

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56

.TITLE CZUDAGO UDA FMTR DATA FILE
.SBTTL USER DOCUMENTATION
.REM ~

IDENTIFICATION

PRODUCT CODE: AC-T355A-MC
PRODUCT NAME: CZUDGAG UDA FMTR DATA FILE
PRODUCT DATE: 27-AUG-82
MAINTAINER: DIAGNOSTIC ENGINEERING
AUTHOR: MATT TEDONE

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C) 1982 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL PDP UNIBUS MASSBUS
DEC DECUS DECTAPE

CZUDAGO UDA FMTR DATA FILE MACRO V04.00 23-AUG-82 14:32:58 PAGE 1-1
 SYMBOL TABLE

. ABS. 000000 000
 000000 001
 ERRORS DETECTED: 0

VIRTUAL MEMGRY USED: 8192 WORDS (32 PAGES)
 DYNAMIC MEMORY AVAILABLE FOR 72 PAGES
 ,B:ZUDGAO=B:ZUDGAO.TTL

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22
 TABLE OF CONTENTS

2-	1	MCALLS
3-	1	EQUATES
5-	1	DATA STRUCTURES
7-	1	MATH SUBROUTINES
12-	1	SDI SUBROUTINES
14-	1	OVERLAY PROCESSING ROUTINES
17-	1	MISCELLANEOUS COMMON ROUTINES
22-	1	DUP DM<->HOST STARTUP OVERLAY
40-	1	INITIALIZATION OVERLAY (G1)
49-	1	DBN/XBN' FORMAT OVERLAY (F1)
52-	1	DBN/XBN TRACK FORMAT OVERLAY (G7)
55-	1	LBN FORMATTING OVERLAY (F2)
58-	1	LBN FORMAT IMAGE SETUP OVERLAY (F8)
60-	1	L/RBN COMPUTE OVERLAY (G8)
65-	1	FCT DOWN-LINE LOAD OVERLAY (F3)
66-	1	RCT UPDATE OVERLAY (F4)
70-	1	RCT READ OVERLAY (H1)
71-	2	FCT->RCT CONVERSION OVERLAY (F5)
76-	1	RCT INITIALIZE OVERLAY (F7)
78-	1	FCT READ OVERLAY (F6)
79-	1	GET FCT BLOCK FOR D/XBN FORMAT (G2)
80-	1	GET FCT BLOCK FOR LBN FORMAT (G3)
81-	1	RCT CLEANUP OVERLAY (G4)
88-	1	FINAL CHECK OVERLAY (H2)
93-	1	FCT WRITE OVERLAY (F9)
94-	1	PBN->D,X,L,RBN CONVERSION OVERLAY (G5)
96-	1	ERROR MESSAGE OVERLAY (G6)

1
2
3 .TITLE UDAF52 - UDA-52 FORMATTER
4 .IDENT /03.01/
5 .NLIST BEX
6
7

8 : COPYRIGHT (C) 1980,1981,1982
9 : DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.
10

11 : THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON A
12 : SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH THE INCLUSION
13 : OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE, OR ANY OTHER
14 : COPIES THEREOF, MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE
15 : TO ANY OTHER PERSON EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE
16 : WHO AGREES TO THESE LICENSE TERMS. TITLE TO AND OWNERSHIP OF
17 : THE SOFTWARE SHALL AT ALL TIMES REMAIN IN DIGITAL.
18

19 : THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT
20 : NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL
21 : EQUIPMENT CORPORATION.
22

23 : DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF
24 : ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
25

26 :
27 : VERSION 01.00
28

29 : M. A. PARENTI 16-MAY-80
30

31 : MODIFIED BY:
32

33 : VERSION 02.00
34

35 : M. A. PARENTI 09-DEC-80
36 : NEW DM INSTRUCTIONS (MEM -> MEM)
37 : CODE OPTIMIZATION
38

39 : 31-MAR-81
40 : M. A. PARENTI
41 : ADDED DOUBLE WORD ADDRESSING FOR OVERLAY ADDRESSES
42

43 : 21-APR-81
44 : M. A. PARENTI
45 : FIX GROUP OFFSET CALCULATION
46

47 : 23-APR-81
48 : M. A. PARENTI
49 : FIX ZERO GROUP PROBLEM
50

51 : 24-APR-81
52 : M. A. PARENTI
53 : FIX LBN GROUP PROBLEM
54 : FIX RECAL WAIT PROBLEM
55
56
57

58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114

05-MAY-81
M. A. PARENTI
FIX SIZE PROBLEM FOR RA81

13-MAY-81
M. A. PARENTI
ADD LIMITED DHP FUNCTIONALITY

15-MAY-81
M. A. PARENTI
FIX SUBUNIT MASK PROBLEM

15-MAY-81
M. A. PARENTI
ONLY WRITE NON-PAD BLOCKS OF FCT

15-MAY-81
M. A. PARENTI
FIX COMPUTATION OF NON-PAD FCT BLOCKS

15-MAY-81
M. A. PARENTI
FIX BLOCK ZERO FCT PROBLEM

28-MAY-81
M. A. PARENTI
FIX SUBUNIT PROBLEMS
WRITE ONLY NON-PAD BLOCKS OF RCT

01-JUN-81
M. A. PARENTI
FIX DOUBLE COMPARE PROBLEM

08-JUN-81
M. A. PARENTI
FIX EXISTING FCT FORMAT PROBLEM

17-JUN-81
M. A. PARENTI
FIX SUBUNIT WRITE PROTECT PROBLEM

17-JUN-81
M. A. PARENTI
ADD STATUS UPDATES AT VARIOUS PLACES

22-JUN-81
M. A. PARENTI
FIX RBN STARTING BITS PROBLEM

22-JUN-81
M. A. PARENTI
FIX NON-PAD RCT INITIALIZE PROBLEM

115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171

06-JUL-81
M. A. PARENTI
FIX STATISTICS COUNT OF BAD BLOCKS TO NOT INCLUDE RBN

18-AUG-81
M. A. PARENTI
FIX CLEARING OF ECC THRESHOLD

16-SEP-81
M. A. PARENTI
KLUDGE RECIR AND LAST BITS FOR UDA BUG

6-OCT-81
M. A. PARENTI
FIX JUSTIFICATION ON RESPONSES

19-OCT-81
M. A. PARENTI
FIX ECC CHECKING AND MINOR FIXES TO QUESTIONS

13-NOV-81
M. A. PARENTI
FIX CONVERSION BLOCK, CLEANUP COUNT
CHECK FOR NO CORRECTION IF 0 THRESHOLD

20-NOV-81
M. A. PARENTI
FIX PROBLEM WITH BAD RBN WHEN PRIMARY IN FCT
ADD HEAD VERIFICATION ROUTINE AND MAKE FINAL CHECK
ROUTINES A SEPARATE OVERLAY

25-JAN-82
M. A. PARENTI
CHECK FOR VALID NUMERICS IN USER RESPONSES
CHECK FOR UNIBUS ERRORS
MODIFY LONG TIMEOUT
CHECK FOR VALID STATUS IN GETUNT

09-FEB-82
M. A. PARENTI
FIX VAX DATE ROUTINE

18-FEB-82
M. A. PARENTI
CHANGE RETRY RECOVERY ON NON-CHECK PASS READS/Writes TO NOT
ISSUE A RECAL/RESEEK SEQUENCE AFTER RECOVERY LEVEL 0

10-MAR-82
M. A. PARENTI
CONVERT TO UDA52 XFC'S AND MAKE 512/576

24-MAR-82
M. A. PARENTI
FIX PCOM OFFSET PROBLEM
FIX BAD RBN WITH PRIMARY IN FCT PROBLEM

1
2
3
4
5

.SBTTL MCALLS
.MCALL DMCODE,DMEND,DMOVLY
.MCALL JMP,BR,BEQ,CALL,BPL,BCC,BNE,BMI,RETURN
.MCALL DMODT

UD
DA

```

1          .SBTTL EQUATES
2          EQUATES
3          :
4          :
5          :
6          :
7          :
8          :
9          :
10         000000      PROD      =      0          ;PRODUCTION (1)
11         :          :          :          ;NOT PRODUCTION (0)
12         :          :          :          ;IF DEFINED THEN DEBUG MODE
13         :
14         :          :
15         000000      FT.BUF   =      0.          ;BUFFER POINTER OFFSET
16         000001      FT.LOW   =      1.          ;LOW ORDER HEADER OFFSET
17         000002      FT.HI    =      2.          ;HI ORDER HEADER OFFSET
18         000003      FT.FLA   =      3.          ;FLAG WORD
19         :
20         :
21         :
22         :
23         :
24         :
25         :
26         :          :
27         000000      RW.STAT  =      0.          ; STATUS (12-15), NEXT BUFR PTR (0-14)
28         000001      RW.ER1   =      1.          ; ALSO USED AS ECC ERROR INDICATOR
29         000006      RW.DUM   =      6.          ; POINTER TO DUMMY SDI CONTRL BLOCK
30         000002      RW.BUF   =      2.          ; POINTER TO DATA BUFFER
31         000003      RW.LOW   =      3.          ; 1ST HEADER WORD (LO ORDER LBN)
32         000004      RW.HI    =      4.          ; 2ND HEADER WORD (HI ORDER LBN)
33         000000      RW.CMD   =      5.          ; SDI RT CMD (8-15), HEAD ADDR (0-7)
34         000400      RW.EDC   =      256.        ; 1ST WORD OF 256 WORD DATA BUFFER
35         000440      RW.E76   =      288.        ; EDC FOR 512 MODE
36         000401      RW.ER2   =      257.        ; EDC FOR 576 MODE
37         :          :
38         :          :
39         :          :
40         :          :
41         :          :
42         :          :
43         :          :
44         000000      RB.LOW   =      0.          ;LOW ORDER BLOCK NUMBER
45         000001      RB.HI    =      1.          ;HIGH ORDER BLOCK NUMBER
46         000002      RB.CMD   =      2.          ;READ COMMAND AND TRACK NUMBER
47         000003      RB.IM    =      3.          ;IMAGE COUNTER
48         :
49         :
50         :          :
51         000000      BREAK   =      0.          ;BREAKPOINT XFC CODE
52         000001      FORMAT  =      1.          ;FORMAT TRACK XFC CODE
53         000002      READ    =      2.          ;READ N SECTORS XFC CODE
54         000003      WRITE   =      3.          ;WRITE N SECTORS XFC CODE
55         000004      SEND    =      4.          ;SEND SDI COMMAND XFC CODE
56         000005      RCV     =      5.          ;RECEIVE SDI MESSAGE XFC CODE
57         000006      CMPDAT  =      6.          ;COMPARE DATA PATTERN XFC CODE
    
```


58	000007	STATUS	=	7.	:RETURN DRIVE STATUS XFC CODE
59	000010	ECHO	=	8.	:ECHO DATA TO DRIVE XFC CODE
60	000011	DINIT	=	9.	:DRIVE INITIALIZE XFC CODE
61	000012	SIP	=	10.	:WAIT FOR SECTOR/INDEX PULSE XFC CODE
62	000013	UREAD	=	11.	:READ UNIBUS MEMORY XFC CODE
63	000014	UWRITE	=	12.	:WRITE UNIBUS MEMORY XFC CODE
64	000015	ECC	=	13.	:DO ECC ON BUFFER XFC CODE
65	000016	MAINTR	=	14.	:SEND MAINT READ DATA XFC CODE
66	000017	MAINTW	=	15.	:RECEIVE MAINT WRITE DATA XFC CODE
67	000020	CVT	=	16.	:CONVERT TO PHYSICAL ADDRESS XFC CODE
68	000021	DONE	=	17.	:TERMINATE DM PROGRAM XFC CODE
69	000022	UPDATE	=	18.	:UPDATE DUP PROGRESS INDICATOR XFC
70		:			
71		:			
72		:			
73	000000	SHORTO	=	0.	:SHORT TIME OUT
74	000001	FRCPY	=	1.	:NUMBER OF F/RCT COPIES
75	000001	RTRY	=	1.	:NUMBER OF RETRIES
76	000001	LONGTO	=	1.	:LONG TIMEOUT
77	000001	SSBIT	=	1.	:SS BIT (SECTOR SIZES ALLOWABLE)
78	000002	ERRSYM	=	2.	:NUMBER OF ALLOWABLE ECC ERRORS
79	000002	ERCV	=	2.	:ERROR RECOVERY LEVELS SUPPORTED
80	000007	REVSEC	=	7.	:REVS/SECOND
81	000011	OFFS12	=	9.	:GROUP OFFSET 512 BYTE
82	000015	OFFS76	=	13.	:GROUP OFFSET 576 BYTE
83	000000	CYLBN	=	0.	:CYLINDERS IN LBN AREA
84	000002	STLBN	=	2.	:HIGH ORDER STARTING LBN
85	000003	STRBN	=	3.	:HIGH ORDER STARTING RBN
86	000002	STXBN	=	2.	:HIGH ORDER STARTING XBN
87	000003	STDBN	=	3.	:HIGH ORDER STARTING DBN
88	000001	STCYL	=	1.	:HIGH ORDER STARTING CYLINDER
89	000011	LBNT12	=	9.	:NUMBER OF LBNS PER TRACK (512)
90	000015	LBNT76	=	13.	:NUMBER OF LBNS PER TRACK (576)
91	000004	RBNTRK	=	4.	:NUMBER OF RBNS PER TRACK
92	000021	XBNCYL	=	17.	:NUMBER OF CYLINDERS IN XBN AREA
93	000022	DBNCYL	=	18.	:NUMBER OF CYLINDERS IN DBN AREA
94	000012	LBNH12	=	10.	:NUMBER OF LBN'S IN HOST AREA (512)
95	000016	LBNH76	=	14.	:NUMBER OF LBN'S IN HOST AREA (576)
96	000002	GRPCYL	=	2.	:GROUPS/CYLINDER
97	000003	TRKGRP	=	3.	:TRACKS/GROUP
98	000010	FCTSZ	=	8.	:FCT SIZE IN SECTORS
99	000014	RCTS12	=	12.	:RCT SIZE IN LBN'S (512)
100	000020	RCTS76	=	16.	:RCT SIZE IN LBN'S (576)
101	000005	DATA	=	5.	:DATA PREAMBLE SIZE
102	000005	HEAD	=	5.	:HEADER PREAMBLE SIZE
103		:			
104		:			
105		:			
106		:			
107		:			
108	000001	BIT0	=	000001	
109	000002	BIT1	=	000002	
110	000004	BIT2	=	000004	
111	000010	BIT3	=	000010	
112	000020	BIT4	=	000020	
113	000040	BIT5	=	000040	
114	000100	BIT6	=	000100	

GET CHARACTERISTICS OFFSETS

BIT DEFINITIONS

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 3-2
EQUATES

UDA
MAT

115	000200	BIT7	=	000200	
116	000400	BIT8	=	000400	
117	001000	BIT9	=	001000	
118	002000	BIT10	=	002000	
119	004000	BIT11	=	004000	
120	010000	BIT12	=	010000	
121	020000	BIT13	=	020000	
122	040000	BIT14	=	040000	
123	100000	BIT15	=	100000	
124		:			
125		:			
126		:			
127		:			
128		:			
129		:			
130	177400	HIBYTE	=	177400	:HIGH BYTE MASK
131	000377	LOBYTE	=	000377	:LOW BYTE MASK
132	177700	HI2BYTE	=	177700	:HIGH BYTE PLUS 2 BITS
133	177600	HI1BYTE	=	177600	:HIGH BYTE PLUS 1 BIT
134	007777	LO	=	007777	:ALL BUT HEADER CODE
135	177760	FCLR	=	177760	:CLEAR FOR FRCPY
136	170377	STCLR	=	170377	:CLEAR FOR STARTING BITS
137	007777	BUFMSK	=	007777	:BUFFER CLEAR MASK
138	000004	VLD	=	BIT2	:STATUS VALID BIT(1=VALID)
139	000010	VLD1	=	BIT3	:STATUS VALID BIT(1=VALID)
140	000200	PARITY	=	BIT7	:STATUS PARITY BIT(1=PARITY ERROR)
141	000400	PARIT1	=	BIT8	:REAL TIME ERROR(1=ERROR)
142	100000	SS	=	BIT15	:SECTOR SIZE BIT (0=512 ONLY,1=512/576)
143		:			
144		:			
145		:			
146	000175	UNSEC	=	000175	:UNSUCCESSFUL COMPLETION
147		:			
148		:			
149		:			
150	000000	HD.LBN	=	000000	:GOOD LBN
151	060000	HD.RBN	=	060000	:GOOD RBN, PERHAPS UNUSED
152	030000	HD.REV	=	030000	:REVECTORED LBN
153	110000	HD.BAD	=	110000	:BAD BLOCK
154	050000	HD.PRIV	=	050000	:PRIMARY REVECTORED BLOCK
155	170000	HD.CLR	=	170000	:CLEAR HDR LODE
156	140000	HD.DBN	=	140000	:GOOD DBN
157	120000	HD.XBN	=	120000	:GOOD XBN
158	100000	PRMY	=	BIT15	:PRIMARY BIT IN FCT
159	010000	FBDHD	=	BIT12	:BAD HEADER CODE IN FCT
160		:			
161		:			
162		:			
163	000000	RC.FRE	=	000000	:FREE REPLACEMENT BLOCK
164	020000	RC.PRIV	=	020000	:PRIMARY REVECTOR
165	030000	RC.SND	=	030000	:SECONDARY REVECTOR
166	040000	RC.UNU	=	040000	:BAD REPLACEMENT BLOCK
167	100000	RC.NUL	=	100000	:NULL(FILL) BLOCK
168		:			
169		:			
170		:			
171	100000	RWRDY	=	BIT15	:READ/WRITE READY BIT POSITION

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 3-3
EQUATES

UDA
MAT

```

172      000002      ATTN      =      BIT1      :ATTENTION
173      000001      RCVRDY   =      BIT0      :RECEIVER READY
174      :
175      :
176      :
177      :
178      :
179      000001      ST.RU    =      BIT0      :RUN/STOP SWITCH 1=IN
180      000002      ST.PS    =      BIT1      :PORT SWITCH 1=IN
181      000040      ST.DR    =      BIT5      :DIAGNOSTIC REQUESTED 1=YES
182      170000      ST.WP    =      BIT12+BIT13+BIT14+BIT15 :WRITE PROTECT SWITCH SU:0,1 1=IN
183      000020      ST.SR    =      BIT4      :SPINDLE READY 1=YES
184      001000      ST.DB    =      BIT9      :DIAG CYL ACCESS ENABLED 1=YES
185      002000      ST.FO    =      BIT10     :FORMAT CYL ACCESS ENABLED 1=YES
186      000004      ST.IN    =      BIT2      :DRIVE INITIALIZED 1=YES
187      000010      ST.WE    =      BIT3      :WRITE ERROR (WRITE LOCKED)
188      000020      ST.DF    =      BIT4      :DIAG FAILED - CANNOT DRIVE CLEAR
189      000374      ST.ERR   =      000374  :COMBINED CLEARABLE ERRORS BITS SET
190      000002      ST.ERB   =      2.      :ERROR BYTE OFFSET (3RD WORD)
191      :
192      :
193      :
194      :
195      :
196      000023      OVCNT    =      19.     :NUMBER OF OVERLAYS
197      000003      OVLEN    =      3.      :LENGTH OF 1 OVERLAY BLOCK
198      000000      LEN      =      0.      :WORD COUNT OF OVERLAY
199      000001      HSTLO    =      1.      :LOW ORDER UNIBUS ADDRESS
200      000002      HSTHI    =      2.      :HI ORDER UNIBUS ADDRESS
201      000000      F1       =      0.      :OFFSET INTO TABLE
202      000003      F2       =      3.      :SECOND OVERLAY OFFSET INTO TABLE
203      000006      F3       =      6.      :THIRD OVERLAY OFFSET INTO TABLE
204      000011      F4       =      9.      :FOURTH OVERLAY OFFSET INTO TABLE
205      000014      F5       =      12.     :FIFTH OVERLAY OFFSET INTO TABLE
206      000017      F6       =      15.     :SIXTH OVERLAY OFFSET INTO TABLE
207      000022      F7       =      18.     :SEVENTH OVERLAY OFFSET INTO TABLE
208      000025      F8       =      21.     :EIGHTH OVERLAY
209      000030      F9       =      24.     :NINTH OVERLAY
210      000033      G2       =      27.     :ELEVENTH OVERLAY
211      000036      G3       =      30.     :TWELVTH OVERLAY
212      000041      G4       =      33.     :THIRTEENTH OVERLAY
213      000044      G5       =      36.     :FOURTEENTH OVERLAY
214      000047      G7       =      39.     :SIXTEENTH OVERLAY
215      000052      G8       =      42.     :SEVENTEENTH OVERLAY
216      000055      H1       =      45.     :NINETEENTH OVERLAY
217      000060      G1       =      48.     :TENTH OVERLAY
218      000063      G6       =      51.
219      000066      H2       =      54.
220      :
221      :
222      :
223      :
224      :
225      000001      FCTAVL   =      BIT0      :FCT AVAILABLE
226      000010      DBN      =      BIT3      :FORMAT DBN AREA
227      000100      REVECT   =      BIT6      :REVECTOR FLAG
228      001000      PRIM      =      BIT9      :PRIMARY FOUND FLAG

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 3-4
EQUATES

UDAI
MATI

```

229      000002      FCTEMT =      BIT1      ;FCT EMPTY FLAG
230      000020      GOBAD  =      BIT4      ;DO BEST GUESS IF FCT BAD
231      000040      RCINIT  =      BIT5      ;RCT LAST BLOCK FIXED UP
232      000004      FCTBAD  =      BIT2      ;FCT FOUND BAD (FOR STATS)
233      000200      MANU    =      BIT7      ;MANUFACTURING FORMAT
234      000400      DLL     =      BIT8      ;DOWN-LINE LOAD FLAG
235      002000      BSTGS   =      BIT10     ;BEST GUESS FORMAT
236      004000      NDLL    =      BIT11     ;ONLY WRITE FCT SCRATCH
237      020000      INIRCT  =      BIT13     ;INIT RCT FLAG
238      040000      FINI    =      BIT14     ;FORMAT FINISHED FLAG
239      010000      CHRDN   =      BIT12     ;CHARACTERISTICS DONE FLAG
240      100000      RTY     =      BIT15     ;RETRY FLAG
241      :
242      :
243      :      FLAG1  EQUATES
244      :
245      :
246      000001      WP      =      BIT0      ;WRITE PPROTECT FLAG
247      000002      RTYDN  =      BIT1      ;RETRY DONE ON THIS SECTOR
248      000004      RPRIM  =      BIT2      ;FLAG FOR PRIMARY GOOD EDC
249      000010      ERDN   =      BIT3      ;FLAG FOR ERROR EXIT TRY
250      000020      DEAD   =      BIT4      ;HOST GONE FLAG
251      000040      BDHD   =      BIT5      ;BAD HEADER ON CHECK PASS READ
252      000100      RCINDN =      BIT6      ;RCT INIT DONE (WITH ONE FULL PAD BLK)
253      000200      QUESDN =      BIT7      ;STARTUP QUESTIONS FINISHED
254      000400      FLIPON =      BIT8      ;FLIP FLAG FOR CONVERSIONS
255      001000      REPEAT =      BIT9      ;REPEAT QUESTION FLAG FOR STARTUP
256      002000      GTFLAG =      BIT10     ;FLAG FOR GETB MACRO
257      004000      STFLAG =      BIT11     ;FLAG FOR STOB MACRO
258      010000      BDTST  =      BIT12     ;FLAG FOR TEST OF BAD HEADER IN VERHD
259      020000      MODE   =      BIT13     ;FORMAT MODE (0=512,1=576)
260      040000      FPRIM  =      BIT14     ;PRIMARY IN FCT
261      :
262      :
263      :      PHYSICAL CONVERSION XFC BLOCK EQUATES
264      :
265      000000      V1     =      0        ;CYLINDER PARAMETER
266      000002      V2     =      2        ;BLOCK NUMBER PARAMETER
267      000004      V3     =      4        ;BLOCKS PER TRACK PARAMETER
268      000005      V4     =      5        ;ONLY FOR RBN'S
269      000006      CYL    =      6        ;CYLINDER RETURNED
270      000010      GRP    =      8        ;GROUP RETURNED
271      000011      TRK    =      9        ;TRACK RETURNED
272      000012      STSC   =      10       ;STARTING SECTOR RETURNED
273      000013      INDSEC =      11       ;SECTOR FROM INDEX
274      :
275      :
276      :      DMBUF OFFSETS
277      :
278      :
279      000016      DMBUFL =      14.      ;BUFFER LENGTH
280      :
281      :
282      :
283      :      FCT BLOCK OFFSETS
284      :
285      :

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 3-5
EQUATES

```

286      000002      FSER      =      2      ;SERIAL NUMBER
287      000001      INST      =      1      ;FORMATIN INSTANCE NUMBER
288      000016      C512      =      14.     ;COUNT OF USED 512 ENTRIES IN FCT
289      000020      C576      =      16.     ;COUNT OF USED 576 ENTRIES IN FCT
290      000012      FDAT      =      10.     ;MOST RECENT FORMAT DATE
291      000025      FCTFLG    =      21.     ;FCT FLAG FOR GOOD/BAD FCT
292      100000      NOFCT     =      BIT15   ;FLAG - 0 - FCT GOOD
293                                     ;      1 - FCT KNOW BAD
294      :
295      :
296      :           RCT BLOCK OFFSETS
297      :
298      :
299      000000      RSER      =      0      ;SERIAL NUMBER OFFSET
300      :
301      :           MISC   DEFINITIONS
302      :
303      :
304      :
305      000006      TWOB      =      6.     ;LENGTH OF 2 IMAGE ENTRIES
306      000011      THREB     =      9.     ;LENGTH OF 3 IMAGE ENTRIES
307      000004      RDLEN     =      4.     ;LENGTH OF CHECK PASS READ BLOCK
308      000002      ERLN      =      2.     ;LENGTH OF REVECTOR TABLE ENTRY
309      000004      REVLEN    =      4.     ;LENGTH OF SECONDARY TABLE
310      013400      RWCMD     =      013400 ;SDI READ COMMAND
311      122400      WRCMD     =      122400 ;SDI WRITE COMMAND
312      100000      RDCMD     =      100000 ;SIGNAL TO XFC NO MORE BLOCKS
313      010000      ECCF      =      BIT12   ;ECC ERROR BIT
314      000200      RBNRPT    =      128.   ;NUMBER OF RBN COPIES IN REVECTOR
315      000400      SECS16    =      256.   ;SECTOR SIZE IN WORDS FOR 512 BYTE
316      000440      SECS18    =      288.   ;SECTOR SIZE IN WORDS FOR 576 BYTE
317      000003      IMLEN     =      3.     ;LENGTH OF IMAGE BLOCK
318      100000      BD        =      BIT15   ;BAD FLAG IN IMAGE BUFFER
319      100000      LAST      =      BIT15   ;LAST FLAG IN IMAGE BUFFER
320      040000      RECIR     =      BIT14   ;RECIRCULATE IN FORMAT IMAGE BUFFER
321      126736      M512      =      126736 ;FCT MODE INDICATOR FOR 512
322      074161      M576      =      074161 ;FCT MODE INDICATOR FOR 576
323      100000      TIMVAL    =      32768. ;TIMER LOOP VALUE
324      000010      MAXTRY    =      8.     ;FINAL SECONDARY WRITE RETRY LIMIT
325      001362      DUPOVL    =      OVE.MN-2 ;OVERLAY STARTING ADDRESS FROM DUP
326      000040      LOBL      =      00040   ;BLANK IN LOW ORDER BYTE
327      020040      BLANWD    =      20040   ;WORD WITH 2 ASCII BLANKS
328      :
329      :
330      :           STATUS OFFSETS
331      :
332      :
333      000000      MASK      =      0      ;SUBUNIT OFFSET MASK
334      000000      UID       =      0      ;UNIT NUMBER OFFSET
335      :
336      :
337      :           BUFFER DEFINITIONS
338      :           BUFFERS ARE 301 WORDS LONG
339      :
340      010000      BUF1      =      010000   ;BUFFER 1
341      010455      BUF2      =      BUF1+301. ;BUFFER 2
342      011132      BUF3      =      BUF2+301. ;BUFFER 3

```

UDAF02 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 3-6
EQUATES

343	011607	BUF4	=	BUF3+301.	:BUFFER 4
344	012264	BUF5	=	BUF4+301.	:BUFFER 5
345	013022	BUF6	=	BUF5+350.	:BUFFER 6
346	013477	BUF7	=	BUF6+301.	:BUFFER 7
347	014154	BUF8	=	BUF7+301.	:BUFFER 8
348	014631	BUF9	=	BUF8+301.	:BUFFER 9
349	015306	BUF10	=	BUF9+301.	:BUFFER 10
350	015763	BUF11	=	BUF10+301.	:BUFFER 11
351	016747	SODT	=	BUF11+500.	:START OF ODT
352		:			
353		:			
354		:			
355	010000	RDBUF	=	BUF1	:READ/WRITE BUFFER
356	010455	PBNBUF	=	BUF2	:BUFFER OF BAD PBN'S
357	011132	GDBLK	=	BUF3	:DATA FOR GOOD SECTOR
358	011607	PRMBUF	=	BUF4	:DATA PATTERN FOR PRIMARY REVECTOR
359	012264	REVBUF	=	BUF5	:SECONDARY REVECTOR BUFFER
360	013022	CMDBUF	=	BUF6	:READ COMMAND BUFFERS
361	013477	RCTBUF	=	BUF7	:RCT BLOCK BUFFER
362	014154	RBNBUF	=	BUF8	:RBN FORMAT BUFFER
363	014631	BDLST	=	BUF9	:HEAD VERIFICATION BUFFER
364	015306	CLBUF	=	BUF10	:USED IN FINAL CLEANUP
365	015763	IMAGE	=	BUF11	:FORMAT IMAGE BUFFER
366					:BUFFER EXCESS AFTER FORMAT IMAGE
367					:IS USED TO HOLD BLOCKS TO BE
368					:REVECTORED. MAX BLOCKS BEFORE
369					:REVECTOR ROUTINE IS CALLED VARIES
370					:WITH THE SIZE OF THE FORMAT BUFFER AREA
371	037777	BMAX	=	37777	:MAX BUFFER ADDRESS
372					
373					
374					

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 5
DATA STRUCTURES

1			.SBTTL DATA STRUCTURES	
2			:	
3	000000		DMCODE UDAFM,0,1364,13,255.	
4	001364	000000	.WORD 0	:DMODT ADDRESS
5	001365		ENTRY: JMP START	:JUMP TO START LOCATION
6			:	
7			DATA STRUCTURES	
8			:	
9			RETRY COUNTERS	
10			:	
11	001367	000000	UN.ERR: .WORD 0	:UNSUCCESSFUL CMD RETRY CNTR
12	001370	000000	UN.ERT: .WORD 0	:TRANSMISSION ERROR RETRY CNTR
13	001371	000000	UN.ERI: .WORD 0	:INITIALIZATION ERROR RETRY CNTR
14	001372	000000	UN.SEK: .WORD 0	:SEEK RETRY COUNT
15			:	
16			READ COMMAND BLOCK	
17			:	
18			:	
19			WRBLK:	
20	001373		RDBLK: .WORD 100000	:STATUS POINTER
21	001373	100000	.WORD 0	:POINTER TO DATA BUFFER
22	001374	000000	.WORD 0	:FIRST WORD OF EXPECTED HDR
23	001375	000000	.WORD 0	:SECOND WORD
24	001376	000000	.WORD 0	:REAL-TIME SDI COMMAND
25	001377	000000	.WORD 0	:POINTER TO SDI BLOCK
26	001400	000000	.WORD 0	
27			:	
28			DUMMY DOUBLE WORDS AND DUMMY SDI COMMAND	
29			:	
30	001401	000200	HSLIM: .WORD 200	:HEADER CMP LIMIT
31	001402	001520	.WORD SCR-5	:POINTER TO SUBUNIT CHAR
32	001403	000000	DDUMMY: .WORD 0	:DUMMY DOUBLE WORD FOR ONE
33	001404	000000	.WORD 0	:BYTE OPERAND CONVERSION
34	001405		TEMP2:	:ALSO USE AS TEMP
35	001405		MULPC: .BLKW 2	:MULTIPLICATION BUFFER
36	001407	000000	.WORD 0	:RESERVED LOCATION (A+7)
37	001410		OFFSET:	:FOR EASIER REFERENCE
38	001410		TEMP: .BLKW 2	:USED FOR COMPUTATIONS
39			:	
40			:	
41			CURRENT UDA PORT	
42			:	
43			:	
44	001412	000000	UNIT: .WORD 0	:SDI INTERCONNECT
45	001413	000000	UNNO: .WORD 0	:UNIT NUMBER ENTERED
46			:	
47			MESSAGE TABLES	
48			:	
49	001414		CR.GST: MSG GST,1,ST,7	:GET STATUS
50	001420		CR.GCR: MSG GCR,1,CR,11	:GET CHARACTERISTICS
51	001424		CR.GSR: MSG GSR,2,SCR,19	:GET SUBUNIT CHARACTERISTICS
52	001430		CR.DIS: MSG DIS,2,ST,6	:UNLOAD DRIVE
53	001434		CR.RUN: MSG RUN,1,ST,6	:LOAD DRIVE
54	001440		CR.ACC: MSG ACC,3,ST,6	:SET FORMAT ACCESS
55	001444		CR.CLR: MSG DCLR,2,ST,6	:DRIVE CLEAR
56	001450		CR.SEK: MSG ISEEK,6,ST,6	:SEEK
57	001454		CR.RCL: MSG IRECAL,1,ST,6	:RECALIBRATE

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 5-1
DATA STRUCTURES

58	001460	CR.ERV: MSG	ERECOV,2,ST,6	:ERROR RECOVERY COMMAND
59	001464	CR.ONL: MSG	ONLINE,2,ST,6	:ONLINE COMMAND
60		:		
61		:		
62		:	MESSAGES AND COMMANDS	
63	001470	GST: .WORD	000011*256.	:GET STATUS COMMAND
64	001471	GCR: .WORD	000207*256.	:GET CHARACTERISTICS
65	001472	GSR: .WORD	00210*256.	:GET SUBUNIT CHARACTERISTICS
66	001473		0	:SUBUNIT MASK
67	001474	DIS: .WORD	000204*256.	:UNLOAD DRIVE
68	001475		0	:NO SPIN DOWN MODIFIER
69	001476	RUN: .WORD	000014*256.	:INITIATE LOAD
70	001477	ACC: .WORD	000201*256.	:ACCESS DIAG AND FMT CYL
71	001500		3006	:MASK BYTE/MODE BYTE
72	001501	DCLR: .WORD	000005*256.	:DRIVE CLEAR
73	001502		374	:BITS TO CLEAR
74	001503	ST: .BLKW	7	:STATUS MESSAGE BUFFER
75	001512	CR: .BLKW	11.	:CHARACTERISTICS MESSAGE BUFF
76	001525	SCR: .BLKW	19.	:SUBUNIT CHARACTERISTICS BUFF
77	001550	ISEEK: .WORD	000012*256.	:INITIATE SEEK
78	001551		0	:..
79	001552		0	:..
80	001553		0	:..
81	001554	IRECAL: .WORD	000216*256.	:INITIATE RECAL
82	001555	ERECOV: .WORD	000006*256.	:ERROR RECOVERY COMMAND
83	001556		0	:RECOVERY LEVEL
84	001557	ONLINE: .WORD	000213*256.	:ONLINE COMMAND
85	001560		377	:COMMAND TIMEOUT (SECS)
86		:		
87		:		
88		:	DISK LOCATION POINTERS	
89	001561	CURRBN: .WORD	0	:CURRENT RBN
90	001562		0	:..
91	001563	CURPBN: .WORD	0	:CURRENT PBN
92	001564		0	:..
93	001565	CUPTRK: .WORD	0	:CURRENT TRACK
94	001566	CURBN: .WORD	0	:CURRENT BLOCK NUMBER
95	001567		0	:..
96	001570	CURLBN: .WORD	0	:FOR RCT INIT
97	001571		0	:..
98	001572	CURXBN: .WORD	0	:CURRENT XBN NUMBER
99	001573		0	:..
100	001574	STASEC: .WORD	0	:FOR HEAD VERIFICATION ROUTINE
101	001575		0	:..
102	001576	HOLDBN: .WORD	0	:BLOCK NUMBER OF FIRST BLOCK ON CYL
103	001577		0	:..
104	001600	HOLRBN: .WORD	0	:BLOCK NUM OF FIRST RBN ON CYLINDER
105	001601		0	:..
106	001602	HOLDPN: .WORD	0	:PBN OF FIRST SECTR ON TRACK
107	001603		0	:..
108	001604	CYLNUM: .WORD	0	:CURRENT CYLINDER NUMBER
109	001605		0	:..
110	001606	SECTRK: .WORD	0	:SECTORS/TRACK (CURRENT VALUE)
111	001607		0	:..
112	001610	SECT76: .WORD	0	:SECTORS/TRACK 576
113	001611		0	:..
114	001612	SECT12: .WORD	0	:SECTORS/TRACK 512

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 5-2
DATA STRUCTURES

115	001613	000000							
116	001614	000000	SECTCY:	.WORD	0				: SECTORS/CYLINDER
117	001615	000000		.WORD	0				: "
118	001616		LBNLBN:	.BLKW	2				: LBN'S IN LBN AREA
119	001620		RBNLBN:	.BLKW	2				: RBN'S IN LBN AREA
120	001622		XBNSEC:	.BLKW	2				: SECTORS IN LBN AREA
121	001624		TRKCYL:	.BLKW	2				: TRACKS/CYLINDER
122	001626		LBNCYL:	.BLKW	2				: NUMBER OF LBN CYLINDERS
123	001630		LBNPCY:	.BLKW	2				: LBN'S/CYLINDER
124	001632		RBNPCY:	.BLKW	2				: RBN'S/CYLINDER
125	001634	000000	REVRBN:	.WORD	0				: REVECTORED RBN NUMBER
126	001635	000000		.WORD	0				: "
127	001636	000000	CUROVL:	.WORD	0				: CURRENT OVERLAY
128	001637		HGHPBN:	.BLKW	2				: HIGHEST PBN IN LBN AREA
129									
130									
131									
132									
133									
134									
135									
136	001641			.BLKW	31.				: STACK
137	001700	000000	STACK:	.WORD	0				: TOP OF STACK
138	001701	000000	STCKSV:	.WORD	0				: STACK PTR TEMP SAVE
139									
140									
141									
142									
143									
144	001702	000000	FLAG:	.WORD	0				: FLAG WORD
145	001703	000000	FLAG1:	.WORD	0				: FLAG WORD
146	001704	000000	ERFLAG:	.WORD	0				: RE-FORMAT FLAG
147	001705	000000	WRFLG:	.WORD	0				: RCT WRAP FLAG
148	001706	000000	BADPBN:	.WORD	0				: POINTER TO PBNTAB ENTRY
149	001707	000000	ERRBUF:	.WORD	0				: POINTER TO BEGINNING OF REVECTOR BUFFER
150	001710	000000	EMAX:	.WORD	0				: MAX NUMBER OF REVECTORS BEFORE
151									: RCT UPDATE ROUTINE IS CALLED
152	001711	000000	ERR:	.WORD	0				: NUMBER OF SECTORS IN ERROR
153	001712	000000	HOLD:	.WORD	0				: DOUBLE WORD TEMP STORAGE
154	001713	000000		.WORD	0				
155	001714	000000	EIMAGE:	.WORD	0				: ADDRESS OF END IMAGE BLOCK
156	001715	000000	STARIT:	.WORD	0				: STARTING ADDRESS OF THIS PASS
157	001716	000000	SKPCNT:	.WORD	0				: OFFSET FOR FIRST READ CHECK
158	001717	000000	TBLK:	.WORD	0				: INTERLEAVE
159									: 6 - BI-LEAVE
160									: 9 - TRI-LEAVE
161	001720		RCTTOT:						: ALSO RCT TOTAL HOLDING AREA
162	001720	004704	CUTOF:	.WORD	2500.				: SECT/SECOND CUTOFF
163	001721	000000		.WORD	0				: DOUBLE WORD
164	001722	000000	FCNT:	.WORD	0				: COUNT OF USED FCT ENTRIES FOR FORMATTING
165	001723		FCTSUB:	.BLKW	2				: SIZE OF FCT SUBTABLE
166	001725		FCTFMT:	.BLKW	2				: SIZE OF ONE FCT COPY
167	001727		RCTFMT:	.BLKW	2				: SIZE OF ONE RCT COPY
168	001731	000000	FCTCPY:	.WORD	0				: NUMBER OF FCT COPIES
169	001732	000000	NEXT1:	.WORD	0				: MULTI-COPY COUNTER
170	001733	000105	INI:	.WORD	69.				: INITIAL VALUE FOR EDC
171	001734	000400	CNT:	.WORD	SECSI6				: COUNT FOR EDC

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 5-3
DATA STRUCTURES

172	001735	000100	LTO:	.WORD	100	:LONG TIMEOUT
173	001736	002000	STO:	.WORD	1024.	:SHORT TIMEOUT (IN MILLESECS)
174	001737	000000	ERPNT:	.WORD	0	:REVECTOR LIST POINTER
175	001740	000000	BUFPNT:	.WORD	0	:BUFFER POINTER FOR FCT READ
176	001741	000000	REVCNT:	.WORD	0	:REVECTOR COUNT
177	001742	000000	FCTPTR:	.WORD	0	:POINT TO CURRENT LOCATION IN FCT LBOXC
178	001743	000001	FCTCNT:	.WORD	1	:CURRENT FCT BLOCK
179	001744	000000		.WORD	0	
180	001745	000000	FCTNPD:	.WORD	0	:NON-PAD FCT BLOCKS
181	001746	000000	RCTLBN:	.WORD	0	:LBN'S IN RCT
182	001747	000000	MNCNT:	.WORD	0	:USED FCT ENTRIES
183	001750	000016	DMBUF:	.REPT	14.	:MAINTENANCE BUFFER
184				.WORD	0	:MAKE SURE IT IS 0
185				.ENDM		
186	001766		DATE:	.BLKW	4	:DATE BUFFER
187	001772		SERNUM:	.BLKW	4	:SERIAL NUMBER
188	001776	000000	FCTREV:	.WORD	0	:FCT ENTRIES AT CERTAIN POINTS
189	001777	000000	LBNBAD:	.WORD	0	:TOTAL REVECTORED LBN'S
190	002000	000000	RCTBAD:	.WORD	0	:TOTAL BAD RCT BLOCKS
191	002001	000000	DBBAD:	.WORD	0	:TOTAL DBN BAD BLOCKS
192	002002	000000	XBBAD:	.WORD	0	:TOTAL LBN BAD BLOCKS
193	002003	060001	FCMSG:	.WORD	60001	:DUP CODE
194	002004	000000	IMSTAR:	.WORD	0	:POINTER TO START OF IMAGE
195	002005	000000	HPREA:	.WORD	0	:HEADER PREAMBLE LENGTH
196	002006	000000	DPREA:	.WORD	0	:DATA PREAMBLE LENGTH
197	002007	000000	ST.LBN:	.WORD	0	:STARTING LBN BITS
198	002010	000000	ST.RBN:	.WORD	0	:STARTING RBN BITS
199	002011	000000	ST.XBN:	.WORD	0	:STARTING XBN BITS
200	002012	000000	ST.DBN:	.WORD	0	:STARTING DBN BITS
201						
202						
203						
204						
205						
206						
207						
208						
209	002013	001231	OVLTBL:	.WORD	OVL.F1	:LENGTH OF FIRST OVERLAY
210	002014	015156		.WORD	OVS.F1	:LOW ORDER HOST ADDRESS
211	002015	000000		.WORD	0	:HIGH ORDER HOST ADDRESS
212	002016	001660		.WORD	OVL.F2	:LENGTH OF SECOND OVERLAY
213	002017	020714		.WORD	OVS.F2	:LOW ORDER HOST ADDRESS
214	002020	000000		.WORD	0	:HIGH ORDER HOST ADDRESS
215	002021	000465		.WORD	OVL.F3	:LENGTH OF THIRD OVERLAY
216	002022	027330		.WORD	OVS.F3	:LOW ORDER HOST ADDRESS
217	002023	000000		.WORD	0	:HIGH ORDER HOST ADDRESS
218	002024	000776		.WORD	OVL.F4	:LENGTH OF FOURTH OVERLAY
219	002025	030502		.WORD	OVS.F4	:LOW ORDER HOST ADDRESS
220	002026	000000		.WORD	0	:HIGH ORDER HOST ADDRESS
221	002027	001411		.WORD	OVL.F5	:LENGTH OF FIFTH OVERLAY
222	002030	033512		.WORD	OVS.F5	:LOW ORDER HOST ADDRESS
223	002031	000000		.WORD	0	:HIGH ORDER HOST ADDRESS
224	002032	000407		.WORD	OVL.F6	:LENGTH OF SIXTH OVERLAY
225	002033	040056		.WORD	OVS.F6	:LOW ORDER HOST ADDRESS
226	002034	000000		.WORD	0	:HIGH ORDER HOST ADDRESS
227	002035	000551		.WORD	OVL.F7	:LENGTH OF SEVENTH OVERLAY
228	002036	036534		.WORD	OVS.F7	:LOW ORDER HOST ADDRESS

OVERLAY POINTERS
NOTE:

WHEN ADDING AN ENTRY TO THIS TABLE EQUATE
OVCNT MUST BE INCREMENTED

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 5-4
DATA STRUCTURES

229	002037	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
230	002040	000725	.WORD	OVL.F8	:LENGTH OF EIGHTH OVERLAY
231	002041	024454	.WORD	OVS.F8	:LOW ORDER HOST ADDRESS
232	002042	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
233	002043	000234	.WORD	OVL.F9	:LENGTH OF NINTH OVERLAY
234	002044	051676	.WORD	OVS.F9	:LOW ORDER HOST ADDRESS
235	002045	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
236	002046	000053	.WORD	OVL.G2	:LENGTH OF TENTH OVERLAY
237	002047	041074	.WORD	OVS.G2	:LOW ORDER HOST ADDRESS
238	002050	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
239	002051	000307	.WORD	OVL.G3	:LENGTH OF ELEVENTH OVERLAY
240	002052	041222	.WORD	OVS.G3	:LOW ORDER HOST ADDRESS
241	002053	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
242	002054	002142	.WORD	OVL.G4	:LENGTH OF TWELFTH OVERLAY
243	002055	042040	.WORD	OVS.G4	:LOW ORDER HOST ADDRESS
244	002056	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
245	002057	000460	.WORD	OVL.G5	:LENGTH OF THIRTEENTH OVERLAY
246	002060	052366	.WORD	OVS.G5	:LOW ORDER HOST ADDRESS
247	002061	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
248	002062	000426	.WORD	OVL.G7	:LENGTH OF FOURTEENTH OVERLAY
249	002063	017640	.WORD	OVS.G7	:LOW ORDER HOST ADDRESS
250	002064	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
251	002065	000401	.WORD	OVL.G8	:LENGTH OF FIFTEENTH OVERLAY
252	002066	026326	.WORD	OVS.G8	:LOW ORDER HOST ADDRESS
253	002067	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
254	002070	000406	.WORD	OVL.H1	:LENGTH OF SIXTEENTH OVERLAY
255	002071	032476	.WORD	OVS.H1	:LOW ORDER HOST ADDRESS
256	002072	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
257	002073	001531	.WORD	OVL.G1	:LENGTH OF SEVENTEENTH OVERLAY
258	002074	011674	.WORD	OVS.G1	:LOW ORDER HOST ADDRESS
259	002075	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
260	002076	000525	.WORD	OVL.G6	:LENGTH OF EIGHTEENTH OVERLAY
261	002077	053526	.WORD	OVS.G6	:LOW ORDER HOST ADDRESS
262	002100	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
263	002101	001555	.WORD	OVL.H2	:LENGTH OF NINETEENTH OVERLAY
264	002102	046344	.WORD	OVS.H2	:LOW ORDER HOST ADDRESS
265	002103	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
266	002104		.WORD	4	:FOR BUFFER OVERLAYS

OVLBLK: .BLKW

1					
2					
3					
4					
5					
6	002110		CONBLK: .BLKW 12.		;CONVERSION BUFFER
7					
8					
9					
10					
11	002124	000377	NUM: .WORD 255.		;NUMBER OF WORDS IN PATTERN
12	002125	010001	CBUF: .WORD RDBUF+1		;BUFFER TO COMPARE(NOT FIRST WORD)
13	002126	155555	FWRD: .WORD 155555		;FIRST WORD OF PATTERN
14	002127	133333	SWRD: .WORD 133333		;SECOND WORD OF PATTERN
15	002130	066666	TWRD: .WORD 066666		;THIRD WORD OF PATTERN
16	002131	177777	DWRD: .WORD 177777		;DIAGNOSTIC WORD(FIRST IN SECTOR)
17					
18					
19	002132	030206	EDC: .WORD 30206		;EDC FOR ABOVE DATA PATTERN
20	002133	147571	BADEDC: .WORD 147571		;FORCED ERROR EDC FOR ABOVE
21	002134	111014	EDC76: .WORD 111014		;EDC FOR 576 DATA PATTERN
22	002135	066763	BADE76: .WORD 066763		;FORCED ERROR EDC FOR ABOVE
23					
24					
25					
26					
27	002136	000000	ERRCNT: .WORD 0		;FOR TESTING VERIFICATION
28	002137	000000	SECCNT: .WORD 0		;SECTOR COUNT
29	002140	000000	N: .WORD 0		;NUMBER OF ORIGINAL CHECK PASS READ
30	002141	000000	N1: .WORD 0		;NUMBER OF ERROR READS
31	002142	000000	NN1: .WORD 0		;DITTO
32	002143	000000	CNTCYL: .WORD 0		;NUMBER OF CYLINDERS TO FURMAT
33	002144	000000			
34	002145	000000	HD.CUR: .WORD 0		;CURRENT HEADER
35	002146	000000	CURGRP: .WORD 0		;CURRENT GROUP
36	002147	000000	GRPCNT: .WORD 0		;NUMBER OF GROUPS TO DO
37	002150	000000	TRKCNT: .WORD 0		;NUMBER OF TRACKS TO DO
38	002151	000001	ONE: .WORD 1		;WORD CONSTANT OF 1
39	002152	000000			;DOUBLE WORD
40	002153	000002	TWOC: .WORD 2		;WORD CONSTANT OF 2
41	002154	000000			;DOUBLE WORD
42	002155	000000	SNDCNT: .WORD 0		;COUNT OF SECONDARY REVECTORS
43	002156	000000	RTYCNT: .WORD 0		;COUNT OF SECTORS RETRYED
44	002157		CURPNT: .WORD 0		;POINT FOR HEAD VERIFICATION
45	002157	000000	UPDPNT: .WORD 0		;POINTER FOR RCT UPDATE
46	002160	000000	TOTRCT: .WORD 0		;TOTAL LBN'S IN RCT'S
47	002161	000000			
48	002162	000000	RCTCNT: .WORD 0		;CURRENT RCT BLOCK
49	002163	000000	PCNT: .WORD 0		;PBN BLOCK COUNTER
50	002164	000000	COUNT: .WORD 0		;COUNT FOR XBN DLL
51	002165	000005	RETRY: .WORD 5		;RETRIES FROM SDI
52	002166	000000	RECOV: .WORD 0		;RECOVERY LEVELS SUPPORTE BY DRIVER
53	002167	000000	TMPTRY: .WORD 0		;TEMP FOR RETRY COUNT
54	002170	000000	RECTMP: .WORD 0		;TEMP FOR ERROR RECOVERY LEVEL
55	002171	000000	SECSIZ: .WORD 0		;CURRENT SECTOR SIZE

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 7
MATH SUBROUTINES

1			.SBTTL MATH SUBROUTINES	
2				
3				
4			SUBROUTINES	
5				
6				
7			DOUBLE ADD ROUTINE	
8				
9			INPUT PARAMETERS	
10				
11			R3 CONTAINS POINTER TO OPERAND 1	
12				
13			R4 CONTAINS POINTER TO OPERAND 2	
14				
15			OUTPUT PARAMETER	
16				
17			R4 CONTAINS THE RESULT	
18				
19	002172		DADD: PUSH R5	;SAVE A SCRATCH REGISTER
20	002173		PUSH R1	;SAVE ANOTHER
21	002174	104235	MOV (R3)+,R5	;GET LOW ORDER OPERAND
22	002175	104131	MOV (R3),R1	;GET HIGH ORDER OPERAND
23	002176	105245	ADD (R4)+,R5	;ADD LOW ORDER OPERAND
24	002177		BCC DADD1	;BRANCH IF NO CARRY
25	002201	115401	INC R1	;ADD ONE TO HIGH IF CARRY
26	002202	105141	DADD1: ADD (R4),R1	;ADD OP 2
27	002203	100141	MOV R1,(R4)	;SAVE HIGH ORDER
28	002204	100445	MOV R5,-(R4)	;SAVE LOW ORDER
29	002205		POP R1	;RESTORE R1
30	002206		POP R5	;RESTORE R5
31	002207	117403	DEC R3	;RESTORE R3
32	002210		RETURN	
33				

1					
2	002212		DSUB:		
3				: ++	
4				: DOUBLE PRECISION FIXED POINT SUBTRACT ROUTINE	
5				: INPUTS:	
6				: R3 = POINTER TO OPERAND 1 (SUBTRAHEND)	
7				: R4 = POINTER TO OPERAND 2 (MINUEND)	
8				: OUTPUT:	
9				: R4 = POINTER TO RESULT WHERE (R4) = (R4) - (R3)	
10				: --	
11					
12					
13					
14	002212			PUSH R1,R5	: SAVE REGISTERS
15	002214	104245		MOV (R4),R5	: GET LO ORDER MINUEND
16	002215	104141		MOV (R4),R1	: GET HI ORDER MINUEND
17	002216	107135		SUB (R3),R5	: SUBTRACT LOW ORDER OPERANDS
18	002217			BCC 10\$: POSITIVE RESULT
19	002221	117401		DEC R1	: BORROW FROM HI ORDER OPERAND
20	002222	107631	000001	SUB 1(R3),R1	: SUBTRACT HI ORDER OPERANDS
21	002224	100141		MOV R1,(R4)	: STORE HI ORDER RESULT
22	002225	100445		MOV R5,-(R4)	: STORE LO ORDER RESULT
23	002226			POP R5,R1	: RESTORE REGISTERS
24	002230			RETURN	
25					

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 9
MATH SUBROUTINES

UD
MI

```

1 002232          DMUL:
2
3                : ++
4                : DOUBLE PRECISION FIXED POINT MULTIPLY ROUTINE
5                : INPUTS:
6                :       R3 = POINTER TO MULTIPLIER (SINGLE PRECISION)
7                :       R4 = POINTER TO MULTIPLICANT (DOUBLE PRECISION)
8                : OUTPUT:
9                :       R4 = POINTER TO RESULT WHERE (R4) = (R4) * (R3)
10               : --
11
12 002232          PUSH    R0,R3                ; SAVE R0 & R3
13 002234 104137   MOV     (R3),R0              ; GET MULTIPLIER
14 002235          BNE    5$                    ; MULTIPLIER NOT = 0
15 002237 100147   MOV     R0,(R4)              ; LOAD LO ORDER RESULT
16 002240 100647 000001  MOV    R0,1(R4)          ; LOAD HI ORDER RESULT
17 002242          BR     20$                    ; RETURN
18 002244 104140 001405 5$:  MOV    (R4),MULPC      ; COPY MULTIPLICANT FOR DADD
19 002246 104640 000001 001406  MOV    1(R4),MULPC+1
20 002251 104203 001405  MOV    #MULPC,R3
21 002253 117407 10$:  DEC     R0                ; ADJUST MULTIPLIER FOR *1
22 002254          BEQ    20$                    ; MULTIPLIER = 0, EXIT
23 002256          CALL   DADD                    ; PERFORM ITERATIVE ADDS
24 002260          BR     10$
25 002262 20$:  POP     R3,R0                ; RESTORE R0 & R3
26 002264          RETURN
27

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 10
MATH SUBROUTINES

UDA
MIS

```

1 002266          DDIV:
2
3                : ++
4                : DOUBLE PRECISION FIXED POINT DIVIDE
5                : INPUTS:
6                :   R3 = POINTER TO DIVISOR (SINGLE PRECISION)
7                :   HIGH ORDER WORD MUST BE ZERO
8                :   R4 = POINTER TO DIVIDENT (DOUBLE PRECISION)
9                : OUTPUT:
10               :   R3 = POINTER TO REMAINDER
11               :   R4 = POINTER TO QUOTIENT
12               :
13               : NOTE - THE CASES WHERE EITHER THE DIVISOR OR DIVIDENT ARE ZERO,
14               : ARE NOT CONSIDERED IN THIS ROUTINE.
15               : --
16
17 002266          PUSH    R0,R1,R2,R5          ; SAVE REGISTERS
18 002272 114007    CLR     R0                  ; CLR LO ORDER QUOTIENT REG
19 002273 114001    CLR     R1                  ; CLR HI ORDER QUOTIENT REG
20 002274 104132    MOV     (R3),R2            ; GET DIVISOR
21 002275 104645 000001 10$: MOV     1(R4),R5        ; GET HI ORDER DIVIDENT
22 002277          BNE     20$                  ; DIVISOR NOT = 0
23 002301 104145    MOV     (R4),R5            ; GET LO ORDER DIVIDENT
24 002302 106052 15$: CMP     R5,R2            ; IS DIVIDENT < DIVISOR ?
25 002303          BCC     20$                  ; NO, CONTINUE
26 002305          BR      30$                  ; YES, STOP
27 002307          CALL   DSUB                    ; SYNTHESIZE DIVIDE
28 002311 105207 000001 ADD     #1,R0                  ; INCR LO ORDER QUOTIENT
29 002313          BCC     10$                  ; DID NOT OVERFLOW
30 002315 115401    INC     R1                  ; ADJUST HI ORDER QUOTIENT
31 002316          BR      10$
32 002320 104145 30$: MOV     (R4),R5            ; GET REMAINDER
33 002321 100147    MOV     R0,(R4)            ; LOAD LO ORDER QUOTIENT
34 002322 100641 000001 MOV     R1,1(R4)            ; LOAD HI ORDER QUOTIENT
35 002324 100135    MOV     R5,(R3)            ; LOAD REMAINDER
36 002325          POP     R5,R2,R1,R0          ; RESTORE REGISTERS
37 002331          RETURN
38

```


UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 11
MATH SUBROUTINES

UDA
MIS

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14 002333
15 002334
16 002335 104141
17 002336 104647 000001
18 002340 106637 000001
19 002342
20 002344
21 002346
22 002347
23 002350 106204 000000
24 002352
25 002354
26 002355
27 002356 106204 077777
28 002360
29 002362 106131
30 002363
31 002365
32 002366
33 002367 106044
34 002370
35 002372 106131
36 002373
37 002375

```

```

.....
DOUBLE COMPARE
INPUT PARAMETERS
R3     CONTAINS A POINTER TO THE FIRST OPERAND
R4     CONTAINS A POINTER TO THE SECOND OPERAND
OUTPUT PARAMETERS
THE FLAGS ARE SET AS IF A SINGLE PRECISION 'CMP' HAD OCCURED
DCMP:  PUSH    R0           ;SAVE R0 FOR USE AS SCRATCH
        PUSH    R1           ;SAVE R1 FOR USE AS SCRATCH
        MOV     (R4),R1      ;GET LOW ORDER DEST OPERAND
        MOV     1(R4),R0     ;GET HIGH ORDER DEST OPERAND
        CMP     1(R3),R0     ;DO ACTUAL HIGH ORDER TEST
        BEQ    DCMP1        ;GO DO ADDITIONAL TESTING
        BCC    DCMP2        ;SRC HI, CLEAN UP AND RTN
DCMP4:  POP     R1           ;RESTORE R1
        POP     R0           ;RESTORE R0
        CMP     #0,R4       ;SET CONDITION CODES - SRC LSS
        RETURN          ;AND RETURN
DCMP2:  POP     R1           ;RESTORE R1
        POP     R0           ;RESTORE R0
        CMP     #077777,R4  ;SET CONDITION CODES - DST LSS
        RETURN          ;AND RETURN
DCMP1:  CMP     (R3),R1     ;TEST LOW ORDER
        BNE    DCMP3        ;BRANCH IF NOT EQUAL
        POP     R1           ;RESTORE R1
        POP     R0           ;RESTORE R0
        CMP     R4,R4       ;SET CONDITION CODES - EQUAL
        RETURN          ;AND RETURN
DCMP3:  CMP     (R3),R1     ;COMPARE AGAIN
        BCC    DCMP2        ;BRANCH ON SRC HI
        BR     DCMP4        ;BRANCH ON SRC LOW

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 12
SDI SUBROUTINES

.SBTTL SDI SUBROUTINES

GET STATUS

OUTPUT PARAMETERS

CLEARS DRIVE STATUS AND GETS CHARACTERISTICS
IF NOT ALREADY RECEIVED

```

1
2
3
4
5
6
7
8
9
10
11
12 002377          GSTATS: PUSH      R3          ;PUSH R3
13 002400          PUSH      R5          ;PUSH R5
14 002401 104203 001414  STATST: MOV      #CR.GST,R3      ;POINT TO GET STATUS TABLE
15 002403          CALL      TALK        ;GET STATUS
16 002405 104207 001503          MOV      #ST,R0        ;POINT TO SUBUNIT CHARACTERISTICS
17 002407 104673 000002          MOV      ST.ERB(R0),R3      ;GET ERROR BYTE
18 002411 103203 177420          BIC      #ST.DF+HIBYTE,R3      ;CLEAR HIGH BYTE AND DF BIT
19 002413 115003          TST      R3          ;ANY NEED TO ISSUE DRIVE CLEAR ?
20 002414          BEQ      STSK1        ;NOPE - SKIP IT
21 002416 104030 001502          MOV      R3,DCLR+1        ;STORE MASK IN DRIVE CLR COMMAND
22 002420          CALL      CLEAR        ;DO A DRIVE CLEAR
23 002422 104205 001504  STSK1:  MOV      #ST+1,R5      ;POINT TO FIRST WORD OF STATUS
24 002424 104253          MOV      (R5)+,R3      ;GET FIRST WORD OF STATUS
25 002425 104202 000001          MOV      #1,R2          ;ERROR SUBCODE IN CASE
26 002427 102203 000040          BIT      #ST.DR,R3      ;IS DRIVE IN DIAGNOSTIC REQUEST MODE
27 002431          BNE      STPNIC        ;YES, WE LOSE
28 002433 115402          INC      R2          ;ERROR SUBCODE 2
29 002434 102203 000001          BIT      #ST.RU,R3      ;IS RUN STOP SWITCH OUT
30 002436          BEQ      STPNIC        ;YES, LOSE AGAIN
31 002440 104202 000004          MOV      #4,R2          ;SUBCODE
32 002442 102203 000002          BIT      #ST.PS,R3      ;PORT SWITCH OUT ?
33 002444          BEQ      STPNIC        ;YES - DIE PAINFULLY
34 002446 104032          MOV      R3,R2          ;GET STATUS MODE BYTE
35 002447 110702          SWAB     R2          ;SWITCH WRITE PROTECT TO LOW BYTE
36 002450 102302 001473          BIT      GSR+1,R2      ;THIS SUBUNIT WRITE PROTECTED ?
37 002452          BEQ      SRCK          ;IF NOT CHECK IF SPINNING
38 002454 104202 000003          MOV      #3,R2          ;IN CASE IT'S FATAL
39 002456 102200 000001 001703  BIT      #WP,FLAG1      ;BEEN HERE ONCE ?
40 002461          BNE      STPNIC        ;YUP - GIVE UP
41 002463 101200 000001 001703  BIS      #WP,FLAG1      ;SET BEEN HERE FLAG
42 002466          CALL     ACCESS        ;TRY TO RESET IT
43 002470          BR       STATST        ;AND SEE IF IT WORKED
44 002472 102203 000020  SRCK:  BIT      #ST.SR,R3      ;IS PACK SPINNING?
45 002474          BNE      STFORM        ;YES, TEST FOR FORMAT ENABLE
46 002476          CALL     LOAD          ;NO, SPIN PACK
47 002500          JMP      STATST        ;SEE IF ANYTHING CHANGED
48 002502 102203 002000  STFORM: BIT      #ST.FO,R3      ;IS FORMATTING ENABLED?
49 002504          BNE      STDIAG        ;YES, TEST FOR DIAG ACCESS
50 002506          CALL     ACCESS        ;NO, SET UP DIAG/FORM ACCESS
51 002510          JMP      STATST        ;SEE IF ANYTHING CHANGED
52 002512 102203 001000  STDIAG: BIT      #ST.DB,R3      ;IS DIAG CYL ACCESS ALLOWED
53 002514          BNE      STWLK        ;YES, CHECK FOR ERRORS
54 002516          CALL     ACCESS        ;NO, SET UP DIAG/FORM ACCESS
55 002520          JMP      STATST        ;SEE IF ANYTHING CHANGED
56 002522 104153          STWLK: MOV      (R5),R3      ;GET SECOND STATUS WORD
57 002523 102203 000010          BIT      #ST.WE,R3      ;ANY WRITE ENABLE ERRORS

```


UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 13
SDI SUBROUTINES

```

1
2
3
4
5
6
7
8
9
10
11
12
13 002563          TALK:  PUSH  R3
14 002564          PUSH  R4
15 002565 104663 000001 LOOP1:  MOV   1(SP),R3      ;RESTORE R3 FOR RETRIES
16 002567 104237          MOV   (R3)+,R0      ;GET COMMAND ADDRESS
17 002570 104231          MOV   (R3)+,R1      ;GET COMMAND SIZE
18 002571 104302 001412          MOV   UNIT,R2      ;MAKE SURE HAVE INTERCONNECT
19 002573 060004          XFC   SEND      ;SEND GET STATUS COMMAND
20 002574 115001          TST   R1      ;SUCCESSFUL?
21 002575          BEQ   MSG1      ;YES, BRANCH
22 002577 115400 001370          INC   UN.ERT      ;INCREMENT ERROR COUNT
23 002601 104201 000002          MOV   #2,R1      ;ERROR NUMBER IN CASE
24 002603          BR    TCLEAR      ;DO RECOVERY
25 002605 102200 100000 001702 MSG1:  BIT   #RTY,FLAG      ;IN A RETRY ?
26 002610          BNE   LOOP2      ;YES - DON'T CLEAR COUNTER
27 002612 114000 001370          CLR   UN.ERT      ;FOR RESET
28 002614 104231          MOV   (R3)+,R1      ;POINT TO RCV BUFFER
29 002615 104137          MOV   (R3),R0      ;SET SIZE OF REPLY
30 002616 104302 001412          MOV   UNIT,R2      ;MAKE SURE HAVE INTERCONNECT
31 002620 060005          XFC   RCV      ;RCV REPLY TO GET STATUS
32 002621 115001          TST   R1      ;SUCCESSFUL?
33 002622          BEQ   TALKDN      ;YES, CHECK STATUS
34 002624 115400 001367          INC   UN.ERR      ;INCREMENT ERROR COUNT
35 002626 104201 000004          MOV   #4,R1      ;ERROR CODE IN CASE
36 002630          BR    TCLEAR      ;DO RECOVERY
37 002632 106207 000175          TALKDN: CMP  #UNSEC,R0      ;WAS CMD UNSUCCESSFUL?
38 002634          BNE   TALKRT      ;YES, DONE
39 002636 115400 001367          INC   UN.ERR      ;INCREMENT ERROR COUNT
40 002640 104201 000003          MOV   #3,R1      ;ERROR CODE IN CASE
41 002642          BR    TCLEAR      ;NO, TRY AGAIN
42 002644 102200 100000 001702 TALKRT: BIT  #RTY,FLAG      ;IN A RETRY ?
43 002647          BNE   TALKP      ;YUP - SKIP CLEAR
44 002651 114000 001367          CLR   UN.ERR      ;CLEAR FOR REST
45 002653          TALKP: POP  R4      ;RESTORE R4
46 002654          POP  R3      ;RESTORE R3
47 002655          RETURN
48 002657 114002          ERRT:  CLR   R2      ;CLEAR SUBCODE
49 002660          CALL  ERRMNT      ;ERROR EXIT
50 002662 102200 100000 001702 TCLEAR: BIT  #RTY,FLAG      ;IN A RETRY ?
51 002665          BNE   TALKIP      ;YUP - SKIP FLAG SET AND STACK SAVE
52 002667 101200 100000 001702          BIS  #RTY,FLAG      ;SET FLAG
53 002672 104060 001701          MOV   SP,STCKSV      ;SAVE STACK POINTER
54 002674          BR    TALIP1      ;SKIP RETRY HANDLING
55 002676 104306 001701          TALKIP: MOV  STCKSV,SP      ;RESTORE STACK POINTER
56 002700 106300 002165 001370 TALIP1: CMP  RETRY,UN.ERT      ;DONE RETRIES ?
57 002703          BMI  ERRT      ;YUP - CAN IT

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 3-1
SDI SUBROUTINES

```

58 002705 106300 002165 001367      CMP      RETRY,UN.ERR      :OVER THE LIMIT ?
59 002710                               BMI      ERRT             :YUP
60 002712 104302 001412              MOV      UNIT,Rc         :GET UNIT
61 002714 060011                       XFC      DINIT           :INIT THE DRIVE
62 002715                               CALL     STATVL          :TST DRIVER STATUS VALIDITY
63 002717                               BNE      TERR            :IF NOT ZERO - NO GOOD
64 002721                               CALL     TIMER           :WAIT ANOTHER 2 SECONDS
65 002723                               CALL     TIMER           :TO MAKE SURE DRIVER HAS ENOUGH TIME
66 002725                               CALL     STATVL          :GET VALID STATUS AGAIN
67 002727                               BNE      TERR            :IF NO GOOD - ERROR
68 002731 102201 000001              BIT      #RCVRDY,R1      :IS RECEIVER READY SET
69 002733                               BNE      TATTN1          :YES - ALL SET
70 002735 104201 000024      TERR:  MOV      #20.,R1    :SET ERROR CODE
71 002737 114002                       CLR      R2              :CLEAR SUBCODE
72 002740                               CALL     ERRMNT          :DIE PEACEFULLY
73 002742                               CALL     GSTATS         :GET STATUS AND CLEAR ERRORS
74 002744 103200 100000 001702      BIC      #RTY,FLAG      :CLEAR RETRY FLAG
75 002747                               BR       LOOP1           :AND TRY AGAIN
76                                     :
77                                     :
78                                     :
79 002751                               :
80 002752 104203 001454      RECAL:  PUSH     R3        :SAVE R3
81 002754                               MOV      #CR.RCL,R3     :POINT TO RECAL TABLE
82                                     :
83                                     :
84                                     :
85 002756                               :
86 002757 114000 001371      LOAD:   PUSH     R3        :SAVE R3
87 002761 104203 001434      LOADS:  CLR      UN.ERI    :FOR INIT
88 002763                               MOV      #CR.RUN,R3     :POINT TO LOAD DRIVE TABLE
89 002764 104137                               R4        :SAVE R4
90 002765 104631 000001      MOV      (R3),R0       :GETCOMMAND ADDRESS
91 002767 104302 001412      MOV      1(R3),R1      :GET COMMAND SIZE
92 002771 060004                               MOV      UNIT,R2       :GET INTERCONNECT
93 002772 115001                               XFC      SEND          :ISSUE GET STATUS COMMAND
94 002773                               TST      R1            :SUCCESSFUL ?
95 002775 115400 001371      BEQ     LOAD2          :YUP - SKIP RETRY
96 002777 106300 002165 001371      INC     UN.ERI        :INC COUNT
97 003002                               CMP      RETRY,UN.ERI  :DONE ALL RETIES ?
98 003004                               BMI      LOADER        :YUP
99 003006 114000 001371      BR      LOAD5          :
100 003010 104304 001735      LOAD2:  CLR      UN.ERI  :FOR ERROR CLEAR
101 003012 104631 000002      MOV     LTO,R4        :LONG TIMEOUT VALUE (SECONDS)
102 003014 104637 000003      LOAD3:  MOV     2(R3),R1 :GET RECEIVE BUFFRE
103 003016 104302 001412      MOV     3(R3),R0     :GET BUFFER LENGTH
104 003020 060005                               MOV     UNIT,R2      :GET INTERCONNECT
105 003021 115001                               XFC     RCV          :RECEIVE SDI RESPONSE
106 003022                               TST     R1           :SUCCESSFUL ?
107 003024 117404                               BEQ     LOAD4        :YUP - SKIP RETRY
108 003025                               DEC     R4           :DECREMENT COUNTER
109 003027                               BNE     LOAD3        :LOOP TILL DONE
110 003031      LOAD4:  BR      LOADER    :IF NOT DONE YET - THEN ERROR
111 003032                               POP     R4           :RESTORE R4
112 003033                               POP     R3           :RESTORE R3
113 003035 104201 000003      RETURN :RETURN TO CALLER
114 003037 114002      LOADER: MOV     #3,R1   :ERROR CODE
                               CLR     R2             :CLEAR SUBCODE

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 13-2
SDI SUBROUTINES

```

115 003040          CALL   ERRMNT          :DIE
116                :
117                :
118                :
119 003042          ACCESS: PUSH   R3          :SAVE R3
120 003043 104203 001440      MOV   #CR.ACC,R3      :POINT TO ACCESS TABLE
121 003045          CALL   TALK          :SEND ACCESS CMD
122 003047          POP    R3          :RESTORE R3
123 003050          RETURN          :RETURN TO CALLER
124                :
125                :
126                :
127 003052          CLEAR:  PUSH   R3          :SAVE R3
128 003053 104203 001444      MOV   #CR.CLR,R3      :POINT TO CLEAR TABLE
129 003055          CALL   TALK          :SEND CLEAR CMD
130 003057          POP    R3          :RESTORE R3
131 003060          RETURN          :RETURN TO CALLER
132                :
133                :
134                :
135 003062          SEEK:   PUSH   R3          :SAVE R3
136 003063          PUSH   R0          :SAVE R0
137 003064 104302 001412      SEEK0: MOV   UNIT,R2      :MAKE SURE HAVE UNIT
138 003066 104203 001450      MOV   #CR.SEK,R3      :POINT TO SEEK TABLE
139 003070          CALL   TALK          :SEND SEEK COMMAND
140 003072 104303 001736      MOV   STO,R3      :SHORT TIMEOUT
141 003074          SEEK1: CALL   STATVL      :CHECK FOR STATUS VALIDITY
142 003076          BNE   SEEK5          :IF NOT ZERO - DIE
143 003100 102201 000002      BIT   #ATTN,R1      :ANY PROBLEMS
144 003102          BNE   SEEK2          :YES, BRANCH
145 003104 102201 100000      BIT   #RWRDY,R1      :NO, DONE?
146 003106          BNE   SEEK6          :ALL DONE
147 003110 117403          DEC    R3          :DECREMENT COUNTER
148 003111          BEQ   SEEK3          :IF ZERO THEN DEAD
149 003113 104207 000240      MOV   #160.,R0      :1MS DELAY
150 003115 117407          SEEK7: DEC    R0          :DECREMENT COUNTER
151 003116          BNE   SEEK7          :DELAY LOOP
152 003120          BR    SEEK1          :TRY AGAIN
153 003122 114001          SEEK6: CLR    R1          :CLEAR ERROR FLAG
154 003123 114000 001372      SEEK4: CLR   UN.SEK      :FOR RESET
155 003125          POP    R0          :YES, RESTORE R0
156 003126          POP    R3          :RESTORE R3
157 003127          RETURN          :RETURN TO CALLER
158 003131          SEEK5: CALL   INITPT      :INIT THE DRIVE
159 003133 115400 001372      SEEK2: INC   UN.SEK      :INCREMENT RETRY COUNTER
160 003135 106300 002165 001372  CMP   RETRY,UN.SEK  :HAVE WE DONE ALL RETRIES?
161 003140          BEQ   SEEK3          :YES, PANIC
162 003142          CALL   GSTATS          :PANIC AND CALL GET STATUS
163 003144          CALL   RECAL          :RECAL DRIVE
164 003146          BR    SEEK0          :AND TRY AGAIN
165 003150 104201 177775      SEEK3: MOV   #-3,R1      :SET ERROR CODE
166 003152          BR    SEEK4          :RESTORE REGS AND RETURN
167                :
168                :
169                :
170 003154          DISCON: PUSH   R3          :SAVE R3
171 003155 104203 001430      MOV   #CR.DIS,R3      :DISCONNECT WITH

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 13-3
SDI SUBROUTINES

```

172 003157          CALL    TALK          :SEND UNLOAD CMD
173 003161          POP     R3           :RESTORE R3
174 003162          RETURN          :RETURN TO CALLER
175                :
176                :
177                :
178 003164          ONLIN:  PUSH    R3           :SAVE R3
179 003165          104203 001464      MOV     #CR.ONL,R3      :ONLINE COMMAND
180 003167          CALL    TALK          :BRING DRIVE ONLINE
181 003171          POP     R3           :RESTORE R3
182 003172          RETURN          :RETURN TO CALLER
183                :
184                :
185                :
186 003174          INITIT: PUSH    R1           :SAVE R1
187 003175          PUSH    R3,R4        :SAVE R3 AND R4
188 003177          104204 000001      MOV     #1,R4         :START WITH PORT 0
189 003201          104203 000004      MOV     #4,R3         :INIT PORT COUNTER
190 003203          104042          MOV     R4,R2         :SET UP INTERCONNECT
191 003204          060011          XFC     DINIT         :INIT DRIVE
192 003205          104207 066540      MOV     #28000.,R0    :TIMER (APPROX 2 SECS)
193 003207          ATTN1:  CALL    STATVL        :CHECK STATUS VALIDITY
194 003211          BNE     AOUT         :IF NOT ZERO - NO GOOD
195 003213          117407          DEC     R0           :DEC COUNT
196 003214          BEQ     AOUT         :IF ZERO THEN DEAD
197 003216          102201 000001      BIT     #RCVRDY,R1    :IS RECEIVER READY SET ?
198 003220          BEQ     ATTN1        :NO, TRY AGAIN
199 003222          110204          AOUT:  ROL     R4           :NEXT PORT
200 003223          117403          DEC     R3           :DECREMENT COUNTER
201 003224          BNE     INIT5        :IF NOT DONE DO NEXT PORT
202 003226          POP     R4,R3        :RESTORE R3 AND R4
203 003230          POP     R1           :RESTORE R1
204 003231          RETURN          :AND RETURN TO CALLER
205                :
206                :
207                :
208                :
209 003233          104302 001412      INITPT: MOV     UNIT,R2        :GET PORT NUMBER
210 003235          060011          XFC     DINIT         :DO THE INIT
211 003236          104207 066540      MOV     #28000.,R0    :1 SECOND TIMER
212 003240          INITP1: CALL    STATVL        :VALIDATE STATUS
213 003242          BNE     INITDD        :DEAD IF NOT VALID
214 003244          117407          DEC     R0           :DECREMETN COUTNER
215 003245          BEQ     INITDD        :DEAD IF COUNT EXPIRED
216 003247          102201 000001      BIT     #RCVRDY,R1    :DONE INIT ?
217 003251          BEQ     INITP1        :NOPE - KEEP TRYING
218 003253          RETURN          :EXIT
219 003255          104201 000024      INITDD: MOV     #20.,R1  :ERROR CODE
220 003257          114002          CLR     R2           :NO SUBCODE
221 003260          CALL    ERRMNT        :ERROR EXIT

```

UDAF52 - UDA-52 FORMATER DMACR X04.01 23-AUG-82 13:14:22 PAGE 14
 OVERLAY PROCESSING ROUTINES

```

1
2
3
4
5
6
7
8 003262 104204 002013      NEXT:  MOV    #OVLTL,R4          :GET POINTER TO OVERLAY TABLE
9 003264 105014              ADD    R1,R4                   :INDEX INTO TABLE
10 003265 104203 004014     MOV    #START,R3              :UDA ADDRESS TO LOAD AT
11 003267              NEXT5: CALL  OVRLAY             :CALL ROUTINE TO DO OVERLAY
12 003271 115007              TST   R0                       :CHECKSUM O.K. ??
13 003272              BNE   OERR                    :YES - RETRY IF POSSIBLE
14 003274 114C00 001367     CLR   UN.ERR                 :CLEAR ERROR COUNT
15 003276              POP   R1                      :POP CURRENT RETURN ADDRESS
16 003277              BR    START                  :GO TO OVERLAY
17 003301 106300 002165 001367 OERR:  CMP    RETRY,UN.ERR          :DONE ALL RETRIES ?
18 003304              BEQ   OERR2                 :YUP
19 003306 115400 001367     INC   UN.ERR                 :INC ERROR AND
20 003310              BR    NEXT5                 :TRY AGAIN
21
22
23
24 003312 104012              OERR2: MOV   R1,R2              :GET ERROR CODE FROM XFC
25 003313 101200 000020 001703 BIS    #DEAD,FLAG1            :INDICATE HOST GONE
26
27
28
29 003316 104201 000005     UERR:  MOV   #5,R1              :SET UNIBUS I/O ERROR
30 003320              CALL  ERRMNT                 :ERROR RETURN

```

.SBTTL OVERLAY PROCESSING ROUTINES

OVERLAY PROCESSING ROUTINES
 R1 = OFFSET INTO TABLE

NEXT CALLS OVERLAY FOR NEXT CODE OVERLAY

DEAD HOST EXIT FOR ALL ROUTINES

UNIBUS ERROR EXIT FOR ALL ROUTINES

UDAF52 - UDA-52 FORMATTER CHACR X04.01 23-AUG-82 13:14:22 PAGE 16
 OVERLAY PROCESSING ROUTINES

UD
DU

1					
2					
3					
4					
5					
6					
7	003411	104647	000001	OVRLAY:	MOV HSTLO(R4),R0 ;LOW ORDER UNIBUS ADDRESS
8	003413	104641	000002		MOV HSTHI(R4),R1 ;HI ORDER UNIBUS ADDRESS
9	003415	104642	000000		MOV LEN(R4),R2 ;WORD COUNT TO OVERLAY
10	003417	060013			XFC UREAD ;ISSUE UNIBUS READ
11	003420	115001			TST R1 ;ANY UNIBUS PROBLEMS ?
12	003421				BNE UERR ;YUP - EXIT WITH UNIBUS ERROR
13	003423				RETURN ;RETURN TO CALLING ROUTINE
14					
15					
16					
17	003425	104207	001750	SNDMNT:	MOV #DMBUF,R0 ;POINT TO BUFFER
18	003427	104201	000016		MOV #DMBUFL,R1 ;LENGTH
19	003431	060016			XFC MAINTR ;ISSUE COMMAND
20	003432	115001			TST R1 ;ANY UNIBUS PROBLEMS ?
21	003433				BNE UERR ;YUP - EXIT WITH UNIBUS ERROR
22	003435				RETURN
23					
24					
25					
26					
27					
28	003437	104207	001750	RCVMNT:	MOV #DMBUF,R0 ;POINT TO BUFFER
29	003441	104201	000016		MOV #DMBUFL,R1 ;SIZE OF BUFFER
30	003443	060017			XFC MAINTW ;RECEIVE MAINT WRITE DATA
31	003444	115001			TST R1 ;ANY UNIBUS PROBLEMS ?
32	003445				BNE UERR ;YUP - EXIT WITH UNIBUS ERROR
33	003447				RETURN

UDAF52 - UDA-52 FORMATTER DM/CR X04.01 23-AUG-82 13:14:22 PAGE 17
 MISCELLANEOUS COMMON ROUTINES

UD
DU

```

1
2
3
4
5
6
7
8 003451 102200 000010 001703 ERRMNT: BIT #ERDN,FLAG1 ;TRIED TO QUIT YET ?
9 003454 ;BNE ALLOVR ;YUP - DISCON FAILED
10 003456 101200 000010 001703 BIS #ERDN,FLAG1 ;SET FLAG
11 003461 104207 001750 MOV #DMBUF,R0 ;POINT TO MAINT BUFFER
12 003463 100171 MOV R1,(R0) ;PUT ERROR NUMBER IN MSG
13 003464 100672 0C0001 MOV R2,1(R0) ;PUT IN ERROR SUBCODE
14 003466 104201 000063 MOV #G6,R1 ;ERROR MESSAGE OVERLAY
15 003470 CALL PAGE ;BRING IT IN
16 003472 CALL DISCON ;DISCONNECT/SPINDOWN DRIVE
17 003474 114007 ALLOVR: CLR R0 ;IN CASE O.K
18 003475 102200 000020 001703 BIT #DEAD,FLAG1 ;DIE OR JUST QUIT ?
19 003500 BEQ ALLOV1 ;JUST QUIT
20 003502 115407 INC R0 ;MAKE NON ZERO
21 003503 060021 ALLOV1: XFC DONE ;EXIT DM MODE
22
23
24
25 003504 104300 002170 001556 ERRHND: MOV RECTMP,ERECOV+1 ;STORE LEVEL IN COMMAND
26 003507 104203 001460 MOV #CR.ERV,R3 ;POINT TO COMMAND
27 003511 CALL TALK ;DO ERROR RECOVERY
28 003513 RETURN ;RETURN

```


UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 21
MISCELLANEOUS COMMON ROUTINES

COMPUTE ECC SYMBOLS IN ERROR

1					
2					
3					
4					
5					
6	003763	104202	001512	ECCCK:	MOV #CR,R2 ;POINT TO CHARACTERISTICS
7	003765	104623	000002		MOV ERRSYM(R2),R3 ;GET THRESHOLD
8	003767	103203	000377		BIC #LOBYTE,R3 ;CLEAR LOW GARBAGE
9	003771	110703			SWAB R3 ;GET IN LOW ORDER
10	003772	115003			TST R3 ;IS THRESHOLD 0 ?
11	003773				BEQ 105\$;YUP - NO POINT IN CORRECTING
12	003775	104207	001373		MOV #RDBLK,R0 ;POINT TO COMMAND BLOCK
13	003777	104302	002171		MOV SECSIZ,R2 ;SECTOR SIZE IN WORDS
14	004001	060015			XFC ECC ;PERFORM ECC CORRECTION
15	004002	115001			TST R1 ;SUCCESSFUL ?
16	004003				BNE 105\$;NOPE
17	004005	106073			CMP R0,R3 ;WITHIN BOUNDS ?
18	004006				BMI GDECC ;YES CONSIDER GOOD
19	004010	104201	177777	105\$:	MOV #-1,R1 ;ELSE SIGNAL BAD
20	004012			GDECC:	RETURN ;RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 22-1
 DUP DM<->HOST STARTUP OVERLAY

```

58
59 004236 030000      :
60 004237      106    157    162  FMTSTA: .WORD 30000      ;DUP WORD
61      000010      FMSTL = .ASCIZ 'Format begun' ;MESSAGE
62      :           =      .-FMTSTA      ;LENGTH
63      :           INPUT ERROR MESSAGE
64      :
65 004246 030100      :
66 004247      111    156    160  INPERR: .WORD 30100      ;DUP WORD
67      000007      INPEL = .ASCIZ 'Input Error' ;MESSAGE
68      :           =      .-INPERR      ;LENGTH
69      :           .DSABL LC
70      :
71      :           MISCELLANEOUS DUP STORAGE
72      :
73 004255 000000      DATBUF: .WORD 0      ;CONVERT BUFFER FOR DATE
74 004256 000000      .WORD 0      ;MAKE SURE ALL 2 WORDS
75 004257 000000      .WORD 0      ;ARE 0
76 004260 000000      MLEN: .WORD 0      ;LENGTH STORAGE
77      000002      MENTLN = 2      ;CONSTANT ENTRY LENGTH
78 004261 000051      MSGOFF = 1      ;OFFSET OF MESSAGE LENGTH
79      000001      TBUFF: .REPT 41.      ;ZERO 41 WORDS
80      :           .WORD 0      ;TERMINAL BUFFER
81      :           .ENDM
82      000051      TBUFFL = .-TBUFF
83      000131      Y = 131      ;ASCII 'Y'
84      000055      DASH = 055      ;ASCII '-'
85      000057      SLAS = 057      ;ASCII '/'
86      036031      DAYS = 14.+31.+365.+<365.*4+1*10.>+365.
87      :           ;DAYS FROM NOV. 17,1858 TO
88      :           ;JAN. 1,1901
89      :
90      :           DATE CONVERSION TABLES
91      :
92 004332 000000      .WORD 0      ;TERMINATOR
93 004333 000012      .WORD 10.      ;100'S NANoseconds PER MICROSECOND
94 004334 023420      .WORD 10000.      ;MICROSECONDS PER HUNDREDTH SECOND
95 004335 000144      .WORD 100.      ;HUNDREDTHS OF SECOND PER SECOND
96 004336 000074      .WORD 60.      ;SECONDS PER MINUTE
97 004337 000074      .WORD 60.      ;MINUTES PER HOUR
98 004340 000030      .WORD 24.      ;HOURS PER DAY
99      :
100      :           QUARTER DAYS PER MONTH TABLE
101      :
102      :
103 004341 000000      TIMTBL: .WORD 0      ;TERMINATOR FOR MONTH TABLE
104 004342 000174      .WORD 31.*4      ;QUARTER DAYS IN JANUARY
105 004343 000161      .WORD 28.*4+1      ;QUARTER DAYS IN FEBRUARY
106 004344 000174      .WORD 31.*4      ;QUARTER DAYS IN MARCH
107 004345 000170      .WORD 30.*4      ;QUARTER DAYS IN APRIL
108 004346 000174      .WORD 31.*4      ;QUARTER DAYS IN MAY
109 004347 000170      .WORD 30.*4      ;QUARTER DAYS IN JUNE
110 004350 000174      .WORD 31.*4      ;QUARTER DAYS IN JULY
111 004351 000174      .WORD 31.*4      ;QUARTER DAYS IN AUGUST
112 004352 000170      .WORD 30.*4      ;QUARTER DAYS IN SEPTEMBER
113 004353 000174      .WORD 31.*4      ;QUARTER DAYS IN OCTOBER
114 004354 000170      .WORD 30.*4      ;QUARTER DAYS IN NOVEMBER

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 22-2
DUP DM<->HOST STARTUP OVERLAY

115 004355 177777
116
117

.WORD -1

;PRETEND INFINITE DAYS IN DECEMBER,
;SINCE NOTHING COMES AFTER IT

UDAF52 - UDA-52 FORMATER DMACR X04.01 23-AUG-82 13:14:22 PAGE 25
 DUP DM<->HOST STARTUP OVERLAY

```

1
2
3
4 004516
5 004517 104060 001701
6 004521 104202 004255
7 004523 103200 006000 001703
8 004526 104204 000003
9 004530
10 004552 115001
11 004553
12 004555 106201 000055
13 004557
14 004561 106201 000057
15 004563
16 004565
17 004607
18 004611
19 004612 104207 004255
20 004614
21 004616
22 004620 104073
23 004621
24 004622
25 004623 103200 004000 001703
26 004626 104202 004255
27 004630 104201 000003
28 004632 114005
29 004633 100225
30 004634 117401
31 004635
32 004637 104202 004255
33 004641 117404
34 004642
35 004644
36 004645
37 004646
38 004647
39 004651 115005
40 004652
41 004654 104203 004675
42 004656
43 004657 104204 000006
44 004661 114005
45 004662
46 004663 117404
47 004664
48 004666
49 004667
50 004670
51 004671 104207 001766
52 004673
53 004675
54 004676
55 004700 104306 001701
56
57 004702

```

DO DATE CONVERSION

```

DATCON: PUSH R2 ;SAVE STRING COUNTER
        MOV SP,STCKSV ;SAVE STACK PTR IN CASE OF ERROR
        MOV #DATBUF,R2 ;POINT TO DATA BUFFER
        BIC #GTFLAG+STFLAG,FLAG1 ;CLEAR FLAGS
        MOV #3,R4 ;COUNT TO CONVERT(MONTH,DAY,YEAR)
DATLP: GETB R0,R1 ;GET A BYTE FROM R0 INTO R1
        TST R1 ;IS IT A ZERO BYTE ?
        BEQ DONONE ;YES - DONE THIS STRING
        CMP #DASH,R1 ;IS IT A '-'
        BEQ DONONE ;YES - DONE THIS STRING
        CMP #SLAS,R1 ;IS IT A '/'
        BEQ DONONE ;YES - DONE THIS STRING
        STOB R1,R3,R2 ;STORE BYTE IN R1 AT LOCATION IN R2
        BR DATLP ;LOOP BACK
DONONE: PUSH R0 ;SAVE POINTER TO STRING
        MOV #DATBUF,R0 ;POINT TO BUFFER TO CONVERT
        CALL GENCON ;CONVERT THE STRING
        BNE DATAGN ;ILLEGAL CHARS - ASK AGAIN
        MOV R0,R3 ;GET RESULT
        POP R0 ;RESTORE POINTER TO STRING
        PUSH R3 ;STORE RESULT ON STACK
        BIC #STFLAG,FLAG1 ;SET STORE FLAG TO LOW BYTE
        MOV #DATBUF,R2 ;POINT TO CONVERT BUFFER
        MOV #3,R1 ;FOR BUFFER CLEAR
        CLR R5 ;CLEAR FOR STORE
DATLP1: MOV R5,(R2)+ ;CLEAR WORD
        DEC R1 ;DEC COUNTER
        BNE DATLP1 ;CONTINUE TILL DONE
        MOV #DATBUF,R2 ;RESTORE POINTER
        DEC R4 ;DECREMENT PARAMETER COUNTER
        BNE DATLP ;CONTINUE TILL DONE
        POP R0 ;GET YEAR
        POP R1 ;GET DAY
        POP R2 ;GET MONTH
        CALL DATVER ;VERIFY DATE
        TST R5 ;ANY ERROR ?
        BMI DATAGN ;YUP - PROMPT AGAIN
        MOV #DATRET,R3 ;RETURN ADDRESS
        PUSH R3 ;PUSH ON STACK
        MOV #6,R4 ;COUNT OF ZERO STACK ENTRIES
        CLR R5 ;CLEAR FOR STORE
DATLP2: PUSH R5 ;ZERO ENTRY FOR CONVERSION
        DEC R4 ;DEC COUNTER
        BNE DATLP2 ;LOOP TILL DONE
        PUSH R1 ;STORE DAY
        PUSH R2 ;STORE MONTH
        PUSH R0 ;STORE YEAR
        MOV #DATE,R0 ;POINT TO RESULT BUFFER
        JMP VAXTME ;CALL CONVERT TO VAX TIME ROUTINE
DATRET: POP R2 ;RESTORE MESSAGE POINTER
DATRT1: RETURN
DATAGN: MOV STCKSV,SP ;RESTORE STACK PTR - MAY HAVE PUSHED
        POP R2 ;SOME RESULTS ONTO STACK
        ;GET MESSAGE POINTER

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 25-1
DUP DM<->HOST STARTUP OVERLAY

58 004703 107202 000002
59 004705 101200 001000 001703
60 004710

SUB #MENTLN,R2
BIS #REPEAT,FLAG1
BR DATRT1

:POINT BACK AT BEGINNING OF QUESTION
:SET TO REPEAT - ANSWER MUST BE VALID
:RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 28
DUP DM<->HOST STARTUP OVERLAY

1				:	FORMAT IN 576 ?	
2				:		
3				:		
4				:	RO -> STRING	
5				:		
6	005016			SECCN:	PUSH R2	:SAVE MESSAGE POINTER
7	005017				CALL FIDANS	:GET RESPONSE
8	005021	115001			TST R1	:TEST RESPONSE
9	005022				BEQ SECNO	:NULL - DEFAULT TO 512
10	005024				BMI SECNO	:NO - DEFAULT TO 512
11	005026	101200	020000 001703		BIS #MODE,FLAG1	:YES - DO 576
12	005031			SECNO:	POP R2	:RESTORE MESSAGE POINTER
13	005032				RETURN	

U
11

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 29
 DUP DM<->HOST STARTUP OVERLAY

1									
2									
3									
4	005034								
5	005035								
6	005037								
7	005041	115007							
8	005042								
9	005044	115001							
10	005045								
11	005047	115002							
12	005050								
13	005052	115003							
14	005053								
15	005055								
16	005056	107202	000002						
17	005060	101200	001000	001703					
18	005063								
19	005065	104070	001772						
20	005067	104010	001773						
21	005071	104020	001774						
22	005073	104030	001775						
23	005075								
24	005076	101200	000200	001703					
25	005101								

			SERIAL NUMBER HANDLER	
	SERCON:	PUSH	R2	:SAVE MESSAGE POINTER
		CALL	GENCON	:CONVERT HIGH ORDER
		BNE	SERBD	:IF NE THEN ILLEGAL CHARS - PROMPT AGN
		TST	R0	:IS IT ZERO ?
		BNE	SEROK	:NO - ALL BITS CAN'T BE ZERO
		TST	R1	:IS IT ZERO ?
		BNE	SEROK	:NO - ALL BITS CAN'T BE ZERO
		TST	R2	:IS IT ZERO ?
		BNE	SEROK	:NO - ALL BITS CAN'T BE ZERO
		TST	R3	:IS IT ZERO ?
		BNE	SEROK	:NO - ALL BITS CAN'T BE ZERO
	SERBD:	POP	R2	:GET MESSAGE POINTER
		SUB	#MENTLN,R2	:POINT BACK AT BEGINNING OF QUESTION
		BIS	#REPEAT,FLAG1	:SET TO REPEAT - ANSWER MUST BE NON-ZERO
		BR	SERRT	:RETURN
	SEROK:	MOV	R0,SERNUM	:LOW ORDER WORD
		MOV	R1,SERNUM+1	:LOW MIDDLE
		MOV	R2,SERNUM+2	:HIGH MIDDLE
		MOV	R3,SERNUM+3	:HIGH ORDER
		POP	R2	:RESTORE MESSAGE POINTER
		BIS	#QUESDN,FLAG1	:SET QUESTIONS DONE
	SERRT:	RETURN		:RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14.22 PAGE 30
 DUP DM<->HOST STARTUP OVERLAY

```

1
2
3
4 005103          :
5 005104          : DOWN LINE LOAD FILE NAME HANDLER
6 005106 115001   : DLLFLE: PUSH      R2          :SAVE MESSAGE POINTER
7 005107          : CALL      FIDANS         :GET RESPONSE
8 005111          : TST      R1             :TEST RESPONSE
9 005113 101200 000400 001702 : BEQ      DLLNO          :NULL - DEFAULT TO BEST GUESS
10 005116         : BMI      DLLNO          :NO - DO BEST GUESS
11 005120 101200 000200 001703 : BIS      #DLL,FLAG      :YES - DO DOWN-LINE LOAD
12 005123         : BR       DLLDN          :EXIT
13 005124         : BIS      #QUESDN,FLAG1  :SET END OF QUESTIONS FLAG
14          : DLLDN: POP      R2      :RESTORE MESSAGE POINTER
15          : DLLDN1: RETURN
16          :
17          : THIS SECTION SKIPS THE NEXT QUESTION
18          : WHICH PERTAINS TO CONTINUING IF FCT IS BAD
19 005126 101200 002000 001702 : DLLNO: BIS      #BSTGS,FLAG :DO BEST GUESS
20 005131         : POP      R2             :GET MESSAGE POINTER
21 005132 105302 004260         : ADD     MLEN,R2         :ADD CURRENT MESSAGE LENGTH
22 005134 104621 000001         : MOV     MSGOFF(R2),R1   :GET NEXT MESSAGE LENGTH
23 005136 105202 000002         : ADD     #MENTLN,R2     :PAST THE FRONT ENTRIES
24 005140 104010 004260         : MOV     R1,MLEN        :MAKE THIS MESSAGE THE NEW LENGTH
25 005142         : BR      DLLDN1        :RETURN

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 31
 DUP DM<->HOST STARTUP OVERLAY

UD
IN

```

1
2
3
4
5 005144          GENCON: PUSH    R4          :SAVE R4
6 005145          CALL    FINLEN        :FIND LENGTH OF STRING
7 005147 101200 000400 001703  BIS    #FLIPON,FLAG1  :FLIP INDICATOR
8 005152 104074          MOV    RO,R4          :POINT AT STRING
9 005153 114007          CLR    R0             :RESULT REGISTER
10 005154 114001         CLR    R1             :RESULT REGISTER
11 005155 114002         CLR    R2             :RESULT REGISTER
12 005156 114003         CLR    R3             :RESULT REGISTER
13 005157 115005         TST    R5             :IS STRING NULL ?
14 005160          BEQ    CONDON          :YJP - EXIT
15 005162 104140 001410  CNLP1: MOV    (R4),TEMP      :GET WORD OF STRING
16 005164 103200 177400 001410  CNLP: BIC    #HIBYTE,TEMP  :CLEAR HIGH BYTE
17 005167 106200 000057 001410  CMP    #57,TEMP        :LESS THAN 0 (60 OCTAL) ?
18 005172          BPL    CONER1          :YES - ERROR
19 005174 106200 000071 001410  CMP    #'9,TEMP        :GREATER THAN 9 ?
20 005177          BMI    CONER1          :YES - ERROR
21 005201 103200 177760 001410  BIC    #'^C17,TEMP     :SUB ASCII 60 FROM CHARACTER
22 005204 105307 001410          ADD    TEMP,RO        :ADD TO RESULT REGISTER
23 005206          BCC    NOCERR          :IF NO CARRY THEN CONTINUE
24 005210 115401          INC    R1             :ELSE INC NEXT REGISTER
25 005211          BCC    NOCERR          :IF NO CARRY THEN CONTINUE
26 005213 115402          INC    R2             :ELSE INC NEXT REGISTER
27 005214          BCC    NOCERR          :NO CARRY - CONT
28 005216 115403          INC    R3             :INC NEXT REGISTER
29 005217          BCC    NOCERR          :NO CARRY - CONT
30 005221          BR     CONERR          :ERROR IF CARRY WAS SET
31 005223 117405          NOCERR: DEC    R5             :DECREMENT COUNTER
32 005224          BEQ    CONDON          :ALL DONE
33 005226          CALL    MULT10        :MULTIPLY BY TEN
34 005230 102200 000400 001703  BIT    #FLIPON,FLAG1  :WHICH BYTE ARE WE ON ?
35 005233          BEQ    CONLOW          :DONE ALL OF THIS WORD
36 005235 104240 001410          MOV    (R4)+,TEMP     :GET WORD AGAIN
37 005237          PUSH   RO             :SAVE RO
38 005240 104307 001410          MOV    TEMP,RO        :GET FOR SWAB
39 005242 110707          SWAB   RO             :MAKE HIGH BYTE LOW
40 005243 104070 001410          MOV    RO,TEMP        :STORE BACK
41 005245          POP     RO             :RESTORE RO
42 005246 103200 000400 001703  BIC    #FLIPON,FLAG1  :CLEAR FLAG
43 005251          BR     CNLP             :PROCESS HIGH BYTE
44 005253 101200 000400 001703  CONLOW: BIS    #FLIPON,FLAG1  :SET FLIP FLAG
45 005256          BR     CNLP1          :DO LOW BYTE
46 005260          CONDON: POP    R4             :RESTORE R4
47 005261 106011          CMP    R1,R1          :SET Z FLAG
48 005262          CONDO1: RETURN
49 005264 104201 000025          CONERR: MOV    #21,,R1      :ERROR 21 - RESPONSE ERROR
50 005266          CALL   ERRMNT        :ERROR RETURN
51 005270          CONER1: POP    R4             :RESTORE R4
52 005271 106200 177777 004260  CMP    #177777,MLEN    :SET NOT EQUAL-(MLEN - MESSAGE LENGTH)
53 005274          BR     CONDO1        :AND RETURN

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 32
 DUP DM<->HOST STARTUP OVERLAY

1			:			
2			:	MULTIPLY BY 2		
3			:			
4			:	RO,R1,R2,R3 = 64-BIT VALUE TO BE MULTIPLIED		
5			:			
6	005276	105207	MULT2:	ADD #0,R0		:CLEAR CARRY
7	005300	110207		ROL R0		:SHIFT FIRST WORD
8	005301	110201		ROL R1		:SHIFT CARRY AND SECOND WORD
9	005302	110202		ROL R2		:SHIFT CARRY AND THIRD WORD
10	005303	110203		ROL R3		:SHIFT CARRY AND FOURTH WORD
11	005304			BCC MULDN		:IF NO CARRY THEN DONE
12	005306			BR MULERR		:ELSE ERROR
13	005310		MULDN:	RETURN		:RETURN
14	005312	104201	MULERR:	MOV #21.,R1		:RESPONSE ERROR
15	005314	000025		CALL ERRMNT		:ERROR EXIT

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 34
 DUP DM<->HOST STARTUP OVERLAY

1				:		
2				:	FIND THE LENGTH OF A STRING	
3				:		
4				:	RO -> STRING	
5				:		
6				:	OUTPUT:	
7				:		
8				:	R5 = COUNT	
9				:		
10	005367			:	FINLEN: PUSH R0	:SAVE R0
11	005370	114005		:	CLR R5	:CLEAR COUNTER
12	005371	104171		:	FINLN1: MOV (R0),R1	:GET WORD
13	005372	103201	177400	:	BIC #HIBYTE,R1	:CLEAR HIGH BYTE
14	005374			:	BEQ FINDON	:IF ZERO THEN DONE
15	005376	115405		:	INC R5	:INCREMENT COUNT
16	005377	104271		:	MOV (R0)+,R1	:GET WORD FOR HIGH BYTE
17	005400	103201	000377	:	BIC #LOBYTE,R1	:CLEAR LOW BYTE
18	005402			:	BEQ FINDON	:IF ZERO THE DONE
19	005404	115405		:	INC R5	:INCREMENT COUNT
20	005405			:	BR FINLN1	:REPEAT WITH NEXT WORD
21	005407			:	FINDON: POP R0	:RESTORE R0
22	005410			:	RETURN	

UDAF52 - UDA-52 FORMATTER DMACR XG4.01 23-AUG-82 13:14:22 PAGE 35
 DUP DM<->HOST STARTUP OVERLAY

1
2
3
4
5
6
7
8
9
10
11
12 005412 114001
13 005413 104172
14 005414 103202 177400
15 005416
16 005420 103202 000040
17 005422 106202 000131
18 005424
19 005426 115401
20 005427
21 005431 117401
22 005432

```

:
: DETERMINE IF VALUF IS 'Y',NULL, OR NOT Y
: INPUT:
: RO -> STRING
: OUTPUT:
: R1 = 1 IF 'Y'
:       0 IF NULL
:       -1 IF NOT Y
:
: FIDANS: CLR R1 ;CLEAR OUTPUT
: MOV (R0),R2 ;GET WORD
: BIC #HIBYTE,R2 ;CLEAR HIGH STUFF
: BEQ FIDNUL ;IF ZERO THEN IT'S NULL
: BIC #BIT5,R2 ;MAKE IT UPPER CASE
: CMP #Y,R2 ;IS IT 'Y' ?
: BNE NOTY ;NOPE
: INC R1 ;MAKE IT 1
: FIDNUL: RETURN ;RETURN
: NOTY: DEC R1 ;MAKE IT NEGATIVE
: BR FIDNUL ;AND EXIT

```


UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 36
 DUP DM<->HOST STARTUP OVERLAY

```

1
2
3
4
5
6 005434          :
7 005435          :
8 005437 115001   :
9 005440          :
10 005442 101200 000001 001702 :
11
12
13
14
15 005445          :
16 005446 105302 004260 :
17 005450 104621 000001 :
18 005452 105202 000002 :
19 005454 104010 004260 :
20 005456          :
21
22
23 005460          :
24 005461          :

```

USE EXISTING FCT ?

