	C5Z04			4
	C5Z07			7
	C5Z11			11
	C5Y02			15
	C5Y05			18
	C5Y08			21
	C5Y12			25

NOTES:
1. 26 CONDUCTOR FLAT CABLE. MAXIMUM LENGTH - 50 FT.
2. NO SIGNALS GATED BY UNIT SELECTED.



	BUS BITS									
	0	1	2	3	4	5	6	7	8	9
TAG 1 CYLINDER SELECT	2^0	2^1	2^2	2^3	2^4	2^5	2^6	2^7	2^8	2^9
TAG 2 HEAD SELECT	2^0	2^1	2^2	2^3	2^4	—	—	—	—	—
TAG 3 CONTROL SELECT	WRITE	READ	OFFSET +	OFFSET −	FAULT CLEAR	AM ENABLE	RTZ	STROBE EARLY	STROBE LATE	RELEASE (2)

NOTES

(1) CONNECTOR PLUGS ON TO LOGIC CHASSIS BACKPANEL PINS

(2) DUAL CHANNEL ONLY

CONTROL DATA
NORMANDALE DIVISION

CHANNEL II
I/O HARNESS WIRING AND
TAG / BUS DECODE
(50 PIN)

CODE IDENT 19333 | C | 83322310 | K | B

CROSS REF NO 783 | SHEET 3 | PAGE 3-173/174