RO -> STRING

EXTFCT: PUSH R2 ;SAVE MESSAGE POINTER
 CALL FIDANS ;WHAT'S THE ANSWER
 TST R1 ;TEST THE RESPONSE
 BMI EXTRET ;NO - EXIT
 BIS #FCTAVL,FLAG ;YES - SET THE FLAG

THIS SECTION SKIPS THE NEXT QUESTION
 WHICH PERTAINS TO DOWN-LINE LOADING

POP R2 ;GET MESSAGE POINTER
 ADD MLEN,R2 ;ADD CURRENT MESSAGE LENGTH
 MOV MSGOFF(R2),R1 ;GET NEXT MESSAGE LENGTH
 ADD #MENTLN,R2 ;PAST THE FRONT ENTRIES
 MOV R1,MLEN ;MAKE THIS MESSAGE THE NEW LENGTH
 BR EXTRT1 ;RETURN

EXTRET: POP R2 ;RESTORE MESSAGE POINTER
 EXTRT1: RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 38
 DUP DM<->HOST STARTUP OVERLAY

```

1
2
3
4
5
6 005505 104204 170140 VAXTME: MOV #<DAYS-1>*4,R4 ;QUARTER DAYS FROM BEGIN TO 1901
7 005507 104262 MOV (SP)+,R2 ;GET YEARS
8 005510 BNE 30$ ;BRANCH IF NOT 1900
9 005512 117404 DEC R4 ;CAUSE 1900 TO NOT BE A LOOP YEAR
10 005513 104261 30$: MOV (SP)+,R1 ;GET MONTH
11 005514 117402 DEC R2 ;YEAR AFTER LEAP YEAR DIVISABLE BY 4
12 005515 104020 001403 MOV R2,DDUMMY ;FOR MULTIPLY
13 005517 114000 001404 CLR DDUMMY+1 ;CLEAR HIGH ORDER
14 005521 104200 002665 001410 MOV #<365.*4+1>,TEMP ;QUARTER DAYS IN A YEAR
15 005524 114000 001411 CLR TEMP+1 ;CLEAR HIGH ORDER
16 005526 PUSH R4 ;SAVE DAYS
17 005527 104203 001403 MOV #DDUMMY,R3 ;FOR MULT
18 005531 104204 001410 MOV #TEMP,R4 ;DITTO
19 005533 CALL DMUL ;GET YEAR TIMES QUARTER DAYS IN A YEAR
20 005535 POP R4 ;RESTORE DAYS
21 005536 104303 001410 MOV TEMP,R3 ;LOW ORDER
22 005540 104302 001411 MOV TEMP+1,R2 ;HIGH ORDER
23 : NOTE: LOW TWO BITS OF R3+R4 ARE ONES IFF. LEAP YEAR
24 005542 115003 TST R3 ;AFTER 1900 ?
25 005543 BPL 40$ ;YES
26 005545 115002 TST R2 ;AFTER 1900 ?
27 005546 BPL 40$ ;YES
28 005550 104165 MCV (SP),R5 ;GET DAYS
29 005551 115405 INC R5 ;ADJUST FOR 1 DAY LESS DAY IN 1900
30 005552 100165 MOV R5,(SP) ;PUT BACK
31 005553 117401 40$: DEC R1 ;BIAS MONTH TO ZERO
32 005554 105201 004342 ADD #TIMTBL,R1 ;INDEX INTO MONTH TABLE
33 005556 105043 50$: ADD R4,R3 ;ADD TO DAYS
34 005557 BCC 51$ ;IF NO CARRY SKIP
35 005561 115402 INC R2 ;PROP CARRY
36 005562 104414 51$: MOV -(R1),R4 ;GET LENGTH OF NEXT PREVIOUS MONTH
37 005563 BNE 50$ ;BRANCH UNTIL END OF TABLE
38 005565 117401 DEC R1 ;POINT TO FIRST TIME MULTILPLIER
39
40 :
41 :
42 005566 105203 000000 ADD #0,R3 ;MAKE SURE CARRY IS 0
43 005570 110603 ROR R3 ;SHIFT RIGHT 1
44 005571 105203 000000 ADD #0,R3 ;CLEAR CARRY
45 005573 110603 ROR R3 ;ROTATE RIGHT 1
46 005574 105203 000000 ADD #0,R3 ;CLEAR CARRY
47 005576 110602 ROR R2 ;ROTATE HIGH ORDER RIGHT 1
48 005577 BCC 52$ ;IF NO CARRY THEN SKIP
49 005601 101203 040000 BIS #40000,R3 ;DO "SHIFT" INTO LOW ORDER
50 005603 105203 000000 ADD #0,R3 ;CLEAR CARRY
51 005605 110602 52$: ROR R2 ;SHIFT HIGH ORDER RIGHT 1
52 005606 BCC 53$ ;IF NO CARRY THEN SKIP
53 005610 101203 100000 BIS #100000,R3 ;DO "SHIFT" INTO LOW ORDER
54
55 :
56 005612 105263 53$: ADD (SP)+,R3 ;ADD IN DAY OF MONTH
57 005613 BCC 54$ ;SKIP INC IF NO CARRY

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 38-1
 DUP DM<->MOST STARTUP OVERLAY

```

58 005615 115402          INC      R2                ;PROP CARRY
59                      :
60                      :      INIT QUAD-WORD TO DAYS SINCE BASE DATE
61                      :
62 005616 104074          54$:  MOV      R0,R4                ;QUAD WORD POINTER
63 005617 100243          MOV      R3,(R4)+          ;COPY LOW ORDER DAYS
64 005620 100242          MOV      R2,(R4)+          ;COPY HIGH ORDER DAYS
65 005621 114005          CLR      R5                ;FOR REST OF QUAD-WORD CLEAR
66 005622 100245          MOV      R5,(R4)+          ;CLEAR
67 005623 100145          MOV      R5,(R4)          ;CLEAR
68                      :
69                      :      LOOP TO MERGE TIME OF DAY INTO DAYS SINCE BASE DATE
70                      :
71 005624 104074          60$:  MOV      R0,R4                ;COPY QUAD WORD POINTER
72 005625 104205 000004  MOV      #4,R5            ;INNER LOOP COUNT
73                      :
74                      :      CALCULATE QUAD-WORD <- QUAD-WORD * (R1) + (SP)+
75                      :
76 005627 104142          70$:  MOV      (R4),R2          ;FETCH NEXT WORD OF QUAD-WORD
77 005630 104113          MOV      (R1),R3          ;GET MULTIPLIER
78 005631 104030 001403  MOV      R3,DDUMMY        ;STORE FOR MULTIPLY
79 005633 104020 001410  MOV      R2,TEMP          ;FOR MULTIPLY
80 005635 114000 001404  CLR      DDUMMY+1        ;CLEAR HIGH ORDER
81 005637 114000 001411  CLR      TEMP+1          ;DITTO
82 005641          PUSH     R4                ;SAVE QUAD-WORD POINTER
83 005642 104203 001403  MOV      #DDUMMY,R3      ;FOR MULTIPLY
84 005644 104204 001410  MOV      #TEMP,R4        ;DITTO
85 005646          CALL     DMUL             ;DO MULTIPLY
86 005650 104243          MOV      (R4)+,R3        ;GET LOW ORDER RESULT
87 005651 104142          MOV      (R4),R2        ;GET HIGH ORDER RESULT
88 005652          POP      R4                ;RESTORE QUAD-WORD POINTER
89 005653 115002          TST     R2                ;IS IT POSITIVE
90 005654          BPL     80$             ;O.K.
91 005656 105112          ADD     (R1),R2          ;MAKE IT AN UNSIGNED MULTIPLY
92 005657 105163          80$:  ADD     (SP),R3          ;ADD HIGH ORDER OF PREVIOUS MUL.
93 005660          BCC     81$             ;SKIP INC IF NO CARRY
94 005662 115402          INC     R2                ;ADD CARRY
95 005663 100243          81$:  MOV      R3,(R4)+          ;STORE WORD INTO QUAD-WORD
96 005664 100162          MOV      R2,(SP)         ;SAVE HIGH ORDER WORD
97 005665 117405          DEC     R5                ;DEC COUNT
98 005666          BNE     70$             ;CONTINUE TILL DONE
99 005670 104415          MOV     -(R1),R5         ;"POP" R1
100 005671          BEQ     90$             ;IF ZERO THEN DONE
101 005673 104265          MOV     (SP)+,R5         ;POP STACK
102 005674          BR     60$             ;DO NEXT MULTIPLIER
103 005676 104265          90$:  MOV     (SP)+,R5         ;DO FINAL POP
104 005677          RETURN                ;AND RETURN

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 39
DUP DM<->MOST STARTUP OVERLAY

1
2
3
4
5
6
7
8
9

```
...      BRING IN ODT IN HIGH MEMORY  
...  
.IF DF DEBUG  
  .NLIST  
  . = SODT  
  DMODT  
  .LIST  
.ENDC
```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 40
INITIALIZATION OVERLAY (G1)

```

1          .SBTTL  INITIALIZATION OVERLAY (G1)
2          :
3          :
4          :
5          :
6          :
7          :
8          :
9          :
10         :
11         :
12         :
13         :
14         :
15         :
16         :
17         :
18         :
19         :
20         :
21         :
22         :
23         :
24         :
25         :
26         :
27         :
28         :
29         :
30         :
31         :
32         :
33         :
34         :
35         :
36         :
37         :
38         :
39         :
40         :
41         :
42         :
43         :
44         :
45         :
46         :
47         :
48         :
49         :
50         :
51         :
52         :
53         :
54         :
55         :
56         :
57         :

```

005701				DMOVLY	G1,START	
004014				CALL	INITL	: INITIALIZE DISK
004016	102200	000001	001702	BIT	#FCTAVL,FLAG	: USE RESIDENT FCT ?
004021				BNE	DOLBN	: YES - ONLY DO LBN
004023	104201	000000		MOV	#F1,R1	: ELSE DO D/XBN FIRST
004025				BR	DXBN	: SKIP LBN FLAGGING
004027	104201	000003		DOLBN: MOV	#F2,R1	: SIGNAL LBN FORMAT
004031				DXBN: CALL	NEXT	: BRING IN NEXT OVERLAY

```

13         :
14         :
15         :
16         :
17         :
18         :
19         :
20         :
21         :
22         :
23         :
24         :
25         :
26         :
27         :
28         :
29         :
30         :
31         :
32         :
33         :
34         :
35         :
36         :
37         :
38         :
39         :
40         :
41         :
42         :
43         :
44         :
45         :
46         :
47         :
48         :
49         :
50         :
51         :
52         :
53         :
54         :
55         :
56         :
57         :

```

004033	104207	001525		CONINT: MOV	#SCR,R0	: POINT TO SUB CHARACTERISTICS
004035	104673	000003		MOV	TRKGRP(R0),R3	: LOAD TRACKS/GROUP
004037	103203	177400		BIC	#HI BYTE,R3	: CLEAR HIGH BYT
004041	104030	001403		MOV	R3,DDUMMY	: STORE IN DUMMY AREA
004043	114000	001404		CLR	DDUMMY+1	: CLEAR FOR STORE
004045	104673	000002		MOV	GRPCYL(R0),R3	: GET GROUPS/CYLINDER
004047	103203	177400		BIC	#HI BYTE,R3	: CLEAR HIGH BYTE
004051	104030	001410		MOV	R3,TEMP	: STOPE IN TEMP AREA
004053	114000	001411		CLR	TEMP+1	: CLEAR HIGH ORDER
004055	104203	001403		MOV	#DDUMMY,R3	: SETR UP FOR MULT
004057	104204	001410		MOV	#TEMP,R4	: DITTO
004061				CALL	DMUL	: COMPUTE IT
004063	104240	001624		MOV	(R4)+,TRKCYL	: LOAD FOR STORE
004065	104140	001625		MOV	(R4),TRKCYL+1	: LOAD FOR STORE

```

35         :
36         :
37         :
38         :
39         :
40         :
41         :
42         :
43         :
44         :
45         :
46         :
47         :
48         :
49         :
50         :
51         :
52         :
53         :
54         :
55         :
56         :
57         :

```

004067	102200	020000	001703	BIT	#MODE,FLAG1	: WHAT MODE
004072				BNE	1\$: IF SET THEN 576
004074	104673	000011		MOV	LBNT12(R0),R3	: GET LBN/TRACK FOR 512
004076				BR	2\$: SKIP 576 SETUP
004100	104673	000015		1\$: MOV	LBNT76(R0),R3	: GET LBN/TRACK FOR 576
004102	103203	177400		2\$: BIC	#HI BYTE,R3	: CLEAR HIGH BYTE
004104	104030	001410		MOV	R3,TEMP	: FOR MULT
004106	114000	001411		CLR	TEMP+1	: FOR STORE
004110	104204	001410		MOV	#TEMP,R4	: FOR MULTIPLY
004112	104203	001624		MOV	#TRKCYL,R3	: DITTO
004114				CALL	DMUL	: GET LBN'S/CYL
004116	104240	001630		MOV	(R4)+,LBNPCY	: GET LOW ORDER
004120	104140	001631		MOV	(R4),LBNPCY+1	: GET HIGH ORDER

```

51         :
52         :
53         :
54         :
55         :
56         :
57         :

```

004122	104673	000004		MOV	RBNTRK(R0),R3	: GET RBN'S/TRACK
004124	103203	177600		BIC	#HI1BYTE,R3	: CLEAR OUT GARBAGE
004126	104030	001410		MOV	R3,TEMP	: STORE FOR MULT
004130	114000	001411		CLR	TEMP+1	: FOR STORE

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 40-1
INITIALIZATION OVERLAY (G1)

```

58 004132 104204 001410      MOV      #TEMP,R4          ;FOR MULTIPLY
59 004134 104203 001624      MOV      #TRKCYL,R3       ;DITTO
60 004136                      CALL     DMUL              ;GET RBN'S/CYL
61 004140 104240 001632      MOV      (R4)+,RBNPCY     ;GET LOW ORDER
62 004142 104140 001633      MOV      (R4),RBNPCY+1   ;GET HIGH ORDER
63
64      :
65      :
66      :
67      :
68 004144 104207 001525      MOV      #SCR,R0          ;POINT TO CHARACTERISTICS
69 004146 104670 000000 001410  MOV      CYLBN(R0),TEMP    ;GET LBN CYLINDERS
70 004151 104673 000001      MOV      CYLBN+1(R0),R3   ;GET HIGH ORDER
71 004153 103203 170000      BIC      #HD.CLR,R3       ;CLEAR STARTING CYLINDER BITS
72 004155 104030 001411      MOV      R3,TEMP+1        ;STORE IT
73 004157 104204 001410      MOV      #TEMP,R4          ;FOR MULT
74 004161 104203 001630      MOV      #LBNPCY,R3       ;POINT TO LBN'S/CYLINDER
75 004163                      CALL     DMUL              ;GET LBN'S IN LBN AREA
76 004165 104140 001616      MOV      (R4),LBNLBN      ;GET LOW ORDER
77 004167 104640 000001 001617  MOV      1(R4),LBNLBN+1   ;GET HIGH ORDER
78 004172 102200 020000 001703  BIT      #MODE,FLAG1      ;WHAT MODE
79 004175                      BNE      3$                ;IF SET THEN 576
80 004177 104670 000012 001403  MOV      LBNH12(R0),DDUMMY ;GET LBN'S IN HOST AREA (512)
81 004202 104670 000013 001404  MOV      LBNH12+1(R0),DDUMMY+1 ;GET HIGH ORDER
82 004205                      BR       4$                ;SKIP 576 SETUP
83 004207 104670 000016 001403 3$:  MOV      LBNH76(R0),DDUMMY ;GET LBN'S IN HOST AREA (576)
84 004212 104670 000017 001404  MOV      LBNH76+1(R0),DDUMMY+1 ;GET HIGH ORDER
85 004215 104203 001403 4$:  MOV      #DDUMMY,R3       ;FOR SUB
86 004217                      CALL     DSUB              ;SUBTRACT TO GET LBN'S IN RCT
87 004221 104240 002160      MOV      (R4)+,TOTRCT     ;GET LOW ORDER
88 004223 104140 002161      MOV      (R4),TOTRCT+1   ;GET HIGH ORDER
89
90      :
91      :
92      :
93 004225 104207 001525      MOV      #SCR,R0          ;POINT TO CHARACTERISTICS
94 004227 104670 000000 001410  MOV      CYLBN(R0),TEMP    ;GET LBN CYLINDERS
95 004232 104673 000001      MOV      CYLBN+1(R0),R3   ;GET HIGH ORDER
96 004234 103203 170000      BIC      #HD.CLR,R3       ;CLEAR STARTING CYLINDER BITS
97 004236 104030 001411      MOV      R3,TEMP+1        ;STORE IT
98 004240 104204 001410      MOV      #TEMP,R4          ;FOR MULT
99 004242 104203 001632      MOV      #RBNPCY,R3       ;POINT TO RBN'S/CYLINDER
100 004244                      CALL     DMUL              ;GET LBN'S IN LBN AREA
101 004246 104240 001620      MOV      (R4)+,RBNLBN     ;GET LOW ORDER
102 004250 104140 001621      MOV      (R4),RBNLBN+1   ;GET HIGH ORDER
103
104      :
105      :
106 004252 104673 000004      MOV      RBNTRK(R0),R3    ;LOAD RBN'S/TRACK
107 004254 103203 177600      BIC      #H1BYTE,R3       ;CLEAR OUT GARBAGE
108 004256 104674 000011      MOV      LBNT12(R0),R4    ;LOAD LBN'S/TRACK(512)
109 004260 103204 177400      BIC      #H1BYTE,R4       ;CLEAR OUT HIGH BYTE
110 004262 105043                      ADD      R4,R3            ;ADD FOR SECT/TRACK
111 004263 104030 001606      MOV      R3,SECTRK       ;STORE IT
112 004265 104030 001612      MOV      R3,SECT12       ;STORE IT
113 004267 114000 001607      CLR      SECTRK+1        ;CLEAR FOR STORE
114 004271 114000 001613      CLR      SECT12+1        ;CLEAR FOR STORE

```


UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 40-3
INITIALIZATION OVERLAY (G1)

UD
DB

```

172 004431 104240 001604      MOV      (R4)+,CYLNUM      ;MAKE IT CURRENT CYLINDER
173 004433 104140 001605      MOV      (R4),CYLNUM+1    ;LOAD HI ORDER
174                                     :
175                                     :
176                                     :
177 004435 104300 001620 001410      MOV      RBNLBN,TEMP      ;GET LOW ORDER LBN'S IN HOST AREA
178 004440 104300 001621 001411      MOV      RBNLBN+1,TEMP+1  ;GET HIGH ORDER
179 004443 104200 000177 001403      MOV      #127.,DDUMMY     ;ADD 127 FOR DIV FUNCTION
180 004446 114000 001404          CLR      DDUMMY+1         ;FOR CLEAR
181 004450 104204 001410          MOV      #TEMP,R4        ;FOR ADD
182 004452 104203 001403          MOV      #DDUMMY,R3      ;DITTO
183 004454          CALL     DADD           ;ADD
184 004456 104200 000200 001403      MOV      #128.,DDUMMY    ;FOR DIVIDE (128 RBN/RCT BLOCK)
185 004461 114000 001404          CLR      DDUMMY+1         ;FOR STORE
186 004463 104203 001403          MOV      #DDUMMY,R3      ;POINT TO IT
187 004465          CALL     DDIV           ;DO DIVIDE
188 004467 104140 001746          MOV      (R4),RCTLBN     ;GET LOW ORDER QUOTIENT
189 004471 105200 000002 001746      ADD      #2,RCTLBN       ;FOR CONTROL BLOCKS
190                                     :
191                                     :
192                                     :
193 004474 104300 001620 001410      MOV      RBNLBN,TEMP      ;GET LOW ORDER LBN'S IN HOST AREA
194 004477 104300 001621 001411      MOV      RBNLBN+1,TEMP+1  ;GET HIGH ORDER
195 004502 104200 000002 001403      MOV      #2,DDUMMY       ;FOR DIVIDE BY 2
196 004505 114000 001404          CLR      DDUMMY+1         ;DITTO
197 004507 104204 001410          MOV      #TEMP,R4        ;SETUP
198 004511 104203 001403          MOV      #DDUMMY,R3      ;SETUP
199 004513          CALL     DDIV           ;CALL DIVIDE
200 004515 104200 000177 001403      MOV      #127.,DDUMMY    ;ADD 127 FOR DIV FUNCTION
201 004520 114000 001404          CLR      DDUMMY+1         ;FOR CLEAR
202 004522 104204 001410          MOV      #TEMP,R4        ;FOR ADD
203 004524 104203 001403          MOV      #DDUMMY,R3      ;DITTO
204 004526          CALL     DADD           ;ADD
205 004530 104200 000200 001403      MOV      #128.,DDUMMY    ;FOR DIVIDE (128 RBN/RCT BLOCK)
206 004533 114000 001404          CLR      DDUMMY+1         ;FOR STORE
207 004535 104203 001403          MOV      #DDUMMY,R3      ;POINT TO IT
208 004537          CALL     DDIV           ;DO DIVIDE
209 004541 104140 001723          MOV      (R4),FCTSUB     ;FCT SUBTABLE SIZE
210 004543 104640 000001 001724      MOV      1(R4),FCTSUB+1  ;HIGH ORDER
211 004546 104200 000002 001403      MOV      #2,DDUMMY       ;FOR MULT
212 004551 114000 001404          CLR      DDUMMY+1         ;CLEAR HIGH WORD
213 004553 104203 001403          MOV      #DDUMMY,R3      ;FOR DIVIDE
214 004555          CALL     DMUL           ;DO MULTIPLY
215 004557 104140 001745          MOV      (R4),FCTNPD     ;NON-PAD FCT BLOCKS
216 004561 115400 001745          INC      FCTNPD          ;FOR NON-PAD FCT BLOCKS
217                                     :
218                                     :
219                                     :
220 004563 104203 011132          MOV      #GDBLK,R3       ;POINT TO BUFFER
221 004565 104302 002131          MOV      DWRD,R2         ;DIAGNOSTIC WORD
222 004567 100232          MOV      R2,(R3)+        ;STORE IT
223 004570 104204 000125          MOV      #85.,R4        ;SET COUNTER
224 004572 104302 002126          MOV      FWRD,R2         ;FIRST WORD OF PATTERN
225 004574 100232          MOV      R2,(R3)+        ;STORE IT
226 004575 104302 002127          MOV      SWRD,R2         ;SECOND WORD OF PATTERN
227 004577 100232          MOV      R2,(R3)+        ;STORE IT
228 004600 104302 002130          MOV      TWRD,R2         ;THIRD WORD OF PATTERN

```

OVER:

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 40-4
INITIALIZATION OVERLAY (G1)

229	004602	100232			MOV	R2,(R3)+		;STORE IT
230	004603	117404			DEC	R4		;DECREMENT COUNTER
231	004604				BNE	OVER		;REPEAT TILL DONE
232	004606	104302	002132		MOV	EDC,R2		;EDC FOR PATTERN
233	004610	100232			MOV	R2,(R3)+		;STORE IT
234								
235								
236								
237								
238	004611	104203	001606		MOV	#SECTRK,R3		;SEC/TRACK
239	004613	104200	000003	001410	MOV	#IMLEN,TEMP		;FOR MULT
240	004616	114000	001411		CLR	TEMP+1		;FOR STORE
241	004620	104204	001410		MOV	#TEMP,R4		;SET UP FOR MULT
242	004622				CALL	DMUL		;GET LENGTH OF IMAGE BLOCK
243	004624	104200	015763	001403	MOV	#IMAGE,DDUMMY		;FOR ADD
244	004627	114000	001404		CLR	DDUMMY+1		;CLEAR HIGH BYTE
245	004631	104203	001403		MOV	#DDUMMY,R3		;SET UP FOR ADD
246	004633				CALL	DADD		;ADD TO GET ADDRESS
247	004635	104140	001714		MOV	(R4),EIMAGE		;GET ADDRESS

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 41
INITIALIZATION OVERLAY (G1)

1									
2									
3									
4									
5									
6	004637	104207	001512						
7	004641	104673	000007						
8	004643	110703							
9	004644	103203	177400						
10	004646	104030	001410						
11	004650	114000	001411						
12	004652	104204	001410						
13	004654	104203	001606						
14	004656								
15	004660	106300	001720	001410					
16	004663								
17	004665	104200	000011	001717					
18	004670	104200	000044	001716					
19	004673								
20	004675	104200	000006	001717	TWO:				
21	004700	104200	000033	001716					
22	004703				ISKIP:				

COMPUTE INTERLEAVE FACTOR

```

MOV    #CR,R0           ;POINT TO CHARACTERISTICS BLOCK
MOV    REVSEC(R0),R3    ;GET REVS/SECOND
SWAB   R3               ;GET INTO LOW BYTE
BIC    #HIBYTE,R3      ;CLEAR HIGH BYTE
MOV    R3,TEMP          ;FOR MULTIPLY
CLR    TEMP+1           ;CLEAR FOR STORE
MOV    #TEMP,R4         ;SET UP FOR MULTIPLY
MOV    #SECTRK,R3       ;SECTORS/TRACK
CALL   DMUL             ;GET SECTORS/SECOND
CMP    CUTOFF,TEMP      ;WITHIN LIMITS ?
BPL    TWO              ;DO BI-LEAVE
MOV    #THREB,TBLK     ;ELSE DO TRI-LEAVE
MOV    #44,SKPCNT      ;INIT CHECK PASS OFFSET
BR     ISKIP           ;SKIP BI-LEAVE SETUP
MOV    #TWOB,TBLK      ;DO BI-LEAVE
MOV    #33,SKPCNT      ;INIT CHECK PASS OFFSET

```

UDAF52 - UDA-52 FORMATTER DMAFR X04.01 23-AUG-82 13:14:22 PAGE 42
INITIALIZATION OVERLAY (G1)

1									
2									
3									
4	004703	104207	001512						
5	004705	104673	000001						
6	004707	110703							
7	004710	103203	177760						
8	004712	104030	001731						
9	004714	104207	001525						
10	004716	104670	000010	001725					
11	004721	114000	001726						
12	004723	102200	020000	001703					
13	004726								
14	004730	104670	000020	001727					
15	004733								
16	004735	104670	000014	001727	1\$:				
17	004740	114000	001730		2\$:				
18									
19									
20									
21									
22									
23									
24	004742	104300	001616	001410					
25	004745	104300	001617	001411					
26	004750	104204	001410						
27	004752	104203	001620						
28	004754								
29	004756	104203	002151						
30	004760								
31	004762	104642	000001						
32	004764	107302	002007						
33	004766	104020	001640						
34	004770	104140	001637						

FILL IN FCT INFO

```

MOV #CR,R0 ;POINT TO CHARACTERISTICS BLK
MOV FRCPY(R0),R3 ;GET F/RCT COPIES
SWAB R3 ;GET INTO LOW BYTE
BIC #FCLR,R3 ;CLEAR OUT REST OF GARBAGE
MOV R3,FCTCPY ;STORE
MOV #SCR,R0 ;POINT TO SUBUNIT CHARACTERISTICS
MOV FCTSZ(R0),FCTFMT ;GET FCT SIZE IN SECTORS
CLR FCTFMT+1 ;CLEAR HIGH ORDER
BIT #MODE,FLAG1 ;WHAT MODE ARE WE IN ?
BEQ 1$ ;IF CLEAR THEN IN 512 MODE
MOV RCTS76(R0),RCTFMT ;ELSE GET 576 RCT SIZE
BR 2$ ;SKIP 512 STUFF
MOV RCTS12(R0),RCTFMT ;GET RCT SIZE
CLR RCTFMT+1 ;CLEAR HIGH ORDER

```

COMPUTE HIGHEST PBN IN LBN AREA

```

MOV LBNLBN,TEMP ;GET LOW ORDER
MOV LBNLBN+1,TEMP+1 ;GET HIGH ORDER
MOV #TEMP,R4 ;FOR SUB
MOV #RBNLBN,R3 ;POINT TO RBN'S IN LBN AREA
CALL DADD ;ADD TO GET HIGHEST PBN
MOV #ONE,R3 ;DITTO
CALL DSUB ;TO GET ACTUAL PBN NUMBER
MOV 1(R4),R2 ;GET HIGH ORDER
SUB ST.LBN,R2 ;SUBTRACT TO GET RELATIVE LAST BLOCK
MOV R2,HGHPBN+1 ;STORE IT
MOV (R4),HGHPBN ;GET HIGH ORDER

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 43
INITIALIZATION OVERLAY (G1)

```

1
2
3
4 004772 104200 000003 001410
5 004775 114000 001411
6 004777 104204 001410
7 005001 104203 001606
8 005003
9 005005 104303 001410
10 005007 104204 015763
11 005011 105034
12 005012 104040 001707
13 005014 104040 001737
14 005016 104200 037777 001410
15 005021 114000 001411
16 005023 104040 001403
17 005025 114000 001404
18 005027 104204 001410
19 005031 104203 001403
20 005033
21 005035 104304 001410
22 005037 110604
23 005040 104040 001710
24
25
26
27 005042 104207 001512
28 005044 104673 000002
29 005046 103203 177400
30 005050 104030 002166
31 005052 104030 002170
32 005054 104673 000001
33 005056 110603
34 005057 110603
35 005060 110603
36 005061 110603
37 005062 103203 177700
38 005064 115403
39 005065 104030 002165
40 005067 114000 002167
41
42
43
44
45
46 005071 104207 001512
47 005073 104673 000001
48 005075 103203 177760
49 005077
50 005101 115403
51 005102 104201 000001
52 005104 105201 000000
53 005106 110201
54 005107 117403
55 005110
56 005112 104070 001410
57 005114 114000 001411

```

COMPUTE REVECTOR BUFFER ADDRESS AND MAX REVECTOR COUNT
 MOV #IMLEN,TEMP ;GET LENGTH OF FORMAT IMAGE BLOCK
 CLR TEMP+1 ;CLEAR FOR STORE
 MOV #TEMP,R4 ;FOR MULT
 MOV #SECTRK,R3 ;SECTORS/TRACK
 CALL DMUL ;GET LENGTH OF FORMAT BUFFER TABLE
 MOV TEMP,R3 ;GET LENGTH
 MOV #IMAGE,R4 ;GET IMAGE BUFFER START ADDRESS
 ADD R3,R4 ;GET START ADDRESS OF REVECTOR BUFFER
 MOV R4,ERRBUF ;STORE IT
 MOV R4,ERPNT ;INIT POINTER
 MOV #BMAX,TEMP ;GET MAX BUFFER ADDRESS
 CLR TEMP+1 ;FOR CLEAR
 MOV R4,DDUMMY ;STORE BEGINNING ADDRESS
 CLR DDUMMY+1 ;CLEAR HIGH WORD
 MOV #TEMP,R4 ;POINT TO END ADDRESS
 MOV #DDUMMY,R3 ;POINT TO BEGINNING ADDRESS
 CALL DSUB ;SUBTRACT TO GET LENGTH
 MOV TEMP,R4 ;GET LENGTH
 ROR R4 ;DIVIDE BY 2 (LENGTH OF 1 ENTRY)
 MOV R4,EMAX ;STORE AS MAX NUMBER

STORE RETRY AND RECOVERY LEVELS
 MOV #CR,R0 ;POINT TO CHARACTERISTICS
 MOV ERVC(R0),R3 ;GET RECOVERY LEVELS
 BIC #HIBYTE,R3 ;CLEAR HIGH ORDER
 MOV R3,RECOV ;STORE IT
 MOV R3,RECTMP ;INIT COUNTER
 MOV RTRY(R0),R3 ;GET RETRY NUMBER
 ROR R3 ;SHIFT BY FOUR TO GET INTO LOW ORDER NIBBLE
 ROR R3
 ROR R3
 ROR R3
 BIC #HIBYTE,R3 ;CLEAR HIGH ORDER JUNK
 INC R3 ;ONE MORE BECAUSE OF WRONG INC
 MOV R3,RETRY ;STORE IT
 CLR TMPTRY ;FOR STORE

SET UP LONG TIMEOUT
 MOV #CR,R0 ;POINT TO COMMON CHARACTERISTICS
 MOV LONGTO(R0),R3 ;GET LONG TIMEOUT IN LOG2
 BIC #FCLR,R3 ;CLEAR ALL BUT TIMEOUT
 BNE TIMLO1 ;IF NOT ZERO THEN CONTINUE
 INC R3 ;MAKE IT AT LEAST 1
 TIMLO1: MOV #1,R1 ;INIT COUNTER
 ADD #0,R1 ;CLEAR CARRY
 TIMLOP: ROL R1 ;SHIFT
 DEC R3 ;DECREMENT SHIFT COUNT
 BNE TIMLOP ;CONTINUE TILL DONE
 MOV R0,TEMP ;FOR DIVIDE
 CLR TEMP+1 ;CLEAR HIGH WORD

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 43-1
INITIALIZATION OVERLAY (G1)

```

58 005116 104200 000012 001403      MOV    #10.,DDUMMY      :FOR DIVIDE BY 10
59 005121 114000 001404              CLR    DDUMMY+1        :CLEAR HIGH WORD
60 005123 104204 001410              MOV    #TEMP,R4        :DIVIDE BY 10 (UDA RECEIVE TIMEOUT)
61 005125 104203 001403              MOV    #DDUMMY,R3     :TO GET VALUE TO USE AS LTO
62 005127              CALL   DDIV             :DO DIVIDE
63 005131 104140 001735              MOV    (R4),LTO       :STORE IT FOR US LATER
64 005133 115400 001735              INC    LTO            :MAKE SURE AT LEAST 1
65
66      :
67      :
68      :
69      :
70 005135 104207 001512              MOV    #CR,R0         :POINT TO COMM CHAR
71 005137 104673 000000              MOV    SHORTO(R0),R3  :GET SHORT TIMEOUT WORD
72 005141 103203 177760              BIC    #FCLR,R3       :CLEAR WORD
73 005143 117403              DEC    R3             :FOR LOG CALCULATION
74 005144 104201 000001              MOV    #1,R1         :INIT COUNTER
75 005146 105201 000000              ADD    #0,R1         :CLEAR CARRY
76 005150 110201              TILOP: ROL    R1      :ROTATE (MULT BY 2)
77 005151 117403              DEC    R3             :DECREMENT COUNTER
78 005152              BNE    TILOP         :KEEP GOING TILL DONE
79 005154 104203 000012              MOV    #10.,R3       :SHIFT COUNT FOR MILSEC CONVERSION
80 005156 110201              TILOP1: ROL    R1    :GET NUMBER IN MILSECS
81 005157 117403              DEC    R3             :GO TILL DONE
82 005160              BNE    TILOP1       :AGAIN
83 005162 104010 001736              MOV    R1,STO        :STORE IT
84
85      :
86      :
87      :
88      :
89      :
90 005164 104207 001525              MOV    #SCR,R0       :POINT TO CHARACTERISTICS BLK
91 005166 104670 000005 002006      MOV    DATA(R0),DPREA :DATA PREAMBLE LENGTH
92 005171 103200 177400 002006      BIC    #HIBYTE,DPREA  :CLEAR OUT HIGH BYTE GARBAGE
93 005174 104673 000005              MOV    HEAD(R0),R3   :HEADER PREAMBLE LENGTH
94 005176 110703              SWAB   R3            :GET INTO LOW BYTE
95 005177 103203 177400              BIC    #HIBYTE,R3    :CLEAR OUT HIGH BYTE GARBAGE
96 005201 104030 002005              MOV    R3,HPREA      :STORE IT
97 005203 114001              CLR    R1            :FOR NO ERROR

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 44
INITIALIZATION OVERLAY (G1)

1				...		
2				...	SET UP STARTING ADDRESS BITS	
3				...		
4	005204	104203	001525	...	MOV #SCR,R3	;POINT TO SUBUNIT CHAR BLOCK
5				...		
6				...	LBN	
7				...		
8	005206	104637	000002	...	MOV STLBN(R3),RO	;GET THE WORD
9	005210	103207	170377	...	BIC #STCLR,RO	;CLEAR THE REST
10	005212	104070	002007	...	MOV RO,ST.LBN	;STORE IT
11				...		
12				...	RBN	
13				...		
14	005214	104637	000003	...	MOV STRBN(R3),RO	;GET RBN WORD
15	005216	103207	170377	...	BIC #STCLR,RO	;CLEAR THE REST
16	005220	104070	002010	...	MOV RO,ST.RBN	;STORE IT
17				...		
18				...	XBN	
19				...		
20	005222	104637	000002	...	MOV STXBN(R3),RO	;GET THE WORD
21	005224	110607		...	ROR RO	
22	005225	110607		...	ROR RO	
23	005226	110607		...	ROR RO	
24	005227	110607		...	ROR RO	;GET INTO THE RIGHT NIBBLE
25	005230	103207	170377	...	BIC #STCLR,RO	;CLEAR THE REST
26	005232	104070	002011	...	MOV RO,ST.XBN	;STORE IT
27				...		
28				...	DBN	
29				...		
30	005234	104637	000003	...	MOV STDBN(R3),RO	;GET THE WORD
31	005236	110607		...	ROR RO	
32	005237	110607		...	ROR RO	
33	005240	110607		...	ROR RO	
34	005241	110607		...	ROR RO	;GET INTO THE RIGHT NIBBLE
35	005242	103207	170377	...	BIC #STCLR,RO	;CLEAR THE REST
36	005244	104070	002012	...	MOV RO,ST.DBN	;STORE IT

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 45
INITIALIZATION OVERLAY (G1)

1					
2					
3					
4	005246	104203	001512	CLEAR ECC THRESHOLD	
5	005250	104632	000002	MOV #CR,R3	:POINT TO CHARACTERISTICS
6	005252	103202	177400	MOV ERRSYM(R3),R2	:GET THE WORD
7	005254	100632	000002	BIC #HIBYTE,R2	:CLEAR THE THRESHOLD
8	005256			MOV R2,ERRSYM(R3)	:STORE IT BACK
				RETURN	

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 46
INITIALIZATION OVERLAY (G1)

```

1
2
3
4
5
6 005260 104205 001750          GETUNT: MOV    #DMBUF,R5          ;POINT TO STARTUP INFO
7 005262 114000 002164          CLR    COUNT                    ;FOR INTERCONNECT INIT
8 005264 104200 000001 001412  GOVER1: MOV    #1,UNIT           ;INIT INTERCONNECT
9 005267 104302 001412          MOV    UNIT,R2                  ;GET INTERCONNECT INTO R2
10 005271          CALL  STATVL                    ;ANY DRIVE THERE ???
11 005273          BNE   NOTHER                    ;NOPE - TRY NEXT ONE
12 005275 104203 001414          GOVER: MOV    #CR.GST,R3        ;FOR GET STATUS
13 005277 104237          MOV    (R3)+,R0                 ;GETCOMMAND ADDRESS
14 005300 104231          MOV    (R3)+,R1                 ;GET COMMAND SIZE
15 005301 104302 001412          MOV    UNIT,R2                  ;GET INTERCONNECT
16 005303 060004          XFC   SEND                      ;ISSUE GET STATUS COMMAND
17 005304 115001          TST   R1                        ;SUCCESSFUL ?
18 005305          BEQ   GSKIP1                    ;YUP - SKIP PTRY
19 005307 115400 001371          INC   UN.ERI                    ;INC COUNT
20 005311 106300 002165 001371  CMP   RETRY,UN.ERI              ;DONE ALL RETIES ?
21 005314          BMI   NOTHER                    ;YUP
22 005316          BR   GOVER                      ;
23 005320 114000 001371          GSKIP1: CLR   UN.ERI             ;FOR ERROR CLEAR
24 005322 104231          MOV   (R3)+,R1                  ;GET RECEIVE BUFFRE
25 005323 104137          MOV   (R3),R0                   ;GET BUFFER LENGTH
26 005324 060005          XFC   RCV                       ;RECEIVE SDI RESPONSE
27 005325 115001          TST   R1                        ;SUCCESSFUL ?
28 005326          BEQ   GSKIP2                    ;YUP - SKIP RETRY
29 005330 115400 002164          INC   COUNT                      ;INC ERROR COUNT
30 005332 106300 002165 002164  CMP   RETRY,COUNT              ;DONE ALL RETRIES ?
31 005335          BMI   NOTHER                    ;YUP
32 005337          BR   GOVER                      ;TRY AGAIN
33 005341 114000 002164          GSKIP2: CLR   COUNT              ;FOR COUNT CLEAR
34 005343 104204 001503          MOV   #ST,R4                    ;POINT TO STATUS BUFFER
35 005345 104641 000000          MOV   UID(R4),R1                ;GET UNIT NUMBER FROM STATUS
36 005347 103201 170000          BIC   #HD.CLR,R1                ;CLEAR GARBAGE
37 005351 106301 001413          CMP   UNNO,R1                   ;IS IT THE ONE ?
38 005353          BMI   NOTHER                    ;NOT EVEN A POSSIBLE SUBUNIT
39 005355          BNE   GSKIP3                    ;NO - SEE IF A SUBUNIT
40 005357 104202 000001          MOV   #1,R2                      ;SET SUBUNIT MASK
41 005361          BR   GDONE                       ;AND EXIT
42 005363 104643 000000          GSKIP3: MOV   MASK(R4),R3        ;GET SUBUNIT MASK
43 005365 103203 007777          BIC   #LO,R3                     ;CLEAR GARBAGE
44 005367 110703          SWAB  R3                         ;GET INTO LOW BYTE
45 005370 110603          ROR   R3                         ;TO GET IN LOW NIBBLE
46 005371 110603          ROR   R3                         ;TO GET IN LOW NIBBLE
47 005372 110603          ROR   R3                         ;TO GET IN LOW NIBBLE
48 005373 110603          ROR   R3                         ;TO GET IN LOW NIBBLE
49 005374 114007          CLR   RO                          ;CLEAR COUNTER
50 005375 102203 000002          BIT   #BIT1,R3                  ;ANYTHING THERE ?
51 005377          BEQ   GSKIP                       ;NOPE - ONLY 1
52 005401 115407          INC   RO                          ;YUP - INCREMENT COUNTER
53 005402 102203 000004          BIT   #BIT2,R3                  ;ANYTHING HERE ?
54 005404          BEQ   GSKIP                       ;NOPE - ONLY 2
55 005406 115407          INC   RO                          ;INC COUNTER
56 005407 102203 000010          BIT   #BIT3,R3                  ;ANYTHING HERE ?
57 005411          BEQ   GSKIP                       ;NOPE

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 46-1
INITIALIZATION OVERLAY (G1)

58	005413	115407		INC	R0		:YUP
59	005414	105071		GSKIP: ADD	R0,R1		:ADD TO UNIT NUMBER
60	005415	107301	001413		SUB	UNNO,R1	:GET RELATIVE OFFSET FROM GIVEN UNIT NUMBER
61	005417				BMI	NOTHER	:IF NEGATIVE THEN NOT IN THE RANGE
62	005421	104303	001413		MOV	UNNO,R3	:GET DESIRED
63	005423	104641	000000		MOV	UID(R4),R1	:GET ORIGINAL
64	005425	103201	170000		BIC	#HD.CLR,R1	:CLEAR SUBUNIT MASK
65	005427	107013			SUB	R1,R3	:FIGURE OUT WHICH SUBUNIT
66	005430	104202	000001		MOV	#1,R2	:INIT SUBUNIT MASK
67	005432	105202	000000		ADD	#0,R2	:TO CLEAR CARRY
68	005434	110202		SFTRPT: ROL	R2		:SHIFT LEFT
69	005435	117403			DEC	R3	:DECREMTN COUNTER
70	005436				BNE	SFTRPT	:REPEAT SHIFT
71	005440	105202	000000	GDONE: ADD	#0,R2		:CLEAR CARRY
72	005442	110202			ROL	R2	:HAVE TO ROTATE
73	005443	110202			ROL	R2	:4 TIMES TO DO A
74	005444	110202			ROL	R2	:SWAP NIBBLE
75	005445	110202			ROL	R2	:FOR SUBUNIT MASK
76	005446	104020	001473		MOV	R2,GSR+1	:STORE MASK IN SUBUNIT CHAR COMMAND
77	005450	101020	001500		BIS	R2,ACC+1	:SET IN FOR CHANGE MODE COMMAND
78	005452				RETURN		:RETURN
79	005454	104302	001412	NOTHER: MOV	UNIT,R2		:GET CURRENT INTERCONNECT
80	005456	105022			ADD	R2,R2	:NEXT PORT
81	005457	104020	001412		MOV	R2,UNIT	:SAVE BACK
82	005461	114000	001371		CLR	UN.ERI	:FOR RESTORE
83	005463	114000	002164		CLR	COUNT	:CLEAR ERROR COUNT
84	005465	106202	000010		CMP	#8.,R2	:ALL DONE ?
85	005467				BPL	GOVER1	:NOPE - TRY THIS INTERCONNECT
86	005471	104652	000002		MOV	2(R5),R2	:UNIT SEARCHING FOR
87	005473	104201	000007		MOV	#7,R1	:SIGNAL NON-EXISTANT UNIT
88	005475				CALL	ERRMNT	:ERROR RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 47
 INITIALIZATION OVERLAY (G1)

```

1
2
3
4 005477 102200 020000 001703 CKSS: BIT #MODE,FLAG1 ;WHAT MODE DO WE WANT ?
5 005502 BEQ N0576 ;512 - ALL DRIVES HANDLE IT
6 005504 104207 001512 MOV #CR,R0 ;POINT TO CHARACTERISTICS
7 005506 104677 000001 MOV SSBIT(R0),R0 ;GET SSBIT WORD
8 005510 102207 100000 BIT #SS,R0 ;CHECK SS BIT
9 005512 BEQ CKERR ;DRIVE ONLY ALLOWS 512 - ERROR
10 005514 N0576: RETURN ;RETURN
11 005516 104201 000026 CKERR: MOV #22.,R1 ;ERROR 22
12 005520 CALL ERRMNT ;DIE

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 48
INITIALIZATION OVERLAY (G1)

```
1  
2  
3  
4  
5 005522 104302 C01412  
6 005524  
7 005526  
8 005530  
9 005532  
10 005534  
11 005536  
12 005540  
13 005542
```

.....
INITL: MOV UNIT,R2
 CALL INITI
 CALL GETUNT
 CALL ONLIN
 CALL GSTATS
 CALL CKSS
 CALL RECAL
 CALL CONINT
 RETURN

 :SELECT UNIT
 :INIT DRIVE
 :GET THE SDI INTERCONNECT
 :BRING IT ONLINE
 :GET STATUS
 :CHECK VALIDITY OF FORMAT MODE
 :RECALIBRATE
 :COMPUTE DISK CONSTANTS
 :AND STOP

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 49
 DBN/XBN FORMAT OVERLAY (F1)

```

1          .SBTTL  DBN/XBN FORMAT OVERLAY (F1)
2
3          :
4          :
5          :
6          :
7          :
8 005544   DMOVLY  F1,START
9
10 004014  104200  000000  001636  MOV      #F1,CUROVL      ;OVERLAY #1
11 004017  104200  000400  002171  MOV      #SECSI6,SECSIZ ;MAKE SECTOR SIZE 512 FOR D/XBN
12 004022  101200  000010  001702  BIS      #DBN,FLAG      ;SET DBN FORMAT
13 004025   CALL     DXFORM      ;FORMAT DBN AREA
14 004027  104303  001776   MOV      FCTREV,R3      ;STARTING FCT ENTRY COUNT
15 004031  107303  001722   SUB     FCNT,R3         ;TOTAL BAD IN DBN AREA
16 004033  105303  002136   ADD     ERRCNT,R3      ;GET FINAL TOTAL
17 004035  104030  002001   MOV     R3,DBBAD      ;STORE IT FOR STATS
18 004037  114000  002136   CLR     ERRCNT        ;FOR CLEAR
19 004041  104300  001722  001776  MOV     FCNT,FCTREV    ;FOR BAD BLOCK COUNT
20 004044  103200  000010  001702  BIC     #DBN,FLAG      ;DO XBN AREA
21 004047   CALL     DXFORM      ;FORMAT XBN AREA
22 004051  104303  001776   MOV     FCTREV,R3     ;STARTING FCT ENTRY COUNT
23 004053  107303  001722   SUB     FCNT,R3         ;TOTAL BAD IN XBN AREA
24 004055  105303  002136   ADD     ERRCNT,R3      ;GET FINAL TOTAL
25 004057  104030  002002   MOV     R3,XBBAD      ;STORE IT FOR STATS
26 004061  114000  002136   CLR     ERRCNT        ;FOR CLEAR
27 004063  104201  000006   MOV     #F3,R1        ;FCT DLL OVERLAY
28 004065   CALL     NEXT        ;BRING IN NEXT OVERLAY
29
30 004067  104207  001525   DXFORM: MOV    #SCR,R0   ;POINT TO CHARACTERISTICS BLOCK
31 004071  102200  000010  001702  BIT     #DBN,FLAG      ;DO DBN AREA ?
32 004074   BEQ     XBNIT        ;NO - DO XBN AREA
33 004076  104673  000022   MOV     DBNCYL(R0),R3  ;GET NUMBER OF CYLS TO FM
34 004100  110703   SWAB    R3            ;GET INTO LOW BYTE
35 004101  103203  177400   BIC     #HIBYTE,R3    ;CLEAR HI BYTE
36 004103  104030  002143   MOV     R3,CNTCYL     ;SET UP COUNTER
37 004105  102200  002000  001702  BIT     #BSTGS,FLAG    ;DOING BEST GUESS ???
38 004110   BNE    SKIP4        ;YES - SKIP FCT SET UP
39 004112  104200  000200  002163  MOV     #128,PCNT     ;FOR PBN COUNT INIT
40 004115  114000  001743   CLR     FCTCNT        ;FOR INIT FCT READ
41 004117   CALL   DXFCPG       ;READ IT IN
42 004121  104200  010455  001706  MOV     #PBNBUF,BADPBN ;ADDR OF BAD PBN LIST
43 004124  104300  010473  001722  MOV     PBNBUF+C512,FCNT ;GET COUNT OF USED ENTRIES
44 004127  104300  010473  001776  MOV     PBNBUF+C512,FCTREV ;STORE IT FOR STAT COMPUTATION
45 004132   BEQ    SKIP19       ;IF ZERO - THEN NO ENTRIES
46 004134  115400  001743   INC     FCTCNT        ;START WITH SECOND FCT BLOCK
47 004136   CALL   DXFCPG       ;BRING IT IN
48 004140   BR     SKIP4        ;SKIP NO ENTRY STUFF
49 004142  101200  000002  001702  SKIP19: BIS    #FCTEMT,FLAG ;SET EMPTY FLAG
50 004145  104200  140000  002145  SKIP4:  MOV     #HD.DBN,HD.CUR ;GET DBN HEADER CODE
51 004150   CALL   NUMDBN      ;GET DBN OF FIRST BLOCK ON LAST CYLINDER
52 004152   BR     SKIP1        ;SKIP XBN SETUP
53 004154  104200  120000  002145  XBNIT:  MOV     #HD.XBN,HD.CUR ;GET XBN HEADER CODE
54 004157  104670  000021  002143  MOV     XBNCYL(R0),CNTCYL ;GET CYLINDERS IN XBN AREA
55 004162   CALL   NUMXBN      ;GET XBN OF FIRST BLOCK ON LAST XBN CYL
56 004164  104140  001576  SKIP1:  MOV     (R4),HOLDBN    ;LO ORDER FIRST BLOCK NUM TO DO
57 004166  104240  001566   MOV     (R4)+,CURBN   ;AND MAKE IT CURRENT NUMBER

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 49-1
DBN/XBN FORMAT OVERLAY (F1)

58	004170	104140	001577		MOV	(R4),HOLDBN+1	:HI ORDER FIRST BLOCK NUM TO DO
59	004172	104140	001567		MOV	(R4),CURBN+1	:AND MAKE IT CURRENT NUMBER
60	004174	104207	001525		SLEEK: MOV	#SCR,R0	:POINT TO CHARACTERISTICS BLK
61	004176	104670	000002	002147	MOV	GRPCYL(R0),GRPCNT	:LOAD GROUPS/CYL
62	004201	103200	177400	002147	BIC	#HIBYTE,GRPCNT	:CLEAR OUT HIGH GARBAGE
63	004204	104300	002147	002146	MOV	GRPCNT,CURGRP	:GET GROUP NUMBER BY
64	004207	117400	002146		DEC	CURGRP	:DECREMENTING
65	004211	104300	001604	001551	SLEEK2: MOV	CYLNUM,ISEEK+1	:GET LO ORDER WORD OF CYLINDER NUMBER
66	004214	104300	001605	001552	MOV	CYLNUM+1,ISEEK+2	:LOAD HIGH ORDER WORD OF CYL NUM
67	004217	104300	002146	001553	MOV	CURGRP,ISEEK+3	:LOAD GROUP NUMBER
68	004222	103200	100000	001552	BIC	#SS,ISEEK+2	:MAKE SURE IT IS IN 512 MODE
69	004225				CALL	SEEK	:SEEK TO CURRENT CYL NUM
70	004227	115001			TST	R1	:ANY ERRORS ?
71	004230				BMI	SKERR	:YUP - QUIT
72	004232	104207	001525		MOV	#SCR,R0	:POINT TO CHARACTERISTICS BLOCK
73	004234	104673	000003		MOV	TRKGRP(R0),R3	:LOAD TRACKS/GROUP
74	004236	103203	177400		BIC	#HIBYTE,R3	:CLEAR OUT HIGH GARBAGE
75	004240	104030	002150		MOV	R3,TRKCNT	:STORE IN COUNTER
76	004242	117403			DEC	R3	:TRACK NUMBER IS ONE LESS
77	004243	104030	001565		MOV	R3,CURTRK	:RESET CURRENT TRACK NUMBER
78	004245	104201	000047		SKIP3: MOV	#G7,R1	:FORMAT SETUP OVERLAY
79	004247				CALL	PAGE	:DO IT
80	004251	104304	002006		SKIP7: MOV	DPREA,R4	:DATA PREAMBLE LENGTH
81	004253	104303	002005		MOV	HPREA,R3	:HEADER PREAMBLE LENGTH
82	004255	104307	002004		MOV	IMSTAR,R0	:POINT TO FORMAT IMAGE START POINT
83	004257	104301	001565		MOV	CURTRK,R1	:TRACK TO FORMAT
84	004261	104302	002171		MOV	SECSIZ,R2	:SECTOR SIZE IN WORDS
85	004263	104205	015763		MOV	#IMAGE,R5	:RECIRCULATION POINTER
86	004265	060001			XFC	FORMAT	:FORMAT THE TRACK
87	004266	115001			TST	R1	:ANY ERRORS ?
88	004267				BEQ	SKIP6	:NO - DO CHECK PASS
89	004271	115400	001371		INC	UN.ERI	:INCREMENT IT
90	004273	106300	002165	001371	CMP	RETRY,UN.ERI	:DONE ALL RETRIES ?
91	004276				BMI	FERR	:YUP - ERROR
92	004300				CALL	INITPT	:REINIT
93	004302				CALL	CLEAR	:DRIVE CLEAR
94	004304				CALL	SEEK	:RE-SEEK AND GROUP SELECT
95	004306				BR	SKIP7	:RETRY FORMAT
96	004310	114000	001371		SKIP6: CLR	UN.ERI	:FOR STORE
97	004312				CALL	DXCHEC	:DO CHECK PASS
98	004314	117400	001565		DEC	CURTRK	:DECREMENT IT
99	004316	104204	001576		MOV	#HOLDBN,R4	:PREPARE FOR BEGINNING BLOCK DECREMENT
100	004320	104203	001606		MOV	#SECTRK,R3	:DECREMENT BY SECTORS/TRACK
101	004322				CALL	DSUB	:DO DECREMENT
102	004324	104300	001576	001566	MOV	HOLDBN,CURBN	:LO ORDER NEW BLOCK NUMBER
103	004327	104300	001577	001567	MOV	HOLDBN+1,CURBN+1	:HI ORDER NEW BLOCK NUMBER
104	004332	117400	002150		DEC	TRKCNT	:DECREMENT IT
105	004334				BNE	SKIP3	:NO - DO NEXT TRACK
106	004336	117400	002146		DEC	CURGRP	:DECREMENT GROUP NUMBER
107	004340	117400	002147		DEC	GRPCNT	:DECREMENT IT
108	004342				BNE	SLEEK2	:NO - DO NEXT GROUP
109	004344	117400	002143		DEC	CNTCYL	:DECREMENT IT
110	004346	060022			XFC	UPDATE	:UPDATE PROGRESS INDICATOR
111	004347	104204	001604		MOV	#CYLNUM,R4	:PREPARE FOR CYL NUM DECREMENT
112	004351	104203	002151		MOV	#ONE,R3	:DECREMENT BY ONE
113	004353				CALL	DSUB	:DO SUBTRACT
114	004355	115000	002143		TST	CNTCYL	:ARE WE DONE ?

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 49-2
 DBN/XBN FORMAT OVERLAY (F1)

115 004357
 116 004361
 117 004363 104012
 118 004364 104201 000010
 119 004366
 120 004370 104302 001604
 121 004372 104201 000012
 122 004374

BNE SLEEK
 RETURN
 FERR: MOV R1,R2
 MOV #8.,R1
 BR DXERR
 SKERR: MOV CYLNUM,R2
 MOV #10.,R1
 DXERR: CALL ERRMNT

:DONE ? NO - DO NEXT CYLINDER
 :YES - ALL DONE
 :GET XFC FAILURE CODE
 :SET FORMAT ERROR
 :CYLINDER WHICH FAILED ON
 :SEEK ERROR
 :ERROR RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 50
 DBN/XBN FORMAT OVERLAY (F1)

```

1
2
3
4 004376
5 004377 104201 000033
6 004401
7 004403 104200 000200 002164
8 004406 104200 010455 001706
9 004411
10 004412
11
12
13
14
15
16
17 004414 104673 000022
18 004416 110703
19 004417 103203 177400
20 004421 104030 001410
21 004423 114000 001411
22 004425 104204 001410
23 004427 104203 001614
24 004431
25 004433 104641 000001
26 004435 105301 002012
27 004437 100641 000001
28 004441 104203 001606
29 004443
30 004445
31
32
33
34
35
36
37 004447 104670 000021 001410
38 004452 114000 001411
39 004454 104204 001410
40 004456 104203 001614
41 004460
42 004462 104641 000001
43 004464 105301 002011
44 004466 100641 000001
45 004470 104203 001606
46 004472
47 004474
48
49
50
51
52
53 004476
54 004477 114000 001711
55 004501 114000 001704
56 004503 102200 000600 001702
57 004506

```

```

:
: PAGE IN NEW FCT BLOCK
:
: DXFCPG: PUSH      RO
:           MOV      #G2,R1           ;DLL OVERLAY
:           CALL     PAGE             ;EXECUTE OVERLAY
:           MOV      #128.,COUNT    ;FOR INIT
:           MOV      #PBNBUF,BADPBN  ;FOR POINTER RESET
:           POP      RO              ;RESTORE RO
:           RETURN                     ;RETURN
:
:
: COMPUTE NUMBER OF FIRST DBN ON LAST DBN TRACK
: RO -> CHARACTERISTICS BLOCK
:
:
: NUMDBN: MOV      DBNCYL(RO),R3      ;GET NUMBER OF CYLINDERS IN DBN AREA
:           SWAB    R3                ;GET INTO LOW BYTE
:           BIC     #HIBYTE,R3        ;CLEAR OUT OTHER INFO
:           MOV     R3,TEMP            ;MOVE TO TEMP AREA
:           CLR     TEMP+1            ;CLEAR FOR STORE
:           MOV     #TEMP,R4          ;POINT R4 AT TEMP AREA
:           MOV     #SECTCY,R3        ;POINT TO NUM OF SECTORS/CYLINDER
:           CALL    DMUL              ;MULTIPLY TO GET SECTORS BEFORE LAST CYL
:           MOV     1(R4),R1          ;GET HIGH ORDER
:           ADD     ST.DBN,R1         ;ADD HIGH ORDER STARTING DBN
:           MOV     R1,1(R4)          ;STORE BACK
:           MOV     #SECTRK,R3        ;WANT FIRST DN OF LAST TRACK
:           CALL    DSUB              ;SUB TO GET IT
:           RETURN
:
:
: COMPUTE NUMBER OF FIRST XBN ON LAST XBN TRACK
: RO -> CHARACTERISTICS BLOCK
:
:
: NUMXBN: MOV      XBNCYL(RO),TEMP    ;GET NUMBER OF CYLINDERS IN XBN AREA
:           CLR     TEMP+1            ;CLEAR FOR STORE
:           MOV     #TEMP,R4          ;POINT TO TEMP AREA
:           MOV     #SECTCY,R3        ;POINT TO SECTORS/CYLINDER
:           CALL    DMUL              ;MULTIPLY TO GET SECTORS IN XBN AREA
:           MOV     1(R4),R1          ;GET HIGH ORDER
:           ADD     ST.XBN,R1         ;ADD HIGH ORDER STARTING XBN
:           MOV     R1,1(R4)          ;STORE BACK
:           MOV     #SECTRK,R3        ;WANT XBN OF LAST TRACK
:           CALL    DSUB              ;SUB TO GET IT
:           RETURN
:
:
: CHECK PASS
:
: DXCHEC: PUSH      RO
: DXCH:   CLR      ERR                ;SAVE PTR TO CHARACTERISTICS BLK
:         CLR     EQFLAG              ;FOR ERROR COUNT RESET
:         BIT     #MANU:DLL,FLAG      ;CLEAR RE-FORMAT FLAG
:         BEQ    CSKIP                ;FCT AVAILABLE ?
:         ;NO - DO EXTENSIVE READ

```


UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 50-1
DBN/XBN FORMAT OVERLAY (F1)

58	004510	104200	000001	002140	MOV	#1,N	:SET UP FOR STORE
59	004513	104200	000005	002141	MOV	#5,N1	:SET UP
60	004516	104300	002141	002142	MOV	N1,NM1	:SAVE FOR LATER RESET
61	004521				BR	CSKIP2	:SKIP EXTENSIVE READ SETUP
62	004523	104200	000003	002140	CSKIP: MOV	#3,N	:EXTENSIVE REGULAR READ
63	004526	104200	000024	002141	MOV	#20.,N1	:EXTENSIVE ERROR READS
64	004531	104300	002141	002142	MOV	N1,NM1	:SAVE FOR LATER RESET
65	004534				CSKIP2: CALL	FIXIT	:DO IT
66	004536	104302	001412		CSKIP1: MOV	UNIT,R2	:SET IN PORT NUMBER
67	004540	060012			XFC	SIP	:SYNCH WITH SECTOR/INDEX PULSE
68	004541	104302	002171		MOV	SECSIZ,R2	:SECTOR SIZE IN WORDS
69	004543	104300	001606	002137	MOV	SECTRK,SECCNT	:LOAD SECTORS/TRACK
70	004546	104205	013022		MOV	#CMDBUF,R5	:POINT TO COMMAND BUFFER
71	004550	104207	001373		AGAIN: MOV	#RDBLK,R0	:POINT TO READ COMMAND BLOCK
72	004552	104653	000002		MOV	RB.CMD(R5),R3	:ZERO COMMAND ?
73	004554				BEQ	NOERR	:YES - SKIP CHECKS
74	004556	100673	000005		MOV	R3,RW.CMD(R0)	:ELSE STORE IT
75	004560	104653	000000		MOV	RB.LOW(R5),R3	:GET LOW ORDER BLOCK NUMBER
76	004562	100673	000003		MOV	R3,RW.LOW(R0)	:STORE IN COMMAND BLOCK
77	004564	104653	000001		MOV	RB.HI(R5),R3	:LOAD HIGH ORDER BLOCK NUMBER
78	004566	100673	000004		MOV	R3,RW.HI(R0)	:STORE IN COMMAND BLOCK
79	004570	104203	010000		MOV	#RDBUF,R3	:GET BUFFER POINTER
80	004572	100673	000002		MOV	R3,RW.BUF(R0)	:STORE IN COMMAND BLOCK
81	004574	104203	001400		MOV	#HSLIM-1,R3	:POINTER TO DUMMY SDI BLOCK
82	004576	100673	000006		MOV	R3,RW.DUM(R0)	:STORE IT IN READ BLOCK
83	004600	104207	101773		READ1: MOV	#<BIT15!RDBLK>,R0	:MAKE SURE POINTING AT BLOCK
84	004602	104203	100000		MOV	#RDCMD,R3	:RESET STATUS POINTER
85	004604	100673	000000		MOV	R3,RW.STAT(R0)	:STORE IT BACK
86	004606	060002			XFC	READ	:READ 1 SECTOR
87	004607	115001			TST	R1	:ANY ERRORS ?
88	004610				BNE	RRERR	:YES - UH OH
89	004612	104307	001374		MOV	RDBLK+RW.ER1,R0	:GET ECC STATUS WORD
90	004614	102207	010000		BIT	#ECCF,R0	:ANY ECC ERROR ?
91	004616				BNE	RRERR	:YUP - MARK AS BAD FOR NOW
92	004620	104207	002124		MOV	#NUM,R0	:POINT TO COMPARE BLOCK
93	004622	060006			XFC	CPDAT	:DO DATA COMPARE
94	004623	115001			TST	R1	:ANY ERROR IN COMPARE ?
95	004624				BEQ	NOERR	:NOPE - CONTINUE LOOP
96	004626				RRERR:		
97	004626	103200	010000	001374	BIC	#ECCF,RDBLK+RW.ER1	:CLEAR ECC ERROR BIT
98	004631	104654	000003		MOV	RB.IM(R5),R4	:GET POINTER TO IMAGE
99	004633	102204	100000		BIT	#BD,R4	:ALREADY MARKED BAD ??
100	004635				BNE	NOERR	:YUP - DON'T COUNT AGAIN
101	004637	101204	100000		BIS	#BD,R4	:FLAG AS BAD
102	004641	100654	000003		MOV	R4,RB.IM(R5)	:STORE BACK
103	004643	115400	001711		INC	ERR	:INCREMENT ERROR COUNT
104	004645	115400	002156		INC	RTYCNT	:INC COUNTER
105	004647	105205	000004		NOERR: ADD	#RDLEN,R5	:POINT TO NEXT READ CMD BLOCK
106	004651	117400	002137		DEC	SECCNT	:DECREMENT COUNTER
107	004653				BNE	AGAIN	:NO - DO NEXT SECTOR
108	004655	117400	002140		DEC	N	:DECREMENT COUNTER
109	004657				BNE	CSKIP1	:NO - REPEAT TRACK READ AND COMPARE
110	004661	115000	001711		TST	ERR	:ANY ERRORS ?
111	004663				BEQ	CDONE	:NO - ALL DONE CHECK PASS
112	004665	104204	013022		MOV	#CMDBUF,R4	:POINT TO COMMAND BUFFER
113	004667	104643	000003		HERE: MOV	RB.IM(R4),R3	:GET IMAGE POINTER
114	004671	102203	100000		BIT	#BD,R3	:IS IT BAD

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 50-2
DBN/XBN FORMAT OVERLAY (F1)

115	004673				BEQ	CSKIP7			:NO - SKIP IT
116	004675				PUSH	R4			:SAVE POINTER TO COMMAND BLOCK
117	004676	104644	000003		MOV	RB.IM(R4),R4			:GET POINTER TO IMAGE ENTRY
118	004700	104643	000002		MOV	FT.HI(R4),R3			:GET HIGH ORDER
119	004702	103203	007777		BIC	#LO,R3			:CLEAR OUT ALL BUT HEADER
120	004704	106303	002145		CMP	HD.CUR,R3			:IS IT A 'GOOD X/D BN' ?
121	004706				BEQ	READX			:YES - CHECK IT
122	004710				POP	R4			:ELSE RESTORE COMMAND POINTER
123	004711				BR	CSKIP7			:AND CHECK NEXT ENTRY
124	004713	104302	001412		READX:	MOV	UNIT,R2		:GET PORT NUMBER
125	004715	060012			XFC	SIP			:WAIT FOR PULSE
126	004716	104302	002171		MOV	SECSIZ,R2			:SECTOR SIZE IN WORDS
127	004720	104207	001373		MOV	#RDBLK,R0			:PREPARE FOR READ SECTORS
128	004722	104203	001400		MOV	#HSLIM-1,R3			:POINTER TO DUMMY SDI BLOCK
129	004724	100573	000006		MOV	R3,RW.DUM(R0)			:STORE IN COMMAND BLOCK
130	004726	104643	000001		MOV	FT.LOW(R4),R3			:LO ORDER BLOCK NUMBER
131	004730	100673	000003		MOV	R3,RW.LOW(R0)			:STORE IN READ CMD BLOCK
132	004732	104643	000002		MOV	FT.HI(R4),R3			:HI ORDER BLOCK NUM AND CODE
133	004734	100673	000004		MOV	R3,RW.HI(R0)			:STORE IN READ CMD BLOCK
134	004736	104203	010000		MOV	#RDBUF,R3			:LOAD ADDRESS OF DATA BUFFER
135	004740	100673	000002		MOV	R3,RW.BUF(R0)			:STORE IN COMMAND BUFFER
136	004742	104203	013400		MOV	#RWCMD,R3			:LOAD SDI READ COMMAND
137	004744	104301	001565		MOV	CURTRK,R1			:GET CURRENT HEAD NUMBER IN R1
138	004746	101013			BIS	R1,R3			:SET IT IN COMMAND
139	004747	100673	000005		MOV	R3,RW.CMD(R0)			:STORE BACK
140	004751	104207	101373		READ2:	MOV	#<BIT15!RDBLK>,R0		:MAKE SURE POINTING AT BLOCK
141	004753	104203	100000		MOV	#RDCMD,R3			:MARK AS ONLY REQUEST
142	004755	100173			MOV	R3,(R0)			:STORE IN CMD BLOCK
143	004756	104302	002171		MOV	SECSIZ,R2			:SECTOR SIZE IN WORDS
144	004760	060002			XFC	READ			:READ 1 SECTOR
145	004761	115001			TST	R1			:ANY ERROR IN READ ?
146	004762				BNE	ER1			:YES - CONSIDER BAD
147	004764	104673	000001		MOV	RW.ER1(R0),R3			:LOAD ECC ERROR INDICATOR FOR TEST
148	004766	102203	010000		BIT	#ECCF,R3			:ERROR ?
149	004770				BEQ	CSKIP6			:NO - CHECK EDC
150	004772	103200	010000	001374	BIC	#ECCF,RDBLK+RW.ER1			:CLEAR ECC ERROR BIT
151	004775				CALL	ECCCK			:ELSE FIND HOW MANY SYMBOLS IN ERROR
152	004777	115001			TST	R1			:WITHIN BOUNDS ?
153	005000				BMI	ER1			:NOPE - CONSIDER BAD
154	005002	106300	002131	010000	CSKIP6:	CMP	DWRD,RDBUF		:FIRST WORD O.K.??
155	005005				BNE	ER1			:NOPL - BARF
156	005007	104202	010000		MOV	#RDBUF,R2			:POINT TO BUFFER
157	005011	104207	000400		MOV	#SECS16,R0			:SECTOR SIZE IN WORDS
158	005013				CALL	CEDC			:COMPUTE EDC - RETURNED IN R3
159	005015	104205	010000		MOV	#RDBUF,R5			:POINT TO BUFFER
160	005017	106653	000400		CMP	RW.EDC(R5),R3			:EDC O.K. ?
161	005021				BEQ	OK			:NO ERROR
162	005023				ER1:				
163	005023	104643	000002		MOV	FT.HI(R4),R3			:GET HI ORDER BLOCK NUM AND HDR CODE
164	005025	103203	170000		BIC	#HD.CLR,R3			:CLEAR THE HEADER
165	005027	101203	110000		BIS	#HD.BAD,R3			:MARK AS BAD
166	005031	100643	000002		MOV	R3,FT.HI(R4)			:STORE BACK IN IMAGE
167	005033	115400	001704		INC	ERFLAG			:SET RE-FORMAT FLAG
168	005035	115400	002136		INC	ERRCNT			:UP COUNTER OF BAD BLOCKS
169	005037				BR	CSKIP3			:NO NEED TO RE-READ ANY MORE THIS SECTOR
170	005041	117400	002141		OK:	DEC	N1		:DECREMENT COUNTER - DONE ?
171	005043				BNE	READX			:NO - RE-READ SECTOR IN ERROR

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 50-3
 DBN/XBN FORMAT OVERLAY (F1)

172	005045	104300	002142	002141	CSKIP3:	MOV	NN1,N1	:GET SAVED VALUE
173	005050	117400	001711			DEC	ERR	:DECREMENT IT
174	005052					POP	R4	:RESTORE COMMAND POINTER
175	005053	105204	000004		CSKIP7:	ADD	#RDLEN,R4	:POINT TO NEXT ENTRY
176	005055	115000	001711			TST	ERR	:ALL DONE ERROR SECTORS
177	005057					BNE	HERE	:NO - DO NEXT SECTOR
178	005061	115000	001704			TST	ERFLAG	:WERE THERE ANY BAD SECTORS FOUND
179	005063					BEQ	CDONE	:NOPE - ALL DONE
180	005065	104304	002006		OVER2:	MOV	DPREA,R4	:DATA PREAMBLE LENGTH
181	005067	104303	002005			MOV	HPREA,R3	:HEADER PREAMBLE LENGTH
182	005071	104307	002004			MOV	IMSTAR,R0	:POINT TO TRACK IMAGE START POINT
183	005073	104301	001565			MOV	CURTRK,R1	:CURRENT TRACK NUMBER
184	005075	104302	002171			MOV	SECSIZ,R2	:SECTOR SIZE IN WORDS
185	005077	104205	015763			MOV	#IMAGE,R5	:RECIRCULATION ADDRESS
186	005101	060001				XFC	FORMAT	:RE-FORMAT
187	005102	115001				TST	R1	:ANY PROBLEMS ??
188	005103					BEQ	OVER1	:NO -REDO CHECK PASS
189	005105	115400	001371			INC	UN.ERI	:INCREMENT IT
190	005107	106300	002165	001371		CMP	RETRY,UN.ERI	:DONE ALL RETRIES ?
191	005112					BMI	FERR	:YUP - ERROR
192	005114					CALL	INITPT	:REINIT
193	005116					CALL	CLEAR	:DRIVE CLEAR
194	005120					CALL	SEEK	:RE-SEEK AND GROUP SELECT
195	005122					BR	OVER2	:RETRY FORMAT
196	005124	114000	001371		OVER1:	CLR	UN.ERI	:CLEAR RETRY COUNT
197	005126					BR	DXCH	:RE-CYCLE CHECK PASS
198	005130				CDONE:	POP	R0	:RESTORE CHARACTERISTICS PTR
199	005131					RETURN		

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 51
 DBN/XBN FORMAT OVERLAY (F1)

```

1
2
3
4
5
6 005133 104300 001606 002137 FIXIT: MOV SECTRK,SECCNT ;INIT COUNTER
7 005136 104207 013022 ;MOV #CMDBUF,R0 ;COMMAND BUFFER
8 005140 104205 015763 ;MOV #IMAGE,R5 ;POINT TO TRACK IMAGE
9 005142 104303 001716 ;MOV SKPCNT,R3 ;GET STARTING OFFSET(TUNED)
10 005144 105035 ;ADD R3,R5 ;POINT TO FIRST ENTRY
11 005145 104050 001715 ;MOV R5,STARIT ;MARK STARTING ADDRESS
12 005147 104653 000002 MORE: MOV 2(R5),R3 ;SET UP FOR HSR CODE COMPARE
13 005151 103203 007777 ;BIC #LO,R3 ;ISOLATE HI 4 BITS(HDR CODE)
14 005153 106203 120000 ;CMP #HD.XBN,R3 ;GOOD XBN ?
15 005155 ;BEQ FKIP2 ;YES - MARK AS GOOD TO CHECK
16 005157 106203 140000 ;CMP #HD.DBN,R3 ;GOOD DBN ?
17 005161 ;BEQ FKIP2 ;YES - MARK AS GOOD TO CHECK
18 005163 114003 ;CLR R3 ;CLEAR FOR STORE
19 005164 100673 000002 ;MOV R3,RB.CMD(R0) ;STORE AS BAD SECTOR FLAG
20 005166 105207 000004 ;ADD #RDLEN,R0 ;POINT PAST BLOCK
21 005170 ;BR FKIP1 ;SKIP GOOD MARK
22 005172 104653 000001 FKIP2: MOV 1(R5),R3 ;LO ORDER BLOCK NUMBER
23 005174 100273 ;MOV R3,(R0)+ ;STORE IN READ CMD BLOCK
24 005175 104653 000002 ;MOV 2(R5),R3 ;HI ORDER BLOCK NUM AND CODE
25 005177 100273 ;MOV R3,(R0)+ ;STORE IN READ CMD BLOCK
26 005200 104203 013400 ;MOV #RWCMD,R3 ;LOAD SDI READ COMMAND
27 005202 101303 001565 ;BIS CURTRK,R3 ;SET IN CURRENT TRACK NUMBER
28 005204 100273 ;MOV R3,(R0)+ ;STORE IN BLOCK
29 005205 100275 ;MOV R5,(R0)+ ;SAVE PTR TO IMAGE BLK ENTRY
30 005206 105305 001717 FKIP1: ADD TBLK,R5 ;ADD TO GET NEXT SECTOR
31 005210 106305 001714 ;CMP EIMAGE,R5 ;SEE IF HAVE TO LOOP BACK TO TOP
32 005212 ;BEQ REDO ;NEED TO RESET
33 005214 ;BPL FKP1 ;NO NEED - JUST CONTINUE
34 005216 107305 001714 ;SUB EIMAGE,R5 ;SUBTRACT TO GET LOOP AMOUNT
35 005220 105205 015763 ;ADD #IMAGE,R5 ;AND ADD OFFSET
36 005222 ;BR FKP1 ;SKIP ZERO CONDITION
37 005224 104205 015763 REDO: MOV #IMAGE,R5 ;ZERO SIMPLY MOVE TO FRONT
38 005226 106305 001715 FKP1: CMP STARIT,R5 ;AT BEGINNING ADDRESS ?
39 005230 ;BNE FKIP10 ;NO - JUST CONTINUE
40 005232 105205 000003 ;ADD #IMLEN,R5 ;ELSE POINT TO NEXT ENTRY
41 005234 104050 001715 ;MOV R5,STARIT ;MAKE IT NEW STARTING ADDRESS
42 005236 117400 002137 FKIP10: DEC SECCNT ;DECREMENT
43 005240 ;BNE MORE ;NO - DO NEXT SECTOR
44 005242 ;RETURN

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 52
DBN/XBN TRACK FORMAT OVERLAY (G7)

```

1
2
3
4
5
6
7
8
9
10 005244
11 004014 104200 000047 001636
12 004017 104300 001606 002137
13 004022 104205 015763
14 004024 102200 000600 001702
15 004027
16 004031 102200 000010 001702
17 004034
18 004036
19 004040
20 004042
21 004044 104203 011132
22 004046 100253
23 004047 104303 001566
24 004051 100253
25 004052 104303 001567
26 004054 103203 170000
27 004056 104301 002145
28 004060 101013
29 004061 100253
30 004062
31 004070 117400 002137
32 004072
33 004074 104203 001525
34 004076 104632 000011
35 004100 110702
36 004101 103202 177400
37 004103 104204 015763
38 004105 115002
39 004106
40 004110 115000 002146
41 004112
42 004114 104020 001410
43 004116 114000 001411
44 004120 104300 002146 001403
45 004123 114000 001404
46 004125 104203 001410
47 004127 104204 001403
48 004131
49 004133 106300 001606 001403
50 004136
51 004140 107300 001606 001403
52 004143
53 004145 104200 000003 001410
54 004150 114000 001411
55 004152 104203 001410
56 004154
57 004156 104143

```

.SBTTL DBN/XBN TRACK FORMAT OVERLAY (G7)

SET UP TRACK FORMAT

```

DMOVLY G7, START
MOV #G7, CUROVL
MOV SECTRK, SECCNT
MOV #IMAGE, R5
BIT #MANU+DLL, FLAG
BEQ TKIP2
BIT #DBN, FLAG
BEQ TKIP1
CALL DPBN
BR TKIP2
TKIP1: CALL
TKIP2: MOV #GDBLK, R3
MOV R3, (R5)+
MOV CURBN, R3
MOV R3, (R5)+
MOV CURBN+1, R3
BIC #HD.CLR, R3
MOV HD.CUR, R1
BIS R1, R3
MOV R3, (R5)+
DUBINC CURBN
SKIP5: DEC SECCNT
BNE TKIP2
MOV #SCR, R3
MOV OFFS12(R3), R2
SWAB R2
BIC #HIBYTE, R2
MOV #IMAGE, R4
TST R2
BEQ TKIP5
TST CURGRP
BEQ TKIP5
MOV R2, TEMP
CLR TEMP+1
MOV CURGRP, DDUMMY
CLR DDUMMY+1
MOV #TEMP, R3
MOV #DDUMMY, R4
CALL DMUL
TKIP8: CMP SECTRK, DDUMMY
BPL TKIP9
SUB SECTRK, DDUMMY
BR TKIP8
TKIP9: MOV #IMLEN, TEMP
CLR TEMP+1
MOV #TEMP, R3
CALL DMUL
MOV (R4), R3

```

```

:GET OVERLAY INDICATOR
:MOVE SECTOR COUNT INTO R3
:POINT TO FORMAT TRACK IMAGE
:SEE IF FCT AVAILABLE
:NO - SKIP PBN COMPUTATION
:DO DBN AREA ??
:NO - DO XBN AREA
:COMPUTE PBN FOR STARTING DBN
:SKIP XBN COMPUTATION
:COMPUTE PBN FOR STARTING XBN
:POINT R3 AT GOOD DATA BLOCK
:AND STORE PTR IN IMAGE BLOCK
:GET LOW ORDER BLOCK NUMBER
:AND STORE IN IMAGE BLOCK
:HI ORDER BLOCK NUM AND HDR CODE
:CLEAR HEADER CODE
:GET CURRENT HEADER CODE (XBN OR DBN)
:SET TO GOOD HEADER CODE
:AND STORE IN IMAGE BLOCK
:INCREMENT IT
:DECREMENT IT
:NO - NO NEXT SECTOR
:POINT TO CHARACTERISTICS
:GET GROUP OFFSET
:GET INTO LOWBYTE
:CLEAR HIGH GARBAGE
:POINT TO IMAGE
:ANY OFFSET ?
:NO - SKIP CALCULATIONS
:IS GROUP ZERO ???
:YES - NO OFFSET
:STORE IT
:FOR STORE
:GET CURRENT GROUP
:CLEAR HIGH WORD
:FOR MUL
:DITTO
:MULTIPLY TO GET OFFSET FOR THIS GROUP
:IS TOTAL OFFSET MORE THAN NUMBER OF SECTORS ?
:NO - ALL IS FINE
:YES - SUBTRACT TILL IT IS
:CHECK AGAIN
:GET LENGTH OF IMAGE BLOCK
:FOR STORE
:FOR MULT
:GET LENGTH TO OFFSET
:GET RESULT

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 52-1
 OBN/XBN TRACK FORMAT OVERLAY (G7)

58	004157	104304	001714			MOV	EIMAGE,R4		:GET ADDRESS OF END OF IMAGE
59	004161	107034				SUB	R3,R4		:SUBTRACT TO GET STARTING LOCATION
60	004162	104040	002004		TKIP5:	MOV	R4,IMSTAR		:STORE IT
61	004164	102200	002002	001702		BIT	#BSTGS+FCTEMT,FLAG		:IS FCT AVAILABLE AND NON-EMPTY ?
62	004167					BNE	TKIP3		:NO - CONSIDER GOOD
63	004171	104300	001606	002137		MOV	SECTRK,SECCNT		:RESET SECTOR COUNT
64	004174	104302	001706		TKIP12:	MOV	BADPBN,R2		:POINT TO PBN
65	004176	104121				MOV	(R2),R1		:GET LOW ORDER PBN
66	004177	106010	001563			CMP	R1,CURPBN		:ARE THEY EQUAL ?
67	004201					BNE	TKIP4		:NO - SKIP REST OF COMPARE
68	004203	104621	000001			MOV	1(R2),R1		:GET HIGH ORDER BAD
69	004205	103201	170000			BIC	#HD.CLR,R1		:CLEAR HEADER FOR COMPARE
70	004207	106010	001564			CMP	R1,CURPBN+1		:EQUAL ?
71	004211					BNE	TKIP4		:NO - MARK AS GOOD
72	004213	117400	002164			DEC	COUNT		:DECREMENT IT
73	004215	117400	001722			DEC	FCNT		:DEC IT
74	004217					BNE	TKIP7		:IF NOT EMPTY THEN CONTINUE
75	004221	101200	000002	001702		BIS	#FCTEMT,FLAG		:SET FCT EMPTY FLAG
76	004224	104643	000002		TKIP7:	MOV	FT.HI(R4),R3		:HI ORDER BLOCK NUM AND HDR CODE
77	004226	103203	170000			BIC	#HD.CLR,R3		:CLEAR THE HEADER CODE
78	004230	101203	110000			BIS	#HD.BAD,R3		:SET TO BAD HEADER CODE
79	004232	100643	000002			MOV	R3,FT.HI(R4)		:AND STORE IN IMAGE BLOCK
80	004234	105200	000002	001706		ADD	#2,BADPBN		:MOVE PTR TO NEXT BAD BLOCK
81	004237	115000	002164			TST	COUNT		:DONE WITH THIS FCT BLOCK ?
82	004241					BNE	TKIP4		:IF NOT DONE SKIP
83	004243	115400	001743			INC	FCTCNT		:GET NEXT BLOCK NUMBER
84	004245					CALL	DXFCP1		:ELSE PAGE IN NEW FCT BLOCK
85	004247	105204	000003		TKIP4:	ADD	#IMLEN,R4		:POINT TO NEXT IMAGE ENTRY
86	004251	106304	001714			CMP	EIMAGE,R4		:AT THE END ?
87	004253					BNE	TKIP11		:NOPE - CARRY ON
88	004255	104204	015763			MOV	#IMAGE,R4		:POINT TO START
89	004257				TKIP11:	DUBINC	CURPBN		:INCREMENT CURRENT PBN COUNTER
90	004265	117400	002137			DEC	SECCNT		:DECREMENT SECTOR COUNTER
91	004267					BNE	TKIP12		:CONTINUE TILL DONE ALL SECTORS
92									
93	004271	104304	002004		TKIP3:	MOV	IMSTAR,R4		:POINT TO FIRST TO FORMAT ENTRY
94	004273	104303	001714			MOV	EIMAGE,R3		:GET END ADDRESS
95	004275	107203	000003			SUB	#IMLEN,R3		:POINT TO FLAG OF LAST ENTRY
96	004277	106043				CMP	R4,R3		:FIRST = LAST ?
97	004300					BEQ	TKIP14		:NO - SKIP SPECIAL STUFF
98	004302	104135				MOV	(R3),R5		:GET FLAG WORD
99	004303	101205	040000			BIS	#RECIR,R5		:SET RECIRCULATION FLAG
100	004305	100135				MOV	R5,(R3)		:STORE IT BACK
101	004306					BR	TKIP13		:SKIP KLUDGE FIX TO UDA
102	004310	101200	040000	002004	TKIP14:	BIS	#RECIR,IMSTAR		:SET BIT IN POINTER
103	004313	104143			TKIP13:	MOV	(R4),R3		:GET BUFF POINTER
104	004314	101203	100000			BIS	#LAST,R3		:SIGNAL AS LAST
105	004316	100143				MOV	R3,(R4)		:STORE IT BACK
106	004317					RETURN			

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 53
 DBN/XBN TRACK FORMAT OVERLAY (G7)

1									
2									
3									
4									
5									
6									
7									
8	004321	104300	001622	001410	DPBN:	MOV	XBNSEC,TEMP	:	GET NUMBER OF SECTORS IN XBN AREA
9	004324	104300	001623	001411		MOV	XBNSEC+1,TEMP+1	:	GET HI ORDER
10	004327	104204	001410			MOV	#TEMP,R4	:	POINT R4 AT TEMP AREA
11	004331	104203	001616			MOV	#LBNLBN,R3	:	POINT AT NUM OF LBN'S IN LBN AREA
12	004333					CALL	DADD	:	ADD
13	004335	104203	001620			MOV	#RBNLBN,R3	:	POINT TO NUM OF RBN'S IN LBN AREA
14	004337					CALL	DADD	:	ADD TO GET SECTORS IN LBN + XBN AREA
15	004341	104203	001566			MOV	#CURBN,R3	:	POINT TO CURRENT BLOCK NUMBER(DBN)
16	004343					CALL	DADD	:	GET RELATIVE PBN
17	004345	104641	000001			MOV	1(R4),R1	:	GET HIGH ORDER
18	004347	107301	002012			SUB	ST.DBN,R1	:	SUBTRACT HIGH ORDER STARTING DBN
19	004351	104140	001563			MOV	(R4),CURPBN	:	GET LO ORDER PBN
20	004353	104010	001564			MOV	R1,CURPBN+1	:	STORE HIGH ORDER
21	004355					RETURN			
22									
23									
24									
25									
26									
27									
28									
29									
30									
31	004357	104300	001616	001410	XPBN:	MOV	LBNLBN,TEMP	:	GET NUMBER OF LBN'S IN LBN AREA
32	004362	104300	001617	001411		MOV	LBNLBN+1,TEMP+1	:	GET HIGH ORDER
33	004365	104204	001410			MOV	#TEMP,R4	:	POINT R4 TO TEMP AREA
34	004367	104203	001620			MOV	#RBNLBN,R3	:	POINT R3 AT RBN'S IN LBN AREA
35	004371					CALL	DADD	:	ADD TO GET TOTAL SECTORS IN LBN AREA
36	004373	104203	001566			MOV	#CURBN,R3	:	POINT R3 AT CURRENT BLOCK NUMBER
37	004375					CALL	DADD	:	ADD TO GET RELATIVE PBN
38	004377	104641	000001			MOV	1(R4),R1	:	GET HIGH ORDER
39	004401	107301	002011			SUB	ST.XBN,R1	:	SUBTRACT HIGH ORDER STARTING XBN
40	004403	104140	001563			MOV	(R4),CURPBN	:	GET LO ORDER OF PBN
41	004405	104010	001564			MOV	R1,CURPBN+1	:	SAVE HIGH ORDER
42	004407					RETURN			

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 55
LBN FORMATTING OVERLAY (F2)

```

1          .SBTTL LBN FORMATTING OVERLAY (F2)
2
3          :
4          :
5 004441   :
6          :
7          :
8 004014   104200 000003 001636 LFORM: MOV    #F2,CUROVL      ;OVERLAY #2
9 004017   102200 000001 001702      BIT    #FCTAVL,FLAG    ;FCT AVAILAABLE ?
10 004022      BEQ    XSKIP1      ;NO - SKIP SET UP
11 004024   104201 000036      MOV    #G3,R1         ;OVERLAY TO GET RIGHT FCT BLOCK
12 004026      CALL   PAGE          ;EXECUTE IT
13 004030   104207 001525      MOV    #SCR,R0        ;POINT TO CHARACTERISTICS BLOCK
14 004032   104300 001626 001604 XSKIP1: MOV    LBNCYL,CYLNUM    ;GET LO ORDER CYLINDER COUNT
15 004035   104300 001626 002143      MOV    LBNCYL,CNTCYL  ;MAKE LO ORDER COUNT
16 004040   104300 001627 001605      MOV    LBNCYL+1,CYLNUM+1 ;GET HIGH ORDER
17 004043   104300 001627 002144      MOV    LBNCYL+1,CNTCYL+1 ;STORE IT
18 004046   103200 170000 002144      BIC    #HD.CLR,CNTCYL+1 ;CLEAR STARTING CYLINDER BITS
19 004051   104204 001604      MOV    #CYLNUM,R4     ;SUBTRACT TO GET CYLINDER NUMBER
20 004053   104203 002151      MOV    #ONE,R3        ;1 - BECAUSE START AT 0
21 004055      CALL   DSUB           ;DO SUBTRACT
22 004057   104300 002160 001720      MOV    TOTRCT,RCTTOT  ;GET TOTAL RCT LBN'S
23 004062   104201 000052      MOV    #G8,R1         ;POINT TO OVERLAY
24 004064      CALL   PAGE          ;COMPUTE VARIOUS CONSTANTS
25 004066   104207 001525      MOV    #SCR,R0        ;POINT TO CHARACTERISTICS
26 004070   104673 000002      MOV    GRPCYL(R0),R3  ;GET GROUPS/CYLINDER
27 004072   103203 177400      BIC    #HIBYTE,R3     ;CLEAR OUT GARBAGE
28 004074   104030 002147      MOV    R3,GRPCNT      ;USE AS COUNTER
29 004076   104030 002146      MOV    R3,CURGRP      ;GROUP NUMBER
30 004100   117400 002146      DEC    CURGRP         ;DECREMENT TO GET ACTUAL NUMBER
31 004102   104300 001604 001551 XSLEK2: MOV    CYLNUM,ISEEK+1    ;GET CURRENT CYLINDER NUMBER
32 004105   104300 001605 001552      MOV    CYLNUM+1,ISEEK+2 ;GET HIGH ORDER
33 004110   104300 002146 001553      MOV    CURGRP,ISEEK+3   ;LOAD GROUP NUMBER
34 004113   102200 020000 001703      BIT    #MODE,FLAG1     ;WHAT MODE
35 004116      BNE    1$             ;IF SET THEN 576
36 004120   103200 100000 001552      BIC    #SS,ISEEK+2     ;SET 512 MODE SEEK
37 004123      BR     2$             ;SKIP 576 SETUP
38 004125   101200 100000 001552 1$:  BIS    #SS,ISEEK+2     ;SET 576 MODE SEEK
39 004130      2$:  CALL   SEEK          ;DO THE SEEK
40 004132   115001      TST    R1             ;ANY ERROR ?
41 004133      BMI    SEEKER       ;YUP - CUT OUT
42 004135   104207 001525      MOV    #SCR,R0        ;POINT TO CHARACTERISTICS
43 004137   104673 000003      MOV    TRKGRP(R0),R3  ;GET TRACKS/GROUP
44 004141   103203 177400      BIC    #HIBYTE,R3     ;CLEAR OUT GARBAGE
45 004143   104030 002150      MOV    R3,TRKCNT      ;MAKE COUNTER
46 004145   117403      DEC    R3             ;WANT LAST TRACK NUMBER
47 004146   104030 001565      MOV    R3,CURTRK      ;MAKE CURRENT TRACK 0
48 004150   104201 000025      XSKIP3: MOV    #F8,R1   ;TRACK SET UP OVERLAY
49 004152      CALL   PAGE          ;SET UP TRACK FORMAT
50 004154   104304 002006      XSKIP2: MOV    DPREA,R4  ;GET DATA PREAMBLE LENGTH
51 004156   104303 002005      MOV    HPREA,R3       ;GET HEADER PREAMBLE LENGTH
52 004160   104307 002004      MOV    IMSTAR,R0      ;POINT TO TRACK IMAGE START POINT
53 004162   104301 001565      MOV    CURTRK,R1     ;TRACK TO FORMAT
54 004164   104302 002171      MOV    SECSIZ,R2     ;SECTOR SIZE IN WORDS
55 004166   104205 015763      MOV    #IMAGE,R5     ;RECIRCULATION ADDRESS
56 004170   060001      XFC    FORMAT        ;DO FORMAT
57 004171   115001      TST    R1            ;ANY ERROR ?

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 55-1
LBN FORMATTING OVERLAY (F2)

58	004172				BEQ	LSKIP4		:NO - DO CHECK PASS
59	004174	115400	001371		INC	UN.ERI		:INCREMENT IT
60	004176	106300	002165	001371	CMP	RETRY,UN.ERI		:DONE ALL RETRIES ?
61	004201				BMI	FORERR		:YUP - ERROR
62	004203				CALL	INITPT		:REINIT
63	004205				CALL	CLEAR		:DRIVE CLEAR
64	004207				CALL	SEEK		:RE-SEEK AND GROUP SELECT
65	004211				BR	XSKIP2		:NOPE - RETRY
66	004213	114000	001371		LSKIP4: CLR	UN.ERI		:FOR STORE
67	004215	104301	001710		MOV	EMAX,R1		:GET MAX REVECTORS
68	004217	107301	001741		SUB	REVCNT,R1		:SUBTRACT CURRENT ENTRIES
69	004221	106301	001606		CMP	SECTRK,R1		:ENOUGH LEFT FOR WHOLE TRACK ??
70	004223				BMI	XSKIP4		:YES - CONTINUE
71	004225	104201	000011		MOV	#F4,R1		:SIGNAL RCT UPDATE OVERLAY
72	004227				CALL	PAGE		:PAGE IT IN
73	004231	114000	001741		CLR	REVCNT		:FOR STORE
74	004233	104300	001707	001737	MOV	ERRBUF,ERPNT		:FOR RESET
75	004236				XSKIP4: CALL	LCHEC		:DO CHECK PASS
76	004240	102200	020000	001702	BIT	#INIRCT,FLAG		:TIME TO INIT RCT ?
77	004243				BEQ	XSKIP5		:NOPE
78	004245	101200	000100	001702	BIS	#REVECT,FLAG		:SET REVECTOR ON
79	004250	103200	020000	001702	BIC	#INIRCT,FLAG		:REST FLAG
80	004253	102200	002000	001702	BIT	#BSTGS,FLAG		:DOING BEST GUESS ?
81	004256				BEQ	XSKIP6		:NO - GO ALL THE WAY
82	004260	104201	000022		MOV	#F7,R1		:RCT INIT OVERLAY
83	004262				CALL	PAGE		:EXECUTE IT
84	004264				BR	XSKIP5		:SKIP OTHER
85	004266	104201	000014		XSKIP6: MOV	#F5,R1		:DO FCT->RCT AND INIT
86	004270				CALL	PAGE		:EXECUTE IT
87	004272	117400	001565		XSKIP5: DEC	CURTRK		:DECREMENT IT
88	004274	104204	001576		MOV	#HOLDBN,R4		:GET STARTING BLOCK NUMBER
89	004276	104207	001525		MOV	#SCR,R0		:POINT TO CHARACTERISTICS
90	004300	102200	020000	001703	BIT	#MODE,FLAG1		:WHAT MODE
91	004303				BNE	L576		:IF SET THEN 576
92	004305	104673	000011		MOV	LBNT12(R0),R3		:GET LBN/TRACK FOR 512
93	004307				BR	XSKIP8		:SKIP 576 SETUP
94	004311	104673	000015		L576: MOV	LBNT76(R0),R3		:GET LBN/TRACK FOR 576
95	004313	103203	177400		XSKIP8: BIC	#HIBYTE,R3		:CLEAR HIGH BYTE
96	004315	104030	001403		MOV	R3,DDUMMY		:STORE IT
97	004317	114000	001404		CLR	DDUMMY+1		:FOR STORE
98	004321	104203	001403		MOV	#DDUMMY,R3		:LBN/TRACK
99	004323				CALL	DSUB		:GET STARTING LBN FOR NEW TRACK
100	004325	104300	001576	001566	MOV	HOLDBN,CURBN		:GET LOW ORDER
101	004330	104300	001577	001567	MOV	HOLDBN+1,CURBN+1		:GET HIGH ORDER
102	004333	104204	001600		MOV	#HOLRBN,R4		:GET STARTING RBN NUMBER
103	004335	104673	000004		MOV	RBNTRK(R0),R3		:GET RBN/TRACK
104	004337	103203	177600		BIC	#HI1BYTE,R3		:CLERA OUT GARBAGE
105	004341	104030	001403		MOV	R3,DDUMMY		:STORE IT
106	004343	114000	001404		CLR	DDUMMY+1		:FOR STORE
107	004345	104203	001403		MOV	#DDUMMY,R3		:RBN'S/TRACK
108	004347				CALL	DSUB		:GET STARTING RBN FOR NEW TRACK
109	004351	104300	001600	001561	MOV	HOLRBN,CURRBN		:GET LOW ORDER
110	004354	104300	001601	001562	MOV	HOLRBN+1,CURRBN+1		:GET HI ORDER
111	004357	117400	002150		DEC	TRKCNT		:DECREMENT IT
112	004361				BNE	XSKIP3		:NO - DO NEXT TRACK
113	004363	117400	002146		DEC	CURGRP		:DECREMENT GROUP NUMBER
114	004365	117400	002147		DEC	GRPCNT		:DECREMENT IT

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 55-2
LBN FORMATTING OVERLAY (F2)

115	004367			BNE	XSLEK2		:NO - DO NEXT GROUP
116	004371	104204	002143	MOV	#CNTCYL,R4		:GET READY TO DEC CYLINDER CNT
117	004373	104203	002151	MOV	#ONE,R3		:CONSTANT WORD OF 1
118	004375			CALL	DSUB		:DECREMENT IT
119	004377	060022		XFC	UPDATE		:UPDATE PROGRESS INDICATOR
120	004400	104204	001604	MOV	#CYLNUM,R4		:GET CURRENT CYLINDER NUMBER
121	004402	104203	002151	MOV	#ONE,R3		:FOR DECREMENT
122	004404			CALL	DSUB		:DECREMENT FOR NEW CYLINDER NUM
123	004406	104304	002143	MOV	CNTCYL,R4		:LOW ORDER ZERO ?
124	004410			BNE	XSLEEK		:NO - CONTINUE
125	004412	104304	002144	MOV	CNTCYL+1,R4		:HIGH ORDER ZERO ?
126	004414			BNE	XSLEEK		:NO - CONTINUE
127	004416	104303	001741	MOV	REVCNT,R3		:ANY LEFTOVER REVECTORS ?
128	004420			BEQ	XDONE		:NOPE
129	004422	101200	040000	BIS	#FINI,FLAG	001702	:SIGNAL NOT TO SEEK
130	004425	104201	000011	MOV	#F4,R1		:SIGNAL RCT UPDATE OVERLAY
131	004427			CALL	PAGE		:UPDATE IT
132	004431	104303	001777	XDONE: MOV	LBNBAD,R3		:GET BAD BLOCKS FROM FCT
133	004433	104304	002136	MOV	ERRCNT,R4		:GET CHECK PASS BAD
134	004435	105043		ADD	R4,R3		:ADD TO GET TOTAL
135	004436	104030	001777	MOV	R3,LBNBAD		:STORE BACK
136	004440	104201	000041	MOV	#G4,R1		:RCT CLEANUP
137	004442			CALL	NEXT		:GET NEXT OVERLAY
138	004444			RETURN			
139	004446	104012		FORERR: MOV	R1,R2		:XFC ERROR CODE
140	004447	104201	000014	MOV	#12.,R1		:SIGNAL LBN FORMAT ERROR
141	004451			BR	LFERR		
142	004453	104302	001604	SEEKER: MOV	CYLNUM,R2		:CYLINDER FAILED ON
143	004455	104201	000012	MOV	#10.,R1		:SEEK ERROR
144	004457			LFERR: CALL	ERRMNT		:ERROR RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 56
LBN FORMATTING OVERLAY (F2)

```

1
2
3
4
5
6 004461 114000 001711          LCHEC: CLR      ERR          :FOR ERROR COUNT RESET
7 004463 114000 001704          CLR      ERFLAG         :CLEAR REFORMAT FLAG
8 004465 102200 002000 001702  BIT      #BSTGS,FLAG     :BEST GUESS ?
9 004470          BNE      LSKIP         :YES - DO EXTENSIVE READ
10 004472 104200 000001 002140  MOV      #1,N           :SET UP FOR STORE
11 004475 104200 000005 002141  MOV      #5,N1          :SET UP
12 004500 104300 002141 002142  MOV      N1,NN1         :SAVE FOR LATER RESET
13 004503          BR       LSKIP2        :SKIP EXTENSIVE READ SETUP
14 004505 104200 000003 002140  LSKIP:  MOV      #3,N           :EXTENSIVE REGULAR READ
15 004510 104200 000024 002141  MOV      #20.,N1        :EXTENSIVE ERROR READS
16 004513 104300 002141 002142  MOV      N1,NN1         :SAVE FOR LATER RESET
17 004516          LSKIP2: CALL   LFIXIT        :EXECUTE IT
18 004520 104302 001412          LSKIP1: MOV      UNIT,R2       :GET PORT NUMBER
19 004522 060012          XFC      SIP            :WAIT FOR PULSE
20 004523 104302 002171          MOV      SECSIZ,R2      :SECTOR SIZE IN WORDS
21 004525 104300 001606 002137  MOV      SECTRK,SECCNT  :LOAD SECTORS/TRACK
22 004530 104205 013022          MOV      #CMDBUF,R5     :POINT TO COMMAND BUFFER
23 004532 104207 001373          LAGAIN: MOV      #RDBLK,R0    :POINT TO READ COMMAND BLOCK
24 004534 104653 000002          MOV      RB.CMD(R5),R3  :READ COMMAND ZERO ?
25 004536          BEQ      LNOERR        :SECTOR BAD - SKIP CHECKS
26 004540 100673 000005          MOV      R3,RW.CMD(R0)  :ELSE STORE IN COMMAND BLOCK
27 004542 104653 000000          MOV      RB.LOW(R5),R3  :LOAD LOW ORDER SECTOR NUMBER
28 004544 100673 000003          MOV      R3,RW.LOW(R0)  :STORE IN COMMAND BLOCK
29 004546 104653 000001          MOV      RB.HI(R5),R3   :LOAD HIGH ORDER BLOCK NUMBER
30 004550 100673 000004          MOV      R3,RW.HI(R0)   :STORE IN COMMAND BLOCK
31 004552 104203 010000          MOV      #RDBUF,R3     :GET BUFFER POINTER
32 004554 100673 000002          MOV      R3,RW.BUF(R0)  :STORE IN COMMAND BLOCK
33 004556 104203 001400          MOV      #HSLIM-1,R3    :POINTER TO DUMMY SDI BLOCK
34 004560 100673 000006          MOV      R3,RW.DUM(R0)  :STORE IN READ BLOCK
35 004562 104207 101373          READ3:  MOV      #<BIT15!RDBLK>,R0 :MAKE SURE POINTING AT BLOCK
36 004564 104203 100000          MOV      #RDCMD,R3     :RESET STATUS POINTER
37 004566 100673 000000          MOV      R3,RW.STAT(R0) :STORE IT BACK
38 004570 060002          XFC      READ          :READ 1 SECTOR
39 004571 115001          TST      R1            :ANY ERRORS ?
40 004572          BNE      LERR          :YES - UM OH
41 004574 104307 001374          MOV      RDBLK+RW.ER1,R0 :GET ECC STATUS WORD
42 004576 102207 010000          BIT      #ECCF,R0      :ECC ERROR ?
43 004600          BNE      LERR          :YES - MARK AS BAD FOR NOW
44 004602 104207 002124          MOV      #NUM,R0       :POINT TO COMPARE BLOCK
45 004604 060006          XFC      CMPDAT        :DO DATA COMPARE
46 004605 115001          TST      R1            :ANY ERROR IN COMPARE ?
47 004606          BEQ      LNOERR        :NOPE - CONTINUE LOOP
48 004610          LERR:
49 004610 103200 010000 001374  BIC      #ECCF,RDBLK+RW.ER1 :CLEAR ECC ERROR BIT
50 004613 104654 000003          MOV      RB.IM(R5),R4   :GET IMAGE POINTER
51 004615 102204 100000          BIT      #BD,R4        :ALREADY BEEN HERE ??
52 004617          BNE      LNOERR        :YUP - DON'T COUNT AGAIN
53 004621 101204 100000          BIS      #BD,R4        :MARK AS BAD
54 004623 100654 000003          MOV      R4,RB.IM(R5)  :STORE BACK
55 004625 115400 002156          INC      RTYCNT        :INC IT
56 004627 115400 001711          INC      ERR           :INCREMENT ERROR COUNT
57 004631 105205 000004          LNOERR: ADD     #RDLEN,R5   :POINT TO NEXT READ CMD BLOCK

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 56-1
LBN FORMATTING OVERLA? (F2)

58	004633	117400	002137	DEC	SECCNT	:DECREMENT COUNTER
59	004635			BNE	LAGAIN	:NO - DO NEXT SECTOR
60	004637	117400	002140	DEC	N	:DECREMENT COUNTER
61	004641			BNE	LSKIP1	:NO - REPEAT TRACK READ AND COMPARE
62	004643	115000	001711	TST	ERR	:ANY ERRORS ON FIRST PASS ?
63	004645			BEQ	LDONE	:NO - ALL DONE CHECK PASS
64	004647	104204	013022	MOV	#CMDBUF,R4	:POINT TO COMMAND BUFFER
65	004651	104643	000003	LHERE: MOV	RB.IM(R4),R3	:GET IMAGE POINTER WORD
66	004653	102203	100000	BIT	#BD,R3	:IS IT BAD ?
67	004655			BEQ	LSKIP7	:NO - SKIP IT
68	004657			PUSH	R4	:SAVE COMMAND POINTER
69	004660	104644	000003	MOV	RB.IM(R4),R4	:GET IMAGE POINTER
70	004662	104643	000002	MOV	FT.HI(R4),R3	:GET HI ORDER BLOCK NUM AND HDR CODE
71	004664	103203	007777	BIC	#LO,R3	:CLEAR LOW ORDER
72	004666	106203	000000	CMP	#HD.LBN,R3	:IS IT A 'GOOD' LBN
73	004670			BEQ	LSKIP8	:YES - DO IT
74	004672	106203	060000	CMP	#HD.RBN,R3	:IS IT AN RBN ???
75	004674			BEQ	LSKIP8	:YES - DO IT
76	004676			POP	R4	:ELSE RESTORE COMMAND POINTER
77	004677			BR	LSKIP7	:AND DO NEXT ENTRY
78	004701	104302	001412	LSKIP8: MOV	UNIT,R2	:GET FORT NUMBER
79	004703	060012		XFC	SIP	:WAIT FOR PULSE
80	004704	104302	002171	MOV	SECSIZ,R2	:SECTOR SIZE IN WORDS
81	004706	104207	001373	MOV	#RDBLK,R0	:PREPARE FOR READ SECTORS
82	004710	104203	001400	MOV	#HSLIM-1,R3	:POINTER TO DUMMY SDI BLOCK
83	004712	100673	000006	MOV	R3,RW.DUM(R0)	:STORE IN COMMAND BLOCK
84	004714	104643	000001	MOV	1(R4),R3	:LO ORDER BLOCK NUMBER
85	004716	100673	000003	MOV	R3,RW.LOW(R0)	:STORE IN READ CMD BLOCK
86	004720	104643	000002	MOV	2(R4),R3	:HI ORDER BLOCK NUM AND CODE
87	004722	100673	000004	MOV	R3,RW.HI(R0)	:STORE IN READ CMD BLOCK
88	004724	104203	010000	MOV	#RDBUF,R3	:LOAD ADDRESS OF DATA BUFFER
89	004726	100673	000002	MOV	R3,RW.BUF(R0)	:STORE IN COMMAND BUFFER
90	004730	104203	013400	MOV	#RWCMD,R3	:LOAD SDI READ COMMAND
91	004732	101303	001565	BIS	CURTRK,R3	:SET CURRENT HEAD ADDRESS IN COMMAND
92	004734	100673	000005	MOV	R3,RW.CMD(R0)	:STORE BACK
93	004736	104207	101373	READ4: MOV	#<BIT15!RDBLK>,R0	:MAKE SURE POINTING AT BLOCK
94	004740	104203	100000	MOV	#RDCMD,R3	:MARK AS ONLY REQUEST
95	004742	100173		MOV	R3,(R0)	:STORE IN CMD BLOCK
96	004743	104302	002171	MOV	SECSIZ,R2	:SECTOR SIZE IN WORDS
97	004745	060002		XFC	READ	:READ 1 SECTOR
98	004746	115001		TST	R1	:ANY ERROR IN READ ?
99	004747			BNE	LER3	:YES - CONSIDER BAD
100	004751	104673	000001	MOV	RW.ER1(R0),R3	:LOAD ECC ERROR INDICATOR FOR TEST
101	004753	102203	010000	BIT	#ECCF,R3	:TEST FOR ECC ERROR
102	004755			BEQ	LSKIP6	:NO - CHECK EDC
103	004757	103200	010000 001374	BIC	#ECCF,RDBLK+RW.ER1	:CLEAR ECC ERROR BIT
104	004762			CALL	ECCCK	:FIND OUT HOW MANY SYMBOLS IN ERROR
105	004764	115001		TST	R1	:TOO MANY ?
106	004765			BMI	LER1	:YUP - CONSIDER BAD
107	004767			LSKIP6: MOV		
108	004767	106300	002131 010000	CMP	DWRD,RDBUF	:IS FIRST WORD O.K. ?
109	004772			BNE	LER1	:NOPE - BARF
110	004774	104207	002124	MOV	#NUM,R0	:POINT TO COMPARE BLOCK
111	004776	060006		XFC	CHPDAT	:DO DATA COMPARE
112	004777	115001		TST	R1	:ANY ERROR IN COMPARE ?
113	005000			BEQ	LOK	:NO ERROR
114	005002			BR	LER1	:SKIP BAD HEADER FLAGGING

UDAF52 - LDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 56-2
 .BN FORMATTING OVERLAY (F2)

115	005004	101200	000040	001703	LER3:	BIS	#BDHD,FLAG1	:FLAG AS BAD HEADER
116	005007	102200	000100	001702	LER1:	BIT	#REVECT,FLAG	:IN RCT ???
117	005012					BNE	LER2	:NOPE - SKIP NEXT STUFF
118	005014	115400	002000			INC	RCTBAD	:INC BAD COUNTER
119	005016					BR	BDIRCT	:THEN MARK BAD
120	005020	104643	000002		LER2:	MOV	FT.HI(R4),R3	:GET HI ORDER BLOCK NUM AND HDR CODE
121	005022	103203	007777			BIC	#LO,R3	:CLEAR LOW ORDER
122	005024	106203	060000			CMF	#HD.RBN,R3	:IS IT A BAD RBN ?
123	005026					BEQ	BRBN	:YUP - GO HANDLE IT
124	005030	104643	000002			MOV	FT.HI(R4),R3	:RELOAD HEADER
125	005032	103203	170000			BIC	#HD.CLR,R3	:CLEAR THE HEADER
126	005034	102200	001000	001702		BIT	#PRIM,FLAG	:ANY PRIMARY YET ?
127	005037					BNE	SND	:YUP - THIS ONE SECONDARY
128	005041	104030	001404			MOV	R3,DDUMMY+1	:STORE HIGH ORDER FOR RBN COMPUTATION
129	005043	101200	001000	001702		BIS	#PRIM,FLAG	:SET PRIMARY FLAG
130	005046	101200	000004	001703		BIS	#RPRIM,FLAG1	:SET GOOD RBN EDC NEEDED
131	005051	101203	050000			BIS	#HD.PRIV,R3	:MARK AS PRIMARY
132	005053	100643	000002			MOV	R3,FT.HI(R4)	:STORE BACK IN IMAGE
133	005055	104640	000001	001403		MOV	FT.LOW(R4),DDUMMY	:STORE LOW ORDER FOR RBN COMPUTATION
134	005060	104042				MOV	R4,R2	:SAVE IMAGE POINTER
135	005061	104207	001525			MOV	#SCR,R0	:MAKE SURE POINT TO CHAR BLOCK
136	005063	104204	001403			MOV	#DDUMMY,R4	:POINT TO BLOCK NUMBER
137	005065					CALL	PRIMRB	:GET PRIMARY RBN NUMBER
138	005067	104307	001634			MOV	REVRBN,R0	:GET NUMBER OF REVECTORED RBN
139	005071	104301	001635			MOV	REVRBN+1,R1	:GET HIGH ORDER
140	005073	105301	002010			ADD	ST.RBN,R1	:ADD IN STARTING BITS
141	005075	101201	060000			BIS	#HD.RBN,R1	:SET IN RBN HEADER CODE
142	005077	104205	011607			MOV	#PRMBUF,R5	:USE RDBUF TO HOLD 128 COPIES OF RBN
143	005101	104203	000200			MOV	#RBNRPT,R3	:COUNT OF REPLICATED RBN'S
144	005103	100257			RPT1:	MOV	R0,(R5)+	:STORE A COPY
145	005104	100251				MOV	R1,(R5)+	:AND HIGH ORDER
146	005105	117403				DEC	R3	:DECREMENT COUNTER - DONE ?
147	005106					BNE	RPT1	:NO - STORE ANOTHER COPY
148	005110	104024				MOV	R2,R4	:RESTORE IMAGE POINTER
149	005111	104205	011607			MOV	#PRMBUF,R5	:POINT TO BEGINNING OF BUFFER
150	005113	104642	000000			MOV	FT.BUF(R4),R2	:GET BUFFER POINTER
151	005115	103202	007777			BIC	#BUFMSK,R2	:CLEAR ONLY BUFFER POINTER
152	005117	101052				BIS	R5,R2	:OR IN NEW BUFFER POINTER
153	005120	100642	000000			MOV	R2,FT.BUF(R4)	:STORE IT BACK
154	005122	104202	011607			MOV	#PRMBUF,R2	:FOR EDC COMPUTATION
155	005124	104307	002171			MOV	SECSIZ,R0	:SECTOR SIZE IN WORDS
156	005126					CALL	CEDC	:COMPUTE IT
157	005130	102200	020000	001703		BIT	#MODE,FLAG1	:WHAT MODE ARE WE IN
158	005133					BEQ	1\$:IF CLEAR THEN 512
159	005135	100623	000440			MOV	R3,RW.E76(R2)	:STORE IT 576 BUFFER
160	005137					BR	2\$:SKIP 512 SETUP
161	005141	100623	000400		1\$:	MOV	R3,RW.EDC(R2)	:STORE IT 512 BUFFER
162	005143	103200	000040	001703	2\$:	BIC	#BDHD,FLAG1	:WAN'T TO STAY PRIMARY
163	005146					BR	LSND	:BRANCH AROUND SECONDARY
164	005150	117400	002136		BRBN:	DEC	ERRCNT	:DEC ERR CNT SO PRIMARY STATS WILL BE RIGHT
165	005152	102200	001000	001702		BIT	#PRIM,FLAG	:IS THERE A PRIMARY ON THIS TRACK ?
166	005155					BEQ	7\$:NO - SKIP HEADER RESET
167	005157	104203	015763			MOV	#IMAGE,R3	:POINT TO FORMAT TABLE
168	005161	104632	000002		5\$:	MOV	FT.HI(R3),R2	:GET HEADER WORD
169	005163	103202	007777			BIC	#LO,R2	:CLEAR ALL BUT HEADER
170	005165	106202	050000			CMF	#HD.PRIV,R2	:IS IT THE PRIMARY ?
171	005167					BEQ	6\$:YES - DONE

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 56-3
LBN FORMATTING OVERLAY (F2)

172	005171	105203	000003		ADD	#IMLEN,R3	:NO - POINT TO NEXT ENTRY
173	005173				BR	5\$:CHECK NEXT ENTRY
174	005175	104632	000002	6\$:	MOV	FT.HI(R3),R2	:RESET TO HI ORDER
175	005177	103202	170000		BIC	#HD.CLR,R2	:CLEAR HEADER
176	005201	101202	030000		BIS	#HD.REV,R2	:MARK AS SECONDARY
177	005203	100632	000002		MOV	R2,FT.HI(R3)	:STORE BACK
178	005205	104202	011132		MOV	#GDBLK,R2	:POINT TO GOOD BLOCK
179	005207	104635	000000		MOV	FT.BUF(R3),R5	:GET BUFFER POINTER AND FLAGS
180	005211	103205	007777		BIC	#BUFMSK,R5	:CLEAR ONLY BUFFER POINTER
181	005213	101025			BIS	R2,R5	:OR IN NEW BUFFER POINTER
182	005214	100635	000000		MOV	R5,FT.BUF(R3)	:MOVE IN BUFFER POINTER AND FLAGS
183	005216	104305	001737		MOV	ERPNT,R5	:GET REVECTOR POINTER
184	005220	102200	040000	001703	BIT	#FPRIM,FLAG1	:WAS IT A PRIMARY FROM THE FCT ?
185	005223				BNE	9\$:YES - UPDATE WILL MAKE IT SECONDARY
186	005225	107205	000002	8\$:	SUB	#ERLEN,R5	:LOOK FOR PRIMARY BACKWARDS
187	005227	104652	000001		MOV	1(R5),R2	:GET HIGH ORDER
188	005231	103202	007777		BIC	#LO,R2	:CLEAR LO STUFF
189	005233	106202	050000		CMP	#HD.PRIV,R2	:IS IT THE PRIMARY ?
190	005235				BNE	8\$:NO - KEEP LOOKING
191	005237	104652	000001		MOV	1(R5),R2	:GET HIGH ORDER AGAIN
192	005241	103202	170000		BIC	#HD.CLR,R2	:CLEAR THE HEADER
193	005243	101202	030000		BIS	#HD.REV,R2	:MAKE IT SECONDARY
194	005245	100652	000001		MOV	R2,1(R5)	:STORE IT BACK
195	005247	115400	002155	9\$:	INC	SNDCNT	:INC SECONDARY COUNTER
196	005251	103200	000004	001703	BIC	#RPRIM,FLAG1	:DON'T NEED GOOD EDC ANY LONGER
197	005254	103200	000040	001703	7\$:	BIC	#BDHD,FLAG1
198	005257	101200	001000	001702	BIS	#PRIM,FLAG	:CLEAR SC WILL PUT IN AS RBN
199	005262	104643	000002		BDIRCT: MOV	FT.HI(R4),R3	:SET SO NONE WILL BE PRIMARY
200	005264	103203	170000		BIC	#HD.CLR,R3	:RELOAD HEADER
201	005266	101203	110000		BIS	#HD.BAD,R3	:CLEAR THE HEADER
202	005270	100643	000002		MOV	R3,FT.HI(R4)	:MARK AS BAD
203	005272	102200	000100	001702	BIT	#REVECT,FLAG	:STORE BACK IN IMAGE
204	005275				BNE	LSND	:IN RCT ?
205	005277				BR	LSKIP3	:NO - PUT IN TO REVECTOR
206	005301	102200	000040	001703	SND:	#BDHD,FLAG1	:ELSE DO NEXT ENTRY
207	005304				BNE	BDIRCT	:BAD HEADER ?
208	005306	101203	030000		BIS	#HD.REV,R3	:YUP - MARK AS BAD
209	005310	100643	000002		MOV	R3,FT.HI(R4)	:MARK AS SECONDARY
210	005312	115400	002155		INC	SNDCNT	:STORE BACK IN IMAGE
211	005314	115400	001704	LSND:	INC	ERFLAG	:INC IT
212	005316	104303	001737		MOV	ERPNT,R3	:SET RE-FORMAT FLAG
213	005320	104642	000001		MOV	FT.LOW(R4),R2	:STORE BACK
214	005322	100232			MOV	R2,(R3)+	:GET LOW ORDER BLOCK NUMBER
215	005323	104642	000002		MOV	FT.HI(R4),R2	:STORE FOR RCT UPDATE
216	005325	102200	000040	001703	BIT	#BDHD,FLAG1	:GET HIGH ORDER
217	005330				BEQ	LSKIP9	:BAD HEADER ?
218	005332	103202	170000		BIC	#HD.CLR,R2	:NO - HANDLE AS USUAL
219	005334	101202	030000		BIS	#HD.REV,R2	:ELSE CLEAR BAD HEADER CODE
220	005336	115400	002155		INC	SNDCNT	:AND PUT IN SECONDARY CODE
221	005340	100232		LSKIP9:	MOV	R2,(R3)+	:INC SECONDARY COUNT
222	005341	104030	001737		MOV	R3,ERPNT	:STORE FOR RCT UPDATE
223	005343	115400	001741		INC	REVCNT	:STORE BACK
224	005345	115400	002136		INC	ERRCNT	:INCREMENT :T
225	005347				BR	LSKIP3	:UP COUNTER OF BAD BLOCKS
226	005351	117400	002141	LOK:	DEC	N1	:NO NEED TO RE-READ ANY MORE THIS SECTOR
227	005353				BNE	LSKIP8	:DECREMENT COUNTER
228	005355	104300	002142	002141	LSKIP3: MOV	NN1,N1	:NO - RE-READ SECTOR IN ERROR
							:GET SAVED VALUE

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 56-4
LBN FORMATTING OVERLAY (F2)

229	005360	117400	001711		DEC	ERR	:DECREMENT I
230	005362				POP	R4	:RESTORE COMMAND POINTER
231	005363	103200	000040	001703	LSKIP7: BIC	#BDHD,FLAG1	:CLEAR BAD HEADER FLAG
232	005366	105204	000004		ADD	#RDLEN,R4	:POINT TO NEXT ENTRY
233	005370	115000	001711		TST	ERR	:DONE ALL SECTORS ?
234	005372				BNE	LHERE	:NO - DO NEXT SECTOR
235	005374	115000	001704		TST	ERFLAG	:WERE THERE ANY BAD SECTORS FOUND
236	005376				BEQ	LDONE	:NOPE - ALL DONE
237	005400	104207	014154		MOV	#RBNBUF,R0	:POINT TO RBN BUFFER
238	005402	104301	002131		MOV	DWRD,R1	:DIAGNOSTIC WORD
239	005404	100271			MOV	R1,(R0)+	:STORE IT
240	005405	102200	020000	001703	BIT	#MODE,FLAG1	:WHAT MODE ARE WE IN
241	005410				BEQ	1\$:IF CLEAR THEN 512
242	005412	104204	000437		MOV	#287.,R4	:ELSE 576 - USE 287 AS WORD COUNT
243	005414				BR	2\$:SKIP 512 SETUP
244	005416	104204	000377	1\$:	MOV	#255.,R4	:WORD COUNT FOR 512 MODE
245	005420	104301	002126	2\$:	MOV	FWRD,R1	:FIRST WORD OF PATTERN
246	005422	104302	002127		MOV	SWRD,R2	:SECOND WORD OF PATTERN
247	005424	104303	002130		MOV	TWRD,R3	:THIRD WORD OF PATTERN
248	005426	100271		LOVER:	MOV	R1,(R0)+	:STORE IT
249	005427	117404			DEC	R4	:DECREMENT COUNT
250	005430				BEQ	LOVER5	:QUIT IF 0
251	005432	100272			MOV	R2,(R0)+	:STORE IT
252	005433	117404			DEC	R4	:DECREMENT COUNT
253	005434				BEQ	LOVER5	:QUIT IF 0
254	005436	100273			MOV	R3,(R0)+	:STORE IT
255	005437	117404			DEC	R4	:DECREMENT COUNTER
256	005440				BNE	LOVER	:REPEAT TILL DONE
257	005442	102200	020000	001703	LOVER5: BIT	#MODE,FLAG1	:WHAT MODE ARE WE IN
258	005445				BEQ	1\$:IF CLEAR THEN 512
259	005447	104302	002135		MOV	BADE76,R2	:EDC FOR PATTERN (576) FORCED ERR
260	005451				BR	2\$:SKIP 512 SETUP
261	005453	104302	002133	1\$:	MOV	BADEDC,R2	:EDC FOR PATTERN (512) FORCED ERR
262	005455	100272		2\$:	MOV	R2,(R0)+	:STORE IT
263	005456	102200	000004	001703	BIT	#RPRIM,FLAG1	:NEED GOOD RBN EDC ???
264	005461				BEQ	LOVER2	:NOPE
265	005463	104203	015763		MOV	#IMAGE,R3	:POINT TO IMAGE
266	005465	104205	011132		MOV	#GDBLK,R5	:POINT TO GOOD BLOCK
267	005467	104632	000002	LOVER4:	MOV	FT.HI(R3),R2	:GET HI ORDER
268	005471	103202	007777		BIC	#LO,R2	:CLEAR JUNK
269	005473	106202	060000		CMP	#HD,RBN,R2	:IS IT THE PRIMARY ?
270	005475				BEQ	LOVER3	:YUP - HANDLE IT
271	005477	105203	000003		ADD	#IMLEN,R3	:CHECK NEXT ENTRY
272	005501				BR	LOVER4	:TRY AGAIN
273	005503	104632	000000	LOVER3:	MOV	FT.BUF(R3),R2	:GET BUFFER POINTER
274	005505	103202	007777		BIC	#BUFMSK,R2	:CLEAR ONLY BUFFER POINTER
275	005507	101052			BIS	R5,R2	:SET IN NEW BUFFER POINTER
276	005510	100632	000000		MOV	R2,FT.BUF(R3)	:STORE IT
277	005512	104304	002006	LOVER2:	MOV	DPREA,R4	:DATA PREAMBLE LENGTH
278	005514	104303	002005		MOV	HPREA,R3	:HEADER PREAMBLE LENGTH
279	005516	104307	002004		MOV	IMSTAR,R0	:POINT TO TRACK IMAGE START POINT
280	005520	104301	001565		MOV	CURTRK,R1	:CURRENT TRACK NUMBER
281	005522	104302	002171		MOV	SECSIZ,R2	:SECTOR SIZE IN WORDS
282	005524	104205	015763		MOV	#IMAGE,R5	:RECIRCULATION ADDRESS
283	005526	060001			XFC	FORMAT	:RE-FORMAT
284	005527	115001			TST	R1	:ANY PROBLEMS ??
285	005530				BEQ	LOVER1	:NO - DO CHECK PASS

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 56-5
LBN FORMATTING OVERLAY (F2)

286	005532	115400	001371		INC	UN.ERI	:INCREMENT IT
287	005534	106300	002165	001371	CMP	RETRY,UN.ERI	:DONE ALL RETRIES ?
288	005537				BMI	FORERR	:YUP - ERROR
289	005541				CALL	INITPT	:REINIT
290	005543				CALL	CLEAR	:DRIVE CLEAR
291	005545	104300	002146	001553	MOV	CURGRP,ISEEK+3	:GROUP
292	005550				CALL	SEEK	:RE-SEEK AND GROUP SELECT
293	005552				BR	LOVER2	:NOPE - RETRY
294	005554	114000	001371		LOVER1: CLR	UN.ERI	:CLEAR RETRY COUNT
295	005556				BR	LCHEC	:RE-CYCLE CHECK PASS
296	005560				LDONE: RETURN		

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 57
LBN FORMATTING OVERLAY (F2)

```

1
2
3
4
5
6 005562 104300 001606 002137 LFIXIT: MOV SECTRK,SECCNT :INIT COUNTER
7 005565 104207 013022 :MOV #CMDBUF,R0 :COMMAND BUFFER
8 005567 104205 015763 :MOV #IMAGE,R5 :POINT TO TRACK IMAGE
9 005571 104303 001716 :MOV SKPCNT,R3 :GET STARTING OFFSET(TUNED)
10 005573 105035 :ADD R3,R5 :POINT TO FIRST ENTRY
11 005574 104050 001715 :MOV R5,STARIT :MARK STARTING ADDRESS
12 005576 104653 000002 LMORE: MOV 2(R5),R3 :SET UP FOR HSR CODE COMPARE
13 005600 193203 007777 :BIC #LO,R3 :ISOLATE HI 4 BITS(HDR CODE)
14 005602 106203 000000 :CMP #HD.LBN,R3 :GOOD LBN ?
15 005604 :BEQ FLKIP2 :YES - MARK AS GOOD TO CHECK
16 005606 106203 060000 :CMP #HD.RBN,R3 :GOOD RBN ?
17 005610 :BEQ FLKIP2 :YES - MARK AS GOOD TO CHECK
18 005612 114003 :CLR R3 :CLEAR FOR STORE
19 005613 100673 000002 :MOV R3,RB.CMD(R0) :STORE AS BAD SECTOR FLAG
20 005615 105207 000004 :ADD #RDLEN,R0 :POINT TO NEXT BLOCK
21 005617 :BR FLKIP1 :SKIP GOOD MARK
22 005621 104653 000001 FLKIP2: MOV 1(R5),R3 :LO ORDER BLOCK NUMBER
23 005623 100273 :MOV R3,(R0)+ :STORE IN READ CMD BLOCK
24 005624 104653 000002 :MOV 2(R5),R3 :HI ORDER BLOCK NUM AND CODE
25 005626 100273 :MOV R3,(R0)+ :STORE IN READ CMD BLOCK
26 005627 104203 013400 :MOV #RWCMD,R3 :LOAD SDI READ COMMAND
27 005631 101303 001565 :BIS CURTRK,R3 :SET IN CURRENT TRACK NUMBER
28 005633 100273 :MOV R3,(R0)+ :STORE IN BLOCK
29 005634 100275 :MOV R5,(R0)+ :SAVE PTR TO IMAGE BLK ENTRY
30 005635 105305 001717 FLKIP1: ADD TBLK,R5 :ADD TO GET NEXT SECTOR
31 005637 106305 001714 :CMP EIMAGE,R5 :SEE IF HAVE TO LOOP BACK TO TOP
32 005641 :BEQ LREDO :NEED TO RESET
33 005643 :BPL FLKIP1 :NO NEED - JUST CONTINUE
34 005645 107305 001714 :SUB EIMAGE,R5 :SUBTRACT TO GET LOOP AMOUNT
35 005647 105205 015763 :ADD #IMAGE,R5 :AND ADD OFFSET
36 005651 :BR FLKIP1 :SKIP ZERO CONDITION
37 005653 104205 015763 LREDO: MOV #IMAGE,R5 :IF ZERO SIMPLY MOVE TO FRONT
38 005655 106305 001715 FLKIP1: CMP STARIT,R5 :AT BEGINNING ADDRESS ?
39 005657 :BNE FKIP9 :NO - JUST CONTINUE
40 005661 105205 000003 :ADD #IMLEN,R5 :ELSE POINT TO NEXT ENTRY
41 005663 104050 001715 :MOV R5,STARIT :MAKE IT NEW STARTING ADDRESS
42 005665 117400 002137 FKIP9: DEC SECCNT :DECREMENT
43 005667 :BNE LMORE :NO - DO NEXT SECTOR
44 005671 :RETURN

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 58
LBN FORMAT IMAGE SETUP OVERLAY (F8)

```

1          .SBTTL LBN FORMAT IMAGE SETUP OVERLAY (F8)
2 005673   DMOVLY F8,START
3
4          :
5          SET UP TRACK FORMAT
6 004014   104200 000025 001636 LTRK:  MOV   #F8,CUROVL      :GET OVERLAY OFFSET
7 004017   104207 001525          MOV   #SCR,RO        :POINT TO CHARACTERISTICS BLOCK
8 004021   102200 020000 001703   BIT   #MODE,FLAG1    :WHAT MODE
9 004024          BNE   1$                :IF SET THEN 576
10 004026   104673 000011          MOV   LBNT12(RO),R3   :GET LBN/TRACK FOR 512
11 004030          BR    2$                :SKIP 576 SETUP
12 004032   104673 000015          MOV   LBNT76(RO),R3  :GET LBN/TRACK FOR 576
13 004034   103203 177400          BIC   #HIBYTE,R3     :CLEAR HIGH BYTE
14 004036   104030 002137          MOV   R3,SECCNT      :USE AS COUNTER
15 004040   104205 015763          MOV   #IMAGE,R5      :POINT TO TRACK IMAGE BUFFER
16 004042   104203 011132          MOV   #GDBLK,R3      :POINT TO DATA BLOCK
17 004044   104304 002132          MOV   EDC,R4         :GET GOOD EDC
18 004046   102200 020000 001703   BIT   #MODE,FLAG1    :WHAT MODE ARE WE IN
19 004051          BEQ   3$                :IF CLEAR THEN 512
20 004053   100624 000440          MOV   R4,RW.E76(R2)  :STORE IT 576 BUFFER
21 004055          BR    4$                :SKIP 512 SETUP
22 004057   100624 000400          MOV   R4,RW.EDC(R2)  :STORE IT 512 BUFFER
23 004061   103200 001000 001702   3$:  BIC   #PRIM,FLAG    :CLEAR PRIMARY FLAG
24 004064   104203 011132          4$:  BIC   #GDBLK,R3     :POINT TO GOOD DATA BLOCK
25 004066   100653 000000          LKIP2: MOV   R3,FT.BUF(R5) :STORE IN IMAGE BLOCK
26 004070   104303 001566          MOV   CURBN,R3       :GET LOW ORDER BLOCK NUMBER
27 004072   100653 000001          MOV   R3,FT.LOW(R5)  :STORE IN IMAGE BLOCK
28 004074   104303 001567          MOV   CURBN+1,R3     :GET HIGH ORDER BLOCK NUMBER
29 004076   103203 170000          BIC   #HD.CLR,R3     :CLEAR HEADER
30 004100   101203 000000          BIS   #HD.LBN,R3     :SET IN LBN HEADER
31 004102   100653 000002          MOV   R3,FT.HI(R5)   :STORE IN IMAGE BLOCK
32 004104   105205 000003          ADD   #IMLEN,R5      :POINT TO NEXT ENTRY
33 004106          DUBINC CURBN          :INCREMENT BLOCK NUMBER
34 004114   102200 000100 001702   BIT   #REVECT,FLAG   :STILL IN RCT AREA ?
35 004117          BNE   LKIP9          :NO - NO NEED TO DECREMENT
36 004121   117400 001720          DEC   RCTTOT         :DECREMENT IT
37 004123          BNE   LKIP9          :OUT OF RCT ? - NO
38 004125   101200 020000 001702   BIS   #INIRCT,FLAG   :SET TO INIT RCT
39 004130   104300 001722 001777   MOV   FCNT,LBNBAD    :GET FCT ENTRY COUNT - AFTER RCT
40 004133          LKIP9:
41 004133   117400 002137          DEC   SECCNT         :DECREMENT COUNTER
42 004135          BNE   LKIP2          :CONTINUE TILL DONE ALL
43
44          :
45          RBN SETUP
46 004137          PUSH   R5                :SAVE POINTER TO FIRST RBN ENTRY
47 004140   104207 001525          MOV   #SCR,RO        :POINT TO CHARACTERISTICS
48 004142   104673 000004          MOV   RBNTRK(RO),R3  :GET RBN'S/TRACK
49 004144   103203 177600          BIC   #HIBYTE,R3     :CLEAR HIGH GARBAGE
50 004146   104030 002137          MOV   R3,SECCNT      :USE AS COUNTER
51 004150   104207 014154          LKIP8: MOV   #RBNBUF,RO   :POINT TO RBN BUFFER
52 004152   104201 011132          MOV   #GDBLK,R1      :POINT TO GOOD BLOCK
53 004154   104302 002171          MOV   SECSIZ,R2      :COUNT OF WORDS IN SECTOR
54 004156   104213          LKIP20: MOV   (R1)+,R3     :GET WORD
55 004157   100273          MOV   R3,(R0)+      :STORE TI
56 004160   117402          DEC   R2             :DECREMENT COUNTER
57 004161          BNE   LKIP20          :DO WHOLE BUFFER

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 58-1
LBN FORMAT IMAGE SETUP OVERLAY (F8)

```

58 004163 102200 020000 001703 BIT #MODE,FLAG1 ;WHAT MODE ARE WE IN
59 004166 BEQ 1$ ;IF CLEAR THEN 512
60 004170 104302 002135 MOV BADE76,R2 ;EDC FOR PATTERN (FORCED ERROR IND)
61 004172 BR 2$ ;SKIP 512
62 004174 104302 002133 1$: MOV BADEDC,R2 ;EDC FOR PATTERN (FORCED ERROR IND)
63 004176 100272 2$: MOV R2,(R0)+ ;STORE IT
64 004177 104203 014154 LKIP21: MOV #RBNBUF,R3 ;POINT TO BUFFER
65 004201 100653 000000 MOV R3,FT.BUF(R5) ;STORE IN IMAGE
66 004203 104303 001561 MOV CURRBN,R3 ;GET LOW ORDER R&N
67 004205 100653 000001 MOV R3,FT.LOW(R5) ;STORE IN IMAGE
68 004207 104303 001562 MOV CURRBN+1,R3 ;GET HIGH ORDER R&N
69 004211 103203 170000 BIC #HD.CLR,R3 ;CLEAR HEADER
70 004213 101203 060000 BIS #HD.RBN,R3 ;SET IN R&N HEADER
71 004215 100653 000002 MOV R3,FT.HI(R5) ;STORE IN IMAGE
72 004217 105205 000003 ADD #IMLEN,R5 ;POINT TO NEXT ENTRY
73 004221 DUBINC CURRBN ;INCREMENT R&N NUMBER
74 004227 117400 002137 DEC SECCNT ;DECREMENT COUNTER
75 004231 BNE LKIP21 ;CONTINUE TILL DONE
76 ;
77 ;
78 004233 104300 001606 002137 MOV SECTRK,SECCNT ;SET UP COUNTER
79 004236 104204 015763 MOV #IMAGE,R4 ;POINT TO IMAGE
80 004240 104203 001525 MOV #SCR,R3 ;POINT TO CHARACTERISTICS
81 004242 102200 020000 001703 BIT #MODE,FLAG1 ;WHAT MODE ARE WE IN
82 004245 BEQ 1$ ;IF CLEAR THEN 512
83 004247 104632 000015 MOV OFFS76(R3),R2 ;GET GROUP OFFSET (576)
84 004251 BR 2$ ;SKIP 512
85 004253 104632 000011 1$: MOV OFFS12(R3),R2 ;GET GROUP OFFSET (512)
86 004255 110702 2$: SWAB R2 ;GET INTO LOWBYTE
87 004256 103202 177400 BIC #HIBYTE,R2 ;CLEAR HIGH GARBAGE
88 004260 115002 TST R2 ;ANY OFFSET ?
89 004261 BEQ LKIP22 ;NO - SKIP CALCULATIONS
90 004263 115000 002146 TST CURGRP ;IS GROUP ZERO ???
91 004265 BEQ LKIP22 ;YES - NO OFFSET
92 004267 104020 001410 MOV R2,TEMP ;STORE IT
93 004271 114000 001411 CLR TEMP+1 ;FOR STORE
94 004273 104300 002146 001403 MOV CURGRP,DDUMMY ;GET CURRENT GROUP
95 004276 114000 001404 CLR DDUMMY+1 ;CLEAR HIGH WORD
96 004300 104203 001410 MOV #TEMP,R3 ;FOR MUL
97 004302 104204 001403 MOV #DDUMMY,R4 ;DITTO
98 004304 CALL DMUL ;MULTIPLY TO GET OFFSET FOR THIS GROUP
99 004306 106300 001606 001403 LKIP23: CMP SECTRK,DDUMMY ;IS TOTAL OFFSET MORE THAN NUMBER OF SECTORS ?
100 004311 BPL LKIP24 ;NO - ALL IS FINE
101 004313 107300 001606 001403 SUB SECTRK,DDUMMY ;YES - SUBTRACT TILL IT IS
102 004316 BR LKIP23 ;CHECK AGAIN
103 004320 104200 000003 001410 LKIP24: MOV #IMLEN,TEMP ;GET LENGTH OF IMAGE BLOCK
104 004323 104000 001411 CLR TEMP+1 ;FOR STORE
105 004325 104203 001410 MOV #TEMP,R3 ;FOR MUL
106 004327 CALL DMUL ;GET LENGTH TO OFFSET
107 004331 104143 MOV (R4),R3 ;GET RESULT
108 004332 104304 001714 MOV EIMAGE,R4 ;GET ADDRESS OF END OF IMAGE
109 004334 107034 SUB R3,R4 ;SUBTRACT TO GET STARTING LOCATION
110 004335 104040 002004 LKIP22: MOV R4,IMSTAR ;STORE IMAGE POINTER
111 004337 104300 001602 001563 MOV HOLDPN,CURPBN ;GET LOW ORDER P&N
112 004342 104300 001603 001564 MOV HOLDPN+1,CURPBN+1 ;GET HIGH ORDER
113 004345 102200 000001 001702 LKIP27: BIT #FCTAVL,FLAG ;IS FCT AVAILABLE ?
114 004350 BEQ LKIP28 ;NO - ASSUME BLOCK GOOD

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 58-2
LBN FORMAT IMAGE SETUP OVERLAY (F8)

115	004352	102200	000002	001702	BIT	#FCTEMT,FLAG	:FCT EMPTY ?
116	004355				BNE	LKIP28	:YUP - BLOCKIS GOOD
117	004357	104302	001706		MOV	BADPBN,R2	:GET FCT POINTER
118	004361	104121			MOV	(R2),R1	:GET LOW ORDER BAD PBN
119	004362	106010	001563		CMP	R1,CURPBN	:ARE THEY EQUAL ?
120	004364				BNE	LKIP4	:NO - SKIP REST OF COMPARE
121	004366	104621	000001		MOV	1(R2),R1	:GET HIGH ORDER
122	004370	103201	170000		BIC	#HD.CLR,R1	:CLEAR THE HEADER
123	004372	106010	001564		CMP	R1,CURPBN+1	:ARETHEY EQUAL ?
124	004374				BNE	LKIP4	:NO - MUST BE GOOD
125	004376	117400	002163		DEC	PCNT	:DECREMENT IT
126	004400	117400	001722		DEC	FCNT	:DECREMENT FCT COUNT
127	004402				BNE	LKIP12	:IF NON - ZERO SKIP FLAG SET
128	004404	101200	000002	001702	BIS	#FCTEMT,FLAG	:SET EMPTY FLAG
129	004407					LKIP12:	
130	004407	104643	000002		MOV	FT.HI(R4),R3	:GET HEADER WORD
131	004411	103203	007777		BIC	#LO,R3	:CLEAR ALL BUT HEADER
132	004413	106203	060000		CMP	#HD.RBN,R3	:IS IT AN RBN ?
133	004415				BNE	LKIP25	:NOPE - SKIP RBN STUFF
134	004417	101200	001000	001702	BIS	#PRIM,FLAG	:SO WON'T GET PRIMARY ON CHECK PASS
135	004422				BR	MARBAD	:GO MARK AS BAD
136	004424	102200	000100	001702	LKIP25:	BIT	#REVECT,FLAG
137	004427				BNE	6\$:IN RCT ?
138	004431	115400	002000		INC	RCTBAL	:NO - SKIP RCT STUFF
139	004433				BR	MARBAD	:INCREMENT IT
140	004435	104623	000001		MOV	1(R2),R3	:GO MARK BAD
141	004437	102203	100000		BIT	#PRMY,P3	:GET BAD PBN HDR
142	004441				BEQ	LKIP5	:IS IT SECONDARY ?
143	004443	101200	001000	001702	BIS	#PRIM,FLAG	:YES - GO DO IT
144	004446	101200	000004	001703	BIS	#RPRIM,FLAG1	:SET FLAG FOR PRIMARY FOUND
145	004451	101200	040000	001703	BIS	#FPRIM,FLAG1	:SET GOOD EDC NEEDED
146	004454	104643	000002		MOV	FT.HI(R4),R3	:SET PRIMARY IN FCT
147	004456	103203	170000		BIC	#HD.CLR,R3	:GET HIGH ORDER HEADER
148	004460	101203	050000		BIS	#HD.PRIV,R3	:CLEAR HEADER CODE
149	004462	100643	000002		MOV	R3,FT.HI(R4)	:MARK AS PRIMARY REVECTOR
150	004464				PUSH	R4	:STORE IN IMAGE BLOCK
151	004465	104207	001525		MOV	#SCR,R0	:SAVE IMAGE POINTER
152	004467	104204	001566		MOV	#CURBN,R4	:MAKE SURE POINT TO CHAR BLOCK
153	004471				CALL	PRIMRB	:POINT TO BLOCK NUMBER
154	004473				POP	R4	:GET PRIMARY RBN NUMBER
155	004474	104307	001634		MOV	REVRBN,R0	:RESTORE IMAGE POINTER
156	004476	104301	001635		MOV	REVRBN+1,R1	:GET NUMBER OF REVECTORED RBN
157	004500	105301	002010		ADD	ST.RBN,R1	:GET HIGH ORDER
158	004502	101201	060000		BIS	#HD.RBN,R1	:ADD IN STARTING BITS
159	004504	104205	011607		MOV	#PRMBUF,R5	:SET IN RBN HEADER CODE
160	004506	104203	000200		MOV	#RBNRPT,R3	:USE RDBUF TO HOLD 128 COPIES OF RBN
161	004510	100257			MOV	R0,(R5)+	:COUNT OF REPLICATED RBN'S
162	004511	100251			MOV	R1,(R5)+	:STORE A COPY
163	004512	117403			DEC	R3	:AND HIGH ORDER
164	004513				BNE	RPT	:DECREMENT COUNTER - DONE ?
165	004515	104203	011607		MOV	#PRMBUF,R3	:NO - STORE ANOTHER COPY
166	004517	100643	000000		MOV	R3,FT.BUF(R4)	:POINT TO BEGINNING OF BUFFER
167	004521	105200	000002	001706	ADD	#2,BADPBN	:STORE NEW BUFFER PTR IN IMAGE
168	004524	104202	011607		MOV	#PRMBUF,R2	:INCREMENT BADPBN POINTER
169	004526	104307	002171		MOV	SECSIZ,R0	:POINT TO BUFFER
170	004530				CALL	CEDC	:SECTOR SIZE IN WORDS
171	004532	102200	020000	001703	BIT	#MODE,FLAG1	:COMPUTE EDC - RETURNED IN R3
							:WHAT MODE ARE WE IN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 58-3
LBN FORMAT IMAGE SETUP OVERLAY (F8)

172	004535				BEQ	1\$:IF CLEAR THEN 512
173	004537	100623	000440		MOV	R3,RW.E76(R2)		:STORE IT 576 BUFFER
174	004541				BR	2\$:SKIP 512 SETUP
175	004543	100623	000400		1\$: MOV	R3,RW.EDC(R2)		:STORE IT 512 BUFFER
176	004545				2\$: BR	LKIP4		:SKIP SECONDARY REVECTOR
177	004547				LKIP5:			
178	004547	115400	002155		INC	SNDCNT		:INC IT
179	004551	102202	010000		BIT	#FBDHD,R2		:HEADER IN ERROR CODE IN FCT ?
180	004553				BNE	MARBAD		:YUP - MARK BAD
181	004555	104643	000002		MOV	FT.HI(R4),R3		:GET HIGH ORDER HEADER
182	004557	103203	170000		BIC	#HD.CLR,R3		:CLEAR HEADER CODE
183	004561	101203	030000		BIS	#HD.REV,R3		:SET HEADER TO SECONDARY REVECTOR
184	004563	100643	000002		MOV	R3,FT.HI(R4)		:STORE IN IMAGE
185	004565	105200	000002	001706	ADD	#2,BADPBN		:INCREMENT BADPBN POINTER
186	004570				BR	LKIP4		:SKIP GOOD MARK
187	004572				MARBAD:			
188	004572	104643	000002		MOV	FT.HI(R4),R3		:GET HIGH ORDER HEADER
189	004574	103203	170000		BIC	#HD.CLR,R3		:CLEAR HEADER CODE
190	004576	101203	110000		BIS	#HD.BAD,R3		:MARK AS BAD
191	004600	100643	000002		MOV	R3,FT.HI(R4)		:STORE IN IMAGE
192	004602	105200	000002	001706	ADD	#2,BADPBN		:UPDATE COUNTER
193	004605				LKIP4:			
194	004605	105204	000003		ADD	#IMLEN,R4		:POINT TO NEXT IMAGE ENTRY
195	004607	106304	001714		CMP	EIMAGE,R4		:AT THE END ?
196	004611				BNE	LKIP29		:NOPE - CARRY ON
197	004613	104204	015763		MOV	#IMAGE,R4		:POINT TO START
198	004615				LKIP29: DUBINC	CURPBN		:DO THE INCREMENT
199	004623	115000	002163		TST	PCNT		:DONE WITH THIS BLOCK OF PBNs?
200	004625				BNE	LKIP10		:IF NOT DONE SKIP
201	004627				CALL	FCPG		:ELSE PAGE IN NEW FCT BLOCK
202	004631	117400	002137		LKIP10: DEC	SECCNT		:DECREMENT IT
203	004633				BNE	LKIP27		:NO - DO NEXT BLOCK
204					:			
205	004635	102200	000004	001703	LKIP28: BIT	#RPRIM,FLAG1		:NEED GOOD EDC ?
206	004640				BEQ	LKIP7		:NOPE
207	004642	103200	000004	001703	BIC	#RPRIM,FLAG1		:CLEAR FLAG
208	004645				POP	R4		:GET POINTER TO FIRST RBN ENTRY
209	004646	104203	011132		MOV	#GDBLK,R3		:GET GOOD EDC BLOCK
210	004650	100643	000000		MOV	R3,FT.BUF(R4)		:STORE IT IN BUFFER POINTER
211	004652				BR	LKIP33		:SKIP POP
212								
213	004654				LKIP7: POP	R4		:POP STACK (RBN RENTRY ADDRESS)
214	004655	104304	002004		LKIP33: MOV	IMSTAR,R4		:POINT TO FIRST TO FORMAT ENTRY
215	004657	104303	001714		MOV	EIMAGE,R3		:GET END ADDRESS
216	004661	107203	000003		SUB	#IMLEN,R3		:POINT TO FLAG OF LAST ENTRY
217	004663	106043			CMP	R4,R3		:FIRST = LAST ?
218	004664				BEQ	LKIP30		:NO - SKIP SPECIAL STUFF
219	004666	104135			MOV	(R3),R5		:GET FLAG WORD
220	004667	101205	040000		BIS	#RECIR,R5		:SET RECIRCULATION FLAG
221	004671	100135			MOV	R5,(R3)		:STORE IT BACK
222	004672				BR	LKIP31		:SKIP KLUDGE FIX TO UDA
223	004674	101200	040000	002004	LKIP30: BIS	#RECIR,IMSTAR		:SET BIT IN POINTER
224	004677	104143			LKIP31: MOV	(R4),R3		:GET BUFF POINTER
225	004700	101203	100000		BIS	#LAST,R3		:SIGNAL AS LAST
226	004702	100143			MOV	R3,(R4)		:STORE IT BACK
227	004703	104204	001602		MOV	#HOLDPN,R4		:FOR DECREMENT
228	004705	104203	001606		MOV	#SECTRK,R3		:DITTO

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 58-4
LBN FORMAT IMAGE SETUP OVERLAY (F8)

229 004707
230 004711

CALL DSUB
RETURN

;SUBTRAC TO GET NEXT TRACK

UI
RI

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 59
 LBN FORMAT IMAGE SETUP OVERLAY (F8)

UD
 RC1

1				:			
2				:			
3				:			
4	004713			:	PAGE IN NEXT FCT BLOCK		
5	004715	102200	000002	:	FPCG: PUSH R5,R4	:	SAVE R5 AND R4
6	004720			:	BIT #FCTEMT,FLAG	:	EMPTY FCT
7	004722	104200	010455	:	BNE 1\$:	YUP - DON'T GET BLOCK
8	004725	104201	000017	:	MOV #PBNBUF,BUFPNT	:	POINT TO BUFFER
9	004727			:	MOV #F6,R1	:	OVERLAY F6 DOES IT
10	004731	104200	000200	:	CALL PAGE	:	EXECUTE IT
11	004734			:	MOV #128,,PCNT	:	RESET COUNT
12	004736			:	1\$: POP R4,R5	:	RESTORE R4 AND R5
				:	RETURN	:	RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 60
L/RBN COMPUTE OVERLAY (G8)

1					.SBTTL L/RBN COMPUTE OVERLAY (G8)	
2						
3						
4					THIS OVERLAY COMPUTES LBN AND RBN OF THE LAST TRACK ON LAST LBN CYLINDER	
5					AND COMPUTES THE PBN OF THAT LBN	
6	004740				DMOVLY G8,START	
7						
8						
9	004014	104200	000052	001636	MOV #G8,CUROVL	:FOR RECORDING
10	004017	104207	001525		MOV #SCR,R0	:POINT TO CHARACTERISTICS BLOCK
11	004021				CALL NUMLBN	:GET NUMBER OF FIRST LBN ON LAST LBN CYL
12	004023	104140	001566		MOV (R4),CURBN	:GET LOW ORDER
13	004025	104140	001576		MOV (R4),HOLDBN	:SAVE FOR LATER
14	004027	104640	000001	001567	MOV 1(R4),CURBN+1	:GET HIGH ORDER
15	004032	104640	000001	001577	MOV 1(R4),HOLDBN+1	:SAVE FOR LATER
16	004035				CALL NUMRBN	:GET NUM OF FIRST RBN ON LAST LBN CYLINDER
17	004037	104140	001561		MOV (R4),CURRBN	:GET LOW ORDER
18	004041	104640	000001	001562	MOV 1(R4),CURRBN+1	:GET HIGH ORDER
19	004044	104140	001600		MOV (R4),HOLRBN	:SAVE LOW FOR LATER
20	004046	104640	000001	001601	MOV 1(R4),HOLRBN+1	:SAVE HIGH FOR LATER
21	004051				CALL LPBN	:GET PBN OF FIRST SECTOR ON LAST TRACK
22	004053				CALL SETSIZ	:SET UP 512/576 VARIABLES
23	004055				RETURN	

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 61
L/RBN COMPUTE OVERLAY (G8)

1					:				
2					:				
3					:				
4					:				
5	004057	104300	001626	001410	:	NUMLBN:	MOV	LBNCYL,TEMP	:GET LOW ORDER NUM OF LBN CYLINDERS
6	004062	104300	001627	001411	:		MOV	LBNCYL+1,TEMP+1	:GET HIGH ORDER
7	004065	103200	170000	001411	:		BIC	#HD.CLR,TEMP+1	:CLEAR STARTING CYLINDER BITS
8	004070	104204	001410		:		MOV	#TEMP,R4	:DITTO
9	004072	104203	001630		:		MOV	#LBNPCY,R3	:GET LBN'S/CYLINDER
10	004074				:		CALL	DMUL	:GET FIRST LBN ON LAST CYLINDER
11	004076	104641	000001		:		MOV	1(R4),R1	:GET LBN
12	004100	105301	002007		:		ADD	ST.LBN,R1	:ADD STARTING LBN TO GET ABSOLUTE LBN
13	004102	100641	000001		:		MOV	R1,1(R4)	:STORE BACK
14	004104	102200	020000	001703	:		BIT	#MODE,FLAG1	:WHAT MODE
15	004107				:		BNE	1\$:IF SET THEN 576
16	004111	104673	000011		:		MOV	LBNT12(R0),R3	:GET LBN/TRACK FOR 512
17	004113				:		BR	2\$:SKIP 576 SETUP
18	004115	104673	000015		:	1\$:	MOV	LBNT76(R0),R3	:GET LBN/TRACK FOR 576
19	004117	103203	177400		:	2\$:	BIC	#HIBYTE,R3	:CLEAR HIGH BYTE
20	004121	104030	001403		:		MOV	R3,DDUMMY	:STORE IT
21	004123	114000	001404		:		CLR	DDUMMY+1	:FOR STORE
22	004125	104203	001403		:		MOV	#DDUMMY,R3	:LBN/TRACK
23	004127				:		CALL	DSUB	:WANT LBN ON LAST TRACK
24	004131				:		RETURN		

UDAF52 - UDA-52 FORMATTER DMACR X94.01 23-AUG-82 13:14:22 PAGE 62
L/RBN COMPUTE OVERLAY (G8)

1									
2									
3									
4									
5	004133	104300	001626	001410	NUMRBN:	MOV	LBN CYL, TEMP		:GET LOW ORDER NUMBER OF LBN CYLINDER
6	004136	104300	001627	001411		MOV	LBN CYL+1, TEMP+1		:GET HIGH ORDER
7	004141	103200	170000	001411		BIC	#HD.CLR, TEMP+1		:CLEAR STARTING CYLINDER BITS
8	004144	104204	001410			MOV	#TEMP, R4		:DITTO
9	004146	104203	001632			MOV	#RBNPCY, R3		:GET RBN'S/CYLINDER
10	004150					CAL!	DMUL		:GET FIRST RBN ON LAST CYLINDER
11	004152	104641	000001			MOV	1(R4), R1		:GET HIGH ORDER
12	004154	105301	002010			ADD	ST.RBN, R1		:ADD TO GET ABSOLUTE LBN
13	004156	100641	000001			MOV	R1, 1(R4)		:STORE BACK
14	004160	104673	000004			MOV	RBNTRK(R0), R3		:GET RBN/TRACK
15	004162	103203	177600			BIC	#H11BYTE, R3		:CLEAR OUT GARBAGE
16	004164	104030	001403			MOV	R3, DDUMMY		:STORE IT
17	004166	114000	001404			CLR	DDUMMY+1		:FOR STORE
18	004170	104203	001403			MOV	#DDUMMY, R3		:WANT LAST TRACK
19	004172					CALL	DSUB		:GET IT
20	004174					RETURN			

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 63
L/RBN COMPUTE OVERLAY (G8)

1									
2									
3									
4									
5	004176	104300	001566	001410	LPBN:	MOV	CURBN,TEMP	:	GET LOW ORDER
6	004201	104300	001567	001411		MOV	CURBN+1,TEMP+1	:	GET HIGH ORDER
7	004204	104204	001410			MOV	#TEMP,R4	:	FOR SUBTRACT
8	004206	104641	000001			MOV	1(R4),R1	:	GET HIGH ORDER
9	004210	107301	002007			SUB	ST.LBN,R1	:	SUB STARTING LBN TO GET ABSOLUTE LBN
10	004212	100641	000001			MOV	R1,1(R4)	:	STORE BACK
11	004214	102200	020000	001703		BIT	#MODE,FLAG1	:	WHAT MODE
12	004217					BNE	1\$:	IF SET THEN 576
13	004221	104673	000011			MOV	LBNT12(R0),R3	:	GET LBN/TRACK FOR 512
14	004223					BR	2\$:	SKIP 576 SETUP
15	004225	104673	000015		1\$:	MOV	LBNT76(R0),R3	:	GET LBN/TRACK FOR 576
16	004227	103203	177400		2\$:	BIC	#HI1BYTE,R3	:	CLEAR HIGH BYTE
17	004231	104030	001403			MOV	R3,DDUMMY	:	STORE FOR COMPUTATION
18	004233	114000	001404			CLR	DDUMMY+1	:	CLEAR FOR STORE
19	004235	104203	001403			MOV	#DDUMMY,R3	:	FOR DIVIDE
20	004237					CALL	DDIV	:	GET NUMBER OF TRACKS
21	004241	104673	000004			MOV	RENTRK(R0),R3	:	GET RBN'S/TRACK
22	004243	103203	177600			BIC	#HI1BYTE,R3	:	CLEAR GARBAGE
23	004245	104030	001403			MOV	R3,DDUMMY	:	FOR COMPUTATION
24	004247	114000	001404			CLR	DDUMMY+1	:	CLEAR HIGH WORD
25	004251	104203	001403			MOV	#DDUMMY,R3	:	FOR MULTIPLY
26	004253					CALL	DMUL	:	GET NUMBER OF RBN'S
27	004255	104300	001566	001403		MOV	CURBN,DDUMMY	:	GET LOW ORDER CURRENT BLOCK NUMBER
28	004260	104300	001567	001404		MOV	CURBN+1,DDUMMY+1	:	GET HIGH ORDER
29	004263	107300	002007	001404		SUB	ST.LBN,DDUMMY+1	:	SUBTRACT STARTING
30	004266	104203	001403			MOV	#DDUMMY,R3	:	FOR ADD
31	004270					CALL	DADD	:	ADD TO GET PBN
32	004272	104140	001602			MOV	(R4),HOLDPN	:	GET LOW ORDER
33	004274	104640	000001	001603		MOV	1(R4),HOLDPN+1	:	STORE HIGH ORDER
34	004277					RETURN			

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 65
 FCT DOWN-LINE LOAD OVERLAY (F3)

```

1          .SBTTL FCT DOWN-LINE LOAD OVERLAY (F3)
2
3          :
4          DOWNLINE LOADER FOR FCT
5          FDLL: DMOVLY F3,START          ;OVERLAY #3
6
7          004014 104200 000006 001636   MOV      #F3,CUROVL          ;OVERLAY #3
8          004017 104200 000001 002164   MOV      #1,COUNT           ;INIT COUNT TO 1
9          004022 104300 002011 001573   MOV      ST.XBN,CURXBN+1    ;ALSO INITIALIZE XBN COUNTER
10         004025 104300 002011 001567   MOV      ST.XBN,CURBN+1    ;HIGH ORDER
11         004030 114000 001572           CLR      CURXBN             ;LOW ORDER IS ZERO
12         004032 114000 001566           CLR      CURBN              ;DITTO
13         004034 104207 002110           MOV      #CONBLK,R0         ;POINT TO CONVERT BLOCK
14         004036 104203 001525           MOV      #SCR,R3            ;POINT TO CHARACTERISTICS
15         004040 104632 000000           MOV      CYLBN(R3),R2       ;GET LBN CYLINDERS
16         004042 100672 000000           MOV      R2,V1(R0)          ;STORE IN CONVERT BLOCK
17         004044 104632 000001           MOV      CYLBN+1(R3),R2     ;GET HIGH ORDER
18         004046 100672 000001           MOV      R2,V1+1(R0)        ;STORE IT
19         004050 104303 001606           MOV      SECTRK,R3          ;GET SECTORS/TRACK
20         004052 100673 000004           MOV      R3,V3(R0)          ;STORE IN CONVERT BLOCK
21         004054 102200 002000 001702   BIT      #BSTGS,FLAG        ;IN BEST GUESS MODE ?
22         004057           BNE      NODLL              ;YUP - FIX UP FIRST BLOCK
23         004061           BR       LOOP               ;START LOOP
24         004063 104203 010000   NODLL: MOV      #RDBUF,R3      ;POINT TO BUFFER
25         004065 114002           CLR      R2                 ;SET MEDIA MODE TO 0 (IN FORMAT)
26         004066 100132           MOV      R2,(R3)            ;STORE IT
27         004067 104204 001772           MOV      #SERNUM,R4         ;POINT TO SERIAL NUMBER
28         004071 105203 000002           ADD      #FSER,R3           ;POINT TO ENTRY IN FCT BLOCK
29         004073 104205 000004           MOV      #4,R5              ;INIT COUNTER
30         004075 104242   9%:   MOV      (R4)+,R2           ;GET WORD
31         004076 100232           MOV      R2,(R3)+          ;STORE WORD
32         004077 117405           DEC      R5                 ;DECREMENT COUNTER
33         004100           BNE      9%                 ;CONTINUE TILL DONE
34         004102 114002           CLR      R2                 ;FOR INSTANCE NUMBER
35         004103 104203 010000   MOV      #RDBUF,R3          ;RESET POINTER
36         004105 100632 000001   MOV      R2,INST(R3)        ;STORE INSTANCE NUMBER IN BLOCK
37         004107 101202 100000   BIS      #NOFCT,R2          ;SET NO FCT AVAILABLE BIT
38         004111 100632 000025   MOV      R2,FCTFLG(R3)     ;STORE IT IN FCT INFO BLOCK
39         004113           BR       LOOPP2             ;SKIP DLL STUFF
40         004115 104205 001750   LOOP:  MOV      #DMBUF,R5      ;POINT TO MAINT BUFFER
41         004117 104303 002003   MOV      FCMMSG,R3          ;GET DUP CODE
42         004121 100153           MOV      R3,(R5)            ;STORE IT IN MESSAGE
43         004122 104303 001566   MOV      CURBN,R3           ;GET BLOCK NUMBER DESIRED
44         004124 100653 000001   MOV      R3,1(R5)           ;STORE IT
45         004126           CALL     SNDMNT              ;SEND REQUEST
46         004130           CALL     RCVMNT              ;RECEIVE ANSWER
47         004132 104153           MOV      (R5),R3            ;GET STATUS WORD
48         004133           BNE      DONDLL             ;IF NOT ZERO THEN DONE
49         004135 104650 000001 002105   MOV      1(R5),OVLBLK+1     ;GET LOW HOST ADDRESS
50         004140 104650 000002 002106   MOV      2(R5),OVLBLK+2     ;GET HIGH HOST ADDRESS
51         004143 104200 000401 002104   MOV      #257,OVLBLK        ;NUMBER OF WORDS TO TRANSFER
52         004146 104204 002104           MOV      #OVLBLK,R4         ;FOR OVERLAY ROUTINE
53         004150 104203 010000   MOV      #RDBUF,R3          ;POINT TO BUFFER
54         004152           CALL     OVRLAY              ;GET THE SECTOR
55         004154 114005   LOOPP2: CLR      R5             ;CLEAR WRITE ERROR COUNT
56         004155 104050 001732           MOV      R5,NEXT1           ;CLEAR REPEAT COUNT
57         004157 106200 000001 002164   CMP      #1,COUNT           ;IS IT THE FIRST ONE ?

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 65-1
 FCT DOWN-LINE LOAD OVERLAY (F3)

58	004162			BNE	LOOPP		:NO - SKIP THIS STUFF
59	004164	104204	001766	MOV	#DATE,R4		:POINT TO SERIAL NUMBER
60	004166	104203	010000	MOV	#RDBUF,R3		:POINT TO BUFFER
61	004170	114005		CLR	R5		:FOR MEDIA FORMAT UPDATE
62	004171	100135		MOV	R5,(R3)		:SET FORMAT IN PROGRESS
63	004172	105203	000012	ADD	#FDAT,R3		:POINT TO ENTRY IN FCT BLOCK
64	004174	104205	000004	MOV	#4,R5		:INIT COUNTER
65	004176	104242		10\$:	MOV	(R4)+,R2	:GET WORD
66	004177	100232		MOV	R2,(R3)+		:STORE WORD
67	004200	117405		DEC	R5		:DECREMENT COUNTER
68	004201			BNE	10\$:CONTINUE TILL DONE
69	004203	114005		CLR	R5		:CLEAR R5 (ERROR COUNTER)
70	004204	104202	010000	LOOPP:	MOV	#RDBUF,R2	:POINT TO BUFFER
71	004206	104207	000400	MOV	#SECSI6,R0		:SECTOR SIZE IN WORDS
72	004210			CALL	CEDC		:COMPUTE EDC
73	004212	100623	000400	MOV	R3,RW.EDC(R2)		:STORE IT
74	004214	104300	001566	001410	MOV	CURBN,TEMP	:GET LOW ORDER
75	004217	104300	001567	001411	MOV	CURBN+1,TEMP+1	:GET HIGH ORDER
76	004222	104204	001410	MOV	#TEMP,R4		:FOR SUB
77	004224	104641	000001	MOV	1(R4),R1		:GET HIGH ORDER
78	004226	107301	002011	SUB	ST.XBN,R1		:SUBTRACT STARTING XBN
79	004230	100641	000001	MOV	R1,1(R4)		:STORE BACK
80	004232			CALL	CVTSK		:CONVERT AND SEEK
81	004234	104207	001373	MOV	#WRBLK,R0		:POINT TO COMMAND BLOCK
82	004236	104203	122400	MOV	#WRCMD,R3		:GET WRITE COMMAND
83	004240	104302	001565	MOV	CURTRK,R2		:GET CURRENT TRACK
84	004242	101023		BIS	R2,R3		:SET TRACK FOR WRITE
85	004243	100673	000005	MOV	R3,RW.CMD(R0)		:STORE IN COMMAND BLOCK
86	004245	104203	010000	MOV	#RDBUF,R3		:POINT TO BUFFER
87	004247	100673	000002	MOV	R3,RW.BUF(R0)		:STICK IN COMMAND BLOCK
88	004251	104303	001566	MOV	CURBN,R3		:GET LOW ORDER HEADER
89	004253	100673	000003	MOV	R3,RW.LOW(R0)		:STORE IN WRITE BLOCK
90	004255	104303	001567	MOV	CURBN+1,R3		:GET HIGH ORDER
91	004257	101203	120000	BIS	#HD.XBN,R3		:SET HEADER
92	004261	100673	000004	MOV	R3,RW.HI(R0)		:STORE IN WRITE BLOCK
93	004263	104203	001400	MOV	#HSLIM-1,R3		:GET DUMMY SDI POINTER
94	004265	100673	000006	MOV	R3,RW.DUM(R0)		:POINT IN COMMAND BLOCK
95	004267	104303	002005	WRITE1:	MOV	HPREA,R3	:GET HEADER PREAMBLE
96	004271	104304	002006	MOV	DPREA,R4		:GET DATA PREAMBLE
97	004273	104302	001412	MOV	UNIT,R2		:GET PORT NUMBER
98	004275	104207	001373	MOV	#WRBLK,R0		:MAKE SURE POINTING AT BLOCK
99	004277	101207	100000	BIS	#BIT15,R0		:SET NO REVECTORING
100	004301	060012		XFC	SIP		:WAIT FOR SECTOR PULSE
101	004302	104202	000400	MOV	#SECSI6,R2		:SECTOR SIZE IN WORDS
102	004304	060003		XFC	WRITE		:WRITE SECTOR
103	004305	115001		TST	R1		:ANY EPROR ?
104	004306			BEQ	NO		:NOPE
105	004310	106300	002165	002167	CMP	RETRY,TMPTRY	:MAX ?
106	004313			BEQ	1\$:YES - TRY SOME RECOVERY
107	004315	115400	002167	INC	TMPTRY		:INC RETRY COUNT
108	004317			BR	WRITE1		:DO RETRY
109	004321	104303	002170	1\$:	MOV	RECTMP,R3	:GET CURRENT ERROR RECOVERY LEVEL
110	004323			BMI	2\$:IF NEGATIVE THEN FRIED
111	004325	115000	002166	TST	RECOV		:IS THERE ONLY RECOVERY LEVEL 0 ?
112	004327			BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY
113	004331			CALL	ERRHND		:TRY RECOVERY
114	004333	114000	002167	3\$:	CLR	TMPTRY	:FOR INIT

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 65-2
 FCT DOWN-LINE LOAD OVERLAY (F3)

115	004335	117400	002170		DEC	RECTMP			:DECREMENT IT
116	004337				BR	WRITE1			:RETRY
117	004341			2\$:					
118	004341	115405			INC	R5			:YUP - INCREMENT COUNTER
119	004342	115400	001732	NO:	INC	NEXT1			:INCREMENT IT
120	004344	114000	002167		CLR	TMPTRY			:FOR RESET
121	004346	104300	002166	002170	MOV	RECOV,RECTMP			:GET RECOVERY LEVELS
122	004351	104204	001566		MOV	#CURBN,R4			:FOR ADD
123	004353	104203	001725		MOV	#FCTFMT,R3			:FOR ADD
124	004355				CALL	DADD			:POINT TO NEXT COPY
125	004357	106300	001731	001732	CMP	FCTCPY,NEXT1			:DONE THIS SECTOR ?
126	004362				BNE	LOOPP			:NO - WRITE NEXT FCT COPY
127	004364	106305	001731		CMP	FCTCPY,R5			:ERROR ON EVERY WRITE ?
128	004366				BEQ	ERROR			:YUP - BIG TROUBLE
129	004370	102200	002000	001702	BIT	#BSTGS,FLAG			:BEST GUESS ?
130	004373				BNE	DLLRET			:YUP - JUST WANT TO WRITE FIRST BLOCK
131	004375	102200	004000	001702	BIT	#NDLL,FLAG			:ALL DONE ???
132	004400				BNE	DLLRT1			:YUP - EXIT
133	004402	060022			XFC	UPDATE			:LET MOST KNOW STILL ALIVE
134	004403	115400	002164		INC	COUNT			:INCREMENT IT
135	004405				DUBINC	CURXBN			:INCREMENT IT
136	004413	104300	001572	001566	MOV	CURXBN,CURBN			:GET LOW ORDER
137	004416	104300	001573	001567	MOV	CURXBN+1,CURBN+1			:GET HIGH ORDER
138	004421	106300	001745	002164	CMP	FCTNPD,COUNT			:AT THE LAST NON-PAD ENTRY
139	004424				BNE	12\$:NOPE
140	004426	117400	001731		DEC	FCTCPY			:DECREMENT - NO PAD ON LAST COPY
141	004430	106300	001725	002164	12\$:	CMP	FCTFMT,COUNT		:DONE ?
142	004433				BNE	LOOP			:NOPE - DO NEXT SECTOR
143	004435	101200	000001	001702	DLLRT1:	BIS	#FCTAVL,FLAG		:SET FCT AVAILABLE
144	004440	104207	001512		MOV	#CR,R0			:POINT TO CHARACTERISTICS BLK
145	004442	104673	000001		MOV	FRCPY(R0),R3			:GET F/RCT COPIES
146	004444	110703			SWAB	R3			:GET INTO LOW BYTE
147	004445	103203	177760		BIC	#FCLR,R3			:CLEAR OUT REST OF GARBAGE
148	004447	104030	001731		MOV	R3,FCTCPY			:RESTORE NUM OF COPIES
149	004451	104201	000003		DLLRET:	MOV	#F2,R1		:FOR OVERLAY #2
150	004453				CALL	NEXT			:LBN FORMATTING
151	004455	104205	000400		DONDLL:	MOV	#SECS16,R5		:COUNT FOR BLOCK INIT
152	004457	104204	010000		MOV	#RDBUF,R4			:POINT TO BUFFER
153	004461	114003			CLR	R3			:INIT TO 0
154	004462	100243			LOOP3:	MOV	R3,(R4)+		:CLEAR ONE WORD
155	004463	117405			DEC	R5			:DEC COUNTER
156	004464				BNE	LOOP3			:CONTINUE TILL DONE
157	004466	101200	004000	001702	BIS	#NDLL,FLAG			:SET FLAG
158	004471				BR	LOOPP2			:CONTINUE
159	004473	104012			ERROR:	MOV	R1,R2		:GET XFC ERROR CODE
160	004474	104201	000015		MOV	#15,R1			:FCT WRITE ERROR
161	004476				DLERT:	CALL	ERRMNT		:ERROR RETURN


```

1
2 004500
3
4
5
6
7
8
9
10
11 004014 104200 000011 001636 RCTUPD: MOV #F4,CUROVL ;GET OVERLAY
12 004017 104303 001565 MOV CURTRK,R3 ;GET CURRENT TRACK
13 004021 PUSH R3 ;SAVE FOR RESTORE
14 004022 104303 001604 MOV CYLNUM,R3 ;GET LOW ORDRE CYLINDER
15 004024 PUSH R3 ;SAVE FOR RESTORE
16 004025 104303 001605 MOV CYLNUM+1,R3 ;GET HIGH ORDER
17 004027 PUSH R3 ;SAVE FOR RESTORE
18 004030 104300 001616 001712 MOV LBNLBN,HOLD ;GET LOW ORDER COUNT OF LBN'S
19 004033 104300 001617 001713 MOV LBNLBN+1,HOLD+1 ;GET HIGH ORDER
20 004036 104203 002160 MOV #TOTRCT,R3 ;FOR SUBTRACT
21 004040 104204 001712 MOV #HOLD,R4 ;DITTC
22 004042 CALL DSUB ;GET STARTING RCT LBN
23 004044 104300 001707 002157 MOV ERRBUF,UPDPNT ;POINT TO ERROR BUFFER
24 004047 104302 002157 ROVER: MOV UPDPNT,R2 ;GET POINTER TO BAD LIST
25 004051 104120 001403 MOV (R2),DDUMMY ;GET LOW ORDER
26 004053 104620 000001 001404 MOV 1(R2),DDUMMY+1 ;GET HIGH ORDER
27 004056 102200 100000 001404 BIT #BIT15,DDUMMY+1 ;IS IT AN RBN ??
28 004061 BEQ ROVER1 ;NO - REGULAR HASH
29 004063 104201 177777 MOV #-1,R1 ;HASH FOR RBN
30 004065 103200 170000 001404 ROVER1: BIC #HD.CLR,DDUMMY+1 ;CLEAR THE HEADER
31 004070 104204 001403 MOV #DDUMMY,R4 ;FOR HASH
32 004072 CALL UMASH ;FIND THE RCT ENTRY FOR CURRENT ERR BLOCK
33 004074 104143 MOV (R4),R3 ;GET BLOCK NUMBER
34 004075 105203 000002 ADD #2,R3 ;ADD TO GET PAST FIRST 2 BLOCKS
35 004077 100143 MOV R3,(R4) ;STORE BACK
36 004100 104030 002162 MOV R3,RCTCNT ;FOR LATER PING-PONG
37 004102 104203 001712 MOV #HOLD,R3 ;FOR ADD
38 004104 CALL DADD ;TO GET LBN OF RCT BLOCK
39 004106 104040 001740 MOV R4,BUFPNT ;STORE POINTER TO BLOCK NUMBER
40 004110 104201 000055 MOV #H1,R1 ;RCT READ OVERLAY
41 004112 CALL PAGE ;DO IT
42 004114 104205 013477 MOV #RCTBUF,R5 ;POINT TO BUFFER
43 004116 104303 001410 MOV OFFSET,R3 ;GET OFFSET
44 004120 105035 ADD R3,R5 ;POINT TO HIT ENTRY
45 004121 104302 002157 MOV UPDPNT,R2 ;RESTORE POINTER
46 004123 104623 000001 MOV 1(R2),R3 ;GET THE HEADER
47 004125 103203 007777 BIC #LO,R3 ;CLEAR ALL BUT HEADER
48 004127 106203 110000 CMP #HD.BAD,R3 ;IS IT A BAD RBN ?
49 004131 BNE NOTR ;NOPE - CHECK FOR PRIMARY
50 004133 104650 000000 001405 MOV 0(R5),TEMP2 ;GET LOW ORDER CURRENT RESIDENT
51 004136 104650 000001 001406 MOV 1(R5),TEMP2+1 ;GET HIGH ORDER
52 004141 103203 170000 BIC #H.CLR,R3 ;CLEAR HEADER
53 004143 101203 040000 BIS #RC.UNU,R3 ;MARK AS UNUSABLE
54 004145 103203 007777 BIC #LO,R3 ;CLEAR LOW ORDER
55 004147 100653 000001 MOV R3,1(R5) ;STORE IT BACK
56 004151 114003 CLR R3 ;CLEAR FOR STORE
57 004152 100153 MOV R3,(R5) ;CLEAR LOW ORDER
    
```

.SBTTL RCT UPDATE OVERLAY (F4)
 DMOVLY F4,START

THIS OVERLAY UPDATES THE RCT WITH THE SECTORS
 IN THE ERRBUF BUFFER

.....

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 66-1
 RCT UPDATE OVERLAY (F4)

58	004153	102200	020000	001406	BIT	#BIT13,TEMP2+1	;ANY THING DISPLACED ???
59	004156				BEQ	BOTTOM	;NO - NO NEED TO PING-PONG
60	004160				CALL	RCTWT	;WRITE UT BLOCK
61	004162				CALL	PNGPG	;FIND IT A NEW HOME
62	004164	104204	013477		MOV	#RCTBUF,R4	;POINT TO BUFFER
63	004166	105054			ADD	R5,R4	;ADD OFFSET
64	004167	104202	001405		MOV	#TEMP2,R2	;POINT TO OLD RESIDENT
65	004171	104123			MOV	(R2),R3	;GET LOW ORDER
66	004172	100143			MOV	R3,(R4)	;PUT IT IN
67	004173	104623	000001		MOV	1(R2),R3	;GET HIGH ORDER
68	004175	103203	170000		BIC	#HD.CLR,R3	;CLEAR HEADER
69	004177	101203	030000		BIS	#RC.SND,R3	;MARK AS SECONDARY
70	004201	100643	000001		MOV	R3,1(R4)	;STORE IT
71	004203				BR	BOTTOM	;GO TO BOTTOM OF LOOP
72	004205	106203	050000		NOTR: CMP	#HD.PRIV,R3	;PRIMARY REVECTOR ??
73	004207				BNE	SECNDY	;NO - TREAT AS SECONDARY
74	004211	104653	000001		MOV	1(R5),R3	;GET RCT HEADER
75	004213	103203	007777		BIC	#LO,R3	;CLEAR ALL BIT HEADER
76	004215	106203	000000		CMP	#RC.FRE,R3	;IS IT FREE ??
77	004217				BNE	SWAP	;NO - SWAP ENTRIES
78	004221	104123			MOV	(R2),R3	;GET LOW BLOCK NUMBER
79	004222	100153			MOV	R3,(R5)	;STORE IN RCT
80	004223	104623	000001		MOV	1(R2),R3	;GET HIGH ORDER
81	004225	107303	002007		SUB	ST.LBN,R3	;SUBTRACT STARTING LBN BITS
82	004227	103203	170000		BIC	#HD.CLR,R3	;CLEAR HEADER
83	004231	101203	020000		BIS	#RC.PRIV,R3	;SIGNAL PRIMARY REVECTOR IN RCT
84	004233	100653	000001		MOV	R3,1(R5)	;STORE IN RCT
85	004235				BR	BOTTOM	;GO TO BOTTOM OF LOOP
86	004237	104650	000000	001405	SWAP: MOV	0(R5),TEMP2	;GET LOW ORDER CURRENT RESIDENT
87	004242	104650	000001	001406	MOV	1(R5),TEMP2+1	;GET HIGH ORDER CURRENT RESIDENT
88	004245	104123			MOV	(R2),R3	;GET LOW ORDER NEW RESIDENT
89	004246	100153			MOV	R3,(R5)	;PUT IN RCT
90	004247	104623	000001		MOV	1(R2),R3	;GET HIGH ORDER NEW RESIDENT
91	004251	107303	002007		SUB	ST.LBN,R3	;SUBTRACT STARTING LBN BITS
92	004253	103203	170000		BIC	#HD.CLR,R3	;CLEAR THE HEADER
93	004255	101203	020000		BIS	#RC.PRIV,R3	;SET AS PRIMARY
94	004257	100653	000001		MOV	R3,1(R5)	;PUT IN RCT
95	004261				CALL	RCTWT	;WRITE OUT PRIMARY BLOCK
96	004263	104202	001405		MOV	#TEMP2,R2	;POINT TO OLD RESIDENT
97	004265				SECNDY: CALL	PNGPG	;FIND RCT ENTRY FOR SECONDARY
98	004267	104204	013477		MOV	#RCTBUF,R4	;POINT TO BUFFER
99	004271	105054			ADD	R5,R4	;ADD OFFSET
100	004272	104123			MOV	(R2),R3	;GET LOW ORDER NEW ENTRY
101	004273	100143			MOV	R3,(R4)	;PUT IN RCT
102	004274	104623	000001		MOV	1(R2),R3	;GET HIGH ORDER NEW ENTRY
103	004276	107303	002007		SUB	ST.LBN,R3	;SUBTRACT STARTING LBN BITS
104	004300	103203	170000		BIC	#HD.CLR,R3	;CLEAR HEADER
105	004302	101203	030000		BIS	#RC.SND,R3	;FLAG AS SECONDARY
106	004304	100643	000001		MOV	R3,1(R4)	;STORE IN RCT
107	004306				CALL	RCTWT	;WRITE OUT RCT BLOCK
108	004310	105200	000002	002157	ADD	#ERLEN,UPDPNT	;POINT TO NEXT ERROR SECOTR
109	004313	117400	001741		DEC	REVCNT	;DECREMENT IT
110	004315				BNE	ROVER	;NOT DONE - DO NEXT SECTOR
111	004317				POP	R3	;GET HIGH ORDER CYL
112	004320	104030	001605		MOV	R3,CYLNUM+1	;RESTORE IT
113	004322	104030	001552		MOV	R3,ISEEK+2	;PUT IN SEEK COMMAND
114	004324				POP	R3	;GET LOW ORDER

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 66-2
 RCT UPDATE OVERLAY (F4)

115	004325	104030	001604		MOV	R3,CYLNUM	
116	004327	104030	001551		MOV	R3,ISEEK+1	;PUT IN SEEK COMMAND
117	004331				POP	R3	;GET TRACK NUMBER
118	004332	104030	001565		MOV	R3,CURTRK	;RESTORE IT
119	004334	102200	040000	001702	BIT	#FINI,FLAG	;DO THE SEEK ?
120	004337				BNE	NOSEK	;NOPE
121	004341	104300	002146	001553	MOV	CURGRP,ISEEK+3	;RESTORE GROUP TO SEEK
122	004344	102200	020000	001703	BIT	#MODE,FLAG1	;ARE WE IN 576 MODE ?
123	004347				BEQ	2\$;NO - EVERYTHING FINE
124	004351	101200	100000	001552	BIS	#SS,ISEEK+2	;ELSE SET 576 MODE IN SEEK
125	004354				2\$:	CALL	SEEK
126	004356				NOSEK:	RETURN	;GET BACK TO RIGHT CYLINDER

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 67
 RCT UPDATE OVERLAY (F4)

```

1
2
3      :
4      :      COMPUTE RCT ADDRESS FOR GIVEN LBN
5      :
6      :
7      :
8      :
9      :
10     :
11     :
12     :
13     :
14     :
15     :
16     :
17     :
18     :
19     :
20     :
21     :
22     :
23     :

```

4	004360	115001			UHASH:	TST	R1		:NEED TO COMPUTE PRIMARY RBN ?
5	004361					BMI	UHKIP1		:NO - SKIP IT
6	004363	104207	001525			MOV	#SCR,R0		:POINT TO CUBUNIT CHARACTERISTICS
7	004365					CALL	PRIMRB		:COMPUTE PRIMARY RBN
8	004367	104200	000200	001410	UHKIP:	MOV	#128,TEMP		:DIVIDE BY 128 TO GET BLOCK NUMBER
9	004372	114000	001411			CLR	TEMP+1		:FOR STORE
10	004374	104300	001634	001403		MOV	REVRBN,DDUMMY		:GET PRIMARY RBN
11	004377	104300	001635	001404		MOV	REVRBN+1,DDUMMY+1		:GET HIGH ORDER
12	004402	104204	001403			MOV	#DDUMMY,R4		:FOR DIVIDE
13	004404	104203	001410			MOV	#TEMP,R5		:DITTO
14	004406					CALL	DDIV		:DDUMMY=RCT BLOCK NUMBER
15									:TEMP=OFFSET
16	004410	104131				MOV	(R3),R1		:GET OFFSET
17	004411	105011				ADD	R1,R1		:MULTIPLY BY 2
18	004412	100131				MOV	R1,(R3)		:STORE BACK
19	004413					RETURN			
20	004415	104140	001634		UHKIP1:	MOV	(R4),REVRBN		:FORDIVIDE SETUP
21	004417	104640	000001	001635		MOV	1(R4),REVRBN+1		:DITTC
22	004422	107300	002010	001635		SUB	ST.RBN,REVRBN+1		:SUBTRACT STARTING RBN BITS
23	004425					BR	UHKIP		:DO DIVIDE

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 69
 RCT UPDATE OVERLAY (F4)

```

1
2
3
4 004672          :
5 004673 114000 001705 : SEARCH FOR OPEN ENTRY IN RCT
6 004675 114002 :
7 004676 104303 001410 : PNGPG: PUSH R2 ;SAVE R2
8 004700 104035 : CLR WRFLG ;CLEAR WRAP FLAG
9 004701 105025 : XNGBLK: CLR R2 ;FOR FLOP SET
10 004702 102205 000400 : MOV OFFSET,R3 ;GET OFFSET
11 004704 : XAGAIN: MOV R3,R5 ;MOV OFFSET INTO BUFF POINTER
12 004706 104651 013500 : ADD R2,R5 ;ADD FLOP VALUE
13 004710 103201 007777 : BIT #BIT8,R5 ;PAST ONE END (OR BOTH)
14 004712 106201 100000 : BNE XFLIP ;YUP - FLIP OTHER DIRECTION
15 004714 : MOV RCTBUF+1(R5),R1 ;GET HEADER CODE
16 004716 106201 000000 : BIC #LO,R1 ;CLEAR LOW ORDER
17 004720 : CMP #RC.NUL,R1 ;END OF RCT ?
18 004722 104025 : BEQ XEORCT ;YUP - WRAP TO FIRST BLOCK
19 004723 114002 : CMP #RC.FRE,R1 ;FREE ?
20 004724 107052 : BEQ XPRET ;YUP - ALL DONE
21 004725 : XFLIP: MOV R2,R5 ;GET FLIP VALUE
22 004727 105202 000002 : CLR R2
23 004731 106202 000400 : SUB R5,R2 ;NEGATE IT
24 0C4733 : BMI XNOINC ;IF NEGATIVE DON'T INC
25 : ADD #2,R2 ;ADD TO NEXT VALUE
26 : XNOINC: CMP #SECS16,R2 ;DONE EVERY SLOT IN BLOCK ?
27 : BNE XAGAIN ;NOPE - TRY NEXT ONE
28
29
30 004735 115400 002162 :
31 004737 104303 002162 : IN THIS SECTION THE BLOCKS ARE PING-PONGED BUT
32 004741 104204 001403 : THE SEARCH WITHIN BLOCKS IS LINEAR FROM HIGHEST BUFFER
33 004743 100143 : ADDRESS TO LOWEST
34 004744 114003 :
35 004745 100643 000001 : XPNGRD: INC RCTCNT ;INC TO NEXT ONE
36 004747 104030 001410 : MOV RCTCNT,R3 ;FOR STORE
37 004751 104203 001712 : MOV #DDUMMY,R4 ;FOR ADD
38 004753 : MOV R3,(R4) ;STORE BLOCK NUMBER
39 004755 104040 001740 : CLR R3 ;FOR RESETS
40 004757 104201 000055 : MOV R3,1(R4) ;CLEAR HIGH WORD
41 004761 : MOV R3,OFFSET ;MAKE OFFSET AT BEGINNING
42 004763 : CALL #HOLD,R3 ;POINT TO FIRST RCT LBN
43 004765 104303 001705 : XEORCT: MOV RCTCNT,R3 ;GET LBN OF THIS RCT BLOCK
44 004767 : BNE XPERR ;STORE BLOCK NUMBER
45 004771 104200 000002 002162 : MOV #2,RCTCNT ;READ RCT OVERLAY
46 004774 104200 000002 001705 : MOV #2,WRFLG ;DO IT
47 004777 : BR XPNGRD ;SEARCH THIS BLOCK
48 : ;GET WRAP FLAG
49 : ;IF BEEN HERE ONCE THEN RCT FULL
50 005001 : XPRET: POP R2 ;RESTORE R2
51 005002 : RETURN ;SUCCESSFUL RETURN
52 005004 104201 000020 : XPERR: MOV #16.,R1 ;RCT FULL
53 005006 114002 : CLR R2 ;NO SUBCODE
54 005007 : CALL ERRMNT ;ERROR RETURN

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 70
RCT READ OVERLAY (H1)

1				.SBTTL RCT READ OVERLAY (H1)	
2	005011			DMOVL H1,SIART	
3				...	
4				READ A BLOCK OF THE RCT	
5				...	
6	004014			PUSHA	
7	004022	104200	000055	MOV #H1,CUROVL	;FOR INIT
8	004025	104304	001740	MOV BUFPNT,R4	;GET POINTER TO BLOCK NUMBER
9	004027	114005		CLR R5	;CLEAR ERROR COUNTER
10	004030	104203	001525	MOV #SCR,R3	;POINT TO CHARACTERISTICS
11	004032	104632	000001	MOV STCYL(R3),R2	;GET STARTING CYLINDER
12	004034	103202	007777	BIC #LO,R2	;CLEAR REST OF WORD
13	004036	104207	002110	MOV #CONBLK,R0	;POINT TO CONVERT BLOCK
14	004040	100672	000001	MOV R2,V1+1(R0)	;STORE FOR CONVERT
15	004042	114002		CLR R2	;FOR STORE
16	004043	100672	000000	MOV R2,V1(R0)	;LOW ORDER ALWAYS 0
17	004045	102200	020000	BIT #MODE,FLAG1	;WHAT MODE
18	004050			BNE 1\$;IF SET THEN 576
19	004052	104632	000011	MOV LBNT12(R3),R2	;GET LBN/TRACK FOR 512
20	004054			BR 2\$;SKIP 576 SETUP
21	004056	104632	000015	1\$: MOV LBNT76(R3),R2	;GET LBN/TRACK FOR 576
22	004060	103202	177400	2\$: BIC #HIBYTE,R2	;CLEAR HIGH BYTE
23	004062	100672	000004	MOV R2,V3(R0)	;STORE IN CONVERT BLOCK
24	004064			OCLOOP:	
25	004064			CALL CVTSK	;CONVERT RCT BLOCK NUMBER AND SEEK
26	004066	104207	001373	MOV #RDBLK,R0	;PREPARE FOR READ SECTORS
27	004070	104203	001400	MOV #HSLIM-1,R3	;POINTER TO DUMMY SDI BLOCK
28	004072	100673	000006	MOV R3,RW.DUM(R0)	;STORE IN COMMAND BLOCK
29	004074	104143		MOV (R4),R3	;LO ORDER BLOCK NUMBER
30	004075	100673	000003	MOV R3,RW.LOW(R0)	;STORE IN READ CMD BLOCK
31	004077	104643	000001	MOV 1(R4),R3	;HI ORDER BLOCK NUM AND CODE
32	004101	105303	002007	ADD ST.LBN,R3	;ADD STARTING LBN BITS
33	004103	100673	000004	MOV R3,RW.HI(R0)	;STORE IN READ CMD BLOCK
34	004105	104203	013477	MOV #RCTBUF,R3	;LOAD ADDRESS OF DATA BUFFER
35	004107	100673	000002	MOV R3,RW.BUF(R0)	;STORE IN COMMAND BUFFER
36	004111	104203	013400	MOV #RWCMD,R3	;LOAD SDI READ COMMAND
37	004113	104301	001565	MOV CURTRK,R1	;GET CURRENT HEAD NUMBER IN R1
38	004115	101013		BIS R1,R3	;SET IT IN COMMAND
39	004116	100673	000005	MOV R3,RW.CMD(R0)	;STORE BACK
40	004120	104207	001373	READ11: MOV #RDBLK,R0	;MAKE SURE POINTING AT BLOCK
41	004122	104203	100000	MOV #RDCMD,R3	;MARK AS ONLY REQUEST
42	004124	100173		MOV R3,(R0)	;STORE IN CMD BLOCK
43	004125	104302	001412	MOV UNIT,R2	;GET PORT NUMBER
44	004127	101207	100000	BIS #BIT15,R0	;SET NO REVECTORING
45	004131	060012		XFC SIP	;WAIT FOR PULSE
46	004132	104302	002171	MOV SECSIZ,R2	;SECTOR SIZE IN WORDS
47	004134	060002		XFC READ	;READ 1 SECTOR
48	004135	115001		TST R1	;ANY ERRORS ?
49	004136			BNE 100\$;YES - TRY RECOVERY
50	004140	104673	000001	MOV RW.ER1(R0),R3	;GET ECC STATUS WORD
51	004142	102203	010000	BIT #ECCF,R3	;ECC ERROR ?
52	004144			BEQ 101\$;NONE - VERIFY EDC
53	004146	103200	010000	BIC #ECCF,RDBLK+RW.ER1	;CLEAR ECC ERROR BIT
54	004151			CALL ECCCK	;CORRECT ECC
55	004153	115001		TST R1	;TEST FLAG
56	004154			BNE 100\$;UNCORRECTABLE
57	004156	104202	013477	101\$: MOV #RCTBUF,R2	;POINT TO BUFFER

UDAF52 - UGA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 70-1
 RCT READ OVERLAY (H1)

58	004160	104307	002171		MOV	SECSIZ,R0		:SECTOR SIZE IN WORDS
59	004162				CALL	CEDC		:COMPUTE EDC
60	004164	102200	020000	001703	BIT	#MODE,FLAG1		:WHAT MODE ARE WE IN
61	004167				BEQ	4\$:IF CLEAR THEN 512
62	004171	106623	000440		CMP	RW.E76(R2),R3		:O.K. ?
63	004173				BEQ	102\$:YUP - CONSIDER GOOD
64	004175				BR	100\$:NOPE - RETRY
65	004177	106623	000400	4\$:	CMP	RW.EDC(R2),R3		:O.K. ?
66	004201				BEQ	102\$:YUP - CONSIDER GOOD
67	004203	106300	002165	002167	100\$:	CMP	RETRY, TMPTRY	:MAX ?
68	004206				BEQ	1\$:YES - TRY SOME RECOVERY
69	004210	115400	02167		INC	TMPTRY		:INC RETRY COUNT
70	004212				BR	READ11		:DO RETRY
71	004214	104303	002170	1\$:	MOV	RECTMP,R3		:GET CURRENT ERROR RECOVERY LEVEL
72	004216				BMI	2\$:IF NEGATIVE THEN FRIED
73	004220	115000	002166		TST	RECOV		:IS THERE ONLY RECOVERY LEVEL 0 ?
74	004222				BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY
75	004224				CALL	ERRHND		:TRY RECOVERY
76	004226	114000	002167	3\$:	CLR	TMPTRY		:FOR INIT
77	004230	117400	002170		DEC	RECTMP		:DECREMENT IT
78	004232				BR	READ11		:RETRY
79	004234			2\$:				
80	004234	115405			INC	R5		:INCREMENT BAD COUNTER
81	004235	106305	001731		CMP	FCTCPY,R5		:ALL BAD ?
82	004237				BEQ	ORFTAL		:YUP - ALL OVER
83	004241	104203	001727		MOV	#RCTFMT,R3		:SIZE OF TABLE - R4 -> BLOCK NUMBER
84	004243				CALL	DADD		:ADD TO POINT TO NEXT COPY
85	004245	114000	002167		CLR	TMPTRY		:RESET RETRY COUNT
86	004247	104300	002166	002170	MOV	RECOV,RECTMP		:DITTO RECOVERY LEVELS
87	004252				BR	OCLOOP		:BRANCH BACK
88	004254			102\$:				
89	004254	114000	002167	002170	OCDONE:	CLR	TMPTRY	:FOR RESET
90	004256	104300	002166		MOV	RECOV,RECTMP		:GET RECOVERY LEVELS
91	004261	115005			TST	R5		:ANY ERRORS ?
92	004262				BEQ	RLDONE		:NO - EXIT
93	004264	104203	001727		MOV	#RCTFMT,R3		:SIZE OF TABLE
94	004266				CALL	DSUB		:GET BACK TO PREVIOUS COPY
95	004270				CALL	CVTSK		:CONVERT AND SEEK
96	004272	104207	001373		MOV	#WRBLK,R0		:POINT TO COMMAND BLOCK
97	004274	104203	122400		MOV	#WRCMD,R3		:GET WRITE COMMAND
98	004276	104302	001565		MOV	CURTRK,R2		:GET CURRENT TRACK
99	004300	101023			BIS	R2,R3		:SET TRACK FOR WRITE
100	004301	100673	000005		MOV	R3,RW.CMD(R0)		:STORE IN COMMAND BLOCK
101	004303	104203	013477		MOV	#RCTBUF,R3		:POINT TO BUFFER
102	004305	100673	000002		MOV	R3,RW.BUF(R0)		:STICK IN COMMAND BLOCK
103	004307	104143			MOV	(R4),R3		:GET LOW ORDER HEADER
104	004310	100673	000003		MOV	R3,RW.LOW(R0)		:STORE IN WRITE BLOCK
105	004312	104643	000001		MOV	1(R4),R3		:GET HIGH ORDER
106	004314	105303	002007		ADD	ST.LBN,R3		:ADD STARTING LBN BITS
107	004316	100673	000004		MOV	R3,RW.HI(R0)		:STORE IN WRITE BLOCK
108	004320	104203	001400		MOV	#HSLIM-1,R3		:GET DUMMY SDI POINTER
109	004322	100673	000006		MOV	R3,RW.DUM(R0)		:POINT IN COMMAND BLOCK
110	004324	104303	002005	WRIT12:	MOV	HPREA,R		:GET HEADER PREAMBLE
111	004326	104304	002006		MOV	DPREA,R4		:GET DATA PREAMBLE
112	004330	104302	001412		MOV	UNIT,R2		:GET PORT NUMBER
113	004332	104207	001373		MOV	#WRBLK,R0		:MAKE SURE POINTING AT BLOCK
114	004334	101207	100000		BIS	#BIT15,R0		:SET NO REVECTORING


```
1  
2  
3  
4  
5  
6  
7 004421  
8  
9  
10  
11 004014
```

```
.....  
      .SBTTL  FCT->RCT CONVERSION OVERLAY (F5)  
      CONVERT FCT INTO RCT  
      DMOVLY  F5,START  
      .....  
      JMP     START2                ;SKIP SUBROUTINES
```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 72
 FCT->RCT CONVERSION OVERLAY (F5)

1											
2											
3											
4	004016	115001									
5	004017										
6	004021	104207	001525								
7	004023										
8	004025	104200	000200	001410	HKIP:						
9	004030	114000	001411								
10	004032	104300	001634	001403							
11	004035	104300	001635	001404							
12	004040	104204	001403								
13	004042	104203	001410								
14	004044										
15											
16	004046	104131									
17	004047	105011									
18	004050	100131									
19	004051										
20	004053	104140	001634		HKIP1:						
21	004055	104640	000001	001635							
22	004060	107300	002010	001635							
23	004063										

COMPUTE RCT ADDRESS FOR GIVEN LBN

```

:
:
HASH:  TST      R1                ;NEED TO COMPUTE PRIMARY RBN ?
        BMI     HKIP1            ;NO - SKIP IT
        MOV     #SCR,R0          ;POINT TO CUBUNIT CHARACTERISTICS
        CALL    PRIMRB           ;COMPUTE PRIMARY RBN
        MOV     #128,TEMP        ;DIVIDE BY 128 TO GET BLOCK NUMBER
        CLR     TEMP+1           ;FOR STORE
        MOV     REVRBN,DDUMMY     ;GET PRIMARY RBN
        MOV     REVRBN+1,DDUMMY+1 ;GET HIGH ORDER
        MOV     #DDUMMY,R4       ;FOR DIVIDE
        MOV     #TEMP,R3         ;DITTO
        CALL    DDIV             ;DDUMMY=RCT BLOCK NUMBER
        ;TEMP=OFFSET
        MOV     (R3),R1          ;GET OFFSET
        ADD     R1,R1            ;MULTIPLY BY 2
        MOV     R1,(R3)         ;STORE BACK
        RETURN
HKIP1:  MOV     (R4),REVRBN       ;FGRDIVIDE SETUP
        MOV     1(R4),REVRBN+1   ;DITTC
        SUB     ST.RBN,REVRBN+1  ;SUBTRACT STARTING RBN BITS
        BR      HKIP            ;DO DIVIDE

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 73
 FCT->RCT CONVERSION OVERLAY (F5)

```

1
2
3          :
4          : SEARCH FOR OPEN ENTRY IN RCT
5          :
6          :
7          :
8          :
9          :
10         :
11         :
12         :
13         :
14         :
15         :
16         :
17         :
18         :
19         :
20         :
21         :
22         :
23         :
24         :
25         :
26         :
27         :
28         :
29         :
30         :
31         :
32         :
33         :
34         :
35         :
36         :
37         :
38         :
39         :
40         :
41         :
42         :
43         :
44         :
45         :
46         :
47         :
48         :
49         :
50         :
51         :
52         :

```

4	004065	114000	001705	PNGPNG:	CLR	WRFLG		:CLEAR WRAP FLAG
5	004067	114002		PNGBLK:	CLR	R2		:FOR FLOP SET
6	004070	104303	001410		MOV	OFFSET,R3		:GET OFFSET
7	004072	104035		PAGAIN:	MOV	R3,R5		:MOV OFFSET INTO BUFF POINTER
8	004073	105025			ADD	R2,R5		:ADD FLOP VALUE
9	004074	102205	000400		BIT	#BIT8,R5		:PAST ONE END (OR BOTH)
10	004076				BNE	FLIP		:YUP - FLIP OTHER DIRECTION
11	004100	104651	013500		MOV	RCTBUF+1(R5),R1		:GET HEADER CODE
12	004102	103201	007777		BIC	#LO,R1		:CLEAR LOW ORDER STUFF
13	004104	106201	100000		CMP	#RC.NUL,R1		:END OF RCT ?
14	004106				BEQ	EORCT		:YUP - WRAP TO FIRST BLOCK
15	004110	106201	000000		CMP	#RC.FRE,R1		:FREE ?
16	004112				BEQ	PRET		:YUP - ALL DONE
17	004114	104025		FLIP:	MOV	R2,R5		:GET FLIP VALUE
18	004115	114002			CLR	R2		
19	004116	107052			SUB	R5,R2		:NEGATE IT
20	004117				BMI	NOINC		:IF NEGATIVE DON'T INC
21	004121	105202	000002		ADD	#2,R2		:ADD TO NEXT VALUE
22	004123	106202	000400	NOINC:	CMP	#SECS16,R2		:DONE EVERY SLOT IN BLOCK ?
23	004125				BNE	PAGAIN		:NOPE - TRY NEXT ONE
24								
25								
26								
27								
28								
29	004127	115400	002162		INC	RCTCNT		:INC TO NEXT ONE
30	004131	104303	002162	PNGRD:	MOV	RCTCNT,R3		:FOR STORE
31	004133	104204	001403		MOV	#DDUMMY,R4		:FOR ADD
32	004135	100143			MOV	R3,(R4)		:STORE BLOCK NUMBER
33	004136	114000	001404		CLR	DDUMMY+1		:FOR RESETS
34	004140	114000	001410		CLR	OFFSET		:MAKE IT AT ZERO
35	004142	104203	001712		MOV	#HOLD,R3		:POINT TO FIRST RCT LBN
36	004144				CALL	DADD		:GET LBN OF THIS RCT BLOCK
37	004146	104040	001740		MOV	R4,BUFPNT		:STORE POINTER TO BLOCK NUMBER
38	004150	104201	000055		MOV	#H1,R1		:RCT READ OVERLAY
39	004152				CALL	PAGE		:DO IT
40	004154				BR	PNGBLK		:SEARCH THIS BLOCK
41	004156	104303	001705	EORCT:	MOV	WRFLG,R3		:GET WRAP FLAG
42	004160				BNE	PERR		:IF BEEN HERE ONCE THEN RCT FULL
43	004162	104200	000002 002162		MOV	#2,RCTCNT		:FOR FIRST RCT BLOCK
44	004165	104200	000002 001705		MOV	#2,WRFLG		:MAKE WRAP FLAG NON-ZERO
45	004170				BR	PNGRD		:READ IT AND CONTINUE
46								
47								
48	004172			PRET:	RETURN			:SUCCESSFUL RETURN
49								:R5=OFFSET
50	004174	104201	000020	PERR:	MOV	#16.,R1		:RCT FULL
51	004176	114002			CLR	R2		:NO SUBCODE
52	004177				CALL	ERRMNT		:ERROR RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 74
 FCT->RCT CONVERSION OVERLAY (F5)

```

1
2
3
4
5
6
7 004201
8 004201 104200 000014 001636
9 004204 104303 001565
10 004206
11 004207 104303 001604
12 004211
13 004212 104303 001605
14 004214
15 004215 104204 001743
16 004217 104203 002151
17 004221
18 004223 104143
19 004224
20 004225 114000 001743
21 004227 104200 010000 001740
22 004232 104201 000017
23 004234
24 004236 104204 010000
25 004240 102200 020000 001703
26 004243
27 004245 104640 000020 001747
28 004250
29 004252 104640 000016 001747 1$:
30 004255 104200 000200 002137 2$:
31 004260 104300 001616 001712
32 004263 104300 001617 001713
33 004266 104203 002160
34 004270 104204 001712
35 004272
36 004274 104201 000022
37 004276
38 004300 104303 001747
39 004302
40 004304 102200 020000 001703
41 004307
42 004311 105300 001723 001743
43 004314 104200 010000 001740 FBEGIN:
44 004317 104201 000017
45 004321
46 004323 104200 010000 001742
47 004326 104304 001742
48 004330 104140 001563 FBEG2:
49 004332 104640 000001 001564
50 004335 104201 000044
51 004337
52 004341 104303 001567
53 004343 103203 007777
54 004345 106203 000000
55 004347
56 004351 104203 001566
57 004353 104204 001576
    
```

THIS OVERLAY UPDATES THE RCT WITH THE SECTORS
 IN THE ERRBUF BUFFER

```

START2:
FCTRCT: MOV #F5,CUROVL ;GET OVERLAY NUMBER
MOV CURTRK,R3 ;GET CURRENT TRACK
PUSH R3 ;SAVE IT
MOV CYLNUM,R3 ;GET LOW ORDER CYLINDR
PUSH R3 ;SAVE FOR RESTORE
MOV CYLNUM+1,R3 ;GET HIGH ORDER
PUSH R3 ;SAVE FOR RESTORE
MOV #FCTCNT,R4 ;FOR SUB
MOV #ONE,R3 ;DITTO
CALL DSUB ;SUB TO GET CURRENT FT BLOCK NUM
MOV (R4),R3 ;GET IT
PUSH R3 ;FOR LATER RESTORE
CLR FCTCNT ;CLEAR FOR INIT
MOV #RDBUF,BUFPT ;POINT TO BUFFER
MOV #F6,R1 ;READ A BLOCK OF THE FCT
CALL PAGE ;EXECUTE IT
MOV #RDBUF,R4 ;MAKE SURE POINT TO IT
BIT #MODE,FLAG1 ;WHAT MODE ARE WE IN
BEQ 1$ ;IF CLEAR THEN 512 MODE
MOV C576(R4),MNCNT ;GET COUNT OF USED ENTRIES +576)
BR 2$ ;SKIP 512 STUFF
1$: MOV C512(R4),MNCNT ;GET COUNT OF USED ENTRIES
2$: MOV #128,,SECCNT ;ENTRIES IN A FCT SECTOR
MOV LBNLBN,HOLD ;GET LOW ORDER COUNT OF LBN'S
MOV LBNLBN+1,HOLD+1 ;GET HIGH ORDER
MOV #TOTRCT,R3 ;FOR SUBTRACT
MOV #HOLD,R4 ;DITTO
CALL DSUB ;GET STARTING RCT LBN
MOV #F7,R1 ;RCT INIT OVERLAY
CALL PAGE ;INIT RCT
MOV MNCNT,R3 ;GET COUNT
BEQ FCTSP ;QUIT IF NO ENTRIES
BIT #MODE,FLAG1 ;WHAT MODE ARE WE IN
BEQ FBEGIN ;512 MODE - NO ADJUSTMENT
ADD FCTSUB,FCTCNT ;POINT TO 576 TABLE
FBEGIN: MOV #RDBUF,BUFPT ;POINT TO BUFFER
MOV #F6,R1 ;FCT READ OVERLAY
CALL PAGE ;DO IT
MOV #RDBUF,FCTPTR ;MAKE SURE POINT TO TI
MOV FCTPTR,R4 ;FOR USE
FBEG2: MOV (R4),CURPBN ;GET LOW ORDER PBN
MOV 1(R4),CURPBN+1 ;GET HIGH ORDER
MOV #G5,R1 ;OVERLAY TO CONVERT FORM PBN TO OTHER BN
CALL PAGE ;EXECUTE IT
MOV CURBN+1,R3 ;GET HIGH ORDER CONVERTED BLOCK NUM
BIC #LO,R3 ;CLEAR ALL BUT HEADER
CMP #HD,LBN,R3 ;IS IT AN LBN ?
BNE NOLBN ;NO - SKIP DOWN
MOV #CURBN,R3 ;FOR COMAPRE
MOV #HOLDBN,R4 ;DITTO
    
```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 74-1
 FCT->RCT CONVERSION OVERLAY (F5)

58	004355			CALL	DCMP			:IN RCT ??
59	004357			BPL	XYZ1			:YUP
60	004361	104204	001566	MOV	#CURBN,R4			:POINT TO BLOCK NUMBER
61	004363			CALL	HASH			:COMPUTE RCT ENTRY
62	004365	104143		MOV	(R4),R3			:GET RCT BLOCK
63	004366	105203	000002	ADD	#2,R3			:ADD TO GET BY FIRST 2 BLOCKS
64	004370	100143		MOV	R3,(R4)			:STORE BACK
65	004371	104030	002162	MOV	R3,RCTCNT			:SAVE FOR LATER PNGPNG
66	004373	104203	001712	MOV	#HOLD,R3			:FOR ADD
67	004375			CALL	DADD			:TO GET LBN OF RCT BLOCK
68	004377	104040	001740	MOV	R4,BUFPT			:STORE POINTER TO BLOCK NUMBER
69	004401	104201	000055	MOV	#H1,R1			:RCT READ OVERLAY
70	004403			CALL	PAGE			:EXECUTE IT
71	004405	104205	013477	MOV	#RCTBUF,R5			:POINT TO BUFFER
72	004407	104303	001410	MOV	OFFSET,R3			:GET OFFSET
73	004411	105035		ADD	R3,R5			:POINT TO ENTRY
74	004412	104653	000001	MOV	1(R5),R3			:GET HIGH ORDER
75	004414	103203	007777	BIC	#LO,R3			:CLEAR ALL BIT HEADER
76	004416	106203	000000	CMF	#RC.FRE,R3			:IS IT FREE ?
77	004420			BEQ	FILLIT			:YES - FILL IT
78	004422	106203	040000	CMF	#RC.UNU,R3			:UNUSABLE RBN ?
79	004424			BEQ	BADRBN			:YES - MUST BE SECONDARY
80	004426	104150	001405	MOV	(R5),TEMP2			:ELSE SWITCH
81	004430	104650	000001	MOV	1(R5),TEMP2+1	001406		:HIGH ORDER
82	004433	104303	001566	MOV	CURBN,R3			:GET NEW RESIDENT LOW ORDER
83	004435	100153		MOV	R3,(R5)			:STORE IN RCT
84	004436	104303	001567	MOV	CURBN+1,R3			:GET HIGH ORDER
85	004440	107303	002007	SUB	ST.LBN,R3			:SUBTRACT STARTING LBN BITS
86	004442	103203	170000	BIC	#HD.CLR,R3			:CLEAR THE HEADER
87	004444	101203	020000	BIS	#RC.PRIV,R3			:MARK AS PRIMARY
88	004446	100653	000001	MOV	R3,1(R5)			:STORE IT
89	004450	102200	000400	BIT	#DLL,FLAG	001702		:DID WE CREATE THE FCT ?
90	004453			BEQ	FCTSKP			:NO - THEN DON'T CHANGE IT
91	004455	104302	001742	MOV	FCTPTR,R2			:GET POINTER TO CURRENT FCT BLOCK POS
92	004457	104623	000001	MOV	1(R2),R3			:GET HIGH ORDER FCT ENTRY
93	004461	101203	100000	BIS	#PRIV,R3			:MAKE IT SECONDARY
94	004463	100623	000001	MOV	R3,1(R2)			:STORE IT BACK
95	004465	106202	010000	CMF	#RDBUF,R2			:IS THIS THE FIRST ENTRY IN THE BLOCK
96	004467			BNE	FCTSK1			:NOPE WE'RE SAFE
97	004471	104300	001403	MOV	DDUMMY,CURXBN	001572		:SAVE RCT BLOCK NUMBER
98	004474	104300	001404	MOV	DDUMMY+1,CURXBN+1	001573		:DITTO
99	004477			CALL	FIXFCT			:YUP - GOT SOME GYRATIONS TO DO
100	004501	104300	001572	MOV	CURXBN,DDUMMY	001403		:RESTORE RCT BLOCK NUMBER
101	004504	104300	001573	MOV	CURXBN+1,DDUMMY+1	001404		:DITTO
102	004507			BR	FCTSKP			:THEN CONTINUE ON
103	004511	107202	000001	FCTSK1: SUB	#1,R2			:POINT BACK ONE
104	004513	104123		MOV	(R2),R3			:GET HIGH ORDER
105	004514	103203	100000	BIC	#PRIV,R3			:CLEAR PRIMARY IF SET
106	004516	100123		MOV	R3,(R2)			:STORE IT
107	004517			FCTSKP: CALL	RCTWRT			:WRITE OUT (CHANGED) BLOCK
108	004521			CALL	PNGPNG			:FIND IT A NEW HOME
109	004523	104204	013477	MOV	#RCTBUF,R4			:POINT TO BUFFER
110	004525	105054		ADD	R5,R4			:ADD OFFSET
111	004526	104202	001405	MOV	#TEMP2,R2			:POINT TO OLD RESIDENT
112	004530	104123		MOV	(R2),R3			:GET LOW ORDER
113	004531	100143		MOV	R3,(R4)			:PUT IT IN
114	004532	104623	000001	MOV	1(R2),R3			:GET HIGH ORDER

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 74-2
 FCT->RCT CONVERSION OVERLAY (F5)

115	004534	103203	170000		BIC	#HD.CLR,R3	:CLEAR HEADER	
116	004536	101203	030000		BIS	#RC.SND,R3	:MARK AS SECONDARY	
117	004540	100643	000001		MOV	R3,1(R4)	:STORE IT	
118	004542				BR	XYZ	:SKIP TO END	
119	004544	106203	060000	NOLBN:	CMP	#HD.RBN,R3	:BAD RBN ?	
120	004546				BNE	XYZ1	:NO - THEN DON'T CARE ABOUT IT	
121	004550	104303	001567		MOV	CURBN+1,R3	:GET HEADER	
122	004552	103203	170000		BIC	#HD.CLR,R3	:CLEAR IT	
123	004554	104030	001567		MOV	R3,CURBN+1	:STORE IT BACK	
124	004556	104201	177777		MOV	#-1,R1	:SIGNAL RCT BLOCK	
125	004560	104204	001566		MOV	#CURBN,R4	:POINT TO BLOCK NUMBER	
126	004562				CALL	HASH	:GET RCT BLOCK AND OFFSET	
127	004564	104143			MOV	(R4),R3	:GET RCT BLOCK	
128	004565	105203	000002		ADD	#2,R3	:ADD TO GET BY 2 BLOCKS	
129	004567	100143			MOV	R3,(R4)	:STORE BACK	
130	004570	104030	002162		MOV	R3,RCTCNT	:SAVE FOR LATER PNGPNG	
131	004572	104203	001712		MOV	#HOLD,R3	:FOR ADD	
132	004574				CALL	DADD	:TO GET LBN OF RCT BLOCK	
133	004576	104040	001740		MOV	R4,BUFPT	:STORE POINTER TO BLOCK NUMBER	
134	004600	104201	000055		MOV	#H1,R1	:RCT READ OVERLAY	
135	004602				CALL	PAGE	:DO IT	
136	004604	104205	013477		MOV	#RCTBUF,R5	:POINT TO BLOCK	
137	004606	104304	001410		MOV	OFFSET,R4	:GET OFFSET	
138	004610	105045			ADD	R4,R5	:POINT TO ENTRY	
139	004611	104653	000001		MOV	1(R5),R3	:GET HIGH ORDER	
140	004613	103203	007777		BIC	#LO,R3	:CLEAR ALL BUT HEADER	
141	004615	106203	000000		CMP	#RC.FRE,R3	:IS IT FREE ?	
142	004617				BNE	RRPL	:NO - RELOCATE CURRENT RESIDENT	
143	004621	103203	170000		BIC	#HD.CLR,R3	:CLEAR THE HEADER	
144	004623	101203	040000		BIS	#RC.UNU,R3	:MARK AS UNUSABLE	
145	004625	100653	000001		MOV	R3,1(R5)	:STORE IT BACK	
146	004627				BR	XYZ	:BRANCH TO THE END	
147	004631	104650	000000	001405	RRPL:	MOV	0(R5),TEMP2	:GET LOW ORDER CURRENT RESIDENT
148	004634	104650	000001	001406		MOV	1(R5),TEMP2+1	:GET HIGH ORDER
149	004637	103203	170000		BIC	#HD.CLR,R3	:CLEAR HEADER	
150	004641	101203	040000		BIS	#RC.UNU,R3	:MARK AS UNUSABLE	
151	004643	103203	007777		BIC	#LO,R3	:CLEAR LOW ORDER	
152	004645	100653	000001		MOV	R3,1(R5)	:STORE IT BACK	
153	004647	114003			CLR	R3	:CLEAR FOR STORE	
154	004650	100153			MOV	R3,(R5)	:CLEAR LOW ORDER	
155	004651				CALL	RCTWRT	:WRITE UT BLOCK	
156	004653				CALL	PNGPNG	:FIND IT A NEW HOME	
157	004655	104204	013477		MOV	#RCTBUF,R4	:POINT TO BUFFER	
158	004657	105054			ADD	R5,R4	:POINT TO ENTRY	
159	004660	104202	001405		MOV	#TEMP2,R2	:POINT TO OLD RESIDENT	
160	004662	104123			MOV	(R2),R3	:GET LOW ORDER	
161	004663	100143			MOV	R3,(R4)	:PUT IT IN	
162	004664	104623	000001		MOV	1(R2),R3	:GET HIGH ORDER	
163	004666	103203	170000		BIC	#HD.CLR,R3	:CLEAR HEADER	
164	004670	101203	030000		BIS	#RC.SND,R3	:MARK AS SECONDARY	
165	004672	100643	000001		MOV	R3,1(R4)	:STORE IT	
166	004674	102200	000400	001702	BIT	#DLL,FLAG	:DID WE CREATE THE FCT ?	
167	004677				BEQ	FCTSLP	:NO - THEN DON'T CHANGE IT	
168	004701	104302	001742		MOV	FCTPTR,R2	:GET FCT PPOINTER	
169	004703	107202	000001		SUB	#1,R2	:POINT BACK ONE	
170	004705	104123			MOV	(R2),R3	:GET HIGH ORDER	
171	004706	103203	100000		BIC	#PRMY,R3	:CLEAR PRIMARY IF SET	

UDAF52 - UDA-52 FORMATTFR DMACR X04.01 23-AUG-82 13:14:22 PAGE 74-3
 FCT->RCT CONVERSION OVERLAY (F5)

172	004710	100123			MOV	R3,(R2)		:STORE IT
173	004711				FCTSLP: BR	XYZ		:GO TO END
174	004713				BADRBN: CALL	PNGPNG		:FIND A NEW SLOT
175	004715	104204	013477		MOV	#RCTBUF,R4		:POINT TO BUFFER
176	004717	105054			ADD	R5,R4		:POINT TO ENTRY
177	004720	104202	001566		MOV	#CURBN,R2		:POINT TO OLD RESIDENT
178	004722	104123			MOV	(R2),R3		:GET LOW ORDER
179	004723	100143			MOV	R3,(R4)		:PUT IT IN
180	004724	104623	000001		MOV	1(R2),R3		:GET HIGH ORDER
181	004726	103203	170000		BIC	#HD.CLR,R3		:CLEAR HEADER
182	004730	101203	030000		BIS	#RC.SND,R3		:MARK AS SECONDARY
183	004732	100643	000001		MOV	R3,1(R4)		:STORE IT
184	004734				BR	XYZ		:GO TO END
185	004736	104303	001566		FILLIT: MOV	CURBN,R3		:GET LOW ORDER BN
186	004740	100153			MOV	R3,(R5)		:PUT IN RCT
187	004741	104303	001567		MOV	CURBN+1,R3		:GET HIGH ORDER AND HEADER
188	004743	107303	002007		SUB	ST.LBN,R3		:SUBTRACT STARTING LBN BITS
189	004745	103203	170000		BIC	#HD.CLR,R3		:CLEAR HEADER
190	004747	101203	020000		BIS	#RC.PRIV,R3		:MARK AS PRIMARY
191	004751	100653	000001		MOV	R3,1(R5)		:STORE IN RCT
192	004753	102200	000400	001702	BIT	#DLL,FLAG		:DID WE CREATE THE FCT ?
193	004756				BEQ	XYZ		:NO - THEN DON'T CHANGE IT
194	004760	104303	001742		MOV	FCTPTR,R3		:GET POINTER TO FCT ENTRY
195	004762	104634	000001		MOV	1(R3),R4		:GET HIGH ORDER
196	004764	101204	100000		BIS	#PRIV,R4		:SET AS PRIMARY
197	004766	100634	000001		MOV	R4,1(R3)		:STORE BACK
198	004770				XYZ: CALL	RCTWRT		:WRITE OUT BUFFER
199	004772	105200	000002	001742	XYZ1: ADD	#2,FCTPTR		:POINT TO TO NEXT ENTRY
200	004775	117400	001747		DEC	MNCNT		:DEREMENT IT
201	004777				BEQ	FRDONE		:IF ZERO THEN DONE
202	005001	117400	002137		DEC	SECCNT		:DECREMENT IT
203	005003				BNE	FRSKP		:IF STILL IN BLOCK - CONTINUE
204	005005	102200	000400	001702	BIT	#DLL,FLAG		:DID WE CREATE THE FCT ?
205	005010				BEQ	FBEGIN		:NO - THEN DON'T CHANGE IT
206	005012	104200	010000	001740	MOV	#RDBUF,BUFPT		:POINT TO BUFFER
207	005015	104201	000030		MOV	#F9,R1		:FCT WRITE OVERLAY
208	005017				CALL	PAGE		:EXECUTE IT
209	005021				BR	FBEGIN		:AND GET A NEW ONE
210	005023	104304	001742		FRSKP: MOV	FCTPTR,R4		:ELSE GET CURRENT POINTER
211	005025	060022			XFC	UPDATE		:LET HOST KNOW STILL ALIVE
212	005026				BR	FBEG2		:AND DO NEXT ENTRY
213	005030	102200	000400	001702	FRDONE: BIT	#DLL,FLAG		:DID WE CREATE THE FCT ?
214	005033				BEQ	FCTSP		:NO - THEN DON'T CHANGE IT
215	005035	104200	010000	001740	MOV	#RDBUF,BUFPT		:POINT TO BUFFER
216	005040	104201	000030		MOV	#F9,R1		:WRITE LAST FCT BLOCK
217	005042				CALL	PAGE		:DO IT
218	005044				FCTSP: POP	R3		:GET CURRENT FCT BLOCK NUM
219	005045	104030	001743		MOV	R3,FCTCNT		:RESTORE IT
220	005047	104200	010455	001740	MOV	#PBNBUF,BUFPT		:RE-READ IT IN CASE OF HEADER CHANGES
221	005052	104201	000017		MOV	#F6,R1		:FCT READ OVERLAY
222	005054				CALL	PAGE		:DO IT
223	005056				POP	R3		:GET HIGH ORDER CYLINDER
224	005057	104030	001605		MOV	R3,CYLNUM+1		:STORE IT
225	005061	104030	001552		MOV	R3,ISEEK+2		:STORE IN SEEK COMMAND
226	005063				POP	R3		:GET LOW ORDER
227	005064	104030	001604		MOV	R3,CYLNUM		:RESTORE IT
228	005066	104030	001551		MOV	R3,ISEEK+1		:STORE IN SEEK COMMAND

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 74-4
 FCT->RCT CONVERSION OVERLAY (F5)

```

229 005070          POP      R3          ;GET CURRENT TRACK
230 005071 104030 001565      MOV      R3,CURTRK      ;RESTORE IT
231 005073 104300 002146 001553  MOV      CURGRP,ISEEK+3 ;RESTORE GROUP
232 005076 102200 020000 001703  BIT      #MODE,FLAG1    ;ARE WE IN 576 MODE ?
233 005101          BEQ      2$          ;NO - EVERYTHING FINE
234 005103 101200 100000 001552  BIS      #SS,ISEEK+2    ;ELSE SET 576 MODE IN SEEK
235 005106          CALL     SEEK          ;RESTORE TO PREVIOUS CYLINDER
236 005110          RETURN
237
238
239          :
240          :
241          :
242          :
243 005112 104204 001743      FIXFCT: MOV      #FCTCNT,R4      ;FOR SUBTRACT
244 005114 104203 002153      MOV      #TWOC,R3      ;DOUBLE WORD OF 2
245 005116          CALL     DSUB          ;SUBTRACT TO GET BACK TO RIGHT NUM
246 005120 104200 015763 001740  MOV      #IMAGE,BUFPT  ;USE IMAGE BUFFER
247 005123 104201 000017      MOV      #F6,R1        ;FCT READ OVERLAY
248 005125          CALL     PAGE          ;READ THE BLOCK
249 005127 104203 015763      MOV      #IMAGE,R3     ;POINT TO BUFFER
250 005131 105203 000377      ADD      #255.,R3      ;POINT TO LAST ENTRY
251 005133 104632 000001      MOV      1(R3),R2      ;GET HIGH ORDER
252 005135 103202 100000      BIC      #PRMY,R2      ;CLEAR FLAG
253 005137 100632 000001      MOV      R2,1(R3)      ;STORE IT BACK
254 005141 104200 015763 001740  MOV      #IMAGE,BUFPT  ;STORE BUFFER POINTER
255 005144 104201 000030      MOV      #F9,R1        ;FCT WRITE OVERLAY
256 005146          CALL     PAGE          ;WRITE IT BACK OUT
257 005150          DUBINC  FCTCNT      ;GET FCTCNT BACK TO NORMAL
258 005156          RETURN

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 75
 FCT->RCT CONVERSION OVERLAY (F5)

1				WRITE AN RCT BLOCK	
2				DDUMMY = BLOCK NUMBER TO BE WRITTEN	
3					
4					
5	005160	114005		RCTWRT: CLR R5	:CLEAR ERROR COUNTER
6	005161	104050	001732	MOV R5,NEXT1	:INIT NEXT COPY COUNTER
7	005163	104204	001403	MOV #DDUMMY,R4	:POINT TO RCT LBN
8	005165	104203	001525	RCTRLP: MOV #SCR,R3	:POINT TO CHARACTERISTICS
9	005167	104207	002110	MOV #CONBLK,R0	:POINT TO CONVERT BLOCK
10	005171	102200	020000	001703 BIT #MODE,FLAG1	:WHAT MODE
11	005174			BNE 1\$:IF SET THEN 576
12	005176	104632	000011	MOV LBNT12(R3),R2	:GET LBN/TRACK FOR 512
13	005200			BR 2\$:SKIP 576 SETUP
14	005202	104632	000015	1\$: MOV LBNT76(R3),R2	:GET LBN/TRACK FOR 576
15	005204	103202	177400	2\$: BIC #HIBYTE,R2	:CLEAR HIGH BYTE
16	005206	100672	000004	MOV R2,V3(R0)	:FOR CONVERT
17	005210	104632	000001	MOV STCYL(R3),R2	:STARTING CLYLINDER
18	005212	103202	007777	BIC #LO,R2	:CLEAR REST OF WORD
19	005214	100672	000001	MOV R2,V1+1(R0)	:STORE
20	005216	114002		CLR R2	:FOR STORE
21	005217	100672	000000	MOV R2,V1(R0)	:LOW ORDER ALWAYS 0
22	005221			CALL CVTSK	:CONVERT AND SEEK
23	005223	104202	013477	MOV #RCTBUF,R2	:POINT TO BUFFER
24	005225	104307	002171	MOV SECSIZ,R0	:SECTOR SIZE IN WORDS
25	005227			CALL CEDC	:COMPUTE EDC - RETURNED IN R3
26	005231	102200	020000	001703 BIT #MODE,FLAG1	:WHAT MODE ARE WE IN
27	005234			BEQ 3\$:IF CLEAR THEN 512
28	005236	100623	000440	MOV R3,RW.E76(R2)	:STORE IT 576 BUFFER
29	005240			BR 4\$:SKIP 512 SETUP
30	005242	100623	000400	3\$: MOV R3,RW.EDC(R2)	:STORE IT 512 BUFFER
31	005244	104207	001373	4\$: MOV #WRBLK,R0	:POINT TO COMMAND BLOCK
32	005246	100672	000002	MOV R2,RW.BUF(R0)	:STICK BUFFER PTR IN COMMAND BLOCK
33	005250	104203	122400	MOV #WRCMD,R3	:GET WRITE COMMAND
34	005252	104302	001565	MOV CURTRK,R2	:GET CURRENT TRACK
35	005254	101023		BIS R2,R3	:SET TRACK FOR WRITE
36	005255	100673	000005	MOV R3,RW.CMD(R0)	:STORE IN COMMAND BLOCK
37	005257	104143		MOV (R4),R3	:GET LOW ORDER HEADER
38	005260	100673	000003	MOV R3,RW.LOW(R0)	:STORE IN WRITE BLOCK
39	005262	104643	000001	MOV 1(R4),R3	:GET HIGH ORDER
40	005264	105303	002007	ADD ST.LBN,R3	:ADD STARTING LBN BITS
41	005266	101203	000000	BIS #HD.LBN,R3	:SET HEADER
42	005270	100673	000004	MOV R3,RW.HI(R0)	:STORE IN WRITE BLOCK
43	005272	104203	001400	MOV #HSLIM-1,R3	:GET DUMMY SDI POINTER
44	005274	100673	000006	MOV R3,RW.DUM(R0)	:POINT IN COMMAND BLOCK
45	005276	104303	002005	WRITE3: MOV HPREA,R3	:GET HEADER PREAMBLE
46	005300	104304	002006	MOV DPREA,R4	:GET DATA PREAMBLE
47	005302	104302	001412	MOV UNIT,R2	:GET PORT NUMBER
48	005304	104207	001373	MOV #WRBLK,R0	:MAKE SURE POINTING AT BLOCK
49	005306	101207	100000	BIS #BIT15,R0	:SET NO REVECTORING
50	005310	060012		XFC SIP	:WAIT FOR SECTOR PULSE
51	005311	104302	002171	MOV SECSIZ,R2	:SECTOR SIZE IN WORDS
52	005313	060003		XFC WRITE	:WRITE SECTOR
53	005314	115001		TST R1	:ANY ERROR ?
54	005315			BEQ RWGOOD	:NOPE
55	005317	106300	002165	002167 CMP RETRY,TMPTRY	:MAX ?
56	005322			BEQ 1\$:YES - TRY SOME RECOVERY
57	005324	115400	002167	INC TMPTRY	:INC RETRY COUNT

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 76
 RCT INITIALIZE OVERLAY (F7)

```

1          .SBTTL RCT INITIALIZE OVERLAY (F7)
2 005424   DMOVLY F7,START
3          :
4          :
5          :
6 004014   104200 000014 001636 RCTINI: MOV    #F5,CUROVL      ;FOR OVERLAY INIT
7 004017   104303 002171          MOV    SECSIZ,R3      ;SECTOR WORD COUNT
8 004021   114001          CLR    R1             ;FOR INIT OF RCT WORDS
9 004022   104204 010000          MOV    #RDBUF,R4     ;BUFFER
10 004024   100241          RCLP: MOV    R1,(R4)+   ;STORE IN BUFFER
11 004025   117403          DEC    R3            ;DECREMENT COUNTER
12 004026          BNE    RCLP          ;BRANCH BACK TILL DONE
13 004030   104201 001772          MOV    #SERNUM,R1    ;POINT TO SERIAL NUMBER
14 004032   104204 010000          MOV    #RDBUF,R4     ;POINT TO BUFFER
15 004034   105204 000090          ADD    #RSER,R4     ;POINT TO SERIAL NUMBER
16 004036   104205 000004          MOV    #4,R5         ;COUNTER
17 004040   104212          6$:  MOV    (R1)+,R2     ;GET WORD
18 004041   100242          MOV    R2,(R4)+     ;STORE WORD
19 004042   117405          DEC    R5            ;DECREMENT COUNTER
20 004043          BNE    6$           ;CONTINUE TILL DONE
21 004045   104303 001565          MOV    CURTRK,R3     ;GET CURRENT TRACK
22 004047          PUSH   R3            ;SAVE IT
23 004050   104303 001604          MOV    CYLNUM,R3     ;GET LOW ORDER CYLINDR
24 004052          PUSH   R3            ;SAVE FOR RESTORE
25 004053   104303 001605          MOV    CYLNUM+1,R3   ;GET HIGH ORDER
26 004055          PUSH   R3            ;SAVE FOR RESTORE
27 004056   104300 001620 001410  MOV    RBNLBN,TEMP   ;GET NUMBER OF RBN'S IN LBN AREA
28 004061   104300 001621 001411  MOV    RBNLBN+1,TEMP+1 ;HIGH ORDER
29 004064   104204 001410          MOV    #TEMP,R4      ;FOR ADD
30 004066   104200 000400 001403  MOV    #256,,DDUMMY  ;2 BLOCKS(CONTROL) WORTH OF RBN'S
31 004071   114000 001404          CLR    DDUMMY+1     ;CLEAR HIGH ORDER
32 004073   104203 001403          MOV    #DDUMMY,R3   ;FOR ADD
33 004075          CALL   DADD          ;ADD TO GET 'REAL' NUMBER OF RBN'S
34 004077   104300 001616 001712  MOV    LBNLBN,HOLD   ;GET LOW ORDER COUNT OF LBN'S
35 004102   104300 001617 001713  MOV    LBNLBN+1,HOLD+1 ;GET HIGH ORDER
36 004105   104203 002160          MOV    #TOTRCT,R3   ;FOR SUBTRACT
37 004107   104204 001712          MOV    #HOLD,R4     ;DITTO
38 004111          CALL   DSUB          ;GET STARTING RCT LBN
39 004113   104300 001712 001566  MOV    HOLD,CURBN   ;GET STARTING RCT BLOCK NUMBER
40 004116   104300 001712 001570  MOV    HOLD,CURLBN  ;ALSO SAVE
41 004121   104300 001713 001567  MOV    HOLD+1,CURBN+1 ;GET HIGH ORDER
42 004124   104300 001713 001571  MOV    HOLD+1,CURLBN+1 ;AND SAVE
43 004127   114000 002164          CLR    COUNT        ;CLEAR BLOCK COUNTER
44 004131   104203 001525          MOV    #SCR,R3      ;POINT TO CHARACTERISTICS
45 004133   104207 002110          MOV    #CONBLK,R0   ;POINT TO CONVERT BLOCK
46 004135   102200 020000 001703  BIT    #MODE,FLAG1  ;WHAT MODE
47 004140          BNE    1$           ;IF SET THEN 576
48 004142   104632 000011          MOV    LBNT12(R3),R2 ;GET LBN/TRACK FOR 512
49 004144          BR     2$           ;SKIP 576 SETUP
50 004146   104632 000015          1$:  MOV    LBNT76(R3),R2 ;GET LBN/TRACK FOR 576
51 004150   103202 177400          2$:  BIC    #HIBYTE,R2   ;CLEAR HIGH BYTE
52 004152   100672 000004          MOV    R2,V3(R0)    ;FOR CONVERT
53 004154   104632 000001          MOV    STCYL(R3),R2 ;STARTING CYLINDER
54 004156   103202 007777          BIC    #LO,R2       ;CLEAR REST OF WORD
55 004160   100672 000001          MOV    R2,V1+1(R0) ;STORE
56 004162   114002          CLR    R2           ;FOR STORE
57 004163   100672 000000          MOV    R2,V1(R0)   ;LOW ORDER ALWAYS 0

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 76-1
 RCT INITIALIZE OVERLAY (F7)

58	004165	114005		RCINLP:	CLR	R5	:	CLEAR ERROR COUNTER
59	004166	104050	001732		MOV	R5,NEXT1	:	INIT COPY COUNT
60	004170	104204	001566	RCLP2:	MOV	#CURBN,R4	:	FOR CONVERT
61	004172				CALL	CVTSK	:	CONVERT AND SEEK
62	004174	104202	010000		MOV	#RDBUF,R2	:	POINT TO BUFFER
63	004176	104307	002171		MOV	SECSIZ,R0	:	SECTOR SIZE IN WORDS
64	004200				CALL	CEDC	:	COMPUTE EDC - RETURNED IN R3
65	004202	102200	020000	001703	BIT	#MODE,FLAG1	:	WHAT MODE ARE WE IN
65	004205				BEQ	3\$:	IF CLEAR THEN 512
67	004207	100623	000440		MOV	R3,RW.E76(R2)	:	STORE IT 576 BUFFER
68	004211				BR	4\$:	SKIP 512 SETUP
69	004213	100623	000400	3\$:	MOV	R3,RW.EDC(R2)	:	STORE IT 512 BUFFER
70	004215	104207	001373	4\$:	MOV	#WRBLK,R0	:	POINT TO COMMAND BLOCK
71	004217	100672	000002		MOV	R2,R!.BUF(R0)	:	STICK BUFFER PTR IN COMMAND BLOCK
72	004221	104203	122400		MOV	#WRCMD,R3	:	GET WRITE COMMAND
73	004223	104302	001565		MOV	CURTRK,R2	:	GET CURRENT TRACK
74	004225	101023			BIS	R2,R3	:	SET TRACK FOR WRITE
75	004226	100673	000005		MOV	R3,RW.CMD(R0)	:	STORE IN COMMAND BLOCK
76	004230	104303	001566		MOV	CURBN,R3	:	GET LOW ORDER HEADER
77	004232	100673	000003		MOV	R3,RW.LOW(R0)	:	STORE IN WRITE BLOCK
78	004234	104303	001567		MOV	CURBN+1,R3	:	GET HIGH ORDER
79	004236	105303	002007		ADD	ST.LBN,R3	:	ADD STARTING LBN BITS
80	004240	101203	000000		BIS	#HD.LBN,R3	:	SET HEADER
81	004242	100673	000004		MOV	R3,RW.HI(R0)	:	STORE IN WRITE BLOCK
82	004244	104203	001400		MOV	#HSLIM-1,R3	:	GET DUMMY SDI POINTER
83	004246	100673	000006		MOV	R3,RW.DUM(R0)	:	POINT IN COMMAND BLOCK
84	004250	104303	002005	WRITE4:	MOV	HPREA,R3	:	GET HEADER PREAMBLE
85	004252	104304	002006		MOV	DPREA,R4	:	GET DATA PREAMBLE
86	004254	104302	001412		MOV	UNIT,R2	:	GET PORT NUMBER
87	004256	104207	101373		MOV	#<WRBLK!BIT15>,R0	:	MAKE SURE POINTING AT BLOCK
88	004260	060012			XFC	SIP	:	WAIT FOR SECTOR PULSE
89	004261	104302	002171		MOV	SECSIZ,R2	:	SECTOR SIZE IN WORDS
90	004263	060003			XFC	WRITE	:	WRITE SECTOR
91	004264	115001			TST	R1	:	ANY ERROR ?
92	004265				BEQ	NOGOOD	:	NOPE
93	004267	106300	002165	002167	CMP	RETRY,IMPTRY	:	MAX ?
94	004272				BEQ	1\$:	YES - TRY SOME RECOVERY
95	004274	115400	002167		INC	IMPTRY	:	INC RETRY COUNT
96	004276				BR	WRITE4	:	DO RETRY
97	004300	104303	002170	1\$:	MOV	RECTMP,R3	:	GET CURRENT ERROR RECOVERY LEVEL
98	004302				BMI	2\$:	IF NEGATIVE THEN FRIED
99	004304	115000	002166		TST	RECOV	:	IS THERE ONLY RECOVERY LEVEL 0 ?
100	004306				BEQ	3\$:	YES - NO NEED TO ISSUE IT - JUST RETRY
101	004310				CALL	ERRHND	:	TRY RECOVERY
102	004312	114000	002167	3\$:	CLR	IMPTRY	:	FOR INIT
103	004314	117400	002170		DEC	RECTMP	:	DECREMENT IT
104	004316				BR	WRITE4	:	RETRY
105	004320			2\$:			:	
106	004320	115405			INC	R5	:	YUP - INCREMENT COUNTER
107	004321	115400	001732		INC	NEXT1	:	INCREMENT IT
108	004323	114000	002167		CLR	IMPTRY	:	FOR RESET
109	004325	104300	002166	002170	MOV	RECOV,RECTMP	:	GET RECOVERY LEVELS
110	004330	104204	001566		MOV	#CURBN,R4	:	FOR ADD
111	004332	104203	001727		MOV	#RCTFMT,R3	:	FOR ADD
112	004334				CALL	DADD	:	POINT TO NEXT COPY
113	004336	106300	001731	001732	CMP	FCTCPY,NEXT1	:	DONE THIS SECTOR ?
114	004341				BNE	RCLP2	:	NO - WRITE NEXT FCT COPY

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 76-2
 RCT INITIALIZE OVERLAY (F7)

115	004343	106305	001731		CMP	FCTCPY,R5	:ERROR ON EVERY WRITE ?
116	004345				BEQ	RCINER	:YUP - BIG TROUBLE
117	004347	102200	000040	001702	BIT	#RCINIT,FLAG	:ALREADY FIXED IT UP
118	004352				BNE	4\$:YUP - NO NEED TO DO IT AGAIN
119	004354	104204	001410		MOV	#TEMP,R4	:FOR SUBTRACT (RBN'S NOT DONE)
120	004356	104200	000200	001403	MOV	#128.,DDUMMY	:SUBTRACT ONE BLOCKS WORTH
121	004361	114000	001404		CLR	DDUMMY+1	:FOR CLEAR
122	004363	104203	001403		MOV	#DDUMMY,R3	:FOR SUBTRACT
123	004365				CALL	DSUB	:SUBTRACT
124	004367				CALL	DCMP	:IN LAST BLOCK ?
125	004371				BMI	4\$:NOPE
126	004373				BEQ	4\$:IF EQUAL - NO PARTIAL BLOCK
127	004375				CALL	FIXBLK	:YES - CHANGE HEADERS TO NULL
128	004377	115000	002164	4\$:	TST	COUNT	:ON FIRST BLOCK ?
129	004401				BNE	7\$:NO - NO NEED TO FIX UP
130	004403	114005			CLR	R5	:FOR BLOCK FIXUP
131	004404	104204	010000		MOV	#RDB'IF,R4	:POINT TO BUFFER
132	004406	104201	000004		MOV	#4,R1	:COUNTER
133	004410	100245		5\$	MOV	R5,(R1)+	:CLEAR DATE AREA
134	004411	117401			DEF	R1	:DECREMENT COUNTER
135	004412				BNE	5\$:CONT TILL DONE
136	004414	102200	000100	001703	BIT	#P.INDN,FLAG1	:ALL DONE ??
137	004417				BNE	RCLP6	:YUP - CUT OUT
138	004421				DUBINC	CURLBN	:INCREMENT IT
139	004427	104300	001570	001566	MOV	CURLBN,CURBN	:GET LOW ORDER
140	004432	104300	001571	001567	MOV	CURLBN+1,CURBN+1	:GET HIGH ORDER
141	004435	115400	002164		INC	COUNT	:INCREMENT BLOCK COUNTER
142	004437	106300	001746	002164	CMP	RCTLBN,COUNT	:DONE RCT BLOCKS (NOT PAD)
143	004442				BEQ	RCFIX	:YUP - REINIT BLOCK
144	004444	060022			XFC	UPDATE	:LET HOST KNOW STILL ALIVE
145	004445	106300	001727	002164	CMP	RCTFMT,COUNT	:DONE ?
146	004450				BNE	RCINLP	:NOPE - DO NEXT SECTOR
147	004452				RCLP6:	POP	:GET HIGH ORDER CYLINDER
148	004453	104030	001605		MOV	R3,CYLNUM+1	:STORE IT
149	004455	104030	001552		MOV	R3,ISEEK+2	:STORE IN SEEK COMMAND
150	004457				POP	R3	:GET LOW ORDER
151	004460	104030	001604		MOV	R3,CYLNUM	:RESTORE IT
152	004462	104030	001551		MOV	R3,ISEEK+1	:STORE IN SEEK COMMAND
153	004464				POP	R3	:GET CURRENT TRACK
154	004465	104030	001565		MOV	R3,CURTRK	:RESTORE IT
155	004467	104300	002146	001553	MOV	CURGRP,ISEEK+3	:RESTORE GROUP NUMBER
156	004472	102200	020000	001703	BIT	#MODE,FLAG1	:ARE WE IN 576 MODE ?
157	004475				BEQ	2\$:NO - EVERYTHING FINE
158	004477	101200	100000	001552	BIS	#SS,ISEEK+2	:ELSE SET 576 MODE IN SEEK
159	004502				CALL	SEEK	:RESTORE TO PREVIOUS CYLINDER
160	004504				RETURN		
161	004506	104202	000200		RCFIX:	MOV	#128.,R2
162	004510	104204	010000		MOV	#RDBUF,R4	:INIT COUNT
163	004512	114003			CLR	R3	:INIT POINTER
164	004513	114005			CLR	R5	:FOR STORE
165	004514	101205	100000		BIS	#RC.NUL,R5	:DITTO
166	004516	100243			RCLP3:	MOV	R3,(R4)+
167	004517	100245			MOV	R5,(R4)+	:SET NULL HEADER
168	004520	117402			DEC	R2	:STORE LOW ORDER
169	004521				BNE	RCLP3	:STORE HIGH ORDER
170	004523	101200	000100	001703	BIS	#RCINDN,FLAG1	:DECREMENT COUNTER
171	004526				BR	RCLP4	:LOOP UNTIL DONE
							:DONE ALL NON-PAD - ONE MORE THEN FINISH
							:CONTINUE WITH LAST SECTOR

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 76-3
RCT INITIALIZE OVERLAY (F7)

172 004530 104012
173 004531 104201 000017
174 004533

RCINER: MOV R1,R2
MOV #15.,R1
CALL ERRMNT

:XFC ERROR CODE
:RCT INIT ERROR
:ERROR RETURN

JDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 78
 FCT READ OVERLAY (F6)

1					.SBTTL FCT READ OVERLAY (F6)	
2	004564				DMOVLY F6,START	
3						
4					READ A BLOCK OF THE FCT	
5					R5 -> BUFFER	
6						
7	004C14	104200	000017	001636	MOV #F6,CUROVL	:OVERLAY NUMBER
8	004017	104300	001743	001403	MOV FCTCNT,DDUMMY	:GET CURRENT COUNT
9	004022	114000	001404		CLR DDUMMY+1	:FOR HIGH ORDER STORE
10	004024	114005			CLR R5	:CLEAR ERROR COUNTER
11	004025	104204	001403		FOLOOP: MOV #DDUMMY,R4	:FOR CONVERT
12	004027	104303	001626		MOV LBNCYL,R3	:GET LBN CYLINDERS
13	004031	104207	002110		MOV #CONBLK,R0	:POINT TO CONVERT BLOCK
14	004033	100673	000000		MOV R3,V1(R0)	:STORE IT FOR CONVERT
15	004035	104303	001627		MOV LBNCYL+1,R3	:HIGH ORDER
16	004037	100673	000001		MOV R3,V1+1(R0)	:STORE IT
17	004041	104303	001612		MOV SECT12,R3	:GET SECTORS/TRACK (512)
18	004043	100673	000004		MOV R3,V3(R0)	:STORE FOR CONVERT
19	004045				CALL CVTSK	:CONVERT FCT BLOCK NUMBER AND SEEK
20	004047	104207	001373		MOV #RDBLK,R0	:PREPARE FOR READ SECTORS
21	004051	104203	001400		MOV #HSLIM-1,R3	:POINTER TO DUMMY SDI BLOCK
22	004053	100673	000006		MOV R3,RW.DUM(R0)	:STORE IN COMMAND BLOCK
23	004055	104303	001403		MOV DDUMMY,R3	:LO ORDER BLOCK NUMBER
24	004057	100673	000003		MOV R3,RW.LOW(R0)	:STORE IN READ CMD BLOCK
25	004061	104303	001404		MOV DDUMMY+1,R3	:GET HIGH ORDER
26	004063	105303	002011		ADD ST.XBN,R3	:ADD STARTING LBN BITS
27	004065	101203	120000		BIS #HD.XBN,R3	:HEADER CODE
28	004067	100673	000004		MOV R3,RW.HI(R0)	:STORE IN READ CMD BLOCK
29	004071	104303	001740		MOV BUFPT,R3	:GET BUFFER POINTER
30	004073	100673	000002		MOV R3,RW.BUF(R0)	:STORE BUFFER ADDRESS IN COMMAND BUFFER
31	004075	104203	013400		MOV #RWCMD,R3	:LOAD SDI READ COMMAND
32	004077	104301	001500		MOV CURTRK,R1	:GET CURRENT HEAD NUMBER IN R1
33	004101	101013			BIS R1,R3	:SET IT IN COMMAND
34	004102	100673	000005		MOV R3,RW.CMD(R0)	:STORE BACK
35	004104	104207	001370		READ7: MOV #RDBLK,R0	:MAKE SURE POINTING AT BLOCK
36	004106	104203	100000		MOV #RDCMD,R3	:MARK AS ONLY REQUEST
37	004110	104302	001412		MOV UNIT,R2	:GET PORT NUMBER
38	004112	100173			MOV R3,(R0)	:STORE IN CMD BLOCK
39	004113	101207	100000		BIS #BIT15,R0	:SET NO REVECTING
40	004115	060012			XFC SIP	:WAIT FOR PULSE
41	004116	104202	000400		MOV #SECSI6,R2	:SECTOR SIZE IN WORDS
42	004120	060002			XFC READ	:READ 1 SECTOR
43	004121	115001			TST R1	:ANY ERRORS ?
44	004122				BNE 100\$:YES - TRY RECOVERY
45	004124	104673	000001		MOV RW.ER1(R0),R3	:GET ECC STATUS WORD
46	004126	102203	010000		BIT #ECCF,R3	:ECC ERROR ?
47	004130				BEQ 101\$:NOPE - VERIFY EDC
48	004132	103200	010000	001374	BIC #ECCF,RDBLK+RW.ER1	:CLEAR ECC ERROR BIT
49	004135				CALL ECCCK	:CORRECT ECC
50	004137	115001			TST R1	:TEST FLAG
51	004140				BNE 100\$:UNCORRECTABLE
52	004142	104302	001740		101\$: MOV BUFPT,R2	:POINT TO BUFFER
53	004144	104207	000400		MOV #SECSI6,R0	:SECTOR SIZE IN WORDS
54	004146				CALL CEDC	:COMPUTE EDC
55	004150	106623	000400		CMP RW.EDC(R2),R3	:O.K. ?
56	004152				BEQ 102\$:YUP - CONSIDER GOOD
57	004154	106300	002165	002167	100\$: CMP RETRY,IMPTRY	:MAX ?

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 78-1
 FCT READ OVERLAY (F6)

58	004157			BEQ	1\$:YES - TRY SOME RECOVERY
59	004161	115400	002167	INC	TMPTRY				:INC RETRY COUNT
60	004163			BR	READ7				:DO RETRY
61	004165	104303	002170	1\$:	MOV	RECTMP,R3			:GET CURRENT ERROR RECOVERY LEVEL
62	004167			BMI	2\$:IF NEGATIVE THEN FRIED
63	004171	115000	002166	TST	RECOV				:IS THERE ONLY RECOVERY LEVEL 0 ?
64	004173			BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY
65	004175			CALL	ERRHND				:TRY RECOVERY
66	004177	114000	002167	3\$:	CLR	TMPTRY			:FOR INIT
67	004201	117400	002170	DEC	RECTMP				:DECREMENT IT
68	004203			BR	READ7				:RETRY
69	004205			2\$:					
70	004205	115405		INC	R5				:INCREMENT BAD COUNTER
71	004206	106305	001731	CMP	FCTCPY,R5				:ALL BAD ?
72	004210			BEQ	OFATAL				:YUP - ALL OVER
73	004212	104204	001405	MOV	#DDUMMY,R4				:POINT TO COUNT
74	004214	104203	001725	MOV	#FCTFMT,R3				:SIZE OF TABLE
75	004216			CALL	DADD				:ADD TO POINT TO NEXT COPY
76	004220	114000	002167	CLR	TMPTRY				:RESET RETRY LEVEL
77	004222	104300	002166	002170	MOV	RECOV,RECTMP			:DITTO RECOVERY LEVELS
78	004225			BR	FOLOOP				:BRANCH BACK
79	004227			102\$:					
80	004227	114000	002167	FODONE:	CLR	TMPTRY			:FOR RESET
81	004231	104300	002166	002170	MOV	RECOV,RECTMP			:GET RECOVERY LEVELS
82	004234	115005		TST	R5				:ANY ERRORS ?
83	004235			BEQ	OLDONE				:NO - EXIT
84	004237	104204	001403	MOV	#DDUMMY,R4				:POINT TO BLOCK COUNT
85	004241	104203	001725	MOV	#FCTFMT,R3				:SIZE OF TABLE
86	004243			CALL	DSUB				:GET BACK TO PREVIOUS COPY
87	004245			CALL	CVTSK				:CONVERT AND SEEK
88	004247	104207	001373	MOV	#WRBLK,R0				:POINT TO COMMAND BLOCK
89	004251	104203	122400	MOV	#WRCMD,R3				:GET WRITE COMMAND
90	004253	104302	001565	MOV	CURTRK,R2				:GET CURRENT TRACK
91	004255	101023		BIS	R2,R3				:SET TRACK FOR WRITE
92	004256	100673	000005	MOV	R3,RW.CMD(R0)				:STORE IN COMMAND BLOCK
93	004260	104303	001740	MOV	BUFPT,R3				:GET BUFFER ADDRESS
94	004262	100673	000002	MOV	R3,RW.BUF(R0)				:STICK IN COMMAND BLOCK
95	004264	104303	001403	MOV	DDUMMY,R3				:GET LOW ORDER HEADER
96	004266	100673	000003	MOV	R3,RW.LOW(R0)				:STORE IN WRITE BLOCK
97	004270	104303	001404	MOV	DDUMMY+1,R3				:GET HIGH ORDER
98	004272	105303	002011	ADD	ST.XBN,R3				:ADD STARTING XBN BITS
99	004274	101203	120000	BIS	#HD.XBN,R3				:HEADER CODE
100	004276	100673	000004	MOV	R3,RW.HI(R0)				:STORE IN WRITE BLOCK
101	004300	104203	001400	MOV	#HSLIM-1,R3				:GET DUMMY SDI POINTER
102	004302	100673	000006	MOV	R3,RW.DUM(R0)				:POINT IN COMMAND BLOCK
103	004304	104303	002005	WRITE8:	MOV	HPREA,R3			:GET HEADER PREAMBLE
104	004306	104304	002006	MOV	DPREA,R4				:GET DATA PREAMBLE
105	004310	104302	001412	MOV	UNIT,R2				:GET PORT NUMBER
106	004312	104207	001373	MOV	#WRBLK,R0				:MAKE SURE POINTING AT BLOCK
107	004314	101207	100000	BIS	#BIT15,R0				:SET NO REVECTORING
108	004316	060012		XFC	SIP				:WAIT FRO SECTOR PULSE
109	004317	104202	000400	MOV	#SECSI6,R2				:SECTOR SIZE IN WORDS
110	004321	060003		XFC	WRITE				:WRITE SECTOR
111	004322	115001		TST	R1				:ANY ERROR ?
112	004323			BEQ	2\$:NO - SKIP RETRY
113	004325	106300	002165	002167	CMP	RETRY,TMPTRY			:MAX ?
114	004330			BEQ	1\$:YES - TRY SOME RECOVERY

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 78-2
 FCT READ OVERLAY (F6)

115	004332	115400	002167		INC	TMPTRY		:INC RETRY COUNT
116	004334				BR	WRITE8		:DO RETRY
117	004336	104303	002170	1\$:	MOV	RECTMP,R3		:GET CURRENT ERROR RECOVERY LEVEL
118	004340				BMI	2\$:IF NEGATIVE THEN FRIED
119	004342	115000	002166		TST	RECOV		:IS THERE ONLY RECOVERY LEVEL 0 ?
120	004344				BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY
121	004346				CALL	ERRHND		:TRY RECOVERY
122	004350	114000	002167	3\$:	CLR	TMPTRY		:FOR INIT
123	004352	117400	002170		DEC	RECTMP		:DECREMENT IT
124	004354				BR	WRITE8		:RETRY
125	004356			2\$:				
126	004356	117405			DEC	R5		:DEREMENT COUNTER
127	004357	104300	002166	002170	MOV	RECOV,RECTMP		:GET RECOVERY LEVELS
128	004362				BR	FOONE		:SEE IF ANY MORE TO DO
129	004364	115400	001743	OLDONE:	INC	FCTCNT		:INCREMENT IT
130	004366				RETURN			:ALL DONE
131	004370	115000	001743	OFATAL:	TST	FCTCNT		:SECTOR 0 ? (MEDIA INFO)
132	004372				BEQ	OQUIT		:IF YES THEN IT'S ALL OVER
133	004374	102200	000020	001702	BIT	#GOBAD,FLAG		:CONTINUE AS BEST GUESS?
134	004377				BEQ	OQUIT		:NOPE - GIVE UP
135	004401	101200	002004	001702	BIS	#FCTBAD+BSTGS,FLAG		:SET FCT BAD FLAG
136	004404	103200	000001	001702	BIC	#FCTAVL,FLAG		:NO MORE FCT
137	004407	104201	000000		MOV	#F1,R1		:SET IN D/XBN OVERLAY
138	004411				CALL	NEXT		:START OVER IN BEST GUESS MODE
139	004413				RETURN			:RETURN
140	004415	104012		OQUIT:	MOV	R1,R2		:XFC ERROR CODE
141	004416	104201	000021		MOV	#17,R1		:FCT READ ERROR
142	004420				CALL	ERRMNT		:ERROR RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 79
 GET FCT BLOCK FOR D/XBN FORMAT (G2)

1					.SBTTL	GET FCT BLOCK FOR D/XBN FORMAT (G2)	
2							
3							
4							
5							
6							
7	004422				DMOVLY	G2,START	
8							
9							
10							
11	004014	104200	000033	001636	MOV	#G2,CUROVL	:SIGNAL OVERLAY 11
12	004017	104205	001750		MOV	#DMBUF,R5	:POINT TO MAINT BUFFER
13	004021	104303	002003		MOV	FCMSG,R3	:GET DUP CODE
14	004023	100153			MOV	R3,(R5)	:STORE IT
15	004024	104303	001743		MOV	FCTCNT,R3	:GET BLOCK NUMBER DESIRED
16	004026	100653	000001		MOV	R3,1(R5)	:STORE IT
17	004030				CALL	SNDMNT	:SEND REQUEST
18	004032				CALL	RCVMNT	:RECEIVE ANSWER
19	004034	104153			MOV	(R5),R3	:GET STATUS WORD
20	004035				BNE	DLERR	:ERROR IF NOT ZERO
21	004037	104650	000001	002105	MOV	1(R5),OVLBLK+1	:GET LOW HOST ADDRESS
22	004042	104650	000002	002106	MOV	2(R5),OVLBLK+2	:GET HIGH HOST ADDRESS
23	004045	104200	000401	002104	MOV	#257.,OVLBLK	:GET LENGTH
24	004050	104204	002104		MOV	#OVLBLK,R4	:FOR OVERLAY ROUTINE
25	004052	104203	010455		MOV	#PBNBUF,R3	:POINT TO BUFFER
26	004054				CALL	OVLAY	:GET THE SECTOR
27	004056				RETURN		
28	004060	104201	000023		DLERR: MOV	#19.,R1	:SIGNAL DLL ERROR
29	004062	104302	001743		MOV	FCTCNT,R2	:BLOCK FAILED ON
30	004064				CALL	ERRMNT	:ERROR RETURN

UDAF52 - .DA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 80
GET FCT BLOCK FOR LBN FORMAT (G3)

```

1          .SBTTL GET FCT BLOCK FOR LBN FORMAT (G3)
2
3          GET RIGHT FCT BLOCK FOR LBN FORMATTING
4
5
6
7 004066   DMOVLY G3,START
8
9
10
11 004014 104200 000036 001636   MOV    #G3,CUROVL      ;FOR CURRENT OVERLAY
12 004017 114000 001743          CLR    FCTCNT         ;FOR FIRST FCT BLOCK
13 004021 104200 010455 001740   MOV    #PBNBUF,BUFPT  ;POINT TO BUFFER
14 004024 104201 000017          MOV    #F6,R1         ;FCT READ OVERLAY
15 004026          CALL   PAGE           ;READ IT IN
16 004030 102200 000001 001702   BIT    #FCTAVL,FLAG   ;FCT STILL HERE ?
17 004033          BEQ    NGD           ;NOPE - CAN IT
18 004035 104207 010455          MOV    #PBNBUF,R0     ;POINT TO BUFFER
19 004037 104173          MOV    (R0),R3        ;GET FORMAT MEDIA WORD
20 004040 106203 126736          CMP    #M512,R3       ;IS IT 512 ?
21 004042          BEQ    13$           ;YUP -- O.K.
22 004044 106203 074161          CMP    #M576,R3       ;IS IT 576 ?
23 004046          BEQ    13$           ;YUP - O.K.
24 004050 115003          TST   R3              ;IS IT FORMAT IN PROGRESS
25 004051          BNE   NGD           ;NOPE - FCT NO GOOD
26 004053 104673 000025          13$: MOV    FCTFLG(R0),R3  ;GET FLAG WORD
27 004055 102203 100000          BIT    #NOFCT,R3     ;IS THERE REALLY AN FCT ??
28 004057          BNE   NGD1          ;NOPE - FIND OUT IF WE QUIT OR ROUGH IT
29 004061 104204 010455          MOV    #PBNBUF,R4     ;POINT TO BUFFER
30 004063 102200 020000 001703   BIT    #MODE,FLAG1    ;WHAT MODE ARE WE IN
31 004066          BEQ    1$            ;IF CLEAR THEN 512 MODE
32 004070 104643 000020          MOV    C576(R4),R3    ;GET COUNT OF USED ENTRIES +576)
33 004072          BR    2$            ;SKIP 512 STUFF
34 004074 104643 000016          1$: MOV    C512(R4),R3  ;GET COUNT OF USED ENTRIES
35 004076 104030 001722          2$: MOV    R3,FCNT     ;STORE IT
36 004100          BNE   12$           ;IF NOT ZERO THEN ENTRIES EXIST
37 004102 101200 000002 001702   BIS    #FCTEMT,FLAG   ;SET EMPTY FLAG
38 004105 114003          12$: CLR    R3         ;FOR FCT INIT
39 004106 100173          MOV    R3,(R0)        ;SIGNAL FORMAT IN PROGRESS
40 004107 104673 000001          MOV    INST(R0),R3    ;FORMAT INSTANCE NUMBER
41 004111 115403          INC   R3              ;INCREMENT IT
42 004112 100673 000001          MOV    R3,INST(R0)   ;STORE IT BACK
43 004114 104203 010455          MOV    #PBNBUF,R3    ;POINT TO BUFFER
44 004116 105203 000002          ADD   #FSER,R3       ;POINT TO SERIAL NUMBER
45 004120 104204 001772          MOV    #SERNUM,R4    ;SERIAL NUMBER BLOCK
46 004122 104205 000004          MOV    #4,R5         ;COUNTER
47 004124 104232          8$: MOV    (R3)+,R2     ;GET WORD
48 004125 100242          MOV    R2,(R4)+     ;STORE WORD
49 004126 117405          DEC   R5             ;DECREMENT COUNTER
50 004127          BNE   8$            ;CONT TILL DONE
51 004131 104200 010455 001740   MOV    #PBNBUF,BUFPT  ;POINT TO BUFFER
52 004134 104201 000030          MOV    #F9,R1         ;FCT WRITE OVERLAY
53 004136          CALL   PAGE           ;DO IT
54 004140 115000 001722          TST   FCNT           ;ANY ENTRIES ?
55 004142          BEQ   RDONE1         ;NOPE - ALL DONE
56 004144 104200 000001 001743   MOV    #1,FCTCNT     ;FOR FCT COUNT INIT
57 004147 102200 020000 001703   BIT    #MODE,FLAG1    ;WHAT MODE ARE WE IN

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 80-1
GET FCT BLOCK FOR LBN FORMAT (G3)

UD
RC'

58	004152				BEQ	RLOOP	:512 - NO ADJUSTMENT NEEDED
59	004154	105300	001723	001743	ADD	FCTSUB,FCTCNT	:POINT TO 576 TABLE
60	004157	104200	010455	001740	MOV	#PBNBUF,BUFPT	:POINT TO BUFFER
61	004162	104201	000017		MOV	#F6,R1	:FCT READ OVERLAY
62	004164				CALL	PAGE	:DO THE READ
63	004166	102200	000001	001702	BIT	#FCTAVL,FLAG	:STILL HAVE FCT ?
64	004171				BEQ	NGD	:NOPE - CAN IT
65	004173	104204	010455		MOV	#PBNBUF,R4	:POINT TO THE BUFFER
66	004175	105204	000376		ADD	#254.,R4	:POINT TO LAST ENTRY
67	004177	104203	001637		MOV	#HGHPBN,R3	:HIGHEST PBN IN LBN AREA
68	004201				CALL	DCMP	:IS IT RIGHT BLOCK ?
69	004203				BPL	BLKFND	:YES - FIND RIGHT ENTRY
70	004205	107200	000200	001722	SUB	#128.,FCNT	:SUBTRACT ONE BLOCKS WORTH
71	004210				BR	RLOOP	:BRANCH BACK
72	004212	104200	000200	002163	BLKFND: MOV	#128.,PCNT	:FOR INIT OF COUNT
73	004215	104204	010455		MOV	#PBNBUF,R4	:POINT TO PBN BUFFER
74	004217	104647	000001		RLOOP1: MOV	1(R4),R0	:GET HIGH ORDER
75	004221	104071			MOV	R0,R1	:SAVE IT TEMPORARILY
76	004222	103207	170000		BIC	#HD.CLR,R0	:CLEAR FOR COMPARE
77	004224	100647	000001		MOV	R0,1(R4)	:STORE IT BACK
78	004226	104203	001637		MOV	#HGHPBN,R3	:POINT TO HIGHEST PBN
79	004230				CALL	DCMP	:COMPARE
80	004232				BPL	RDONE	:IF LESS THAN OR EQUAL THEN FOUND FIRST LBBN
81	004234	100641	000001		MOV	R1,1(R4)	:STORE HEADER BACK
82	004236	117400	002163		DEC	PCNT	:DECREMENT COUNT
83	004240	117400	001722		DEC	FCNT	:DEC IT
84	004242	105204	000002		ADD	#2,R4	:POINT TO NEXT ENTRY
85	004244	104200	000001	001722	C/FP	#1,FCNT	:COUNT AT 1 ?
86	004247				B=Q	RDONE	:YUP - THEN LAST ENTRY IS IT
87	004251				BR	RLOOP1	:TRY NEXT ENTRY
88	004253	100641	000001		RDONE: MOV	R1,1(R4)	:STORE HEADER BACK
89	004255	104040	001706		MOV	R4,BADPBN	:MAKE CURRENT BAD POINTER
90	004257	104300	001722	001776	RDONE1: MOV	FCNT,FCTREV	:FCT ENTRY COUNT FOR LATER USE
91	004262				RETURN		:RETURN
92	004264	102200	000020	001702	NGD: BIT	#GOBAD,FLAG	:CONTINUE AS BEST GUESS ?
93	004267				BEQ	RQUIT	:NOPE - GIVE UP
94	004271	101200	002004	001702	BIS	#FCTBAD+BSTGS,FLAG	:SET BAD FCT FLAG
95	004274	103200	000001	001702	BIC	#FCTAVL,FLAG	:NO MORE FCT
96	004277	104201	000000		MOV	#F1,R1	:POINT TO D/XBN OVERLAY
97	004301				CALL	NEXT	:START OVER IN BEST GUESS MODE
98	004303				RDONE2: RETURN		
99	004305	101200	002004	001702	NGD1: BIS	#FCTBAD+BSTGS,FLAG	:SET FCT NOT USED FLAG
100	004310	103200	000001	001702	BIC	#FCTAVL,FLAG	:NO MORE FCT (SO WILL DO EXTEN READS)
101	004313				BR	RDONE2	:EXIT
102	004315	104201	000022		RQUIT: MOV	#18.,R1	:ERROR CODE
103	004317	114002			CLR	R2	:NO SUBCODE
104	004320				CALL	ERRMNT	:ERROR RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 81
 RCT CLEANUP OVERLAY (G4)

```

1          .SBTTL  RCT CLEANUP OVERLAY (G4)
2          :
3          :
4          :
5 004322   :
6          :
7          :
8          :
9 004014   104200 000041 001636   MOV      #G4,CUROVL      :FOR OVERLAY IDENT
10 004017   104207 001750           MOV      #DMBUF,R0       :MESSAGE BUFFER
11 004021   104303 002155           MOV      SDCNT,R3        :ANY SECONDARY REVECTORS ?
12 004023   100173           MOV      R3,(R0)         :STORE IT
13 004024   115003           TST      R3              :ARE THERE ANY ?
14 004025           BEQ      CLSKP3          :NOPE - JUST EXIT
15 004027   104202 012264           MOV      #REVBUFF,R2    :POINT TO REVECTOR BUFFER
16 004031   104200 000100 002164   MOV      #64.,COJNT     :COUNT OF MAX TO REVECTOR AT ONCE
17 004034   114090 001561           CLR      CURRBN          :CLEAR FOR INIT
18 004036   114000 001562           CLR      CURRBN+1       :HIGH ORDER TOO
19 004040   104200 000002 002162   MOV      #2,RCTCNT      :INIT RCT BLOCK
20 004043   104200 000200 002137 CLELP2: MOV      #128.,SECCNT  :GET COUNT OF RCT ENTRIES
21 004046   104304 002162           MOV      RCTCNT,R4      :GET CLOCK NUMBER TO READ
22 004050           CALL     RRC             :READ IT
23 004052   104205 013477           MOV      #RCTBUF,R5     :POINT TO BUFFER
24 004054   104653 000001 CLELP:  MOV      1(R5),R3     :GET HEADER
25 004056   103203 007777           BIC      #LO,R3         :CLEAR OUT LOW GARBAGE
26 004060   106203 030000           CMP      #RC.SND,R3     :IS IT A SECONDARY ?
27 004062           CLEXP                    :NO - SKIP REVECTORING
28 004064   104153           MOV      (R5),R3        :GET LOW ORDER
29 004065   100223           MOV      R3,(R2)+       :STORE IN REVECTOR BUFFER
30 004066   104653 000001           MOV      1(R5),R3        :GET HIGH ORDER
31 004070   103203 170000           BIC      #HD.CLR,R3     :CLEAR HEADER
32 004072   101203 030000           BIS      #HD.REV,R3     :SET AS AN LBN REVECTOR
33 004074   100223           MOV      R3,(R2)+       :STORE IT
34 004075   104303 001561           MOV      CURRBN,R3      :GET LOW ORDER RBN NUMBER
35 004077   100223           MOV      R3,(R2)+       :STORE IT
36 004100   104303 001562           MOV      CURRBN+1,R3    :GET HIGH ORDER
37 004102   100223           MOV      R3,(R2)+       :STORE IT
38 004103   117400 002164           DEC      COUNT          :DEC NUM OF EMPTY REVECTOR SLOTS
39 004105   117400 002155           DEC      SDCNT          :DECREMENT IT
40 004107           BEQ      CLSKP4          :IF ZERO THEN DONE
41 004111   104303 002164           MOV      COUNT,R3       :FULL BLOCK ?
42 004113           BEQ      CLSKP2          :IF 0 - PROCESS BLOCK
43 004115           CLESKP: DUBINC          :INCREMENT IT
44 004123   105205 000002           ADD      #2,R5          :POINT TO NEXT RBN ENTR
45 004125   117400 002137           DEC      SECCNT         :DECREMENT IT
46 004127           BNE      CLELP          :DO NEXT ENTRY IF NOT ZERO
47 004131   115400 002162           INC      RCTCNT         :INCREMENT IT
48 004133   106300 001727 002162   CMP      RCTFMT,RCTCNT  :DONE ?
49 004136           BNE      CLELP2          :NOPE - READ IN NEXT BLOCK
50 004140           BR       CLEDON          :ELSE DONE
51 004142           CLSKP2: CALL     CLEWRT   :PROCESS THE BLOCK
52 004144   104200 000100 002164   MOV      #64.,COUNT   :FOR COUNTER INIT
53 004147   104202 012264           MOV      #REVBUFF,R2    :RESET POINTER
54 004151           BR       CLEXP          :BRANCH BACK
55 004153   106200 000100 002164 CLEDON: CMP      #64.,COUNT :DONE ANY ?
56 004156           BEQ      CLSKP3          :NO - DONE
57 004160           CLSKP4: CALL     CLEWRT  :WRITE OUT ANY LEFTOVERS

```


UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 81-1
RCT CLEANUP OVERLAY (G4)

58 004162
59 004162 104201 000066
60 004164
61 004166
62 004170
63 004172 114007
64 004173 060021

CLSKP3:

MOV #H2,R1
CALL PAGE
CALL SNDRES
CALL DISCON
CLR RO
XFC DONE

;FINAL CHECK OF FCT,RCT,HEADS OVERLAY
;PAGE IT IN
;SEND FINAL STATS
;DISCONNECT/SPINDOWN DRIVE
;MAKE SURE QUILTS NICELY
;DONE

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 82
 RCT CLEANUP OVERLAY (G4)

```

1
2
3
4 004174
5
6
7
8
9
10 004203 030001
11 004204 106 157 162
12 000012
13
14
15
16 004215 040014
17 004216 106 103 124
18 000014
19
20
21
22 004231 040015
23 004232 106 103 124
24 000012
25
26
27
28 004243 030020
29 004244 127 101 122
30 000040
31
32
33 004303
34
35
36
37 004303 000015
38 004304 004311
39 004305 030002
40 004306
41 004311 004311
42 004311 040 122 145
43 000015
44
45
46
47 004322 000021
48 004323 004330
49 004324 030003
50 004325
51 004330 004330
52 004330 040 120 162
53 000021
54
55
56
57 004345 000026

```

DATA STRUCTURES FOR STATUS RESPONSE

```

PARMTB: .BLKW 7 ;PARAMETER TABLE

FORMAT COMPLETE

.ENABL LC
FINMSG: .WORD 30001 ;DUP WORD
.ASCIZ 'Format completed'
LFINMS = .-FINMSG

FCT USED

FCTUSD: .WORD 40014 ;DUP WORD
.ASCIZ 'FCT used successfully'
LFCTUS = .-FCTUSD

FCT NOT USED

FCTNOT: .WORD 40015 ;DUP WORD
.ASCIZ 'FCT was not used'
LFCTNT = .-FCTNOT

WARNING OF POSSIBLE HEAD PROBLEM

WRN: .WORD 30020 ;DUP WORD
.ASCIZ 'WARNING - possible head addressing problem - run diagnostics'
WRNLN = .-WRN

RESTAB:

REVECTORED LBNS

1$: .WORD LLEN ;LENGTH
.LWORD LBUFE ;END OF BUFFER
.WORD 30002 ;DUP WORD
.BLKW 3 ;CONVERT BUFFER
LBUFE =
.ASCIZ 'Revectored LBNS'
LLEN = .-1$

PRIMARY REVECTORS

2$: .WORD PLEN ;LENGTH
.LWORD PBUFE ;END OF BUFFER
.WORD 30003 ;DUP WORD
.BLKW 3 ;CONVERT BUFFER
PBUFE =
.ASCIZ 'Primary revectored LBNS'
PLEN = .-2$

SECONDARY REVECTORS

.SWORD SLEN ;LENGTH

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 82-1
 RCT CLEANUP OVERLAY (G4)

```

58 004346 004353          .WORD  SBUFE          ;END OF BUFFER
59 004347 030004          3$: .WORD  30004          ;DUP WORD
60 004350          .BLKW   3          ;CONVERT BUFFER
61          004353          SBUFE   =          ;
62 004353          040     123   145 .ASCIZ  ; Secondary/tertiary revector'd LBNS'
63          000026          SLEN   =          ; -3$
64          :              :
65          :              RCT BAD BLOCKS
66          :              :
67 004375 000034          .WORD  RCLEN          ;LENGTH
68 004376 004403          .WORD  RCBUFE         ;END OF BUFFER
69 004377 030005          4$: .WORD  30005         ;DUP WORD
70 004400          .BLKW   3          ;CONVERT BUFFER
71          004403          RCBUFE  =          ;
72 004403          040     102   141 .ASCIZ  ; Bad blocks in the RCT area due to data errors'
73          000034          RCLEN   =          ; -4$
74          :              :
75          :              DBN BAD BLOCKS
76          :              :
77          :              :
78 004433 000034          .WORD  DBLEN          ;LENGTH
79 004434 004441          .WORD  DBBUFE         ;END OF BUFFER
80 004435 030007          5$: .WORD  30007         ;DUP WORD
81 004436          .BLKW   3          ;CONVERT BUFFER
82          004441          DBBUFE  =          ;
83 004441          040     102   141 .ASCIZ  ; Bad blocks in the DBN area due to data errors'
84          000034          DBLEN   =          ; -5$
85          :              :
86          :              XBN BAD BLOCKS
87          :              :
88 004471 000034          .WORD  XBLEN          ;LENGTH
89 004472 004477          .WORD  XBBUFE         ;END OF BUFFER
90 004473 030010          6$: .WORD  30010         ;DUP WORD
91 004474          .BLKW   3          ;CONVERT BUFFER
92          004477          XBBUFE  =          ;
93 004477          040     102   141 .ASCIZ  ; Bad blocks in the XBN area due to data errors'
94          000034          XBLEN   =          ; -6$
95          :              :
96          :              RETRIED BLOCKS
97          :              :
98 004527 000025          .WORD  RELEN          ;LENGTH
99 004530 004535          .WORD  REBUFE         ;END OF BUFFER
100 004531 030013          7$: .WORD  30013         ;DUP WORD
101 004532          .BLKW   3          ;CONVERT BUFFER
102          004535          REBUFE  =          ;
103 004535          040     102   154 .ASCIZ  ; Blocks retried on the check pass'
104          000025          RELEN   =          ; -7$
105          :              .DSABL  LC
106          :              :
107          :              :
108          :              END OF LIST
109          :              :
110 004556 000000          .WORD  0
111          :              :
112          :              BUILD AND SEND STAT RESPONSES
113          :              :
114 004557 104207 004203  SNDRES: MOV    #FINMSG,R0          ;POINT TO 'COMPLETE' MESSAGE

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 82-2
 RCT CLEANUP OVERLAY (G4)

115	004561	104201	000012		MOV	#LFINMS,R1		:LENGTH OF MESSAGE
116	004563	060016			XFC	MAINTR		:SEND IT
117	004564	104204	004303		MOV	#RESTAB,R4		:POINT TO TABLE
118	004566	104203	004174		MOV	#PARMTB,R3		:PARAMETER TABLE
119	004570	104302	001777		MOV	LBNBAD,R2		:LBN'S BAD
120	004572	100232			MOV	R2,(R3)+		:STORE IT
121	004573	107302	001750		SUB	DMBUF,R2		:SUBTRACT STORED SECONDARIES TO GET PRIMARY
122	004575	100232			MOV	R2,(R3)+		:STORE IT
123	004576	104302	001750		MOV	DMBUF,R2		:STORED SECONDARY COUNT
124	004600	100232			MOV	R2,(R3)+		:STORE IT
125	004601	104302	002000		MOV	RCTBAD,R2		:RCT BAD BLOCKS
126	004603	100232			MOV	R2,(R3)+		:STORE IT
127	004604	104302	002001		MOV	DBBAD,R2		:DBN BAD BLOCKS
128	004606	100232			MOV	R2,(R3)+		:STORE IT
129	004607	104302	002002		MOV	XBBAD,R2		:XBN BAD BLOCKS
130	004611	100232			MOV	R2,(R3)+		:STORE IT
131	004612	104302	002156		MOV	RTYCNT,R2		:RETRIES
132	004614	100232			MOV	R2,(R3)+		:STORE IT
133	004615	104203	004174		MOV	#PARMTB,R3		:POINT BACK TO BEGINNING
134	004617	104242			MOV	(R4)+,R2		:GET LENGTH OF MESSAGE
135	004620	104245			MOV	(R4)+,R5		:GET END OF BUFFER ADDRESS (FOR CONVERT)
136	004621			SNDLP:	CALL	CLRBUF		:INITIALIZE THE BUFFER
137	004623	104047			MOV	R4,R0		:MOVE ADDRESS OF STRING TO R0
138	004624	103200	000400	001703	BIC	#FLIPON,FLAG1		:CLEAR FLAG (FOR CONVERT)
139	004627	104231			MOV	(R3)+,R1		:GET WORD TO CONVERT
140	004630				CALL	DECASC		:CONVERT TO ASCII
141	004632	104021			MOV	R2,R1		:GET LENGTH IN R1
142	004633				PUSH	R2		:SAVE LENGTH
143	004634	060016			XFC	MAINTR		:SEND TO THE HOST
144	004635				POP	R2		:RESTORE LENGTH
145	004636	105024			ADD	R2,R4		:POINT TO THE NEXT MESSAGE
146	004637	104242			MOV	(R4)+,R2		:GET LENGTH
147	004640				BNE	SNDLP		:BRANCH IF NOT ZERO
148	004642	102200	002000	001702	BIT	#BSTGS,FLAG		:DID WE USE FCT ?
149	004645				BEQ	USDFCT		:YES - PRINT THAT MESSAGE
150	004647	104207	004231		MOV	#FCTNOT,R0		:POINT TO NOT USED MESSAGE
151	004651	104201	000012		MOV	#LFCTNT,R1		:LENGTH OF MESSAGE
152	004653	060016			XFC	MAINTR		:SEND IT
153	004654				BR	DONFCT		:EXIT
154	004656	104207	004215		USDFCT: MOV	#FCTUSD,R0		:POINT TO USED MESSAGE
155	004660	104201	000014		MOV	#LFCTUS,R1		:LENGTH OF MESSAGE
156	004662	060016			XFC	MAINTR		:SEND IT
157	004663				DONFCT: RETURN			:RETURN

UDAF52 - UDA-52 FORMATTER DMACK X04.01 23-AUG-82 13:14:22 PAGE 85
 RCT CLEANUP OVERLAY (G4)

1						
2						
3						
4	004774					
5	005002	104204	012264			
6	005004	104300	002164	001410		
7	005007	104642	000002			
8	005011	104643	000003			
9	005013	101203	060000			
10	005015	104205	015306			
11	005017	104201	000200			
12	005021	100252				
13	005022	100253				
14	005023	117401				
15	005024					
16	005026	104203	001525			
17	005030	102200	020000	001703		
18	005033					
19	005035	104632	000011			
20	005037					
21	005041	104632	000015	1\$:		
22	005043	103202	177400	2\$:		
23	005045	104207	002110			
24	005047	100672	000004			
25	005051	104632	000001			
26	005053	103202	007777			
27	005055	100672	000001			
28	005057	114002				
29	005060	100672	000000			
30	005052					
31	005064	104202	015306			
32	005066	104307	002171			
33	005070					
34	005072	102200	020000	001703		
35	005075					
36	005077	100623	000440			
37	005101					
38	005103	100623	000400	3\$:		
39	005105	104207	001373	4\$:		
40	005107	100672	000002			
41	005111	104203	122400			
42	005113	104302	001565			
43	005115	101023				
44	005116	100673	000005			
45	005120	104143				
46	005121	100673	000003			
47	005123	104643	000001			
48	005125	105303	002007			
49	005127	100673	000004			
50	005131	104203	001400			
51	005133	100673	000006			
52	005135	104040	001403			
53	005137	104303	002005	WRITE9:		
54	005141	104304	002006			
55	005143	104302	001412			
56	005145	104207	001373			
57	005147	101207	100000			

PROCESS REVECTOR BLOCK

```

CLEWRT: PUSHA
MOV #NEVBUF,R4          :POINT TO BUFFER
MOV COUNT,TEMP         :GET COUNT
CLHERE: MOV 2(R4),R2    :GET LOW ORDER RBN
MOV 3(R4),R3           :GET HIGH ORDER RBN
BIS #HD.RBN,R3        :SET IN HDR CODE
MOV #CLBUF,R5         :POINT TO BUFFER
MOV #RBNRPT,R1        :INIT COUNTER
WLOOP: MOV R2,(R5)+    :STORE LOW ORDER
MOV R3,(R5)+          :STORE HIGH ORDER
DEC R1                :DECREMENT COUNTER
BNE WLOOP             :CONTINUE TILL DONE
MOV #SCR,R3           :POINT TO CHARACTERISTICS
BIT #MODE,FLAG1      :WHAT MODE
SNE 1$                :IF SET THEN 576
MOV LBNT12(R3),R2    :GET LBN/TRACK FOR 512
BR 2$                 :SKIP 576 SETUP
1$: MOV LBNT76(R3),R2 :GET LBN/TRACK FOR 576
2$: BIC #HIBYTE,R2   :CLEAR HIGH BYTE
MOV #CONBLK,R0       :POINT TO CONVERT BLOCK
MOV R2,V3(R0)        :FOR CONVERT
MOV STCYL(R3),R2    :STARTING CLYLINDER
BIC #LO,R2           :CLEAR REST OF WORD
MOV R2,V1+1(R0)     :STORE
CLR R2               :FOR STORE
MOV R2,V1(R0)       :LOW ALWAYS ZERO
CALL CS              :CONVERT AND SEEK
MOV #CLBUF,R2        :POINT TO BUFFER
MOV SECSIZ,R0        :SECTOR SIZE IN WORDS
CALL CEDC            :COMPUTE EDC - RETURNED IN R3
BIT #MODE,FLAG1     :WHAT MODE ARE WE IN
BEQ 3$               :IF CLEAR THEN 512
MOV R3,RW.E76(R2)   :STORE IT 576 BUFFER
BR 4$                :SKIP 512 SETUP
3$: MOV R3,RW.EDC(R2) :STORE IT 512 BUFFER
4$: MOV #WRBLK,R0    :POINT TO COMMAND BLOCK
MOV R2,RW.BUF(R0)   :STICK BUFFER PTR IN COMMAND BLOCK
MOV #WRCMD,R3       :GET WRITE COMMAND
MOV CURTRK,R2      :GET CURRENT TRACK
BIS R2,R3           :SET TRACK FOR WRITE
MOV R3,RW.CMD(R0)   :STORE IN COMMAND BLOCK
MOV (R4),R3         :GET LOW ORDER HEADER
MOV R3,RW.LOW(R0)   :STORE IN WRITE BLOCK
MOV 1(R4),R3        :GET HIGH ORDER
ADD ST.LBN,R3       :ADD STARTING LBN BITS
MOV R3,RW.HI(R0)    :STORE IN WRITE BLOCK
MOV #HSLIM-1,R3     :GET DUMMY SDI POINTER
MOV R3,RW.DUM(R0)   :POINT IN COMMAND BLOCK
MOV R4,DDUMMY       :SAVE R4
WRITE9: MOV HPREA,R3 :GET HEADER PREAMBLE
MOV DPREA,R4        :GET DATA PREAMBLE
MOV UNIT,R2         :GET PORT NUMBER FOR SIP
MOV #WRBLK,R0       :MAKE SURE POINTING AT BLOCK
BIS #BIT15,R0      :SET NO REVECTORING

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 85-1
 RCT CLEANUP OVERLAY (G4)

```

58 005151 060012          XFC      SIP          ;WAIT FOR SECTOP PULSE
59 005152 104302 002171  MOV      SECSIZ,R2    ;SECTOR SIZE IN WORDS
60 005154 060003          XFC      WRITE       ;WRITE SECTOR
61 005155 115001          TST      R1          ;ANY ERROR ?
62 005156          BEQ      1$      ;NO - SKIP RETRY
63 005160 106200 000010 002167  CMP      #MAXTRY, TMPTRY ;MAX ?
64 005163          BEQ      1$      ;YES - GIVE UP
65 005165 115400 002167          INC      TMPTRY     ;INC RETRY COUNT
66 005167          BR       WRITE9    ;DO RETRY
67 005171          1$:
68 005171 114000 002167          CLR      TMPTRY     ;CLEAR RETRY COUNT
69 005173 104304 001403          MOV      DDUMMY,R4   ;RESTORE R4
70 005175 105204 000004          ADD      #REVLEN,R4  ;POINT TO NEXT ENTRY
71 005177 115400 002164          INC      COUNT      ;INC COUNTER
72 005201 106200 000100 002164  CMP      #64,COUNT   ;DONE ?
73 005204          BNE     CLHERE    ;NO - REPEAT
74 005206          CALL    RBNWRT   ;WRITE GOOD EDC'S TO RBN'S
75 005210          POPA
76 005216          RETURN
77          :
78          :
79          :
80          :
81 005220 104207 002110          CS:    MOV      #CONBLK,R0 ;POINT TO CONVERT BLOCK
82 005222 104143          MOV      (R4),R3     ;GET LOW ORDR
83 005223 100673 000002          MOV      R3,V2(R0)  ;STORE IT
84 005225 104643 000001          MOV      1(R4),R3   ;HIGH ORDER
85 005227 103203 170000          BIC     #HD,CLR,R3  ;CLEAR HEADER
86 005231 100673 000003          MOV      R3,V2+1(R0);STORE IT
87 005233 104201 001525          MOV      #SCR,R1    ;POINT TO SUBUNIT CHARACTERISTICS
88 005235 060020          XFC      CVT        ;CONVERT IT
89 005236 104670 000011 001565  MOV      TRK(R0),CURTRK ;GET TRACK NUMBER
90 005241 104670 000006 001551  MOV      CYL(R0),ISEEK+1 ;LOW ORDER CYLINDER
91 005244 104670 000007 001552  MOV      CYL+1(R0),ISEEK+2 ;HIGH ORDER CYLINDER
92 005247 104670 000010 001553  MOV      GRP(R0),ISEEK+3 ;GROUP NUMBER
93 005252 102200 020000 001703  BIT      #MODE,FLAG1 ;WHAT MODE ARE WE IN
94 005255          BEQ      1$      ;512 - SEEK BIT O.K.
95 005257 104671 000004          MOV      V3(R0),R1  ;GET VARIABLE 3
96 005261 106010 001612          CMP      R1,SECT12  ;IF EQ TO SECT12 THEN D/XBN
97 005263          BEQ      1$      ;AND SEEK BIT O.K. (512)
98 005265 101200 100000 001552  BIS     #SS,ISEEK+2  ;ELSE L/RBN - SET 576 BIT
99 005270          1$:    CALL    SEEK        ;DO SEEK
100 005272 115001          TST     R1          ;ANY ERROR
101 005273          BNE     CKR        ;YUP
102 005275          RETURN
103 005277 104201 000012          CKR:   MOV      #10,R1     ;SEEK ERROR
104 005301 104207 002110          MOV      #CONBLK,R0 ;CONVERT BLOCK
105 005303 104672 000006          MOV      CYL(R0),R2 ;CYLINDER FAILD ON
106 005305          CALL    ERRMNT   ;ERROR RETURN

```


UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 86
 RCT CLEANUP OVERLAY (G4)

1	005307	104204	012264		RBNWRT:	MOV	#REV9UF,R4	:POINT TO BUFFER
2	005311	104203	011132			MOV	#GDBLK,R3	:POINT TO GOOD BLOCK
3	005313	104302	002132			MOV	EDC,R2	:GET GOOD EDC
4	005315	102200	020000	001703		BIT	#MODE,FLAG1	:WHAT MODE ARE WE IN
5	005320					BEQ	3\$:IF CLEAR THEN 512
6	005322	100622	000440			MOV	R2,RW.E76(R2)	:STORE IT 576 BUFFER
7	005324					BR	4\$:SKIP 512 SETUP
8	005326	100622	000400		3\$:	MOV	R2,RW.EDC(R2)	:STORE IT 512 BUFFER
9	005330	104203	001525		4\$:	MOV	#SCR,R3	:POINT TO CHARACTERISTICS
10	005332	104632	000004			MOV	RBNTRK(R3),R2	:GET RBN/TRACK
11	005334	103202	177600			BIC	#HI1BYTE,R2	:CLEAR HIGH GARBAGE
12	005336	104207	002110			MOV	#CONBLK,R0	:POINT TO CONVERT BLOCK
13	005340	100672	000004			MOV	R2,V3(R0)	:FOR CONVERT
14	005342	102200	020000	001703		BIT	#MODE,FLAG1	:WHAT MODE
15	005345					BNE	1\$:IF SET THEN 576
16	005347	104632	000011			MOV	LBNT12(R3),R2	:GET LBN/TRACK FOR 512
17	005351					BR	2\$:SKIP 576 SETUP
18	005353	104632	000015		1\$:	MOV	LBNT76(R3),R2	:GET LBN/TRACK FOR 576
19	005355	103202	177400		2\$:	BIC	#HI1BYTE,R2	:CLEAR HIGH BYTE
20	005357	100672	000005			MOV	R2,V4(R0)	:SET UP FOR RBN'S
21	005361	104632	000001			MOV	STCYL(R3),R2	:STARTING CLYINDER
22	005363	103202	007777			BIC	#LO,R2	:CLEAR REST OF WORD
23	005365	100672	000001			MOV	R2,V1+1(R0)	:STORE
24	005367	114002				CLR	R2	:FOR STORE
25	005370	100672	000000			MOV	R2,V1(R0)	:LOW ALWAYS ZERO
26	005372	104640	000002	001405	RNWHER:	MOV	2(R4),TEMP2	:GET LOW ORDER RBN
27	005375	104640	000003	001406		MOV	3(R4),TEMP2+1	:GET HIGH ORDER
28	005400	104040	001403			MOV	R4,DDUMMY	:SAVE R4
29	005402	104204	001405			MOV	#TEMP2,R4	:FOR CONVERT
30	005404					CALL	CVTSK	:CONVERT AND SEEK
31	005406	104207	001373			MOV	#WRBLK,R0	:POINT TO COMMAND BLOCK
32	005410	104203	122400			MOV	#WRCMD,R3	:GET WRITE COMMAND
33	005412	104302	001565			MOV	CURTRK,R2	:GET CURRENT TRACK
34	005414	101023				BIS	R2,R3	:SET TRACK FOR WRITE
35	005415	100673	000005			MOV	R3,RW.CMD(R0)	:STORE IN COMMAND BLOCK
36	005417	104202	011132			MOV	#GDBLK,R2	:POINT TO BLOCK
37	005421	100672	000002			MOV	R2,RW.BUF(R0)	:STICK IN COMMAND BLOCK
38	005423	104143				MOV	(R4),R3	:GET LOW ORDER HEADER
39	005424	100673	000003			MOV	R3,RW.LOW(R0)	:STORE IN WRITE BLOCK
40	005426	104643	000001			MOV	1(R4),R3	:GET HIGH ORDER
41	005430	105303	002010			ADD	ST.RBN,R3	:ADD STARTING RBN BITS
42	005432	101203	060000			BIS	#HD.RBN,R3	:GIVE RBN HEADER
43	005434	100673	000004			MOV	R3,RW.HI(R0)	:STORE IN WRITE BLOCK
44	005436	104203	001400			MOV	#HSLIM-1,R3	:GET DUMMY SDI POINTER
45	005440	100673	000006			MOV	R3,RW.DUM(R0)	:POINT IN COMMAND BLOCK
46	005442	104303	002005		WRIT13:	MOV	HPREA,R3	:GET HEADER PREAMBLE
47	005444	104304	002006			MOV	DPREA,R4	:GET DATA PREAMBLE
48	005446	104302	001412			MOV	UNIT,R2	:GET PORT NUMBER FOR SIP
49	005450	104207	001373			MOV	#WRBLK,R0	:MAKE SURE POINTING AT BLOCK
50	005452	101207	100000			BIS	#BIT15,R0	:SET NO REVECTORING
51	005454	060012				XFC	SIP	:WAIT FOR SECTOR PULSE
52	005455	104302	002171			MOV	SECSIZ,R2	:SECTOR SIZE IN WORDS
53	005457	060003				XFC	WRITE	:WRITE SECTOR
54	005460	115001				TST	R1	:ANY ERROR ?
55	005461					BEQ	2\$:NO - SKIP RETRY
56	005463	106300	002165	002167		CMP	RETRY,IMPTRY	:MAX ?
57	005466					BEQ	1\$:YES - TRY SOME RECOVERY

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 87
 RCT CLEANUP OVERLAY (G4)

1									
2									
3									
4									
5	005543				RRC:	PUSHA			
6	005551	114005				CLR	R5		:CLEAR ERROR COUNTER
7	005552	104203	001525			MOV	#SCR,R3		:POINT TO CHARACTERISTICS
8	005554	104632	000001			MOV	STCYL(R3),R2		:GET LOW ORDER STARTING CYLINDER
9	005556	103202	007777			BIC	#LO,R2		:CLER REST OF WORD
10	005560	104207	002110			MOV	#CONBLK,R0		:POINT TO CONVERT BLOCK
11	005562	100672	000001			MOV	R2,V1+1(R0)		:STORE FOR CONVERT
12	005564	114002				CLR	R2		:FOR STORE
13	005565	100672	000000			MOV	R2,V1(R0)		:LOW ORDER ALWAYS ZERO
14	005567	102200	020000	001703		BIT	#MODE,FLAG1		:WHAT MODE
15	005572					BNE	1\$:IF SET THEN 576
16	005574	104632	000011			MOV	LBNT12(R3),R2		:GET LBN/TRACK FOR 512
17	005576					BR	2\$:SKIP 576 SETUP
18	005600	104632	000015		1\$:	MOV	LBNT76(R3),R2		:GET LBN/TRACK FOR 576
19	005602	103202	177400		2\$:	BIC	#HIBYTE,R2		:CLEAR HIGH BYTE
20	005604	100672	000004			MOV	R2,V3(R0)		:STORE IN CONVERT BLOCK
21	005606	104040	001403			MOV	R4,DDUMMY		:STORE BLOCK NUMBER
22	005610	114000	001404			CLR	DDUMMY+1		:FOR STORE
23	005612	104204	001403			MOV	#DDUMMY,R4		:POINT FOR ADD
24	005614	104203	001712			MOV	#HOLD,R3		:STARTING RCT LBN
25	005616					CALL	DADD		:GET RCT LBN
26	005620				RCL:	CALL	CS		:CONVERT FCT BLOCK NUMBER AND SEEK
27	005622	104207	001373			MOV	#RDBLK,R0		:PREPARE FOR READ SECTORS
28	005624	104203	001400			MOV	#HSLIM-1,R3		:POINTER TO DUMMY SDI BLOCK
29	005626	100673	000006			MOV	R3,RW.DUM(R0)		:STORE IN COMMAND BLOCK
30	005630	104143				MOV	(R4),R3		:LO ORDER BLOCK NUMBER
31	005631	100673	000003			MOV	R3,RW.LOW(R0)		:STORE IN READ CMD BLOCK
32	005633	104643	000001			MOV	1(R4),R3		:HI ORDER BLOCK NUM AND CODE
33	005635	105303	002007			ADD	ST.LBN,R3		:ADD STARTING LBN BITS
34	005637	100673	000004			MOV	R3,RW.HI(R0)		:STORE IN READ CMD BLOCK
35	005641	104203	013477			MOV	#RCTBUF,R3		:LOAD ADDRESS OF DATA BUFFER
36	005643	100673	000002			MOV	R3,RW.BUF(R0)		:STORE IN COMMAND BUFFER
37	005645	104203	013400			MOV	#RWCMD,R3		:LOAD SDI READ COMMAND
38	005647	104301	001565			MOV	CURTRK,R1		:GET CURRENT HEAD NUMBER IN R1
39	005651	101013				BIS	R1,R3		:SET IT IN COMMAND
40	005652	100673	000005			MOV	R3,RW.CMD(R0)		:STORE BACK
41	005654	104207	001373		READ8:	MOV	#RDBLK,R0		:MAKE SURE POINTING AT BLOCK
42	005656	104203	100000			MOV	#RDCMD,R3		:MARK AS ONLY REQUEST
43	005660	100173				MOV	R3,(R0)		:STORE IN CMD BLOCK
44	005661	104302	001412			MOV	UNIT,R2		:GET PORT NUMBER FOR SIP
45	005663	101207	100000			BIS	#BIT15,R0		:SET NO REVECTORING
46	005665	060012				XFC	SIP		:WAIT FOR PULSE
47	005666	104302	002171			MOV	SECSIZ,R2		:SECTOR SIZE IN WORDS
48	005670	060002				XFC	READ		:READ 1 SECTOR
49	005671	115001				TST	R1		:ANY ERRORS ?
50	005672					BNE	100\$:YES - TRY RECOVERY
51	005674	104673	000001			MOV	RW.ER1(R0),R3		:GET STATUS WORD
52	005676	102203	010000			BIT	#ECCF,R3		:ECC ERROR ?
53	005700					BEQ	101\$:NOPE - VERIFY EDC
54	005702	103200	010000	001374		BIC	#ECCF,RDBLK+RW.ER1		:CLEAR ECC ERROR BIT
55	005705					CALL	ECCCK		:CORRECT ECC
56	005707	115001				TST	R1		:TEST FLAG
57	005710					BNE	100\$:UNCORRECTABLE

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 87-1
 RCT CLEANUP OVERLAY (G4)

UD
PB

58	005712	104202	013477	101\$:	MOV	#RCTBUF,R2	:POINT TO BUFFER	
59	005714	104307	002171		MOV	SECSIZ,R0	:SECTOR SIZE IN WORDS	
60	005716				CALL	CEDC	:COMPUTE EDC	
61	005720	102200	020000	001703	BIT	#MODE,FLAG1	:WHAT MODE ARE WE IN	
62	005723				BEQ	4\$:IF CLEAR THEN 512	
63	005725	106623	000440		CMP	RW.E76(R2),R3	:O.K. ?	
64	005727				BEQ	102\$:YUP - CONSIDER GOOD	
65	005731				BR	100\$:NOPE - RETRY	
66	005733	106623	000400	4\$:	CMP	RW.EDC(R2),R3	:O.K. ?	
67	005735				BEQ	102\$:YUP - CONSIDER GOOD	
68	005737	106300	002165	002167	100\$:	CMP	RETRY,TMPTRY	:MAX ?
69	005742				BEQ	1\$:YES - TRY SOME RECOVERY	
70	005744	115400	002167		INC	TMPTRY	:INC RETRY COUNT	
71	005746				BR	READ8	:DO RETRY	
72	005750	104303	002170	1\$:	MOV	RECTMP,R3	:GET CURRENT ERROR RECOVERY LEVEL	
73	005752				BMI	2\$:IF NEGATIVE THEN FRIED	
74	005754	115000	002166		TST	RECOV	:IS THERE ONLY RECOVERY LEVEL 0 ?	
75	005756				BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY	
76	005760				CALL	ERRHND	:TRY RECOVERY	
77	005762	114000	002167	3\$:	CLR	TMPTRY	:FOR INIT	
78	005764	117400	002170		DEC	RECTMP	:DECREMENT IT	
79	005766				BR	READ8	:RETRY	
80	005770			2\$:				
81	005770	115405			INC	R5	:INCREMENT BAD COUNTER	
82	005771	106305	001731		CMP	FCTCPY,R5	:ALL BAD ?	
83	005773				BEQ	RFTL	:YUP - ALL OVER	
84	005775	104203	001727		MOV	#RCTFMT,R3	:SIZE OF TABLE - R4 -> BLOCK NUMBER	
85	005777				CALL	DADD	:ADD TO POINT TO NEXT COPY	
86	006001	114000	002167		CLR	TMPTRY	:RESET RETRY LEVEL	
87	006003	104300	002166	002170	MOV	RECOV,RECTMP	:DITTO RECOVERY LEVELS	
88	006006				BR	RCL	:BRANCH BACK	
89	006010			102\$:				
90	006010	114000	002167	RCD:	CLR	TMPTRY	:FOR RESET	
91	006012	104300	002166	002170	MOV	RECOV,RECTMP	:GET RECOVERY LEVELS	
92	006015	115005			TST	R5	:ANY ERRORS ?	
93	006016				BEQ	RLD	:NO - EXIT	
94	006020	104203	001727		MOV	#RCTFMT,R3	:SIZE OF TABLE	
95	006022				CALL	DSUB	:GET BACK TO PREVIOUS COPY	
96	006024				CALL	CS	:CONVERT AND SEEK	
97	006026	104207	001373		MOV	#WRBLK,R0	:POINT TO COMMAND BLOCK	
98	006030	104203	122400		MOV	#WRCMD,R3	:GET WRITE COMMAND	
99	006032	104302	001565		MOV	CURTRK,R2	:GET CURRENT TRACK	
100	006034	101023			BIS	R2,R3	:SET TRACK FOR WRITE	
101	006035	100673	000005		MOV	R3,RW.CMD(R0)	:STORE IN COMMAND BLOCK	
102	006037	104203	013477		MOV	#RCTBUF,R3	:POINT TO BUFFER	
103	006041	100673	000002		MOV	R3,RW.BUF(R0)	:STICK IN COMMAND BLOCK	
104	006043	104143			MOV	(R4),R3	:GET LOW ORDER HEADER	
105	006044	100673	000003		MOV	R3,RW.LOW(R0)	:STORE IN WRITE BLOCK	
106	006046	104643	000001		MOV	1(R4),R3	:GET HIGH ORDER	
107	006050	105303	002007		ADD	ST.LBN,R3	:ADD STARTING LBN BITS	
108	006052	100673	000004		MOV	R3,RW.HI(R0)	:STORE IN WRITE BLOCK	
109	006054	104203	001400		MOV	#HSLIM-1,R3	:GET DUMMY SDI POINTER	
110	006056	100673	000006		MOV	R3,RW.DUM(R0)	:POINT IN COMMAND BLOCK	
111	006060	104303	002005	WRIT10:	MOV	HPREA,R3	:GET HEADER PREAMBLE	
112	006062	104304	002006		MOV	DPREA,R4	:GET DATA PREAMBLE	
113	006064	104302	001412		MOV	UNIT,R2	:GET PORT NUMBER FOR SIP	
114	006066	104207	001373		MOV	#WRBLK,R0	:MAKE SURE POINTING AT BLOCK	

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 87-2
 RCT CLEANUP OVERLAY (G4)

115	006070	101207	100000		BIS	#BIT15,R0		;SET NO REVECTORING
116	006072	060012			XFC	SIP		;WAIT FOR SECTOR PULSE
117	006073	104302	002171		MOV	SECSIZ,R2		;SECTOR SIZE IN WORDS
118	006075	060003			XFC	WRITE		;WRITE SECTOR
119	006076	115001			TST	R1		;ANY ERROR ?
120	006077				BEQ	2\$;NO - SKIP RETRY
121	006101	106300	002165	002167	CMP	RETRY,TMPTRY		;MAX ?
122	006104				BEQ	1\$;YES - TRY SOME RECOVERY
123	006106	115400	002167		INC	TMPTRY		;INC RETRY COUNT
124	006110				BR	WRIT10		;DO RETRY
125	006112	104303	002170	1\$:	MOV	RECTMP,R3		;GET CURRENT ERROR RECOVERY LEVEL
126	006114				BMI	2\$;IF NEGATIVE THEN FRIED
127	006116	115000	002166		TST	RECOV		;IS THERE ONLY RECOVERY LEVEL 0 ?
128	006120				BEQ	3\$;YES - NO NEED TO ISSUE IT - JUST RETRY
129	006122				CALL	ERRHND		;TRY RECOVERY
130	006124	114000	002167	3\$:	CLR	TMPTRY		;FOR INIT
131	006126	117400	002170		DEC	RECTMP		;DECREMENT IT
132	006130				BR	WRIT10		;RETRY
133	006132			2\$:				
134	006132	117405			DEC	R5		;DEREMENT COUNTER
135	006133	104300	002166	002170	MOV	RECOV,RECTMP		;GET RECOVERY LEVELS
136	006136				BR	RCD		;SEE IF ANY MORE TO DO
137	006140			RLD:	POPA			
138	006146				RETURN			;ALL DONE
139	006150	104012		RFTL:	MOV	R1,R2		;XFC ERROR CODE
140	006151	104201	000016		MOV	#14,R1		;RCT READ ERROR
141	006153				CALL	ERRMNT		;ERROR RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 88
 FINAL CHECK OVERLAY (H2)

1					.SBTTL FINAL CHECK OVERLAY (H2)	
2	006155				DMOVLY H2,START	
3						
4					DO CHECK OF:	
5						
6					FCT	
7					RCT	
8					CORRECT HEAD ADDRESSING	
9						
10	004014	104200	000066	001636	FINCHK: MOV #H2,CUROVL	:MAKE THIS THE CURRENT OVERLAY
11	004017				CALL VERHD	:VERIFY ALL HEADS ACCESSIBLE
12	004021				CALL FCTCK	:VERIFY FCT
13	004023				CALL RCTCK	:VERIFY RCT
14	004025				RETURN	

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 89
 FINAL CHECK OVERLAY (H2)

1					...	VERIFY FCT	
2					...		
3					...		
4	004027	114000	001743		FCTCK:	CLR FCTCNT	:FOR INIT
5	004031	115400	001745			INC FCTNPD	:SO CHECK THE NULL BLOCK ALSO
6	004033	104200	011607	001740		MOV #PRMBUF,BUFPNT	:POINT TO BUFFER
7	004036	104201	000017			MOV #F6,R1	:FCT READ OVERLAY
8	004040					CALL PAGE	:READ FIRST BLOCK
9	004042	104205	011607			MOV #PRMBUF,R5	:POINT TO BUFFER
10	004044	102200	020000	001703		BIT #MODE,FLAG1	:WHAT MODE
11	004047					BNE 1\$:IF SET THEN 576
12	004051	104203	126736			MOV #M512,R3	:GET MODE INDICATOR FOR 512
13	004053					BR 2\$:SKIP 576 SETUP
14	004055	104203	07416i		1\$:	MOV #M576,R3	:GET MODE INDICATOR FOR 576
15	004057	100153			2\$:	MOV R3,(R5)	:SIGNAL DONE FORMAT
16	004060	104050	001740			MOV R5,BUFPNT	:STORE BUFFER POINTER
17	004062	104201	000030			MOV #F9,R1	:FCT WRITE OVERLAY
18	004064					CALL PAGE	:WRITE IT OUT
19	004066	114000	001743			CLR FCTCNT	:FOR FCTCNT INIT
20	004070	104300	002011	001573		MOV ST.XBN,CURXBN+1	:ALSO INITIALIZE XBN COUNTER
21	004073	104300	002011	001567		MOV ST.XBN,CURBN+1	:HIGH ORDER
22	004076	114000	001572			CLR CURXBN	:LOW ORDER ALWAYS 0
23	004100	114000	001566			CLR CURBN	:AND BLOCK NUMBER
24	004102	104203	001525			MOV #SCR,R3	:POINT TO CHARACTERISTICS
25	004104	104207	002110			MOV #CONBLK,R0	:POINT TO CONVERT BLOCK
26	004106	104632	000000			MOV CYLBN(R3),R2	:GET LOW ORDER CYLINDER
27	004110	100672	000000			MOV R2,V1(R0)	:STORE IT
28	004112	104632	000001			MOV CYLBN+1(R3),R2	:GET HIGH ORDER
29	004114	100672	000001			MOV R2,V1+1(R0)	:STORE IT
30	004116	104303	001612			MOV SECT12,R3	:GET SECTORS/TRACK (512)
31	004120	100673	000004			MOV R3,V3(R0)	:STORE IT
32	004122	114005			FCTCLP:	CLR R5	:CLEAR WRITE ERROR COUNT
33	004123	104050	001732			MOV R5,NEXT1	:CLEAR REPEAT COUNT
34	004125	104204	001410		FCTCL1:	MOV #TEMP,R4	:POINT TO BLOCK
35	004127	104300	001566	001410		MOV CURBN,TEMP	:FOR CONVERSION
36	004132	104300	001567	001411		MOV CURBN+1,TEMP+1	:DITTO
37	004135	107300	002011	001411		SUB ST.XBN,TEMP+1	:SUBTRACT STARTING XBN BITS
38	004140					CALL CS1	:CONVERT AND SEEK
39	004142	104207	001373			MOV #RDBLK,R0	:POINT TO COMMAND BLOCK
40	004144	104203	013400			MOV #RWCMD,R3	:GET READ COMMAND
41	004146	104302	001565			MOV CURTRK,R2	:GET CURRENT TRACK
42	004150	101023				BIS R2,R3	:SET TRACK FOR WRITE
43	004151	100673	000005			MOV R3,RW.CMD(R0)	:STORE IN COMMAND BLOCK
44	004153	104203	011607			MOV #PRMBUF,R3	:POINT TO BUFFER
45	004155	100673	000002			MOV R3,RW.BUF(R0)	:STICK IN COMMAND BLOCK
46	004157	104303	001566			MOV CURBN,R3	:GET LOW ORDER HEADER
47	004161	100673	000003			MOV R3,RW.LOW(R0)	:STORE IN WRITE BLOCK
48	004163	104303	001567			MOV CURBN+1,R3	:GET HIGH ORDER
49	004165	101203	120000			BIS #HD.XBN,R3	:SET HEADER
50	004167	100673	000004			MOV R3,RW.HI(R0)	:STORE IN WRITE BLOCK
51	004171	104203	001400			MOV #HSLIM-1,R3	:GET DUMMY SDI POINTER
52	004173	100673	000006			MOV R3,RW.DUM(R0)	:POINT IN COMMAND BLOCK
53	004175	104207	001373		READ9:	MOV #RDBLK,R0	:MAKE SURE POINTING AT BLOCK
54	004177	104203	100000			MOV #RDCMD,R3	:GET READ COMMAND
55	004201	104302	001412			MOV UNIT,R2	:GET PORT NUMBER FOR SIP
56	004203	100673	000000			MOV R3,RW.STAT(R0)	:STORE IT
57	004205	101207	100000			BIS #BIT15,R0	:SET NO REVECTORING

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 90-1
FINAL CHECK OVERLAY (H2)

58	004546	104673	000001		MOV	RW.ER1(R0),R3	:GET STATUS WORD
59	004550	102203	010000		BIT	#ECCF,R3	:ECC ERROR ?
60	004552				BEQ	101\$:NOPE - VERIFY EDC
61	004554	103200	010000	001374	BIC	#ECCF,RDBLK+RW.ER1	:CLEAR ECC ERROR BIT
62	004557				CALL	ECCCK	:CORRECT ECC
63	004561	115001			TST	R1	:TEST FLAG
64	004562				BNE	100\$:UNCORRECTABLE
65	004564	104202	011607	101\$:	MOV	#PRMBUF,R2	:POINT TO BUFFER
66	004566	104307	002171		MOV	SECSIZ,R0	:SECTOR SIZE IN WORDS
67	004570				CALL	CEDC	:COMPUTE EDC
68	004572	102200	020000	001703	BIT	#MODE,FLAG1	:WHAT MODE ARE WE IN
69	004575				BEQ	4\$:IF CLEAR THEN 512
70	004577	106623	000440		CMP	RW.E76(R2),R3	:O.K. ?
71	004601				BEQ	102\$:YUP - CONSIDER GOOD
72	004603				BR	100\$:NOPE - RETRY
73	004605	106623	000400	4\$:	CMP	RW.EDC(R2),R3	:O.K. ?
74	004607				BEQ	102\$:YUP - CONSIDER GOOD
75	004611	106300	002165	002167	100\$:	CMP	RETRY, TMPTRY
76	004614				BEQ	1\$:YES - TRY SOME RECOVERY
77	004616	115400	002167		INC	TMPTRY	:INC RETRY COUNT
78	004620				BR	READ10	:DO RETRY
79	004622	104303	002170	1\$:	MOV	RECTMP,R3	:GET CURRENT ERROR RECOVERY LEVEL
80	004624				BMI	RCTNGD	:IF NEGATIVE SKIP GOOD INCREMENT
81	004626	115000	002166		TST	RECOV	:IS THERE ONLY RECOVERY LEVEL 0 ?
82	004630				BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY
83	004632				CALL	ERRHND	:TRY RECOVERY
84	004634	114000	002167	3\$:	CLR	TMPTRY	:FOR INIT
85	004636	117400	002170		DEC	RECTMP	:DECREMENT IT
86	004640				BR	READ10	:RETRY
87	004642			102\$:			
88	004642			2\$:			
89	004642	115405			INC	R5	:YUP - INCREMENT COUNTER
90	004643	115400	001732	RCTNGD:	INC	NEXT1	:INCREMENT IT
91	004645	114000	002167		CLR	TMPTRY	:FOR RESET
92	004647	104300	002166	002170	MOV	RECOV,RECTMP	:GET RECOVERY LEVELS
93	004652	106205	000002		CMP	#2,R5	:FOUND 2 GOOD ONES ?
94	004654				BEQ	RCTCKD	:YUP - GO TO NEXT BLOCK
95	004656	104204	001566		MOV	#CURBN,R4	:FOR ADD
96	004660	104203	001727		MOV	#RCTFMT,R3	:FOR ADD
97	004662				CALL	DADD	:POINT TO NEXT COPY
98	004664	114000	002167		CLR	TMPTRY	:RESET RETRY LEVEL
99	004666	104300	002166	002170	MOV	RECOV,RECTMP	:DITTO RECOVERY LEVELS
100	004671	106300	001731	001732	CMP	FCTCPY,NEXT1	:DONE THIS SECTOR ?
101	004674				BNE	RCTCL1	:NO - READ NEXT FCT COPY
102	004676				BR	RCTCKE	:2 NOT GOOD - TROUBLE
103	004700	060022		RCTCKD:	XFC	UPDATE	:LET HOST KNOW STILL ALIVE
104	004701	115400	002164		INC	COUNT	:INCREMENT IT
105	004703				DUBINC	CURLBN	:INCREMENT IT
106	004711	104300	001570	001566	MOV	CURLBN,CURBN	:GET LOW ORDER
107	004714	104300	001571	001567	MOV	CURLBN+1,CURBN+1	:GET HIGH ORDER
108	004717	106300	001746	002164	CMP	RCTLBN,COUNT	:DONE ?
109	004722				BNE	RCTCLP	:NOPE - DO NEXT SECTOR
110	004724				RETURN		
111	004726	104201	000013	RCTCKE:	MOV	#11,R1	:SET ERROR CODE
112	004730	104302	002164		MOV	COUNT,R2	:RCT BLOCK FAILED ON
113	004732				CALL	ERRMNT	:ERROR RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 91
FINAL CHECK OVERLAY (H2)

```

1
2
3
4 004734 114000 002136 VERHD: CLR ERRCNT :CLEAR ERROR COUNT
5 004736 114000 001572 CLR CURXBN :CLEAR SECTOR NUMBER LOW
6 004740 114000 001573 CLR CURXBN+1 :CLEAR SECTOR NUMBER HIGH
7 004742 114000 001574 CLR STASEC :CLEAR STARTING SECTOR LOW
8 004744 114000 001575 CLR STASEC+1 :CLEAR STARTING SECTOR HIGH
9 004746 104200 014631 002157 MOV #BDLST,CURPNT :STORE POINTER TO BAD LIST
10 004751 104207 001525 MOV #SCR,R0 :POINT TO CHARACTERISTICS BLK
11 004753 104670 000002 002147 MOV GRPCYL(R0),GRPCNT :LOAD GROUPS/CYL
12 004756 103200 177400 002147 BIC #HIBYTE,GRPCNT :CLEAR OUT HIGH GARBAGE
13 004761 114000 002146 CLR CURGRP :CLEAR GROUP NUMBER
14 004763 104207 001525 MOV #SCR,R0 :POINT TO CHARACTERISTICS BLOCK
15 004765 104673 000003 MOV TRKGRP(R0),R3 :LOAD TRACKS/GROUP
16 004767 103203 177400 BIC #HIBYTE,R3 :CLEAR OUT HIGH GARBAGE
17 004771 104030 002150 MOV R3,TRKCNT :STORE IN COUNTER
18 004773 114000 001565 CLR CURTRK :RESET CURRENT TRACK NUMBER
19 004775 104673 000000 MOV CYLBN(R0),R3 :GET STARTING XBN CYLINDER
20 004777 104030 001604 MOV R3,CYLNUM :STORE IT
21 005001 104673 000001 MOV CYLBN+1(R0),R3 :GET HIGH ORDER
22 005003 104030 001605 MOV R3,CYLNUM+1 :STORE IT
23 005005 104300 001612 002137 MOV SECT12,SECCNT :LOAD SECTORS/TRACK (512)
24 005010 104300 001604 001551 4$: MOV CYLNUM,ISEEK+1 :GET LO ORDER WORD OF CYLINDER NUMBER
25 005013 104300 001605 001552 MOV CYLNUM+1,ISEEK+2 :LOAD HIGH ORDER WORD OF CYL NUM
26 005016 104300 002146 001553 MOV CURGRP,ISEEK+3 :LOAD GROUP NUMBER
27 005021 CALL SEEK :SEEK TO CURRENT CYL NUM
28 005023 115001 TST R1 :ANY ERRORS ?
29 005024 BMI SKERR :YUP - QUIT
30 005026 104207 001373 2$: MOV #RDBLK,R0 :POINT TO READ COMMAND BLOCK
31 005030 104203 011607 MOV #PRMBUF,R3 :BUFFER POINTER
32 005032 100673 000002 MOV R3,RW.BUF(R0) :STORE IT
33 005034 104303 001572 MOV CURXBN,R3 :GET LOW ORDER BLOCK NUMBER
34 005036 100673 000003 MOV R3,RW.LOW(R0) :STORE IN COMMAND BLOCK
35 005040 104303 001573 MOV CURXBN+1,R3 :LOAD HIGH ORDER BLOCK NUMBER
36 005042 105303 002011 ADD ST.XBN,R3 :ADD STARTING XBN BITS
37 005044 101203 120000 BIS #HD.XBN,R3 :SET IN HEADER CODE
38 005046 100673 000004 MOV R3,RW.HI(R0) :STORE IN COMMAND BLOCK
39 005050 104203 013400 5$: MOV #RWCMD,R3 :GET READ COMMAND
40 005052 101303 001565 BIS CURTRK,R3 :SET IN TRACK NUMBER
41 005054 100673 000005 MOV R3,RW.CMD(R0) :STORE IN COMMAND BLOCK
42 005056 104203 001400 MOV #HSLIM-1,R3 :POINTER TO DUMMY SDI BLOCK
43 005060 100673 000006 MOV R3,RW.DUM(R0) :STORE IT IN READ BLOCK
44 005062 104207 101373 MOV #<BIT15!RDBLK>,R0 :MAKE SURE POINTING AT BLOCK
45 005064 104203 100000 MOV #RDCMD,R3 :RESET STATUS POINTER
46 005066 100673 000000 MOV R3,RW.STAT(R0) :STORE IT BACK
47 005070 104302 001412 MOV UNIT,R2 :GET PORT NUMBER FOR SIP
48 005072 060012 XFC SIP :SYNCH WITH SECTOR/INDEX PULSE
49 005073 104202 000400 MOV #SECS16,R2 :SECTOR SIZE IN WORDS
50 005075 060002 XFC READ :READ 1 SECTOR
51 005076 115001 TST R1 :ANY ERROR ?
52 005077 BEQ 3$: NO - THIS HEAD O.K.
53 005101 102200 010000 001703 BIT #BDTST,FLAG1 :HAVE WE TESTED BAD HEADER CODE ?
54 005104 BNE 1$: YUP - DON'T TRY AGAIN
55 005106 104207 001373 MOV #RDBLK,R0 :POINT TO READ CONTROL BLOCK
56 005110 104673 000004 MOV RW.HI(R0),R3 :GET HEADER WORD
57 005112 103203 170000 BIC #HD.CLR,R3 :CLEAR HEADER

```


UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 91-2
FINAL CHECK OVERLAY (H2)

115	005310	104300	001575	001573	MOV	STASEC+1,CURXBN+1	:RESET HIGH ORDER
116	005313				DUBINC	CYLNUM	:INCREMENT CYLINDER NUMBER
117	005321	104300	001604	001551	MOV	CYLNUM,ISEEK+1	:GET LO ORDER WORD OF CYLINDER NUMBER
118	005324	104300	001605	001552	MOV	CYLNUM+1,ISEEK+2	:LOAD HIGH ORDER WORD OF CYL NUM
119	005327	104300	002146	001553	MOV	CURGRP,ISEEK+3	:LOAD GROUP NUMBER
120	005332				CALL	SEEK	:SEEK TO CURRENT CYL NUM
121	005334	115001			TST	R1	:ANY ERRORS ?
122	005335				BMI	SKERR	:YUP - QUIT
123	005337	104207	001373		11\$: MOV	#RDBLK,R0	:POINT TO READ COMMAND BLOCK
124	005341	104203	011607		MOV	#PRMBUF,R3	:BUFFER POINTER
125	005343	100673	000002		MOV	R3,RW.BUF(R0)	:STORE IT
126	005345	104303	001572		MOV	CURXBN,R3	:GET LOW RORDER BLOCK NUMBER
127	005347	100673	000003		MOV	R3,RW.LOW(R0)	:STORE IN COMMAND BLOCK
128	005351	104303	001573		MOV	CURXBN+1,R3	:LOAD HIGH ORDER BLOCK NUMBER
129	005353	105303	002011		ADD	ST.XBN,R3	:ADD STARTING XBN BITS
130	005355	101203	120000		BIS	#HD.XBN,R3	:SET IN XBN HEADER CODE
131	005357	100673	000004		MOV	R3,RW.HI(R0)	:STORE IN COMMAND BLOCK
132	005361	104203	013400		16\$: MOV	#RWCMD,R3	:GET READ COMMAND
133	005363	101303	001565		BIS	CURTRK,R3	:SET IN TRACK NUMBER
134	005365	100673	000005		MOV	R3,RW.CMD(R0)	:STORE IN COMMAND BLOCK
135	005367	104203	001400		MOV	#HSLIM-1,R3	:POINTER TO DUMMY SDI BLOCK
136	005371	100673	000006		MOV	R3,RW.DUM(R0)	:STORE IT IN READ BLOCK
137	005373	104207	101373		MOV	#<BIT15:RDBLK>,R0	:MAKE SURE POINTING AT BLOCK
138	005375	104203	100000		MOV	#RDCMD,R3	:RESET STATUS POINTER
139	005377	100673	000000		MOV	R3,RW.STAT(R0)	:STORE IT BACK
140	005401	104302	001412		MOV	UNIT,R2	:GET PORT NUMBER FOR SIP
141	005403	060012			XFC	SIP	:SYNCH WITH SECTOR/INDEX PULSE
142	005404	104202	000400		MOV	#SECSI6,R2	:SECTOR SIZE IN WORDS
143	005406	060002			XFC	READ	:READ 1 SECTOR
144	005407	115001			TST	R1	:ANY ERROR ?
145	005410				BEQ	10\$:NO - THIS HEAD O.K.
146	005412	102200	010000	001703	BIT	#BDTST,FLAG1	:HAVE WE TESTED BAD HEADER CODE ?
147	005415				BNE	17\$:YUP - DON'T TRY AGAIN
148	005417	104207	001373		MOV	#RDBLK,R0	:POINT TO READ CONTROL BLOCK
149	005421	104673	000004		MOV	RW.HI(R0),R3	:GET HEADER WORD
150	005423	103203	170000		BIC	#HD.CLR,R3	:CLEAR HEADER
151	005425	101203	110000		BIS	#HD.BAD,R3	:SET IN BAD HEADER CODE
152	005427	100673	000004		MOV	R3,RW.HI(R0)	:STORE IT BACK
153	005431	101200	010000	001703	BIS	#BDTST,FLAG1	:SET THAT WE TRIED BAD HEADER CODE
154	005434				BR	16\$:AND TRY READ AGAIN
155	005436	103200	010000	001703	17\$: BIC	#BDTST,FLAG1	:CLEAR BAD HEADER TESTED FLAG
156	005441				DUBINC	CURXBN	:INCREMENT CURRENT SECTOR
157	005447	117400	002137		DEC	SECCNT	:DECREMENT SECTOR/TRACK COUNT
158	005451				BNE	11\$:TRY IT AGAIN WITH NEXT SECTOR
159	005453	117400	002143		DEC	CNTCYL	:DECREMENT CYLINDER COUNT
160	005455				BEQ	12\$:FAILED ON ALL SECTORS - SEND WARNING
161	005457				BR	13\$:TRY NEXT CYLINDER
162	005461	103200	014631	001703	10\$: BIC	#BDLST,FLAG1	:CLEAR FLAG IN CASE IT WAS SET
163	005464	117400	002136		DEC	ERRCNT	:DECREMENT ERROR COUNT
164	005466				BNE	14\$:CONTINUE TILL DONE ALL
165	005470				15\$: RETURN		:ALL DONE RETURN
166	005472	104207	004243		12\$: MOV	#WRN,R0	:ADDRESS OF WARNING
167	005474	104201	000040		MOV	#WRNLN,R1	:LENGTH OF WARNING
168	005476	060016			XFC	MAINTR	:SEND IT
169	005477				BR	15\$:RETURN

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 93
 FCT WRITE OVERLAY (F9)

```

1          .SBTTL FCT WRITE OVERLAY (F9)
2 005570   DMOVLY F9,START
3          :
4          :
5          :
6          :
7 004014   104200 000030 001636 FCTWRT: MOV    #F9,CUROVL      ;FOR INIT
8 004017   114005          CLR    R5              ;CLEAR ERROR COUNTER
9 004020   104050 001732          MOV    R5,NEXT1        ;INIT NEXT COPY COUNTER
10 004022   104204 001743          MOV    #FCTCNT,R4     ;POINT TO FCT BLOCK NUMBER
11 004024   104203 002151          MOV    #ONE,R3        ;FOR SUB
12 004026          CALL   DSUB              ;SUB TO GET RIGHT ONE
13 004030   104305 001740          MOV    BUFPNT,R5      ;GET BUFFER POINTER
14 004032   104303 001626          FCTRLP: MOV   LBNCYL,R3  ;GET LBN CYLINDERS
15 004034   104207 002110          MOV    #CONBLK,R0     ;POINT TO CONVERT BLOCK
16 004036   100673 000000          MOV    R3,V1(R0)      ;STORE FOR CONVERT
17 004040   104303 001627          MOV    LBNCYL+1,R3    ;GET HIGH ORDER
18 004042   100673 000001          MOV    R3,V1+1(R0)    ;STORE IT
19 004044   104303 001612          MOV    SECT12,R3      ;GET SECTORS/TRACK (512)
20 004046   100673 000004          MOV    R3,V3(R0)      ;STORE IT
21 004050          CALL   CVTSK            ;CONVERT AND SEEK
22 004052   104052          MOV    R5,R2          ;POINT TO BUFFER
23 004053   104207 000400          MOV    #SECSI6,R0     ;SECTOR SIZE IN WORDS
24 004055          CALL   CEDC            ;COMPUTE EDC - RETURNED IN R3
25 004057   100623 000400          MOV    R3,RW.EDC(R2)  ;STORE IT IN THE BUFFER
26 004061   104207 001373          4$:  MOV    #WRBLK,R0   ;POINT TO COMMAND BLOCK
27 004063   100672 000002          MOV    R2,RW.BUF(R0)  ;STICK BUFFER PTR IN COMMAND BLOCK
28 004065   104203 122400          MOV    #WRCMD,R3      ;GET WRITE COMMAND
29 004067   104302 001565          MOV    CURTRK,R2      ;GET CURRENT TRACK
30 004071   101023          BIS    R2,R3          ;SET TRACK FOR WRITE
31 004072   100673 003005          MOV    R3,RW.CMD(R0)  ;STORE IN COMMAND BLOCK
32 004074   104143          MOV    (R4),R3        ;GET LOW ORDER HEADER
33 004075   100673 000003          MOV    R3,RW.LOW(R0)  ;STORE IN WRITE BLOCK
34 004077   104643 000001          MOV    1(R4),R3       ;GET HIGH ORDER
35 004101   105303 002011          ADD    ST.XBN,R3      ;ADD STARTING XBN BITS
36 004103   101203 120000          BIS    #HD.XBN,R3     ;SET HEADER
37 004105   100673 000004          MOV    R3,RW.HI(R0)   ;STORE IN WRITE BLOCK
38 004107   104203 001400          MOV    #HSLIM-1,R3    ;GET DUMMY SDI POINTER
39 004111   100673 000006          MOV    R3,RW.DUM(R0)  ;POINT IN COMMAND BLOCK
40 004113   104303 002005          WRITES: MOV   HPREA,R3  ;GET HEADER PREAMBLE
41 004115   104304 002006          MOV    DPREA,R4       ;GET DATA PREAMBLE
42 004117   104302 001412          MOV    UNIT,R2        ;GET PORT NUMBER FOR SIP
43 004121   104207 001373          MOV    #WRBLK,R0     ;MAKE SURE POINTING AT BLOCK
44 004123   101207 100000          BIS    #BIT15,R0     ;SET NO RVECTORING
45 004125   060012          XFC    SIP           ;WAIT FOR SECTOR PULSE
46 004126   104202 000400          MOV    #SECSI6,R2    ;SECTOR SIZE IN WORDS
47 004130   060003          XFC    WRITE         ;WRITE SECTOR
48 004131   115001          TST    R1            ;ANY ERROR ?
49 004132          BEQ    FWGOC          ;NOPE
50 004134   106300 002165 002167          CMP    RETRY,TMPTRY   ;MAX ?
51 004137          BEQ    1$           ;YES - TRY SOME RECOVERY
52 004141   115400 002167          INC    TMPTRY        ;INC RETRY COUNT
53 004143          BR    WRITES        ;DO RETRY
54 004145   104303 002170          1$:  MOV    RECTMP,R3     ;GET CURRENT ERROR RECOVERY LEVEL
55 004147          BMI    2$           ;IF NEGATIVE THEN FRIED
56 004151   115000 002166          TST    RECOV         ;IS THERE ONLY RECOVERY LEVEL 0 ?
57 004153          BEQ    3$           ;YES - NO NEED TO ISSUE IT - JUST RETRY

```


UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 94
 PBN->D,X,L,RBN CONVERSION OVERLAY (G5)

```

1          .SBTTL  PBN->D,X,L,RBN CONVERSION OVERLAY (G5)
2 004247  DMOVLY  G5,START
3          :
4          :
5          :   CONVERT FROM PBN TO D,X,L,RBN
6          :
7          PCON:  BIC      #HD.CLR,CURPBN+1      ;CLEAR THE HEADER
8          004014 103200 170000 001564  MOV      #HGHPBN,R3      ;HIGHEST PBN IN LBN AREA
9          004017 104203 001637          MOV      #CURPBN,R4      ;CURRENT PBN
10         004021 104204 001563          CALL     DCMP           ;IS IT LBN OR RBN ?
11         004023          BPL      LRBN           ;YUP - GO COMPUTE IT
12         004025          CALL     DSUB           ;SUBTRACT HIGH LBN PBN
13         004027          MOV      #XBNSEC,R3      ;TOTAL XBN SECTORS
14         004031 104203 001622          CALL     DCMP           ;IS IT AN XBN ?
15         004033          BMI      XBNFND        ;YUP - GO FIXIT
16         004035          MOV      #CURPBN,R4      ;ELSE DBN - GET VALUE
17         004037 104204 001563          MOV      #XBNSEC,R3      ;TOTAL XBN SECTORS
18         004041 104203 001622          CALL     DSUB           ;SUBTRACT TO GET RELATIVE DBN
19         004043          CALL     CONGRP        ;COMPUTE THE 'OFFSET' SECTOR
20         004045          MOV      #SECTRK,R3      ;SECTORS/TRACK
21         004047 104203 001606          MOV      #CURPBN,R4      ;TRACK
22         004051 104204 001563          CALL     DMUL          ;MULTIPLY TO GET STARTING PBN ON TRACK
23         004053          MOV      #DDUMMY,R3      ;SECTOR ON TRACK
24         004055 104203 001403          CALL     DADD          ;ADD TO GET ACTUAL PBN
25         004057          MOV      1(R4),R1      ;GET HIGH ORDER
26         004061 104641 000001          ADD      ST.DBN,R1      ;ADD TO GET ABSOLUTE DBN
27         004063 105301 002012          BIC      #HD.CLR,R1      ;CLEAR THE HEADER
28         004065 103201 170000          BIS      #HD.DBN,R1      ;MARK AS DBN
29         004067 101201 140000          MOV      R1,CURPBN+1    ;STORE BACK
30         004071 104010 001564          BR       PDONE          ;CLEAN UP AND RETURN
31         004073          XBNFND: CALL     CONGRP        ;COMPUTE THE 'OFFSET' SECTOR
32         004075          MOV      #SECTRK,R3      ;SECTORS/TRACK
33         004077 104203 001606          MOV      #CURPBN,R4      ;TRACK
34         004101 104204 001563          CALL     DMUL          ;MULTIPLY TO GET STARTING PBN ON TRACK
35         004103          MOV      #DDUMMY,R3      ;SECTOR ON TRACK
36         004105 104203 001403          CALL     DADD          ;ADD TO GET ACTUAL PBN
37         004107          MOV      1(R4),R1      ;GET HIGH ORDER
38         004111 104641 000001          ADD      ST.XBN,R1      ;ADD TO GET ABSOLUTE XBN
39         004113 105301 002011          BIC      #HD.CLR,R1      ;CLEAR HEADER
40         004115 103201 170000          BIS      #HD.XBN,R1      ;MARK AS XBN
41         004117 101201 120000          MOV      R1,CURPBN+1    ;STORE BACK
42         004121 104010 001564          BR       PDONE          ;CLEAN UP AND RETURN
43         :
44         :   LRBN:  CALL     CONGRP        ;COMPUTE THE OFFSET SECTOR
45         004123          MOV      #SCR,R5        ;POINT TO CHARACTERISTICS
46         004125 104205 001525          BIT      #MODE,FLAG1    ;WHAT MODE
47         004127 102200 020000 001703  BNE      1$            ;IF SET THEN 576
48         004131          MOV      LBNT12(R5),R3  ;GET LBN/TRACK FOR 512
49         004133          BR       2$            ;SKIP 576 SETUP
50         004135          1$:  MOV      LBNT76(R5),R3  ;GET LBN/TRACK FOR 576
51         004137 104653 000011          BIC      #HIBYTE,R3     ;CLEAR HIGH BYTE
52         004139          MOV      #DDUMMY,R1     ;POINT TO REMAINDER
53         004141          MOV      (R1),R4        ;GET IT
54         004143          CMP      R4,R3          ;COMPARE
55         004145          BMI      LBNFND        ;IF MINUS THEN LBN
56         004147 104653 000004          MOV      RBNTRK(R5),R3  ;GET RBN/TRACK
57         004149 103203 177600          BIC      #HI1BYTE,R3   ;CLEAR OUT GARBAGE
    
```

UD
 CR
 CR
 CR
 CS
 CS
 CS
 CS
 CS
 CS
 CS
 CU
 CU
 CU
 CU
 CU
 CU
 CU
 CU
 CU
 CU
 CU
 CU
 CU
 CU
 DA
 DA
 DA
 DA
 DA
 DA
 DA
 DA

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 94-1
 PBN->D,X,L,RBN CONVERSION OVERLAY (G5)

58	004160	104030	001410		MOV	R3,TEMP		:STORE IT
59	004162	114000	001411		CLR	TEMP+1		:FOR STORE
60	004164	104203	001410		MOV	#TEMP,R3		:FOR MULTIPLY
61	004166	104204	001563		MOV	#CURPBN,R4		:DITTO - NUM OF TRACKS
62	004170				CALL	DMUL		:MULTIPLY BY TRACK NUMBER
63	004172	104204	001403		MOV	#DDUMMY,R4		:FOR SUBTRACT
64	004174	102200	020000	001703	BIT	#MODE,FLAG1		:WHAT MODE
65	004177				BNE	3\$:IF SET THEN 576
66	004201	104653	000011		MOV	LBNT12(R5),R3		:GET LBN/TRACK FOR 512
67	004203				BR	4\$:SKIP 576 SETUP
68	004205	104653	000015	3\$:	MOV	LBNT76(R5),R3		:GET LBN/TRACK FOR 576
69	004207	103203	177400	4\$:	BIC	#HIBYTE,R3		:CLEAR HIGH BYTE
70	004211	104030	001410		MOV	R3,TEMP		:STORE IT
71	004213	114000	001411		CLR	TEMP+1		:FOR CLEAR
72	004215	104203	001410		MOV	#TEMP,R3		:POINT FOR SUBTRACT
73	004217				CALL	DSUB		:SUBTRACT TO GET RESIDUE RBN
74	004221	104204	001563		MOV	#CURPBN,R4		:TO GET RBN NUMBER
75	004223	104203	001403		MOV	#DDUMMY,R3		:DITTO
76	004225				CALL	DADD		:GIVES RELATIVE RBN
77	004227	104641	000001		MOV	1(R4),R1		:GET HIGH ORDER
78	004231	105301	002010		ADD	ST.RBN,R1		:ADD TO GET ABSOLUTE RBN
79	004233	103201	170000		BIC	#HD.CLR,R1		:CLEAR TH EHEADER
80	004235	101201	060000		BIS	#HD.RBN,R1		:SET AS A RBN
81	004237	104010	001564		MOV	R1,CURPBN+1		:STORE BACK
82	004241				BR	PDONE		:CLEAN UP AND RETURN
83					:			
84					:			
85	004243	104204	001563		LBNFND: MOV	#CURPBN,R4		:MULT NUM OF TRACKS
86	004245	102200	020000	001703	BIT	#MODE,FLAG1		:WHAT MODE
87	004250				BNE	1\$:IF SET THEN 576
88	004252	104653	000011		MOV	LBNT12(R5),R3		:GET LBN/TRACK FOR 512
89	004254				BR	2\$:SKIP 576 SETUP
90	004256	104653	000015	1\$:	MOV	LBNT76(R5),R3		:GET LBN/TRACK FOR 576
91	004260	103203	177400	2\$:	BIC	#HIBYTE,R3		:CLEAR HIGH BYTE
92	004262	104030	001410		MOV	R3,TEMP		:STORE IT
93	004264	114000	001411		CLR	TEMP+1		:FOR CLEAR
94	004266	104203	001410		MOV	#TEMP,R3		:POINT FOR MULT
95	004270				CALL	DMUL		:GET LBN'S
96	004272	104203	001403		MOV	#DDUMMY,R3		:PLUS RESIDUE
97	004274				CALL	DADD		:GIVES LBN NUMBER
98	004276	104207	001525		MOV	#SCR,R0		:POINT TO CHARACTERISTICS
99	004300	104641	000001		MOV	1(R4),R1		:GET HIGH ORDER
100	004302	105301	002007		ADD	ST.LBN,R1		:ADD TO GET ABSOLUTE LBN
101	004304	103201	170000		BIC	#HD.CLR,R1		:CLEAR HEADER
102	004306	101201	000000		BIS	#HD.LBN,R1		:SET AS LBN
103	004310	104010	001564		MOV	R1,CURPBN+1		:STORE BACK
104	004312	104300	001563	001566	PDONE: MOV	CURPBN,CURBN		:GET LOW ORDER
105	004315	104300	001564	001567	MOV	CURPBN+1,CURBN+1		:HIGH ORDER
106	004320				RETURN			

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 95
 PBN->D,X,L,RBN CONVERSION OVERLAY (G5)

```

1
2
3
4 004322 104300 001606 001403 CONGRP: MOV SECTRK,DDUMMY ;GET SECTORS/TRACK
5 004325 114000 001404 CLR DDUMMY+1 ;CLEAR HIGH ORDER
6 004327 104203 001403 MOV #DDUMMY,R3 ;FOR DIVIDE
7 004331 104204 001563 MOV #CURPBN,R4 ;DITTO
8 004333 CALL DDIV ;DIVIDE PBN/SECTRK TO GET TRACK #
9 004335 104141 MOV (R4),R1 ;GET CURRENT TRACK
10 004336 PUSH R1 ;SAVE IT ON STACK
11 004337 104642 000001 MOV 1(R4),R2 ;DITTO HIGH ORDER
12 004341 PUSH R2 ;SAVE AGAIN
13 004342 115001 TST R1 ;IS LOW ORDER TRACK 0 ?
14 004343 BNE 1$ ;NOPE - TRACK # CAN'T BE 0
15 004345 115002 TST R2 ;IS HIGH ORDER TRACK 0 ?
16 004346 BEQ 5$ ;YES - NO OFFSET OR NEED TO GO ON
17 004350 104131 1$: MOV (R3),R1 ;GET CURRENT SECTOR NUMBER
18 004351 PUSH R1 ;SAVE IT
19 004352 104207 001525 MOV #SCR,R0 ;POINT TO CHARACTERISTICS BLOCK
20 004354 104671 000003 MOV TRKGRP(R0),R1 ;LOAD TRACKS/GROUP
21 004356 103201 177400 BIC #HIBYTE,R1 ;CLEAR OUT HIGH GARBAGE
22 004360 100131 MOV R1,(R3) ;STORE TRACK/GROUP IN DDUMMY
23 004361 CALL DDIV ;DIVIDE TO GET GROUP NUMBER
24 004363 104141 MOV (R4),R1 ;GET ABSOLUTE GROUP NUMBER
25 004364 BNE 3$ ;CAN'T BE 0 - CONTINUES
26 004366 104641 000001 MOV 1(R4),R1 ;GET HIGH ORDER
27 004370 BEQ 4$ ;GROUP IS 0 - NO OFFSET
28 004372 104207 001525 3$: MOV #SCR,R0 ;POINT TO CHARACTERISTICS
29 004374 104671 000002 MOV GRPCYL(R0),R1 ;GET GROUPS/CYLINDER
30 004376 103201 177400 BIC #HIBYTE,R1 ;CLEAR OUT GARBAGE
31 004400 100131 MOV R1,(R3) ;STORE GROUPS/CYLINDER
32 004401 CALL DDIV ;DIVIDE TO GET RELATIVE GROUP
33 004403 104131 MOV (R3),R1 ;GET GROUP NUMBER
34 004404 BEQ 4$ ;IF ZERO THEN DONE - NO OFFSET
35 004406 104207 001525 MOV #SCR,R0 ;POINT TO CHARACTERISTICS
36 004410 102200 020000 001703 BIT #MODE,FLAG1 ;WHAT MODE ARE WE IN
37 004413 BEQ 6$ ;IF CLEAR THEN 512
38 004415 104632 000015 MOV OFFS76(R3),R2 ;GET GROUP OFFSET (576)
39 004417 BR 7$ ;SKIP 512
40 004421 104632 000011 6$: MOV OFFS12(R3),R2 ;GET GROUP OFFSET (512)
41 004423 110702 7$: SWAB R2 ;GET INTO LOWBYTE
42 004424 103202 177400 BIC #HIBYTE,R2 ;CLEAR HIGH GARBAGE
43 004426 115002 TST R2 ;ANY OFFSET ?
44 004427 BEQ 4$ ;NO - NO NEED TO FIX UP
45 004431 100142 MOV R2,(R4) ;STORE FOR MULT
46 004432 114002 CLR R2 ;CLEAR FOR STORE
47 004433 100642 000001 MOV R2,1(R4) ;CLEAR HIGH ORDER
48 004435 CALL DMUL ;MULTIPLY GROUP*OFFSET
49 004437 POP R2 ;RESTORE ORIGINAL SECTOR
50 004440 104145 MOV (R4),R5 ;GET TOTAL OFFSET
51 004441 105025 ADD R2,R5 ;ADD TO GET SECTOR NUMBER
52 004442 100145 MOV R5,(R4) ;STORE IT
53 004443 114002 CLR R2 ;FOR STROE
54 004444 100642 000001 MOV R2,1(R4) ;CLEAR HIGH ORDER
55 004446 104301 001606 MOV SECTRK,R1 ;SECTOR'S/TRACK
56 004450 100131 MOV R1,(R3) ;FOR MOD FUNCTION
57 004451 100632 000001 MOV R2,1(R3) ;CLEAR HIGH ORDER

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 95-1
 PBN->D,X,L,RBN CONVERSION OVERLAY (G5)

58	004453										
59	004455			5\$:	CALL	DDIV				:	REMAINDER IS NEW SECTOR NUMBER
60	004456	100641	000001		POP	R1				:	RESTORE TRACK NUMBER LOW
61	004460				MOV	R1,1(R4)				:	STORE IT
62	004461	100141			POP	R1				:	RESTORE TRACK NUMBER HIGH
63	004462				MOV	R1,(R4)				:	STORE IT
64	004464				RETURN						
65	004465	100131		4\$:	POP	R1				:	SECTOR NUMBER
66	004466	114001			MOV	R1,(R3)				:	STORE IT
67	004467	100631	000001		CLR	R1				:	FOR STORE
68	004471				MOV	R1,1(R3)				:	CLEAR HIGH ORDER
					BR	5\$:	AND EXIT

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 96
ERROR MESSAGE OVERLAY (G6)

```

1          .SBTTL  ERROR MESSAGE OVERLAY (G6)
2
3          :
4          :      ERROR MESSAGE OVERLAY
5          :
6          :      DMOVLY  G6,START
7          :
8          :      MOV      #DMBUF,R0          ;POINT TO BUFFER
9          :      MOV      1(R0),R1          ;GET SUB CODE
10         :      MOV      (R0),R2          ;GET ERROR NUMBER
11         :      DEC      R2                ;MAKE RELATIVE TO 0
12         :      ROL      R2                ;MULTIPLY BY 2
13         :      ADD      #ERRTBL,R2       ;OFFSET TO MESSAGE ADDRESS
14         :      MOV      (R2),R0          ;MESSAGE TO SEND
15         :      MOV      1(R2),R1        ;LENGTH OF MESSAGE
16         :      XFC      MAINTR          ;SEND IT
17         :      RETURN                   ;GO BACK AND DIE
18
19         :      .ENABL  LC
20         :      M.ER1: .WORD  50001
21         :      .asciz  'GET STATUS failure'
22         :      *****
23         :      M.ER2: .WORD  50002
24         :      .asciz  'SDI send error'
25         :      *****
26         :      M.ER3: .WORD  50003
27         :      .asciz  'Unsuccessful SDI command'
28         :      *****
29         :      M.ER4: .WORD  50004
30         :      .asciz  'SDI receive error'
31         :      *****
32         :      M.ER5: .WORD  50005
33         :      .asciz  'UNIBUS I/O error'
34         :      *****
35         :      M.ER6: .WORD  50006
36         :      .asciz  'Formatter initialization error'
37         :      *****
38         :      M.ER7: .WORD  50007
39         :      .asciz  'Non-existent unit number'
40         :      *****
41         :      M.ER8: .WORD  50010
42         :      .asciz  'DBN/XBN format error (FORMAT XFC failed)'
43         :      *****
44         :      M.ER9: .WORD  50011
45         :      .asciz  'FCT check error'
46         :      *****
47         :      M.ER10: .WORD  50012
48         :      .asciz  'SEEK error'
49         :      *****
50         :      M.ER11: .WORD  50013
51         :      .asciz  'RCT check error'
52         :      *****
53         :      M.ER12: .WORD  50014
54         :      .asciz  'LBN format error (FORMAT XFC failed)'
55         :      *****
56         :      M.ER13: .WORD  50015
57         :      .asciz  'FCT write error'
58         :      *****

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 96-1
 ERROR MESSAGE OVERLAY (G6)

58	004274	050016			M.ER14: .WORD 50016
59	004275	122	103	124	.asciz 'RCT read error'
60					*****
61	004305	050017			M.ER15: .WORD 50017
62	004306	122	103	124	.asciz 'RCT write error'
63					*****
64	004316	050020			M.ER16: .WORD 50020
65	004317	122	103	124	.asciz 'RCT full'
66					*****
67	004324	050021			M.ER17: .WORD 50021
68	004325	106	103	124	.asciz 'FCT read error'
69					*****
70	004335	050022			M.ER18: .WORD 50022
71	004336	106	103	124	.asciz 'FCT non-existent'
72					*****
73	004347	050023			M.ER19: .WORD 50023
74	004350	106	103	124	.asciz 'FCT Down Line Load error (FCT block not avbl.)'
75					*****
76	004400	050024			M.ER20: .WORD 50024
77	004401	104	162	151	.asciz 'Drive init timeout'
78					*****
79	004413	050025			M.ER21: .WORD 50025
80	004414	111	156	166	.asciz 'Invalid response to question'
81					*****
82	004433	050026			M.ER22: .WORD 50026
83	004434	104	162	151	.asciz 'Drive does not support 576 format on this media'
84					*****
85	004464				M.ER23:
86					.DSABL LC

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 97
 ERROR MESSAGE OVERLAY (G6)

```

1
2
3
4 004464 004033
5 004465 000013
6 004466 004046
7 004467 000011
8 004470 004057
9 004471 000016
10 004472 004075
11 004473 000012
12 004474 004107
13 004475 000012
14 004476 004121
15 004477 000021
16 004500 004142
17 004501 000016
18 004502 004160
19 004503 000026
20 004504 004206
21 004505 000011
22 004506 004217
23 004507 000007
24 004510 004226
25 004511 000011
26 004512 004237
27 004513 000024
28 004514 004263
29 004515 000011
30 004516 004274
31 004517 000011
32 004520 004305
33 004521 000011
34 004522 004316
35 004523 000006
36 004524 004324
37 004525 000011
38 004526 004335
39 004527 000012
40 004530 004347
41 004531 000031
42 004532 004400
43 004533 000013
44 004534 004413
45 004535 000020
46 004536 004433
47 004537 000031
48 004540
49 000001

:
:
:
ERRTBL: .WORD M.ER1          ;MESSAGE 1
          .WORD M.ER2-M.ER1 ;LENGTH OF MESSAGE 1
          .WORD M.ER2          ;MESSAGE 2
          .WORD M.ER3-M.ER2   ;LENGTH OF MESSAGE 2
          .WORD M.ER3          ;MESSAGE 3
          .WORD M.ER4-M.ER3   ;LENGTH OF MESSAGE 3
          .WORD M.ER4          ;MESSAGE 4
          .WORD M.ER5-M.ER4   ;LENGTH OF MESSAGE 4
          .WORD M.ER5          ;MESSAGE 5
          .WORD M.ER6-M.ER5   ;LENGTH OF MESSAGE 5
          .WORD M.ER6          ;MESSAGE 6
          .WORD M.ER7-M.ER6   ;LENGTH OF MESSAGE 6
          .WORD M.ER7          ;MESSAGE 7
          .WORD M.ER8-M.ER7   ;LENGTH OF MESSAGE 7
          .WORD M.ER8          ;MESSAGE 8
          .WORD M.ER9-M.ER8   ;LENGTH OF MESSAGE 8
          .WORD M.ER9          ;MESSAGE 9
          .WORD M.ER10-M.ER9  ;LENGTH OF MESSAGE 9
          .WORD M.ER10         ;MESSAGE 10
          .WORD M.ER11-M.ER10 ;LENGTH OF MESSAGE 10
          .WORD M.ER11         ;MESSAGE 11
          .WORD M.ER12-M.ER11 ;LENGTH OF MESSAGE 11
          .WORD M.ER12         ;MESSAGE 12
          .WORD M.ER13-M.ER12 ;LENGTH OF MESSAGE 12
          .WORD M.ER13         ;MESSAGE 13
          .WORD M.ER14-M.ER13 ;LENGTH OF MESSAGE 13
          .WORD M.ER14         ;MESSAGE 14
          .WORD M.ER15-M.ER14 ;LENGTH OF MESSAGE 14
          .WORD M.ER15         ;MESSAGE 15
          .WORD M.ER16-M.ER15 ;LENGTH OF MESSAGE 15
          .WORD M.ER16         ;MESSAGE 16
          .WORD M.ER17-M.ER16 ;LENGTH OF MESSAGE 16
          .WORD M.ER17         ;MESSAGE 17
          .WORD M.ER18-M.ER17 ;LENGTH OF MESSAGE 17
          .WORD M.ER18         ;MESSAGE 18
          .WORD M.ER19-M.ER18 ;LENGTH OF MESSAGE 18
          .WORD M.ER19         ;MESSAGE 19
          .WORD M.ER20-M.ER19 ;LENGTH OF MESSAGE 19
          .WORD M.ER20         ;MESSAGE 20
          .WORD M.ER21-M.ER20 ;LENGTH OF MESSAGE 20
          .WORD M.ER21         ;MESSAGE 21
          .WORD M.ER22-M.ER21 ;LENGTH OF MESSAGE 21
          .WORD M.ER22         ;MESSAGE 22
          .WORD M.ER23-M.ER22 ;LENGTH OF MESSAGE 22
          DMEND
          .END

```

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 9r-1

SYMBOL TABLE

ACC	001477	CHRDNE=	010000	CURBN	001566	DLL	=	000400	EXTR11	005461		
ACCESS	003042	CKERR	005516	CURGRP	002146	DLLDN	005123	FBDHD	=	010000		
AGAIN	004550	CKR	005277	CURLBN	001570	DLLDN1	005124	FBEGIN	004314			
ALLOVR	003474	CKR1	005560	CUROVL	001636	DLLFLE	005103	FBEG2	004330			
ALLOV1	003503	CKSS	005477	CURPBN	001563	DLLNO	005126	FCFXLP	004214			
AOUT	003222	CLBUF	=	015306	CURPNT	002157	DLLRET	004451	FCLR	=	177760	
ATTN	=	CLEAR	003052	CURRBN	001561	DLLRT1	004435	FCMSG	020003			
ATTN1	003207	CLEDON	004153	CURTRK	001565	DMBUF	001750	FCNT	001722			
BADEDC	002133	CLELP	004054	CURXBN	001572	DMBUFL	=	000016	FCPG	004713		
BADE76	002135	CLFLP2	004043	CUTOF	001720	DMUL	002232	FCTAVL	=	000001		
BADPBN	001706	CLESKP	004115	CVT	=	000020	DOLBN	004027	FCTBAD	=	000004	
BADRBN	004713	CLEWRT	004774	CVTERR	003667	DONDLL	004455	FCTCK	004027			
BD	=	CLHERE	005007	CVTSK	003612	DONE	=	000021	FCTCKD	004330		
BDHD	=	CLRBUF	004665	CYL	=	000006	DONFCT	004663	FCTCKE	004363		
BDIRCT	005262	CLRLP	004673	CYLBN	=	000000	DONONE	004611	FCTCLP	004122		
BDLST	=	CLSKP2	004142	CYLNUM	001604	DPBN	004321	FCTCL1	004125			
BDTST	=	CLSKP3	004162	C512	=	000016	DPREA	002006	FCTCNT	001743		
BIT0	=	CLSKP4	004160	C576	=	000020	DSUB	002212	FCTCPY	001731		
BIT1	=	CMBUF	=	013022	DADD	002172	DUPOVL	=	00362	FCTEMT	=	000002
BIT10	=	CMPDAT	=	000006	DADD1	002202	DWRD	002131	FCTFLG	=	000025	
BIT11	=	CNLP	005164	DASH	=	000055	DXBN	004031	FCTFMT	001725		
BIT12	=	CNLP1	005162	DATA	=	000005	DXCH	004477	FCTMOT	004231		
BIT13	=	CNT	001734	DATAGN	004700	DXCHEC	004476	FCTMPD	001745			
BIT14	=	CNTCYL	002143	DATBUF	004255	DXERR	004374	FCTPTR	001742			
BIT15	=	CONBAD	005463	DATCON	004516	DXFCPG	004376	FCTRCT	004201			
BIT2	=	CONBLK	002110	DATE	001766	DXFCP1	004411	FCTREV	001776			
BIT3	=	CONDON	005260	DATERR	004770	DXFORM	004067	FCTRLP	004032			
BIT4	=	CONDO1	005262	DATLP	004530	DXTRK	004017	FCTSKP	004517			
BIT5	=	CONERR	005264	DATLP1	004633	EAGAIN	003522	FCTSK1	004511			
BIT6	=	CONER1	005270	DATLP2	004662	ECC	=	000015	FCTSLP	004711		
BIT7	=	CONEXT	005477	DATRET	004675	ECCCK	003763	FCTSP	005044			
BIT8	=	CONGRP	004322	DATRT1	004676	ECCF	=	010000	FCTSUB	001723		
BIT9	=	CONINT	004033	DATVER	004712	ECHO	=	000010	FCTSZ	=	000010	
BLANW	=	CONLOW	005253	DATVL1	004755	EDC	002132	FCTUSD	004215			
BLKFND	004212	CONON	004454	DATVL2	004766	EDC76	002134	FCTWRT	004014			
BMAX	=	CONTEX	005502	DAYS	=	036031	EIMAGE	001714	FCWERR	004242		
BOTTOM	004306	COUNT	002164	DBBAD	002001	EMAX	001710	FDAT	=	000012		
BRBN	005150	CR	001512	DBBUF	=	004441	ENTRY	001365	FDLL	004414		
BREAK	=	CR.ACC	001440	DBLEN	=	000034	EORCT	004156	FERR	004363		
BSTGS	=	CR.CLR	001444	DBN	=	000010	ERCV	=	000002	FIDANS	005412	
BUFMSK	=	CR.DIS	001430	DBNCYL	=	000022	ERDN	=	000010	FIDMUL	005427	
BUFPNT	001740	CR.ERV	001460	DCLR	001501	ERECOV	001555	FILLIT	004736			
BUF1	=	CR.GCR	001420	DCMP	002333	ERFLAG	001704	FINCHK	004014			
BUF10	=	CR.GSR	001424	DCMP1	002362	ERLEN	=	000002	FINDON	005407		
BUF11	=	CR.GST	001414	DCMP2	002354	ERPNT	001737	FINI	=	040000		
BUF2	=	CR.ONL	001464	DCMP3	002372	ERR	001711	FINLEN	005367			
BUF3	=	CR.RCL	001454	DCMP4	002346	ERRBUF	001707	FINLN1	005371			
BUF4	=	CR.RUN	001434	DDIV	002266	ERRCNT	002136	FINMSG	004203			
BUF5	=	CR.SEK	001450	DDUMMY	001403	ERRHND	003504	FIXBLK	004535			
BUF6	=	CS	005220	DEAD	=	000020	ERRMNT	003451	FIXFCT	005112		
BUF7	=	CSKIP	004523	DECALP	004707	ERROR	004473	FIXIT	005133			
BUF8	=	CSKIP1	004536	DECASC	004703	ERRSYM	=	000002	FIXLP	004552		
BUF9	=	CSKIP2	004534	DINIT	=	000011	ERRT	002657	FKIP1	005206		
CBUF	002125	CSKIP3	005045	DIS	001474	ERRTBL	004464	FKIP10	005236			
CDONE	005130	CSKIP6	005002	DISCON	003154	ER1	005023	FKIP2	005172			
CEDC	003515	CSKIP7	005053	DLERR	004060	EXTFCT	005434	FKIP9	005665			
CHAR	002533	CS1	005501	DLERT	004476	EXTRET	005460	FKP1	005226			

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 97-2
 SYMBOL TABLE

FLAG	001702	G4	= 000041	LBNH76=	000016	LOOP1	002565	M.ER10	004217
FLAG1	001703	G5	= 000044	LBNLBN	001616	LOOP2	002614	M.ER11	004226
FLIP	004114	G6	= 000063	LBNPCY	001630	LOOP3	004462	M.ER12	004237
FLIPON=	000400	G7	= 000047	LBNT12=	000011	LOVER	005426	M.ER13	004263
FLKIP1	005635	G8	= 000052	LBNT76=	000015	LOVER1	005554	M.ER14	004274
FLKIP2	005621	HASH	004016	LBUFE =	004311	LOVER2	005512	M.ER15	004305
FLKP1	005655	HD.BAD=	110000	LCHEC	004461	LOVER3	005503	M.ER16	004316
FMSTL =	000010	HD.CLR=	170000	LDONE	005560	LOVER4	005467	M.ER17	004324
FMTSTA	004236	HD.CUR	002145	LEN =	000000	LOVER5	005442	M.ER18	004335
FODONE	004227	HD.DBN=	140000	LERR	004610	LPBN	004176	M.ER19	004347
FOLOOP	004025	HD.LBN=	000000	LER1	005007	LRBN	004125	M.ER2	004046
FORERR	004446	HD.PRV=	050000	LER2	005020	LREDO	005653	M.ER20	004400
FORMAT=	000001	HD.RBN=	060000	LER3	005004	LSKIP	004505	M.ER21	004413
FPRIM =	040000	HD.REV=	030000	LFCTNT=	000012	LSKIP1	004520	M.ER22	004433
FRCPY =	000001	HD.XBN=	120000	LFCTUS=	000014	LSKIP2	004516	M.ER23	004464
FRDONE	005030	HEAD =	000005	LFERR	004457	LSKIP3	005355	M.ER3	004057
FRSKP	005023	HERE	004667	LFINMS=	000012	LSKIP4	004213	M.ER4	004075
F SER =	000002	HGHPBN	001637	LFIXIT	005562	LSKIP6	004767	M.ER5	004107
FT.BUF=	000000	HIBYTE=	177400	LFORM	004014	LSKIP7	005363	M.ER6	004121
FT.FLA=	000003	HI1BYT=	177600	LHERE	004651	LSKIP8	004701	M.ER7	004142
FT.HI =	000002	HI2BYT=	177700	LKIP10	004631	LSKIP9	005340	M.ER8	004160
FT.LOW=	000001	HKIP	004025	LKIP12	004407	LSND	005314	M.ER9	004206
FWGOOD	004166	HKIP1	004053	LKIP2	004064	LTO	001735	M512 =	126736
FWRD	002126	HOLD	001712	LKIP20	004156	LTRK	004014	M576 =	074161
FWTDON	004232	HOLDBN	001576	LKIP21	004177	L576	004311	N	002140
F1 =	000000	HOLDPN	001602	LKIP22	004335	MAINTR=	000016	NDLL =	004000
F2 =	000003	HOLRBN	001600	LKIP23	004306	MAINTW=	000017	NEXT	003262
F3 =	000006	HPREA	002005	LKIP24	004320	MANU =	000200	NEXT1	001732
F4 =	000011	HSLIM	001401	LKIP25	004424	MARBAD	004572	NEXT5	003267
F5 =	000014	HSTHI =	000002	LKIP27	004345	MASK =	000000	NGD	004264
F6 =	000017	HSTLO =	000001	LKIP28	004635	MAXTRY=	000010	NGD1	004305
F7 =	000022	H1 =	000055	LKIP29	004615	MENTLN=	000002	NN1	002142
F8 =	000025	H2 =	000066	LKIP30	004674	MLEN	004260	NO	004342
F9 =	000030	IMAGE =	015763	LKIP31	004677	MNCNT	001747	NOCERR	005223
GCR	001471	IMLEN =	000003	LKIP33	004655	MODE =	020000	NOCRY	003535
GDBLK =	011132	IMSTAR	002004	LKIP4	004605	MORE	005147	NODLL	004063
GDECC	004012	INDSEC=	000013	LKIP5	004547	MSGLOP	004365	NOERR	004647
GDONE	005440	INI	001733	LKIP7	004654	MSGLP2	004437	NOFACT =	100000
GENCON	005144	INIRCT=	020000	LKIP8	004150	MSGLP3	004444	NOGOOD	004321
GETUNT	005260	INITDD	003255	LKIP9	004133	MSGLP4	004435	NOINC	004123
GOBAD =	000020	INITIT	003174	LLEN =	000015	MSGOFF=	000001	NOLBN	004544
GOVER	005275	INITL	005522	LMORE	005576	MSGTBL	004016	NOSEK	004356
GOVER1	005267	INITPT	003233	LNOERR	004631	MSG1	002605	NOTHER	005454
GRP =	000010	INITP1	003240	LO =	007777	MSG1LN=	000016	NOTR	004205
GRPCNT	002147	INIT5	003203	LOAD	002756	MSG2LN=	000022	NOTY	005431
GRPCYL=	000002	INPEL =	000007	LOADER	003035	MSG3LN=	000016	NO576	005514
GSKIP	005414	INPERR	004246	LOAD2	003006	MSG4LN=	000022	NUM	002124
GSKIP1	005320	INST =	000001	LOAD3	003012	MSG5LN=	000026	NUMDBN	004414
GSKIP2	005341	IRECAL	001554	LOAD4	003031	MSG6LN=	000035	NUMLBN	004057
GSKIP3	005363	ISEEK	001550	LOAD5	002763	MSG7LN=	000016	NUMRBN	004133
GSR	001472	ISKIP	004703	LOBL =	000040	MULDN	005310	NUMXBN	004447
GST	001470	LAGAIN	004532	LOBYTE=	000377	MULERR	005312	N1	002141
GSTATS	002377	LAST =	100000	LOK	005351	MULER1	005363	OCDONE	004254
GTFLAG=	002000	LBNBAD	001777	LONGTO=	000001	MULPC	001405	OCLoop	004064
G1 =	000060	LBNCYL	001626	LOOP	004115	MULT10	005316	OERR	003301
G2 =	000033	LBNFND	004243	LOOPP	004204	MULT2	005276	OERR2	003312
G3 =	000036	LBNH12=	000012	LOOPP2	004154	M.ER1	004033	OFATAL	004370

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 97-3
SYMBOL TABLE

OFFSET	001410	OVL...=	025760	RBNTRK=	000004	RDONE1	004257	RW.BUF=	000002
OFFS12=	000011	OVRLAY	003411	RBNWRT	005307	RDONE2	004303	RW.CMD=	000005
OFFS76=	000015	OVS.F1=	015156	RB.CMD=	000002	READ =	000002	RW.DAT=	000000
OK	005041	OVS.F2=	020714	RB.HI =	000001	READX	004713	RW.DUM=	000006
OLDONE	004364	OVS.F3=	027330	RB.IM =	000003	READ1	004600	RW.EDC=	000400
ONE	002151	OVS.F4=	030502	RB.LOW=	000000	READ10	004526	RW.ER1=	000001
ONLIN	003164	OVS.F5=	033512	RCBUFE=	004403	READ11	004120	RW.ER2=	000401
ONLINE	001557	OVS.F6=	040056	RCD	006010	READ2	004751	RW.E76=	000440
OQUIT	004415	OVS.F7=	036534	RCFIX	004506	READ3	004562	RW.HI =	000004
ORFTAL	004414	OVS.F8=	024454	PCFXLP	005377	READ4	004736	RW.LOW=	000003
OVCNT =	000023	OVS.F9=	051676	RCINDN=	000100	READ7	004104	RW.STA=	000000
OVER	004572	OVS.G1=	011674	RCINER	004530	READ8	005654	SBUFE =	004353
OVER1	005124	OVS.G2=	041074	RCINIT=	000040	READ9	004175	SCR	001525
OVER2	005065	OVS.G3=	041222	RCINLP	004165	REBUFE=	004535	SECCN	005016
OVE.F1=	004014	OVS.G4=	042040	RCL	005620	RECAL	002751	SECCNT	002137
OVE.F2=	004014	OVS.G5=	052366	RCLN =	000034	RECIR =	040000	SECNDY	004265
OVE.F3=	004014	OVS.G6=	053526	RCLP	004024	RECOV	002166	SECNO	005031
OVE.F4=	004014	OVS.G7=	017640	RCLP2	004170	RECTMP	002170	SECSIZ	002171
OVE.F5=	004014	OVS.G8=	026326	RCLP3	004516	REDO	005224	SECSI6=	000400
OVE.F6=	004014	OVS.H1=	032476	RCLP4	004444	RELEN =	000025	SECSI8=	000440
OVE.F7=	004014	OVS.H2=	046344	RCLP6	004452	REPEAT=	001000	SECTCY	001614
OVE.F8=	004014	OVS.MN=	001040	RCTBAD	002000	RESTAB	004303	SECTRK	001606
OVE.F9=	004014	OV... =	026400	RCTBUF=	013477	RETAD	004412	SECT12	001612
OVE.G1=	004014	PAERR	003400	RCTCK	004371	RETRY	002165	SECT76	001610
OVE.G2=	004014	PAGAIN	004072	RCTCKD	004700	REVBUF=	012264	SEEK	003062
OVE.G3=	004014	PAGE	003322	RCTCKE	004726	REVCNT	001741	SEEKER	004453
OVE.G4=	004014	PAGER	003355	RCTCLP	004426	REVECT=	000100	SEEK0	003064
OVE.G5=	004014	PALP1	003332	RCTCL1	004431	REVLN=	000004	SEEK1	003074
OVE.G6=	004014	PALP2	003343	RCTCNT	002162	REVRBN	001634	SEEK2	003133
OVE.G7=	004014	PALP3	003376	RCTERR	004665	REVSEC=	000007	SEEK3	003150
OVE.G8=	004014	PALP4	003365	RCTFMT	001727	RFTL	006150	SEEK4	003123
OVE.H1=	004014	PARITY=	000200	RCTINI	004014	RLD	006140	SEEK5	003131
OVE.H2=	004014	PARIT1=	000400	RCTLBN	001746	RLDONE	004404	SEEK6	003122
OVE.MN=	001364	PARMTB	004174	RCTNGD	004643	RLOOP	004157	SEEK7	003115
OVLBLK	002104	PBNBUF=	010455	RCTRLP	005165	RLOOP1	004217	SEND =	000004
OVLN =	000003	PBUFE =	004330	RCTS12=	000014	RMMER	005372	SERBD	005055
OVLTLB	002013	PCNT	002163	RCTS76=	000020	ROVER	004047	SERCOM	005034
OVL.F1=	001231	PCON	004014	RCTTOT	001720	ROVER1	004065	SERDUM	001772
OVL.F2=	001660	PDONE	004312	RCTUPD	004014	RPRIM =	000004	SEROK	005065
OVL.F3=	000465	PERR	004174	RCTWLP	004434	RPT	004510	SERRT	005101
OVL.F4=	000776	PHYSA =	001000	RCTWRT	005160	RPT1	005103	SETOVL	004465
OVL.F5=	001411	PLEN =	000021	RCTWT	004427	RQUIT	004315	SETRET	004412
OVL.F6=	000407	PNGBLK	004067	RCV =	000005	RRC	005543	SETSIZ	004301
OVL.F7=	000551	PNGPG	004672	RCVMNT	003437	RRERR	004626	SETS12	004407
OVL.F8=	000725	PNGPNG	004065	RCVRDY=	000001	RRPL	004631	SFTRPT	005434
OVL.F9=	000234	PNGRD	004131	RCWERR	005417	RSER =	000000	SHORTO=	000000
OVL.G1=	001531	PRET	004172	RCXLP	004644	RTDON	004662	SIP =	000012
OVL.G2=	000053	PRIM =	001000	RC.FRE=	000000	RTRY =	000001	SKERR	004370
OVL.G3=	000307	PRIMRB	003677	RC.NUL=	100000	RTY =	100000	SKIP1	004164
OVL.G4=	002142	PRMBUF=	011607	RC.PRV=	020000	RTYCNT	002156	SKIP19	004142
OVL.G5=	000460	PRMY =	100000	RC.SND=	030000	RTYDN =	000002	SKIP3	004245
OVL.G6=	000525	PROD =	000000	RC.UNU=	040000	RUN	001476	SKIP4	004145
OVL.G7=	000426	QUESDN=	000200	RDBLK	001373	RWCMD =	013400	SKIP5	004070
OVL.G8=	000401	RBNBUF=	014154	RDBUF =	010000	RWGD	004616	SKIPJ	004310
OVL.H1=	000406	RBNLBN	001620	RDCMD =	100000	RWGOOD	005351	SKIP7	004251
OVL.H2=	001555	RBNPCY	001632	RDLN =	000004	RWRDY =	100000	SKPCNT	001716
OVL.MN=	004316	RBNRPT=	000200	RDONE	004253	RWTDON	005415	SLAS =	000057

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE 97-4

SYMBOL TABLE

SLEEK	004174	STSC =	000012	THREB =	000011	UID =	000000	WRITE9	005137
SLEEK2	004211	STSKP	002553	TILOP	005150	UNIT	001412	WRIT10	006060
SLEN =	000026	STSK1	002422	TILOP1	005156	UNITBD	005006	WRIT12	004324
SLOOP	004505	STWLK	002522	TIMER	003545	UNITCN	004774	WRIT13	005442
SND	005301	STXBN =	000002	TIMLOP	005106	UNITRT	005004	WRN	004243
SNDCNT	002155	ST.DB =	001000	TIMLO1	005102	UNNO	001413	WRMLN =	000040
SNDLP	004620	ST.DBN	002012	TIMLP	003550	UNSEC =	000175	XAGAIN	004700
SNDPMT	003425	ST.DF =	000020	TIMTBL	004342	UN.ERI	001371	XBBAD	002002
SNDRES	004557	ST.DR =	000040	TIMVAL=	100000	UN.ERR	001367	XBBUFE=	004477
SODT =	016747	ST.ERB=	000002	TKIP1	004042	UN.ERT	001370	XBLEN =	000034
SRCK	002472	ST.ERR=	000374	TKIP11	004257	UN.SEK	001372	XBNCYL=	000021
SS =	100000	ST.FO =	002000	TKIP12	004174	UPDATE=	000022	XBNFND	004075
SSBIT =	000001	ST.IN =	000004	TKIP13	004313	UPDPNT	002157	XBNIT	004154
ST	001503	ST.LBN	002007	TKIP14	004310	UREAD =	000013	XBNSEC	001622
STACK	001700	ST.PS =	000002	TKIP2	004044	USDFCT	004656	XDONE	004431
STARIT	001715	ST.RBN	002010	TKIP3	004271	UWRITE=	000014	XEORCT	004765
START	004014	ST.RU =	000001	TKIP4	004247	VAXTME	005505	XFLIP	004722
START2	004201	ST.SR =	000020	TKIP5	004162	VERHD	004734	XNGBLK	004675
START3	004356	ST.WE =	000010	TKIP7	004224	VLD =	000004	XNOINC	004731
STASEC	001574	ST.WP =	170000	TKIP8	004133	VLD1 =	000010	XPBN	004357
STATFR	003605	ST.XBN	002011	TKIP9	004145	V1 =	000000	XPERR	005004
STATRE	003565	SWAP	004237	TKIP11	004145	V2 =	000002	XPNGRD	004737
STATRT	003603	SJRD	002127	TMPTRY	002167	V3 =	000004	XPRET	005001
STATST	002401	TALIP1	002700	TOTRCT	002160	V4 =	000005	XSKIP1	004030
STATUS=	000007	TALK	002563	TRK =	000011	WLOOP	005021	XSKIP2	004154
STATVL	003562	TALKDN	002632	TRKCNT	002150	WP =	000001	XSKIP3	004150
STCKSV	001701	TALKIP	002676	TRKCYL	001624	WRBLK	001373	XSKIP4	004236
STCLR =	170377	TALKP	002653	TRKGRP=	000003	WRCMD =	122400	XSKIP5	004272
STCYL =	000001	TALKRT	002644	TWO =	000006	WRFLG	001705	XSKIP6	004266
STDBN =	000003	TATTN1	002742	TWOC	002153	WRITE =	000003	XSKIP8	004313
STDIAG	002512	TBLK	001717	TWRD	002130	WRITE1	004267	XSLEEK	004066
STFLAG=	004000	TBUFF	004261	UDAFM =	001000 G	WRITE2	004545	XSLEK2	004102
STFORM	002502	TBUFFL=	000051	UERR	003316	WRITE3	005276	XYZ	004770
STLBN =	000002	TCLEAR	002662	UHASH	004360	WRITE4	004250	XYZ1	004772
STO	001736	TEMP	001410	UHKIP	004367	WRITE5	004113	Y =	000131
STPNIC	002557	TEMP2	001405	UHKIP1	004415	WRITE8	004304	YES	004300
STRBN =	000003	TERR	002735						

. ABS. 055000 000
000000 001
ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 17056 WORDS (67 PAGES)
DYNAMIC MEMORY AVAILABLE FOR 71 PAGES
.B:UDAF52.LST/C=[30,30]DMAC52/M,B:UDAF52

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE S-12
 CROSS REFERENCE TABLE (CREF V04.00)

MSGTBL	22-12#	23-9													
MULDN	32-11	32-13#													
MULER1	33-29	33-31#													
MULERR	32-12	32-14#													
MULPC	5-35#	9-18*	9-19*	9-20											
MULT10	31-33	33-11#													
MULT2	32-6#	33-11	33-16	33-17											
N	6-29#	50-58*	50-62*	50-108*	56-10*	56-14*	56-60*								
N1	6-30#	50-59*	50-60	50-63*	50-64	50-170*	50-172*	56-11*	56-12	56-15*	56-16	56-226*	56-228*		
NDLL	3-236#	65-131	65-157												
NEXT	14-8#	23-52	40-12	49-28	55-137	65-150	78-138	80-97							
NEXT1	5-169#	65-56*	65-119*	65-125*	68-5*	68-67*	68-73*	68-77	68-82*	75-6*	75-69*	75-75*	75-79	75-84*	
	76-59*	76-107*	76-113*	89-33*	89-90*	89-98*	90-16*	90-90*	90-100*	93-9*	93-64*	93-70*	93-74	93-79*	
NEXT5	14-11#	14-20													
NGD	80-17	80-25	80-64	80-92#											
NGD1	80-28	80-99#													
NN1	6-31#	50-60*	50-64*	50-172	56-12*	56-16*	56-228								
NO	65-104	65-119#													
N0576	47-5	47-10#													
NOCERR	31-23	31-25	31-27	31-29	31-31#										
NOCRY	18-13	18-15#													
MODLL	65-22	65-24#													
NOERR	50-73	50-95	50-100	50-105#											
NOFCT	3-292#	65-37	80-27												
NOGOOD	76-92	76-107#													
NOINC	73-20	73-22#													
NOLBN	74-55	74-119#													
NOSEK	66-120	66-126#													
NOTHER	46-11	46-21	46-31	46-38	46-61	46-79#									
NOTR	66-49	66-72#													
NOTY	35-18	35-21#													
NUM	6-11#	50-92	56-44	56-110	64-10*										
NUMDBN	49-51	50-17#													
NUMLBN	60-11	61-5#													
NUMRBN	60-16	62-5#													
NUMXBN	49-55	50-37#													
OCDONE	70-29#	70-135													
OCLOOP	70-29#	70-87													
OERR	14-13	14-17#													
OERR2	14-18	14-24#	15-40												
OFATAL	78-72	78-131#													
OFFS12	3-81#	52-34	58-85	95-40											
OFFS76	3-82#	58-83	95-38												
OFFSET	5-37#	66-43	69-7	69-36*	73-6	73-34*	74-72	74-137							
OK	50-161	50-170#													
OLDONE	78-83	78-129#													
ONE	6-38#	40-170	42-29	49-112	55-20	55-117	55-121	74-16	93-11						
ONLIN	13-178#	48-8													
ONLINE	5-59	5-84#													
OQUIT	78-132	78-134	78-140#												
ORFTAL	70-82	70-138#													
OV...	5-3	5-3	5-3#	40-5	40-5	40-5	40-5#	49-8	49-8	49-8	49-8#	52-10	52-10	52-10	
	52-10#	55-5	55-5	55-5	55-5#	58-2	58-2	58-2	58-2#	60-6	60-6	60-6	60-6#	65-5	
	65-5	65-5	65-5#	66-2	66-2	66-2	66-2#	70-2	70-2	70-2	70-2#	71-7	71-7	71-7	
	71-7#	76-2	76-2	76-2	76-2#	78-2	78-2	78-2	78-2#	79-7	79-7	79-7	79-7#	80-7	
	80-7	80-7	80-7#	81-5	81-5	81-5	81-5#	88-2	88-2	88-2	88-2#	93-2	93-2	93-2	

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE S-14
 CROSS REFERENCE TABLE (CREF V04.00)

OVS.F5	5-222	71-7#												
OVS.F6	5-225	78-2#												
OVS.F7	5-228	76-2#												
OVS.F8	5-231	58-2#												
OVS.F9	5-234	93-2#												
OVS.G1	5-258	24-6	40-5#											
OVS.G2	5-237	79-7#												
OVS.G3	5-240	80-7#												
OVS.G4	5-243	81-5#												
OVS.G5	5-246	94-2#												
OVS.G6	5-261	96-5#												
OVS.G7	5-249	52-10#												
OVS.G8	5-252	60-6#												
OVS.H1	5-255	70-2#												
OVS.H2	5-264	88-2#												
OVS.PRN	5-3#													
PAERR	15-18	15-36	15-39#											
PAGAIN	73-7#	73-23												
PAGE	15-10#	17-15	49-79	50-6	54-6	55-12	55-24	55-49	55-72	55-83	55-86	55-131	59-9	66-41
	69-41	73-39	74-23	74-37	74-45	74-51	74-70	74-135	74-208	74-217	74-222	74-248	74-256	80-15
	80-53	80-62	81-60	89-8	89-18									
PAGER	15-21	15-28#												
PALP1	15-15#	15-19												
PALP2	15-17	15-20#												
PALP3	15-35	15-38#												
PALP4	15-33#	15-37												
PARIT1	3-141#	18-48												
PARITY	3-140#													
PARMTB	82-4#	82-118	82-133											
PBNBUF	3-356#	49-42	49-43	49-44	50-8	54-8	59-7	74-220	79-25	80-13	80-18	80-29	80-43	80-51
	80-60	80-65	80-73											
PBUFE	82-48	82-51#												
PCNT	6-49#	49-39*	58-125*	58-199*	59-10*	80-72*	80-82*							
PCON	94-6#													
PDONE	94-29	94-41	94-82	94-104#										
PERR	73-42	73-50#												
PHYSA	5-3	5-3	5-3	5-3	5-3#									
PLEN	82-47	82-53#												
PNGBLK	73-5#	73-40												
PNGPG	66-61	66-97	69-4#											
PNGPNG	73-4#	74-108	74-156	74-174										
PNGRD	73-30#	73-45												
PRET	73-16	73-48#												
PRIM	3-228#	56-126	56-129	56-165	56-198	58-23	58-134	58-143						
PRIMRB	20-7#	56-137	58-153	67-7	72-7									
PRMBUF	3-358#	56-142	56-149	56-154	58-159	58-165	58-168	89-6	89-9	89-44	89-70	90-38	90-65	91-31
	91-124													
PRMY	3-158#	58-141	74-93	74-105	74-171	74-196	74-252							
PROD	3-9#													
QUESDN	3-253#	23-25	29-24	30-11	37-12									
RB.CMD	3-46#	50-72	51-19*	56-24	57-19*									
RB.HI	3-45#	50-77	56-29											
RB.IM	3-47#	50-98	50-102*	50-113	50-117	56-50	56-54*	56-65	56-69					
RB.LOW	3-44#	50-75	56-27											
RBNBUF	3-362#	56-237	58-51	58-64										
RBNLBN	5-119#	40-101*	40-102*	40-177	40-178	40-193	40-194	42-27	53-13	53-34	76-27	76-28		

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE S-16
 CROSS REFERENCE TABLE (CREF V04.00)

	65-70	65-86	65-152	74-21	74-24	74-43	74-46	74-95	74-206	74-215	76-9	76-14	76-62	76-131
RDCMD	76-162	77-10												
RDLEN	3-312#	50-84	50-141	56-36	56-94	70-41	78-36	87-42	89-54	90-49	91-45	91-138		
RDONE	3-307#	50-105	50-175	51-20	56-57	56-232	57-20							
RDONE1	80-80	80-86	80-88#											
RDONE2	80-55	80-90#												
READ	80-98#	80-101												
READ1	3-53#	50-86	50-144	56-38	56-97	70-47	78-42	87-48	89-60	90-55	91-50	91-143		
READ10	50-83#													
READ11	90-48#	90-78	90-86											
READ2	70-40#	70-70	70-78											
READ3	50-140#													
READ4	56-35#													
READ7	56-93#													
READ8	78-35#	78-60	78-68											
READ9	87-41#	87-71	87-79											
READX	89-53#	89-78	89-86											
REBUFE	50-121	50-124#	50-171											
RECAL	82-99	82-102#												
RECIR	13-79#	13-163	48-11											
RECOV	3-320#	52-99	52-102	58-220	58-223									
	6-52#	43-30*	65-111*	65-121	68-59*	68-69	70-73*	70-86	70-90	70-126*	70-134	75-61*	75-71	76-99*
	76-109	78-63*	78-77	78-81	78-119*	78-127	86-62*	86-70	87-74*	87-87	87-91	87-127*	87-135	89-81*
	89-97	90-81*	90-92	90-99	93-56*	93-66								
RECTMP	6-54#	17-25	43-31*	65-109	65-115*	65-121*	68-57	68-63*	68-69*	70-71	70-77*	70-86*	70-90*	70-124
	70-130*	70-134*	75-59	75-65*	75-71*	76-97	76-103*	76-109*	78-61	78-67*	78-77*	78-81*	78-117	78-123*
	78-127*	86-60	86-66*	86-70*	87-72	87-78*	87-87*	87-91*	87-125	87-131*	87-135*	89-79	89-85*	89-97*
	90-79	90-85*	90-92*	90-99*	93-54	93-60*	93-66*							
REDO	51-32	51-37#												
RELEN	82-98	82-104#												
REPEAT	3-255#	23-11	23-27	25-59	27-12	29-17								
RESTAB	82-33#	82-117												
RETAD	23-21	23-25#												
RETRY	6-51#	13-56	13-58	13-96	13-160	14-17	15-39	43-39*	46-20	46-30	49-90	50-190	55-60	56-287
	65-105	68-53	70-67	70-120	75-55	76-93	78-57	78-113	86-56	87-68	87-121	89-75	90-75	93-50
REVBUFF	3-359#	81-15	81-53	85-5	86-1									
REVCNT	5-176#	55-68	55-73*	55-127	56-223*	66-109*								
REVECT	3-227#	55-78	56-116	56-203	58-34	58-136								
REVLEN	3-309#	85-70	86-72											
REVRBN	5-125#	20-29*	20-30*	56-138	56-139	58-155	58-156	67-10	67-11	67-20*	67-21*	67-22*	72-10	72-11
	72-20*	72-21*	72-22*											
REVSEC	3-80#	41-7												
RFTL	87-83	87-139#												
RLD	87-93	87-137#												
RLDONE	70-92	70-136#												
RLOOP	80-58	80-60#	80-71											
RLOOP1	80-74#	80-87												
RNUMBER	86-26#	86-75												
ROVER	66-24#	66-110												
ROVER1	66-28	66-30#												
RPRIM	3-248#	56-130	56-196	56-263	58-144	58-205	58-207							
RPT	58-161#	58-164												
RPT1	56-144#	56-147												
RQUIT	80-93	80-102#												
RRC	81-22	87-5#												
RRERR	50-88	50-91	50-96#											

U
D.

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE S-19
 CROSS REFERENCE TABLE (CREF V04.00)

ST.FS	3-180#	12-32												
ST.RBN	5-198#	44-16*	56-140	58-157	62-12	67-22	72-22	86-41	94-78					
ST.RU	3-179#	12-29												
ST.SR	3-183#	12-44												
ST.WE	3-187#	12-57												
ST.WP	3-182#													
ST.XBN	5-199#	44-26*	50-43	53-39	65-9	65-10	65-78	78-26	78-98	89-20	89-21	89-37	91-36	91-129
	93-35	94-37												
STACK	5-137#	23-3												
STARIT	5-156#	51-11*	51-38	51-41*	57-11*	57-38	57-41*							
START	5-5	14-10	14-16	15-14	15-23	15-32	22-5#	40-5	40-5	49-8	49-8	52-10	52-10	55-5
	55-5	58-2	58-2	60-6	60-6	65-5	65-5	66-2	66-2	70-2	70-2	71-7	71-7	76-2
	76-2	78-2	78-2	79-7	79-7	80-7	80-7	81-5	81-5	88-2	88-2	93-2	93-2	94-2
	94-2	96-5	96-5											
START2	71-11	74-7#												
START3	22-5	23-3#												
STASEC	5-100#	91-7*	91-8*	91-67	91-69	91-82	91-84	91-85	91-101*	91-102*	91-111	91-114	91-115	
STATFR	18-45	18-47	18-53#											
STATRE	18-43#	18-49												
STATRT	18-52#	18-55												
STATST	12-14#	12-43	12-47	12-51	12-55	12-60								
STATUS	3-58#	18-43												
STATVL	13-62	13-66	13-141	13-193	13-212	18-41#	46-10							
STCKSV	5-138#	13-53*	13-55	25-5*	25-55									
STCLR	3-136#	44-9	44-15	44-25	44-35									
STCYL	3-88#	40-166	68-16	70-11	75-17	76-53	85-25	86-21	87-8	90-26				
STDBN	3-87#	44-30												
STDIAG	12-49	12-52#												
STFLAG	3-257#	25-7	25-16	25-16	25-16	25-25								
STFORM	12-45	12-48#												
STLBN	3-84#	44-8												
STO	5-173#	13-140	43-83*											
STPNIC	12-27	12-30	12-33	12-40	12-71#									
STRBN	3-85#	44-14												
STSC	3-272#													
STSK1	12-20	12-23#												
STSKP	12-62	12-68#												
STWLK	12-53	12-56#												
STXBN	3-86#	44-20												
SWAP	66-77	66-86#												
SWRD	6-14#	40-226	56-246	64-32										
TALIP1	13-54	13-56#												
TALK	12-15	12-64	12-66	13-13#	13-121	13-129	13-139	13-172	13-180	17-27				
TALKDN	13-33	13-37#												
TALKIP	13-51	13-55#												
TALKP	13-43	13-45#												
TALKRT	13-38	13-42#												
TATTN1	13-69	13-73#												
TBLK	5-158#	41-17*	41-20*	51-30	57-30									
TBUFF	22-78#	22-81	23-17	23-36										
TBUFFL	22-81#	23-18	23-37											
TCLEAR	13-24	13-36	13-41	13-50#										
TEMP	5-38#	20-8*	20-10	31-15*	31-16*	31-17*	31-19*	31-21*	31-22	31-36*	31-38	31-40*	38-14*	38-15*
	38-18	38-21	38-22	38-79*	38-81*	38-84	40-28*	40-29*	40-31	40-44*	40-45*	40-46	40-56*	40-57*
	40-58	40-69*	40-72*	40-73	40-94*	40-97*	40-98	40-138*	40-139*	40-140	40-149*	40-152*	40-153	40-177*
	40-178*	40-181	40-193*	40-194*	40-197	40-202	40-239*	40-240*	40-241	41-10*	41-11*	41-12	41-15*	42-24*

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE S-22
 CROSS REFERENCE TABLE (CREF V04.00)

XPERR	69-44	69-52#			
XPNGRD	69-31#	69-47			
XPRET	69-17	69-50#			
XSKIP1	55-10	55-13#			
XSKIP2	55-50#	55-65			
XSKIP3	55-48#	55-112			
XSKIP4	55-70	55-75#			
XSKIP5	55-77	55-84	55-87#		
XSKIP6	55-81	55-85#			
XSKIP8	55-93	55-95#			
XSLEEK	55-25#	55-124	55-126		
XSLEK2	55-31#	55-115			
XYZ	74-118	74-146	74-173	74-184	74-193
XYZ1	74-59	74-120	74-199#		74-198#
Y	22-82#	35-17			
YES	89-80	89-90#			

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE M-1
CROSS REFERENCE TABLE (CREF V04.00)

BCC	2-4#	7-24	8-18	10-25	10-29	11-20	11-36	18-13	31-23	31-25	31-27	31-29	32-11	33-19
BEQ	33-22	33-25	33-28	38-34	38-48	38-52	38-57	38-93						
	2-4#	9-22	11-19	12-20	12-30	12-33	12-37	12-58	13-21	13-33	13-94	13-106	13-148	13-161
	13-196	13-198	13-215	13-217	14-18	15-17	15-35	15-40	17-19	18-45	18-47	19-17	19-20	20-14
	21-11	23-28	25-9	25-11	25-13	25-15	25-16	26-13	26-20	28-9	30-7	31-14	31-32	31-35
	34-14	34-18	35-15	38-100	42-13	46-18	46-28	46-51	46-54	46-57	47-5	47-9	49-32	49-45
	49-88	50-57	50-73	50-95	50-111	50-115	50-121	50-149	50-161	50-179	50-188	51-15	51-17	51-32
	52-15	52-17	52-39	52-41	52-97	55-10	55-58	55-77	55-81	55-128	56-25	56-47	56-63	56-67
	56-73	56-75	56-102	56-113	56-123	56-158	56-166	56-171	56-217	56-236	56-241	56-250	56-253	56-258
	56-264	56-270	56-285	57-15	57-17	57-32	58-19	58-59	58-82	58-89	58-91	58-114	58-142	58-172
	58-206	58-218	64-6	64-36	64-39	65-104	65-106	65-112	65-128	66-28	66-59	66-123	68-26	68-52
	68-54	68-60	68-76	68-78	69-15	69-17	70-52	70-61	70-63	70-66	70-68	70-74	70-82	70-92
	70-119	70-121	70-127	73-14	73-16	74-26	74-39	74-41	74-77	74-79	74-90	74-167	74-193	74-201
	74-205	74-214	74-233	75-27	75-54	75-56	75-62	75-78	75-80	76-66	76-92	76-94	76-100	76-116
	76-126	76-143	76-157	78-47	78-56	78-58	78-64	78-72	78-83	78-112	78-114	78-120	78-132	78-134
	80-17	80-21	80-23	80-31	80-55	80-58	80-64	80-86	80-93	81-14	81-40	81-42	81-56	82-149
	84-29	85-35	85-62	85-64	85-94	85-97	86-5	86-55	86-57	86-63	87-53	87-62	87-64	87-67
	87-69	87-75	87-83	87-93	87-120	87-122	87-128	89-65	89-74	89-76	89-82	89-92	90-60	90-69
	90-71	90-74	90-76	90-82	90-94	91-52	91-99	91-145	91-160	92-18	92-21	93-49	93-51	93-57
	93-73	93-75	95-16	95-27	95-34	95-37	95-44							
BMI	2-4#	13-57	13-59	13-97	21-18	25-40	26-14	26-16	26-18	26-21	26-34	28-10	30-8	31-20
	36-9	46-21	46-31	46-38	46-61	49-71	49-91	50-153	50-191	55-41	55-61	55-70	56-106	56-288
	65-110	67-5	68-58	69-21	70-72	70-125	72-5	73-20	75-60	76-98	76-125	78-62	78-118	86-61
	87-73	87-126	89-80	90-80	91-29	91-122	93-55	94-14	94-55					
BNE	2-4#	9-14	10-22	11-30	12-27	12-40	12-45	12-49	12-53	12-62	13-26	13-38	13-43	13-51
	13-63	13-67	13-69	13-108	13-142	13-144	13-146	13-151	13-194	13-201	13-213	14-13	16-12	16-21
	16-32	17-9	18-16	18-32	18-49	19-24	21-16	23-26	23-41	23-44	24-16	25-21	25-31	25-34
	25-47	26-31	26-36	27-6	29-6	29-8	29-10	29-12	29-14	35-18	37-9	38-8	38-37	38-98
	40-8	40-39	40-79	40-231	43-49	43-55	43-78	43-82	46-11	46-39	46-70	49-38	49-105	49-108
	49-115	50-88	50-91	50-100	50-107	50-109	50-146	50-155	50-171	50-177	51-39	51-43	52-30	52-32
	52-62	52-67	52-71	52-74	52-82	52-87	52-89	52-91	55-35	55-91	55-112	55-115	55-124	55-126
	56-9	56-40	56-43	56-52	56-59	56-61	56-99	56-109	56-117	56-127	56-147	56-185	56-190	56-204
	56-207	56-227	56-234	56-256	57-39	57-43	58-9	58-33	58-35	58-37	58-42	58-57	58-73	58-75
	58-116	58-120	58-124	58-127	58-133	58-137	58-164	58-180	58-196	58-198	58-200	58-203	59-6	61-15
	63-12	64-42	65-22	65-33	65-48	65-58	65-68	65-126	65-130	65-132	65-135	65-139	65-142	65-156
	66-49	66-73	66-77	66-110	66-120	68-9	68-74	69-11	69-24	69-44	70-18	70-49	70-56	73-10
	73-23	73-42	74-55	74-96	74-120	74-142	74-203	74-257	75-11	75-76	76-12	76-20	76-47	76-114
	76-118	76-129	76-135	76-137	76-138	76-146	76-169	77-18	78-44	78-51	79-20	80-25	80-28	80-36
	80-50	81-27	81-43	81-46	81-49	82-147	83-11	84-18	84-27	85-15	85-18	85-73	85-101	86-15
	86-75	87-15	87-50	87-57	89-11	89-62	89-69	89-99	89-102	89-105	89-109	90-19	90-57	90-64
	90-101	90-105	90-109	91-54	91-63	91-65	91-88	91-97	91-116	91-147	91-156	91-158	91-164	92-25
	93-71	93-81	94-47	94-65	94-87	95-14	95-25							
BPL	2-4#	31-18	38-25	38-27	38-90	41-16	46-85	51-33	52-50	57-33	58-100	74-59	80-69	80-80
	94-10													
BR	2-4#	9-17	9-24	10-26	10-31	11-37	12-43	12-60	13-24	13-36	13-41	13-54	13-75	13-98
	13-109	13-152	13-164	13-166	14-16	14-20	15-19	15-37	18-55	20-16	23-34	25-9	25-16	25-17
	25-60	26-41	27-13	29-18	30-10	30-24	31-30	31-43	31-45	31-53	32-12	33-29	34-20	35-22
	36-20	37-11	38-102	40-10	40-41	40-82	41-19	42-15	46-22	46-32	46-41	49-48	49-52	49-95
	49-119	50-61	50-123	50-169	50-195	50-197	51-21	51-36	52-19	52-52	52-101	55-37	55-65	55-84
	55-93	55-141	56-13	56-77	56-114	56-119	56-160	56-163	56-173	56-205	56-225	56-243	56-260	56-272
	56-293	56-295	57-21	57-36	58-11	58-21	58-61	58-84	58-102	58-135	58-139	58-174	58-176	58-186
	58-211	58-222	61-17	63-14	64-45	65-23	65-39	65-108	65-116	65-158	66-71	66-85	67-23	68-11
	68-28	68-56	68-64	68-83	69-42	69-47	70-20	70-64	70-70	70-78	70-87	70-123	70-131	70-135
	72-23	73-40	73-45	74-28	74-102	74-118	74-146	74-173	74-184	74-209	74-212	75-13	75-29	75-58
	75-66	75-85	76-49	76-68	76-96	76-104	76-171	78-60	78-68	78-78	78-116	78-124	78-128	80-33
	80-71	80-87	80-101	81-50	81-54	82-153	84-21	85-20	85-37	85-66	86-7	86-17	86-59	86-67

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE M-2
CROSS REFERENCE TABLE (CREF V04.00)

	87-17	87-65	87-71	87-79	87-88	87-124	87-132	87-136	89-13	89-78	89-86	89-100	90-21	90-72
	90-78	90-86	90-102	91-61	91-154	91-161	91-169	93-53	93-61	93-80	94-29	94-41	94-49	94-67
CALL	94-82	94-89	95-39	95-68										
	2-4#	9-23	10-27	12-15	12-22	12-42	12-46	12-50	12-54	12-59	12-64	12-66	12-72	13-49
	13-62	13-64	13-65	13-66	13-72	13-73	13-115	13-121	13-129	13-139	13-141	13-158	13-162	13-163
	13-172	13-180	13-193	13-212	13-221	14-11	14-30	15-15	15-18	15-33	15-36	17-15	17-16	17-27
	19-22	19-29	20-22	20-28	23-8	23-52	24-9	24-13	25-20	25-38	27-5	28-7	29-5	30-5
	31-6	31-33	31-50	32-15	33-11	33-16	33-17	33-32	36-7	37-7	38-19	38-85	40-6	40-12
	40-32	40-48	40-60	40-75	40-86	40-100	40-132	40-141	40-157	40-164	40-171	40-183	40-187	40-199
	40-204	40-208	40-214	40-242	40-246	41-14	42-28	42-30	43-8	43-20	43-62	46-10	46-88	47-12
	48-6	48-7	48-8	48-9	48-10	48-11	48-12	49-13	49-21	49-28	49-41	49-47	49-51	49-55
	49-69	49-79	49-92	49-93	49-94	49-97	49-101	49-113	49-122	50-6	50-24	50-29	50-41	50-46
	50-65	50-151	50-158	50-192	50-193	50-194	52-18	52-20	52-48	52-56	52-84	53-12	53-14	53-16
	53-35	53-37	54-6	55-12	55-21	55-24	55-39	55-49	55-62	55-63	55-64	55-72	55-75	55-83
	55-86	55-99	55-108	55-118	55-122	55-131	55-137	55-144	56-17	56-104	56-137	56-156	56-289	56-290
	56-292	58-98	58-106	58-153	58-170	58-201	58-229	59-9	60-11	60-16	60-21	60-22	61-10	61-23
	62-10	62-19	63-20	63-26	63-31	64-18	64-22	65-45	65-46	65-54	65-72	65-80	65-113	65-124
	65-150	65-161	66-22	66-32	66-38	66-41	66-60	66-61	66-95	66-97	66-107	66-125	67-7	67-14
	68-21	68-24	68-61	68-72	68-81	68-88	69-38	69-41	69-54	70-25	70-34	70-59	70-75	70-84
	70-94	70-95	70-128	70-140	72-7	72-14	73-36	73-39	73-52	74-17	74-23	74-35	74-37	74-45
	74-51	74-58	74-61	74-67	74-70	74-99	74-107	74-108	74-126	74-132	74-135	74-155	74-156	74-174
	74-198	74-208	74-217	74-222	74-235	74-245	74-248	74-256	75-22	75-25	75-63	75-74	75-83	75-89
	76-33	76-38	76-61	76-64	76-101	76-112	76-123	76-124	76-127	76-159	76-174	78-19	78-49	78-54
	78-65	78-75	78-86	78-87	78-121	78-138	78-142	79-17	79-18	79-26	79-30	80-15	80-53	80-62
	80-68	80-79	80-97	80-104	81-22	81-51	81-57	81-60	81-61	81-62	82-136	82-140	84-14	85-30
	85-33	85-74	85-99	85-106	86-30	86-64	87-25	87-26	87-55	87-60	87-76	87-85	87-95	87-96
	87-129	87-141	88-11	88-12	88-13	89-8	89-18	89-38	89-67	89-72	89-83	89-95	89-113	90-9
	90-32	90-62	90-67	90-83	90-97	90-113	91-27	91-83	91-112	91-120	92-23	92-30	93-12	93-21
	93-24	93-58	93-69	93-78	93-85	94-9	94-11	94-13	94-17	94-18	94-21	94-23	94-30	94-33
	94-35	94-44	94-62	94-73	94-76	94-95	94-97	95-8	95-23	95-32	95-48	95-58		
DMCODE	2-3#	5-3												
DMEND	2-3#	97-48												
DMODT	2-5#													
DMOV	4-65#													
DMOVLY	2-3#	5-3	40-5	49-8	52-10	55-5	58-2	60-6	65-5	66-2	70-2	71-7	76-2	78-2
	79-7	80-7	81-5	88-2	93-2	94-2	96-5							
DUBINC	4-106#	52-30	52-89	58-33	58-73	58-198	65-135	74-257	76-138	81-43	89-105	90-105	91-63	91-116
	91-156	93-81												
GETB	4-117#	25-9												
JMP	2-4#	5-5	7-32	8-24	9-26	10-37	11-24	11-28	11-34	12-47	12-51	12-55	12-70	13-47
	13-81	13-112	13-123	13-131	13-157	13-174	13-182	13-204	13-218	15-25	15-38	15-42	16-13	16-22
	16-33	17-28	18-18	18-34	18-52	19-25	20-32	21-20	22-5	23-24	24-17	25-52	25-54	26-39
	27-9	28-13	29-25	30-13	31-48	32-13	33-30	34-22	35-20	36-24	37-14	38-104	45-8	46-78
	47-10	48-13	49-116	50-10	50-30	50-47	50-199	51-44	52-106	53-21	53-42	54-10	55-138	56-296
	57-44	58-230	59-12	60-23	61-24	62-20	63-34	64-47	66-126	67-19	68-85	69-51	70-137	71-11
	72-19	73-48	74-236	74-258	75-86	76-160	77-20	78-130	78-139	79-27	80-91	80-98	82-157	83-13
	84-34	85-76	85-102	86-79	87-138	88-14	89-110	90-110	91-165	92-26	93-82	94-106	95-63	96-16
MSG	4-56#	5-49	5-50	5-51	5-52	5-53	5-54	5-55	5-56	5-57	5-58	5-59		
OVTERM	5-3	5-3#	5-3#	40-5	40-5#	49-8	49-8#	52-10	52-10#	55-5	55-5#	58-2	58-2#	60-6
	60-6#	65-5	65-5#	66-2	66-2#	70-2	70-2#	71-7	71-7#	76-2	76-2#	78-2	78-2#	79-7
	79-7#	80-7	80-7#	81-5	81-5#	88-2	88-2#	93-2	93-2#	94-2	94-2#	96-5	96-5#	97-48
POP	4-32#	7-29	7-30	8-23	9-25	10-36	11-21	11-22	11-25	11-26	11-31	11-32	12-68	12-69
	13-45	13-46	13-110	13-111	13-122	13-130	13-155	13-156	13-173	13-181	13-202	13-203	14-15	15-28
	18-17	18-33	18-50	18-53	20-31	23-20	23-33	25-23	25-35	25-36	25-37	25-53	25-57	27-8
	27-10	28-12	29-15	29-23	30-12	30-19	31-41	31-46	31-51	34-21	36-15	36-23	37-13	38-20
	38-88	50-9	50-122	50-174	50-198	56-76	56-230	58-154	58-208	58-213	59-11	66-111	66-114	66-117

UDAF52 - UDA-52 FORMATTER DMACR X04.01 23-AUG-82 13:14:22 PAGE M-3
CROSS REFERENCE TABLE (CREF V04.00)

	69-50	74-218	74-223	74-226	74-229	76-147	76-150	76-153	82-144	83-12	84-33	95-49	95-59	95-61
POPA	95-64													
PUSH	4-85#	54-9	70-136	85-75	87-137									
	4-9#	7-19	7-20	8-14	9-12	10-17	11-14	11-15	12-12	12-13	13-13	13-14	13-79	15-85
	13-88	13-119	13-127	13-135	13-136	13-170	13-178	13-186	13-187	15-11	18-7	18-25	18-41	20-7
	23-15	23-22	23-23	23-31	25-4	25-18	25-24	25-42	25-45	25-48	25-49	25-50	27-4	28-6
	29-4	30-4	31-5	31-37	34-10	36-6	37-6	38-16	38-82	50-4	50-53	50-116	56-68	58-46
	58-150	59-4	66-13	66-15	66-17	69-4	74-10	74-12	74-14	74-19	76-22	76-24	76-26	82-142
PUSHA	83-6	84-7	95-10	95-12	95-18									
RETURN	4-74#	54-4	70-6	85-4	87-5									
	2-4#	7-32	8-24	9-26	10-37	11-24	11-28	11-34	12-70	13-47	13-112	13-123	13-131	13-157
	13-174	13-182	13-204	13-218	15-25	15-38	15-42	16-13	16-22	16-33	17-28	18-18	18-34	18-52
	19-25	20-32	21-20	23-24	24-17	25-54	26-39	27-9	28-13	29-25	30-13	31-48	32-13	33-30
	34-22	35-20	36-24	37-14	38-104	45-8	46-78	47-10	48-13	49-116	50-10	50-30	50-47	50-199
	51-44	52-106	53-21	53-42	54-10	55-138	56-296	57-44	58-230	59-12	60-23	61-24	62-20	63-34
	64-47	66-126	67-19	68-85	69-51	70-137	72-19	73-48	74-236	74-258	75-86	76-160	77-20	78-130
	78-139	79-27	80-91	80-98	82-157	83-13	84-34	85-76	85-102	86-79	87-138	88-14	89-110	90-110
STOB	91-165	92-26	93-82	94-106	95-63	96-16								
XOR	4-133#	25-16												
	4-97#	18-10												

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32
TABLE OF CONTENTS

3-	1	EQUATES
5-	1	DATA STRUCTURES
7-	1	MATH SUBROUTINES
12-	1	SDI SUBROUTINES
14-	1	OVERLAY PROCESSING ROUTINES
17-	1	MISCELLANEOUS COMMON ROUTINES
22-	1	DUP DM<->HOST STARTUP OVERLAY
38-	1	INITIALIZATION OVERLAY (G1)
46-	1	DBN/XBN FORMAT OVERLAY (F1)
49-	1	DBN/XBN TRACK FORMAT OVERLAY (G7)
52-	1	LBN FORMATTING OVERLAY (F2)
56-	1	LBN FORMAT IMAGE SETUP OVERLAY (F8)
58-	1	L/RBN COMPUTE OVERLAY (G8)
62-	1	FCT DOWN-LINE LOAD OVERLAY (F3)
63-	1	RCT UPDATE OVERLAY (F4)
67-	1	RCT READ OVERLAY (H1)
68-	2	FCT->RCT CONVERSION OVERLAY (F5)
73-	1	RCT INITIALIZE OVERLAY (F7)
75-	1	FCT READ OVERLAY (F6)
76-	1	GET FCT BLOCK FOR D/XBN FORMAT (G2)
77-	1	GET FCT BLOCK FOR LBN FORMAT (G3)
78-	1	RCT CLEANUP OVERLAY (G4)
85-	1	FINAL CHECK OVERLAY (H2)
90-	1	FCT WRITE OVERLAY (F9)
91-	1	PBN->D,X,L,RBN CONVERSION OVERLAY (G5)
93-	1	ERROR MESSAGE OVERLAY (G6)

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 2
DIAGNOSTIC MACHINE MACROS - OVERLAY VERSION WITH 'RELOCATION'

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57

.TITLE UDAFM - UDA FORMATTER
.IDENT /02.01/
.NLIST BEX

:
: COPYRIGHT (C) 1980,1981,1982
: DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASS.

: THIS SOFTWARE IS FURNISHED UNDER A LICENSE FOR USE ONLY ON A
: SINGLE COMPUTER SYSTEM AND MAY BE COPIED ONLY WITH THE INCLUSION
: OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE, OR ANY OTHER
: COPIES THEREOF, MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE
: TO ANY OTHER PERSON EXCEPT FOR USE ON SUCH SYSTEM AND TO ONE
: WHO AGREES TO THESE LICENSE TERMS. TITLE TO AND OWNERSHIP OF
: THE SOFTWARE SHALL AT ALL TIMES REMAIN IN DIGITAL.

: THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT
: NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL
: EQUIPMENT CORPORATION.

: DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF
: ITS SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.

: VERSION 01.00

: M. A. PARENTI 16-MAY-80

: MODIFIED BY:

VERSION 02.00

M. A. PARENTI 09-DEC-80
NEW DM INSTRUCTIONS (MEM -> MEM)
CODE OPTIMIZATION

31-MAR-81
M. A. PARENTI
ADDED DOUBLE WORD ADDRESSING FOR OVERLAY ADDRESSES

21-APR-81
M. A. PARENTI
FIX GROUP OFFSET CALCULATION

23-APR-81
M. A. PARENTI
FIX ZERO GROUP PROBLEM

24-APR-81
M. A. PARENTI
FIX LBN GROUP PROBLEM
FIX RECAL WAIT PROBLEM

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 2-1
DIAGNOSTIC MACHINE MACROS - OVERLAY VERSION WITH 'RELOCATION'

58	
59	
60	
61	05-MAY-81
62	M. A. PARENTI
63	FIX SIZE PROBLEM FOR RAB1
64	
65	13-MAY-81
66	M. A. PARENTI
67	ADD LIMITED DUP FUNCTIONALITY
68	
69	15-MAY-81
70	M. A. PARENTI
71	FIX SUBUNIT MASK PROBLEM
72	
73	15-MAY-81
74	M. A. PARENTI
75	ONLY WRITE NON-PAD BLOCKS OF FCT
76	
77	15-MAY-81
78	M. A. PARENTI
79	FIX COMPUTATION OF NON-PAD FCT BLOCKS
80	
81	15-MAY-81
82	M. A. PARENTI
83	FIX BLOCK ZERO FCT PROBLEM
84	
85	28-MAY-81
86	M. A. PARENTI
87	FIX SUBUNIT PROBLEMS
88	WRITE ONLY NON-PAD BLOCKS OF RCT
89	
90	01-JUN-81
91	M. A. PARENTI
92	FIX DOUBLE COMPARE PROBLEM
93	
94	08-JUN-81
95	M. A. PARENTI
96	FIX EXISTING FCT FORMAT PROBLEM
97	
98	
99	
100	17-JUN-81
101	M. A. PARENTI
102	FIX SUBUNIT WRITE PROTECT PROBLEM
103	
104	17-JUN-81
105	M. A. PARENTI
106	ADD STATUS UPDATES AT VARIOUS PLACES
107	
108	22-JUN-81
109	M. A. PARENTI
110	FIX RBN STARTING BITS PROBLEM
111	
112	22-JUN-81
113	M. A. PARENTI
114	FIX NON-PAD RCT INITIALIZE PROBLEM

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 2-2
 DIAGNOSTIC MACHINE MACROS - OVERLAY VERSION WITH 'RELOCATION'

115	:	06-JUL-81
116	:	M. A. PARENTI
117	:	FIX STATISTICS COUNT OF BAD BLOCKS TO NOT INCLUDE RBN
118	:	
119	:	18-AUG-81
120	:	M. A. PARENTI
121	:	FIX CLEARING OF ECC THRESHOLD
122	:	
123	:	16-SEP-81
124	:	M. A. PARENTI
125	:	KLUDGE RECIR AND LAST BITS FOR UDA BUG
126	:	
127	:	6-OCT-81
128	:	M. A. PARENTI
129	:	FIX JUSTIFICATION ON RESPONSES
130	:	
131	:	19-OCT-81
132	:	M. A. PARENTI
133	:	FIX ECC CHECKING AND MINOR FIXES TO QUESTIONS
134	:	
135	:	13-NOV-81
136	:	M. A. PARENTI
137	:	FIX CONVERSION BLOCK, CLEANUP COUNT
138	:	CHECK FOR NO CORRECTION IF 0 THRESHOLD
139	:	
140	:	20-NOV-81
141	:	M. A. PARENTI
142	:	FIX PROBLEM WITH BAD RBN WHEN PRIMARY IN FCT
143	:	ADD HEAD VERIFICATION ROUTINE AND MAKE FINAL CHECK
144	:	ROUTINES A SEPARATE OVERLAY
145	:	
146	:	25-JAN-82
147	:	M. A. PARENTI
148	:	CHECK FOR VALID NUMERICS IN USER RESPONSES
149	:	CHECK FOR UNIBUS ERRORS
150	:	MODIFY LONG TIMEOUT
151	:	CHECK FOR VALID STATUS IN GETUNT
152	:	
153	:	09-FEB-82
154	:	M. A. PARENTI
155	:	FIX VAX DATE ROUTINE
156	:	
157	:	18-FEB-82
158	:	M. A. PARENTI
159	:	CHANGE RETRY RECOVERY ON NON-CHECK PASS READS/Writes TO NOT
160	:	ISSUE A RECAL/RESEEK SEQUENCE AFTER RECOVERY LEVEL 0
161	:	
162	:	15-MAR-82
163	:	M. A. PARENTI
164	:	FIX PBN <-> LBN COMPARE PROBLEM IN FCT->RCT CONVERSION OVERLAY
165	:	
166	:	24-MAR-82
167	:	M. A. PARENTI
168	:	FIX PBN -> D/X/L/RBN CONVERSION TO TAKE INTO ACCOUNT OFFSET
169	:	FIX BAD RBN WITH PRIMARY IN FCT PROBLEM
170	:	
171	:	

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 2-3
DIAGNOSTIC MACHINE MACROS - OVERLAY VERSION WITH 'RELOCATION'

```
172      :           26-MAR-82
173      :           M. A. PARENTI
174      :           FIX PBN INCREMENT IN D/XBN FORMAT SETUP
175      :
176      :           10-JUN-82
177      :           M. A. PARENTI
178      :           FIX RBN START BITS FOR PRIMARY
179      :           FIX RCTTOT DECREMENT
180      :
181      : UDA50 DISK FORMATTER
182      :
183      :
```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 3
EQUATES

```

1          .SBTTL EQUATES
2          :
3          EQUATES
4          :
5          :
6          :
7          :
8          :
9          000000      PROD      =      0          ;PRODUCTION (1)
10         :          :          :NOT PRODUCTION (0)
11         :
12         :          :
13         :          :
14         000000      FT.BUF   =      0.         ;BUFFER POINTER OFFSET
15         000001      FT.LOW   =      1.         ;LOW ORDER HEADER OFFSET
16         000002      FT.HI    =      2.         ;HI ORDER HEADER OFFSET
17         :
18         :
19         :
20         :
21         :
22         :          :
23         :          :
24         000000      RW.STAT  =      0.         ; STATUS (12-15), NEXT BUFR PTR (0-14)
25         000000      RW.ER1   =      0.         ; ALSO USED AS ECC ERROR INDICATOR
26         000005      RW.DUM   =      5.         ; POINTER TO DUMMY SDI CONTROL BLOCK
27         000001      RW.BUF   =      1.         ; POINTER TO DATA BUFFER
28         000002      RW.LOW   =      2.         ; 1ST HEADER WORD (LO ORDER LBN)
29         000003      RW.HI    =      3.         ; 2ND HEADER WORD (HI ORDER LBN)
30         000004      RW.CMD   =      4.         ; SDI RT CMD (8-15), HEAD ADDR (0-7)
31         000000      RW.DAT   =      0.         ; 1ST WORD OF 256 WORD DATA BUFFER
32         000400      RW.EDC   =      256.       ; EDC
33         000401      RW.ER2   =      257.       ; 1ST ECC RESIDUE
34         :
35         :
36         :
37         :
38         :          :
39         :          :
40         :          :
41         000000      RB.LOW   =      0.         ;LOW ORDER BLOCK NUMBER
42         000001      RB.HI    =      1.         ;HIGH ORDER BLOCK NUMBER
43         000002      RB.CMD   =      2.         ;READ COMMAND AND TRACK NUMBER
44         000003      RB.IM    =      3.         ;IMAGE COUNTER
45         :
46         :          :
47         :          :
48         000000      BREAK   =      0.         ;BREAKPOINT XFC CODE
49         000001      FORMAT  =      1.         ;FORMAT TRACK XFC CODE
50         000002      READ    =      2.         ;READ N SECTORS XFC CODE
51         000003      WRITE   =      3.         ;WRITE N SECTORS XFC CODE
52         000004      SEND    =      4.         ;SEND SDI COMMAND XFC CODE
53         000005      RCV     =      5.         ;RECEIVE SDI MESSAGE XFC CODE
54         000006      CMPDAT  =      6.         ;COMPARE DATA PATTERN XFC CODE
55         000007      STATUS  =      7.         ;RETURN DRIVE STATUS XFC CODE
56         000010      ECHO    =      8.         ;ECHO DATA TO DRIVE XFC CODE
57         000011      DINIT   =      9.         ;DRIVE INITIALIZE XFC CODE

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 3-1
EQUATES

58	000012	SIP	=	10.	:WAIT FOR SECTOR/INDEX PULSE XFC CODE
59	000013	UREAD	=	11.	:READ UNIBUS MEMORY XFC CODE
60	000014	UWRITE	=	12.	:WRITE UNIBUS MEMORY XFC CODE
61	000015	ECC	=	13.	:DO ECC ON BUFFER XFC CODE
62	000016	MAINTR	=	14.	:SEND MAINT READ DATA XFC CODE
63	000017	MAINTW	=	15.	:RECEIVE MAINT WRITE DATA XFC CODE
64	000020	CVT	=	16.	:CONVERT TO PHYSICAL ADDRESS XFC CODE
65	000021	DONE	=	17.	:TERMINATE DM PROGRAM XFC CODE
66	000022	UPDATE	=	18.	:UPDATE DUP PROGRESS INDICATOR XFC
67		:			
68		:			
69		:			
70	000000	SHORTO	=	0.	:SHORT TIME OUT
71	000001	FRCPY	=	1.	:NUMBER OF F/RCT COPIES
72	000001	RTRY	=	1.	:NUMBER OF RETRIES
73	000001	LONGTO	=	1.	:LONG TIMEOUT
74	000002	ERRSYM	=	2.	:NUMBER OF ALLOWABLE ECC ERRORS
75	000002	ERCV	=	2.	:ERROR RECOVERY LEVELS SUPPORTED
76	000007	REVSEC	=	7.	:REVS/SECOND
77	000011	OFFS	=	9.	:GROUP OFFSET
78	000000	CYLBN	=	0.	:CYLINDERS IN LBN AREA
79	000002	STLBN	=	2.	:HIGH ORDER STARTING LBN
80	000003	STRBN	=	3.	:HIGH ORDER STARTING RBN
81	000002	STXBN	=	2.	:HIGH ORDER STARTING XBN
82	000003	STDBN	=	3.	:HIGH ORDER STARTING DBN
83	000001	STCYL	=	1.	:HIGH ORDER STARTING CYLINDER
84	000011	LBNTRK	=	9.	:NUMBER OF LBNS PER TRACK (512)
85	000004	RBNTRK	=	4.	:NUMBER OF RBNS PER TRACK
86	000021	XBNCYL	=	17.	:NUMBER OF CYLINDERS IN XBN AREA
87	000022	DBNCYL	=	18.	:NUMBER OF CYLINDERS IN DBN AREA
88	000012	LBNHOST	=	10.	:NUMBER OF LBN'S IN HOST AREA
89	000002	GRPCYL	=	2.	:GROUPS/CYLINDER
90	000003	TRKGRP	=	3.	:TRACKS/GROUP
91	000010	FCTSZ	=	8.	:FCT SIZE IN SECTORS
92	000014	RCTSZ	=	12.	:RCT SIZE IN LBN'S
93	000005	DATA	=	5.	:DATA PREAMBLE SIZE
94	000005	HEAD	=	5.	:HEADER PREAMBLE SIZE
95		:			
96		:			
97		:			
98		:			
99		:			
100	000001	BIT0	=	000001	
101	000002	BIT1	=	000002	
102	000004	BIT2	=	000004	
103	000010	BIT3	=	000010	
104	000020	BIT4	=	000020	
105	000040	BIT5	=	000040	
106	000100	BIT6	=	000100	
107	000200	BIT7	=	000200	
108	000400	BIT8	=	000400	
109	001000	BIT9	=	001000	
110	002000	BIT10	=	002000	
111	004000	BIT11	=	004000	
112	010000	BIT12	=	010000	
113	020000	BIT13	=	020000	
114	040000	BIT14	=	040000	

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 3-2
EQUATES

```

115      100000      BIT15 =      100000
116      :
117      :
118      :
119      :
120      :
121      :
122      177400      HI8YTE =      177400      ;HIGH BYTE MASK
123      000377      LO8YTE =      000377      ;LOW BYTE MASK
124      177700      HI28YTE =      177700      ;HIGH BYTE PLUS 2 BITS
125      177600      HI18YTE =      177600      ;HIGH BYTE PLUS 1 BIT
126      007777      LO      =      007777      ;ALL BUT HEADER CODE
127      177760      FCLR   =      177760      ;CLEAR FOR FRCPY
128      170377      STCLR  =      170377      ;CLEAR FOR STARTING BITS
129      007777      BUFMSK =      007777      ;BUFFER CLEAR MASK
130      000004      VLD    =      BIT2       ;STATUS VALID BIT(1=VALID)
131      000010      VLD1   =      BIT3       ;STATUS VALID BIT(1=VALID)
132      000200      PARITY  =      BIT7       ;STATUS PARITY BIT(1=PARITY ERROR)
133      000400      PARIT1 =      BIT8       ;REAL TIME ERROR(1=ERROR)
134      :
135      :
136      :
137      000175      UNSEC  =      000175      ;UNSUCCESSFUL COMPLETION
138      :
139      :
140      :
141      000000      HD.LBN  =      000000      ;GOOD LBN
142      060000      HD.RBN  =      060000      ;GOOD RBN, PERHAPS UNUSED
143      030000      HD.REV  =      030000      ;REVECTORED LBN
144      110000      HD.BAD  =      110000      ;BAD BLOCK
145      050000      HD.PRIV =      050000      ;PRIMARY REVECTORED BLOCK
146      170000      HD.CLR  =      170000      ;CLEAR HDR CODE
147      140000      HD.DBN  =      140000      ;GOOD DBN
148      120000      HD.XBN  =      120000      ;GOOD XBN
149      100000      PRMY   =      BIT15      ;PRIMARY BIT IN FCT
150      010000      FBCHD  =      BIT12      ;BAD HEADER CODE IN FCT
151      :
152      :
153      :
154      000000      RC.FRE  =      000000      ;FREE REPLACEMENT BLOCK
155      020000      RC.PRIV =      020000      ;PRIMARY REVECTOR
156      030000      RC.SND  =      030000      ;SECONDARY REVECTOR
157      040000      RC.UNU  =      040000      ;BAD REPLACEMENT BLOCK
158      100000      RC.NUL  =      100000      ;NULL(FILL) BLOCK
159      :
160      :
161      :
162      100000      RWRDY  =      BIT15      ;READ/WRITE READY BIT POSITION
163      000002      ATTN   =      BIT1       ;ATTENTION
164      000001      RCVRDY =      BIT0       ;RECEIVER READY
165      :
166      :
167      :
168      :
169      :
170      000001      ST.RU   =      BIT0       ;RUN/STOP SWITCH 1=IN
171      000002      ST.PS   =      BIT1       ;PORT SWITCH 1=IN

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 3-3
EQUATES

172	030040	ST.DR	=	BIT5	:DIAGNOSTIC REQUESTED 1=YES
173	170000	ST.WP	=	BIT12+BIT13+BIT14+BIT15	:WRITE PROTECT SWITCH SU:0,1 1=IN
174	000020	ST.SR	=	BIT4	:SPINDLE READY 1=YES
175	001000	ST.DB	=	BIT9	:DIAG CYL ACCESS ENABLED 1=YES
176	002000	ST.FO	=	BIT10	:FORMAT CYL ACCESS ENABLED 1=YES
177	000004	ST.IN	=	BIT2	:DRIVE INITIALIZED 1=YES
178	000010	ST.WE	=	BIT3	:WRITE ERROR (WRITE LOCKED)
179	000020	ST.DF	=	BIT4	:DIAG FAILED - CANNOT DRIVE CLEAR
180	000374	ST.ERR	=	000374	:COMBINED CLEARABLE ERRORS BITS SET
181	000002	ST.ERB	=	2.	:ERROR BYTE OFFSET (3RD WORD)
182		...			
183		...			
184		...			
185		...			
186		...			
187	000023	OVCNT	=	19.	:NUMBER OF OVERLAYS
188	000003	OVLEN	=	3	:LENGTH OF 1 OVERLAY BLOCK
189	000000	LEN	=	0	:WORD COUNT OF OVERLAY
190	000001	HSTLO	=	1	:LOW ORDER UNIBUS ADDRESS
191	000002	HSTHI	=	2	:HI ORDER UNIBUS ADDRESS
192	000000	F1	=	0	:OFFSET INTO TABLE
193	000003	F2	=	3	:SECOND OVERLAY OFFSET INTO TABLE
194	000006	F3	=	6.	:THIRD OVERLAY OFFSET INTO TABLE
195	000011	F4	=	9.	:FOURTH OVERLAY OFFSET INTO TABLE
196	000014	F5	=	12.	:FIFTH OVERLAY OFFSET INTO TABLE
197	000017	F6	=	15.	:SIXTH OVERLAY OFFSET INTO TABLE
198	000022	F7	=	18.	:SEVENTH OVERLAY OFFSET INTO TABLE
199	000025	F8	=	21.	:EIGHTH OVERLAY
200	000030	F9	=	24.	:NINTH OVERLAY
201	000033	G2	=	27.	:ELEVENTH OVERLAY
202	000036	G3	=	30.	:TWELVTH OVERLAY
203	000041	G4	=	33.	:THIRTEENTH OVERLAY
204	000044	G5	=	36.	:FOURTEENTH OVERLAY
205	000047	G7	=	39.	:SIXTEENTH OVERLAY
206	000052	G8	=	42.	:SEVENTEENTH OVERLAY
207	000055	H1	=	45.	:NINETEENTH OVERLAY
208	000060	G1	=	48.	:TENTH OVERLAY
209	000063	G6	=	51.	
210	000066	H2	=	54.	
211		...			
212		...			
213		...			
214		...			
215		...			
216	000001	FCTAVL	=	BIT0	:FCT AVAILABLE
217	000010	DBN	=	BIT3	:FORMAT DBN AREA
218	000100	REVECT	=	BIT6	:REVECTOR FLAG
219	001000	PRIM	=	BIT9	:PRIMARY FOUND FLAG
220	000002	FCTEMT	=	BIT1	:FCT EMPTY FLAG
221	000020	GOBAD	=	BIT4	:DO BEST GUESS IF FCT BAD
222	000040	RCINIT	=	BIT5	:RCT LAST BLOCK FIXED UP
223	000004	FCTBAD	=	BIT2	:FCT FOUND BAD (FOR STATS)
224	000200	MANU	=	BIT7	:MANUFACTURING FORMAT
225	000400	DLL	=	BIT8	:DOWN-LINE LOAD FLAG
226	002000	BSTGS	=	BIT10	:BEST GUESS FORMAT
227	004000	NDLL	=	BIT11	:ONLY WRITE FCT SCRATCH
228	020000	INIRCT	=	BIT13	:INIT RCT FLAG

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 3-4
EQUATES

```

229      040000      FINI      =      BIT14      ;FORMAT FINISHED FLAG
230      010000      CHRDN     =      BIT12      ;CHARACTERISTICS DONE FLAG
231      100000      RTY       =      BIT15      ;RETRY FLAG
232      :
233      :
234      :          FLAG1     EQUATES
235      :
236      :
237      000001      WP        =      BIT0       ;WRITE PROTECT FLAG
238      000002      RTYDN     =      BIT1       ;RETRY DONE ON THIS SECTOR
239      000004      RPRIM     =      BIT2       ;FLAG FOR PRIMARY GOOD EDC
240      000010      ERDN      =      BIT3       ;FLAG FOR ERROR EXIT TRY
241      000020      DEAD      =      BIT4       ;HOST GONE FLAG
242      000040      BDHD      =      BIT5       ;BAD HEADER ON CHECK PASS READ
243      000100      RCINDN    =      BIT6       ;RCT INIT DONE (WITH ONE FULL PAD BLK)
244      000200      QUESDN    =      BIT7       ;STARTUP QUESTIONS FINISHED
245      000400      FLIPON    =      BIT8       ;FLIP FLAG FOR CONVERSIONS
246      001000      REPEAT    =      BIT9       ;REPEAT QUESTION FLAG FOR STARTUP
247      002000      GTFLAG    =      BIT10      ;FLAG FOR GE1B MACRO
248      004000      STFLAG    =      BIT11      ;FLAG FOR ST0B MACRO
249      010000      BDTST     =      BIT12      ;FLAG FOR TEST OF BAD HEADER IN VERHD
250      020000      FPRIM     =      BIT13      ;FLAG FOR PRIMARY IN FCT
251      :
252      :
253      :          PHYSICAL CONVERSION XFC BLOCK EQUATES
254      :
255      000000      V1         =      0         ;CYLINDER PARAMETER
256      000002      V2         =      2         ;BLOCK NUMBER PARAMETER
257      000004      V3         =      4         ;BLOCKS PER TRACK PARAMETER
258      000005      V4         =      5         ;ONLY FOR RBN'S
259      000006      CYL        =      6         ;CYLINDER RETURNED
260      000010      GRP        =      8         ;GROUP RETURNED
261      000011      TRK        =      9         ;TRACK RETURNED
262      000012      STSC       =      10        ;STARTING SECTOR RETURNED
263      000013      INDSEC     =      11        ;SECTOR FROM INDEX
264      :
265      :
266      :          DMBUF OFFSETS
267      :
268      :
269      000016      DMBUFL     =      14        ;BUFFER LENGTH
270      :
271      :
272      :
273      :          FCT BLOCK OFFSETS
274      :
275      :
276      000002      FSER       =      2         ;SERIAL NUMBER
277      000001      INST       =      1         ;FORMAT IN INSTANCE NUMBER
278      000016      C512       =      14        ;COUNT OF USED 512 ENTRIES IN FCT
279      000012      FDAT       =      10        ;MOST RECENT FORMAT DATE
280      000025      FCTFLG    =      21        ;FCT FLAG FOR GOOD/BAD FCT
281      100000      NOFCT      =      BIT15      ;FLAG - 0 - FCT GOOD
282      :                  ;          1 - FCT KNOW BAD
283      :
284      :
285      :          RCT BLOCK OFFSETS

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 3-5
EQUATES

```

286
287
288      000000      RSER      =      0      ;SERIAL NUMBER OFFSET
289
290      :
291      :      MISC      DEFINITIONS
292      :
293      :
294      000006      TWOB      =      6.      ;LENGTH OF 2 IMAGE ENTRIES
295      000011      THREB      =      9.      ;LENGTH OF 3 IMAGE ENTRIES
296      000004      RDLEN      =      4.      ;LENGTH OF CHECK PASS READ BLOCK
297      000002      ERLN      =      2.      ;LENGTH OF REVECTOR TABLE ENTRY
298      000004      REVLEN     =      4.      ;LENGTH OF SECONDARY TABLE
299      013400      RWCMD      =      013400   ;SDI READ COMMAND
300      122400      WRCMD      =      122400   ;SDI WRITE COMMAND
301      100000      RDCMD      =      100000   ;SIGNAL TO XFC NO MORE BLOCKS
302      010000      ECCF       =      BIT12    ;ECC ERROR BIT
303      000200      RBNRPT     =      128.     ;NUMBER OF RBN COPIES IN REVECTOR
304      000400      SECSI6     =      256.     ;SECTOR SIZE IN WORDS FOR 512 BYTE
305      000440      SECSI8     =      288.     ;SECTOR SIZE IN WORDS FOR 576 BYTE
306      000003      IMLEN      =      3.      ;LENGTH OF IMAGE BLOCK
307      020000      BD         =      BIT13    ;BAD FLAG IN IMAGE BUFFER
308      100000      LAST       =      BIT15    ;LAST FLAG IN IMAGE BUFFER
309      040000      RECIR      =      BIT14    ;RECIRCULATE IN FORMAT IMAGE BUFFER
310      126736      M512       =      126736   ;FCT MODE INDICATOR FOR 512
311      074161      M576       =      074161   ;FCT MODE INDICATOR FOR 576
312      100000      TIMVAL     =      32768.   ;TIMER LOOP VALUE
313      000010      MAXTRY     =      8.      ;FINAL SECONDARY WRITE RETRY LIMIT
314      007774      DUPOVL     =      7774     ;OVERLAY STARTING ADDRESS FROM DUP
315      000040      LOBL       =      00040    ;BLANK IN LOW ORDER BYTE
316      020040      BLANWD     =      20040    ;WORD WITH 2 ASCII BLANKS
317
318      :
319      :      STATUS OFFSETS
320      :
321      :
322      000000      MASK       =      0      ;SUBUNIT OFFSET MASK
323      000000      UID        =      0      ;UNIT NUMBER OFFSET
324
325      :
326      :      BUFFER DEFINITIONS
327      :      BUFFERS ARE 269 WORDS LONG AND ARE LOCATED AT LOC 4535(8)-7777(8)
328      :
329      004535      BUF1       =      004535   ;BUFFER 1 AT LOCATION 4535(8)
330      005152      BUF2       =      005152   ;BUFFER 2 AT LOCATION 5152(8)
331      005567      BUF3       =      005567   ;BUFFER 3 AT LOCATION 5567(8)
332      006204      BUF4       =      006204   ;BUFFER 4 AT LOCATION 6204(8)
333      006621      BUF5       =      006621   ;BUFFER 5 AT LOCATION 6621(8)
334      007321      BUF6       =      007321   ;BUFFER 6 AT LOCATION 7321(8)
335
336      :
337      :      BUFFER ASSIGNMENTS
338      :
338      004535      RDBUF      =      BUF1     ;READ/WRITE BUFFER
339      005152      PBNBUF     =      BUF2     ;BUFFER OF BAD PBN'S
340      005567      GDBLK      =      BUF3     ;DATA FOR GOOD SECTOR
341      006204      PRMBUF     =      BUF4     ;DATA PATTERN FOR PRIMARY REVECTOR
342      006204      REVBUF     =      BUF4     ;SECONDARY REVECTOR BUFFER

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 3-6
 EQUATES

343	006621	CMDBUF	=	BUF5	:READ COMMAND BUFFERS
344	006621	RCTBUF	=	BUF5	:RCT BLOCK BUFFER
345	006621	RBNBUF	=	BUF5	:RBN FORMAT BUFFER
346	006621	BDLST	=	BUF5	:HEAD VERIFICATION BUFFER
347	007321	CLBUF	=	BUF6	:USED IN FINAL CLEANUP
348	007321	IMAGE	=	BUF6	:FORMAT IMAGE BUFFER
349					:BUFFER EXCESS AFTER FORMAT IMAGE
350					:IS USED TO HOLD BLOCKS TO BE
351					:REVECTORED. MAX BLOCKS BEFORE
352					:REVECTOR ROUTINE IS CALLED VARIES
353					:WITH THE SIZE OF THE FORMAT BUFFER AREA
354	007775	BMAX	=	7775	:MAX BUFFER ADDRESS
355					
356					
357					

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 5
DATA STRUCTURES

```

1          .SBTTL  DATA STRUCTURES
2          :
3 000000   DMCODE  UDAFM,0,714,13,255.
4 000714   003023  ENTRY:  JMP          START          ;JUMP TC START LOCATION
5          :
6          DATA STRUCTURES
7          :
8          RETRY COUNTERS
9          :
10 000715   000000  UN.ERR: .WORD  0          ;UNSUCCESSFUL CMD RETRY CNTR
11 000716   000000  UN.ERT: .WORD  0          ;TRANSMISSION ERROR RETRY CNTR
12 000717   000000  UN.ERI: .WORD  0          ;INITIALIZATION ERROR RETRY CNTR
13 000720   000000  UN.SEK: .WORD  0          ;SEEK RETRY COUNT
14          :
15          :
16          READ COMMAND BLOCK
17          :
18          :
19 000721   WRBLK:
20 000721   100000  RDBLK: .WORD  100000      ;STATUS POINTER
21 000722   000000      .WORD  0          ;POINTER TO DATA BUFFER
22 000723   000000      .WORD  0          ;FIRST WORD OF EXPECTED HDR
23 000724   000000      .WORD  0          ;SECOND WORD " "
24 000725   000000      .WORD  0          ;REAL-TIME SDI COMMAND
25 000726   000000      .WORD  0          ;POINTER TO SDI BLOCK
26          :
27          :
28          DUMMY DOUBLE WORDS AND DUMMY SDI COMMAND
29 000727   000200  HSLIM: .WORD  200          ;HEADER CMP LIMIT
30 000730   001046      .WORD  SCR-5      ;POINTER TO SUBUNIT CHAR
31 000731   000000  DDUMMY: .WORD  0          ;DUMMY DOUBLE WORD FOR ONE
32 000732   000000      .WORD  0          ;BYTE OPERAND CONVERSION
33 000733   TEMP2:  ;ALSO USE AS TEMP
34 000733   000000  MULPC: .BLKW  2          ;MULTIPLICATION BUFFER
35 000735   000000      .WORD  0          ;RESERVED LOCATION (A+7)
36 000736   OFFSET: ;FOR EASIER REFERENCE
37 000736   TEMP:  .BLKW  2          ;USED FOR COMPUTATIONS
38          :
39          :
40          :
41          :
42          CURRENT UDA PORT
43 000740   000000  UNIT:  .WORD  0          ;SDI INTERCONNECT
44 000741   000000  UNNO:  .WORD  0          ;UNIT NUMBER ENTERED
45          :
46          MESSAGE TABLES
47          :
48 000742   CR.GST: MSG  GST,1,ST,7      ;GET STATUS
49 000746   CR.GCR: MSG  GCR,1,CR,11     ;GET CHARACTERISTICS
50 000752   CR.GSR: MSG  GSR,2,SCR,19.   ;GET SUBUNIT CHARACTERISTICS
51 000756   CR.DIS: MSG  DIS,2,ST,6      ;UNLOAD DRIVE
52 000762   CR.RUN: MSG  RUN,1,ST,6      ;LOAD DRIVE
53 000766   CR.ACC: MSG  ACC,3,ST,6      ;SET FORMAT ACCESS
54 000772   CR.CLR: MSG  DCLR,2,ST,6     ;DRIVE CLEAR
55 000776   CR.SEK: MSG  ISEEK,6,ST,6    ;SEEK
56 001002   CR.RCL: MSG  IRECAL,1,ST,6   ;RECALIBRATE
57 001006   CR.ERV: MSG  ERECOV,2,ST,6   ;ERROR RECOVERY COMMAND

```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 5-1
DATA STRUCTURES

58	001012		CR.ONL: MSG	ONLINE,2,ST,6		:ONLINE COMMAND
59			:			
60			:	MESSAGES AND COMMANDS		
61			:			
62	001016	004400	GST:	.WORD 000011*256.		:GET STATUS COMMAND
63	001017	103400	GCR:	.WORD 000207*256.		:GET CHARACTERISTICS
64	001020	104000	GSR:	.WORD 00210*256.		:GET SUBUNIT CHARACTERISTICS
65	001021	000000		.WORD 0		:SUBUNIT MASK
66	001022	102000	DIS:	.WORD 000204*256.		:UNLOAD DRIVE
67			.IF EQ PROD			:NOT PRODUCTION - NO SPINDOWN
68	001023	000000		.WORD 0		:NO SPIN DOWN MODIFIER
69			.IFF			:PRODUCTION - SPINDOWN
70				.WORD 1		:SPINDOWN
71			.ENDC			
72	001024	006000	RUN:	.WORD 000014*256.		:INITIATE LOAD
73	001025	100400	ACC:	.WORD 000201*256.		:ACCESS DIAG AND FMT CYL
74	001026	003006		.WORD 3006		:MASK BYTE/MODE BYTE
75	001027	002400	DCLR:	.WORD 000005*256.		:DRIVE CLEAR
76	001030	000374		.WORD 374		:BITS TO CLEAR
77	001031		ST:	.BLKW 7		:STATUS MESSAGE BUFFER
78	001040		CR:	.BLKW 11.		:CHARACTERISTICS MESSAGE BUFF
79	001053		SCR:	.BLKW 19.		:SUBUNIT CHARACTERISTICS BUFF
80	001076	005000	ISEEK:	.WORD 000012*256.		:INITIATE SEEK
81	001077	000000		.WORD 0		:..
82	001100	000000		.WORD 0		:..
83	001101	000000		.WORD 0		:..
84	001102	107000	IRECAL:	.WORD 000216*256.		:INITIATE RECAL
85	001103	003000	ERECOV:	.WORD 000006*256.		:ERROR RECOVERY COMMAND
86	001104	000000		.WORD 0		:RECOVERY LEVEL
87	001105	105400	ONLINE:	.WORD 000213*256.		:ONLINE COMMAND
88	001106	000377		.WORD 377		:COMMAND TIMEOUT (SECS)
89			:			
90			:	DISK LOCATION POINTERS		
91			:			
92	001107	000000	CURRBN:	.WORD 0		:CURRENT RBN
93	001110	000000		.WORD 0		:..
94	001111	000000	CURPBN:	.WORD 0		:CURRENT PBN
95	001112	000000		.WORD 0		:..
96	001113	000000	CURTRK:	.WORD 0		:CURRENT TRACK
97	001114	000000	CURBN:	.WORD 0		:CURRENT BLOCK NUMBER
98	001115	000000		.WORD 0		
99	001116		CURLBN:			:FOR RCT INIT
100	001116	000000	CURXBN:	.WORD 0		:CURRENT XBN NUMBER
101	001117	000000		.WORD 0		
102	001120		STASEC:			:FOR HEAD VERIFICATION ROUTINE
103	001120	000000	HOLDBN:	.WORD 0		:BLOCK NUMBER OF FIRST BLOCK ON CYL
104	001121	000000		.WORD 0		:..
105	001122	000000	HOLRBN:	.WORD 0		:BLOCK NUM OF FIRST RBN ON CYLINDER
106	001123	000000		.WORD 0		
107	001124	000000	HOLDPN:	.WORD 0		:PBN OF FIRST SECOTR ON TRACK
108	001125	000000		.WORD 0		
109	001126	000000	CYLNUM:	.WORD 0		:CURRENT CYLINDER NUMBER
110	001127	000000		.WORD 0		:..
111	001130	000000	SECTRK:	.WORD 0		:SECTORS/TRACK
112	001131	000000		.WORD 0		:..
113	001132	000000	SECTCY:	.WORD 0		:SECTORS/CYLINDER
114	001133	000000		.WORD 0		:..

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 5-2
DATA STRUCTURES

115	001134		LBNLBN: .BLKW	2	:LBN'S IN LBN AREA
116	001136		RBNLBN: .BLKW	2	:RBN'S IN LBN AREA
117	001140		XBNSEC: .BLKW	2	:SECTORS IN LBN AREA
118	001142		TRKCYL: .BLKW	2	:TRACKS/CYLINDER
119	001144		LBNCYL: .BLKW	2	:NUMBER OF LBN CYLINDERS
120	001146		LBNPCY: .BLKW	2	:LBN'S/CYLINDER
121	001150		RBNPCY: .BLKW	2	:RBN'S/CYLINDER
122	001152	000000	REVRBN: .WORD	0	:REVECTORED RBN NUMBER
123	001153	000000		0	:
124	001154	000000	CUROVL: .WORD	0	:CURRENT OVERLAY
125	001155		HGHPBN: .BLKW	2	:HIGHEST PBN IN LBN AREA
126			:		
127			:		
128			:		
129			STACK		
130			:		
131			:		
132			:		
133	001157			31.	:STACK
134	001216	000000	STACK: .WORD	0	:TOP OF STACK
135	001217	000000	STCKSV: .WORD	0	:STACK PTR TEMP SAVE
136			:		
137			:		
138			MISC	DEFINITIONS	
139			:		
140			:		
141	001220	000000	FLAG: .WORD	0	:FLAG WORD
142	001221	000000	FLAG1: .WORD	0	:FLAG WORD
143	001222	000000	ERFLAG: .WORD	0	:RE-FORMAT FLAG
144	001223	000000	WRFLG: .WORD	0	:RCT WRAP FLAG
145	001224	000000	BADPBN: .WORD	0	:POINTER TO PBNTAB ENTRY
146	001225	000000	ERRBUF: .WORD	0	:POINTER TO BEGINNING OF REVECTOR BUFFER
147	001226	000000	EMAX: .WORD	0	:MAX NUMBER OF REVECTORS BEFORE
148					:RCT UPDATE ROUTINE IS CALLED
149	001227	000000	ERR: .WORD	0	:NUMBER OF SECTORS IN ERROR
150	001230	000000	HOLD: .WORD	0	:DOUBLE WORD TEMP STORAGE
151	001231	000000		0	
152	001232	000000	EIMAGE: .WORD	0	:ADDRESS OF END IMAGE BLOCK
153	001233	000000	STARIT: .WORD	0	:STARTING ADDRESS OF THIS PASS
154	001234	000000	SKPCNT: .WORD	0	:OFFSET FOR FIRST READ CHECK
155	001235	000000	TBLK: .WORD	0	:INTERLEAVE
156					:6 - BI-LEAVE
157					:9 - TRI-LEAVE
158	001236		RCTTOT: .WORD	2500.	:ALSO RCT TOTAL HOLDING AREA
159	001236	004704	CUTOF: .WORD	0	:SECT/SECOND CUTOFF
160	001237	000000		0	:DOUBLE WORD
161	001240	000000	FCNT: .WORD	0	:COUNT OF USED FCT ENTRIES FOR FORMATTING
162	001241		FCTFMT: .BLKW	2	:SIZE OF ONE FCT COPY
163	001243		RCTFMT: .BLKW	2	:SIZE OF ONE RCT COPY
164	001245	000000	FCTCPY: .WORD	0	:NUMBER OF FCT COPIES
165	001246	000000	NEXT1: .WORD	0	:MULTI-COPY COUNTER
166	001247	000105	INI: .WORD	69.	:INITIAL VALUE FOR EDC
167	001250	000400	CNT: .WORD	SECS16	:COUNT FOR EDC
168	001251	000100	LTO: .WORD	100	:LONG TIMEOUT
169	001252	002000	STO: .WORD	1024.	:SHORT TIMEOUT (IN MILLESECS)
170	001253	000000	ERPNT: .WORD	0	:REVECTOR LIST POINTER
171	001254	000000	BUFPT: .WORD	0	:BUFFER POINTER FOR FCT READ

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 5-3
DATA STRUCTURES

172	001255	000000	REVCNT:	.WORD	0	:REVECTOR COUNT
173	001256	000000	FCTPTR:	.WORD	0	:POINT TO CURRENT LOCATION IN FCT LBOKL
174	001257	000001	FCTCNT:	.WORD	1	:CURRENT FCT BLOCK
175	001260	000000		.WORD	0	
176	001261	000000	FCTNPD:	.WORD	0	:NON-PAD FCT BLOCKS
177	001262	000000	RCTLBN:	.WORD	0	:LBN'S IN RCT
178	001263	000000	MNCNT:	.WORD	0	:USED FCT ENTRIES
179	001264	000016	DMBUF:	.REPT	14.	:MAINTENANCE BUFFER
180				.WORD	0	:MAKE SURE IT IS 0
181				.ENDM		
182	001302		DATE:	.BLKW	4	:DATE BUFFER
183	001306		SERNUM:	.BLKW	4	:SERIAL NUMBER
184	001312	000000	FCTREV:	.WORD	0	:FCT ENTRIES AT CERTAIN POINTS
185	001313	000000	LBNBAD:	.WORD	0	:TOTAL REVECTORED LBN'S
186	001314	000000	RCTBAD:	.WORD	0	:TOTAL BAD RCT BLOCKS
187	001315	000000	DBBAD:	.WORD	0	:TOTAL DBN BAD BLOCKS
188	001316	000000	XBBAD:	.WORD	0	:TOTAL LBN BAD BLOCKS
189	001317	060001	FCMSG:	.WORD	60001	:DUP CODE
190	001320	000000	IMSTAR:	.WORD	0	:POINTER TO START OF IMAGE
191	001321	000000	HPREA:	.WORD	0	:HEADER PREAMBLE LENGTH
192	001322	000000	DPREA:	.WORD	0	:DATA PREAMBLE LENGTH
193	001323	000000	ST.LBN:	.WORD	0	:STARTING LBN BITS
194	001324	000000	ST.RBN:	.WORD	0	:STARTING RBN BITS
195	001325	000000	ST.XBN:	.WORD	0	:STARTING XBN BITS
196	001326	000000	ST.DBN:	.WORD	0	:STARTING DBN BITS
197						
198						
199						
200						
201						
202						
203						
204						
205	001327	001057	OVLTBL:	.WORD	OVL.F1	:LENGTH OF FIRST OVERLAY
206	001330	013176		.WORD	OVS.F1	:LOW ORDER HOST ADDRESS
207	001331	000000		.WORD	0	:HIGH ORDER HOST ADDRESS
208	001332	001402		.WORD	OVL.F2	:LENGTH OF SECOND OVERLAY
209	001333	016304		.WORD	OVS.F2	:LOW ORDER HOST ADDRESS
210	001334	000000		.WORD	0	:HIGH ORDER HOST ADDRESS
211	001335	000422		.WORD	OVL.F3	:LENGTH OF THIRD OVERLAY
212	001336	023346		.WORD	OVS.F3	:LOW ORDER HOST ADDRESS
213	001337	000000		.WORD	0	:HIGH ORDER HOST ADDRESS
214	001340	000652		.WORD	OVL.F4	:LENGTH OF FOURTH OVERLAY
215	001341	024412		.WORD	OVS.F4	:LOW ORDER HOST ADDRESS
216	001342	000000		.WORD	0	:HIGH ORDER HOST ADDRESS
217	001343	001202		.WORD	OVL.F5	:LENGTH OF FIFTH OVERLAY
218	001344	026764		.WORD	OVS.F5	:LOW ORDER HOST ADDRESS
219	001345	000000		.WORD	0	:HIGH ORDER HOST ADDRESS
220	001346	000334		.WORD	OVL.F6	:LENGTH OF SIXTH OVERLAY
221	001347	032510		.WORD	OVS.F6	:LOW ORDER HOST ADDRESS
222	001350	000000		.WORD	0	:HIGH ORDER HOST ADDRESS
223	001351	000450		.WORD	OVL.F7	:LENGTH OF SEVENTH OVERLAY
224	001352	031370		.WORD	OVS.F7	:LOW ORDER HOST ADDRESS
225	001353	000000		.WORD	0	:HIGH ORDER HOST ADDRESS
226	001354	000573		.WORD	OVL.F8	:LENGTH OF EIGHTH OVERLAY
227	001355	021310		.WORD	OVS.F8	:LOW ORDER HOST ADDRESS
228	001356	000000		.WORD	0	:HIGH ORDER HOST ADDRESS

OVERLAY POINTERS

NOTE:

WHEN ADDING AN ENTRY TO THIS TABLE EQUATE
OVCNT MUST BE INCREMENTED

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 5-4
DATA STRUCTURES

229	001357	000205	.WORD	OVL.F9	:LENGTH OF NINTH OVERLAY
230	001360	042640	.WORD	OVS.F9	:LOW ORDER HOST ADDRESS
231	001361	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
232	001362	000045	.WORD	OVL.G2	:LENGTH OF TENTH OVERLAY
233	001363	033400	.WORD	OVS.G2	:LOW ORDER HOST ADDRESS
234	001364	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
235	001365	000225	.WORD	OVL.G3	:LENGTH OF ELEVENTH OVERLAY
236	001366	033512	.WORD	OVS.G3	:LOW ORDER HOST ADDRESS
237	001367	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
238	001370	001667	.WORD	OVL.G4	:LENGTH OF TWELFTH OVERLAY
239	001371	034164	.WORD	OVS.G4	:LOW ORDER HOST ADDRESS
240	001372	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
241	001373	000351	.WORD	OVL.G5	:LENGTH OF THIRTEENTH OVERLAY
242	001374	043252	.WORD	OVS.G5	:LOW ORDER HOST ADDRESS
243	001375	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
244	001376	000364	.WORD	OVL.G7	:LENGTH OF FOURTEENTH OVERLAY
245	001377	015334	.WORD	OVS.G7	:LOW ORDER HOST ADDRESS
246	001400	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
247	001401	000224	.WORD	OVL.G8	:LENGTH OF FIFTEENTH OVERLAY
248	001402	022676	.WORD	OVS.G8	:LOW ORDER HOST ADDRESS
249	001403	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
250	001404	000313	.WORD	OVL.H1	:LENGTH OF SIXTEENTH OVERLAY
251	001405	026136	.WORD	OVS.H1	:LOW ORDER HOST ADDRESS
252	001406	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
253	001407	001317	.WORD	OVL.G1	:LENGTH OF SEVENTEENTH OVERLAY
254	001410	010340	.WORD	OVS.G1	:LOW ORDER HOST ADDRESS
255	001411	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
256	001412	000471	.WORD	OVL.G6	:LENGTH OF EIGHTEENTH OVERLAY
257	001413	044174	.WORD	OVS.G6	:LOW ORDER HOST ADDRESS
258	001414	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
259	001415	001337	.WORD	OVL.H2	:LENGTH OF NINETEENTH OVERLAY
260	001416	037742	.WORD	OVS.H2	:LOW ORDER HOST ADDRESS
261	001417	000000	.WORD	0	:HIGH ORDER HOST ADDRESS
262	001420		.WORD	0	:FOR BUFFER OVERLAYS
			OVLBLK: .BLKW	4	

UDAFM - UDA FORMATTER DMACH X04.01 23-AJG-82 14:02:32 PAGE 6
DATA STRUCTURES

1					
2					
3					
4					
5					
6	001424		CONBLK: .BLKW 12.		:CONVERSION BUFFER
7					
8					
9					
10					
11	001440	000377	NUM: .WORD 255.		:NUMBER OF WORDS IN PATTERN
12	001441	004536	CBUF: .WORD RDBUF+1		:BUFFER TO COMPARE(NOT FIRST WORD)
13	001442	155555	FWRD: .WORD 155555		:FIRST WORD OF PATTERN
14	001443	133333	SWRD: .WORD 133333		:SECOND WORD OF PATTERN
15	001444	066666	TWRD: .WORD 066666		:THIRD WORD OF PATTERN
16	001445	177777	DWRD: .WORD 177777		:DIAGNOSTIC WORD(FIRST IN SECTOR)
17					
18					
19	001446	030206	EDC: .WORD 30206		:EDC FOR ABOVE DATA PATTERN
20	001447	147571	BAEEDC: .WORD 147571		:BAD EDC FOR RBN BLOCKS
21					
22					
23					
24					
25	001450	000000	ERRCNT: .WORD 0		:FOR TESTING VERIFICATION
26	001451	000000	SECCNT: .WORD 0		:SECTOR COUNT
27	001452	000000	N: .WORD 0		:NUMBER OF ORIGINAL CHECK PASS READ
28	001453	000000	N1: .WORD 0		:NUMBER OF ERROR READS
29	001454	000000	NN1: .WORD 0		:DITTO
30	001455	000000	CNTCYL: .WORD 0		:NUMBER OF CYLINDERS TO FORMAT
31	001456	000000			
32	001457	000000	HD.CUR: .WORD 0		:CURRENT HEADER
33	001460	000000	CURGRP: .WORD 0		:CURRENT GROUP
34	001461	000000	GRPCNT: .WORD 0		:NUMBER OF GROUPS TO DO
35	001462	000000	TRKCNT: .WORD 0		:NUMBER OF TRACKS TO DO
36	001463	000001	ONE: .WORD 1		:WORD CONSTANT OF 1
37	001464	000000			:DOUBLE WORD
38	001465	000002	TWOC: .WORD 2		:WORD CONSTANT OF 2
39	001466	000000			:DOUBLE WORD
40	001467	000000	SNDCNT: .WORD 0		:COUNT OF SECONDARY REVECTORS
41	001470	000000	RTYCNT: .WORD 0		:COUNT OF SECTORS RETRYED
42	001471		CURPNT: .WORD 0		:POINT FOR HEAD VERIFICATION
43	001471	000000	UPDPNT: .WORD 0		:POINTER FOR RCT UPDATE
44	001472	000000	TOTRCT: .WORD 0		:TOTAL LBN'S IN RCT'S
45	001473	000000			
46	001474	000000	RCTCNT: .WORD 0		:CURRENT RCT BLOCK
47	001475	000000	PCNT: .WORD 0		:PBN BLOCK COUNTER
48	001476	000000	COUNT: .WORD 0		:COUNT FOR XBN DLL
49	001477	000005	RETRY: .WORD 5		:RETRIES FROM SDI
50	001500	000000	RECOV: .WORD 0		:RECOVERY LEVELS SUPPORTE BY DRIVER
51	001501	000000	TMPTRY: .WORD 0		:TEMP FOR RETRY COUNT
52	001502	000000	RECTMP: .WORD 0		:TEMP FOR ERROR RECOVERY LEVEL

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 7
MATH SUBROUTINES

1			.SBTTL MATH SUBROUTINES	
2				
3				
4			SUBROUTINES	
5				
6			DOUBLE ADD ROUTINE	
7				
8			INPUT PARAMETERS	
9				
10			R3	CONTAINS POINTER TO OPERAND 1
11				
12			R4	CONTAINS POINTER TO OPERAND 2
13				
14			OUTPUT PARAMETER	
15				
16			R4	CONTAINS THE RESULT
17				
18				
19	001503		DADD:	PUSH R5 ;SAVE A SCRATCH REGISTER
20	001504			PUSH R1 ;SAVE ANOTHER
21	001505	104235		MOV (R3)+,R5 ;GET LOW ORDER OPERAND
22	001506	104131		MOV (R3),R1 ;GET HIGH ORDER OPERAND
23	001507	105245		ADD (R4)+,R5 ;ADD LOW ORDER OPERAND
24	001510	041512		BCC DADD1 ;BRANCH IF NO CARRY
25	001511	115401		INC R1 ;ADD ONE TO HIGH IF CARRY
26	001512	105141	DADD1:	ADD (R4),R1 ;ADD OP 2
27	001513	100141		MOV R1,(R4) ;SAVE HIGH ORDER
28	001514	100445		MOV R5,-(R4) ;SAVE LOW ORDER
29	001515			POP R1 ;RESTORE R1
30	001516			POP R5 ;RESTORE R5
31	001517	117403		DEC R3 ;RESTORE R3
32	001520	000000		RETURN
33				

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 8
MATH SUBROUTINES

1					
2	001521		DSUB:		
3				: ++	
4				: DOUBLE PRECISION FIXED POINT SUBTRACT ROUTINE	
5				: INPUTS:	
6				: R3 = POINTER TO OPERAND 1 (SUBTRAHEND)	
7				: R4 = POINTER TO OPERAND 2 (MINUEND)	
8				: OUTPUT:	
9				: R4 = POINTER TO RESULT WHERE (R4) = (R4) - (R3)	
10				: --	
11					
12					
13					
14	001521			PUSH R1,R5	: SAVE REGISTERS
15	001523	104245		MOV (R4)+,R5	: GET LO ORDER MINUEND
16	001524	104141		MOV (R4),R1	: GET HI ORDER MINUEND
17	001525	107135		SUB (R3),R5	: SUBTRACT LOW ORDER OPERANDS
18	001526	041530		BCC 10\$: POSITIVE RESULT
19	001527	117401		DEC R1	: BORROW FROM HI ORDER OPERAND
20	001530	107631	000001	SUB 1(R3),R1	: SUBTRACT HI ORDER OPERANDS
21	001532	100141		MOV R1,(R4)	: STORE HI ORDER RESULT
22	001533	100445		MOV R5,-(R4)	: STORE LO ORDER RESULT
23	001534			POP R5,R1	: RESTORE REGISTERS
24	001536	000000		RETURN	
25					

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 9
MATH SUBROUTINES

```

1 001537          DMUL:
2
3                : ++
4                : DOUBLE PRECISION FIXED POINT MULTIPLY ROUTINE
5                : INPUTS:
6                :       R3 = POINTER TO MULTIPLIER (SINGLE PRECISION)
7                :       R4 = POINTER TO MULTIPLICANT (DOUBLE PRECISION)
8                : OUTPUT:
9                :       R4 = POINTER TO RESULT WHERE (R4) = (R4) * (R3)
10               : --
11
12 001537          PUSH    R0,R3                ; SAVE R0 & R3
13 001541 104137   MOV     (R3),R0              ; GET MULTIPLIER
14 001542 051547   BNE    5$                    ; MULTIPLIER NOT = 0
15 001543 100147   MOV     R0,(R4)              ; LOAD LO ORDER RESULT
16 001544 100647 000001  MOV     R0,1(R4)          ; LOAD HI ORDER RESULT
17 001546 001562   BR     20$                    ; RETURN
18 001547 104140 000733 5$:  MOV     (R4),MULPC          ; COPY MULTIPLICANT FOR DADD
19 001551 104640 000001 000734  MOV     1(R4),MULPC+1
20 001554 104203 000733  MOV     #MULPC,R3
21 001556 117407 10$:  DEC     R0                ; ADJUST MULTIPLIER FOR *1
22 001557 011562   BEQ    20$                    ; MULTIPLIER = 0, EXIT
23 001560 021503   CALL   DADD                ; PERFORM ITERATIVE ADDS
24 001561 001556   BR     10$
25 001562          20$:  POP     R3,R0                ; RESTORE R0 & R3
26 001564 000000   RETURN
27

```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 10
MATH SUBROUTINES

```

1 001565          DDIV:
2
3          : ++
4          : DOUBLE PRECISION FIXED POINT DIVIDE
5          : INPUTS:
6          :   R3 = POINTER TO DIVISOR (SINGLE PRECISION)
7          :   HIGH ORDER WORD MUST BE ZERO
8          :   R4 = POINTER TO DIVIDENT (DOUBLE PRECISION)
9          : OUTPUT:
10         :   R3 = POINTER TO REMAINDER
11         :   R4 = POINTER TO QUOTIENT
12
13         : NOTE - THE CASES WHERE EITHER THE DIVISOR OR DIVIDENT ARE ZERO,
14         : ARE NOT CONSIDERED IN THIS ROUTINE.
15         : --
16
17 001565          PUSH   R0,R1,R2,R5          ; SAVE REGISTERS
18 001571 114007   CLR    R0                    ; CLP LO ORDER QUOTIENT REG
19 001572 114001   CLR    R1                    ; CLR HI ORDER QUOTIENT REG
20 001573 104132   MOV    (R3),R2              ; GET DIVISOR
21 001574 104645   MOV    1(R4),R5           ; GET HI ORDER DIVIDENT
22 001576 051603   BNE    20$                  ; DIVISOR NOT = 0
23 001577 104145   MOV    (R4),R5           ; GET LO ORDER DIVIDENT
24 001600 106052   CMP    R5,R2              ; IS DIVIDENT < DIVISOR ?
25 001601 041603   BCC    20$                  ; NO, CONTINUE
26 001602 001611   BR     30$                  ; YES, STOP
27 001603 021521   CALL   DSUB                     ; SYNTHESIZE DIVIDE
28 001604 105207   ADD    #1,R0              ; INCR LO ORDER QUOTIENT
29 001606 041574   BCC    10$                  ; DID NCT OVERFLOW
30 001607 115401   INC    R1                    ; ADJUST HI ORDER QUOTIENT
31 001610 0C1574   BR     10$
32 001611 104145   MOV    (R4),R5           ; GET REMAINDER
33 001612 100147   MOV    R0,(R4)          ; LOAD LO ORDER QUOTIENT
34 001613 100641   MOV    R1,1(R4)        ; LOAD HI ORDER QUOTIENT
35 001615 100135   MOV    R5,(R3)        ; LOAD REMAINDER
36 001616          POP    R5,R2,R1,R0        ; RESTORE REGISTERS
37 001622 000000   RETURN
38

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 11
MATH SUBROUTINES

1				:		
2				:	DOUBLE COMPARE	
3				:	INPUT PARAMETERS	
4				:		
5				:		
6				:	R3	CONTAINS A POINTER TO THE FIRST OPERAND
7				:		
8				:	R4	CONATINS A POINTER TO THE SECOND OPERAND
9				:		
10				:	OUTPUT PARAMETERS	
11				:		
12				:	THE FLAGS ARE SET AS IF A SINGLE PRECISSION 'CMP' HAD OCCURED	
13				:		
14	001623			DCMP:	PUSH R0	:SAVE R0 FOR USE AS SCRATCH
15	001624				PUSH R1	:SAVE R1 FOR USE AS SCRATCH
16	001625	104141			MOV (R4),R1	:GET LOW ORDER DEST OPERAND
17	001626	104647	000001		MOV 1(R4),R0	:GET HIGH ORDER DEST OPERAND
18	001630	106637	000001		CMP 1(R3),R0	:DO ACTUAL HIGH ORDER TEST
19	001632	011646			BEQ DCMP1	:GO DO ADDITIONAL TESTING
20	001633	041641			BCC DCMP2	:SRC HI, CLEAN UP AND RTN
21	001634			DCMP4:	POP R1	:RESTORE R1
22	001635				POP R0	:RESTORE R0
23	001636	106204	000000		CMP #0,R4	:SET CONDITION CODES - SRC LSS
24	001640	000000			RETURN	:AND RETURN
25	001641			DCMP2:	POP R1	:RESTORE R1
26	001642				POP R0	:RESTORE R0
27	001643	106204	077777		CMP #077777,R4	:SET CONDITION CODES - DST LSS
28	001645	000000			RETURN	:AND RETURN
29	001646	106131		DCMP1:	CMP (R3),R1	:TEST LOW ORDER
30	001647	051654			BNE DCMP3	:BRANCH IF NOT EQUAL
31	001650				POP R1	:RESTORE R1
32	001651				POP R0	:RESTORE R0
33	001652	106044			CMP R4,R4	:SET CONDITION CODES - EQUAL
34	001653	000000			RETURN	:AND RETURN
35	001654	106131		DCMP3:	CMP (R3),R1	:COMPARE AGAIN
36	001655	041641			BCC DCMP2	:BRANCH ON SRC HI
37	001656	001634			BR DCMP4	:BRANCH ON SRC LOW

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 12
SDI SUBROUTINES

1				.SBTTL SDI SUBROUTINES	
2				GET STATUS	
3				OUTPUT PARAMETERS	
4				CLEARs DRIVE STATUS AND GETS CHARACTERISTICS	
5				IF NOT ALREADY RECEIVED	
6					
7					
8					
9					
10					
11					
12	001657			GSTATS: PUSH R3	:PUSH R3
13	001660			PUSH R5	:PUSH R5
14	001661	104203	000742	STATST: MOV #CR.GST,R3	:POINT TO GET STATUS TABLE
15	001663	022010		CALL TALK	:GET STATUS
16	001664	104207	001031	MOV #ST,R0	:POINT TO SUBUNIT CHARACTERISTICS
17	001666	104673	000002	MOV ST.ÉR(B(R0),R3	:GET ERROR BYTE
18	001670	103203	177420	BIC #ST.DF+HIBYTE,R3	:CLEAR HIGH BYTE AND DF BIT
19	001672	115003		TST R3	:ANY NEED TO ISSUE DRIVE CLEAR ?
20	001673	011677		BEQ STSK1	:NOPE - SKIP IT
21	001674	104030	001030	MOV R3,DCLR+1	:STORE MASK IN DRIVE CLR COMMAND
22	001676	022234		CALL CLEAR	:DO A DRIVE CLEAR
23	001677	104205	001032	STSK1: MOV #ST+1,R5	:POINT TO FIRST WORD OF STATUS
24	001701	104253		MOV (R5)+,R3	:GET FIRST WORD OF STATUS
25	001702	104202	000001	MOV #1,R2	:ERROR SUBCODE IN CASE
26	001704	102203	000040	BIT #ST.DR,R3	:IS DRIVE IN DIAGNOSTIC REQUEST MODE
27	001706	052005		BNE STPNIC	:YES, WE LOSE
28	001707	115402		INC R2	:ERROR SUBCODE 2
29	001710	102203	000001	BIT #ST.RU,R3	:IS RUN STOP SWITCH OUT
30	001712	012005		BEQ STPNIC	:YES, LOSE AGAIN
31	001713	104202	000004	MOV #4,R2	:SUBCODE
32	001715	102203	000002	BIT #ST.PS,R3	:PORT SWITCH OUT ?
33	001717	012005		BEQ STPNIC	:YES - DIE PAINFULLY
34	001720	104032		MOV R3,R2	:GET STATUS MODE BYTE
35	001721	110702		SWAB R2	:SWITCH WRITE PROTECT TO LOW BYTE
36	001722	102302	001021	BIT GSR+1,R2	:THIS SUBUNIT WRITE PROTECTED ?
37	001724	011740		BEQ SRCK	:IF NOT CHECK IF SPINNING
38	001725	104202	000003	MOV #3,R2	:IN CASE IT'S FATAL
39	001727	102200	000001	001221 BIT #WP,FLAG1	:BEEN HERE ONCE ?
40	001732	052005		BNE STPNIC	:YUP - GIVE UP
41	001733	101200	000001	001221 BIS #WP,FLAG1	:SET BEEN HERE FLAG
42	001736	022226		CALL ACCESS	:TRY TO RESET IT
43	001737	001661		BR STATST	:AND SEE IF IT WORKED
44	001740	102203	000020	SRCK: BIT #ST.SR,R3	:IS PACK SPINNING?
45	001742	051745		BNE STFORM	:YES, TEST FOR FORMAT ENABLE
46	001743	022152		CALL LOAD	:NO, SPIN PACK
47	001744	001661		JMP STATST	:SEE IF ANYTHING CHANGED
48	001745	102203	002000	STFORM: BIT #ST.FO,R3	:IS FORMATTING ENABLED?
49	001747	051752		BNE STDIAG	:YES, TEST FOR DIAG ACCESS
50	001750	022226		CALL ACCESS	:NO, SET UP DIAG/FORM ACCESS
51	001751	001661		JMP STATST	:SEE IF ANYTHING CHANGED
52	001752	102203	001000	STDIAG: BIT #ST.DB,R3	:IS DIAG CYL ACCESS ALLOWED
53	001754	051757		BNE STWLK	:YES, CHECK FOR ERRORS
54	001755	022226		CALL ACCESS	:NO, SET UP DIAG/FORM ACCESS
55	001756	001661		JMP STATST	:SEE IF ANYTHING CHANGED
56	001757	104153		STWLK: MOV (R5),R3	:GET SECOND STATUS WORD
57	001760	102203	000010	BIT #ST.WE,R3	:ANY WRITE ENABLE ERRORS

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 12-1
SDI SUBROUTINES

58	001762	011765				BEQ	CHAR		:NO, GET CHARACTERISTICS
59	001763	022226				CALL	ACCESS		:TRY ENABLING LOGICAL WRITE
60	001764	001661				BR	STATST		:AND CHECK WORLD AGAIN
61	001765	102200	010000	001220	CHAR:	BIT	#CHRDNE,FLAG		:CHARACTERISTICS ALREADY RECEIVED ?
62	001770	052002				BNE	STSKP		:YUP - NO NEED TO GET AGAIN
63	001771	104203	000746			MOV	#CR.GCR,R3		:POINT TO GET CHAR CMD TABLE
64	001773	022010				CALL	TALK		:GET CHARACTERISTICS
65	001774	104203	000752			MOV	#CR.GSR,R3		:GET SUBUNIT CHARACTERISTICS
66	001776	022010				CALL	TALK		:GET THEM
67	001777	101200	C10000	001220		BIS	#CHRDNE,FLAG		:SET CHAR DONE BIT
68	002002				STSKP:	POP	R5		:RESTORE R5
69	002003					POP	R3		:RESTORE R3
70	002004	000000				RETURN			:RETURN TO CALLER
71	002005	104201	000001		STPNIC:	MOV	#1,R1		:INDICATE STATUS FAILURE
72	002007	022542				CALL	ERRMNT		:SEND ERROR MSG AND QUIT

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 13
SDI SUBROUTINES

```

1
2
3
4
5
6
7
8
9
10
11
12
13 002010          TALK:  PUSH    R3
14 002011          PUSH    R4
15 002012 104663 000001 LOOP1:  MOV     1(SP),R3      ;RESTORE R3 FOR RETRIES
16 002014 104237          MOV     (R3)+,R0      ;GET COMMAND ADDRESS
17 002015 104231          MOV     (R3)+,R1      ;GET COMMAND SIZE
18 002016 104302 900740          MOV     UNIT,R2      ;MAKE SURE HAVE INTERCONNECT
19 002020 060004          XFC     SEND          ;SEND GET STATUS COMMAND
20 002021 115001          TST     R1            ;SUCCESSFUL?
21 002022 012030          BEQ     MSG1         ;YES, BRANCH
22 002023 115400 000716          INC     UN.ERT      ;INCREMENT ERROR COUNT
23 002025 104201 000002          MOV     #2,R1      ;ERROR NUMBER IN CASE
24 002027 002075          BR     TCLEAR      ;DO RECOVERY
25 002030 102200 100000 001220 MSG1:  BIT     #RTY,FLAG    ;IN A RETRY ?
26 002033 052036          BNE     LOOP2      ;YES - DON'T CLEAR COUNTER
27 002034 114000 000716          CLR     UN.ERT     ;FOR RESET
28 002036 104231          LOOP2: MOV     (R3)+,R1  ;POINT TO RCV BUFFER
29 002037 104137          MOV     (R3),R0    ;SET SIZE OF REPLY
30 002040 104302 000740          MOV     UNIT,R2    ;MAKE SURE HAVE INTERCONNECT
31 002042 060005          XFC     RCV        ;RCV REPLY TO GET STATUS
32 002043 115001          TST     R1            ;SUCCESSFUL?
33 002044 012052          BEQ     TALKDN     ;YES, CHECK STATUS
34 002045 115400 000715          INC     UN.ERR     ;INCREMENT ERROR COUNT
35 002047 104201 000064          MOV     #4,R1      ;ERROR CODE IN CASE
36 002051 002075          BR     TCLEAR      ;DO RECOVERY
37 002052 106207 000175          TALKDN: CMP     #UNSEC,R0 ;WAS CMD UNSUCCESSFUL?
38 002054 052062          BNE     TALKRT     ;YES, DONE
39 002055 115400 000715          INC     UN.ERR     ;INCREMENT ERROR COUNT
40 002057 104201 000003          MOV     #3,R1      ;ERROR CODE IN CASE
41 002061 002075          BR     TCLEAR      ;NO, TRY AGAIN
42 002062 102200 100000 001220 TALKRT: BIT     #RTY,FLAG    ;IN A RETRY ?
43 002065 052070          BNE     TALKP     ;YUP - SKIP CLEAR
44 002066 114000 000715          CLR     UN.ERR     ;CLEAR FOR REST
45 002070          TALKP: POP     R4      ;RESTORE R4
46 002071          POP     R3        ;RESTORE R3
47 002072 000000          RETURN
48 002073 114002          ERRT:  CLR     R2      ;CLEAR SUBCODE
49 002074 022542          CALL   ERRMNT     ;ERROR EXIT
50 002075 102200 100000 001220 TCLEAR: BIT     #RTY,FLAG    ;IN A RETRY ?
51 002100 052107          BNE     TALKIP     ;YUP - SKIP FLAG SET AND STACK SAVE
52 002101 101200 100000 001220          BIS     #RTY,FLAG    ;SET FLAG
53 002104 104060 001217          MOV     SP,STCKSV ;SAVE STACK POINTER
54 002106 002111          BR     TALIP1     ;SKIP RETRY HANDLING
55 002107 104306 001217          TALKIP: MOV    STCKSV,SP ;RESTORE STACK POINTER
56 002111 106300 001477 000716 TALIP1: CMP     RETRY,UN.ERT ;DONE RETRIES ?
57 002114 072073          BMI     ERRT      ;YUP - CAN IT

```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 13-2
SDI SUBROUTINES

```

115 002225 022542          CALL   ERRMNT          ;DIE
116                      ;
117                      ;   ACCESS ROUTINE
118                      ;
119 002226                ACCESS: PUSH   R3          ;SAVE R3
120 002227 104203 000766  MOV    #CR.ACC,R3      ;POINT TO ACCESS TABLE
121 002231 022010        CALL   TALK          ;SEND ACCESS CMD
122 002232                POP    R3          ;RESTORE R3
123 002233 000000        RETURN                ;RETURN TO CALLER
124                      ;
125                      ;   CLEAR ROUTINE
126                      ;
127 002234                CLEAR: PUSH   R3          ;SAVE R3
128 002235 104203 000772  MOV    #CR.CLR,R3      ;POINT TO CLEAR TABLE
129 002237 022010        CALL   TALK          ;SEND CLEAR CMD
130 002240                POP    R3          ;RESTORE R3
131 002241 000000        RETURN                ;RETURN TO CALLER
132                      ;
133                      ;   SEEK ROUTINE
134                      ;
135 002242                SEEK:  PUSH   R3          ;SAVE R3
136 002243                PUSH   R0          ;SAVE R0
137 002244 104302 000740  SEEK0: MOV    UNIT,R2      ;MAKE SURE HAVE UNIT
138 002246 104203 000776  MOV    #CR.SEK,R3      ;POINT TO SEEK TABLE
139 002250 022010        CALL   TALK          ;SEND SEEK COMMAND
140 002251 104303 001252  MOV    STO,R3        ;SHORT TIMEOUT
141 002253 022642        SEEK1: CALL  STATVL      ;CHECK FOR STATUS VALIDITY
142 002254 052300        BNE    SEEK5          ;IF NOT ZERO - DIE
143 002255 102201 000002  BIT    #ATTN,R1      ;ANY PROBLEMS
144 002257 052301        BNE    SEEK2          ;YES, BRANCH
145 002260 102201 100000  BIT    #RWRDY,R1     ;NO, DONE?
146 002262 052272        BNE    SEEK6          ;ALL DONE
147 002263 117403        DEC    R3            ;DECREMENT COUNTER
148 002264 012312        BEQ    SEEK3          ;IF ZERO THEN DEAD
149 002265 104207 000240  MOV    #160.,R0      ;1MS DELAY
150 002267 117407        SEEK7: DEC    R0        ;DECREMENT COUNTER
151 002270 052267        BNE    SEEK7          ;DELAY LOOP
152 002271 002253        BR     SEEK1          ;TRY AGAIN
153 002272 114001        SEEK6: CLR    R1        ;CLEAR ERROR FLAG
154 002273 114000 000720  SEEK4: CLR    UN.SEK      ;FOR RESET
155 002275                POP    R0          ;YES, RESTORE R0
156 002276                POP    R3          ;RESTORE R3
157 002277 000000        RETURN                ;RETURN TO CALLER
158 002300 022362        SEEK5: CALL  INITPT      ;INIT THE DRIVE
159 002301 115400 000720  SEEK2: INC    UN.SEK      ;INCREMENT RETRY COUNTER
160 002303 106300 001477 000720  CMP    RETRY,UN.SEK    ;HAVE WE DONE ALL RETRIES?
161 002306 012312        BEQ    SEEK3          ;YES, PANIC
162 002307 021657        CALL  GSTATS          ;PANIC AND CALL GET STATUS
163 002310 022146        CALL  RECAL          ;RECAL DRIVE
164 002311 002244        BR     SEEK0          ;AND TRY AGAIN
165 002312 104201 177775  SEEK3: MOV    #-3,R1      ;SET ERROR CODE
166 002314 002273        BR     SEEK4          ;RESTORE REGS AND RETURN
167                      ;
168                      ;   UNLOAD ROUTINE
169                      ;
170 002315                DISCON: PUSH   R3          ;SAVE R3
171 002316 104203 000756  MOV    #CR.DIS,R3      ;DISCONNECT WITH

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 13-3
SDI SUBROUTINES

172	002320	022010			CALL TALK	:SEND UNLOAD CMD
173	002321				POP R3	:RESTORE R3
174	002322	000000			RETURN	:RETURN TO CALLER
175				:		
176				:		
177				:	ONLINE ROUTINE	
178	002323			ONLIN:	PUSH R3	:SAVE R3
179	002324	104203	001012		MOV #CR.ONL,R3	:ONLINE COMMAND
180	002326	022010			CALL TALK	:BRING DRIVE ONLINE
181	002327				POP R3	:RESTORE R3
182	002330	000000			RETURN	:RETURN TO CALLER
183				:		
184				:	INITIALIZE ROUTINE	
185				:		
186	002331			INITIT:	PUSH R1	:SAVE R1
187	002332				PUSH R3,R4	:SAVE R3 AND R4
188	002334	104204	000001		MOV #1,R4	:START WITH PORT 0
189	002336	104203	000004		MOV #4,R3	:INIT PORT COUNTER
190	002340	104042		INIT5:	MOV R4,R2	:SET UP INTERCONNECT
191	002341	060011			XFC DINIT	:INIT DRIVE
192	002342	104207	066540		MOV #28000.,R0	:TIMER (APPROX 2 SECS)
193	002344	022642		ATTN1:	CALL STATVL	:CHECK STATUS VALIDITY
194	002345	052353			BNE AOUT	:IF NOT ZERO - NO GOOD
195	002346	117407			DEC R0	:DEC COUNT
196	002347	012353			BEQ AOUT	:IF ZERO THEN DEAD
197	002350	102201	000001		BIT #RCVRDY,R1	:IS REECEIVER READY SET ?
198	002352	012344			BEQ ATTN1	:NO, TRY AGAIN
199	002353	110204		AOUT:	ROL R4	:NEXT PORT
200	002354	117403			DEC R3	:DECREMENT COUNTER
201	002355	052340			BNE INIT5	:IF NOT DONE DO NEXT PORT
202	002356				POP R4,R3	:RESTORE R3 AND R4
203	002360				POP R1	:RESTORE R1
204	002361	000000			RETURN	:AND RETURN TO CALLER
205				:		
206				:		
207				:	INIT GIVEN PORT	
208				:		
209	002362	104302	000740	INITPT:	MOV UNIT,R2	:GET PORT NUMBER
210	002364	060011			XFC DINIT	:DO THE INIT
211	002365	104207	066540		MOV #28000.,R0	:1 SECOND TIMER
212	002367	022642		INITP1:	CALL STATVL	:VALIDATE STATUS
213	002370	052377			BNE INITDD	:DEAD IF NOT VALID
214	002371	117407			DEC R0	:DECREMETN COUTNER
215	002372	012377			BEQ INITDD	:DEAD IF COUNT EXPIRED
216	002373	102201	000001		BIT #RCVRDY,R1	:DONE INIT ?
217	002375	012367			BEQ INITP1	:NOPE - KEEP TRYING
218	002376	000000			RETURN	:EXIT
219	002377	104201	000024	INITDD:	MOV #20.,R1	:ERROR CODE
220	002401	114002			CLR R2	:NO SUBCODE
221	002402	022542			CALL ERRMNT	:ERROR EXIT

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 14
OVERLAY PROCESSING ROUTINES

```

1
2
3
4
5
6
7
8 002403 104204 001327
9 002405 105014
10 002406 104203 003023
11 002410 022510
12 002411 115007
13 002412 052417
14 002413 114000 000715
15 002415
16 002416 003023
17 002417 106300 001477 000715
18 002422 012426
19 002423 115400 000715
20 002425 002410
21
22
23
24 002426 104012
25 002427 101200 000020 001221
26
27
28
29 002433 104201 000005
30 002434 022542

```

```

.SBTTL OVERLAY PROCESSING ROUTINES
OVERLAY PROCESSING ROUTINES
R1 = OFFSET INTO TABLE
NEXT CALLS OVERLAY FOR NEXT CODE OVERLAY
NEXT:  MOV    #OVL TBL,R4      ;GET POINTER TO OVERLAY TABLE
      ADD    R1,R4            ;INDEX INTO TABLE
      MOV    #START,R3       ;UDA ADDRESS TO LOAD AT
NEXT5: CALL   OVRLAY          ;CALL ROUTINE TO DO OVERLAY
      TST    RO               ;CHECKSUM O.K. ??
      BNE    OERR            ;YES - RETRY IF POSSIBLE
      CLR    UN.ERR          ;CLEAR ERROR COUNT
      POP    R1              ;POP CURRENT RETURN ADDRESS
      BR     START           ;GO TO OVERLAY
OERR:  CMP    RETRY,UN.ERR    ;DONE ALL RETRIES ?
      BEQ    OERR2           ;YUP
      INC    UN.ERR          ;INC ERROR AND
      BR     NEXT5          ;TRY AGAIN
DEAD HOST EXIT FOR ALL ROUTINES
OERR2: MOV    R1,R2           ;GET ERROR CODE FROM XFC
      BIS    #DEAD,FLAG1     ;INDICATE HOST GONE
UNIBUS ERROR EXIT FOR ALL ROUTINES
UERR:  MOV    #5,R1           ;SET UNIBUS I/O ERROR
      CALL   ERRMNT          ;ERROR RETURN

```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 16
 OVERLAY PROCESSING ROUTINES

1				:		
2				:		
3				:		
4				:		
5				:		
6				:		
7	002510	104647	000001	OVRLAY:	MOV	HSTLO(R4),R0 ;LOW ORDER UNIBUS ADDRESS
8	002512	104641	000002		MOV	HSTHI(R4),R1 ;HI ORDER UNIBUS ADDRESS
9	002514	104642	000000		MOV	LEN(R4),R2 ;WORD COUNT TO OVERLAY
10	002516	060013			XFC	UREAD ;ISSUE UNIBUS READ
11	002517	115001			TST	R1 ;ANY UNIBUS PROBLEMS ?
12	002520	052432			BNE	UERR ;YUP - EXIT WITH UNIBUS ERROR
13	002521	000000			RETURN	;RETURN TO CALLING ROUTINE
14				:		
15				:		
16				:		
17	002522	104207	001264	SNDMNT:	MOV	#DMBUF,R0 ;POINT TO BUFFER
18	002524	104201	000016		MOV	#DMBUFL,R1 ;LENGTH
19	002526	060016			XFC	MAINTR ;ISSUE COMMAND
20	002527	115001			TST	R1 ;ANY UNIBUS PROBLEMS ?
21	002530	052432			BNE	UERR ;YUP -- EXIT WITH UNIBUS ERROR
22	002531	000000			RETURN	
23				:		
24				:		
25				:		
26				:		
27				:		
28	002532	104207	001264	RCVMNT:	MOV	#DMBUF,R0 ;POINT TO BUFFER
29	002534	104201	000016		MOV	#DMBUFL,R1 ;SIZE OF BUFFER
30	002536	060017			XFC	MAINTW ;RECEIVE MAINT WRITE DATA
31	002537	115001			TST	R1 ;ANY UNIBUS PROBLEMS ?
32	002540	052432			BNE	UERR ;YUP - EXIT WITH UNIBUS ERROR
33	002541	000000			RETURN	

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 17
 MISCELLANEOUS COMMON ROUTINES

```

1
2
3
4
5
6
7
      .SBTTL MISCELLANEOUS COMMON ROUTINES
      .....
      ERROR RETURN TO HOST
8 002542 102200 000010 001221 ERRMNT: BIT      #ERDN,FLAG1      ;TRIED TO QUIT YET ?
9 002545 052562                BNE      ALLOVR      ;YUP - DISCON FAILED
10 002546 101200 000010 001221    BIS      #ERDN,FLAG1    ;SET FLAG
11 002551 104207 001264                MOV      #DMBUF,R0    ;POINT TO MAINT BUFFER
12 002553 100171                MOV      R1,(R0)      ;PUT ERROR NUMBER IN MSG
13 002554 100672 000001                MOV      R2,1(R0)    ;PUT IN ERROR SUBCODE
14 002556 104201 000063                MOV      #G6,R1     ;ERROR MESSAGE OVERLAY
15 002560 022435                CALL     PAGE        ;BRING IT IN
16 002561 022315                CALL     DISCON      ;DISCONNECT/SPINDOWN DRIVE
17 002562 114007                ALLOVR: CLR      R0   ;IN CASE O.K
18 002563 102200 000020 001221    BIT      #DEAD,FLAG1 ;DIE OR JUST QUIT ?
19 002566 012570                BEQ     ALLOV1      ;JUST QUIT
20 002567 115407                INC     R0          ;MAKE NON ZERO
21 002570 060021                ALLOV1: XFC      DONE ;EXIT DM MODE
22
23
24
      .....
      ERROR RETRY SEQUENCER
25 002571 104300 001502 001104 ERRHND: MOV      RECTMP,ERECOV+1 ;STORE LEVEL IN COMMAND
26 002574 104203 001006                MOV      #CR.ERV,R3 ;POIT TO COMMAND
27 002576 022010                CALL     TALK        ;DO ERROR RECOVERY
28 002577 000000                RETURN              ;RETURN

```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 22
 DUP DM<->HOST STARTUP OVERLAY

```

1          .SBTTL  DUP DM<->HOST STARTUP OVERLAY
2          :
3          :
4          :
5          003023  003344      START:  JMP      START3          ;SKIP LOCAL DATA STORAGE
6          :
7          :
8          :
9          :
10         :
11         :
12         003024      MSGTBL: .ENABL  LC
13         003024  003464      .WORD   DATCON          ;DATE CONVERSION ROUTINE
14         003025  000016      .WORD   MSG1LN         ;MESSAGE LENGTH
15         003026  010000      10$:   .WORD   10000         ;DUP WORD
16         003027      105      156    164 .ASCIZ  'Enter date <MM-DD-YYYY> '
17         000016      MSG1LN  =      .-10$         ;MESSAGE LENGTH
18         :
19         003044  003705      .WORD   UNITCN         ;UNIT NUMBER HANDLER
20         003045  000022      .WORD   MSG2LN         ;MESSAGE LENGTH
21         003046  020001      20$:   .WORD   20001         ;DUP WORD
22         003047      105      156    164 .ASCIZ  'Enter unit number to format <0>: '
23         000022      MSG2LN  =      .-20$         ;MESSAGE LENGTH
24         :
25         003070  004236      .WORD   EXTFCT         ;EXISTING FCT ?
26         003071  000026      .WORD   MSG5LN         ;MESSAGE LENGTH
27         003072  020004      50$:   .WORD   20004         ;DUP WORD
28         003073      125      163    145 .ASCIZ  'Use existing bad block information <Y>: '
29         000026      MSG5LN  =      .-50$
30         :
31         003120  003762      .WORD   DLLFLE         ;DOWN-LINE LOAD FILE
32         003121  000016      .WORD   MSG7LN         ;MESSAGE LENGTH
33         003122  020005      70$:   .WORD   20005         ;DUP WORD
34         003123      125      163    145 .ASCIZ  'Use Down-line load <N>: '
35         000016      MSG7LN  =      .-70$
36         :
37         003140  004261      .WORD   CONBAD         ;CONTINUE IF BAD ?
38         003141  000035      .WORD   MSG6LN         ;MESSAGE LENGTH
39         003142  020006      60$:   .WORD   20006         ;DUP WORD
40         003143      103      157    156 .ASCIZ  'Continue if bad block information is inaccessible <N>: '
41         000035      MSG6LN  =      .-60$
42         :
43         003177  003723      .WORD   SERCON         ;SERIAL NUMBER HANDLER
44         003200  000022      .WORD   MSG4LN         ;MESSAGE LENGTH
45         003201  010007      40$:   .WORD   10007         ;DUP WORD
46         003202      105      156    164 .ASCIZ  'Enter a non-zero serial number: '
47         000022      MSG4LN  =      .-40$
48         :
49         003223  000000      .WORD   0              ;END FLAG
50         :
51         :
52         :
53         003224  030000      FMTSTA: .WORD   30000         ;DUP WORD
54         003225      106      157    162 .ASCIZ  'Format begun'         ;MESSAGE
55         000010      FMSTL  =      .-FMTSTA         ;LENGTH
56         :
57         :
58         :
59         :
60         :
61         :
62         :
63         :
64         :
65         :
66         :
67         :
68         :
69         :
70         :
71         :
72         :
73         :
74         :
75         :
76         :
77         :
78         :
79         :
80         :
81         :
82         :
83         :
84         :
85         :
86         :
87         :
88         :
89         :
90         :
91         :
92         :
93         :
94         :
95         :
96         :
97         :
98         :
99         :
100        :
101        :
102        :
103        :
104        :
105        :
106        :
107        :
108        :
109        :
110        :
111        :
112        :
113        :
114        :
115        :
116        :
117        :
118        :
119        :
120        :
121        :
122        :
123        :
124        :
125        :
126        :
127        :
128        :
129        :
130        :
131        :
132        :
133        :
134        :
135        :
136        :
137        :
138        :
139        :
140        :
141        :
142        :
143        :
144        :
145        :
146        :
147        :
148        :
149        :
150        :
151        :
152        :
153        :
154        :
155        :
156        :
157        :
158        :
159        :
160        :
161        :
162        :
163        :
164        :
165        :
166        :
167        :
168        :
169        :
170        :
171        :
172        :
173        :
174        :
175        :
176        :
177        :
178        :
179        :
180        :
181        :
182        :
183        :
184        :
185        :
186        :
187        :
188        :
189        :
190        :
191        :
192        :
193        :
194        :
195        :
196        :
197        :
198        :
199        :
200        :
201        :
202        :
203        :
204        :
205        :
206        :
207        :
208        :
209        :
210        :
211        :
212        :
213        :
214        :
215        :
216        :
217        :
218        :
219        :
220        :
221        :
222        :
223        :
224        :
225        :
226        :
227        :
228        :
229        :
230        :
231        :
232        :
233        :
234        :
235        :
236        :
237        :
238        :
239        :
240        :
241        :
242        :
243        :
244        :
245        :
246        :
247        :
248        :
249        :
250        :
251        :
252        :
253        :
254        :
255        :
256        :
257        :
258        :
259        :
260        :
261        :
262        :
263        :
264        :
265        :
266        :
267        :
268        :
269        :
270        :
271        :
272        :
273        :
274        :
275        :
276        :
277        :
278        :
279        :
280        :
281        :
282        :
283        :
284        :
285        :
286        :
287        :
288        :
289        :
290        :
291        :
292        :
293        :
294        :
295        :
296        :
297        :
298        :
299        :
300        :
301        :
302        :
303        :
304        :
305        :
306        :
307        :
308        :
309        :
310        :
311        :
312        :
313        :
314        :
315        :
316        :
317        :
318        :
319        :
320        :
321        :
322        :
323        :
324        :
325        :
326        :
327        :
328        :
329        :
330        :
331        :
332        :
333        :
334        :
335        :
336        :
337        :
338        :
339        :
340        :
341        :
342        :
343        :
344        :
345        :
346        :
347        :
348        :
349        :
350        :
351        :
352        :
353        :
354        :
355        :
356        :
357        :
358        :
359        :
360        :
361        :
362        :
363        :
364        :
365        :
366        :
367        :
368        :
369        :
370        :
371        :
372        :
373        :
374        :
375        :
376        :
377        :
378        :
379        :
380        :
381        :
382        :
383        :
384        :
385        :
386        :
387        :
388        :
389        :
390        :
391        :
392        :
393        :
394        :
395        :
396        :
397        :
398        :
399        :
400        :
401        :
402        :
403        :
404        :
405        :
406        :
407        :
408        :
409        :
410        :
411        :
412        :
413        :
414        :
415        :
416        :
417        :
418        :
419        :
420        :
421        :
422        :
423        :
424        :
425        :
426        :
427        :
428        :
429        :
430        :
431        :
432        :
433        :
434        :
435        :
436        :
437        :
438        :
439        :
440        :
441        :
442        :
443        :
444        :
445        :
446        :
447        :
448        :
449        :
450        :
451        :
452        :
453        :
454        :
455        :
456        :
457        :
458        :
459        :
460        :
461        :
462        :
463        :
464        :
465        :
466        :
467        :
468        :
469        :
470        :
471        :
472        :
473        :
474        :
475        :
476        :
477        :
478        :
479        :
480        :
481        :
482        :
483        :
484        :
485        :
486        :
487        :
488        :
489        :
490        :
491        :
492        :
493        :
494        :
495        :
496        :
497        :
498        :
499        :
500        :
501        :
502        :
503        :
504        :
505        :
506        :
507        :
508        :
509        :
510        :
511        :
512        :
513        :
514        :
515        :
516        :
517        :
518        :
519        :
520        :
521        :
522        :
523        :
524        :
525        :
526        :
527        :
528        :
529        :
530        :
531        :
532        :
533        :
534        :
535        :
536        :
537        :
538        :
539        :
540        :
541        :
542        :
543        :
544        :
545        :
546        :
547        :
548        :
549        :
550        :
551        :
552        :
553        :
554        :
555        :
556        :
557        :
558        :
559        :
560        :
561        :
562        :
563        :
564        :
565        :
566        :
567        :
568        :
569        :
570        :
571        :
572        :
573        :
574        :
575        :
576        :
577        :
578        :
579        :
580        :
581        :
582        :
583        :
584        :
585        :
586        :
587        :
588        :
589        :
590        :
591        :
592        :
593        :
594        :
595        :
596        :
597        :
598        :
599        :
600        :
601        :
602        :
603        :
604        :
605        :
606        :
607        :
608        :
609        :
610        :
611        :
612        :
613        :
614        :
615        :
616        :
617        :
618        :
619        :
620        :
621        :
622        :
623        :
624        :
625        :
626        :
627        :
628        :
629        :
630        :
631        :
632        :
633        :
634        :
635        :
636        :
637        :
638        :
639        :
640        :
641        :
642        :
643        :
644        :
645        :
646        :
647        :
648        :
649        :
650        :
651        :
652        :
653        :
654        :
655        :
656        :
657        :
658        :
659        :
660        :
661        :
662        :
663        :
664        :
665        :
666        :
667        :
668        :
669        :
670        :
671        :
672        :
673        :
674        :
675        :
676        :
677        :
678        :
679        :
680        :
681        :
682        :
683        :
684        :
685        :
686        :
687        :
688        :
689        :
690        :
691        :
692        :
693        :
694        :
695        :
696        :
697        :
698        :
699        :
700        :
701        :
702        :
703        :
704        :
705        :
706        :
707        :
708        :
709        :
710        :
711        :
712        :
713        :
714        :
715        :
716        :
717        :
718        :
719        :
720        :
721        :
722        :
723        :
724        :
725        :
726        :
727        :
728        :
729        :
730        :
731        :
732        :
733        :
734        :
735        :
736        :
737        :
738        :
739        :
740        :
741        :
742        :
743        :
744        :
745        :
746        :
747        :
748        :
749        :
750        :
751        :
752        :
753        :
754        :
755        :
756        :
757        :
758        :
759        :
760        :
761        :
762        :
763        :
764        :
765        :
766        :
767        :
768        :
769        :
770        :
771        :
772        :
773        :
774        :
775        :
776        :
777        :
778        :
779        :
780        :
781        :
782        :
783        :
784        :
785        :
786        :
787        :
788        :
789        :
790        :
791        :
792        :
793        :
794        :
795        :
796        :
797        :
798        :
799        :
800        :
801        :
802        :
803        :
804        :
805        :
806        :
807        :
808        :
809        :
810        :
811        :
812        :
813        :
814        :
815        :
816        :
817        :
818        :
819        :
820        :
821        :
822        :
823        :
824        :
825        :
826        :
827        :
828        :
829        :
830        :
831        :
832        :
833        :
834        :
835        :
836        :
837        :
838        :
839        :
840        :
841        :
842        :
843        :
844        :
845        :
846        :
847        :
848        :
849        :
850        :
851        :
852        :
853        :
854        :
855        :
856        :
857        :
858        :
859        :
860        :
861        :
862        :
863        :
864        :
865        :
866        :
867        :
868        :
869        :
870        :
871        :
872        :
873        :
874        :
875        :
876        :
877        :
878        :
879        :
880        :
881        :
882        :
883        :
884        :
885        :
886        :
887        :
888        :
889        :
890        :
891        :
892        :
893        :
894        :
895        :
896        :
897        :
898        :
899        :
900        :
901        :
902        :
903        :
904        :
905        :
906        :
907        :
908        :
909        :
910        :
911        :
912        :
913        :
914        :
915        :
916        :
917        :
918        :
919        :
920        :
921        :
922        :
923        :
924        :
925        :
926        :
927        :
928        :
929        :
930        :
931        :
932        :
933        :
934        :
935        :
936        :
937        :
938        :
939        :
940        :
941        :
942        :
943        :
944        :
945        :
946        :
947        :
948        :
949        :
950        :
951        :
952        :
953        :
954        :
955        :
956        :
957        :
958        :
959        :
960        :
961        :
962        :
963        :
964        :
965        :
966        :
967        :
968        :
969        :
970        :
971        :
972        :
973        :
974        :
975        :
976        :
977        :
978        :
979        :
980        :
981        :
982        :
983        :
984        :
985        :
986        :
987        :
988        :
989        :
990        :
991        :
992        :
993        :
994        :
995        :
996        :
997        :
998        :
999        :
1000       :
1001       :
1002       :
1003       :
1004       :
1005       :
1006       :
1007       :
1008       :
1009       :
1010       :
1011       :
1012       :
1013       :
1014       :
1015       :
1016       :
1017       :
1018       :
1019       :
1020       :
1021       :
1022       :
1023       :
1024       :
1025       :
1026       :
1027       :
1028       :
1029       :
1030       :
1031       :
1032       :
1033       :
1034       :
1035       :
1036       :
1037       :
1038       :
1039       :
1040       :
1041       :
1042       :
1043       :
1044       :
1045       :
1046       :
1047       :
1048       :
1049       :
1050       :
1051       :
1052       :
1053       :
1054       :
1055       :
1056       :
1057       :
1058       :
1059       :
1060       :
1061       :
1062       :
1063       :
1064       :
1065       :
1066       :
1067       :
1068       :
1069       :
1070       :
1071       :
1072       :
1073       :
1074       :
1075       :
1076       :
1077       :
1078       :
1079       :
1080       :
1081       :
1082       :
1083       :
1084       :
1085       :
1086       :
1087       :
1088       :
1089       :
1090       :
1091       :
1092       :
1093       :
1094       :
1095       :
1096       :
1097       :
1098       :
1099       :
1100       :
1101       :
1102       :
1103       :
1104       :
1105       :
1106       :
1107       :
1108       :
1109       :
1110       :
1111       :
1112       :
1113       :
1114       :
1115       :
1116       :
1117       :
1118       :
1119       :
1120       :
1121       :
1122       :
1123       :
1124       :
1125       :
1126       :
1127       :
1128       :
1129       :
1130       :
1131       :
1132       :
1133       :
1134       :
1135       :
1136       :
1137       :
1138       :
1139       :
1140       :
1141       :
1142       :
1143       :
1144       :
1145       :
1146       :
1147       :
1148       :
1149       :
1150       :
1151       :
1152       :
1153       :
1154       :
1155       :
1156       :
1157       :
1158       :
1159       :
1160       :
1161       :
1162       :
1163       :
1164       :
1165       :
1166       :
1167       :
1168       :
1169       :
1170       :
1171       :
1172       :
1173       :
1174       :
1175       :
1176       :
1177       :
1178       :
1179       :
1180       :
1181       :
1182       :
1183       :
1184       :
1185       :
1186       :
1187       :
1188       :
1189       :
1190       :
1191       :
1192       :
1193       :
1194       :
1195       :
1196       :
1197       :
1198       :
1199       :
1200       :
1201       :
1202       :
1203       :
1204       :
1205       :
1206       :
1207       :
1208       :
1209       :
1210       :
1211       :
1212       :
1213       :
1214       :
1215       :
1216       :
1217       :
1218       :
1219       :
1220       :
1221       :
1222       :
1223       :
1224       :
1225       :
1226       :
1227       :
1228       :
1229       :
1230       :
1231       :
1232       :
1233       :
1234       :
1235       :
1236       :
1237       :
1238       :
1239       :
1240       :
1241       :
1242       :
1243       :
1244       :
1245       :
1246       :
1247       :
1248       :
1249       :
1250       :
1251       :
1252       :
1253       :
1254       :
1255       :
1256       :
1257       :
1258       :
1259       :
1260       :
1261       :
1262       :
1263       :
1264       :
1265       :
1266       :
1267       :
1268       :
1269       :
1270       :
1271       :
1272       :
1273       :
1274       :
1275       :
1276       :
1277       :
1278       :
1279       :
1280       :
1281       :
1282       :
1283       :
1284       :
1285       :
1286       :
1287       :
1288       :
1289       :
1290       :
1291       :
1292       :
1293       :
1294       :
1295       :
1296       :
1297       :
1298       :
1299       :
1300       :
1301       :
1302       :
1303       :
1304       :
1305       :
1306       :
1307       :
1308       :
1309       :
1310       :
1311       :
1312       :
1313       :
1314       :
1315       :
1316       :
1317       :
1318       :
1319       :
1320       :
1321       :
1322       :
1323       :
1324       :
1325       :
1326       :
1327       :
1328       :
1329       :
1330       :
1331       :
1332       :
1333       :
1334       :
1335       :
1336       :
1337       :
1338       :
1339       :
1340       :
1341       :
1342       :
1343       :
1344       :
1345       :
1346       :
1347       :
1348       :
1349       :
1350       :
1351       :
1352       :
1353       :
1354       :
1355       :
1356       :
1357       :
1358       :
1359       :
1360       :
1361       :
1362       :
1363       :
1364       :
1365       :
1366       :
1367       :
1368       :
1369       :
1370       :
1371       :
1372       :
1373       :
1374       :
1375       :
1376       :
1377       :
1378       :
1379       :
1380       :
1381       :
1382       :
1383       :
1384       :
1385       :
1386       :
1387       :
1388       :
1389       :
1390       :
1391       :
1392       :
1393       :
1394       :
1395       :
1396       :
1397       :
1398       :
1399       :
1400       :
1401       :
1402       :
1403       :
1404       :
1405       :
1406       :
1407       :
1408       :
1409       :
1410       :
1411       :
1412       :
1413       :
1414       :
1415       :
1416       :
1417       :
1418       :
1419       :
1420       :
1421       :
1422       :
1423       :
1424       :
1425       :
1426       :
1427       :
1428       :
1429       :
1430       :
1431       :
1432       :
1433       :
1434       :
1435       :
1436       :
1437       :
1438       :
1439       :
1440       :
1441       :
1442       :
1443       :
1444       :
1445       :
1446       :
1447       :
1448       :
1449       :
1450       :
1451       :
1452       :
1453       :
1454       :
1455       :
1456       :
1457       :
1458       :
1459       :
1460       :
1461       :
1462       :
1463       :
1464       :
1465       :
1466       :
1467       :
1468       :
1469       :
1470       :
1471       :
1472       :
1473       :
1474       :
1475       :
1476       :
1477       :
1478       :
1479       :
1480       :
1481       :
1482       :
1483       :
1484       :
1485       :
1486       :
1487       :
1488       :
1489       :
1490       :
1491       :
1492       :
1493       :
1494       :
1495       :
1496       :
1497       :
1498       :
1499       :
1500       :
1501       :
1502       :
1503       :
1504       :
1505       :
1506       :
1507       :
1508       :
1509       :
1510       :
1511       :
1512       :
1513       :
1514       :
1515       :
1516       :
1517       :
1518       :
1519       :
1520       :
1521       :
1522       :
1523       :
1524       :
1525       :
1526       :
1527       :
1528       :
1529       :
1530       :
1531       :
1532       :
1533       :
1534       :
1535       :
1536       :
1537       :
1538       :
1539       :
1540       :
1541       :
1542       :
1543       :
1544       :
1545       :
1546       :
1547       :
1548       :
1549       :
1550       :
1551       :
1552       :
1553       :
1554       :
1555       :
1556       :
1557       :
1558       :
1559       :
1560       :
1561       :
1562       :
1563       :
1564       :
1565       :
1566       :
1567       :
1568       :
1569       :
1570       :
1571       :
1572       :
1573       :
1574       :
1575       :
1576       :
1577       :
1578       :
1579       :
1580       :
1581       :
1582       :
1583       :
1584       :
1585       :
1586       :
1587       :
1588       :
1589       :
1590       :
1591       :
1592       :
1593       :
1594       :
1595       :
1596       :
1597       :
1598       :
1599       :
1600       :
1601       :
1602       :
1603       :
1604       :
1605       :
1606       :
1607       :
1608       :
1609       :
1610       :
1611       :
1612       :
1613       :
1614       :
1615       :
1616       :
1617       :
1618       :
1619       :
1620       :
1621       :
1622       :
1623       :
1624       :
1625       :
1626       :
1627       :
1628       :
1629       :
1630       :
1631       :
1632       :
1633       :
1634       :
1635       :
1636       :
1637       :
1638       :
1639       :
1640       :
1641       :
1642       :
1643       :
1644       :
1645       :
1646       :
1647       :
1648       :
1649       :
1650       :
1651       :
1652       :
1653       :
1654       :
1655       :
1656       :
1657       :
1658       :
1659       :
1660       :
1661       :
1662       :
1663       :
1664       :
1665       :
1666       :
1667       :
1668       :
1669       :
1670       :
1671       :
1672       :
1673       :
1674       :
1675       :
1676       :
1677       :
1678       :
1679       :
1680       :
1681       :
1682       :
1683       :
1684       :
1685       :
1686       :
1687       :
1688       :
1689       :
1690       :
1691       :
1692       :
1693       :
1694       :
1695       :
1696       :
1697       :
1698       :
1699       :
1700       :
1701       :
1702       :
1703       :
1704       :
1705       :
1706       :
1707       :
1708       :
1709       :
1710       :
1711       :
1712       :
1713       :
1714       :
1715       :
1716       :
1717       :
1718       :
1719       :
1720       :
1721       :
1722       :
1723       :
1724       :
1725       :
1726       :
1727       :
1728       :
1729       :
1730       :
1731       :
1732       :
1733       :
1734       :
1735       :
1736       :
1737       :
1738       :
1739       :
1740       :
1741       :
1742       :
1743       :
1744       :
1745       :
1746       :
1747       :
1748       :
1749       :
1750       :
1751       :

```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 23
DUP DM<->HOST STARTUP OVERLAY

```

1
2
3 003344 104206 001216      :
4 003346 023437             :START3: MOV      #STALK,SP      ;SET UP STACK
5 003347 104202 003024      CALL     SETOVL      ;SET UP OVERLAY POINTERS
6 003351 104223             MOV      #MSGTBL,R2   ;POINT TO MESSAGE TABLE
7 003352 103200 001000 001221 MSGLOP: MOV      (R2)+,R3     ;CONVERSION ROUTINE ADDRESS
8 003355 104221             BIC      #REPEAT,FLAG1 ;CLEAR REPEAT MESSAGE FLAG
9 003356 104010 003246      MOV      (R2)+,R1     ;MESSAGE LENGTH
10 003360 104027            MOV      R1,MLEN      ;SAVE IT
11 003361 060016            MOV      R2,R0        ;POINT TO MESSAGE
12 003362 104207 003247      XFC     MAINTR       ;SEND MESSAGE
13 003364 104201 000051      MOV      #TBUFF,R0   ;ADDRESS OF BUFFER
14 003366 060017            MOV      #TBUFFL,R1  ;LENGTH OF BUFFER
15 003367 104204 003374      XFC     MAINTW       ;GET ANSWER
16 003371             MOV      #RETAD,R4   ;RETURN ADDRESS
17 003372             PUSH     R4          ;SAVE IT
18 003373 000000            PUSH     R3          ;ADDRESS OF CONVERSION ROUTINE
19 003374 102200 000200 001221 RETAD: RETURN ;"CALL" CONVERSION ROUTINE
20 003377 053427            BIT      #QUESDN,FLAG1 ;ALL DONE QUESTIONS ??
21 003400 102200 001000 001221 BNE     CONON        ;YES - SKIP THE REST
22 003403 013412            BIT      #REPEAT,FLAG1 ;REPEAT THE QUESTION ?
23 003404 104207 003234      BEQ     MSGLP4       ;NO - DON'T REPEAT
24 003406 104201 000007      MOV      #INPERR,R0  ;INPUT ERROR MESSAGE
25 003410 060016            MOV      #INPEL,R1   ;LENGTH
26 003411 003414            XFC     MAINTR       ;SEND IT
27 003412 105302 003246      BR      MSGLP2       ;AND REPEAT QUESTION
28 003414 104203 003247      MSGLP4: ADD     MLEN,R2 ;ADD MESSAGE LENGTH
29 003416 104201 000051      MSGLP2: MOV      #TBUFF,R3 ;POINT TO BUFFER
30 003420 114007            MOV      #TBUFFL,R1  ;LENGTH OF BUFFER
31 003421 100237            CLR     R0           ;CLEAP REG
32 003422 117401            MSGLP3: MOV      R0,(R3)+ ;CLEAR BUFFER
33 003423 053421            DEC     R1           ;DECREMENT COUNT
34 003424 104223            BNE     MSGLP3       ;CONTINUE TILL DONE
35 003425 115003            MOV      (R2)+,R3    ;GET CONVERSION ROUTINE ADDRESS
36 003426 053352            TST     R3           ;IF 0 THEN ALL FINISHED
37                               BNE     MSGLOP        ;NOT 0 THEN DO NEXT ENTRY
38                               :
39                               :CONTINUE ON
40 003427 104207 003224      CONON: MOV      #FMTSTA,R0 ;ADDRESS OF INFO MESSAGE
41 003431 104201 000010      MOV      #FMSTL,R1   ;LENGTH OF MESSAGE
42 003433 060016            XFC     MAINTR       ;SEND TO HOST
43 003434 104201 000060      MOV      #G1,R1      ;NEXT OVERLAY
44 003436 022403            CALL     NEXT         ;BRING IT IN

```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 25
 DUP DM<->HOST STARTUP OVERLAY

```

1
2
3
4 003464          :
5 003465 104060 001217  : DO DATE CONVERSION
6 003467 104202 003243  :
7 003471 103200 006000 001221  : DATCON: PUSH R2          :SAVE STRING COUNTER
8 003474 104204 000003          :MOV SP,STCKSV          :SAVE STACK PTR IN CASE OF ERROR
9 003476          :MOV #DATBUF,R2        :POINT TO DATA BUFFER
10 003516 115001          :BIC #GTFLAG+STFLAG,FLAG1 :CLEAR FLAGS
11 003517 013547          :MOV #3,R4             :COUNT TO CONVERT(MONTH,DAY,YEAR)
12 003520 106201 000055          :DATLP: GETB RO,R1      :GET A BYTE FROM RO INTO R1
13 003522 013547          :TST R1                :IS IT A ZERO BYTE ?
14 003523 106201 000057          :BEQ DONONE            :YES - DONE THIS STRING
15 003525 013547          :CMP #DASH,R1          :IS IT A '-'
16 003526          :BEQ DONONE            :YES - DONE THIS STRING
17 003546 003476          :CMP #SLAS,R1          :IS IT A '/'
18 003547          :BEQ DONONE            :YES - DONE THIS STRING
19 003550 104207 003243          :STOB R1,R3,R2        :STORE BYTE IN R1 AT LOCATION IN R2
20 003552 024015          :BR DATLP              :LOOP BACK
21 003553 053625          :DONONE: PUSH RO        :SAVE POINTER TO STRING
22 003554 104073          :MOV #DATBUF,RO        :POINT TO BUFFER TO CONVERT
23 003555          :CALL GENCON           :CONVERT THE STRING
24 003556          :BNE DATAGN            :ILLEGAL CHARS - ASK AGAIN
25 003557 103200 004000 001221  :MOV RO,R3             :GET RESULT
26 003562 104202 003243          :POP RO                :RESTORE POINTER TO STRING
27 003564 104201 000003          :PUSH R3               :STORE RESULT ON STACK
28 003566 114005          :BIC #STFLAG,FLAG1    :SET STORE FLAG TO LOW BYTE
29 003567 100225          :MOV #DATBUF,R2        :POINT TO CONVERT BUFFER
30 003570 117401          :CLR R5                :FOR BUFFER CLEAR
31 003571 053567          :DATLP1: MOV R5,(R2)+   :CLEAR FOR STORE
32 003572 104202 003243          :DEC R1                :CLEAR WORD
33 003574 117404          :BNE DATLP1            :DEC COUNTER
34 003575 053476          :MOV #DATBUF,R2        :CONTINUE TILL DONE
35 003576          :DEC R4                :RESTORE POINTER
36 003577          :BNE DATLP             :DECREMENT PARAMETER COUNTER
37 003600          :POP RO                :CONTINUE TILL DONE
38 003601 023636          :POP R0                :GET YEAR
39 003602 115005          :POP R1                :GET DAY
40 003603 073625          :POP R2                :GET MONTH
41 003604 104203 003623          :CALL DATVER           :VERIFY DATE
42 003606          :TST R5                :ANY ERROR ?
43 003607 104204 000006          :BMI DATAGN            :YUP - PROMPT AGAIN
44 003611 114005          :MOV #DATRET,R3        :RETURN ADDRESS
45 003612          :PUSH R3               :PUSH ON STACK
46 003613 117404          :MOV #6,R4             :COUNT OF ZERO STACK ENTRIES
47 003614 053612          :CLR R5                :CLEAR FOR STORE
48 003615          :DATLP2: PUSH R5        :ZERO ENTRY FOR CONVERSION
49 003616          :DEC R4                :DEC COUNTER
50 003617          :BNE DATLP2            :LOOP TILL DONE
51 003620 104207 001302          :PUSH R1               :STORE DAY
52 003622 004277          :PUSH R2               :STORE MONTH
53 003623          :PUSH R0               :STORE YEAR
54 003624 000000          :MOV #DATE,RO          :POINT TO RESULT BUFFER
55 003625 104306 001217          :JMP VAXTME            :CALL CONVERT TO VAX TIME ROUTINE
56          :DATRET: POP R2        :RESTORE MESSAGE POINTER
57 003627          :DATRT1: RETURN
          :DATAGN: MOV STCKSV,SP :RESTORE STACK PTR - MAY HAVE PUSHED
          :POP R2              :SOME RESULTS ONTO STACK
          :GET MESSAGE POINTER

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 25-1
DUP DM<->HOST STARTUP OVERLAY

58 003630 107202 000002
59 003632 101200 001000 001221
60 003635 003624

SUB #MENTLN,R2
BIS #REPEAT,FLAG1
BR DATRT1

:POINT BACK AT BEGINNING OF QUESTION
:SET TO REPEAT - ANSWER MUST BE VALID
:RETURN

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 27
 DUP DM<->HOST STARTUP OVERLAY

1							
2							
3							
4	003705						
5	003706	024015					
6	003707	053714					
7	003710	104070	000741				
8	003712						
9	003713	000000					
10	003714						
11	003715	107202	000002				
12	003717	101200	001000	001221			
13	003722	003713					

	UNIT NUMBER	HANDLER	
UNITCN:	PUSH	R2	:SAVE MESSAGE POINTER
	CALL	GENCON	:CONVERT TO BINARY
	BNE	UNITBD	:ILLEGAL CHARS
	MOV	RO,UNNO	:MOVE TO UNIT NUMBER
	POP	R2	:RESTORE MESSAGE POINTER
UNITRT:	RETURN		
UNITBD:	POP	R2	:GET MESSAGE POINTER
	SUB	#MENTLN,R2	:POINT BACK AT BEGINNING OF QUESTION
	BIS	#REPEAT,FLAG1	:SET TO REPEAT - ANSWER MUST BE NON-ZERO
	BR	UNITRT	:RETURN

1				:			
2				:			
3				:			
4	003723			SERCON:	PUSH	R2	;SAVE MESSAGE POINTER
5	003724	024015			CALL	GENCON	;CONVERT HIGH ORDER
6	003725	053736			BNE	SERBD	;IF NE THEN ILLEGAL CHARS - PROMPT AGN
7	003726	115007			TST	R0	;IS IT ZERO ?
8	003727	053745			BNE	SEROK	;NO - ALL BITS CAN'T BE ZERO
9	003730	115001			TST	R1	;IS IT ZERO ?
10	003731	053745			BNE	SEROK	;NO - ALL BITS CAN'T BE ZERO
11	003732	115002			TST	R2	;IS IT ZERO ?
12	003733	053745			BNE	SEROK	;NO - ALL BITS CAN'T BE ZERO
13	003734	115003			TST	R3	;IS IT ZERO ?
14	003735	053745			BNE	SEROK	;NO - ALL BITS CAN'T BE ZERO
15	003736			SERBD:	POP	R2	;GET MESSAGE POINTER
16	003737	107202	000002		SUB	#MENTLN,R2	;POINT BACK AT BEGINNING OF QUESTION
17	003741	101200	001000 001221		BIS	#REPEAT,FLAG1	;SET TO REPEAT - ANSWER MUST BE NON-ZERO
18	003744	003761			BR	SERRT	;RETURN
19	003745	104070	001306	SEROK:	MOV	R0,SERNUM	;LOW ORDER WORD
20	003747	104010	001307		MOV	R1,SERNUM+1	;LOW MIDDLE
21	003751	104020	001310		MOV	R2,SERNUM+2	;HIGH MIDDLE
22	003753	104030	001311		MOV	R3,SERNUM+3	;HIGH ORDER
23	003755				POP	R2	;RESTORE MESSAGE POINTER
24	003756	101200	000200 001221		BIS	#QUESDN,FLAG1	;SET QUESTIONS DONE
25	003761	000000		SERRT:	RETURN		;RETURN

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 29
 DUP DM<->HOST STARTUP OVERLAY

```

1
2
3      :
4      :      DOWN LINE LOAD FILE NAME HANDLER
5      :
6      :
7      :
8      :
9      :
10     :
11     :
12     :
13     :
14     :
15     :
16     :
17     :
18     :
19     :
20     :
21     :
22     :
23     :
24     :

```

4	003762				DLLFLE: PUSH	R2		:SAVE MESSAGE POINTER
5	003763	024220			CALL	FIDANS		:GET RESPONSE
6	003764	115001			TST	R1		:TEST RESPONSE
7	003765	014000			BEQ	DLLNO		:NULL - DEFAULT TO BEST GUESS
8	003766	074000			BMI	DLLNO		:NO - DO BEST GUESS
9	003767	101200	000400	001220	BIS	#DLL,FLAG		:YES - DO DOWN-LINE LOAD
10	003772	003776			BR	DLLDN		:EXIT
11	003773	101200	000200	001221	BIS	#QUESDN,FLAG1		:SET END OF QUESTIONS FLAG
12	003776				DLLDN: POP	R2		:RESTORE MESSAGE POINTER
13	003777	000000			DLLDN1: RETURN			
14								
15								
16								
17								
18	004000	101200	002000	001220	DLLNO: BIS	#BSTGS,FLAG		:DO BEST GUESS
19	004003				POP	R2		:GET MESSAGE POINTER
20	004004	105302	003246		ADD	MLEN,R2		:ADD CURRENT MESSAGE LENGTH
21	004006	104621	000001		MOV	#MSGOFF(R2),R1		:GET NEXT MESSAGE LENGTH
22	004010	105202	000002		ADD	#MENTLN,R2		:PAST THE FRONT ENTRIES
23	004012	104010	003246		MOV	R1,MLEN		:MAKE THIS MESSAGE THE NEW LENGTH
24	004014	003777			BR	DLLDN1		:RETURN

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 31
 DUP DM<->HOST STARTUP OVERLAY

1				:			
2				:	MULTIPLY BY 2		
3				:			
4				:			
5				:	RO,R1,R2,R3 = 64-BIT VALUE TO BE MULTIPLIED		
6	004126	105207	000000	MULT2:	ADD #0,R0	:	CLEAR CARRY
7	004130	110207			ROL R0	:	SHIFT FIRST WORD
8	004131	110201			ROL R1	:	SHIFT CARRY AND SECOND WORD
9	004132	110202			ROL R2	:	SHIFT CARRY AND THIRD WORD
10	004133	110203			ROL R3	:	SHIFT CARRY AND FOURTH WORD
11	004134	044136			BCC MULDN	:	IF NO CARRY THEN DONE
12	004135	004137			BR MULERR	:	ELSE ERROR
13	004136	000000		MULDN:	RETURN	:	RETURN
14	004137	104201	000025	MULERR:	MOV #21.,R1	:	RESPONSE ERROR
15	004141	022542			CALL ERRMNT	:	ERROR EXIT

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 32
 DUP DM<->HOST STARTUP OVERLAY

1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11	004142	024126		MULT10.	CALL	MULT2		:GET N * 2
12	004143	104070	007321		MOV	R0,IMAGE		:STORE FIRST WORD
13	004145	104010	007322		MOV	R1,IMAGE+1		:STORE SECOND WORD
14	004147	104020	007323		MOV	R2,IMAGE+2		:STORE THIRD WORD
15	004151	104020	007324		MOV	R2,IMAGE+3		:STORE FOURTH WORD
16	004153	024126			CALL	MULT2		:GET N * 4
17	004154	024126			CALL	MULT2		:GET N * 8
18	004155	105307	007321		ADD	IMAGE,R0		:ADD IN N * 2
19	004157	044161			BCC	10\$:SKIP INC IF CARRY CLEAR
20	004160	115401			INC	R1		:PROP CARRY
21	004161	105301	007322	10\$:	ADD	IMAGE+1,R1		:ADD IN N * 2
22	004163	044165			BCC	20\$:SKIP INC IF CARRY CLEAR
23	004164	115402			INC	R2		:PROP CARRY
24	004165	105302	007323	20\$:	ADD	IMAGE+2,R2		:ADD IN N * 2
25	004167	044171			BCC	30\$:SKIP INC IF CARRY CLEAR
26	004170	115403			INC	R3		:PROP CARRY
27	004171	105303	007324	30\$:	ADD	IMAGE+3,R3		:ADD IN N * 2
28	004173	044175			BCC	40\$:SKIP INC IF CARRY CLEAR
29	004174	004176			BR	MULER1		:ERROR
30	004175	000000		40\$:	RETURN			:ALL DONE
31	004176	104201	000025	MULER1:	MOV	#21.,R1		:RESPONSE ERROR
32	004200	022542			CALL	ERRMNT		:ERROR EXIT
33								

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 33
 DUP DM<->HOST STARTUP OVERLAY

```

1
2
3
4
5
6
7
8
9
10 004201
11 004202 114005
12 004203 104171
13 004204 103201 1,7400
14 004206 014216
15 004207 115405
16 004210 104271
17 004211 103201 000377
18 004213 014216
19 004214 115405
20 004215 004203
21 004216
22 004217 000000
  
```

```

:
:      FIND THE LENGTH OF A STRING
:
:      RO -> STRING
:
:      OUTPUT:
:
:      R5 = COUNT
:
FINLEN: PUSH    RO      ;SAVE RO
        CLR    R5      ;CLEAR COUNTER
FINLN1: MOV    (RO),R1  ;GET WORD
        BIC   #HIBYTE,R1 ;CLEAR HIGH BYTE
        BEQ   FINDON   ;IF ZERO THEN DONE
        INC   R5        ;INCREMENT COUNT
        MOV   (RO)+,R1  ;GET WORD FOR HIGH BYTE
        BIC   #LOBYTE,R1 ;CLEAR LOW BYTE
        BEQ   FINDON   ;IF ZERO THE DONE
        INC   R5        ;INCREMENT COUNT
        BR    FINLN1   ;REPEAT WITH NEXT WORD
FINDON: POP    RO      ;RESTGRE RO
        RETURN
  
```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 34
DUP DM<->HOST STARTUP OVERLAY

```

1
2
3
4
5
6
7
8
9
10
11
12 004220 114001
13 004221 104172
14 004222 103202 177400
15 004224 014233
16 004225 103202 000040
17 004227 106202 000131
18 004231 054234
19 004232 115401
20 004233 00000C
21 004234 117401
22 004235 004233

```

```

:
: DETERMINE IF VALUE IS 'Y',NULL, OR NOT Y
:
: INPUT:
:      RO -> STRING
:
: OUTPUT:
:      R1 = 1 IF 'Y'
:           0 IF NULL
:          -1 IF NOT Y
:
: FIDANS: CLR      R1           ;CLEAR OUTPUT
:         MOV     (R0),R2      ;GET WORD
:         BIC     #HIBYTE,R2   ;CLEAR HIGH STUFF
:         BEQ     FIDNUL       ;IF ZERO THEN IT'S NULL
:         BIC     #BIT5,R2     ;MAKE IT UPPER CASE
:         CMP     #Y,R2       ;IS IT 'Y' ?
:         BNE     NOTY        ;NOPE
:         INC     R1          ;MAKE IT 1
:         RETURN
: FIDNUL: RETURN
: NOTY:  DEC     R1          ;MAKE IT NEGATIVE
:         BR     FIDNUL      ;AND EXIT

```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 37-1
 DUP DM<->HOST STARTUP OVERLAY

```

58 004376 115402          INC      R2          ;PROP CARRY
59                      :
60                      :      INIT QUAD-WORD TO DAYS SINCE BASE DATE
61                      :
62 004377 104074          54$:  MOV      R0,R4          ;QUAD WORD POINTER
63 004400 100243          MOV      R3,(R4)+       ;COPY LOW ORDER DAYS
64 004401 100242          MOV      R2,(R4)+       ;COPY HIGH ORDER DAYS
65 004402 114005          CLR      R5             ;FOR REST OF QUAD-WORD CLEAR
66 004403 100245          MOV      R5,(R4)+       ;CLEAR
67 004404 100145          MOV      R5,(R4)        ;CLEAR
68                      :
69                      :      LOOP TO MERGE TIME OF DAY INTO DAYS SINCE BASE DATE
70                      :
71 004405 104074          60$:  MOV      R0,R4          ;COPY QUAD WORD POINTER
72 004406 104205 000004  MOV      #4,R5          ;INNER LOOP COUNT
73                      :
74                      :      CALCULATE QUAD-WORD <- QUAD-WORD * (R1) + (SP)+
75                      :
76 004410 104142          70$:  MOV      (R4),R2        ;FETCH NEXT WORD OF QUAD-WORD
77 004411 104113          MOV      (R1),R3        ;GET MULTIPLIER
78 004412 104030 000731  MOV      R3,DDUMMY      ;STORE FOR MULTIPLY
79 004414 104020 000736  MOV      R2,TEMP        ;FOR MULTIPLY
80 004416 114000 000732  CLR      DDUMMY+1       ;CLEAR HIGH ORDER
81 004420 114000 000737  CLR      TEMP+1         ;DITTO
82 004422          PUSH     R4             ;SAVE QUAD-WORD POINTER
83 004423 104203 000731  MOV      #DDUMMY,R3     ;FOR MULTIPLY
84 004425 104204 000736  MOV      #TEMP,R4       ;DITTO
85 004427 021537          CALL     DMUL           ;DO MULTIPLY
86 004430 104243          MOV      (R4)+,R3      ;GET LOW ORDER RESULT
87 004431 104142          MOV      (R4),R2       ;GET HIGH ORDER RESULT
88 004432          POP      R4             ;RESTORE QUAD-WORD POINTER
89 004433 115002          TST     R2             ;IS IT POSITIVE
90 004434 034436          BPL     80$            ;O.K.
91 004435 105112          ADD     (R1),R2        ;MAKE IT AN UNSIGNED MULTIPLY
92 004436 105163          80$:  ADD     (SP),R3      ;ADD HIGH ORDER OF PREVIOUS MUL.
93 004437 044441          BCC    81$            ;SKIP INC IF NO CARRY
94 004440 115402          INC     R2             ;ADD CARRY
95 004441 100243          81$:  MOV      R3,(R4)+       ;STORE WORD INTO QUAD-WORD
96 004442 100162          MOV      R2,(SP)       ;SAVE HIGH ORDER WORD
97 004443 117405          DEC     R5             ;DEC COUNT
98 004444 054410          BNE    70$            ;CONTINUE TILL DONE
99 004445 104415          MOV     -(R1),R5       ;"POP" R1
100 004446 014451          BEQ    90$            ;IF ZERO THEN DONE
101 004447 104265          MOV     (SP)+,R5       ;POP STACK
102 004450 004405          BR     60$            ;DO NEXT MULTIPLIER
103 004451 104265          90$:  MOV     (SP)+,R5       ;DO FINAL POP
104 004452 000000          RETURN                ;AND RETURN

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 38
INITIALIZATION OVERLAY (G1)

```

1
2
3
4
5 004453
6 003023 024330
7 003024 102200 000001 001220
8 003027 053033
9 003030 104201 000000
10 003032 003035
11 003033 104201 000003
12 003035 022403
13
14
15
16
17
18
19
20
21 003036 104207 001053
22 003040 104673 000003
23 003042 103203 177400
24 003044 104030 000731
25 003046 114000 000732
26 003050 104673 000002
27 003052 103203 177400
28 003054 104030 000736
29 003056 114000 000737
30 003060 104203 000731
31 003062 104204 000736
32 003064 021537
33 003065 104240 001142
34 003067 104140 001143
35
36
37
38 003071 104673 000011
39 003073 103203 177400
40 003075 104030 000736
41 003077 114000 000737
42 003101 104204 000736
43 003103 104203 001142
44 003105 021537
45 003106 104240 001146
46 003110 104140 001147
47
48
49
50 003112 104673 000004
51 003114 103203 177600
52 003116 104030 000736
53 003120 114000 000737
54 003122 104204 000736
55 003124 104203 001142
56 003126 021537
57 003127 104240 001150

```

.SBTTL INITIALIZATION OVERLAY (G1)

INITIALIZATION OVERLAY

```

DMOVL G1,START
CALL INITL
BIT #FCTAVL,FLAG
BNE DOLBN
MOV #F1,R1
BR DXBN
MOV #F2,R1
CALL NEXT

```

:INITIALIZE DISK
:USE RESIDENT FCT ?
:YES - ONLY DO LBN
:ELSE DO D/XBN FIRST
:SKIP LBN FLAGGING
:SIGNAL LBN FORMAT
:BRING IN NEXT OVERLAY

COMPUTE DISK CONSTANTS FROM CHARACTERISTICS

COMPUTE TRACKS/CYLINDER

```

CONINT: MOV #SCR,R0
MOV TRKGRP(R0),R3
BIC #HIBYTE,R3
MOV R3,DDUMMY
CLR DDUMMY+1
MOV GRPCYL(R0),R3
BIC #HIBYTE,R3
MOV R3,TEMP
CLR TEMP+1
MOV #DDUMMY,R3
MOV #TEMP,R4
CALL DMUL
MOV (R4)+,TRKCYL
MOV (R4),TRKCYL+1

```

:POINT TO SUB CHARACTERISTICS
:LOAD TRACKS/GROUP
:CLEAR HIGH BYTE
:STORE IN DUMMY AREA
:CLEAR FOR STORE
:GET GROUPS/CYLINDER
:CLEAR HIGH BYTE
:STORE IN TEMP AREA
:CLEAR HIGH ORDER
:SET UP FOR MULT
:DITTO
:COMPUTE IT
:LOAD FOR STORE
:LOAD FOR STORE

COMPUTE LBN'S/CYLINDER

```

MOV LBNTRK(R0),R3
BIC #HIBYTE,R3
MOV R3,TEMP
CLR TEMP+1
MOV #TEMP,R4
MOV #TRKCYL,R3
CALL DMUL
MOV (R4)+,LBNPCY
MOV (R4),LBNPCY+1

```

:GET LBN'S/TRACK
:CLEAR HIGH BYTE
:FOR MULT
:FOR STORE
:FOR MULTIPLY
:DITTO
:GET LBN'S/CYL
:GET LOW ORDER
:GET HIGH ORDER

COMPUTE RBN'S/CYLINDER

```

MOV RBNTRK(R0),R3
BIC #HIBYTE,R3
MOV R3,TEMP
CLR TEMP+1
MOV #TEMP,R4
MOV #TRKCYL,R3
CALL DMUL
MOV (R4)+,RBNPCY

```

:GET RBN'S/TRACK
:CLEAR OUT GARBAGE
:STORE FOR MULT
:FOR STORE
:FOR MULTIPLY
:DITTO
:GET RBN'S/CYL
:GET LOW ORDER

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 38-1
INITIALIZATION OVERLAY (G1)

```

58 003131 104140 001151      MOV      (R4),RBNPCY+1      ;GET HIGH ORDER
59
60
61
62
63
64 003133 104207 001053      MOV      #SCR,R0           ;POINT TO CHARACTERISTICS
65 003135 104670 000000 000736  MOV      CYLBN(R0),TEMP     ;GET LBN CYLINDERS
66 003140 104673 000001      MOV      CYLBN+1(R0),R3     ;GET HIGH ORDER
67 003142 103203 170000      BIC      #HD.CLR,R3        ;CLEAR STARTING CYLINDER BITS
68 003144 104030 000737      MOV      R3,TEMP+1         ;STORE IT
69 003146 104204 000736      MOV      #TEMP,R4          ;FOR MULT
70 003150 104203 001146      MOV      #LBNPCY,R3        ;POINT TO LBN'S/CYLINDER
71 003152 021537              CALL     DMUL               ;GET LBN'S IN LBN AREA
72 003153 104140 001134      MOV      (R4),LBNLBN        ;GET LOW ORDER
73 003155 104640 000001 001135  MOV      1(R4),LBNLBN+1     ;GET HIGH ORDER
74 003160 104670 000012 000731  MOV      LBNHOST(R0),DDUMMY ;GET LBN'S IN HOST AREA
75 003163 104670 000013 000732  MOV      LBNHOST+1(R0),DDUMMY+1 ;GET HIGH ORDER
76 003166 104203 000731      MOV      #DDUMMY,R3        ;FOR SUB
77 003170 021521              CALL     DSUB               ;SUBTRACT TO GET LBN'S IN RCT
78 003171 104240 001472      MOV      (R4)+,TOTRCT       ;GET LOW ORDER
79 003173 104140 001473      MOV      (R4),TOTRCT+1     ;GET HIGH ORDER
80
81
82
83
84 003175 104207 001053      MOV      #SCR,R0           ;POINT TO CHARACTERISTICS
85 003177 104670 000000 000736  MOV      CYLBN(R0),TEMP     ;GET LBN CYLINDERS
86 003202 104673 000001      MOV      CYLBN+1(R0),R3     ;GET HIGH ORDER
87 003204 103203 170000      BIC      #HD.CLR,R3        ;CLEAR STARTING CYLINDER BITS
88 003206 104030 000737      MOV      R3,TEMP+1         ;STORE IT
89 003210 104204 000736      MOV      #TEMP,R4          ;FOR MULT
90 003212 104203 001150      MOV      #RBNPCY,R3        ;POINT TO RBN'S/CYLINDER
91 003214 021537              CALL     DMUL               ;GET LBN'S IN LBN AREA
92 003215 104240 001136      MOV      (R4)+,RBNLBN       ;GET LOW ORDER
93 003217 104140 001137      MOV      (R4),RBNLBN+1     ;GET HIGH ORDER
94
95
96
97 003221 104673 000004      MOV      RBNTRK(R0),R3      ;LOAD RBN'S/TRACK
98 003223 103203 177600      BIC      #HI1BYTE,R3       ;CLEAR OUT GARBAGE
99 003225 104674 000011      MOV      LBNTRK(R0),R4      ;LOAD LBN'S/TRACK(512)
100 003227 103204 177400      BIC      #HI1BYTE,R4       ;CLEAR OUT HIGH BYTE
101 003231 105043      ADD      R4,R3              ;ADD FOR SECT/TRACK
102 003232 104030 001130      MOV      R3,SECTRK         ;STORE IT
103 003234 114000 001131      CLR      SECTRK+1          ;CLEAR FOR STORE
104
105
106
107 003236 104300 001130 000731  MOV      SECTRK,DDUMMY       ;LOW ORDER
108 003241 104300 001131 000732  MOV      SECTRK+1,DDUMMY+1   ;HIGH ORDER
109 003244 104204 000731      MOV      #DDUMMY,R4        ;SET UP FOR MULT
110 003246 104203 001142      MOV      #TRKCYL,R3        ;DITTO
111 003250 021537              CALL     DMUL               ;COMPUTE IT
112 003251 104140 001132      MOV      (R4),SECTCY       ;LOAD FOR STORE
113 003253 104640 000001 001133  MOV      1(R4),SECTCY+1     ;LOAD FOR STORE
114

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 38-2
INITIALIZATION OVERLAY (G1)

```

115                                     :      COMPUTE SECTORS IN XBN AREA (R4 -> SECTORS/CYL)
116                                     :
117 003256 104670 000021 000736      MOV      XBNCYL(R0),TEMP      ;LOAD NUMBER OF XBN CYLINDERS
118 003261 114000 000737              CLR      TEMP+1              ;CLEAR FOR STORE
119 003263 104203 000736      MOV      #TEMP,R3           ;SET UP FOR MULT
120 003265 021537              CALL     DMUL                ;GET XBN SECTORS
121 003266 104240 001140      MOV      (R4)+,XBNSEC       ;LOAD FOR STORE
122 003270 104140 001141      MOV      (R4),XBNSEC+1     ;LOAD FOR STORE
123                                     :
124                                     :      COMPUTE LAST CYLINDER NUMBER ON SUBUNIT
125                                     :
126 003272 104207 001053      MOV      #SCR,R0            ;POINT TO CHARACTERISTICS
127 003274 104670 000000 001144      MOV      CYLBN(R0),LBNCYL   ;GET LOW ORDER LBN CYLINDERS
128 003277 104670 000000 000736      MOV      CYLBN(R0),TEMP     ;ALSO FOR MATH
129 003302 104674 000001              MOV      CYLBN+1(R0),R4     ;GET HIGH ORDER
130 003304 104040 001145      MOV      R4,LBNCYL+1       ;STORE HI ORDER
131 003306 104040 000737      MOV      R4,TEMP+1         ;ALSO FOR MATH
132 003310 104204 000736      MOV      #TEMP,R4          ;SET UP FOR ADD
133 003312 104670 000021 000731      MOV      XBNCYL(R0),DDUMMY ;LOAD XBN CYLINDERS
134 003315 114000 000732      CLR      DDUMMY+1          ;CLEAR HIGH ORDER
135 003317 104203 000731      MOV      #DDUMMY,R3        ;SET UP FOR ADD
136 003321 021503              CALL     DADD                ;GET LBN+XBN CYLINDERS
137 003322 104673 000022      MOV      DBNCYL(R0),R3     ;LOAD DBN CYLINDERS
138 003324 110703              SWAB      R3                ;GET INTO LO BYTE
139 003325 103203 177400      BIC      #HIBYTE,R3        ;CLEAR GARBAGE
140 003327 104030 000731      MOV      R3,DDUMMY         ;FOR DIVIDE
141 003331 114000 000732      CLR      DDUMMY+1         ;CLEAR HIGH ORDER
142 003333 104203 000731      MOV      #DDUMMY,R3        ;SET UP FOR ADD
143 003335 021503              CALL     DADD                ;GET LBN+XBN+DBN CYLINDERS
144 003336 104642 000001      MOV      1(R4),R2          ;GET HIGH ORDER
145 003340 104673 000001      MOV      STCYL(R0),R3     ;GET CYLINDER BITS
146 003342 103203 007777      BIC      #LO,R3            ;CLEAR OUT REST OF WORD
147 003344 101032              BIS      R3,R2              ;OR IN CYLINDER HIGH BITS
148 003345 100642 000001      MOV      R2,1(R4)         ;STORE BACK
149 003347 104203 001463      MOV      #ONE,R3          ;DECREMENT TO GET NUM OF LAST CYL
150 003351 021521              CALL     DSUB                ;DO IT
151 003352 104240 001126      MOV      (R4)+,CYLNUM      ;MAKE IT CURRENT CYLINDER
152 003354 104140 001127      MOV      (R4),CYLNUM+1    ;LOAD HI ORDER
153                                     :
154                                     :      COMPUTE NON-PAD RCT
155                                     :
156 003356 104300 001136 000736      MOV      RBNLBN,TEMP       ;GET LOW ORDER LBN'S IN HOST AREA
157 003361 104300 001137 000737      MOV      RBNLBN+1,TEMP+1   ;GET HIGH ORDER
158 003364 104200 000177 000731      MOV      #127,DDUMMY       ;ADD 127 FOR DIV FUNCTION
159 003367 114000 000732      CLR      DDUMMY+1         ;FOR CLEAR
160 003371 104204 000736      MOV      #TEMP,R4          ;FOR ADD
161 003373 104203 000731      MOV      #DDUMMY,R3        ;DITTO
162 003375 021503              CALL     DADD                ;ADD
163 003376 104200 000200 000731      MOV      #128,DDUMMY       ;FOR DIVIDE (128 RBN/RCT BLOCK)
164 003401 114000 000732      CLR      DDUMMY+1         ;FOR STORE
165 003403 104203 000731      MOV      #DDUMMY,R3        ;POINT TO IT
166 003405 021565              CALL     DDIV                ;DO DIVIDE
167 003406 104140 001262      MOV      (R4),RCTIBN      ;GET LOW ORDER QUOTIENT
168 003410 105200 000002 001262      ADD      #2,RCTLBN         ;FOR CONTROL BLOCKS
169                                     :
170                                     :      COMPUTE NON-PAD FCT
171                                     :

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 38-3
INITIALIZATION OVERLAY (G1)

172	003413	104300	001136	000736	MOV	RBNLBN,TEMP	:GET LOW ORDER LBN'S IN HOST AREA
173	003416	104300	001137	000737	MOV	RBNLBN+1,TEMP+1	:GET HIGH ORDER
174	003421	104200	000002	000731	MOV	#2,DDUMMY	:FOR DIVIDE BY 2
175	003424	114000	000732		CLR	DDUMMY+1	:DITTO
176	003426	104204	000736		MOV	#TEMP,R4	:SETUP
177	003430	104203	000731		MOV	#DDUMMY,R3	:SETUP
178	003432	021565			CALL	DDIV	:CALL DIVIDE
179	003433	104200	000177	000731	MOV	#127.,DDUMMY	:ADD 127 FOR DIV FUNCTION
180	003436	114000	000732		CLR	DDUMMY+1	:FOR CLEAR
181	003440	104204	000736		MOV	#TEMP,R4	:FOR ADD
182	003442	104203	000731		MOV	#DDUMMY,R3	:DITTO
183	003444	021503			CALL	DADD	:ADD
184	003445	104200	000200	000731	MOV	#128.,DDUMMY	:FOR DIVIDE (128 RBN/RCT BLOCK)
185	003450	114000	000732		CLR	DDUMMY+1	:FOR STORE
186	003452	104203	000731		MOV	#DDUMMY,R3	:POINT TO IT
187	003454	021565			CALL	DDIV	:DO DIVIDE
188	003455	104200	000002	000731	MOV	#2,DDUMMY	:FOR MULT
189	003460	114000	000732		CLR	DDUMMY+1	:CLEAR HIGH WORD
190	003462	104203	000731		MOV	#DDUMMY,R3	:FOR DIVIDE
191	003464	021537			CALL	DMUL	:DO MULTIPLY
192	003465	104140	001261		MOV	(R4),FCTNPD	:NON-PAD FCT BLOCKS
193	003467	115400	001261		INC	FCTNPD	:FOR NON-PAD FCT BLOCKS
194							
195							
196							
197	003471	104203	005567		MOV	#GDBLK,R3	:POINT TO BUFFER
198	003473	104302	001445		MOV	DWRD,R2	:DIAGNOSTIC WORD
199	003475	100232			MOV	R2,(R3)+	:STORE IT
200	003476	104204	000125		MOV	#85.,R4	:SET COUNTER
201	003500	104302	001442		MOV	FWRD,R2	:FIRST WORD OF PATTERN
202	003502	100232			MOV	R2,(R3)+	:STORE IT
203	003503	104302	001443		MOV	SWRD,R2	:SECOND WORD OF PATTERN
204	003505	100232			MOV	R2,(R3)+	:STORE IT
205	003506	104302	001444		MOV	TWRD,R2	:THIRD WORD OF PATTERN
206	003510	100232			MOV	R2,(R3)+	:STORE IT
207	003511	117404			DEC	R4	:DECREMENT COUNTER
208	003512	053500			BNE	OVER	:REPEAT TILL DONE
209	003513	104302	001446		MOV	EDC,R2	:EDC FOR PATTERN
210	003515	100232			MOV	R2,(R3)+	:STORE IT
211							
212							
213							
214							
215	003516	104203	001130		MOV	#SECTRK,R3	:SEC/TRACK
216	003520	104200	000003	000736	MOV	#3,TEMP	:FOR MULT
217	003523	114000	000737		CLR	TEMP+1	:FOR STORE
218	003525	104204	000736		MOV	#TEMP,R4	:SET UP FOR MULT
219	003527	021537			CALL	DMUL	:GET LENGTH OF IMAGE BLOCK
220	003530	104200	007321	000731	MOV	#IMAGE,DDUMMY	:FOR ADD
221	003533	114000	000732		CLR	DDUMMY+1	:CLEAR HIGH BYTE
222	003535	104203	000731		MOV	#DDUMMY,R3	:SET UP FOR ADD
223	003537	021503			CALL	DADD	:ADD TO GET ADDRESS
224	003540	104140	001232		MOV	(R4),EIMAGE	:GET ADDRESS

OVER:

INITIALIZE GOOD DATA BLOCK

SET UP END OF IMAGE POINTER


```

1
2
3
4 003656 104200 000003 000736      MOV    #IMLEN,TEMP      ;GET LENGTH OF FORMAT IMAGE BLOCK
5 003661 114000 000737              CLR    TEMP+1           ;CLEAR FOR STORE
6 003663 104204 000736              MOV    #TEMP,R4        ;FOR MULT
7 003665 104203 001130              MOV    #SECTRK,R3      ;SECTORS/TRACK
8 003667 021537              CALL   DMUL             ;GET LENGTH OF FORMAT BUFFER TABLE
9 003670 104303 000736              MOV    TEMP,R3         ;GET LENGTH
10 003672 104204 007321             MOV    #IMAGE,R4       ;GET IMAGE BUFFER START ADDRESS
11 003674 105034              ADD    R3,R4           ;GET START ADDRESS OF REVECTOR BUFFER
12 003675 104040 001225             MOV    R4,ERRBUF       ;STORE IT
13 003677 104040 001253             MOV    R4,ERPNT        ;INIT POINTER
14 003701 104200 007775 000736      MOV    #BMAX,TEMP      ;GET MAX BUFFER ADDRESS
15 003704 114000 000737              CLR    TEMP+1           ;FOR CLEAR
16 003706 104040 000731              MOV    R4,DDUMMY       ;STORE BEGINNING ADDRESS
17 003710 114000 000732              CLR    DDUMMY+1        ;CLEAR HIGH WORD
18 003712 104204 000736              MOV    #TEMP,R4        ;POINT TO END ADDRESS
19 003714 104203 000731              MOV    #DDUMMY,R3      ;POINT TO BEGINNING ADDRESS
20 003716 021521              CALL   DSUB             ;SUBTRACT TO GET LENGTH
21 003717 104304 000736              MOV    TEMP,R4         ;GET LENGTH
22 003721 110604              ROR    R4               ;DIVIDE BY 2 (LENGTH OF 1 ENTRY)
23 003722 104040 001226              MOV    R4,EMAX         ;STORE AS MAX NUMBER
24
25
26
27 003724 104207 001040              MOV    #CR,R0          ;POINT TO CHARACTERISTICS
28 003726 104673 000002              MOV    ERV(R0),R3     ;GET RECOVERY LEVELS
29 003730 103203 177400              BIC    #HIBYTE,R3     ;CLEAR HIGH ORDER
30 003732 104030 001500              MOV    R3,RECOV       ;STORE IT
31 003734 104030 001502              MOV    R3,RECTMP      ;INIT COUNTER
32 003736 104673 000001              MOV    RTRY(R0),R3    ;GET RETRY NUMBER
33 003740 110603              ROR    R3               ;SHIFT BY FOUR TO GET INTO LOW ORDER NIBBLE
34 003741 110603              ROR    R3
35 003742 110603              ROR    R3
36 003743 110603              ROR    R3
37 003744 103203 177700              BIC    #HI2BYTE,R3    ;CLEAR HIGH ORDER JUNK
38 003746 115403              INC    R3               ;ONE MORE BECAUSE OF WRONG INC
39 003747 104030 001477              MOV    R3,RETRY       ;STORE IT
40 003751 114000 001501              CLR    TMPTRY          ;FOR STORE
41
42
43
44
45
46 003753 104207 001040              MOV    #CR,R0          ;POINT TO COMMON CHARACTERISTICS
47 003755 104673 000001              MOV    LONGTO(R0),R3  ;GET LONG TIMEOUT IN LOG2
48 003757 103203 177760              BIC    #FCLR,R3       ;CLEAR ALL BUT TIMEOUT
49 003761 053763              BNE    TIMLO1         ;IF NOT ZERO THEN CONTINUE
50 003762 115403              INC    R3               ;MAKE IT AT LEAST 1
51 003763 104201 000001      TIMLO1: MOV    #1,R1        ;INIT COUNTER
52 003765 105201 000000              ADD    #0,R1          ;CLEAR CARRY
53 003767 110201      TIMLOP: ROL    R1         ;SHIFT
54 003770 117403              DEC    R3               ;DECREMENT SHIFT COUNT
55 003771 053767              BNE    TIMLOP         ;CONTINUE TILL DONE
56 003772 104070 000736              MOV    R0,TEMP        ;FOR DIVIDE
57 003774 114000 000737              CLR    TEMP+1         ;CLEAR HIGH WORD

```

58	003776	104200	000012	000731	MOV	#10.,DDUMMY	:FOR DIVIDE BY 10
59	004001	114000	000732		CLR	DDUMMY+1	:CLEAR HIGH WORD
60	004003	104204	000736		MOV	#TEMP,R4	:DIVIDE BY 10 (UDA RECEIVE TIMEOUT)
61	004005	104203	000731		MOV	#DDUMMY,R3	:TO GET VALUE TO USE AS LTO
62	004007	021565			CALL	DDIV	:DO DIVIDE
63	004010	104140	001251		MOV	(R4),LTO	:STORE IT FOR US LATER
64	004012	115400	001251		INC	LTO	:MAKE SURE AT LEAST 1
65							
66							
67							
68							
69							
70	004014	104207	001040		MOV	#CR,R0	:POINT TO COMM CHAR
71	004016	104673	000000		MOV	SHORTO(R0),R3	:GET SHORT TIMEOUT WORD
72	004020	103203	177760		BIC	#FCLR,R3	:CLEAR WORD
73	004022	117403			DEC	R3	:FOR LOG CALCULATION
74	004023	104201	000001		MOV	#1,R1	:INIT COUNTER
75	004025	105201	000000		ADD	#0,R1	:CLEAR CARRY
76	004027	110201			TILOP:	ROL	R1
77	004030	117403			DEC	R3	:ROTATE (MULT BY 2)
78	004031	054027			BNE	TILOP	:DECREMENT COUNTER
79	004032	104203	000012		MOV	#10.,R3	:KEEP GOING TILL DONE
80	004034	110201			TILOP1:	ROL	R1
81	004035	117403			DEC	R3	:SHIFT COUNT FOR MILSEC CONVERSION
82	004036	054034			BNE	TILOP1	:GET NUMBER IN MILSECS
83	004037	104010	001252		MOV	R1,STO	:GO TILL DONE
84							:AGAIN
85							:STORE IT
86							
87							
88							
89							
90	004041	104207	001053		MOV	#SCR,R0	:POINT TO CHARACTERISTICS BLK
91	004043	104670	000005	001322	MOV	DATA(R0),DPREA	:DATA PREAMBLE LENGTH
92	004046	103200	177400	001322	BIC	#HIBYTE,DPREA	:CLEAR OUT HIGH BYTE GARBAGE
93	004051	104673	000005		MOV	HEAD(R0),R3	:HEADER PREAMBLE LENGTH
94	004053	110703			SWAB	R3	:GET INTO LOW BYTE
95	004054	103203	177400		BIC	#HIBYTE,R3	:CLEAR OUT HIGH BYTE GARBAGE
96	004056	104030	001321		MOV	R3,HPREA	:STORE IT
97	004060	114001			CLR	R1	:FOR NO ERROR

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 43
INITIALIZATION OVERLAY (G1)

1
2
3
4
5
6
7
8

004123 104203 001040
004125 104632 000002
004127 103202 177400
004131 100632 000002
004133 000000

⋮
⋮
⋮

CLEAR ECC THRESHOLD

MOV #CR,R3
MOV ERRSYM(R3),R2
BIC #HIBYTE,R2
MOV R2,ERRSYM(R3)
RETURN

:POINT TO CHARACTERISTICS
:GET THE WORD
:CLEAR THE THRESHOLD
:STORE IT BACK

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 44-1
INITIALIZATION OVERLAY (G1)

58	004251	115407			INC	R0		:YUP
59	004252	105071		GSKIP:	ADD	R0,R1		:ADD TO UNIT NUMBER
60	004253	107301	000741		SUB	UNNO,R1		:GET RELATIVE OFFSET FROM GIVEN UNIT NUMBER
61	004255	074307			BMI	NOTHER		:IF NEGATIVE THEN NOT IN THE RANGE
62	004256	104303	000741		MOV	UNNO,R3		:GET DESIRED
63	004260	104641	000000		MOV	UID(R4),R1		:GET ORIGINAL
64	004262	103201	170000		BIC	#HD,CLR,R1		:CLEAR SUBUNIT MASK
65	004264	107013			SUB	R1,R3		:FIGURE OUT WHICH SUBUNIT
66	004265	104202	000001		MOV	#1,R2		:INIT SUBUNIT MASK
67	004267	105202	000000		ADD	#0,R2		:TO CLEAR CARRY
68	004271	110202		SFTRPT:	ROL	R2		:SHIFT LEFT
69	004272	117403			DEC	R3		:DECREMTN COUNTER
70	004273	054271			BNE	SFTRPT		:REPEAT SHIFT
71	004274	105202	000000	GDONE:	ADD	#0,R2		:CLEAR CARRY
72	004276	110202			ROL	R2		:HAVE TO ROTATE
73	004277	110202			ROL	R2		:4 TIMES TO DO A
74	004300	110202			ROL	R2		:SWAP NIBBLE
75	004301	110202			ROL	R2		:FOR SUBUNIT MASK
76	004302	104020	001021		MOV	R2,GSR+1		:STORE MASK IN SUBUNIT CHAR COMMAND
77	004304	101020	001026		BIS	R2,ACC+1		:SET IN FOR CHANGE MODE COMMAND
78	004306	000000			RETURN			:RETURN
79	004307	104302	000740	NOTHER:	MOV	UNIT,R2		:GET CURRENT INTERCONNECT
80	004311	105022			ADD	R2,R2		:NEXT PORT
81	004312	104020	000740		MOV	R2,UNIT		:SAVE BACK
82	004314	114000	000717		CLR	UN,ERI		:FOR RESTORE
83	004316	114000	001476		CLR	COUNT		:CLEAR ERROR COUNT
84	004320	106202	000010		CMP	#8,R2		:ALL DONE ?
85	004322	034143			BPL	GOVER1		:NOPE - TRY THIS INTERCONNECT
86	004323	104652	000002		MOV	2(R5),R2		:UNIT SEARCHING FOR
87	004325	104201	000007		MOV	#7,R1		:SIGNAL NON-EXISTANT UNIT
88	004327	022542			CALL	ERRMNT		:ERROR RETURN

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 45
INITIALIZATION OVERLAY (G1)

1				:		
2				:		
3				:		
4				:		
5	004330	104302	000740	:		
6	004332	022331		INITL:	MOV	UNIT,R2
7	004333	024134			CALL	INITIT
8	004334	022323			CALL	GETUNT
9	004335	021657			CALL	ONLIN
10	004336	022146			CALL	GSTATS
11	004337	023036			CALL	RECAL
12	004340	000000			CALL	CONINT
					RETURN	
						:SELECT UNIT
						:INIT DRIVE
						:GET THE SDI INTERCONNECT
						:BRING IT ONLINE
						:GET STATUS
						:RECALIBRATE
						:COMPUTE DISK CONSTANTS
						:AND STOP

UDAFM ~ UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 46
 DBN/XBN FORMAT OVERLAY (F1)

```

1          .SBTTL  DBN/XBN FORMAT OVERLAY (F1)
2          :
3          :
4          :
5          :
6          :
7          :
8 004341   DMOVLY  F1,START
9
10 003023 104200 000000 001154  MOV  #F1,CUROVL      ;OVERLAY #1
11 003026 101200 000010 001220  BIS  #DBN,FLAG     ;SET DBN FORMAT
12 003031 023070          :      DXFORM          ;FORMAT DBN AREA
13 003032 104303 001312          :      MOV  FCTREV,R3 ;STARTING FCT ENTRY COUNT
14 003034 107303 001240          :      SUB  FCNT,R3   ;TOTAL BAD IN DBN AREA
15 003036 105303 001450          :      ADD  ERRCNT,R3 ;GET FINAL TOTAL
16 003040 104030 001315          :      MOV  R3,DBBAD  ;STORE IT FOR STATS
17 003042 114000 001450          :      CLR  ERRCNT    ;FOR CLEAR
18 003044 104300 001240 001312  MOV  FCNT,FCTREV   ;FOR BAD BLOCK COUNT
19 003047 103200 000010 001220  BIC  #DBN,FLAG     ;DO XBN AREA
20 003052 023070          :      CALL DXFORM    ;FORMAT XBN AREA
21 003053 104303 001312          :      MOV  FCTREV,R3 ;STARTING FCT ENTRY COUNT
22 003055 107303 001240          :      SUB  FCNT,R3   ;TOTAL BAD IN XBN AREA
23 003057 105303 001450          :      ADD  ERRCNT,R3 ;GET FINAL TOTAL
24 003061 104030 001316          :      MOV  R3,XBBAD  ;STORE IT FOR STATS
25 003063 114000 001450          :      CLR  ERRCNT    ;FOR CLEAR
26 003065 104201 000006          :      MOV  #F3,R1   ;FCT DLL OVERLAY
27 003067 022403          :      CALL NEXT     ;BRING IN NEXT OVERLAY
28
29 003070 104207 001053          :      DXFORM: MOV  #SCR,RO ;POINT TO CHARACTERISTICS BLOCK
30 003072 102200 000010 001220  BIT  #DBN,FLAG     ;DO DBN AREA ?
31 003075 013145          :      BEQ  XBNIT     ;NO - DO XBN AREA
32 003076 104673 000022          :      MOV  DBNCYL(RO),R3 ;GET NUMBER OF CYLS TO FM
33 003100 110703          :      SWAB R3       ;GET INTO LOW BYTE
34 003101 103203 177400          :      BIC  #HIBYTE,R3 ;CLEAR HI BYTE
35 003103 104030 001455          :      MOV  R3,CNTCYL ;SET UP COUNTER
36 003105 102200 002000 001220  BIT  #BSTGS,FLAG   ;DOING BEST GUESS ???
37 003110 053140          :      BNE  SKIP4     ;YES - SKIP FCT SET UP
38 003111 104200 000200 001475  MOV  #128,PCNT     ;FOR PBN COUNT INIT
39 003114 114000 001257          :      CLR  FCTCNT    ;FOR INIT FCT READ
40 003116 023341          :      CALL DXFCPG    ;READ IT IN
41 003117 104200 005152 001224  MOV  #PBNBUF,BADPBN ;ADDR OF BAD PBN LIST
42 003122 104300 005170 001240  MOV  PBNBUF+C512,FCNT ;GET COUNT OF USED ENTRIES
43 003125 104300 005170 001312  MOV  PBNBUF+C512,FCTREV ;STORE IT FOR STAT COMPUTATION
44 003130 013135          :      BEQ  SKIP19    ;IF ZERO - THEN NO ENTRIES
45 003131 115400 001257          :      INC  FCTCNT    ;START WITH SECOND FCT BLOCK
46 003133 023341          :      CALL DXFCPG    ;BRING IT IN
47 003134 003140          :      BR   SKIP4     ;SKIP NO ENTRY STUFF
48 003135 101200 000002 001220  SKIP19: BIS  #FCTEMT,FLAG ;SET EMPTY FLAG
49 003140 104200 140000 001457  SKIP4:  MOV  #HD.DBN,HD.CUR ;GET DBN HEADER CODE
50 003143 023355          :      CALL NUMDBN   ;GET DBN OF FIRST BLOCK ON LAST CYLINDER
51 003144 003154          :      BR   SKIP1     ;SKIP XBN SETUP
52 003145 104200 120000 001457  XBNIT:  MOV  #HD.XBN,HD.CUR ;GET XBN HEADER CODE
53 003150 104670 000021 001455  MOV  XBNCYL(RO),CNTCYL ;GET CYLINDERS IN XBN AREA
54 003153 023405          :      CALL NUMXBN   ;GET XBN OF FIRST BLOCK ON LAST XBN CYL
55 003154 104140 001120          :      SKIP1: MOV  (R4),HOLDBN ;LO ORDER FIRST BLOCK NUM TO DO
56 003156 104240 001114          :      MOV  (R4)+,CURBN ;AND MAKE IT CURRENT NUMBER
57 003160 104140 001121          :      MOV  (R4),HOLDBN+1 ;HI ORDER FIRST BLOCK NUM TO DO
    
```

58	003162	104140	001115			MOV	(R4),CURBN+1		:AND MAKE IT CURRENT NUMBER
59	003164	104207	001053			SLEEK: MOV	#SCR,RO		:POINT TO CHARACTERISTICS BLK
60	003166	104670	000002	001461		MOV	GRPCYL(RO),GRPCNT		:LOAD GROUPS/CYL
61	003171	103200	177400	001461		BIC	#HIBYTE,GRPCNT		:CLEAR OUT HIGH GARBAGE
62	003174	104300	001461	001460		MOV	GRPCNT,CURGRP		:GET GROUP NUMBER BY
63	003177	117400	001460			DEC	CURGRP		:DECREMENTING
64	003201	104300	001126	001077		SLEEK2: MOV	CYLNUM,ISEEK+1		:GET LO ORDER WORD OF CYLINDER NUMBER
65	003204	104300	001127	001100		MOV	CYLNUM+1,ISEEK+2		:LOAD HIGH ORDER WORD OF CYL NUM
66	003207	104300	001460	001101		MOV	CURGRP,ISEEK+3		:LOAD GROUP NUMBER
67	003212	022242				CALL	SEEK		:SEEK TO CURRENT CYL NUM
68	003213	115001				TST	R1		:ANY ERRORS ?
69	003214	073334				BMI	SKERR		:YUP - QUIT
70	003215	104207	001053			MOV	SCR,RO		:POINT TO CHARACTERISTICS BLOCK
71	003217	104673	000003			MOV	SKGRP(RO),R3		:LOAD TRACKS/GROUP
72	003221	103203	177400			BIC	#HIBYTE,R3		:CLEAR OUT HIGH GARBAGE
73	003223	104030	001462			MOV	R3,TRKCNT		:STORE IN COUNTER
74	003225	117403				DEC	R3		:TRACK NUMBER IS ONE LESS
75	003226	104030	001113			MOV	R3,CURTRK		:RESET CURRENT TRACK NUMBER
76	003230	104201	000047			SKIP3: MOV	#G7,R1		:FORMAT SETUP OVERLAY
77	003232	022435				CALL	PAGE		:DO IT
78	003233	104304	001322			SKIP7: MOV	DPREA,R4		:DATA PREAMBLE LENGTH
79	003235	104303	001321			MOV	HPREA,R3		:HEADER PREAMBLE LENGTH
80	003237	104307	001320			MOV	IMSTAR,RO		:POINT TO FORMAT IMAGE START POINT
81	003241	104301	001113			MOV	CURTRK,R1		:TRACK TO FORMAT
82	003243	104302	000740			MOV	UNIT,R2		:SDI INTERCONNECT
83	003245	104205	007321			MOV	#IMAGE,R5		:RECIRCULATION POINTER
84	003247	060001				XFC	FORMAT		:FORMAT THE TRACK
85	003250	115001				TST	R1		:ANY ERRORS ?
86	003251	013264				BEQ	SKIP6		:NO - DO CHECK PASS
87	003252	115400	000717			INC	UN.ERI		:INCREMENT IT
88	003254	106300	001477	000717		CMP	RETRY,UN.ERI		:DONE ALL RETRIES ?
89	003257	073330				BMI	FERR		:YUP - ERROR
90	003260	022362				CALL	INITPT		:REINIT
91	003261	022234				CALL	CLEAR		:DRIVE CLEAR
92	003262	022242				CALL	SEEK		:RE-SEEK AND GROUP SELECT
93	003263	003233				BR	SKIP7		:RETRY FORMAT
94	003264	114000	000717			SKIP6: CLR	UN.ERI		:FOR STORE
95	003266	023431				CALL	DXCHEC		:DO CHECK PASS
96	003267	117400	001113			DEC	CURTRK		:DECREMENT IT
97	003271	104204	001120			MOV	#HOLDBN,R4		:PREPARE FOR BEGINNING BLOCK DECREMENT
98	003273	104203	001130			MOV	#SECTR,R3		:DECREMENT BY SECTORS/TRACK
99	003275	021521				CALL	DSUB		:DO DECREMENT
100	003276	104300	001120	001114		MOV	HOLDBN,CURBN		:LO ORDER NEW BLOCK NUMBER
101	003301	104300	001121	001115		MOV	HOLDBN+1,CURBN+1		:HI ORDER NEW BLOCK NUMBER
102	003304	117400	001462			DEC	TRKCNT		:DECREMENT IT
103	003306	053230				BNE	SKIP3		:NO - DO NEXT TRACK
104	003307	117400	001460			DEC	CURGRP		:DECREMENT GROUP NUMBER
105	003311	117400	001461			DEC	GRPCNT		:DECREMENT IT
106	003313	053201				BNE	SLEEK2		:NO - DO NEXT GROUP
107	003314	117400	001455			DEC	CNTCYL		:DECREMENT IT
108	003316	060022				XFC	UPDATE		:UPDATE PROGRESS INDICATOR
109	003317	104204	001126			MOV	#CYLNUM,R4		:PREPARE FOR CYL NUM DECREMENT
110	003321	104203	001463			MOV	#ONE,R3		:DECREMENT BY ONE
111	003323	021521				CALL	DSUB		:DO SUBTRACT
112	003324	115000	001455			TST	CNTCYL		:ARE WE DONE ?
113	003326	053164				BNE	SLEEK		:DONE ? NO - DO NEXT CYLINDER
114	003327	000000				RETURN			:YES - ALL DONE

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 46-2
DBN/XBN FORMAT OVERLAY (F1)

115 003330 104012
116 003331 104201 000010
117 003333 003340
118 003334 104302 001126
119 003336 104201 000012
120 003340 022542

FERR: MOV R1,R2
MOV #8.,R1
BR DXERR
SKERR: MOV CYLNUM,R2
MOV #10.,R1
DXERR: CALL ERRMNT

;GET XFC FAILURE CODE
;SET FORMAT ERROR
;CYLINDER WHICH FAILED ON
;SEEK ERROR
;ERROR RETURN

UDAFM - UDA FORMATTER DMACR X0 .01 23-AUG-82 14:02:32 PAGE 47
 DBN/XBN FORMAT OVERLAY (F1)

```

1
2
3
4
5 003341
6 003342 104201 000033
7 003344 022435
8 003345 104200 000200 001476
9 003350 104200 005152 001224
10 003353
11 003354 000000
12
13
14
15
16
17 003355 104673 000022
18 003357 110703
19 003360 103203 177400
20 003362 104030 000736
21 003364 114000 000737
22 003366 104204 000736
23 003370 104203 001132
24 003372 021537
25 003373 104641 000001
26 003375 105301 001326
27 003377 100641 000001
28 003401 104203 001130
29 003403 021521
30 003404 000000
31
32
33
34
35
36
37 003405 104670 000021 000736
38 003410 114000 000737
39 003412 104204 000736
40 003414 104203 001132
41 003416 021537
42 003417 104641 000001
43 003421 105301 001325
44 003423 100641 000001
45 003425 104203 001130
46 003427 021521
47 003430 000000
48
49
50
51
52
53 003431
54 003432 114000 001227
55 003434 114000 001222
56 003436 102200 000600 001220
57 003441 013454

```

:PAGE IN NEW FCT BLOCK
 DXFCPG: PUSH R0
 MOV #G2,R1 :DLL OVERLAY
 CALL PAGE :EXECUTE OVERLAY
 MOV #128.,COUNT :FOR INIT
 MOV #PBNBUF,BADPBN :FOR POINTER RESET
 POP R0 :RESTORE R0
 RETURN :RETURN

:
 COMPUTE NUMBER OF FIRST DBN ON LAST DBN TRACK
 R0 -> CHARACTERISTICS BLOCK

NUMDBN: MOV DBNCYL(R0),R3 :GET NUMBER OF CYLINDERS IN DBN AREA
 SWAB R3 :GET INTO LOW BYTE
 BIC #HIBYTE,R3 :CLEAR OUT OTHER INFO
 MOV R3,TEMP :MOVE TO TEMP AREA
 CLR TEMP+1 :CLEAR FOR STORE
 MOV #TEMP,R4 :POINT R4 AT TEMP AREA
 MOV #SECTCY,R3 :POINT TO NUM OF SECTORS/CYLINDER
 CALL DMUL :MULTIPLY TO GET SECTORS BEFORE LAST CYL
 MOV 1(R4),R1 :GET HIGH ORDER
 ADD ST.DBN,R1 :ADD HIGH ORDER STARTING DBN
 MOV R1,1(R4) :STORE BACK
 MOV #SECTRK,R3 :WANT FIRST DN OF LAST TRACK
 CALL DSUB :SUB TO GET IT
 RETURN

:
 COMPUTE NUMBER OF FIRST XBN ON LAST XBN TRACK
 R0 -> CHARACTERISTICS BLOCK

NUMXBN: MOV XBNCYL(R0),TEMP :GET NUMBER OF CYLINDERS IN XBN AREA
 CLR TEMP+1 :CLEAR FOR STORE
 MOV #TEMP,R4 :POINT TO TEMP AREA
 MOV #SECTCY,R3 :POINT TO SECTORS/CYLINDER
 CALL DMUL :MULTIPLY TO GET SECTORS IN XBN AREA
 MOV 1(R4),R1 :GET HIGH ORDER
 ADD ST.XBN,R1 :ADD HIGH ORDER STARTING XBN
 MOV R1,1(R4) :STORE BACK
 MOV #SECTRK,R3 :WANT XBN OF LAST TRACK
 CALL DSUB :SUB TO GET IT
 RETURN

:
 CHECK PASS

DXCHEC: PUSH R0 :SAVE PTR TO CHARACTERISTICS BLK
 DXCH: CLR ERR :FOR ERROR COUNT RESET
 CLR ERFLAG :CLEAR RE-FORMAT FLAG
 BIT #MANU+DLL,FLAG :FCT AVAILABLE ?
 BEQ CSKIP :NO - DO EXTENSIVE READ

UDAFM - UDA FORMATTER DMACR X04.01 23-AJG-82 14:02:32 PAGE 47-1
DBN/XBN FORMAT OVERLAY (F1)

58	003442	104200	000001	001452	MOV	#1,N	:SFT UP FOR STORE
59	003445	104200	000005	001453	MOV	#5,N1	:SET UP
60	003450	104300	001453	001454	MOV	N1,NN1	:SAVE FOR LATER RESET
61	003453	003465			BR	CSKIP2	:SKIP EXTENSIVE READ SETUP
62	003454	104200	000003	001452	CSKIP: MOV	#3,N	:EXTENSIVE REGULAR READ
63	003457	104200	000024	001453	MOV	#20,N1	:EXTENSIVE ERROR READS
64	003462	104300	001453	001454	MOV	N1,NN1	:SAVE FOR LATER RESET
65	003465	024001			CSKIP2: CALL	FIXIT	:DO IT
66	003466	104302	000740		CSKIP1: MOV	UNIT,R2	:SDI INTERCONNECT
67	003470	060012			XFC	SIP	:SYNCH WITH SECTOR/INDEX PULSE
68	003471	104300	001130	001451	MGV	SECTRY,SECCNT	:LOAD SECTORS/TRACK
69	003474	104205	006621		MOV	#CMDBUF,R5	:POINT TO COMMAND BUFFER
70	003476	104207	000721		AGAIN: MOV	#RDBLK,R0	:POINT TO READ COMMAND BLOCK
71	003500	104653	000002		MOV	RB,CMD(R5),R3	:ZERO COMMAND ?
72	003502	013565			BEQ	NOERR	:YES - SKIP CHECKS
73	003503	106673	000004		MOV	R3,RW,CMD(R0)	:ELSE STORE IT
74	003505	104653	000000		MOV	RB,LOW(R5),R3	:GET LOW ORDER BLOCK NUMBER
75	003507	100673	000002		MOV	R3,RW,LOW(R0)	:STORE IN COMMAND BLOCK
76	003511	104653	000001		MOV	RB,HI(R5),R3	:LOAD HIGH ORDER BLOCK NUMBER
77	003513	100673	000003		MOV	R3,RW,HI(R0)	:STORE IN COMMAND BLOCK
78	003515	104203	004535		MOV	#RDBUF,R3	:GET BUFFER POINTER
79	003517	100673	000001		MOV	R3,RW,BUF(R0)	:STORE IN COMMAND BLOCK
80	003521	104203	000726		MOV	#HSLIM-1,R3	:POINTER TO DUMMY SDI BLOCK
81	003523	100673	000005		MOV	R3,RW,DUM(R0)	:STORE IT IN READ BLOCK
82	003525	104207	100721		READ1: MOV	#<BIT15!RDBLK>,R0	:MAKE SURE POINTING AT BLOCK
83	003527	104203	100000		MOV	#RDCMD,R3	:RESET STATUS POINTER
84	003531	100673	000000		MOV	R3,RW,STAT(R0)	:STORE IT BACK
85	003533	060002			XFC	READ	:READ 1 SECTOR
86	003534	115001			TST	R1	:ANY ERRORS ?
87	003535	053550			BNE	RRERR	:YES - UM OH
88	003536	104307	000721		MOV	RDBLK,R0	:GET STATUS WORD
89	003540	102207	010000		BIT	#ECCF,R0	:ANY ECC ERROR ?
90	003542	053550			BNE	RRERR	:YUP - MARK AS BAD FOR NOW
91	003543	104207	001440		MOV	#NUM,R0	:POINT TO COMPARE BLOCK
92	003545	060006			XFC	CMPCAT	:DO DATA COMPARE
93	003546	115001			TST	R1	:ANY ERROR IN COMPARE ?
94	003547	013565			BEQ	NOERR	:NOPC - CONTINUE LOOP
95	003550				RRERR:		
96	003550	104653	000003		MOV	RB,IM(R5),R3	:GET POINTER TO IMAGE
97	003552	104134			MOV	(R3),R4	:GET BUFFER POINTER WORD
98	003553	102204	020000		BIT	#BD,R4	:ALREADY MARKED BAD ??
99	003555	053565			BNE	NOERR	:YUP - DON'T COUNT AGAIN
100	003556	101204	020000		BIS	#BD,R4	:FLAG AS BAD
101	003560	100134			MOV	R4,(R3)	:STORE BACK
102	003561	115400	001227		INC	ERR	:INCREMENT ERROR COUNT
103	003563	115400	001470		INC	RTYCNT	:INC COUNTER
104	003565	105205	000004		NOERR?: ADD	#RDLEN,R5	:POINT TO NEXT READ CMD BLOCK
105	003567	117400	001451		DEC	SECCNT	:DECREMENT COUNTER
106	003571	053476			BNE	AGAIN	:NO - DO NEXT SECTOR
107	003572	117400	001452		DEC	N	:DECREMENT COUNTER
108	003574	053466			BNE	CSKIP1	:NO - REPEAT TRACK READ AND COMPARE
109	003575	115000	001227		TST	ERR	:ANY ERRORS ?
110	003577	013777			BEQ	CDONE	:NO - ALL DONE CHECK PASS
111	003600	104204	007321		MOV	#IMAGE,R4	:POINT TO IMAGE BUFFER
112	003602	104143			HERE: MOV	(R4),R3	:GET BUFFER POINTER WORD
113	003603	102203	020000		BIT	#BD,R3	:IS IT BAD
114	003605	013733			BEQ	CSKIP7	:NO - SKIP IT

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 47-2
 DBN/XBN FORMAT OVERLAY (F1)

115	003605	104643	000002		MOV	FT.HI(R4),R3	:GET HIGH ORDER	
116	003610	103203	007777		BIC	#LO,R3	:CLEAR OUT ALL BUT HEADER	
117	003612	106303	001457		CMF	HD.CUR,R3	:IS IT A "GOOD X/D BN" ?	
118	003614	053733			BNE	CSKIP7	:NOPE - ALREADY REVECTORED IT	
119	003615	104302	000740		MOV	UNIT,R2	:SDI INTERCONNECT	
120	003617	060012			XFC	SIP	:WAIT FOR PULSE	
121	003620	104207	000721		MOV	#RDBLK,R0	:PREPARE FOR READ SECTORS	
122	003622	104203	000726		MOV	#HSLIM-1,R3	:POINTER TO DUMMY SDI BLOCK	
123	003624	100673	000005		MOV	R3,RW.DUM(R0)	:STORE IN COMMAND BLOCK	
124	003626	104643	000001		MOV	1(R4),R3	:LO ORDER BLOCK NUMBER	
125	003630	100673	000002		MOV	R3,RW.LOW(R0)	:STORE IN READ CMD BLOCK	
126	003632	104643	000002		MOV	2(R4),R3	:HI ORDER BLOCK NUM AND CODE	
127	003634	100673	000003		MOV	R3,RW.HI(R0)	:STORE IN READ CMD BLOCK	
128	003636	104203	004535		MOV	#RDBUF,R3	:LOAD ADDRESS OF DATA BUFFER	
129	003640	100673	000001		MOV	R3,RW.BUF(R0)	:STORE IN COMMAND BUFFER	
130	003642	104203	013400		MOV	#RWCMD,R3	:LOAD SDI READ COMMAND	
131	003644	104301	001113		MOV	CURTRK,R1	:GET CURRENT HEAD NUMBER IN R1	
132	003646	101013			BIS	R1,R3	:SET IT IN COMMAND	
133	003647	100673	000004		MOV	R3,RW.CMD(R0)	:STORE BACK	
134	003651	104207	100721	READ2:	MOV	#<BIT15!RDBLK>,R0	:MAKE SURE POINTING AT BLOCK	
135	003653	104203	100000		MOV	#RDCMD,R3	:MARK AS ONLY REQUEST	
136	003655	100173			MOV	R3,(R0)	:STORE IN CMD BLOCK	
137	003656	104302	000740		MOV	UNIT,R2	:SDI INTERCONNECT	
138	003660	060002			XFC	READ	:READ 1 SECTOR	
139	003661	115001			TST	R1	:ANY ERROR IN READ ?	
140	003662	053706			BNE	ER1	:YES - CONSIDER BAD	
141	003663	104173			MOV	(R0),R3	:LOAD ECC ERROR INDICATOR FOR TEST	
142	003664	102203	010000		BIT	#ECCF,R3	:ERROR ?	
143	003666	013672			BEQ	CSKIP6	:NO - CHECK EDC	
144	003667	023000			CALL	ECCCK	:ELSE FIND HOW MANY SYMBGLS IN ERROR	
145	003670	115001			TST	R1	:WITHIN BOUNDS ?	
146	003671	073706			BMI	ER1	:NOPE - CONSIDER BAD	
147	003672	106300	001445	004535	CSKIP6:	CMF	DWRD,RDBUF	:FIRST WORD O.K. ??
148	003675	053706			BNE	ER1	:NOPE - BARF	
149	003676	104202	004535		MOV	#RDBUF,R2	:POINT TO BUFFER	
150	003700	022600			CALL	CEDC	:COMPUTE EDC - RETURNED IN R3	
151	003701	104205	004535		MOV	#RDBUF,R5	:POINT TO BUFFER	
152	003703	106653	000400		CMF	RW.EDC(R5),R3	:EDC O.K. ?	
153	003705	013723			BEQ	OK	:NO ERROR	
154	003706			ER1:				
155	003706	104643	000002		MOV	FT.HI(R4),R3	:GET HI ORDER BLOCK NUM AND HDR CODE	
156	003710	103203	170000		BIC	#HD.CLR,R3	:CLEAR THE HEADER	
157	003712	101203	110000		BIS	#HD.BAD,R3	:MARK AS BAD	
158	003714	100643	000002		MOV	R3,FT.HI(R4)	:STORE BACK IN IMAGE	
159	003716	115400	001222		INC	ERFLAG	:SET RE-FORMAT FLAG	
160	003720	115400	001450		INC	ERRCNT	:UP COUNTER OF BAD BLOCKS	
161	003722	003726			BR	CSKIP3	:NO NEED TO RE-READ ANY MORE THIS SECTOR	
162	003723	117400	001453	OK:	DEC	N1	:DECREMENT COUNTER - DONE ?	
163	003725	053602			BNE	HERE	:NO - RE-READ SECTOR IN ERROR	
164	003726	104300	001454	001453	CSKIP3:	MOV	NN1,N1	:GET SAVED VALUE
165	003731	117400	001227		DEC	ERR	:DECREMENT IT	
166	003733	105204	000003	CSKIP7:	ADD	#IMLEN,R4	:POINT TO NEXT ERROR ENTRY	
167	003735	115000	001227		TST	ERR	:ALL DONE ERROR SECTORS	
168	003737	053602			BNE	HERE	:NO - DO NEXT SECTOR	
169	003740	115000	001222		TST	ERFLAG	:WERE THERE ANY BAD SECTORS FOUND	
170	003742	013777			BEQ	CDONE	:NOPE - ALL DONE	
171	003743	104304	001322	OVER2:	MOV	DPREA,R4	:DATA PREAMBLE LENGTH	


```

1
2
3
4
5
6 004001 104300 001130 001451 FIXIT: MOV SECTRK,SECCNT ;INIT COUNTER
7 004004 104207 006621 MOV #CMDBUF,R0 ;COMMAND BUFFER
8 004006 104205 007321 MOV #IMAGE,R5 ;POINT TO TRACK IMAGE
9 004010 104303 001234 MOV SKPCNT,R3 ;GET STARTING OFFSET(TUNED)
10 004012 105035 ADD R3,R5 ;POINT TO FIRST ENTRY
11 004013 104050 001233 MOV R5,STARIT ;MARK STARTING ADDRESS
12 004015 104653 000002 MORE: MOV 2(R5),R3 ;SET UP FOR HSR CODE COMPARE
13 004017 103203 007777 BIC #LO,R3 ;ISOLATE HI 4 BITS(HDR CODE)
14 004021 106203 120000 CMP #HD.XBN,R3 ;GOOD XBN ?
15 004023 014035 BEQ FKIP2 ;YES - MARK AS GOOD TO CHECK
16 004024 106203 140000 CMP #HD.DBN,R3 ;GOOD DBN ?
17 004026 014035 BEQ FKIP2 ;YES - MARK AS GOOD TO CHECK
18 004027 114003 CLR R3 ;CLEAR FOR STORE
19 004030 100673 000002 MOV R3,RB.CMD(R0) ;STORE AS BAD SECTOR FLAG
20 004032 105207 000004 ADD #RDLEN,R0 ;POINT PAST BLOCK
21 004034 004051 BR FKIP1 ;SKIP GOOD MARK
22 004035 104653 000001 FKIP2: MOV 1(R5),R3 ;LO ORDER BLOCK NUMBER
23 004037 100273 MOV R3,(R0)+ ;STORE IN READ CMD BLOCK
24 004040 104653 000002 MOV 2(R5),R3 ;HI ORDER BLOCK NUM AND CODE
25 004042 100273 MOV R3,(R0)+ ;STORE IN READ CMD BLOCK
26 004043 104203 013400 MOV #RWCMD,R3 ;LOAD SDI READ COMMAND
27 004045 101303 001113 BIS CURTRK,R3 ;SET IN CURRENT TRACK NUMBER
28 004047 100273 MOV R3,(R0)+ ;STORE IN BLOCK
29 004050 100275 MOV R5,(R0)+ ;SAVE PTR TO IMAGE BLK ENTRY
30 004051 105305 001235 FKIP1: ADD TBLK,R5 ;ADD TO GET NEXT SECTOR
31 004053 106305 001232 CMP EIMAGE,R5 ;SEE IF HAVE TO LOOP BACK TO TOP
32 004055 014064 BEQ REDO ;NEED TO RESET
33 004056 034066 BPL FKP1 ;NO NEED - JUST CONTINUE
34 004057 107305 001232 SUB EIMAGE,R5 ;SUBTRACT TO GET LOOP AMOUNT
35 004061 105205 007321 ADD #IMAGE,R5 ;AND ADD OFFSET
36 004063 004066 BR FKP1 ;SKIP ZERO CONDITION
37 004064 104205 007321 REDO: MOV #IMAGE,R5 ;IF ZERO SIMPLY MOVE TO FRONT
38 004066 106305 001233 FKP1: CMP STARIT,R5 ;AT BEGINNING ADDRESS ?
39 004070 054075 BNE FKIP10 ;NO - JUST CONTINUE
40 004071 105205 000003 ADD #IMLEN,R5 ;ELSE POINT TO NEXT ENTRY
41 004073 104050 001233 MOV R5,STARIT ;MAKE IT NEW STARTING ADDRESS
42 004075 117400 001451 FKIP10: DEC SECCNT ;DECREMENT
43 004077 054015 BNE MORE ;NO - DO NEXT SECTOR
44 004100 000000 RETURN
    
```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 49
 DBN/XBN TRACK FORMAT OVERLAY (G7)

```

1          .SBTTL  DBN/XBN TRACK FORMAT OVERLAY (G7)
2
3          :
4          :
5          SET UP TRACK FORMAT
6          :
7          :
8          :
9          :
10         DMOVLY  G7, START
11         MOV     #G7, CUROVL          :GET OVERLAY INDICATOR
12         DXTRK: MOV     SECTRK, SECCNT :MOVE SECTOR COUNT INTO R3
13         MOV     #IMAGE, R5          :POINT TO FORMAT TRACK IMAGE
14         BIT     #MANU+DLL, FLAG     :SEE IF FCT AVAILBLF
15         BEQ     TKIP2                :NO - SKIP PBN COMPUTATION
16         BIT     #DBN, FLAG          :DO DBN AREA ??
17         BEQ     TKIP1                :NO - DO XBN AREA
18         CALL    DPBN                :COMPUTE PBN FOR STARTING DBN
19         BR      TKIP2                :SKIP XBN COMPUTATION
20         TKIP1: CALL    XPBN          :COMPUTE PBN FOR STARTING XBN
21         TKIP2: MOV     #GDBLK, R3    :POINT R3 AT GOOD DATA BLOCK
22         MOV     R3, (R5)+           :AND STORE PTR IN IMAGE BLOCK
23         MOV     CURBN, R3           :GET LOW ORDER BLOCK NUMBER
24         MOV     R3, (R5)+           :AND STORE IN IMAGE BLOCK
25         MOV     CURBN+1, R3         :HI ORDER BLOCK NUM AND HDR CODE
26         BIC     #HD.CLR, R3         :CLEAR HEADER CODE
27         MOV     HD.CUR, R1          :GET CURRENT HEADER CODE (XBN OR DBN)
28         BIS     R1, R3              :SET TO GOOD HEADER CODE
29         MOV     R3, (R5)+           :AND STORE IN IMAGE BLOCK
30         DUBINC  CURBN               :INCREMENT IT
31         SKIP5: DEC     SECCNT        :DECREMENT IT
32         BNE     TKIP2                :NO - DO NEXT SECTOR
33         MOV     #SCR, R3             :POINT TO CHARACTERISTICS
34         MOV     OFFS(R3), R2         :GET GROUP OFFSET
35         SWAB    R2                  :GET INTO LOWBYTE
36         BIC     #HIBYTE, R2         :CLEAR HIGH GARBAGE
37         MOV     #IMAGE, R4          :POINT TO IMAGE
38         TST     R2                  :ANY OFFSET ?
39         BEQ     TKIP5                :NO - SKIP CALCULATIONS
40         TST     CURGRP              :IS GROUP ZERO ???
41         BEQ     TKIP5                :YES - NO OFFSET
42         MOV     R2, TEMP            :STORE IT
43         CLR     TEMP+1              :FOR STORE
44         MOV     CURGRP, DDUMMY      :GET CURRENT GROUP
45         CLR     DDUMMY+1            :CLEAR HIGH WORD
46         MOV     #TEMP, R3           :FOR MUL
47         MOV     #DDUMMY, R4         :DITTO
48         CALL    DMUL                :MULTIPLY TO GET OFFSET FOR THIS GROUP
49         TKIP8: CMP     SECTRK, DDUMMY :IS TOTAL OFFSET MORE THAN NUMBER OF SECTORS ?
50         BPL     TKIP9                :NO - ALL IS FINE
51         SUB     SECTRK, DDUMMY      :YES - SUBTRACT TILL IT IS
52         BR      TKIP8                :CHECK AGAIN
53         TKIP9: MOV     #IMLEN, TEMP  :GET LENGTH OF IMAGE BLOCK
54         CLR     TEMP+1              :FOR STORE
55         MOV     #TEMP, R3           :FOR MULT
56         CALL    DMUL                :GET LENGTH TO OFFSET
57         MOV     (R4), R3            :GET RESULT
    
```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 49-1
 DBN/XBM TRACK FORMAT OVERLAY (G7)

58	003151	104304	001232		MOV	EIMAGE,R4		;GET ADDRESS OF END OF IMAGE
59	003153	107034			SUB	R3,R4		;SUBTRACT TO GET STARTING LOCATION
60	003154	104040	001320		TKIP5: MOV	R4,IMSTAR		;STORE IT
61	003156	102200	002002	001220	BIT	#BSTGS+FCTEMT,FLAG		;IS FCT AVAILABLE AND NON-EMPTY ?
62	003161	053252			BNE	TKIP3		;NO - CONSIDER GOOD
63	003162	104300	001130	001451	MOV	SECTRK,SECCNT		;RESET SECTOR COUNT
64	003165	104302	001224		TKIP12: MOV	BADPBN,R2		;POINT TO PBN
65	003167	104121			MOV	(R2),R1		;GET LOW ORDER PBN
66	003170	106010	001111		CMP	R1,CURPBN		;ARE THEY EQUAL ?
67	003172	053233			BNE	TKIP4		;NO - SKIP REST OF COMPARE
68	003173	104621	000001		MOV	1(R2),R1		;GET HIGH ORDER BAD
69	003175	103201	170000		BIC	#HD.CLR,R1		;CLEAR HEADER FOR COMPARE
70	003177	106010	001112		CMP	R1,CURPBN+1		;EQUAL ?
71	003201	053233			BNE	TKIP4		;NO - MARK AS GOOD
72	003202	117400	001476		DEC	COUNT		;DECREMENT IT
73	003204	117400	001240		DEC	FCTNT		;DEC IT
74	003206	053212			BNE	TKIP7		;IF NOT EMPTY THEN CONTINUE
75	003207	101200	000002	001220	BIS	#FCTEMT,FLAG		;SET FCT EMPTY FLAG
76	003212	104643	000002		TKIP7: MOV	FT.HI(R4),R3		;HI ORDER BLOCK NUM AND HDR CODE
77	003214	103203	170000		BIC	#HD.CLR,R3		;CLEAR THE HEADER CODE
78	003216	101203	110000		BIS	#HD.BAD,R3		;SET TO BAD HEADER CODE
79	003220	100643	000002		MOV	R3,FT.HI(R4)		;AND STORE IN IMAGE BLOCK
80	003222	105200	000002	001224	ADD	#2,BADPBN		;MOVE PTR TO NEXT BAD BLOCK
81	003225	115000	001476		TST	COUNT		;DONE WITH THIS FCT BLOCK ?
82	003227	053233			BNE	TKIP4		;IF NOT DONE SKIP
83	003230	115400	001257		INC	FCTCNT		;GET NEXT BLOCK NUMBER
84	003232	023360			CALL	DXFCP1		;ELSE PAGE IN NEW FCT BLOCK
85	003233	105204	000003		TKIP4: ADD	#IMLEN,R4		;POINT TO NEXT IMAGE ENTRY
86	003235	106304	001232		CMP	EIMAGE,R4		;AT THE END ?
87	003237	053242			BNE	TKIP11		;NOPE - CARRY ON
88	003240	104204	007321		MOV	#IMAGE,R4		;POINT TO START
89	003242				TKIP11: DUBINC	CURPBN		;INCREMENT CURRENT PBN COUNTER
90	003247	117400	001451		DEC	SECCNT		;DECREMENT SECTOR COUNTER
91	003251	053165			BNE	TKIP12		;CONTINUE TILL DONE ALL SECTORS
92								
93	003252	104304	001320		TKIP3: MOV	IMSTAR,R4		;POINT TO FIRST TO FORMAT ENTRY
94	003254	104303	001232		MOV	EIMAGE,R3		;GET END ADDRESS
95	003256	107203	000003		SUB	#IMLEN,R3		;POINT TO FLAG OF LAST ENTRY
96	003260	106043			CMP	R4,R3		;FIRST = LAST ?
97	003261	013267			BEQ	TKIP14		;NO - SKIP SPECIAL STUFF
98	003262	104135			MOV	(R3),R5		;GET FLAG WORD
99	003263	101205	040000		BIS	#RECIR,R5		;SET RECIRCULATION FLAG
100	003265	100135			MOV	R5,(R3)		;STORE IT BACK
101	003266	003272			BR	TKIP13		;SKIP KLUDGE FIX TO UDA
102	003267	101200	040000	001320	TKIP14: BIS	#RECIR,IMSTAR		;SET BIT IN POINTER
103	003272	104143			TKIP13: MOV	(R4),R3		;GET BUFF POINTER
104	003273	101203	100000		BIS	#LAST,R3		;SIGNAL AS LAST
105	003275	100143			MOV	R3,(R4)		;STORE IT BACK
106	003276	000000			RETURN			

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 50
 DBN/XBN TRACK FORMAT OVERLAY (G7)

1								
2								
3								
4								
5								
6								
7								
8	003277	104300	001140	000736	DPBN:	MOV	XBNSEC,TEMP	;GET NUMBER OF SECTORS IN XBN AREA
9	003302	104300	001141	000737		MOV	XBNSEC+1,TEMP+1	;GET HI ORDER
10	003305	104204	000736			MOV	#TEMP,R4	;POINT R4 AT TEMP AREA
11	003307	104203	001134			MOV	#LBNLBN,R3	;POINT AT NUM OF LBN'S IN LBN AREA
12	003311	021503				CALL	DADD	;ADD
13	003312	104203	001136			MOV	#RBNLBN,R3	;POINT TO NUM OF RBN'S IN LBN AREA
14	003314	021503				CALL	DADD	;ADD TO GET SECTORS IN LBN + XBN AREA
15	003315	104203	001114			MOV	#CURBN,R3	;POINT TO CURRENT BLOCK NUMBER(DBN)
16	003317	021503				CALL	DADD	;GET RELATIVE PBN
17	003320	104641	000001			MOV	1(R4),R1	;GET HIGH ORDER
18	003322	107301	001326			SUB	ST.DBN,R1	;SUBTRACT HIGH ORDER STARTING DBN
19	003324	104140	001111			MOV	(R4),CURPBN	;GET LO ORDER PBN
20	003326	104010	001112			MOV	R1,CURPBN+1	;STORE HIGH ORDER
21	003330	000000				RETURN		
22								
23								
24								
25								
26								
27								
28								
29								
30								
31	003331	104300	001134	000736	XPBN:	MOV	LBNLBN,TEMP	;GET NUMBER OF LBN'S IN LBN AREA
32	003334	104300	001135	000737		MOV	LBNLBN+1,TEMP+1	;GET HIGH ORDER
33	003337	104204	000736			MOV	#TEMP,R4	;POINT R4 TO TEMP AREA
34	003341	104203	001136			MOV	#RBNLBN,R3	;POINT R3 AT RBN'S IN LBN AREA
35	003343	021503				CALL	DADD	;ADD TO GET TOTAL SECTORS IN LBN AREA
36	003344	104203	001114			MOV	#CURBN,R3	;POINT R3 AT CURRENT BLOCK NUMBER
37	003346	021503				CALL	DADD	;ADD TO GET RELATIVE PBN
38	003347	104641	000001			MOV	1(R4),R1	;GET HIGH ORDER
39	003351	107301	001325			SUB	ST.XBN,R1	;SUBTRACT HIGH ORDER STARTING XBN
40	003353	104140	001111			MOV	(R4),CURPBN	;GET LO ORDER OF PBN
41	003355	104010	001112			MOV	R1,CURPBN+1	;SAVE HIGH ORDER
42	003357	000000				RETURN		

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 52
LBN FORMATTING OVERLAY (F2)

```

1          .SBTTL LBN FORMATTING OVERLAY (F2)
2          :
3          :
4          :
5 003406   :
6          :
7          :
8 003023   104200 000003 001154 LFORM: MOV    #F2,CUROVL      ;OVERLAY #2
9 003026   102200 000001 001220 BIT    #FCTAVL,FLAG  ;FCT AVAILAABLE ?
10 003031   013035          BEQ    XSKIP1      ;NO - SKIP SET UP
11 003032   104201 000036          MOV    #G3,R1      ;OVERLAY TO GET RIGHT FCT BLOCK
12 003034   022435          CALL   PAGE        ;EXECUTE IT
13 003035   104207 001053          XSKIP1: MOV   #SCR,RO  ;POINT TO CHARACTERISTICS BLOCK
14 003037   104300 001144 001126 MOV    LBNCYL,CYLNUM ;GET LO ORDER CYLINDER COUNT
15 003042   104300 001144 001455 MOV    LBNCYL,CNTCYL ;MAKE LO ORDER COUNT
16 003045   104300 001145 001127 MOV    LBNCYL+1,CYLNUM+1 ;GET HIGH ORDER
17 003050   104300 001145 001456 MOV    LBNCYL+1,CNTCYL+1 ;STORE IT
18 003053   103200 170000 001456 BIC    #HD.CLR,CNTCYL+1 ;
19 003056   104204 001126          MOV    #CYLNUM,R4   ;CLEAR STARTING CYLINDER BITS
20 003060   104203 001463          MOV    #ONE,R3      ;SUBTRACT TO GET CYLINDER NUMBER
21 003062   021521          CALL   DSUB         ;1 - BECAUSE START AT 0
22 003063   104300 001472 001236 MOV    TOTRCT,RCTTOT ;DO SUBTRACT
23 003066   104201 000052          MOV    #G8,R1      ;GET TOTAL RCT LBN'S
24 003070   022435          CALL   PAGE        ;POINT TO OVERLAY
25 003071   104207 001053          XSLEEK: MOV   #SCR,RO ;COMPUTE VARIOUS CONSTANTS
26 003073   104673 000002          MOV    GRPCYL(RO),R3 ;POINT TO CHARACTERISTICS
27 003075   103203 177400          BIC    #HIBYTE,R3   ;GET GROUPS/CYLINDER
28 003077   104030 001461          MOV    R3,GRPCNT   ;CLEAR OUT GARBAGE
29 003101   104030 001460          MOV    R3,CURGRP   ;USE AS COUNTER
30 003103   117400 001460          DEC    CURGRP      ;GROUP NUMBER
31 003105   104300 001126 001077 XSLEK2: MOV   CYLNUM,ISEEK+1 ;DECREMENT TO GET ACTUAL NUMBER
32 003110   104300 001127 001100 MOV    CYLNUM+1,ISEEK+2 ;GET CURRENT CYLINDER NUMBER
33 003113   104300 001460 001101 MOV    CURGRP,ISEEK+3 ;GET HIGH ORDER
34 003116   022242          CALL   SEEK        ;LOAD GROUP NUMBER
35 003117   115001          TST    R1          ;DO THE SEEK
36 003120   073372          BMI   SEEKER      ;ANY ERROR ?
37 003121   104207 001053          MOV    #SCR,RO     ;YUP - CUT OUT
38 003123   104673 000003          MOV    TRKGRP(RO),R3 ;POINT TO CHARACTERISTICS
39 003125   103203 177400          BIC    #HIBYTE,R3   ;GET TRACKS/GROUP
40 003127   104030 001462          MOV    R3,TRKCNT   ;CLEAR OUT GARBAGE
41 003131   117403          DEC    R3          ;MAKE COUNTER
42 003132   104030 001113          MOV    R3,CURTRK   ;WANT LAST TRACK NUMBER
43 003134   104201 000025          XSKIP3: MOV   #F8,R1 ;MAKE CURRENT TRACK 0
44 003136   022435          CALL   PAGE        ;TRACK SET UP OVERLAY
45 003137   104304 001322          XSKIP2: MOV   DPREA,R4 ;SET UP TRACK FORMAT
46 003141   104303 001321          MOV    HPREA,R3    ;GET DATA PREAMBLE LENGTH
47 003143   104307 001320          MOV    IMSTAR,RO   ;GET HEADER PREAMBLE LENGTH
48 003145   104301 001113          MOV    CURTRK,R1   ;POINT TO TRACK IMAGE START POINT
49 003147   104302 000740          MOV    UNIT,R2     ;TRACK TO FORMAT
50 003151   104205 007321          MOV    #IMAGE,R5   ;SDI INTERCONNECT
51 003153   060001          XFC   FORMAT      ;RECIRCULATION ADDRESS
52 003154   115001          TST    R1          ;DO FORMAT
53 003155   013170          BEQ    LSKIP4     ;ANY ERROR ?
54 003156   115400 000717          INC    UN.ERI     ;NO - DO CHECK PASS
55 003160   106300 001477 000717 CMP    RETRY,UN.ERI ;INCREMENT IT
56 003163   073366          BMI   FORERR     ;DONE ALL RETRIES ?
57 003164   022362          CALL   INITPT     ;YUP - ERROR
                    ;REINIT

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 52-1
LBN FORMATTING OVERLAY (F2)

58	003165	022234			CALL	CLEAR	:DRIVE CLEAR
59	003166	022242			CALL	SEEK	:RE-SEEK AND GROUP SELECT
60	003167	003137			BR	XSKIP2	:NOPE - RETRY
61	003170	114000	000717		LSKIP4: CLR	UN.ERI	:FOR STORE
62	003172	104301	001226		MOV	EMAX,R1	:GET MAX REVECTORS
63	003174	107301	001255		SUB	REVCNT,R1	:SUBTRACT CURRENT ENTRIES
64	003176	106301	001130		CMP	SECTR,R1	:ENOUGH LEFT FOR WHOLE TRACK ??
65	003200	073211			BMI	XSKIP4	:YES - CONTINUE
66	003201	104201	000011		MOV	#F4,R1	:SIGNAL RCT UPDATE OVERLAY
67	003203	022435			CALL	PAGE	:PAGE IT IN
68	003204	114000	001255		CLR	REVCNT	:FOR STORE
69	003206	104300	001225	001253	MOV	ERRBUF,ERPNT	:FOR RESET
70	003211	023377			XSKIP4: CALL	LCHEC	:DO CHECK PASS
71	003212	102200	020000	001220	BIT	#INIRCT,FLAG	:TIME TO INIT RCT ?
72	003215	013237			BEQ	XSKIP5	:NOPE
73	003216	101200	000100	001220	BIS	#REVECT,FLAG	:SET REVECTOR ON
74	003221	103200	020000	001220	BIC	#INIRCT,FLAG	:RESET FLAG
75	003224	102200	002000	001220	BIT	#BSTGS,FLAG	:DOING BEST GUESS ?
76	003227	013234			BEQ	XSKIP6	:NO - GO ALL THE WAY
77	003230	104201	000022		MOV	#F7,R1	:RCT INIT OVERLAY
78	003232	022435			CALL	PAGE	:EXECUTE IT
79	003233	003237			BR	XSKIP5	:SKIP OTHER
80	003234	104201	000014		XSKIP6: MOV	#F5,R1	:DO FCT->RCT AND INIT
81	003236	022435			CALL	PAGE	:EXECUTE IT
82	003237	117400	001113		XSKIP5: DEC	CURTRK	:DECREMENT IT
83	003241	104204	001120		MOV	#HOLDBN,R4	:GET STARTING BLOCK NUMBER
84	003243	104207	001053		MOV	#SCR,R0	:POINT TO CHARACTERISTICS
85	003245	104673	000011		MOV	LBNTRK(R0),R3	:GET LBN/TRACK
86	003247	103203	177400		BIC	#HI1BYTE,R3	:CLEAR HIGH BYTE
87	003251	104030	000731		MOV	R3,DDUMMY	:STORE IT
88	003253	114000	000732		CLR	DDUMMY+1	:FOR STORE
89	003255	104203	000731		MOV	#DDUMMY,R3	:LBN/TRACK
90	003257	021521			CALL	DSUB	:GET STARTING LBN FOR NEW TRACK
91	003260	104300	001120	001114	MOV	HOLDBN,CURBN	:GET LOW ORDER
92	003263	104300	001121	001115	MOV	HOLDBN+1,CURBN+1	:GET HIGH ORDER
93	003266	104204	001122		MOV	#HOLRBN,R4	:GET STARTING RBN NUMBER
94	003270	104673	000004		MOV	RBNTRK(R0),R3	:GET RBN/TRACK
95	003272	103203	177600		BIC	#HI1BYTE,R3	:CLERA OUT GARBAGE
96	003274	104030	000731		MOV	R3,DDUMMY	:STORE IT
97	003276	114000	000732		CLR	DDUMMY+1	:FOR STORE
98	003300	104203	000731		MOV	#DDUMMY,R3	:RBN'S/TRACK
99	003302	021521			CALL	DSUB	:GET STARTING RBN FOR NEW TRACK
100	003303	104300	001122	001107	MOV	HOLRBN,CURRBN	:GET LOW ORDER
101	003306	104300	001123	001110	MOV	HOLRBN+1,CURRBN+1	:GET HI ORDER
102	003311	117400	001462		DEC	TRKCNT	:DECREMENT IT
103	003313	053134			BNE	XSKIP3	:NO - DO NEXT TRACK
104	003314	117400	001460		DEC	CURGRP	:DECREMENT GROUP NUMBER
105	003316	117400	001461		DEC	GRPCNT	:DECREMENT IT
106	003320	053105			BNE	XSLEK2	:NO - DO NEXT GROUP
107	003321	104204	001455		MOV	#CNTCYL,R4	:GET READY TO DEC CYLINDER CNT
108	003323	104203	001463		MOV	#ONE,R3	:CONSTANT WORD OF 1
109	003325	021521			CALL	DSUB	:DECREMENT IT
110	003326	060022			XFC	UPDATE	:UPDATE PROGRESS INDICATOR
111	003327	104204	001126		MOV	#CYLNUM,R4	:GET CURRENT CYLINDER NUMBER
112	003331	104203	001463		MOV	#ONE,R3	:FOR DECREMENT
113	003333	021521			CALL	DSUB	:DECREMENT FOR NEW CYLINDER NUM
114	003334	104304	001455		MOV	CNTCYL,R4	:LOW ORDER ZERO ?

1									
2									
3									
4									
5									
6	003377	114000	001227		LCHEC:	CLR	ERK		:FOR ERROR COUNT RESET
7	003401	114000	001222			CLR	ERFLAG		:CLEAR REFORMAT FLAG
8	003403	102200	002000	001220		BIT	#BSTGS,FLAG		:BEST GUESS ?
9	003406	053421				BNE	LSKIP		:YES - DO EXTENSIVE READ
10	003407	104200	000001	001452		MOV	#1,N		:SET UP FOR STORE
11	003412	104200	000005	001453		MOV	#5,N1		:SET UP
12	003415	104300	001453	001454		MOV	N1,NN1		:SAVE FOR LATER RESET
13	003420	003432				BR	LSKIP2		:SKIP EXTENSIVE READ SETUP
14	003421	104200	000003	001452	LSKIP:	MOV	#3,N		:EXTENSIVE REGULAR READ
15	003424	104200	000024	001453		MOV	#20.,N1		:EXTENSIVE ERROR READS
16	003427	104300	001453	001454		MOV	N1,NN1		:SAVE FOR LATER RESET
17	003432	024324			LSKIP2:	CALL	LFIXIT		:EXECUTE IT
18	003433	104302	000740		LSKIP1:	MOV	UNIT,R2		:GET SDI INTERCONNECT
19	003435	060012				XFC	SIP		:WAIT FOR PULSE
20	003436	104300	001130	001451		MOV	SECTRK,SECCNT		:LOAD SECTORS/TRACK
21	003441	104205	006621			MOV	#CMDBUF,R5		:POINT TO COMMAND BUFFER
22	003443	104207	000721		LAGAIN:	MOV	#RDBLK,R0		:POINT TO READ COMMAND BLOCK
23	003445	104653	000002			MOV	RB.CMD(R5),R3		:READ COMMAND ZERO ?
24	003447	013532				BEQ	LNOERR		:SECTOR BAD - SKIP CHECKS
25	003450	100673	000004			MOV	R3,RW.CMD(R0)		:ELSE STORE IN COMMAND BLOCK
26	003452	104653	000000			MOV	RB.LOW(R5),R3		:LOAD LOW ORDER SECTOR NUMBER
27	003454	100673	000002			MOV	R3,RW.LOW(R0)		:STORE IN COMMAND BLOCK
28	003456	104653	000001			MOV	RB.HI(R5),R3		:LOAD HIGH ORDER BLOCK NUMBER
29	003460	100673	000003			MOV	R3,RW.HI(R0)		:STORE IN COMMAND BLOCK
30	003462	104203	004535			MOV	#RDBUF,R3		:GET BUFFER POINTER
31	003464	100673	000001			MOV	R3,RW.BUF(R0)		:STORE IN COMMAND BLOCK
32	003466	104203	000726			MOV	#HSLIM-1,R3		:POINTER TO DUMMY SDI BLOCK
33	003470	100673	000005			MOV	R3,RW.DUM(R0)		:STORE IN READ BLOCK
34	003472	104207	100721		READ3:	MOV	#<BIT15!RDBLK>,R0		:MAKE SURE POINTING AT BLOCK
35	003474	104203	100000			MOV	#RDCMD,R3		:RESET STATUS POINTER
36	003476	100673	000000			MOV	R3,RW.STAT(R0)		:STORE IT BACK
37	003500	060002				XFC	READ		:READ 1 SECTOR
38	003501	115001				TST	R1		:ANY ERRORS ?
39	003502	053515				BNE	LERR		:YES - UH OH
40	003503	104307	000721			MOV	RDBLK,R0		:GET STATUS WORD
41	003505	102207	0100C0			BIT	#ECCF,R0		:ECC ERROR ?
42	003507	053515				BNE	LERR		:YES - MARK AS BAD FOR NOW
43	003510	104207	001440			MOV	#NUM,R0		:POINT TO COMPARE BLOCK
44	003512	060006				XFC	CPDAT		:DO DATA COMPARE
45	003513	115001				TST	R1		:ANY ERROR IN COMPARE ?
46	003514	013532				BEQ	LNOERR		:NOPE - CONTINUE LOOP
47	003515				LERR:				
48	003515	104653	000003			MOV	RB.IM(R5),R3		:GET IMAGE POINTER
49	003517	104134				MOV	(R3),R4		:GET BUFFER POINTER WORD
50	003520	102204	020000			BIT	#BD,R4		:ALREADY BEEN HERE ??
51	003522	053532				BNE	LNOERR		:YUP - DON'T COUNT AGAIN
52	003523	101204	020000			BIS	#BD,R4		:MARK AS BAD
53	003525	100134				MOV	R4,(R3)		:STORE BACK
54	003526	115400	001470			INC	RTYCNT		:INC IT
55	003530	115400	001227			INC	ERR		:INCREMENT ERROR COUNT
56	003532	105205	000004		LNOERR:	ADD	#RDLEN,R5		:POINT TO NEXT READ CMD BLOCK
57	003534	117400	001451			DEC	SECCNT		:DECREMENT COUNTER

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 54-2
LBN FORMATTING OVERLAY (F2)

115	003671	106203	060000		CMP	#HD.RBN,R3	: IS IT A BAD RBN ?
116	003673	013774			BEQ	BRBN	: YUP - GO HANDLE IT
117	003674	104643	000002		MOV	FT.HI(R4),R3	: RELOAD HEADER
118	003676	103203	170000		BIC	#HD.CLR,R3	: CLEAR THE HEADER
119	003700	102200	001000	001220	BIT	#PRIM,FLAG	: ANY PRIMARY YET ?
120	003703	054116			BNE	SND	: YUP - THIS ONE SECONDARY
121	003704	104030	000732		MOV	R3,DDUMMY+1	: STORE HIGH ORDER FOR RBN COMPUTATION
122	003706	101200	001000	001220	BIS	#PRIM,FLAG	: SET PRIMARY FLAG
123	003711	101200	000004	001221	BIS	#RPRIM,FLAG1	: SET GOOD RBN EDC NEEDED
124	003714	101203	050000		BIS	#HD.PRIV,R3	: MARK AS PRIMARY
125	003716	100643	000002		MOV	R3,FT.HI(R4)	: STORE BACK IN IMAGE
126	003720	104640	000001	000731	MOV	FT.LOW(R4),DDUMMY	: STORE LOW ORDER FOR RBN COMPUTATION
127	003723	104042			MOV	R4,R2	: SAVE IMAGE POINTER
128	003724	104207	001053		MOV	#SCR,R0	: MAKE SURE POINT TO CHAR BLOCK
129	003726	104204	000731		MOV	#DDUMMY,R4	: POINT TO BLOCK NUMBER
130	003730	022730			CALL	PRIMRB	: GET PRIMARY RBN NUMBER
131	003731	104307	001152		MOV	REVRBN,R0	: GET NUMBER OF REVECTORED RBN
132	003733	104301	001153		MOV	REVRBN+1,R1	: GET HIGH ORDER
133	003735	105301	001324		ADD	ST.RBN,R1	: ADD STARTING RBN BITS
134	003737	101201	060000		RIS	#HD.RBN,R1	: SET IN RBN HEADER CODE
135	003741	104205	006204		MOV	#PRMBUF,R5	: USE RDBUF TO HOLD 128 COPIES OF RBN
136	003743	104203	000200		MOV	#RBNRPT,R3	: COUNT OF REPLICATED RBN'S
137	003745	100257		RPT1:	MOV	R0,(R5)+	: STORE A COPY
138	003746	100251			MOV	R1,(R5)+	: AND HIGH ORDER
139	003747	117403			DEC	R3	: DECREMENT COUNTER - DONE ?
140	003750	053745			BNE	RPT1	: NO - STORE ANOTHER COPY
141	003751	104024			MOV	R2,R4	: RESTORE IMAGE POINTER
142	003752	104205	006204		MOV	#PRMBUF,R5	: POINT TO BEGINNING OF BUFFER
143	003754	104642	000000		MOV	FT.BUF(R4),R2	: GET BUFFER POINTER
144	003756	103202	007777		BIC	#BUFMSK,R2	: CLEAR ONLY BUFFER POINTER
145	003760	101052			BIS	R5,R2	: OR IN NEW BUFFER POINTER
146	003761	100642	000000		MOV	R2,FT.BUF(R4)	: STORE IT BACK
147	003763	104202	006204		MOV	#PRMBUF,R2	: FOR EDC COMPUTATION
148	003765	022600			CALL	CEDC	: COMPUTE IT
149	003766	100623	000400		MOV	R3,RW.EDC(R2)	: STORE IT
150	003770	103200	000040	001221	BIC	#BDHD,FLAG1	: WAN'T TO STAY PRIMARY
151	003773	004130			BR	LSND	: BRANCH AROUND SECONDARY
152	003774	117400	001450	BRBN:	DEC	ERRCNT	: DEC ERR CNT SO PRIMARY STATS WILL BE RIGHT
153	003776	102200	001000	001220	BIT	#PRIM,FLAG	: IS THERE A PRIMARY ON THIS TRACK ?
154	004001	014073			BEQ	7\$: NO - SKIP HEADER RESET
155	004002	104203	007321		MOV	#IMAGE,R3	: POINT TO FORMAT TABLE
156	004004	104632	000002	5\$:	MOV	FT.HI(R3),R2	: GET HEADER WORD
157	004006	103202	007777		BIC	#LO,R2	: CLEAR ALL BUT HEADER
158	004010	106202	050000		CMP	#HD.PRIV,R2	: IS IT THE PRIMARY ?
159	004012	014016			BEQ	6\$: YES - DONE
160	004013	105203	000003		ADD	#IMLEN,R3	: NO - POINT TO NEXT ENTRY
161	004015	004004			BR	5\$: CHECK NEXT ENTRY
162	004016	104632	000002	6\$:	MOV	FT.HI(R3),R2	: RESET TO HI ORDER
163	004020	103202	170000		BIC	#HD.CLR,R2	: CLEAR HEADER
164	004022	101202	030000		BIS	#HD.REV,R2	: MARK AS SECONDARY
165	004024	100632	000002		MOV	R2,FT.HI(R3)	: STORE BACK
166	004026	104202	005567		MOV	#GDBLK,R2	: POINT TO GOOD BLOCK
167	004030	104635	000000		MOV	FT.BUF(R3),R5	: GET BUFFER POINTER AND FLAGS
168	004032	103205	007777		BIC	#BUFMSK,R5	: CLEAR ONLY BUFFER POINTER
169	004034	101025			BIS	R2,R5	: OR IN NEW BUFFER POINTER
170	004035	100635	000000		MOV	R5,FT.BUF(R3)	: MOVE IN BUFFER POINTER AND FLAGS
171	004037	104305	001253		MOV	ERPNT,R5	: GET REVECTOR POINTER

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 54-3
 LBN FORMATTING OVERLAY (F2)

172	004041	102200	020000	001221	BIT	#FPRIM,FLAG1	:WAS IT A PRIMARY FROM THE FCT ?
173	004044	054066			BNE	9\$:YES - UPDATE WILL MAKE IT SECONDARY
174	004045	107205	000002		8\$: SUB	#ERLEN,R5	:LOOK FOR PRIMARY BACKWARDS
175	004047	104652	000001		MOV	1(R5),R2	:GET HIGH ORDER
176	004051	103202	007777		BIC	#LO,R2	:CLEAR LO STUFF
177	004053	106202	050000		CMF	#HD.PRIV,R2	:IS IT THE PRIMARY ?
178	004055	054045			BNE	8\$:NO - KEEP LOOKING
179	004056	104652	000001		MOV	1(R5),R2	:GET HIGH ORDER AGAIN
180	004060	103202	170000		BIC	#HD.CLR,R2	:CLEAR THE HEADER
181	004062	101202	030000		BIS	#HD.REV,R2	:MAKE IT SECONDARY
182	004064	100652	000001		MOV	R2,1(R5)	:STORE IT BACK
183	004066	115400	001467		9\$: INC	SNDCNT	:INC SECONDARY COUNTER
184	004070	103200	000004	001221	BIC	#RPRIM,FLAG1	:DON'T NEED GOOD EDC ANY LONGER
185	004073	103200	000040	001221	7\$: BIC	#BDHD,FLAG1	:CLEAR SO WILL PUT IN AS RBN
186	004076	101200	001000	001220	BIS	#PRIM,FLAG	:SET SO NONE WILL BE PRIMARY
187	004101	104643	000002		BDIRCT: MOV	FT.HI(R4),R3	:RELOAD HEADER
188	004103	103203	170000		BIC	#HD.CLR,R3	:CLEAR THE HEADER
189	004105	101203	110000		BIS	#HD.BAD,R3	:MARK AS BAD
190	004107	100643	000002		MOV	R3,FT.HI(R4)	:STORE BACK IN IMAGE
191	004111	102200	000100	001220	BIT	#REVECT,FLAG	:IN RCT ?
192	004114	054130			BNE	LSND	:NO - PUT IN TO REVECTOR
193	004115	004166			BR	LSKIP3	:ELSE DO NEXT ENTRY
194	004116	102200	000040	001221	SND: BIT	#BDHD,FLAG1	:BAD HEADER ?
195	004121	054101			BNE	BDIRCT	:YUP - MARK AS BAD
196	004122	101203	030000		BIS	#HD.REV,R3	:MARK AS SECONDARY
197	004124	100643	000002		MOV	R3,FT.HI(R4)	:STORE BACK IN IMAGE
198	004126	115400	001467		INC	SNDCNT	:INC IT
199	004130	115400	001222		LSND: INC	ERFLAG	:SET RE-FORMAT FLAG
200	004132	104303	001253		MOV	ERPNT,R3	:STORE BACK
201	004134	104642	000001		MOV	FT.LOW(R4),R2	:GET LOW ORDER BLOCK NUMBER
202	004136	100232			MOV	R2,(R3)+	:STORE FOR RCT UPDATE
203	004137	104642	000002		MOV	FT.HI(R4),R2	:GET HIGH ORDER
204	004141	102200	000040	001221	BIT	#BDHD,FLAG1	:BAD HEADER ?
205	004144	014153			BEQ	LSKIP9	:NO - HANDLE AS USUAL
206	004145	103202	170000		BIC	#HD.CLR,R2	:ELSE CLEAR BAD HEADER CODE
207	004147	101202	030000		BIS	#HD.REV,R2	:AND PUT IN SECONDARY CODE
208	004151	115400	001467		INC	SNDCNT	:INC SECONDARY COUNT
209	004153	100232			LSKIP9: MOV	R2,(R3)+	:STORE FOR RCT UPDATE
210	004154	104030	001253		MOV	R3,ERPNT	:STORE BACK
211	004156	115400	001255		INC	REVCNT	:INCREMENT IT
212	004160	115400	001450		INC	ERRCNT	:UP COUNTER OF BAD BLOCKS
213	004162	004166			BR	LSKIP3	:NO NEED TO RE-READ ANY MORE THIS SECTOR
214	004163	117400	001453		LOK: DEC	N1	:DECREMENT COUNTER
215	004165	053547			BNE	LHERE	:NO - RE-READ SECTOR IN ERROR
216	004166	104300	001454	001453	LSKIP3: MOV	NN1,N1	:GET SAVED VALUE
217	004171	117400	001227		DEC	ERR	:DECREMENT IT
218	004173	103200	000040	001221	LSKIP7: BIC	#BDHD,FLAG1	:CLEAR BAD HEADER FLAG
219	004176	105204	000003		ADD	#IMLEN,R4	:POINT TO NEXT ENTRY
220	004200	115000	001227		TST	ERR	:DONE ALL SECTORS ?
221	004202	053547			BNE	LHERE	:NO - DO NEXT SECTOR
222	004203	115000	001222		TST	ERFLAG	:WERE THERE ANY BAD SECTORS FOUND
223	004205	014323			BEQ	LNONE	:NOPE - ALL DONE
224	004206	104207	006621		MOV	#RBNBUF,R0	:POINT TO RBN BUFFER
225	004210	104301	001445		MOV	DWRD,R1	:DIAGNOSTIC WORD
226	004212	100271			MOV	R1,(R0)+	:STORE IT
227	004213	104204	000125		MOV	#85.,R4	:SET COUNTER OF TRIPLE WORDS
228	004215	104301	001442		MOV	FWRD,R1	:FIRST WORD OF PATTERN

UDAFM - UDA FORMATTER DMACH X04.01 23-AUG-82 14:02:32 PAGE 54-4
LBN FORMATTING OVERLAY (F2)

229	004217	104302	001443		MOV	SWRD,R2		:SECOND WORD OF PATTERN
230	004221	104303	001444		MOV	TWRD,R3		:THIRD WORD OF PATTERN
231	004223	100271		LOVER:	MOV	R1,(R0)+		:STORE IT
232	004224	100272			MOV	R2,(R0)+		:STORE IT
233	004225	100273			MOV	R3,(R0)+		:STORE IT
234	004226	117404			DEC	R4		:DECREMENT COUNTER
235	004227	054223			BNE	LOVER		:REPEAT TILL DONE
236	004230	104302	001447		MOV	BAEDC,R2		:EDC FOR PATTERN (FORCED ERROR IND)
237	004232	100272			MOV	R2,(R0)+		:STORE IT
238	004233	102200	000004	001221	BIT	#RPRIM,FLAG1		:NEED GOOD RBN EDC ???
239	004236	014264			BEQ	LOVER2		:NOPE
240	004237	104203	007321		MOV	#IMAGE,R3		:POINT TO IMAGE
241	004241	104205	005567		MOV	#GDBLK,R5		:POINT TO GOOD BLOCK
242	004243	104632	000002	LOVER4:	MOV	FT.HI(R3),R2		:GET HI ORDER
243	004245	103202	007777		BIC	#LO,R2		:CLEAR JUNK
244	004247	106202	060000		CMP	#HD,RBN,R2		:IS IT THE PRIMARY ?
245	004251	014255			BEQ	LOVER3		:YUP - HANDLE IT
246	004252	105203	000003		ADD	#IMLEN,R3		:CHECK NEXT ENTRY
247	004254	004243			BR	LOVER4		:TRY AGAIN
248	004255	104632	000000	LOVER3:	MOV	FT.BUF(R3),R2		:GET BUFFER POINTER
249	004257	103202	007777		BIC	#BUFMSK,R2		:CLEAR ONLY BUFFER POINTER
250	004261	101052			BIS	R5,R2		:SET IN NEW BUFFER POINTER
251	004262	100632	000000		MOV	R2,FT.BUF(R3)		:STORE IT
252	004264	104304	001322	LOVER2:	MOV	DPREA,R4		:DATA PREAMBLE LENGTH
253	004266	104303	001321		MOV	HPREA,R3		:HEADER PREAMBLE LENGTH
254	004270	104307	001320		MOV	IMSTAR,R0		:POINT TO TRACK IMAGE START POINT
255	004272	104301	001113		MOV	CURTRK,R1		:CURRENT TRACK NUMBER
256	004274	104302	000740		MOV	UNIT,R2		:SDI INTERCONNECT
257	004276	104205	007321		MOV	#IMAGE,R5		:RECIRCULATION ADDRESS
258	004300	060001			XFC	FORMAT		:RE-FORMAT
259	004301	115001			TST	R1		:ANY PROBLEMS ??
260	004302	014320			BEQ	LOVER1		:NO - DO CHECK PASS
261	004303	115400	000717		INC	UN.ERI		:INCREMENT IT
262	004305	106300	001477	000717	CMP	RETRY,UN.ERI		:DONE ALL RETRIES ?
263	004310	073366			BMI	FORERR		:UP - ERROR
264	004311	022362			CALL	INITPT		:REINIT
265	004312	022234			CALL	CLEAR		:DRIVE CLEAR
266	004313	104300	001460	001101	MOV	CURGRP,ISEEK+3		:GROUP
267	004316	022242			CALL	SEEK		:RE-SEEK AND GROUP SELECT
268	004317	004264			BR	LOVER2		:NOPE - RETRY
269	004320	114000	000717	LOVER1:	CLR	UN.ERI		:CLEAR RETRY COUNT
270	004322	003377			BR	LCHEC		:RE-CYCLE CHECK PASS
271	004323	000000		LDONE:	RETURN			

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 55
LBN FORMATTING OVERLAY (F2)

```

1
2
3
4
5
6 004324 104300 001130 001451 LFIXIT: MOV SECTRK,SECCNT ;INIT COUNTER
7 004327 104207 006621 MOV #CMDBUF,R0 ;COMMAND BUFFER
8 004331 104205 007321 MOV #IMAGE,R5 ;POINT TO TRACK IMAGE
9 004333 104303 001234 MOV SKPCNT,R3 ;GET STARTING OFFSET(TUNED)
10 004335 105035 ADD R3,R5 ;POINT TO FIRST ENTRY
11 004336 104050 001233 MOV R5,STARIT ;MARK STARTING ADDRESS
12 004340 104653 000002 LMORE: MOV 2(R5),R3 ;SET UP FOR HSR CODE COMPARE
13 004342 103203 007777 BIC #LO,R3 ;ISOLATE HI 4 BITS(HDR CODE)
14 004344 106203 000000 CMP #HD.LBN,R3 ;GOOD LBN ?
15 004346 014360 BEQ FLKIP2 ;YES - MARK AS GOOD TO CHECK
16 004347 106203 060000 CMP #HD.RBN,R3 ;GOOD RBN ?
17 004351 014360 BEQ FLKIP2 ;YES - MARK AS GOOD TO CHECK
18 004352 114003 CLR R3 ;CLEAR FOR STORE
19 004353 100673 000002 MOV R3,RB.CMD(R0) ;STORE AS BAD SECTR FLAG
20 004355 105207 000004 ADD #RDLEN,R0 ;POINT TO NEXT BLOCK
21 004357 004374 BR FLKIP1 ;SKIP GOOD MARK
22 004360 104653 000001 FLKIP2: MOV 1(R5),R3 ;LO ORDER BLOCK NUMBER
23 004362 100273 MOV R3,(R0)+ ;STORE IN READ CMD BLOCK
24 004363 104653 000002 MOV 2(R5),R3 ;HI ORDER BLOCK NUM AND CODE
25 004365 100273 MOV R3,(R0)+ ;STORE IN READ CMD BLOCK
26 004366 104203 013400 MOV #RWCMD,R3 ;LOAD SDI READ COMMAND
27 004370 101303 001113 BIS CURTRK,R3 ;SET IN CURRENT TRACK NUMBER
28 004372 100273 MOV R3,(R0)+ ;STORE IN BLOCK
29 004373 100275 MOV R5,(R0)+ ;SAVE PTR TO IMAGE BLK ENTRY
30 004374 105305 001235 FLKIP1: ADD TBLK,R5 ;ADD TO GET NEXT SECTOR
31 004376 106305 001232 CMP EIMAGE,R5 ;SEE IF HAVE TO LOOP BACK TO TOP
32 004400 014407 BEQ LREDO ;NEED TO RESET
33 004401 034411 BPL FLKIP1 ;NO NEED - JUST CONTINUE
34 004402 107305 001232 SUB EIMAGE,R5 ;SUBTRACT TO GET LOOP AMOUNT
35 004404 105205 007321 ADD #IMAGE,R5 ;AND ADD OFFSET
36 004406 004411 BR FLKIP1 ;SKIP ZERO CONDITION
37 004407 104205 007321 LREDO: MOV #IMAGE,R5 ;IF ZERO SIMPLY MOVE TO FRONT
38 004411 106305 001233 FLKIP1: CMP STARIT,R5 ;AT BEGINNING ADDRESS ?
39 004413 054420 BNE FKIP9 ;NO - JUST CONTINUE
40 004414 105205 000003 ADD #IMLEN,R5 ;ELSE POINT TO NEXT ENTRY
41 004416 104050 001233 MOV R5,STARIT ;MAKE IT NEW STARTING ADDRESS
42 004420 117400 001451 FKIP9: DEC SECCNT ;DECREMENT
43 004422 054340 BNE LMORE ;NO - DO NEXT SECTOR
44 004423 000000 RETURN

```

UDAFM - JDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 56
LBN FORMAT IMAGE SETUP OVERLAY (F8)

```

1          .SBTTL LBN FORMAT IMAGE SETUP OVERLAY (F8)
2 004424  DMOVLY F8,START
3          :
4          :
5          :
6 003023 104200 000025 001154 LTRK:  MOV    #F8,CUROVL      ;GET OVERLAY OFFSET
7 003026 104207 001053          MOV    #SCR,RO        ;POINT TO CHARACTERISTICS BLOCK
8 003030 104673 000011          MOV    LBNTRK(RO),R3  ;GET LBN'S/TRACK
9 003032 103203 177400          BIC    #HI1BYTE,R3   ;CLEAR HIGH BYTE
10 003034 104030 001451          MOV    R3,SECCNT     ;USE AS COUNTER
11 003036 104205 007321          MOV    #IMAGE,R5     ;POINT TO TRACK IMAGE BUFFER
12 003040 104203 005567          MOV    #GDBLK,R3     ;POINT TO DATA BLOCK
13 003042 104304 001446          MOV    EDC,R4        ;GET GOOD EDC
14 003044 100634 000400          MOV    R4,RW.EDC(R3) ;STORE IN BUFFER
15 003046 103200 001000 001220 LKIP2: BIC    #PRIM,FLAG    ;CLEAR PRIMARY FLAG
16 003051 104203 005567          MOV    #GDBLK,R3     ;POINT TO GOOD DATA BLOCK
17 003053 100653 000000          MOV    R3,FT.BUF(R5) ;STORE IN IMAGE BLOCK
18 003055 104303 001114          MOV    CURBN,R3      ;GET LOW ORDER BLOCK NUMBER
19 003057 100653 000001          MOV    R3,FT.LOW(R5) ;STORE IN IMAGE BLOCK
20 003061 104303 001115          MOV    CURBN+1,R3    ;GET HIGH ORDER BLOCK NUMBER
21 003063 103203 170000          BIC    #HD.CLR,R3    ;CLEAR HEADER
22 003065 101203 000000          BIS    #HD.LBN,R3    ;SET IN LBN HEADER
23 003067 100653 000002          MOV    R3,FT.HI(R5) ;STORE IN IMAGE BLOCK
24 003071 105205 000003          ADD    #IMLEN,R5     ;POINT TO NEXT ENTRY
25 003073          DUBINC CURBN        ;INCREMENT BLOCK NUMBER
26 003100 102200 000100 001220 BIT    #REVECT,FLAG  ;STILL IN RCT AREA ?
27 003103 053115          BNE    LKIP9         ;NO - NO NEED TO DECREMENT
28 003104 117400 001236          DEC    RCTTOT        ;DECREMENT IT
29 003106 053115          BNE    LKIP9         ;OUT OF RCT ? - NO
30 003107 101200 020000 001220 BIS    #INIRCT,FLAG  ;SET TO REVECTOR
31 003112 104300 001240 001313 MOV    FCNT,LBNBAD   ;GET FCT ENTRY COUNT - AFTER RCT
32 003115          :
33 003115 117400 001451          DEC    SECCNT        ;DECREMENT COUNTER
34 003117 053051          BNE    LKIP2        ;CONTINUE TILL DONE ALL
35          :
36          :
37          :
38          :
39 003120          PUSH   R5            ;SAVE POINTER TO FIRST RBN ENTRY
40 003121 104207 001053          MOV    #SCR,RO        ;POINT TO CHARACTERISTICS
41 003123 104673 000004          MOV    RBNTRK(RO),R3  ;GET RBN'S/TRACK
42 003125 103203 177600          BIC    #HI1BYTE,R3   ;CLEAR HIGH GARBAGE
43 003127 104030 001451          MOV    R3,SECCNT     ;USE AS COUNTER
44 003131 104207 006621          LKIP8: MOV    #RBNBUF,RO ;POINT TO RBN BUFFER
45 003133 104201 005567          MOV    #GDBLK,R1     ;POINT TO GOOD BLOCK
46 003135 104202 000400          MOV    #SECSI6,R2    ;COUNT OF WORDS IN SECTOR
47 003137 104213          LKIP20: MOV    (R1)+,R3  ;GET WORD
48 003140 100273          MOV    R3,(R0)+     ;STORE IT
49 003141 117402          DEC    R2            ;DECREMENT COUNTER
50 003142 053137          BNE    LKIP20       ;DO WHOLE BUFFER
51 003143 104302 001447          MOV    BADEDC,R2     ;EDC FOR PATTERN (FORCED ERROR IND)
52 003145 100272          MOV    R2,(R0)+     ;STORE IT
53 003146 104203 006621          LKIP21: MOV    #RBNBUF,R3 ;POINT TO BUFFER
54 003150 100653 000000          MOV    R3,FT.BUF(R5) ;STORE IN IMAGE
55 003152 104303 001107          MOV    CURBN,R3      ;GET LOW ORDER RBN
56 003154 100653 000001          MOV    R3,FT.LOW(R5) ;STORE IN IMAGE
57 003156 104303 001110          MOV    CURBN+1,R3    ;GET HIGH ORDER RBN
          BIC    #HD.CLR,R3 ;CLEAR HEADER

```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 56-1
LBN FORMAT IMAGE SETUP OVERLAY (F8)

58	003162	101203	060000		BIS	#HD.RBN,R3	:SET IN RBN HEADER
59	003164	100653	000002		MOV	R3,FT.HI(R5)	:STORE IN IMAGE
60	003166	105205	000003		ADD	#IMLEN,R5	:POINT TO NEXT ENTRY
61	003170				DUBINC	CURRBN	:INCREMENT RBN NUMBER
62	003175	117400	001451		DEC	SECCNT	:DECREMENT COUNTER
63	003177	053146			BNE	LKIP21	:CONTINUE TILL DONE
64							
65							
66	003200	104300	001130	001451	MOV	SECTRK,SECCNT	:SET UP COUNTER
67	003203	104204	007321		MOV	#IMAGE,R4	:POINT TO IMAGE
68	003205	104203	001053		MOV	#SCR,P3	:POINT TO CHARACTERISTICS
69	003207	104632	000011		MOV	OFFS(R3),R2	:GET GROUP OFFSET
70	003211	110702			S#AB	R2	:GET INTO LOWBYTE
71	003212	103202	177400		BIC	#HIBYTE,R2	:CLEAR HIGH GARBAGE
72	003214	115002			TST	R2	:ANY OFFSET ?
73	003215	013263			BEQ	LKIP22	:NO - SKIP CALCULATIONS
74	003216	115000	001460		TST	CURGRP	:IS GROUP ZERO ???
75	003220	013263			BEQ	LKIP22	:YES - NO OFFSET
76	003221	104020	000736		MOV	R2,TEMP	:STORE IT
77	003223	114000	000737		CLR	TEMP+1	:FOR STORE
78	003225	104300	001460	000731	MOV	CURGRP,DDUMMY	:GET CURRENT GROUP
79	003230	114000	000732		CLR	DDUMMY+1	:CLEAR HIGH WORD
80	003232	104203	000736		MOV	#TEMP,R3	:FOR MUL
81	003234	104204	000731		MOV	#DDUMMY,R4	:DITTO
82	003236	021537			CALL	DMUL	:MULTIPLY TO GET OFFSET FOR THIS GROUP
83	003237	106300	001130	000731	LKIP23: CMP	SECTRK,DDUMMY	:IS TOTAL OFFSET MORE THAN NUMBER OF SECTORS ?
84	003242	033247			BPL	LKIP24	:NO - ALL IS FINE
85	003243	107300	001130	000731	SUB	SECTRK,DDUMMY	:YES - SUBTRACT TILL IT IS
86	003246	003237			BR	LKIP23	:CHECK AGAIN
87	003247	104200	000003	000736	LKIP24: MOV	#IMLEN,TEMP	:GET LENGTH OF IMAGE BLOCK
88	003252	114000	000737		CLR	TEMP+1	:FOR STORE
89	003254	104203	000736		MOV	#TEMP,R3	:FOR MULT
90	003256	021537			CALL	DMUL	:GET LENGTH TO OFFSET
91	003257	104143			MOV	(R4),R3	:GET RESULT
92	003260	104304	001232		MOV	EIMAGE,R4	:GET ADDRESS OF END OF IMAGE
93	003262	107034			SUB	R3,R4	:SUBTRACT TO GET STARTING LOCATION
94	003263	104040	001320		LKIP22: MOV	R4,IMSTAR	:STORE IMAGE POINTER
95	003265	104300	001124	001111	MOV	HOLDPN,CURPBN	:GET LOW ORDER PBN
96	003270	104300	001125	001112	MOV	HOLDPN+1,CURPBN+1	:GET HIGH ORDER
97	003273	102200	000001	001220	LKIP27: BIT	#FCTAVL,FLAG	:IS FCT AVAILABLE ?
98	003276	013523			BEQ	LKIP28	:NO - ASSUME BLOCK GOOD
99	003277	102200	000002	001220	BIT	#FCTEMT,FLAG	:FCT EMPTY ?
100	003302	053523			BNE	LKIP28	:YUP - BLOCK IS GOOD
101	003303	104302	001224		MOV	BADPBN,R2	:GET FCT POINTER
102	003305	104121			MOV	(R2),R1	:GET LOW ORDER BAD PBN
103	003306	106010	001111		CMR	R1,CURPBN	:ARE THEY EQUAL ?
104	003310	053500			BNE	LKIP4	:NO - SKIP REST OF COMPARE
105	003311	104621	000001		MOV	1(R2),R1	:GET HIGH ORDER
106	003313	103201	170000		BIC	#HD.CLR,R1	:CLEAR THE HEADER
107	003315	106010	001112		CMR	R1,CURPBN+1	:ARE THEY EQUAL ?
108	003317	053500			BNE	LKIP4	:NO - MUST BE GOOD
109	003320	117400	001475		DEC	PCNT	:DECREMENT IT
110	003322	117400	001240		DEC	FCNT	:DECREMENT FCT COUNT
111	003324	053330			BNE	LKIP12	:IF NON - ZERO SKIP FLAG SET
112	003325	101200	000002	001220	BIS	#FCTEMT,FLAG	:SET EMPTY FLAG
113	003330				LKIP12:		
114	003330	104643	000002		MOV	FT.HI(R4),R3	:GET HEADER WORD

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 56-2
LBN FORMAT IMAGE SETUP OVERLAY (F8)

115	003332	103203	007777		BIC	#LO,R3	:CLEAR ALL BUT HEADER
116	003334	106203	060000		CMP	#HD,RBN,R3	:IS IT AN RBN ?
117	003336	053343			BNE	LKIP25	:NOPE - SKIP RBN STUFF
118	003337	101200	001000	001220	BIS	#PRIM,FLAG	:SO WON'T GET PRIMARY ON CHECK PASS
119	003342	003465			BR	MARBAD	:GO MARK AS BAD
120	003343	102200	000100	001220	LKIP25:	BIT	#REVECT,FLAG
121	003346	053352			BNE	6\$:IN RCT ?
122	003347	115400	001314		INC	RCTBAD	:NO - SKIP RCT STUFF
123	003351	003465			BR	MARBAD	:INCREMENT IT
124	003352	104623	000001		MOV	1(R2),R3	:GO MARK BAD
125	003354	102203	100000	6\$:	BIT	#PRMY,R3	:GET BAD PBN HDR
126	003356	013444			BEQ	LKIP5	:IS IT SECONDARY ?
127	003357	101200	001000	001220	BIS	#PRIM,FLAG	:YES - GO DO IT
128	003362	101200	000004	001221	BIS	#RPRIM,FLAG1	:SET FLAG FOR PRIMARY FOUND
129	003365	101200	020000	001221	BIS	#FPRIM,FLAG1	:SET GOOD EDC NEEDED
130	003370	104643	000002		MOV	FT.HI(R4),R3	:SET PRIMARY IN FCT FLAG
131	003372	103203	170000		BIC	#HD,CLR,R3	:GET HIGH ORDER HEADER
132	003374	101203	050000		BIS	#HD,PRV,R3	:CLEAR HEADER CODE
133	003376	100643	000002		MOV	R3,FT.HI(R4)	:MARK AS PRIMARY REVECTOR
134	003400				PUSH	R4	:STORE IN IMAGE BLOCK
135	003401	104207	001053		MOV	#SCR,R0	:SAVE IMAGE POINTER
136	003403	104204	001114		MOV	#CURBN,R4	:MAKE SURE POINT TO CHAR BLOCK
137	003405	022730			CALL	PRIMR8	:POINT TO BLOCK NUMBER
138	003406				POP	R4	:GET PRIMARY RBN NUMBER
139	003407	104307	001152		MOV	REVRBN,R0	:RESTORE IMAGE POINTER
140	003411	104301	001153		MOV	REVRBN+1,R1	:GET NUMBER OF REVECTORED RBN
141	003413	105301	001324		ADD	ST.RBN,R1	:GET HIGH ORDER
142	003415	101201	060000		BIS	#HD,RBN,R1	:ADD STARTING RBN BITS
143	003417	104205	006204		MOV	#PRMBUF,R5	:SET IN RBN HEADER CODE
144	003421	104203	000200		MOV	#RBNRPT,R3	:USE RDBUF TO HOLD 128 COPIES OF RBN
145	003423	100257			MOV	R0,(R5)+	:COUNT OF REPLICATED RBN'S
146	003424	100251			MOV	R1,(R5)+	:STORE A COPY
147	003425	117403			DEC	R3	:AND HIGH ORDER
148	003426	053423			BNE	RPT	:DECREMENT COUNTER - DONE ?
149	003427	104203	006204		MOV	#PRMBUF,R3	:NO - STORE ANOTHER COPY
150	003431	100643	000000		MOV	R3,FT.BUF(R4)	:POINT TO BEGINNING OF BUFFER
151	003433	105200	000002	001224	ADD	#2,BADPBN	:STORE NEW BUFFER PTR IN IMAGE
152	003436	104202	006204		MOV	#PRMBUF,R2	:INCREMENT BADPBN POINTER
153	003440	022600			CALL	CEDC	:POINT TO BUFFER
154	003441	100623	000400		MOV	R3,RW.EDC(R2)	:COMPUTE EDC - RETURNED IN R3
155	003443	003500			BR	LKIP4	:STORE IT
156	003444				LKIP5:		:SKIP SECONDARY REVECTOR
157	003444	115400	001467		INC	SNDCNT	:INC IT
158	003446	102203	010000		BIT	#FBDHD,R3	:HEADER IN ERROR CODE IN FCT ?
159	003450	053465			BNE	MARBAD	:YUP - MARK BAD
160	003451	104643	000002		MOV	FT.HI(R4),R3	:GET HIGH ORDER HEADER
161	003453	103203	170000		BIC	#HD,CLR,R3	:CLEAR HEADER CODE
162	003455	101203	030000		BIS	#HD,REV,R3	:SET HEADER TO SECONDARY REVECTOR
163	003457	100643	000002		MOV	R3,FT.HI(R4)	:STORE IN IMAGE
164	003461	105200	000002	001224	ADD	#2,BADPBN	:INCREMENT BADPBN POINTER
165	003464	003500			BR	LKIP4	:SKIP GOOD MARK
166	003465				MARBAD:		
167	003465	104643	000002		MOV	FT.HI(R4),R3	:GET HIGH ORDER HEADER
168	003467	103203	170000		BIC	#HD,CLR,R3	:CLEAR HEADER CODE
169	003471	101203	110000		BIS	#HD,BAD,R3	:MARK AS BAD
170	003473	100643	000002		MOV	R3,FT.HI(R4)	:STORE IN IMAGE
171	003475	105200	000002	001224	ADD	#2,BADPBN	:UPDATE COUNTER

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 56-3
LBN FORMAT IMAGE SETUP OVERLAY (F8)

172	003500				LKIP4:	ADD	#IMLEN,R4	:POINT TO NEXT IMAGE ENTRY
173	003500	105204	000003			CMP	EIMAGE,R4	:AT THE END ?
174	003502	106304	001232			BNE	LKIP29	:NOPE - CARRY ON
175	003504	053507				MOV	#IMAGE,R4	:POINT TO START
176	003505	104204	007321		LKIP29:	DUBINC	CURPBN	:DO THE INCREMENT
177	003507					TST	PCNT	:DONE WITH THIS BLOCK OF PBNS?
178	003514	115000	001475			BNE	LKIP10	:IF NOT DONE SKIP
179	003516	053520				CALL	FCPC	:ELSE PAGE IN NEW FCT BLOCK
180	003517	023573			LKIP10:	DEC	SECCNT	:DECREMENT IT
181	003520	117400	001451			BNE	LKIP27	:NO - DO NEXT BLOCK
182	003522	053273						
183								
184	003523	102200	000004	001221	LKIP28:	BIT	#RPRIM,FLAG1	:NEED GOOD EDC ?
185	003526	013540				BEQ	LKIP7	:NOPE
186	003527	103200	000004	001221		BIC	#RPRIM,FLAG1	:CLEAR FLAG
187	003532					POP	R4	:GET POINTER TO FIRST RBN ENTRY
188	003533	104203	005567			MOV	#GDBLK,R3	:GET GOOD EDC BLOCK
189	003535	100643	000000			MOV	R3,FT.BUF(R4)	:STORE IT IN BUFFER POINTER
190	003537	003541				BR	LKIP33	:SKIP POP
191								
192	003540				LKIP7:	POP	R4	:POP STACK (RBN RENTRY ADDRESS)
193	003541	104304	001320		LKIP33:	MOV	IMSTAR,R4	:POINT TO FIRST TO FORMAT ENTRY
194	003543	104303	001232			MOV	EIMAGE,R3	:GET END ADDRESS
195	003545	107203	000003			SUB	#IMLEN,R3	:POINT TO FLAG OF LAST ENTRY
196	003547	106043				CMP	R4,R3	:FIRST = LAST ?
197	003550	013556				BEQ	LKIP30	:NO - SKIP SPECIAL STUFF
198	003551	104135				MOV	(R3),R5	:GET FLAG WORD
199	003552	101205	040000			BIS	#RECIR,R5	:SET RECIRCULATION FLAG
200	003554	100135				MOV	R5,(R3)	:STORE IT BACK
201	003555	003561				BR	LKIP31	:SKIP KLUDGE FIX TO UDA
202	003556	101200	040000	001320	LKIP30:	BIS	#RECIR,IMSTAR	:SET BIT IN POINTER
203	003561	104143			LKIP31:	MOV	(R4),R3	:GET BUFF POINTER
204	003562	101203	100000			BIS	#LAST,R3	:SIGNAL AS LAST
205	003564	100143				MOV	R3,(R4)	:STORE IT BACK
206	003565	104204	001124			MOV	#HOLDPN,R4	:FOR DECREMENT
207	003567	104203	001130			MOV	#SECTRK,R3	:DITTO
208	003571	021521				CALL	DSUB	:SUBTRAC TO GET NEXT TRACK
209	003572	000000				RETURN		

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 58
L/RBN COMPUTE OVERLAY (G8)

1					.SBTTL L/RBN COMPUTE OVERLAY (G8)	
2						
3						
4					THIS OVERLAY COMPUTES LBN AND RBN OF THE LAST TRACK ON LAST LBN CYLINDER	
5					AND COMPUTES THE PBN OF THAT LBN	
6	003615				DMOVLY G8,START	
7						
8						
9	003023	104200	000052	001154	MOV #G8,CUROVL	;FOR RECORDING
10	003026	104207	001053		MOV #SCR,RO	;POINT TO CHARACTERISTICS BLOCK
11	003030	023060			CALL NUMLBN	;GET NUMBER OF FIRST LBN ON LAST LBN CYL
12	003031	104140	001114		MOV (R4),CURBN	;GET LOW ORDER
13	003033	104140	001120		MOV (R4),HOLDBN	;SAVE FOR LATER
14	003035	104640	000001	001115	MOV 1(R4),CURBN+1	;GET HIGH ORDER
15	003040	104640	000001	001121	MOV 1(R4),HOLDBN+1	;SAVE FOR LATER
16	003043	023120			CALL NUMRBN	;GET NUM OF FIRST RBN ON LAST LBN CYLINDER
17	003044	104140	001107		MOV (R4),CURRBN	;GET LOW ORDER
18	003046	104640	000001	001110	MOV 1(R4),CURRBN+1	;GET HIGH ORDER
19	003051	104140	001122		MOV (R4),HOLRBN	;SAVE LOW FOR LATER
20	003053	104640	000001	001123	MOV 1(R4),HOLRBN+1	;SAVE HIGH FOR LATER
21	003056	023100			CALL LPBN	;GET PBN OF FIRST SECTOR ON LAST TRACK
22	003057	000000			RETURN	

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 59
L/RBN COMPUTE OVERLAY (G8)

1								
2								
3								
4								
5	003060	104300	001144	000736	NUMLBN:	MOV	LBNCYL,TEMP	:GET LOW ORDER NUM OF LBN CYLINDERS
6	003063	104300	001145	000737		MOV	LBNCYL+1,TEMP+1	:GET HIGH ORDER
7	003066	103200	170000	000737		BIC	#HD.CLR,TEMP+1	:CLEAR STARTING CYLINDER BITS
8	003071	104204	000736			MOV	#TEMP,R4	:DITTO
9	003073	104203	001146			MOV	#LBNPCY,R3	:GET LBN'S/CYLINDER
10	003075	021537				CALL	DMUL	:GET FIRST LBN ON LAST CYLINDER
11	003076	104641	000001			MOV	1(R4),R1	:GET LBN
12	003100	105301	001323			ADD	ST.LBN,R1	:ADD STARTING LBN TO GET ABSOLUTE LBN
13	003102	100641	000001			MOV	R1,1(R4)	:STORE BACK
14	003104	104673	000011			MOV	LBNTRK(RO),R3	:GET LBN/TRK
15	003106	103203	177400			BIC	#HIBYTE,R3	:CLERA HIGH BYTE
16	003110	104030	000731			MOV	R3,DDUMMY	:STORE IT
17	003112	114000	000732			CLR	DDUMMY+1	:FOR STORE
18	003114	104203	000731			MOV	#DDUMMY,R3	:LBN/TRACK
19	003116	021521				CALL	DSUB	:WANT LBN ON LAST TRACK
20	003117	000000				RETURN		

UDAFM - UDA FORMATTER DMACH X04.01 23-AUG-82 14:02:32 PAGE 60
L/RBN COMPUTE OVERLAY (G8)

1								
2								
3								
4								
5	003120	104300	001144	000736	NUMRBN:	MOV	LBNCYL,TEMP	:GET LOW ORDER NUMBER OF LBN CYLINDER
6	003123	104300	001145	000737		MOV	LBNCYL+1,TEMP+1	:GET HIGH ORDER
7	003126	103200	170000	000737		BIC	#HD.CLR,TEMP+1	:CLEAR STARTING CYLINDER BITS
8	003131	104204	000736			MOV	#TEMP,R4	:DITTO
9	003133	104203	001150			MOV	#RBNPCY,R3	:GET RBN'S/CYLINDER
10	003135	021537				CALL	DMUL	:GET FIRST RBN ON LAST CYLINDER
11	003136	104641	000001			MOV	1(R4),R1	:GET HIGH ORDER
12	003140	105301	001324			ADD	ST.RBN,R1	:ADD TO GET ABSOLUTE LBN
13	003142	100641	000001			MOV	R1,1(R4)	:STORE BACK
14	003144	104673	000004			MOV	RBNTRK(RO),R3	:GET RBN/TRACK
15	003146	103203	177600			BIC	#HI1BYTE,R3	:CLEAR OUT GARBAGE
16	003150	104030	000731			MOV	R3,DDUMMY	:STORE IT
17	003152	114000	000732			CLR	DDUMMY+1	:FOR STORE
18	003154	104203	000731			MOV	#DDUMMY,R3	:WANT LAST TRACK
19	003156	021521				CALL	DSUB	:GET IT
20	003157	000000				RETURN		

1										
2										
3										
4										
5	003160	104300	001114	000736	LPBN:	MOV	CURBN,TEMP		:GET LOW ORDER	
6	003163	104300	001115	000737		MOV	CURBN+1,TEMP+1		:GET HIGH ORDER	
7	003166	104204	000736			MOV	#TEMP,R4		:FOR SUBTRACT	
8	003170	104641	000001			MOV	1(R4),R1		:GET HIGH ORDER	
9	003172	107301	001323			SUB	ST.LBN,R1		:SUB STARTING LBN TO GET ABSOLUTE LBN	
10	003174	100641	000001			MOV	R1,1(R4)		:STORE BACK	
11	003176	104673	000011			MOV	LBNTRK(R0),R3		:GET LBN'S/TRACK	
12	003200	103203	177400			BIC	#HIBYTE,R3		:CLEAR HIGH WORD	
13	003202	104030	000731			MOV	R3,DDUMMY		:STORE FOR COMPUTATION	
14	003204	114000	000732			CLR	DDUMMY+1		:CLEAR FOR STORE	
15	003206	104203	000731			MOV	#DDUMMY,R3		:FOR DIVIDE	
16	003210	021565				CALL	DDIV		:GET NUMBER OF TRACKS	
17	003211	104673	000004			MOV	RBNTRK(R0),R3		:GET PBN'S/TRACK	
18	003213	103203	177600			BIC	#HI1BYTE,R3		:CLEAR GARBAGE	
19	003215	104030	000731			MOV	R3,DDUMMY		:FOR COMPUTATION	
20	003217	114000	000732			CLR	DDUMMY+1		:CLEAR HIGH WORD	
21	003221	104203	000731			MOV	#DDUMMY,R3		:FOR MULTIPLY	
22	003223	021537				CALL	DMUL		:GET NUMBER OF RBN'S	
23	003224	104300	001114	000731		MOV	CURBN,DDUMMY		:GET LOW ORDER CURRENT BLOCK NUMBER	
24	003227	104300	001115	000732		MOV	CURBN+1,DDUMMY+1		:GET HIGH ORDER	
25	003232	107300	001323	000732		SUB	ST.LBN,DDUMMY+1		:SUBTRACT STARTING	
26	003235	104203	000731			MOV	#DDUMMY,R3		:FOR ADD	
27	003237	021503				CALL	DADD		:ADD TO GET PBN	
28	003240	104140	001124			MOV	(R4),HOLDPN		:GET LOW ORDER	
29	003242	104640	000001	001125		MOV	1(R4),HOLDPN+1		:STORE HIGH ORDER	
30	003245	000000				RETURN				

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 62-1
 FCT DOWN-LINE LOAD OVERLAY (F3)

58	003161	053201			BNE	LOOPP		:NO - SKIP THIS STUFF
59	003162	104204	001302		MOV	#DATE,R4		:POINT TO SERIAL NUMBER
60	003164	104203	004535		MOV	#RDBUF,R3		:POINT TO BUFFER
61	003166	114005			CLR	R5		:FOR MEDIA FORMAT UPDATE
62	003167	100135			MOV	R5,(R3)		:SET FORMAT IN PROGRESS
63	003170	105203	000012		ADD	#FDAT,R3		:POINT TO ENTRY IN FCT BLOCK
64	003172	104205	000004		MOV	#4,R5		:INIT COUNTER
65	003174	104242		10\$:	MOV	(R4)+,R2		:GET WORD
66	003175	100232			MOV	R2,(R3)+		:STORE WORD
67	003176	117405			DEC	R5		:DECREMENT COUNTER
68	003177	053174			BNE	10\$:CONTINUE TILL DONE
69	003200	114005			CLR	R5		:CLEAR R5 (ERROR COUNTER)
70	003201	104202	004535		LOOPP:	MOV	#RDBUF,R2	:POINT TO BUFFER
71	003203	022600			CALL	EDC		:COMPUTE EDC
72	003204	100623	000400		MOV	R3,RW.EDC(R2)		:STORE IT
73	003206	104300	001114	000736	MOV	CURBN,TEMP		:GET LOW ORDER
74	003211	104300	001115	000737	MOV	CURBN+1,TEMP+1		:GET HIGH ORDER
75	003214	104204	000736		MOV	#TEMP,R4		:FOR SUB
76	003216	104641	000001		MOV	1(R4),R1		:GET HIGH ORDER
77	003220	107301	001325		SUB	ST.XBN,R1		:SUBTRACT STARTING XBN
78	003222	100641	000001		MOV	R1,1(R4)		:STORE BACK
79	003224	022665			CALL	CVISK		:CONVERT AND SEEK
80	003225	104207	000721		MOV	#WRBLK,R0		:POINT TO COMMAND BLOCK
81	003227	104203	122400		MOV	#WRCMD,R3		:GET WRITE COMMAND
82	003231	104302	001113		MOV	CURTRK,R2		:GET CURRENT TRACK
83	003233	101023			BIS	R2,R3		:SET TRACK FOR WRITE
84	003234	100673	000004		MOV	R3,RW.CMD(R0)		:STORE IN COMMAND BLOCK
85	003236	104203	004535		MOV	#RDBUF,R3		:POINT TO BUFFER
86	003240	100673	000001		MOV	R3,RW.BUF(R0)		:STICK IN COMMAND BLOCK
87	003242	104303	001114		MOV	CURBN,R3		:GET LOW ORDER HEADER
88	003244	100673	000002		MOV	R3,RW.LOW(R0)		:STORE IN WRITE BLOCK
89	003246	104303	001115		MOV	CURBN+1,R3		:GET HIGH ORDER
90	003250	101203	120000		BIS	#HD.XBN,R3		:SET HEADER
91	003252	100673	000003		MOV	R3,RW.HI(R0)		:STORE IN WRITE BLOCK
92	003254	104203	000726		MOV	#HSLIM-1,R3		:GET DUMMY SDI POINTER
93	003256	100673	000005		MOV	R3,RW.DUR(R0)		:POINT IN COMMAND BLOCK
94	003260	104303	001321		WRITE1:	MOV	HPREA,R3	:GET HEADER PREAMBLE
95	003262	104304	001322		MOV	DPREA,R4		:GET DATA PREAMBLE
96	003264	104302	000740		MOV	UNIT,R2		:SET UNIT
97	003266	104207	000721		MOV	#WRBLK,R0		:MAKE SURE POINTING AT BLOCK
98	003270	101207	100000		BIS	#BIT15,R0		:SET NO REVECTORING
99	003272	060012			XFC	SIP		:WAIT FOR SECTOR PULSE
100	003273	060003			XFC	WRITE		:WRITE SECTOR
101	003274	115001			TST	R1		:ANY ERROR ?
102	003275	013322			BEQ	NO		:NOPE
103	003276	106300	001477	001501	CMP	RETRY,IMPTRY		:MAX ?
104	003301	013305			BEQ	1\$:YES - TRY SOME RECOVERY
105	003302	115400	001501		INC	IMPTRY		:INC RETRY COUNT
106	003304	003260			BR	WRITE1		:DO RETRY
107	003305	104303	001502	1\$:	MOV	RECTMP,R3		:GET CURRENT ERROR RECOVERY LEVEL
108	003307	073321			BMI	2\$:IF NEGATIVE THEN FRIED
109	003310	115000	001500		TST	RECOV		:IS THERE ONLY RECOVERY LEVEL 0 ?
110	003312	013314			BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY
111	003313	022571			CALL	ERRHND		:TRY RECOVERY
112	003314	114000	001501	3\$:	CLR	IMPTRY		:FOR INIT
113	003316	117400	001502		DEC	RECTMP		:DECREMENT IT
114	003320	003260			BR	WRITE1		:RETRY

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 63
 RCT UPDATE OVERLAY (F4)

1					.SBTTL RCT UPDATE OVERLAY (F4)	
2	003444				DMOVL F4,START	
3						
4						
5						
6						
7						
8						
9						
10						
11	003023	104200	000011	001154	RCTUPD: MOV #F4,CUROVL	:GET OVERLAY
12	003026	104303	001113		MOV CURTRK,R3	:GET CURRENT TRACK
13	003030				PUSH R3	:SAVE FOR RESTORE
14	003031	104303	001126		MOV CYLNUM,R3	:GET LOW ORDRE CYLINDER
15	003033				PUSH R3	:SAVE FOR RESTORE
16	003034	104303	001127		MOV CYLNUM+1,R3	:GET HIGH ORDER
17	003036				PUSH R3	:SAVE FOR RESTORE
18	003037	104300	001134	001230	MOV LBNLBN,HOLD	:GET LOW ORDER COUNT OF LBN'S
19	003042	104300	001135	001231	MOV LBNLBN+1,HOLD+1	:GET HIGH ORDER
20	003045	104203	001472		MOV #TOTRCT,R3	:FOR SUBTRACT
21	003047	104204	001230		MOV #HOLD,R4	:DITTC
22	003051	021521			CALL DSUB	:GET STARTING RCT LBN
23	003052	104300	001225	001471	MOV ERRBUF,UPDPNT	:POINT TO ERROR BUFFER
24	003055	104302	001471		MOV UPDPNT,R2	:GET POINTER TO BAD LIST
25	003057	104120	000731		MOV (R2),DDUMMY	:GET LOW ORDER
26	003061	104620	000001	000732	MOV 1(R2),DDUMMY+1	:GET HIGH ORDER
27	003064	102200	100000	000732	BIT #BIT15,DDUMMY+1	:IS IT AN RBN ??
28	003067	013072			BEQ ROVER1	:NO - REGULAR HASH
29	003070	104201	177777		MOV #-1,R1	:HASH FOR RBN
30	003072	103200	170000	000732	ROVER1: BIC #HD.CLR,DDUMMY+1	:CLEAR THE HEADER
31	003075	104204	000731		MOV #DDUMMY,R4	:FOR HASH
32	003077	023333			CALL UHASH	:FIND THE RCT ENTRY FOR CURRENT ERR BLOCK
33	003100	104143			MOV (R4),R3	:GET BLOCK NUMBER
34	003101	105203	000002		ADD #2,R3	:ADD TO GET PAST FIRST 2 BLOCKS
35	003103	100143			MOV R3,(R4)	:STORE BACK
36	003104	104030	001474		MOV R3,RCTCNT	:FOR LATER PING-PONG
37	003106	104203	001230		MOV #HOLD,R3	:FOR ADD
38	003110	021503			CALL DADD	:TO GET LBN OF RCT BLOCK
39	003111	104040	001254		MOV R4,BUFPNT	:STORE POINTER TO BLOCK NUMBER
40	003113	104201	000055		MOV #H1,R1	:RCT READ OVERLAY
41	003115	022435			CALL PAGE	:DO IT
42	003116	104205	006621		MOV #RCTBUF,R5	:POINT TO BUFFER
43	003120	104303	000736		MOV OFFSET,R3	:GET OFFSET
44	003122	105035			ADD R3,R5	:POINT TO HIT ENTRY
45	003123	104302	001471		MOV UPDPNT,R2	:RESTORE POINTER
46	003125	104623	000001		MOV 1(R2),R3	:GET THE HEADER
47	003127	103203	007777		BIC #LO,R3	:CLEAR ALL BUT HEADER
48	003131	106203	110000		CMR #HD.BAD,R3	:IS IT A BAD RBN ?
49	003133	053202			BNE NOTR	:NOPE - CHECK FOR PRIMARY
50	003134	104650	000000	000733	MOV 0(R5),TEMP2	:GET LOW ORDER CURRENT RESIDENT
51	003137	104650	000001	000734	MOV 1(R5),TEMP2+1	:GET HIGH ORDER
52	003142	103203	170000		BIC #HD.CLR,R3	:CLEAR HEADER
53	003144	101203	040000		BIS #RC.UNU,R3	:MARK AS UNUSABLE
54	003146	103203	007777		BIC #LO,R3	:CLEAR LOW ORDER
55	003150	100653	000001		MOV R3,1(R5)	:STORE IT BACK
56	003152	114003			CLR R3	:CLEAR FOR STORE
57	003153	100153			MOV R3,(R5)	:CLEAR LOW ORDER

THIS OVERLAY UPDATES THE RCT WITH THE SECTORS
 IN THE ERBUF BUFFER

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 63-1
 RCT UPDATE OVERLAY (F4)

58	003154	102200	020000	000734	BIT	#BIT13,TEMP2+1	:ANY THING DISPLACED ???	
59	003157	013276			REQ	BOTTOM	:NO - NO NEED TO PING-PONG	
60	003160	023375			CALL	RCTWT	:WRITE UT BLOCK	
61	003161	023571			CALL	PNGPG	:FIND IT A NEW HOME	
62	003162	104204	006621		MOV	#RCTBUF,R4	:POINT TO BUFFER	
63	003164	105054			ADD	R5,R4	:ADD OFFSET	
64	003165	104202	000733		MOV	#TEMP2,R2	:POINT TO OLD RESIDENT	
65	003167	104123			MOV	(R2),R3	:GET LOW ORDER	
66	003170	100143			MOV	R3,(R4)	:PUT IT IN	
67	003171	104623	000001		MOV	1(R2),R3	:GET HIGH ORDER	
68	003173	103203	170000		BIC	#HD.CLR,R3	:CLEAR HEADER	
69	003175	101203	030000		BIS	#RC.SND,R3	:MARK AS SECONDARY	
70	003177	100643	000001		MOV	R3,1(R4)	:STORE IT	
71	003201	003276			BR	BOTTOM	:GO TO BOTTOM OF LOOP	
72	003202	106203	050000	NOTR:	CMP	#HD.PRIV,R3	:PRIMARY REVECTOR ??	
73	003204	053256			BNE	SECNDY	:NO - TREAT AS SECONDARY	
74	003205	104653	000001		MOV	1(R5),R3	:GET RCT HEADER	
75	003207	103203	007777		BIC	#LO,R3	:CLEAR ALL BIT HEADER	
76	003211	106203	000000		CMP	#RC.FRE,R3	:IS IT FREE ??	
77	003213	053231			BNE	SWAP	:NO - SWAP ENTRIES	
78	003214	104123			MOV	(R2),R3	:GET LOW BLOCK NUMBER	
79	003215	100153			MOV	R3,(R5)	:STORE IN RCT	
80	003216	104623	000001		MOV	1(R2),R3	:GET HIGH ORDER	
81	003220	107303	001323		SUB	ST.LBN,R3	:SUBTRACT STARTING LBN BITS	
82	003222	103203	170000		BIC	#HD.CLR,R3	:CLEAR HEADER	
83	003224	101203	020000		BIS	#RC.PRIV,R3	:SIGNAL PRIMARY REVECTOR IN RCT	
84	003226	100653	000001		MOV	R3,1(R5)	:STORE IN RCT	
85	003230	003276			BR	BOTTOM	:GO TO BOTTOM OF LOOP	
86	003231	104650	000000	000733	SWAP:	MOV	0(R5),TEMP2	:GET LOW ORDER CURRENT RESIDENT
87	003234	104650	000001	000734		MOV	1(R5),TEMP2+1	:GET HIGH ORDER CURRENT RESIDENT
88	003237	104123			MOV	(R2),R3	:GET LOW ORDER NEW RESIDENT	
89	003240	100153			MOV	R3,(R5)	:PUT IN RCT	
90	003241	104623	000001		MOV	1(R2),R3	:GET HIGH ORDER NEW RESIDENT	
91	003243	107303	001323		SUB	ST.LBN,R3	:SUBTRACT STARTING LBN BITS	
92	003245	103203	170000		BIC	#HD.CLR,R3	:CLEAR THE HEADER	
93	003247	101203	020000		BIS	#RC.PRIV,R3	:SET AS PRIMARY	
94	003251	100653	000001		MOV	R3,1(R5)	:PUT IN RCT	
95	003253	023375			CALL	RCTWT	:WRITE OUT PRIMARY BLOCK	
96	003254	104202	000733		MOV	#TEMP2,R2	:POINT TO OLD RESIDENT	
97	003256	023571			CALL	PNGPG	:FIND RCT ENTRY FOR SECONDARY	
98	003257	104204	006621		MOV	#RCTBUF,R4	:POINT TO BUFFER	
99	003261	105054			ADD	R5,R4	:ADD OFFSET	
100	003262	104123			MOV	(R2),R3	:GET LOW ORDER NEW ENTRY	
101	003263	100143			MOV	R3,(R4)	:PUT IN RCT	
102	003264	104623	000001		MOV	1(R2),R3	:GET HIGH ORDER NEW ENTRY	
103	003266	107303	001323		SUB	ST.LBN,R3	:SUBTRACT STARTING LBN BITS	
104	003270	103203	170000		BIC	#HD.CLR,R3	:CLEAR HEADER	
105	003272	101203	030000		BIS	#RC.SND,R3	:FLAG AS SECONDARY	
106	003274	100643	000001		MOV	R3,1(R4)	:STORE IN RCT	
107	003276	023375			CALL	RCTWT	:WRITE OUT RCT BLOCK	
108	003277	105200	000002	001471	ADD	#ERLEN,UPDPNT	:POINT TO NEXT ERROR SECTR	
109	003302	117400	001255		DEC	REVCNT	:DECREMENT IT	
110	003304	053055			BNE	ROVER	:NOT DONE - DO NEXT SECTOR	
111	003305				POP	R3	:GET HIGH ORDER CYL	
112	003306	104030	001127		MOV	R3,CYLNUM+1	:RESTORE IT	
113	003310	104030	001100		MOV	R3,ISEEK+2	:PUT IN SEEK COMMAND	
114	003312				POP	R3	:GET LOW ORDER	

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 63-2
 RCT UPDATE OVERLAY (F4)

115	003313	104030	001126		MOV	R3,CYLNUM	
116	003315	104030	001077		MOV	R3,ISEEK+1	:PUT IN SEEK COMMAND
117	003317				POP	R3	:GET TRACK NUMBER
118	003320	104030	001113		MOV	R3,CURTRK	:RESTORE IT
119	003322	102200	040000	001220	BIT	#FINI,FLAG	:DO THE SEEK ?
120	003325	053332			BNE	NOSEK	:NOPE
121	003326	104300	001460	001101	MOV	CURGRP,ISEEK+3	:RESTORE GROUP TO SEEK
122	003331	022242			CALL	SEEK	:GET BACK TO RIGHT CYLINDER
123	003332	000000			NOSEK:	RETURN	

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 64
 RCT UPDATE OVERLAY (F4)

```

1
2
3
4 003333 115001
5 003334 073364
6 003335 104207 001053
7 003337 022730
8 003340 104200 000200 000736 UHKIP:
9 003343 114000 000737
10 003345 104300 001152 000731
11 003350 104300 001153 000732
12 003353 104204 000731
13 003355 104203 000736
14 003357 021565
15
16 003360 104131
17 003361 105011
18 003362 100131
19 003363 000000
20 003364 104140 001152 UHKIP1:
21 003366 104640 000001 001153
22 003371 107300 001324 001153
23 003374 003340
  
```

: COMPUTE RCT ADDRESS FOR GIVEN LBN

```

UHASH: TST R1 ;NEED TO COMPUTE PRIMARY RBN ?
BMI UHKIP1 ;NO - SKIP IT
MOV #SCR,R0 ;POINT TO CUBUNIT CHARACTERISTICS
CALL PRIMRB ;COMPUTE PRIMARY RBN
MOV #128,TEMP ;DIVIDE BY 128 TO GET BLOCK NUMBER
CLR TEMP+1 ;FOR STORE
MOV REVRBN,DDUMMY ;GET PRIMARY RBN
MOV REVRBN+1,DDUMMY+1 ;GET HIGH ORDER
MOV #DDUMMY,R4 ;FOR DIVIDE
MOV #TEMP,R3 ;DITTO
CALL DDIV ;DDUMMY=RCT BLOCK NUMBER
;TEMP=OFFSET
;GET OFFSET
;MULTIPLY BY 2
;STORE BACK
RETURN
MOV (R4),REVRBN ;FOR DIVIDE SETUP
MOV 1(R4),REVRBN+1 ;DITTO
SUB ST.RBN,REVRBN+1 ;SUBTRACT STARTING RBN BITS
BR UHKIP ;DO DIVIDE
  
```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 65
 RCT UPDATE OVERLAY (F4)

1									
2									
3									
4	003375	114005							
5	003376	104050	001246						
6	003400	104204	000731						
7	003402	104203	001053						
8	003404	104632	000011						
9	003406	103202	177400						
10	003410	104207	001424						
11	003412	100672	000004						
12	003414	104632	000001						
13	003416	103202	007777						
14	003420	100672	000001						
15	003422	114002							
16	003423	100672	000000						
17	003425	022665							
18	003426	104207	000721						
19	003430	104203	122400						
20	003432	104302	001113						
21	003434	101023							
22	003435	100673	000004						
23	003437	104202	006621						
24	003441	022600							
25	003442	100623	000400						
26	003444	100672	000001						
27	003446	104143							
28	003447	100673	000002						
29	003451	104643	000001						
30	003453	105303	001323						
31	003455	101203	000000						
32	003457	100673	000003						
33	003461	104203	000726						
34	003463	100673	000005						
35	003465	104303	001321						
36	003467	104304	001322						
37	003471	104302	000740						
38	003473	104207	000721						
39	003475	060012							
40	003476	060003							
41	003477	115001							
42	003500	013525							
43	003501	106300	001477	001501					
44	003504	013510							
45	003505	115400	001501						
46	003507	003465							
47	003510	104303	001502						
48	003512	073524							
49	003513	115000	001500						
50	003515	013517							
51	003516	022571							
52	003517	114000	001501						
53	003521	117400	001502						
54	003523	003465							
55	003524								
56	003524	115405							
57	003525	115400	001246						

```

WRITE AN RCT BLOCK
:
:
:
RCTWT: CLR R5 :CLEAR ERROR COUNTER
:MOV R5,NEXT1 :RESET NEXT COUNTER
:MOV #DDUMMY,R4 :POINT TO BLOCK
RCTWLP: MOV #SCR,R3 :POINT TO CHARACTERISTICS
:MOV LBNTRK(R3),R2 :GET LBN/TRACK
:BIC #HIBYTE,R2 :CLEAR REST OF WORD
:MOV #CONBLK,R0 :POINT TO CONVERT BLOCK
:MOV R2,V3(R0) :FOR CONVERT
:MOV STCYL(R3),R2 :STARTING CLYLINDER
:BIC #LO,R2 :CLEAR REST OF WORD
:MOV R2,V1+1(R0) :STORE
:CLR R2 :CLEAR FOR STORE
:MOV R2,V1(R0) :LOW ORDER ALWAYS 0
:CALL CVTSK :CONVERT AND SEEK
:MOV #WRBLK,R0 :POINT TO COMMAND BLOCK
:MOV #WRCMD,R3 :GET WRITE COMMAND
:MOV CURTRK,R2 :GET CURRENT TRACK
:BIS R2,R3 :SET TRACK FOR WRITE
:MOV R3,RW.CMD(R0) :STORE IN COMMAND BLOCK
:MOV #RCTBUF,R2 :POINT TO BUFFER
:CALL CEDC :COMPUTE EDC - RETURNED IN R3
:MOV R3,RW.EDC(R2) :STORE IT
:MOV R2,RW.BUF(R0) :STICK IN COMMAND BLOCK
:MOV (R4),R3 :GET LOW ORDER HEADER
:MOV R3,RW.LOW(R0) :STORE IN WRITE BLOCK
:MOV 1(R4),R3 :GET HIGH ORDER
:ADD ST.LBN,R3 :ADD STARTING LBN BITS
:BIS #HD.LBN,R3 :SET HEADER
:MOV R3,RW.HI(R0) :STORE IN WRITE BLOCK
:MOV #HSLIM-1,R3 :GET DUMMY SDI POINTER
:MOV R3,RW.DUM(R0) :POINT IN COMMAND BLOCK
WRITE2: MOV HPREA,R3 :GET HEADER PREAMBLE
:MOV DPREA,R4 :GET DATA PREAMBLE
:MOV UNIT,R2 :SET UNIT
:MOV #WRBLK,R0 :MAKE SURE POINTING AT BLOCK
:XFC SIP :WAIT FOR SECTOR PULSE
:XFC WRITE :WRITE SECTOR
:TST R1 :ANY ERROR ?
:BEQ RWGD :NOPE
:CMP RETRY,IMPTRY :MAX ?
:BEQ 1$ :YES - TRY SOME RECOVERY
:INC IMPTRY :INC RETRY COUNT
:BR WRITE2 :DO RETRY
:MOV RECTMP,R3 :GET CURRENT ERROR RECOVERY LEVEL
:1$: BMI 2$ :IF NEGATIVE THEN FRIED
:TST RECOV :IS THERE ONLY RECOVERY LEVEL 0 ?
:BEQ 3$ :YES - NO NEED TO ISSUE IT - JUST RETRY
:CALL ERRHND :TRY RECOVERY
:CLR IMPTRY :FOR INIT
:DEC RECTMP :DECREMENT IT
:BR WRITE2 :RETRY
:2$:
:3$:
:INC R5 :YUP - INCREMENT COUNTER
RWGD: INC NEXT1 :INCREMENT IT

```

UDAFM - UCA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 65-1
 RCT UPDATE OVERLAY (F4)

58	003527	114000	001501		CLR	TMPTRY		:FOR RESET
59	003531	104300	001500	001502	MOV	RECOV,RECTMP		:GET RECOVERY LEVELS
60	003534	104204	000731		MOV	#DDUMMY,R4		:FOR ADD
61	003536	104203	001243		MOV	#RCTFMT,R3		:FOR ADD
62	003540	021503			CALL	DADD		:POINT TO NEXT COPY
63	003541	106300	001245	001246	CMP	FCTCPY,NEXT1		:DONE THIS SECTOR ?
64	003544	053402			BNE	RCTWLP		:NO - WRITE NEXT FCT COPY
65	003545	106305	001245		CMP	FCTCPY,R5		:ERROR ON EVERY WRITE ?
66	003547	013565			BEQ	RCTERR		:YUP - BIG TROUBLE
67	003550	104303	001246		RCXLP: MOV	NEXT1,R3		:ANY REPEATS ?
68	003552	013563			BEQ	RTDON		:NO
69	003553	104204	000731		MOV	#DDUMMY,R4		:TO GET IT BACK
70	003555	104203	001243		MOV	#RCTFMT,R3		:DITTO
71	003557	021521			CALL	DSUB		
72	003560	117400	001246		DEC	NEXT1		:SUB IT
73	003562	003550			BR	RCXLP		:REPEAT
74	003563	060022			RTDON: XFC	UPDATE		:LET HOST KNOW STILL ALIVE
75	003564	000000			RETURN			
76	003565	104012			RCTERR: MOV	R1,R2		:XFC ERROR CODE
77	003566	104201	000017		MOV	#15.,R1		:RCT WRITE ERROR
78	003570	022542			CALL	ERRMNT		:ERROR QUIT

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 66
 RCT UPDATE OVERLAY (F4)

UD
FC

```

1
2
3
4 003571          :
5 003572 114000 001223  PNGPG: PUSH      R2          :SAVE R2
6 003574 114002          :          CLR      WRFLG       :CLEAR WRAP FLAG
7 003575 104303 000736  XNGBLK: CLR      R2          :FOR FLOP SET
8 003577 104035          :          MOV      OFFSET,R3    :GET OFFSET
9 003600 105025          :          MOV      R3,R5       :MOV OFFSET INTO BUFF POINTER
10 003601 102205 000400  :          ADD      R2,R5       :ADD FLOP VALUE
11 003603 053616          :          BIT      #BIT8,R5    :PAST ONE END (OR BOTH)
12 003604 104651 006622  :          BNE     XFLIP       :YUP - FLIP OTHER DIRECTION
13 003606 103201 007777  :          MOV      RCTBUF+1(R5),R1 :GET HEADER CODE
14 003610 106201 100000  :          BIC      #LO,R1     :CLEAR LOW ORDER
15 003612 013654          :          CMP      #RC.NUL,R1  :END OF RCT ?
16 003613 106201 000000  :          BEQ     XEORCT      :YUP - WRAP TO FIRST BLOCK
17 003615 013666          :          CMP      #RC.FRE,R1  :FREE ?
18 003616 104025          :          BEQ     XPRET       :YUP - ALL DONE
19 003617 114002          :          MOV      R2,R5       :GET FLIP VALUE
20 003620 107052          :          CLR      R2
21 003621 073624          :          SUB      R5,R2       :NEGATE IT
22 003622 105202 000002  :          BMI     XNOINC      :IF NEGATIVE DON'T INC
23 003624 106202 000400  :          ADD      #2,R2       :ADD TO NEXT VALUE
24 003626 053577          :          XNOINC: CMP      #SECS16,R2 :DONE EVERY SLOT IN BLOCK ?
25                   :          BNE     XAGAIN     :NOPE - TRY NEXT ONE
26                   :
27                   : IN THIS SECTION THE BLOCKS ARE PING-PONGED BUT
28                   : THE SEARCH WITHIN BLOCKS IS LINEAR FROM HIGHEST BUFFER
29                   : ADDRESS TO LOWEST
30 003627 115400 001474  :
31 003631 104303 001474  :          INC      RCTCNT     :INC TO NEXT ONE
32 003633 104204 000731  :          XPNGRD: MOV      RCTCNT,R3    :FOR STORE
33 003635 100143          :          MOV      #DDUMMY,R4  :FOR ADD
34 003636 114003          :          MOV      R3,(R4)     :STORE BLOCK NUMBER
35 003637 100643 000001  :          CLR      R3         :FOR RESETS
36 003641 104030 000736  :          MOV      R3,1(R4)    :CLEAR HIGH WORD
37 003643 104203 001230  :          MOV      R3,OFFSET   :MAKE OFFSET AT BEGINNING
38 003645 021503          :          MOV      #HOLD,R3    :POINT TO FIRST RCT LBN
39 003646 104040 001254  :          CALL     DADD        :GET LBN OF THIS RCT BLOCK
40 003650 104201 000055  :          MOV      R4,BUFPT    :STJRE BLOCK NUMBER
41 003652 022435          :          MOV      #H1,R1     :READ RCT OVERLAY
42 003653 003574          :          CALL     PAGE       :DO IT
43 003654 104303 001223  :          BR      XNGBLK     :SEARCH THIS BLOCK
44 003656 053670          :          XEORCT: MOV      WRFLG,R3  :GET WRAP FLAG
45 003657 104200 000002 001474 :          BNE     XPERR       :IF BEEN HERE ONCE THEN RCT FULL
46 003662 104200 000002 001223 :          MOV      #2,RCTCNT   :FOR FIRST RCT BLOCK
47 003665 003631          :          MOV      #2,WRFLG   :MAKE WRAP FLAG NON-ZERO
48                   :          BR      XPNGRD     :READ IT AND CONTINUE
49                   :
50 003666          :          XPRET: POP      R2          :RESTORE R2
51 003667 000000          :          RETURN          :SUCCESSFUL RETURN
52 003670 104201 000020  :          XPERR: MOV      #16.,R1 :RCT FULL
53 003672 114002          :          CLR      R2         :NO SUBCODE
54 003673 022542          :          CALL     ERRMNT     :ERROR RETURN

```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 67
 RCT READ OVERLAY (H1)

1					.SBTTL RCT READ OVERLAY (H1)	
2	003674				DMOVLY H1,START	
3				:		
4				:	READ A BLOCK OF THE RCT	
5				:		
6	003023				PUSHA	
7	003031	104200	000055	001154	MOV #H1,CUROVL	:FOR INIT
8	003034	104304	001254		MOV BUFPT,R4	:GET POINTER TO BLOCK NUMBER
9	003036	114005			CLR R5	:CLEAR ERROR COUNTER
10	003037	104203	001053		MOV #SCR,R3	:POINT TO CHARACTERISTICS
11	003041	104632	000001		MOV STCYL(R3),R2	:GET STARTING CYLINDER
12	003043	103202	007777		BIC #LO,R2	:CLEAR REST OF WORD
13	003045	104207	001424		MOV #CONBLK,R0	:POINT TO CONVERT BLOCK
14	003047	100672	000001		MOV R2,V1+1(R0)	:STORE FOR CONVERT
15	003051	114002			CLR R2	:FOR STORE
16	003052	100672	000000		MOV R2,V1(R0)	:LOW ORDER ALWAYS 0
17	003054	104632	000011		MOV LBNTRK(R3),R2	:GET LBN/TRK
18	003056	103202	177400		BIC #HIBYTE,R2	:CLEAR HIGH BYTE
19	003060	100672	000004		MOV R2,V3(R0)	:STORE IN CONVERT BLOCK
20	003062				OCLOOP:	
21	003062	022665			CALL CVTSK	:CONVERT RCT BLOCK NUMBER AND SEEK
22	003063	104207	000721		MOV #RDBLK,R0	:PREPARE FOR READ SECTORS
23	003065	104203	000726		MOV #HSLIM-1,R3	:POINTER TO DUMMY SDI BLOCK
24	003067	100673	000005		MOV R3,RW.DUM(R0)	:STORE IN COMMAND BLOCK
25	003071	104143			MOV (R4),R3	:LO ORDER BLOCK NUMBER
26	003072	100673	000002		MOV R3,RW.LOW(R0)	:STORE IN READ CMD BLOCK
27	003074	104643	000001		MOV 1(R4),R3	:HI ORDER BLOCK NUM AND CODE
28	003076	105303	001323		ADD ST.LBN,R3	:ADD STARTING LBN BITS
29	003100	100673	000003		MOV R3,RW.HI(R0)	:STORE IN READ CMD BLOCK
30	003102	104203	006621		MOV #RCTBUF,R3	:LOAD ADDRESS OF DATA BUFFER
31	003104	100673	000001		MOV R3,RW.BUF(R0)	:STORE IN COMMAND BUFFER
32	003106	104203	013400		MOV #RWCMD,R3	:LOAD SDI READ COMMAND
33	003110	104301	001113		MOV CURTRK,R1	:GET CURRENT HEAD NUMBER IN R1
34	003112	101013			BIS R1,R3	:SET IT IN COMMAND
35	003113	100673	000004		MOV R3,RW.CMD(R0)	:STORE BACK
36	003115	104207	000721		READ11: MOV #RDBLK,R0	:MAKE SURE POINTING AT BLOCK
37	003117	104203	100000		MOV #RDCMD,R3	:MARK AS ONLY REQUEST
38	003121	100173			MOV R3,(R0)	:STORE IN CMD BLOCK
39	003122	104302	000740		MOV UNIT,R2	:SDI INTERCONNECT
40	003124	101207	100000		BIS #BIT15,R0	:SET NO REVECTORING
41	003126	060012			XFC SIP	:WAIT FOR PULSE
42	003127	060002			XFC READ	:READ 1 SECTOR
43	003130	115001			TST R1	:ANY ERRORS ?
44	003131	053147			BNE 100\$:YES - TRY RECOVERY
45	003132	104173			MOV (R0),R3	:GET STATUS WORD
46	003133	102203	010000		BIT #ECCF,R3	:ECC ERROR ?
47	003135	013141			BEQ 101\$:NOPE - VERIFY EDC
48	003136	023000			CALL ECCCK	:CORRECT ECC
49	003137	115001			TST R1	:TEST FLAG
50	003140	053147			BNE 100\$:UNCORRECTABLE
51	003141	104202	006621	101\$:	MOV #RCTBUF,R2	:POINT TO BUFFER
52	003143	022600			CALL CEDC	:COMPUTE EDC
53	003144	106623	000400		CMR RW.EDC(R2),R3	:O.K. ?
54	003146	013207			BEQ 102\$:YUP - CONSIDER GOOD
55	003147	106300	001477	001501	100\$:	CMR RETRY,TMPTRY
56	003152	013156			BEQ 1\$:MAX ?
57	003153	115400	001501		INC TMPTRY	:YES - TRY SOME RECOVERY :INC RETRY COUNT

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 67-1
 RCT READ OVERLAY (HI)

58	003155	003115			BR	READ11		:DO RETRY
59	003156	104303	001502	1\$:	MOV	RECTMP,R3		:GET CURRENT ERROR RECOVERY LEVEL
60	003160	073172			BMI	2\$:IF NEGATIVE THEN FRIED
61	003161	115000	001500		TST	RECOV		:IS THERE ONLY RECOVERY LEVEL 0 ?
62	003163	013165			BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY
63	003164	022571			CALL	ERRHND		:TRY RECOVERY
64	003165	114000	001501	3\$:	CLR	TMPTRY		:FOR INIT
65	003167	117400	001502		DEC	RECTMP		:DECREMENT IT
66	003171	003115			BR	READ11		:RETRY
67	003172			2\$:				
68	003172	115405			INC	R5		:INCREMENT BAD COUNTER
69	003173	106305	001245		CMP	FCTCPY,R5		:ALL BAD ?
70	003175	013331			BEQ	ORFTAL		:YUP - ALL OVER
71	003176	104203	001243		MOV	#RCTFMT,R3		:SIZE OF TABLE - R4 -> BLOCK NUMBER
72	003200	021503			CALL	DADD		:ADD TO POINT TO NEXT COPY
73	003201	114000	001501		CLR	TMPTRY		:RESET RETRY COUNT
74	003203	104300	001500	001502	MOV	RECOV,RECTMP		:DITTO RECOVERY LEVELS
75	003206	003062			BR	OCLOOP		:BRANCH BACK
76	003207			102\$:				
77	003207	114000	001501	OCDONE:	CLR	TMPTRY		:FOR RESET
78	003211	104300	001500	001502	MOV	RECOV,RECTMP		:GET RECOVERY LEVELS
79	003214	115005			TST	R5		:ANY ERRORS ?
80	003215	013322			BEQ	RLDONE		:NO - EXIT
81	003216	104203	001241		MOV	#FCTFMT,R3		:SIZE OF TABLE
82	003220	021521			CALL	DSUB		:GET BACK TO PREVIOUS COPY
83	003221	022665			CALL	CVTSK		:CONVERT AND SEEK
84	003222	104207	000721		MOV	#WRBLK,R0		:POINT TO COMMAND BLOCK
85	003224	104203	122400		MOV	#WRCMD,R3		:GET WRITE COMMAND
86	003226	104302	001113		MOV	CURTRK,R2		:GET CURRENT TRACK
87	003230	101023			BIS	R2,R3		:SET TRACK FOR WRITE
88	003231	100673	000004		MOV	R3,RW.CMD(R0)		:STORE IN COMMAND BLOCK
89	003233	104203	006621		MOV	#RCTBUF,R3		:POINT TO BUFFER
90	003235	100673	000001		MOV	R3,RW.BUF(R0)		:STICK IN COMMAND BLOCK
91	003237	104143			MOV	(R4),R3		:GET LOW ORDER HEADER
92	003240	100673	000002		MOV	R3,RW.LOW(R0)		:STORE IN WRITE BLOCK
93	003242	104643	000001		MOV	1(R4),R3		:GET HIGH ORDER
94	003244	105303	001323		ADD	ST.LBN,R3		:ADD STARTING LBN BITS
95	003246	100673	000003		MOV	R3,RW.HI(R0)		:STORE IN WRITE BLOCK
96	003250	104203	000726		MOV	#HSLIM-1,R3		:GET DUMMY SDI POINTER
97	003252	100673	000005		MOV	R3,RW.DUM(R0)		:POINT IN COMMAND BLOCK
98	003254	104303	001321	WRIT12:	MOV	HPREA,R3		:GET HEADER PREAMBLE
99	003256	104304	001322		MOV	DPREA,R4		:GET DATA PREAMBLE
100	003260	104302	000740		MOV	UNIT,R2		:SET UNIT
101	003262	104207	000721		MOV	#WRBLK,R0		:MAKE SURE POINTING AT BLOCK
102	003264	101207	100000		BIS	#BIT15,R0		:SET NO REVECTORING
103	003266	060012			XFC	SIP		:WAIT FOR SECTOR PULSE
104	003267	060003			XFC	WRITE		:WRITE SECTOR
105	003270	115001			TST	R1		:ANY ERROR ?
106	003271	013315			BEQ	2\$:NO - SKIP RETRY
107	003272	106300	001477	001501	CMP	RETRY,TMPTRY		:MAX ?
108	003275	013301			BEQ	1\$:YES - TRY SOME RECOVERY
109	003276	115400	001501		INC	TMPTRY		:INC RETRY COUNT
110	003300	003254			BR	WRIT12		:DO RETRY
111	003301	104303	001502	1\$:	MOV	RECTMP,R2		:GET CURRENT ERROR RECOVERY LEVEL
112	003303	073315			BMI	2\$:IF NEGATIVE THEN FRIED
113	003304	115000	001500		TST	RECOV		:IS THERE ONLY RECOVERY LEVEL 0 ?
114	003306	013310			BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02.32 PAGE 68
RCT READ OVERLAY (H1)

1
2
3
4
5
6
7 003335
8
9
10
11 003023 003166

.....

```
.SBTTL FCT->RCT CONVERSION OVERLAY (F5)
CONVERT FCT INTO RCT

DMOVLY F5,START

JMP START2 , ;SKIP SUBROUTINES
```


UJAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 70
 FCT->RCT CONVERSION OVERLAY (F5)

```

1
2
3
4 003066 114000 001223
5 003070 114002
6 003071 104303 000736
7 003073 104035
8 003074 105025
9 003075 102205 000400
10 003077 053112
11 003100 104651 006622
12 003102 103201 007777
13 003104 106201 100000
14 003106 013147
15 003107 106201 000000
16 003111 013161
17 003112 104025
18 003113 114002
19 003114 107052
20 003115 073120
21 003116 105202 000002
22 003120 106202 000400
23 003122 053073
24
25
26
27
28
29 003123 115400 001474
30 003125 104303 001474
31 003127 104204 000731
32 003131 100143
33 003132 114000 000732
34 003134 114000 000736
35 003136 104203 001230
36 003140 021503
37 003141 104040 001254
38 003143 104201 000055
39 003145 022435
40 003146 003070
41 003147 104303 001223
42 003151 053162
43 003152 104200 000002 001474
44 003155 104200 000002 001223
45 003160 003125
46
47
48 003161 000000
49
50 003162 104201 000020
51 003164 114002
52 003165 022542

:
: SEARCH FOR OPEN ENTRY IN RCT
:
PNGPNG: CLR WRFLG ;CLEAR WRAP FLAG
:FOR FLOP SET
PNGBLK: CLR R2 ;FOR FLOP SET
:GET OFFSET
MOV OFFSET,R3 ;GET OFFSET
PAGAIN: MOV R3,R5 ;MOV OFFSET INTO BUFF POINTER
:ADD FLOP VALUE
ADD R2,R5 ;ADD FLOP VALUE
:
BIT #BIT8,R5 ;PAST ONE END (OR BOTH)
BNE FLIP ;YUP - FLIP OTHER DIRECTION
MOV RCTBUF+1(R5),R1 ;GET HEADER CODE
BIC #LO,R1 ;CLEAR LOW ORDER STUFF
CMP #RC.NUL,R1 ;END OF RCT ?
BEQ EORCT ;YUP - WRAP TO FIRST BLOCK
CMP #RC.FRE,R1 ;FREE ?
BEQ PRET ;YUP - ALL DONE
FLIP: MOV R2,R5 ;GET FLIP VALUE
:
CLR R2
SUB R5,R2 ;NEGATE IT
BMI NOINC ;IF NEGATIVE DON'T INC
ADD #2,R2 ;ADD TO NEXT VALUE
NOINC: CMP #SECSI6,R2 ;DONE EVERY SLOT IN BLOCK ?
BNE PAGAIN ;NOPE - TRY NEXT ONE

:
: IN THIS SECTION THE BLOCKS ARE PING-PONGED BUT
: THE SEARCH WITHIN BLOCKS IS LINEAR FROM HIGHEST BUFFER
: ADDRESS TO LOWEST
:
PNGRD: INC RCTCNT ;INC TO NEXT ONE
:FOR STORE
MOV RCTCNT,R3 ;FOR STORE
MOV #DDUMMY,R4 ;FOR ADD
MOV R3,(R4) ;STORE BLOCK NUMBER
CLR DDUMMY+1 ;FOR RESETS
CLR OFFSET ;MAKE IT AT ZERO
MOV #HOLD,R3 ;POINT TO FIRST RCT LBN
CALL DADD ;GET LBN OF THIS RCT BLOCK
MOV R4,BUFPNT ;STORE POINTER TO BLOCK NUMBER
MOV #H1,R1 ;RCT READ OVERLAY
CALL PAGE ;DO IT
BR PNGBLK ;SEARCH THIS BLOCK
EORCT: MOV WRFLG,R3 ;GET WRAP FLAG
BNE PERR ;IF BEEN HERE ONCE THEN RCT FULL
MOV #2,RCTCNT ;FOR FIRST RCT BLOCK
MOV #2,WRFLG ;MAKE WRAP FLAG NON-ZERO
BR PNGRD ;READ IT AND CONTINUE

:
PRET: RETURN ;SUCCESSFUL RETURN
:R5=OFFSET
PERR: MOV #16.,R1 ;RCT FULL
CLR R2 ;NO SUBCODE
CALL ERRMNT ;ERROR RETURN

```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 71-1
 FCT->RCT CONVERSION OVERLAY (F5)

58	003321	104030	001474		MOV	R3,RCTCNT	:SAVE FOR LATER PNGPNG
59	003323	104203	001230		MOV	#HOLD,R3	:FOR ADD
60	003325	021503			CALL	DADD	:TO GET LBN OF RCT BLOCK
61	003326	104040	001254		MOV	R4,BUFPNT	:STORE POINTER TO BLOCK NUMBER
62	003330	104201	000055		MOV	#H1,R1	:RCT READ OVERLAY
63	003332	022435			CALL	PAGE	:EXECUTE IT
64	003333	104205	006621		MOV	#RCTBUF,R5	:POINT TO BUFFER
65	003335	104303	000736		MOV	OFFSET,R3	:GET OFFSET
66	003337	105035			ADD	R3,R5	:POINT TO ENTRY
67	003340	104653	000001		MOV	1(R5),R3	:GET HIGH ORDER
68	003342	103203	007777		BIC	#LO,R3	:CLEAR ALL BUT HEADER
69	003344	106203	000000		CMF	#RC.FRE,R3	:IS IT FREE ?
70	003346	013637			BEQ	FILLIT	:YES - FILL IT
71	003347	106203	040000		CMF	#RC.UNU,R3	:UNUSABLE RBN ?
72	003351	013616			BEQ	BADRBN	:YES - MUST BE SECONDARY
73	003352	104150	000733		MOV	(R5),TEMP2	:ELSE SWITCH
74	003354	104650	000001	000734	MOV	1(R5),TEMP2+1	:HIGH ORDER
75	003357	104303	001114		MOV	CURBN,R3	:GET NEW RESIDENT LOW ORDER
76	003361	100153			MOV	R3,(R5)	:STORE IN RCT
77	003362	104303	001115		MOV	CURBN+1,R3	:GET HIGH ORDER
78	003364	107303	001323		SUB	ST.LBN,R3	:SUBTRACT STARTING LBN BITS
79	003366	103203	170000		BIC	#HD.CLR,R3	:CLEAR THE HEADER
80	003370	101203	020000		BIS	#RC.PRIV,R3	:MARK AS PRIMARY
81	003372	100653	000001		MOV	R3,1(R5)	:STORE IT
82	003374	102200	000400	001220	BIT	#DLL.FLAG	:DID WE CREATE THE FCT ?
83	003377	013437			BEQ	FCTSKP	:NO - THEN DON'T CHANGE IT
84	003400	104302	001256		MOV	FCTPTR,R2	:GET POINTER TO CURRENT FCT BLOCK POS
85	003402	104623	000001		MOV	1(R2),R3	:GET HIGH ORDER FCT ENTRY
86	003404	101203	100000		BIS	#PRIV,R3	:MAKE IT SECONDARY
87	003406	100623	000001		MOV	R3,1(R2)	:STORE IT BACK
88	003410	106202	004535		CMF	#RDBUF,R2	:IS THIS THE FIRST ENTRY IN THE BLOCK
89	003412	053431			BNE	FCTSK1	:NOPE WE'RE SAFE
90	003413	104300	000731	001116	MOV	DDUMMY,CURXBN	:SAVE RCT BLOCK NUMBER
91	003416	104300	000732	001117	MOV	DDUMMY+1,CURXBN+1	:DITTO
92	003421	023766			CALL	FIXFCT	:YUP - GOT SOME GYRATIONS TO DO
93	003422	104300	001116	000731	MOV	CURXBN,DDUMMY	:RESTORE RCT BLOCK NUMBER
94	003425	104300	001117	000732	MOV	CURXBN+1,DDUMMY+1	:DITTO
95	003430	003437			BR	FCTSKP	:THEN CONTINUE ON
96	003431	107202	000001		FCTSK1: SUB	#1,R2	:POINT BACK ONE
97	003433	104123			MOV	(R2),R3	:GET HIGH ORDER
98	003434	103203	100000		BIC	#PRIV,R3	:CLEAR PRIMARY IF SET
99	003436	100123			MOV	R3,(R2)	:STORE IT
100	003437	024027			FCTSKP: CALL	RCTWRT	:WRITE OUT CHANGED BLOCK
101	003440	023066			CALL	PNGPNG	:FIND IT A NEW HOME
102	003441	104204	006621		MOV	#RCTBUF,R4	:POINT TO BUFFER
103	003443	105054			ADD	R5,R4	:ADD OFFSET
104	003444	104202	000733		MOV	#TEMP2,R2	:POINT TO OLD RESIDENT
105	003446	104123			MOV	(R2),R3	:GET LOW ORDER
106	003447	100143			MOV	R3,(R4)	:PUT IT IN
107	003450	104623	000001		MOV	1(R2),R3	:GET HIGH ORDER
108	003452	103203	170000		BIC	#HD.CLR,R3	:CLEAR HEADER
109	003454	101203	030000		BIS	#RC.SND,R3	:MARK AS SECONDARY
110	003456	100643	000001		MOV	R3,1(R4)	:STORE IT
111	003450	003670			BR	XYZ	:SKIP TO END
112	003461	106203	060000		NOLBN: CMP	#HD.RBN,R3	:BAD RBN ?
113	003463	053671			BNE	XYZ1	:NO - THEN DON'T CARE ABOUT IT
114	003464	104303	001115		MOV	CURBN+1,R3	:GET HEADER

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 71-2
FCT->RCT CONVERSION OVERLAY (F5)

115	003466	103203	170000		BIC	#HD.CLR,R3	:CLEAR IT	
116	003470	104030	001115		MOV	R3,CURBN+1	:STORE IT BACK	
117	003472	104201	177777		MOV	#-1,R1	:SIGNAL RCT BLOCK	
118	003474	104204	001114		MOV	#CURBN,R4	:POINT TO BLOCK NUMBER	
119	003476	023024			CALL	HASH	:GET RCT BLOCK AND OFFSET	
120	003477	104143			MOV	(R4),R3	:GET RCT BLOCK	
121	003500	105203	000002		ADD	#2,R3	:ADD TO GET BY 2 BLOCKS	
122	003502	100143			MOV	R3,(R4)	:STORE BACK	
123	003503	104030	001474		MOV	R3,RCTCNT	:SAVE FOR LATER PNGPNG	
124	003505	104203	001230		MOV	#HOLD,R3	:FOR ADD	
125	003507	021503			CALL	DADD	:TO GET LBN OF RCT BLOCK	
126	003510	104040	001254		MOV	R4,BUFPT	:STORE POINTER TO BLOCK NUMBER	
127	003512	104201	000055		MOV	#H1,R1	:RCT READ OVERLAY	
128	003514	022435			CALL	PAGE	:DO IT	
129	003515	104205	006621		MOV	#RCTBUF,R5	:POINT TO BLOCK	
130	003517	104304	000736		MOV	OFFSET,R4	:GET OFFSET	
131	003521	105045			ADD	R4,R5	:POINT TO ENTRY	
132	003522	104653	000001		MOV	1(R5),R3	:GET HIGH ORDER	
133	003524	103203	007777		BIC	#LO,R3	:CLEAR ALL BUT HEADER	
134	003526	106203	000000		CMF	#RC.FRE,R3	:IS IT FREE ?	
135	003530	053540			BNE	RRPL	:NO - RELOCATE CURRENT RESIDENT	
136	003531	103203	170000		BIC	#HD.CLR,R3	:CLEAR THE HEADER	
137	003533	101203	040000		BIS	#RC.UNU,R3	:MARK AS UNUSABLE	
138	003535	100653	000001		MOV	R3,1(R5)	:STORE IT BACK	
139	003537	003670			BR	XYZ	:BRANCH TO THE END	
140	003540	104650	000000	000733	RRPL:	MOV	0(R5),TEMP2	:GET LOW ORDER CURRENT RESIDENT
141	003543	104650	000001	000734		MOV	1(R5),TEMP2+1	:GET HIGH ORDER
142	003546	103203	170000		BIC	#HD.CLR,R3	:CLEAR HEADER	
143	003550	101203	040000		BIS	#RC.UNU,R3	:MARK AS UNUSABLE	
144	003552	103203	007777		BIC	#LO,R3	:CLEAR LOW ORDER	
145	003554	100653	000001		MOV	R3,1(R5)	:STORE IT BACK	
146	003556	114003			CLR	R3	:CLEAR FOR STORE	
147	003557	100153			MOV	R3,(R5)	:CLEAR LOW ORDER	
148	003560	024027			CALL	RCTWRT	:WRITE UT BLOCK	
149	003561	023066			CALL	PNGPNG	:FIND IT A NEW HOME	
150	003562	104204	006621		MOV	#RCTBUF,R4	:POINT TO BUFFER	
151	003564	105054			ADD	R5,R4	:POINT TO ENTRY	
152	003565	104202	000733		MOV	#TEMP2,R2	:POINT TO OLD RESIDENT	
153	003567	104123			MOV	(R2),R3	:GET LOW ORDER	
154	003570	100143			MOV	R3,(R4)	:PUT IT IN	
155	003571	104623	000001		MOV	1(R2),R3	:GET HIGH ORDER	
156	003573	103203	170000		BIC	#HD.CLR,R3	:CLEAR HEADER	
157	003575	101203	030000		BIS	#RC.SND,R3	:MARK AS SECONDARY	
158	003577	100643	000001		MOV	R3,1(R4)	:STORE IT	
159	003601	102200	000400	001220	BIT	#DLL,FLAG	:DID WE CREATE THE FCT ?	
160	003604	013615			BEQ	FCTSLP	:NO - THEN DON'T CHANGE IT	
161	003605	104302	001256		MOV	FCTPTR,R2	:GET FCT PPOINTER	
162	003607	107202	000001		SUB	#1,R2	:POINT BACK ONE	
163	003611	104123			MOV	(R2),R3	:GET HIGH ORDER	
164	003612	103203	100000		BIC	#PRMY,R3	:CLEAR PRIMARY IF SET	
165	003614	100123			MOV	R3,(R2)	:STORE IT	
166	003615	003670			FCTSLP:	BR	:GO TO END	
167	003616	023066			BADRBN:	CALL	PNGPNG	:FIND A NEW SLOT
168	003617	104204	006621		MOV	#RCTBUF,R4	:POINT TO BUFFER	
169	003621	105054			ADD	R5,R4	:POINT TO ENTRY	
170	003622	104202	001114		MOV	#CURBN,R2	:POINT TO OLD RESIDENT	
171	003624	104123			MOV	(R2),R3	:GET LOW ORDER	

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 71-3
 FCT->RCT CONVERSION OVERLAY (F5)

172	003625	100143		MOV	R3,(R4)		:PUT IT IN
173	003626	104623	06J001	MOV	1(R2),R3		:GET HIGH ORDER
174	003630	103203	170000	BIC	#HD.CLR,R3		:CLEAR HEADER
175	003632	101203	030000	BIS	#RC.SND,R3		:MARK AS SECONDARY
176	003634	100643	000001	MOV	R3,1(R4)		:STORE IT
177	003636	003670		BR	XYZ		:GO TO END
178	003637	104303	001114	FILLIT: MOV	CURBN,R3		:GET LOW ORDER BN
179	003641	100153		MOV	R3,(R5)		:PUT IN RCT
180	003642	104303	001115	MOV	CURBN+1,R3		:GET HIGH ORDER AND HEADER
181	003644	107303	001323	SUB	ST.LBN,R3		:SUBTRACT STARTING LBN BITS
182	003646	103203	170000	BIC	#HD.CLR,R3		:CLEAR HEADER
183	003650	101203	020000	BIS	#RC.PRV,R3		:MARK AS PRIMARY
184	003652	100653	000001	MOV	R3,1(R5)		:STORE IN RCT
185	003654	102200	000400	001220	BIT	#DLL,FLAG	:DID WE CREATE THE FCT ?
186	003657	013670		BEQ	XYZ		:NO - THEN DON'T CHANGE IT
187	003660	104303	001256	MOV	FCTPTR,R3		:GET POINTER TO FCT ENTRY
188	003662	104634	000001	MOV	1(R3),R4		:GET HIGH ORDER
189	003664	101204	100000	BIS	#PRMY,R4		:SET AS PRIMARY
190	003666	100634	000001	MOV	R4,1(R3)		:STORE BACK
191	003670	024027		XYZ: CALL	RCTWRT		:WRITE OUT BUFFER
192	003671	105200	000002	001256	XYZ1: ADD	#2,FCTPTR	:POINT TO NEXT ENTRY
193	003674	117400	001263	DEC	MNCNT		:DEREMENT IT
194	003676	013721		BEQ	FRDONE		:IF ZERO THEN DONE
195	003677	117400	001451	DEC	SECCNT		:DECREMENT IT
196	003701	053715		BNE	FRSKP		:IF STILL IN BLOCK - CONTINUE
197	003702	102200	000400	001220	BIT	#DLL,FLAG	:DID WE CREATE THE FCT ?
198	003705	013252		BEQ	FBEGIN		:NO - THEN DON'T CHANGE IT
199	003706	104200	004535	001254	MOV	#RDBUF,BUFPT	:POINT TO BUFFER
200	003711	104201	000030	MOV	#F9,R1		:FCT WRITE OVERLAY
201	003713	022435		CALL	PAGE		:EXECUTE IT
202	003714	003252		BR	FBEGIN		:AND GET A NEW ONE
203	003715	104304	001256	FRSKP: MOV	FCTPTR,R4		:ELSE GET CURRENT POINTER
204	003717	060022		XFC	UPDATE		:LET HOST KNOW STILL ALIVE
205	003720	003265		BR	FBEG2		:AND DO NEXT ENTRY
206	003721	102200	000400	001220	FRDONE: BIT	#DLL,FLAG	:DID WE CREATE THE FCT ?
207	003724	013733		BEQ	FCTSP		:NO - THEN DON'T CHANGE IT
208	003725	104200	004535	001254	MOV	#RDBUF,BUFPT	:POINT TO BUFFER
209	003730	104201	000030	MOV	#F9,R1		:WRITE LAST FCT BLOCK
210	003732	022435		CALL	PAGE		:DO IT
211	003733			FCTSP: POP	R3		:GET CURRENT FCT BLOCK NUM
212	003734	104030	001257	MOV	R3,FCTCNT		:RESTORE IT
213	003736	104200	005152	001254	MOV	#PBNBUF,BUFPT	:RE-READ IT IN CASE OF HEADER CHANGES
214	003741	104201	000017	MOV	#F6,R1		:FCT READ OVERLAY
215	003743	022435		CALL	PAGE		:DO IT
216	003744			POP	R3		:GET HIGH ORDER CYLINDER
217	003745	104030	001127	MOV	R3,CYLNUM+1		:STORE IT
218	003747	104030	001100	MOV	R3,ISEEK+2		:STORE IN SEEK COMMAND
219	003751			POP	R3		:GET LOW ORDER
220	003752	104030	001126	MOV	R3,CYLNUM		:RESTORE IT
221	003754	104030	001077	MOV	R3,ISEEK+1		:STORE IN SEEK COMMAND
222	003756			POP	R3		:GET CURRENT TRACK
223	003757	104030	001113	MOV	R3,CURTRK		:RESTORE IT
224	003761	104300	001460	001101	MOV	CURGRP,ISEEK+3	:RESTORE GROUP
225	003764	022242		CALL	SEEK		:RESTORE TO PREVIOUS CYLINDER
226	003765	000000		RETURN			
227							
228							

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 72-1
 FCT->RCT CONVERSION OVERLAY (F5)

58	004160	115405			INC	R5		:YUP - INCREMENT COUNTER
59	004161	115400	001246		RWGOOD: INC	NEXT1		:INCREMENT IT
60	004163	114000	001501		CLR	TMPTRY		:FOR RESET
61	004165	104300	001500	001502	MOV	RECOV,RECTMP		:GET RECOVERY LEVELS
62	004170	104204	000731		MOV	#DDUMMY,R4		:FOR ADD
63	004172	104203	001243		MOV	#RCTFMT,R3		:FOR ADD
64	004174	021503			CALL	DADD		:POINT TO NEXT COPY
65	004175	106300	001245	001246	CMP	FCTCPY,NEXT1		:DONE THIS SECTOR ?
66	004200	054034			BNE	RCTRLP		:NO - WRITE NEXT FCT COPY
67	004201	106305	001245		CMP	FCTCPY,R5		:ERROR ON EVERY WRITE ?
68	004203	014220			BEQ	RCWERR		:YUP - BIG TROUBLE
69	004204	104303	001246		RCFXLP: MOV	NEXT1,R3		:ANY REPEATS ?
70	004206	014217			BEQ	RWTDON		:NO
71	004207	104204	000731		MOV	#DDUMMY,R4		:TO GET IT BACK
72	004211	104203	001243		MOV	#RCTFMT,R3		:DITTO
73	004213	021521			CALL	DSUB		
74	004214	117400	001246		DEC	NEXT1		:SUB IT
75	004216	004204			BR	RCFXLP		:REPEAT
76	004217	000000			RWTDON: RETURN			
77	004220	104012			RCWERR: MOV	R1,R2		:XFC ERROR CODE
78	004221	104201	000017		MOV	#15.,R1		:RCT WRITE ERROR
79	004223	022542			CALL	ERRMNT		:ERROR RETURN

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 73
 RCT INITIALIZE OVERLAY (F7)

```

1          .SBTTL RCT INITIALIZE OVERLAY (F7)
2 004224   DMOVLY F7,START
3          :
4          :
5          :
6 003023   104200 000014 001154 RCTINI: MOV    #F5,CUROVL      ;FOR OVERLAY INIT
7 003026   104203 000400          MOV    #SECSI6,R3    ;SECTOR WORD COUNT
8 003030   114001          CLR    R1            ;FOR INIT OF RCT WORDS
9 003031   104204 004535          MOV    #RDBUF,R4     ;BUFFER
10 003033   100241          RCLP:  MOV    R1,(R4)+ ;STORE IN BUFFER
11 003034   117403          DEC    R3            ;DECREMENT COUNTER
12 003035   053033          BNE   RCLP          ;BRANCH BACK TILL DONE
13 003036   104201 001306          MOV    #SERNUM,R1    ;POINT TO SERIAL NUMBER
14 003040   104204 004535          MOV    #RDBUF,R4     ;POINT TO BUFFER
15 003042   105204 000000          ADD    #RSER,R4     ;POINT TO SERIAL NUMBER
16 003044   104205 000004          MOV    #4,R5        ;COUNTER
17 003046   104212          6$:  MOV    (R1)+,R2     ;GET WORD
18 003047   100242          MOV    R2,(R4)+     ;STORE WORD
19 003050   117405          DEC    R5            ;DECREMENT COUNTER
20 003051   053046          BNE   6$            ;CONTINUE TILL DONE
21 003052   104303 001113          MOV    CURTRK,R3     ;GET CURRENT TRACK
22 003054          PUSH   R3            ;SAVE IT
23 003055   104303 001126          MOV    CYLNUM,R3     ;GET LOW ORDER CYLINDR
24 003057          PUSH   R3            ;SAVE FOR RESTORE
25 003060   104303 001127          MOV    CYLNUM+1,R3   ;GET HIGH ORDER
26 003062          PUSH   R3            ;SAVE FOR RESTORE
27 003063   104300 001136 000736          MOV    RBNLBN,TEMP   ;GET NUMBER OF RBN'S IN LBN AREA
28 003066   104300 001137 000737          MOV    RBNLBN+1,TEMP+1 ;HIGH ORDER
29 003071   104204 000736          MOV    #TEMP,R4     ;FOR ADD
30 003073   104200 000400 000731          MOV    #256.,DDUMMY  ;2 BLOCKS(CONTROL) WORTH OF RBN'S
31 003076   114000 000732          CLR    DDUMMY+1     ;CLEAR HIGH ORDER
32 003100   104203 000731          MOV    #DDUMMY,R3   ;FOR ADD
33 003102   021503          CALL  DADD           ;ADD TO GET 'REAL' NUMBER OF RBN'S
34 003103   104300 001134 001230          MOV    LBNLBN,HOLD   ;GET LOW ORDER COUNT OF LBN'S
35 003106   104300 001135 001231          MOV    LBNLBN+1,HOLD+1 ;GET HIGH ORDER
36 003111   104203 001472          MOV    #TOTRCT,R3   ;FOR SUBTRACT
37 003113   104204 001230          MOV    #HOLD,R4     ;DITTO
38 003115   021521          CALL  DSUB           ;GET STARTING RCT LBN
39 003116   104300 001230 001114          MOV    HOLD,CURBN   ;GET STARTING RCT BLOCK NUMBER
40 003121   104300 001230 001116          MOV    HOLD,CURLBN  ;ALSO SAVE
41 003124   104300 001231 001115          MOV    HOLD+1,CURBN+1 ;GET HIGH ORDER
42 003127   104300 001231 001117          MOV    HOLD+1,CURLBN+1 ;AND SAVE
43 003132   114000 001476          CLR    COUNT        ;CLEAR BLOCK COUNTER
44 003134   104203 001053          MOV    #SCR,R3      ;POINT TO CHARACTERISTICS
45 003136   104207 001424          MOV    #CONBLK,R0   ;POINT TO CONVERT BLOCK
46 003140   104632 000011          MOV    LBNTRK(R3),R2 ;GET LBN/TRACK
47 003142   103202 177400          BIC   #HIBYTE,R2   ;CLEAR HIGH BYTE
48 003144   100672 000004          MOV    R2,V3(R0)    ;FOR CONVERT
49 003146   104632 000001          MOV    STCYL(R3),R2 ;STARTING CLYLINDER
50 003150   103202 007777          BIC   #LO,R2       ;CLEAR REST OF WORD
51 003152   100672 000001          MOV    R2,V1+1(R0)  ;STORE
52 003154   114002          CLR    R2            ;FOR STORE
53 003155   100672 000000          MOV    R2,V1(R0)   ;LOW ORDER ALWAYS 0
54 003157   114005          RCINLP: CLR    R5     ;CLEAR ERROR COUNTER
55 003160   104050 001246          MOV    R5,NEXT1     ;INIT COPY COUNT
56 003162   104204 001114          RCLP2: MOV    #CURBN,R4 ;FOR CONVERT
57 003164   022665          CALL  CVTSK        ;CONVERT AND SEEK

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 73-1
 RCT INITIALIZE OVERLAY (F7)

58	003165	104207	000721	MOV	#WRBLK,R0	:POINT TO COMMAND BLOCK
59	003167	104203	122400	MOV	#WRCMD,R3	:GET WRITE COMMAND
60	003171	104302	001113	MOV	CURTRK,R2	:GET CURRENT TRACK
61	003173	101023		BIS	R2,R3	:SET TRACK FOR WRITE
62	003174	100673	000004	MOV	R3,RW.CMD(R0)	:STORE IN COMMAND BLOCK
63	003176	104202	004535	MOV	#RDBUF,R2	:POINT TO BUFFER
64	003200	022600		CALL	EDC	:COMPUTE EDC - RETURNED IN R3
65	003201	100623	000400	MOV	R3,RW.EDC(R2)	:STORE IT
66	003203	100672	000001	MOV	R2,RW.BUF(R0)	:STICK IN COMMAND BLOCK
67	003205	104303	001114	MOV	CURBN,R3	:GET LOW ORDER HEADER
68	003207	100673	000002	MOV	R3,RW.LOW(R0)	:STORE IN WRITE BLOCK
69	003211	104303	001115	MOV	CURBN+1,R3	:GET HIGH ORDER
70	003213	105303	001323	ADD	ST.LBN,R3	:ADD STARTING LBN BITS
71	003215	101203	000000	BIS	#HD.LBN,R3	:SET HEADER
72	003217	100673	000003	MOV	R3,RW.HI(R0)	:STORE IN WRITE BLOCK
73	003221	104203	000726	MOV	#HSLIM-1,R3	:GET DUMMY SDI POINTER
74	003223	100673	000005	MOV	R3,RW.DUM(R0)	:POINT IN COMMAND BLOCK
75	003225	104303	001321	WRITE4: MOV	HPREA,R3	:GET HEADER PREAMBLE
76	003227	104304	001322	MOV	DPREA,R4	:GET DATA PREAMBLE
77	003231	104302	000740	MOV	UNIT,R2	:SET UNIT
78	003233	104207	100721	MOV	#<WRBLK!BIT15>,RC	:MAKE SURE POINTING AT BLOCK
79	003235	060012		XFC	SIP	:WAIT FOR SECTOR PULSE
80	003236	060003		XFC	WRITE	:WRITE SECTOR
81	003237	115001		TST	R1	:ANY ERROR ?
82	003240	013265		BEQ	NOGOOD	:NOPE
83	003241	106300	001477 001501	CMP	RETRY,TMPTRY	:MAX ?
84	003244	013250		BEQ	1\$:YES - TRY SOME RECOVERY
85	003245	115400	001501	INC	TMPTRY	:INC RETRY COUNT
86	003247	003225		BR	WRITE4	:DO RETRY
87	003250	104303	001502	1\$: MOV	RECTMP,R3	:GET CURRENT ERROR RECOVERY LEVEL
88	003252	073264		BMI	2\$:IF NEGATIVE THEN FRIED
89	003253	115000	001500	TST	RECOV	:IS THERE ONLY RECOVERY LEVEL 0 ?
90	003255	013257		BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY
91	003256	022571		CALL	ERRHND	:TRY RECOVERY
92	003257	114000	001501	3\$: CLR	TMPTRY	:FOR INIT
93	003261	117400	001502	DEC	RECTMP	:DECREMENT IT
94	003263	003225		BR	WRITE4	:RETRY
95	003264			2\$:		
96	003264	115405		INC	R5	:YUP - INCREMENT COUNTER
97	003265	115400	001246	NOGOOD: INC	NEXT1	:INCREMENT IT
98	003267	114000	001501	CLR	TMPTRY	:FOR RESET
99	003271	104300	001500 001502	MOV	RECOV,RECTMP	:GET RECOVERY LEVELS
100	003274	104204	001114	MOV	#CURBN,R4	:FOR ADD
101	003276	104203	001243	MOV	#RCTFMT,R3	:FOR ADD
102	003300	021503		CALL	DADD	:POINT TO NEXT COPY
103	003301	106300	001245 001246	CMP	FCTCPY,NEXT1	:DONE THIS SECTOR ?
104	003304	053162		BNE	RCLF2	:NO - WRITE NEXT FCT COPY
105	003305	106305	001245	CMP	FCTCPY,R5	:ERROR ON EVERY WRITE ?
106	003307	013441		BEQ	RCINER	:YUP - BIG TROUBLE
107	003310	102200	000040 001220	BIT	#RCINIT,FLAG	:ALREADY FIXED IT UP
108	003313	053332		BNE	4\$:YUP - NO NEED TO DO IT AGAIN
109	003314	104204	000736	MOV	#TEMP,R4	:FOR SUBTRACT (RBN'S NOT DONE)
110	003316	104200	000200 000731	MOV	#128,,DDUMMY	:SUBTRACT ONE BLOCKS WORTH
111	003321	114000	000732	CLR	DDUMMY+1	:FOR CLEAR
112	003323	104203	000731	MOV	#DDUMMY,R3	:FOR SUBTRACT
113	003325	021521		CALL	DSUB	:SUBTRACT
114	003326	021623		CALL	DCMP	:IN LAST BLOCK ?

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 73-2
 RCT INITIALIZE OVERLAY (F7)

115	003327	073332				BMI	4\$:NOPE
116	003330	013332				BEQ	4\$:IF EQUAL - NO PARTIAL BLOCK
117	003331	023445				CALL	FIXBLK		:YES - CHANGE HEADERS TO NULL
118	003332	115000	001476		4\$:	TST	COUNT		:ON FIRST BLOCK ?
119	003334	053345				BNE	7\$:NO - NO NEED TO FIX UP
120	003335	114005				CLR	R5		:FOR BLOCK FIXUP
121	003336	104204	004535			MOV	#RDBUF,R4		:POINT TO BUFFER
122	003340	104201	000004			MOV	#4,R1		:COUNTER
123	003342	100245			5\$:	MOV	R5,(R4)+		:CLEAR DATE AREA
124	003343	117401				DEC	R1		:DECREMENT COUNTER
125	003344	053342				BNE	5\$:CONT TILL DONE
126	003345	102200	000100	001221	7\$:	BIT	#RCINDN,FLAG1		:ALL DONE ??
127	003350	053377				BNE	RCLP6		:YUP - CUT OUT
128	003351					DUBINC	CURLBN		:INCREMENT IT
129	003356	104300	001116	001114		MOV	CURLBN,CURBN		:GET LOW ORDER
130	003361	104300	001117	001115		MOV	CURLBN+1,CURBN+1		:GET HIGH ORDER
131	003364	115400	001476			INC	COUNT		:INCREMENT BLOCK COUNTER
132	003366	106300	001262	001476		CMP	RCTLBN,COUNT		:DONE RCT BLOCKS(NOT PAD)
133	003371	013421				BEQ	RCFIX		:YUP - REINIT BLOCK
134	003372	060022			RCLP4:	XFC	UPDATE		:LET HOST KNOW STILL ALIVE
135	003373	106300	001243	001476		CMP	RCTFMT,COUNT		:DONE ?
136	003376	053157				BNE	RCINLP		:NOPE - DO NEXT SECTOR
137	003377				RCLP6:	POP	R3		:GET HIGH ORDER CYLINDER
138	003400	104030	001127			MOV	R3,CYLNUM+1		:STORE IT
139	003402	104030	001100			MOV	R3,ISEEK+2		:STORE IN SEEK COMMAND
140	003404					POP	R3		:GET LOW ORDER
141	003405	104030	001126			MOV	R3,CYLNUM		:RESTORE IT
142	003407	104030	001077			MOV	R3,ISEEK+1		:STORE IN SEEK COMMAND
143	003411					POP	R3		:GET CURRENT TRACK
144	003412	104030	001113			MOV	R3,CURTRK		:RESTORE IT
145	003414	104300	001460	001101		MOV	CURGRP,ISEEK+3		:RESTORE GROUP NUMBER
146	003417	022242				CALL	SEEK		:RESTORE TO PREVIOUS CYLINDER
147	003420	000000				RETURN			
148	003421	104202	000200		RCFIX:	MOV	#126.,R2		:INIT COUNT
149	003423	104204	004535			MOV	#RDBUF,R4		:INIT POINTER
150	003425	114003				CLR	R3		:FOR STORE
151	003426	114005				CLR	R5		:DITTO
152	003427	101205	100000			BIS	#RC.NUL,R5		:SET NULL HEADER
153	003431	100243			RCLP3:	MOV	R3,(R4)+		:STORE LOW ORDER
154	003432	100245				MOV	R5,(R4)+		:STORE HIGH ORDER
155	003433	117402				DEC	R2		:DECREMENT COUNTER
156	003434	053431				BNE	RCLP3		:LOOP UNTIL DONE
157	003435	101200	000100	001221		BIS	#RCINDN,FLAG1		:DONE ALL NON-PAD - ONE MORE THEN FINISH
158	003440	003372				BR	RCLP4		:CONTINUE WITH LAST SECTOR
159	003441	104012			RCINER:	MOV	R1,R2		:XFC ERROR CODE
160	003442	104201	000017			MOV	#15.,R1		:RCT INIT ERROR
161	003444	022542				CALL	ERRMNT		:ERROR RETURN

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 75
FCT READ OVERLAY (F6)

```

1          .SBTTL FCT READ OVERLAY (F6)
2 003472   DMOVLY F6,START
3          :
4          :
5          :
6          :
7 003023   104200 000017 001154   MOV      #F6,CUROVL      ;OVERLAY NUMBER
8 003026   104300 001257 000731   MOV      FCTCNT,DDUMMY  ;GET CURRENT COUNT
9 003031   114000 000732          CLR      DDUMMY+1      ;FOR HIGH ORDER STORE
10 003033   114005          CLR      R5           ;CLEAR ERROR COUNTER
11 003034   104204 000731   FOLOOP: MOV      #DDUMMY,R4   ;FOR CONVERT
12 003036   104303 001144          MOV      LBNCYL,R3    ;GET LBN CYLINDERS
13 003040   104207 001424          MOV      #CONBLK,R0   ;POINT TO CONVERT BLOCK
14 003042   100673 000000          MOV      R3,V1(R0)    ;STORE IT FOR CONVERT
15 003044   104303 001145          MOV      LBNCYL+1,R3  ;HIGH ORDER
16 003046   100673 000001          MOV      R3,V1+1(R0)  ;STORE IT
17 003050   104303 001130          MOV      SECTRK,R3    ;GET SECTORS/TRACK
18 003052   100673 000004          MOV      R3,V3(R0)   ;STORE FOR CONVERT
19 003054   022665          CALL     CVTSK        ;CONVERT FCT BLOCK NUMBER AND SEEK
20 003055   104207 000721          MOV      #RDBLK,R0    ;PREPARE FOR READ SECTORS
21 003057   104203 000726          MOV      #HSLIM-1,R3 ;POINTER TO DUMMY SDI BLOCK
22 003061   100673 000005          MOV      R3,RW.DJM(R0);STORE IN COMMAND BLOCK
23 003063   104303 000731          MOV      DDUMMY,R3    ;LO ORDER BLOCK NUMBER
24 003065   100673 000002          MOV      R3,RW.LOW(R0);STORE IN READ CMD BLOCK
25 003067   104303 000732          MOV      DDUMMY+1,R3  ;GET HIGH ORDER
26 003071   105303 001325          ADD      ST.XBN,R3    ;ADD STARTING LBN BITS
27 003073   101203 120000          BIS      #HD.XBN,R3   ;HEADER CODE
28 003075   100673 000003          MOV      R3,RW.HI(R0) ;STORE IN READ CMD BLOCK
29 003077   104303 001254          MOV      BUFPT,R3     ;GET BUFFER POINTER
30 003101   100673 000001          MOV      R3,RW.BUF(R0);STORE BUFFER ADDRESS IN COMMAND BUFFER
31 003103   104203 013400          MOV      #RW.CMD,R3   ;LOAD SDI READ COMMAND
32 003105   104301 001113          MOV      CURTRK,R1    ;GET CURRENT HEAD NUMBER IN R1
33 003107   101013          BIS      R1,R3       ;SET IT IN COMMAND
34 003110   100673 000004          MOV      R3,RW.CMD(R0);STORE BACK
35 003112   104207 000721   READ7: MOV      #RDBLK,R0  ;MAKE SURE POINTING AT BLOCK
36 003114   104203 100000          MOV      #RDCMD,R3   ;MARK AS ONLY REQUEST
37 003116   100173          MOV      R3,(R0)     ;STORE IN CMD BLOCK
38 003117   104302 000740          MOV      UNIT,R2     ;SDI INTERCONNECT
39 003121   101207 100000          BIS      #BIT15,R0   ;SET NO REVECTORING
40 003123   060012          XFC      SIP         ;WAIT FOR PULSE
41 003124   060002          XFC      READ        ;READ 1 SECTOR
42 003125   115001          TST      R1          ;ANY ERRORS ?
43 003126   053144          BNE      100$        ;YES - TRY RECOVERY
44 003127   104173          MOV      (R0),R3     ;GET STATUS WORD
45 003130   102203 010000          BIT      #ECCF,R3    ;ECC ERROR ?
46 003132   013136          BEQ      101$        ;NOPE - VERIFY EDC
47 003133   023000          CALL     ECCCK       ;CORRECT ECC
48 003134   115001          TST      R1          ;TEST FLAG
49 003135   053144          BNE      100$        ;UNCORRECTABLE
50 003136   104302 001254   101$: MOV      BUFPT,R2    ;POINT TO BUFFER
51 003140   022600          CALL     CEDC        ;COMPUTE EDC
52 003141   106623 000400          CMP      RW.EDC(R2),R3;O.K. ?
53 003143   013206          BEQ      102$        ;YUP - CONSIDER GOOD
54 003144   106300 001477 001501 100$: CMP      RETRY,TMPTRY ;MAX ?
55 003147   013153          BEQ      1$         ;YES - TRY SOME RECOVERY
56 003150   115400 001501          INC      TMPTRY      ;INC RETRY COUNT
57 003152   003112          BR       READ7      ;DO RETRY

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 75-1
FCT READ OVERLAY (F6)

58	003153	104303	001502	1\$:	MOV	RECTMP,R3	:GET CURRENT ERROR RECOVERY LEVEL
59	003155	073167			BMI	2\$:IF NEGATIVE THEN FRIED
60	003156	115000	001500		TST	RECOV	:IS THERE ONLY RECOVERY LEVEL 0 ?
61	003160	013162			BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY
62	003161	022571			CALL	ERRHND	:TRY RECOVERY
63	003162	114000	001501	3\$:	CLR	TMPTRY	:FOR INIT
64	003164	117400	001502		DEC	RECTMP	:DECREMENT IT
65	003166	003112			BR	READ7	:RETRY
66	003167			2\$:			
67	003167	115405			INC	R5	:INCREMENT BAD COUNTER
68	003170	106305	001245		CMP	FCTCPY,R5	:ALL BAD ?
69	003172	013331			BEQ	OFATAL	:YUP - ALL OVER
70	003173	104204	000731		MOV	#DDUMMY,R4	:POINT TO COUNT
71	003175	104203	001241		MOV	#FCTFMT,R3	:SIZE OF TABLE
72	003177	021503			CALL	DADD	:ADD TO POINT TO NEXT COPY
73	003200	114000	001501		CLR	TMPTRY	:RESET RETRY LEVEL
74	003202	104300	001500	001502	MOV	RECOV,RECTMP	:DITTO RECOVERY LEVELS
75	003205	003034			BR	FOLOOP	:BRANCH BACK
76	003206			102\$:			
77	003206	114000	001501	FODONE:	CLR	TMPTRY	:FOR RESET
78	003210	104300	001500	001502	MOV	RECOV,RECTMP	:GET RECOVERY LEVELS
79	003213	115005			TST	R5	:ANY ERRORS ?
80	003214	013326			BEQ	OLDONE	:NO - EXIT
81	003215	104204	000731		MOV	#DDUMMY,R4	:POINT TO BLOCK COUNT
82	003217	104203	001241		MOV	#FCTFMT,R3	:SIZE OF TABLE
83	003221	021521			CALL	DSUB	:GET BACK TO PREVIOUS COPY
84	003222	022665			CALL	CVTSK	:CONVERT AND SEEK
85	003223	104207	000721		MOV	#WRBLK,R0	:POINT TO COMMAND BLOCK
86	003225	104203	122400		MOV	#WRCMD,R3	:GET WRITE COMMAND
87	003227	104302	001113		MOV	CURTRK,R2	:GET CURRENT TRACK
88	003231	101023			BIS	R2,R3	:SET TRACK FOR WRITE
89	003232	100673	000004		MOV	R3,RW.CMD(R0)	:STORE IN COMMAND BLOCK
90	003234	104303	001254		MOV	BUFPNT,R3	:GET BUFFER ADDRESS
91	003236	100673	000001		MOV	R3,RW.BUF(R0)	:STICK IN COMMAND BLOCK
92	003240	104303	000731		MOV	DDUMMY,R3	:GET LOW ORDER HEADER
93	003242	100673	000002		MOV	R3,RW.LOW(R0)	:STORE IN WRITE BLOCK
94	003244	104303	000732		MOV	DDUMMY+1,R3	:GET HIGH ORDER
95	003246	105303	001325		ADD	ST.XBN,R3	:ADD STARTING XBN BITS
96	003250	101203	120000		BIS	#HD.XBN,R3	:HEADER CODE
97	003252	100673	000003		MOV	R3,RW.HI(R0)	:STORE IN WRITE BLOCK
98	003254	104203	000726		MOV	#HSLIM-1,R3	:GET DUMMY SDI POINTER
99	003256	100673	000005		MOV	R3,RW.DUM(R0)	:POINT IN COMMAND BLOCK
100	003260	104303	001321	WRITE8:	MOV	HPREA,R3	:GET HEADER PREAMBLE
101	003262	104304	001322		MOV	DPREA,R4	:GET DATA PREAMBLE
102	003264	104302	000740		MOV	UNIT,R2	:SET UNIT
103	003266	104207	000721		MOV	#WRBLK,R0	:MAKE SURE POINTING AT BLOCK
104	003270	101207	100000		BIS	#BIT15,R0	:SET NO REVECTORING
105	003272	060012			XFC	SIP	:WAIT FOR SECTOR PULSE
106	003273	060003			XFC	WRITE	:WRITE SECTOR
107	003274	115001			TST	R1	:ANY ERROR ?
108	003275	013321			BEQ	2\$:NO - SKIP RETRY
109	003276	106300	001477	001501	CMP	RETRY,TMPTRY	:MAX ?
110	003301	013305			BEQ	1\$:YES - TRY SOME RECOVERY
111	003302	115400	001501		INC	TMPTRY	:INC RETRY COUNT
112	003304	003260			BR	WRITE8	:DO RETRY
113	003305	104303	001502	1\$:	MOV	RECTMP,R3	:GET CURRENT ERROR RECOVERY LEVEL
114	003307	073321			BMI	2\$:IF NEGATIVE THEN FRIED

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:37 PAGE 76
 GET FCT BLOCK FOR D/XBN FORMAT (G2)

1					.SBTTL	GET FCT BLOCK FOR D/XBN FORMAT (G2)	
2							
3							
4							
5						GET'S ONE FCT BLOCK FROM HOST FOR D/BN FORMATTING	
6							
7	003356				DMOVLY	G2,START	
8							
9							
10							
11	003023	104200	000033	001154	MOV	#G2,CUROVL	;SIGNAL OVERLAY 11
12	003026	104205	001264		MOV	#DMBUF,R5	;POINT TO MAINT BUFFER
13	003030	104303	001317		MOV	FMSG,R3	;GET DUP CODE
14	003032	100153			MOV	R3,(R5)	;STORE IT
15	003033	104303	001257		MOV	FCTCNT,R3	;GET BLOCK NUMBER DESIRED
16	003035	100653	000001		MOV	R3,1(R5)	;STORE IT
17	003037	022522			CALL	SNDMNT	;SEND REQUEST
18	003040	022532			CALL	RCVMNT	;RECEIVE ANSWER
19	003041	104153			MOV	(R5),R3	;GET STATUS WORD
20	003042	053062			BNE	DLERR	;ERROR IF NOT ZERO
21	003043	104650	000001	001421	MOV	1(R5),OVLBLK+1	;GET LOW HOST ADDRESS
22	003046	104650	000002	001422	MOV	2(R5),OVLBLK+2	;GET HIGH HOST ADDRESS
23	003051	104200	000401	001420	MOV	#257,OVLBLK	;GET LENGTH
24	003054	104204	001420		MOV	#OVLBLK,R4	;FOR OVERLAY ROUTINE
25	003056	104203	005152		MOV	#PBNBUF,R3	;POINT TO BUFFER
26	003060	022510			CALL	OVRLAY	;GET THE SECTOR
27	003061	000000			RETURN		
28	003062	104201	000023		DLERR: MOV	#19,R1	;SIGNAL DLL ERROR
29	003064	104302	001257		MOV	FCTCNT,R2	;BLOCK FAILED ON
30	003066	022542			CALL	ERRMNT	;ERROR RETURN

UDAFM - UDA FGMATTER DMACR XG4.01 23-AUG-82 14:02:32 PAGE 77
 GET FCT BLOCK FOR LBN FORMAT (G3)

1					.SBTTL GET FCT BLOCK FOR LBN FORMAT (G3)	
2						
3						
4					GET RIGHT FCT BLOCK FOR LBN FORMATTING	
5						
6						
7	003067				DMOVLY G3,START	
8						
9						
10						
11	003023	104200	000036	001154	MOV #G3,CUROVL	:FOR CURRENT OVERLAY
12	003026	114000	001257		CLR FCTCNT	:FOR FIRST FCT BLOCK
13	003030	104200	005152	001254	MOV #PBNBUF,BUFPNT	:POINT TO BUFFER
14	003033	104201	000017		MOV #F6,R1	:FCT READ OVERLAY
15	003035	022435			CALL PAGE	:READ IT IN
16	003036	102200	000001	001220	BIT #FCTAVL,FLAG	:FCT STILL HERE ?
17	003041	013224			BEQ NGD	:NOPE - CAN IT
18	003042	104207	005152		MOV #PBNBUF,R0	:POINT TO BUFFER
19	003044	104173			MOV (R0),R3	:GET FORMAT MEDIA WORD
20	003045	106203	126736		CMP #M512,R3	:IS IT 512 ?
21	003047	013055			BEQ 13\$:YUP - O.K.
22	003050	106203	074161		CMP #M576,R3	:IS IT 576 ?
23	003052	013055			BEQ 13\$:YUP - O.K.
24	003053	115003			TST R3	:IS IT FORMAT IN PROGRESS
25	003054	053224			BNE NGD	:NOPE - FCT NO GOOD
26	003055	104673	000025	13\$:	MOV FCTFLG(R0),R3	:GET FLAG WORD
27	003057	102203	100000		BIT #NOFCT,R3	:IS THERE REALLY AN FCT ??
28	003061	053234			BNE NGD1	:NOPE - CONTINUE BUT USE BST GSS MODE
29	003062	104673	000016		MOV C512(R0),R3	:GET COUNT OF USED ENTRIES
30	003064	104030	001240		MOV R3,FCNT	:STORE IT
31	003066	053072			BNE 12\$:IF NOT ZERO THEN ENTRIES EXIST
32	003067	101200	000002	001220	BIS #FCTEMT,FLAG	:SET EMPTY FLAG
33	003072	114003		12\$:	CLR R3	:FOR FCT INIT
34	003073	100173			MOV R3,(R0)	:SIGNAL FORMAT IN PROGRESS
35	003074	104673	000001		MOV INST(R0),R3	:FORMAT INSTANCE NUMBER
36	003076	115403			INC R3	:INCREMENT IT
37	003077	100673	000001		MOV R3,INST(R0)	:STORE IT BACK
38	003101	104203	005152		MOV #PBNBUF,R3	:POINT TO BUFFER
39	003103	105203	000002		ADD #FSER,R3	:POINT TO SERIAL NUMBER
40	003105	104204	001306		MOV #SERNUM,R4	:SERIAL NUMBER BLOCK
41	003107	104205	000004		MOV #4,R5	:COUNTER
42	003111	104232		8\$:	MOV (R3)+,R2	:GET WORD
43	003112	100242			MOV R2,(R4)+	:STORE WORD
44	003113	117405			DEC R5	:DECREMENT COUNTER
45	003114	053111			BNE 8\$:CONT TILL DONE
46	003115	104200	005152	001254	MOV #PBNBUF,BUFPNT	:POINT TO BUFFER
47	003120	104201	000030		MOV #F9,R1	:FCT WRITE OVERLAY
48	003122	022435			CALL PAGE	:DO IT
49	003123	115000	001240		TST FCNT	:ANY ENTRIES ?
50	003125	013220			BEQ RDONE1	:NOPE - ALL DONE
51	003126	104200	000001	001257	MOV #1,FCTCNT	:FOR FCT COUNT INIT
52	003131	104200	005152	001254	RLOOP: MOV #PBNBUF,BUFPNT	:POINT TO BUFFER
53	003134	104201	000017		MOV #F6,R1	:FCT READ OVERLAY
54	003136	022435			CALL PAGE	:DO THE READ
55	003137	102200	000001	001220	BIT #FCTAVL,FLAG	:STILL HAVE FCT ?
56	003142	013224			BEQ NGD	:NOPE - CAN IT
57	003143	104204	005152		MOV #PBNBUF,R4	:POINT TO THE BUFFER

JDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 77-1
GET FCT BLOCK FOR LBN FORMAT (G3)

58	003145	105204	000376		ADD	#254.,R4		:POINT TO LAST ENTRY
59	003147	104203	001155		MOV	#HGHPBN,R3		:HIGHEST PBN IN LBN AREA
60	003151	021623			CALL	DCMP		:IS IT RIGHT BLOCK ?
61	003152	033157			BPL	BLKFND		:YES - FIND RIGHT ENTRY
62	003153	107200	000200	001240	SUB	#128.,FCNT		:SUBTRACT ONE BLOCKS WORTH
63	003156	003131			BR	RLOOP		:BRANCH BACK
64	003157	104200	000200	001475	BLKFND: MOV	#128.,PCNT		:FOR INIT OF COUNT
65	003162	104204	005152		MOV	#PBNBUF,R4		:POINT TO PBN BUFFER
66	003164	104647	000001		RLOOP1: MOV	1(R4),R0		:GET HIGH ORDER
67	003166	104071			MOV	R0,R1		:SAVE IT TEMPORARILY
68	003167	103207	170000		BIC	#HD.CLR,R0		:CLEAR FOR COMPARE
69	003171	100647	000001		MOV	R0,1(R4)		:STORE IT BACK
70	003173	104203	001155		MOV	#HGHPBN,R3		:POINT TO HIGHEST PBN
71	003175	021623			CALL	DCMP		:COMPARE
72	003176	033214			BPL	RDONE		:IF LESS THAN OR EQUAL THEN FOUND FIRST LBBN
73	003177	100641	000001		MOV	R1,1(R4)		:STORE HEADER BACK
74	003201	117400	001475		DEC	PCNT		:DECREMENT COUNT
75	003203	117400	001240		DEC	FCNT		:DEC IT
76	003205	105204	000002		ADD	#2,R4		:POINT TO NEXT ENTRY
77	003207	106200	000001	001240	CMP	#1,FCNT		:COUNT AT 1 ?
78	003212	013214			BEQ	RDONE		:YUP -- THEN LAST ENTRY IS IT
79	003213	003164			BR	RLOOP1		:TRY NEXT ENTRY
80	003214	100641	000001		RDONE: MOV	R1,1(R4)		:STORE HEADER BACK
81	003216	104040	001224		MOV	R4,BADPBN		:MAKE CURRENT BAD POINTER
82	003220	104300	001240	001312	RDONE1: MOV	FCNT,FCTREV		:FCT ENTRY COUNT FOR LATER USE
83	003223	000000			RETURN			:RETURN
84	003224	102200	000020	001220	NGD: BIT	#GOBAD,FLAG		:CONTINUE AS BEST GUESS ?
85	003227	013243			BEQ	RQUIT		:NOPE - GIVE UP
86	003230	104201	000000		MOV	#F1,R1		:POINT TO D/XBN OVERLAY
87	003232	022403			CALL	NEXT		:START OVER IN BEST GUESS MODE
88	003233	000000			RDONE2: RETURN			
89	003234	101200	002004	001220	NGD1: BIS	#FCTBAD+BSTGS,FLAG		:SET FCT NOT USED FLAG
90	003237	103200	000001	001220	BIC	#FCTAVL,FLAG		:NO MORE FCT (SO WILL DO EXTEN READS
91	003242	003233			BR	RDONE2		:EXIT
92	003243	104201	000022		RQUIT: MOV	#18.,R1		:ERROR CODE
93	003245	114002			CLR	R2		:NO SUBCODE
94	003246	022542			CALL	ERRMNT		:ERROR RETURN

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 78
 RCT CLEANUP OVERLAY (G4)

```

1          .SBTTL RCT CLEANUP OVERLAY (G4)
2          :
3          :
4          :
5          :
6          :
7          :
8          :
9          :
10         :
11         :
12         :
13         :
14         :
15         :
16         :
17         :
18         :
19         :
20         :
21         :
22         :
23         :
24         :
25         :
26         :
27         :
28         :
29         :
30         :
31         :
32         :
33         :
34         :
35         :
36         :
37         :
38         :
39         :
40         :
41         :
42         :
43         :
44         :
45         :
46         :
47         :
48         :
49         :
50         :
51         :
52         :
53         :
54         :
55         :
56         :
57         :

```

003247					DMOVL	G4,START	
003023	104200	000041	001154		MOV	#G4,CUROVL	:FOR OVERLAY IDENT
003026	104207	001264			MOV	#DMBUF,R0	:MESSAGE BUFFER
003030	104303	001467			MOV	SND CNT,R3	:ANY SECONDARY REVECTORS ?
003032	100173				MOV	R3,(R0)	:STORE IT
003033	115003				TST	R3	:ARE THERE ANY ?
003034	013154				BEQ	CLSKP3	:NOPE - JUST EXIT
003035	104202	006204			MOV	#REVBUF,R2	:POINT TO REVECTOR BUFFER
003037	104200	000100	001476		MOV	#64.,COUNT	:COUNT OF MAX TO REVECTOR AT ONCE
003042	114000	001107			CLR	CURRBN	:CLEAR FOR INIT
003044	114000	001110			CLR	CURRBN+1	:HIGH ORDER TOO
003046	104200	000002	001474		MOV	#2,RCTCNT	:INIT RCT BLOCK
003051	104200	000200	001451	CLELP2:	MOV	#128.,SECCNT	:GET COUNT OF RCT ENTRIES
003054	104304	001474			MOV	RCTCNT,R4	:GET BLOCK NUMBER TO READ
003056	024373				CALL	RRC	:READ IT
003057	104205	006621			MOV	#RCTBUF,R5	:POINT TO BUFFER
003061	104653	000001		CLELP:	MOV	1(R5),R3	:GET HEADER
003063	103203	007777			BIC	#LO,R3	:CLEAR OUT LOW GARBAGE
003065	106203	030000			CMF	#RC.SND,R3	:IS IT A SECONDARY ?
003067	053117				CLESKP		:NO - SKIP REVECTORING
003070	104153				MOV	(R5),R3	:GET LOW ORDER
003071	100223				MOV	R3,(R2)+	:STORE IN REVECTOR BUFFER
003072	104653	000001			MOV	1(R5),R3	:GET HIGH ORDER
003074	103203	170000			BIC	#HD.CLR,R3	:CLEAR HEADER
003076	101203	030000			BIS	#HD.REV,R3	:SET AS AN LBN REVECTOR
003100	100223				MOV	R3,(R2)+	:STORE IT
003101	104303	001107			MOV	CURRBN,R3	:GET LOW ORDER RBN NUMBER
003103	100223				MOV	R3,(R2)+	:STORE IT
003104	104303	001110			MOV	CURRBN+1,R3	:GET HIGH ORDER
003106	100223				MOV	R3,(R2)+	:STORE IT
003107	117400	001476			DEC	COUNT	:DEC NUM OF EMPTY REVECTOR SLOTS
003111	117400	001467			DEC	SND CNT	:DECREMENT IT
003113	013153				BEQ	CLSKP4	:IF ZERO THEN DONE
003114	104303	001476			MOV	COUNT,R3	:FULL BLOCK ?
003116	013140				BEQ	CLSKP2	:IF 0 - PROCESS BLOCK
003117				CLESKF:	DUBINC	CURRBN	:INCREMENT IT
003124	105205	000002			ADD	#2,R5	:POINT TO NEXT RBN ENTR
003126	117400	001451			DEC	SECCNT	:DECREMENT IT
003130	053061				BNE	CLELP	:DO NEXT ENTRY IF NOT ZERO
003131	115400	001474			INC	RCTCNT	:INCREMENT IT
003133	106300	001243	001474		CMF	RCTFMT,RCTCNT	:DONE ?
003136	053051				BNE	CLELP2	:NOPE - READ IN NEXT BLOCK
003137	003147				BR	CLEDON	:ELSE DONE
003140	023743			CLSKP2:	CALL	CLEWRT	:PROCESS THE BLOCK
003141	104200	000100	001476		MOV	#64.,COUNT	:FOR COUNTER INIT
003144	104202	006204			MOV	#REVBUF,R2	:RESET POINTER
003146	003117				BR	CLESKP	:BRANCH BACK
003147	106200	000100	001476	CLEDON:	CMF	#64.,COUNT	:DONE ANY ?
003152	013154				BEQ	CLSKP3	:NO - DONE
003153	023743			CLSKP4:	CALL	CLEWRT	:WRITE OUT ANY LEFTOVERS

UDAFM - JDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 78-1
PCT CLEANUP OVERLAY (G4)

58 003154
59 003154 104201 000066
60 003156 022435
61 003157 023546
62 003160 022315
63 003161 *14007
64 003162 J60021

CLSKP3:

MOV #H2,R1
CALL PAGE
CALL SNDRES
CALL DISCON
CLR RO
XFC DONE

;FINAL CHECK OF FCT,RCT,HEADS OVERLAY
;PAGE IT IN
;SEND FINAL STATS
;DISCONNECT/SPINDOWN DRIVE
;MAKE SURE QUILTS NICELY
;DONE

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 79
 RCT CLEANUP OVERLAY (G4)

```

1
2
3
4 003163
5
6
7
8
9
10 003172 030001
11 003173 106 157 162
12 000012
13
14
15
16 003204 040014
17 003205 106 103 124
18 000014
19
20
21
22 003220 040015
23 003221 106 103 124
24 000012
25
26
27
28 003232 030020
29 003233 127 101 122
30 000040
31
32
33 003272
34
35
36
37 003272 000015
38 003273 003300
39 003274 030002
40 003275
41 003300 003300
42 003300 040 122 145
43 000015
44
45
46
47 003311 000021
48 003312 003317
49 003313 030003
50 003314
51 003317 003317
52 003317 040 120 162
53 000021
54
55
56
57 003334 000026

```

```

:
: DATA STRUCTURES FOR STATUS RESPONSE
:
:
: PARMTB: .BLKW 7 ;PARAMETER TABLE
:
:
: FORMAT COMPLETE
:
: .ENABL LC
: FINMSG: .WORD 30001 ;DUP WORD
: .ASCIZ 'Format completed'
: LFINMS = .-FINMSG
:
: FCT USED
:
: FCTUSD: .WORD 40014 ;DUP WORD
: .ASCIZ 'FCT used successfully'
: LFCTUS = .-FCTUSD
:
: FCT NOT USED
:
: FCTNOT: .WORD 40015 ;DUP WORD
: .ASCIZ 'FCT was not used'
: LFCTNT = .-FCTNOT
:
: WARNING OF POSSIBLE HEAD PROBLEM
:
: WRN: .WORD 30020 ;DUP WORD
: .ASCIZ 'WARNING - possible head addressing problem - run diagnostics'
: WRNLN = .-WRN
:
: RESTAB:
:
: REVECTORED LBNS
:
: .WORD LLEN ;LENGTH
: .WORD LBUFE ;END OF BUFFER
: 1$: .WORD 30002 ;DUP WORD
: .BLKW 3 ;CONVERT BUFFER
: LBUFE =
: .ASCIZ 'Revectored LBNS'
: LLEN = .-1$
:
: PRIMARY REVECTORS
:
: .WORD PLEN ;LENGTH
: .WORD PBUFE ;END OF BUFFER
: 2$: .WORD 30003 ;DUP WORD
: .BLKW 3 ;CONVERT BUFFER
: PBUFE =
: .ASCIZ 'Primary revectored LBNS'
: PLEN = .-2$
:
: SECONDARY REVECTORS
:
: .WORD SLEN ;LENGTH

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 79-1
 RCT CLEANUP OVERLAY (G4)

```

58 003335 003342          .WORD  SBUFE          :END OF BUFFER
59 003336 030004          3$:  .WORD  30004          :DUP WORD
60 003337                .BLKW  3              :CONVERT BUFFER
61                SBUFE  =
62 003342 003342 123 145  .ASCIZ  'Secondary/tertiary revector'd LBNS'
63                SLEN  =  .-3$
64                :
65                : RCT BAD BLOCKS
66                :
67 003364 000034          .WORD  RCLEN          :LENGTH
68 003365 003372          .WORD  RCBUFE         :END OF BUFFER
69 003366 030005          4$:  .WORD  30005          :DUP WORD
70 003367                .BLKW  3              :CONVERT BUFFER
71                RCBUFE =
72 003372 003372 102 141  .ASCIZ  'Bad blocks in the RCT area due to data errors'
73                RCLEN  =  .-4$
74                :
75                : DBN BAD BLOCKS
76                :
77                :
78 003422 000034          .WORD  DBLEN          :LENGTH
79 003423 003430          .WORD  DBBUFE         :END OF BUFFER
80 003424 030007          5$:  .WORD  30007          :DUP WORD
81 003425                .BLKW  3              :CONVERT BUFFER
82                DBBUFE =
83 003430 003430 102 141  .ASCIZ  'Bad blocks in the DBN area due to data errors'
84                DBLEN  =  .-5$
85                :
86                : XBN BAD BLOCKS
87                :
88 003460 000034          .WORD  XBLEN          :LENGTH
89 003461 003466          .WORD  XBBUFE         :END OF BUFFER
90 003462 030010          6$:  .WORD  30010          :DUP WORD
91 003463                .BLKW  3              :CONVERT BUFFER
92                XBBUFE =
93 003466 003466 102 141  .ASCIZ  'Bad blocks in the XBN area due to data errors'
94                XBLEN  =  .-6$
95                :
96                : RETRIED BLOCKS
97                :
98 003516 000025          .WORD  RELEN          :LENGTH
99 003517 003524          .WORD  REBUFE         :END OF BUFFER
100 003520 030013          7$:  .WORD  30013          :DUP WORD
101 003521                .BLKW  3              :CONVERT BUFFER
102                REBUFE =
103 003524 003524 102 154  .ASCIZ  'Blocks retried on the check pass'
104                RELEN  =  .-7$
105                .DSABL LC
106                :
107                :
108                : END OF LIST
109                :
110 003545 003000          .WORD  0
111                :
112                : BUILD AND SEND STAT RESPONSES
113                :
114 003546 104207 003172  SNDRES: MOV  #FINMSG,RO  :POINT TO 'COMPLETE' MESSAGE

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 79-2
 RCT CLEANUP OVERLAY (4)

115	003550	104201	000012	MOV	#LFINMS,R1				:LENGTH OF MESSAGE
116	003552	060016		XFC	MAINTR				:SEND IT
117	003553	104204	003272	MOV	#RESTAB,R4				:POINT TO TABLE
118	003555	104203	003163	MOV	#PARMTB,R3				:PARAMETER TABLE
119	003557	104302	001313	MOV	LBNBAD,R2				:LBN'S BAD
120	003561	100232		MOV	R2,(R3)+				:STORE IT
121	003562	107302	001264	SUB	DMBUF,R2				:SUBTRACT STORED SECONDARIES TO GET PRIMARY
122	003564	100232		MOV	R2,(R3)+				:STORE IT
123	003565	104302	001264	MOV	DMBUF,R2				:STORED SECONDARY COUNT
124	003567	100232		MOV	R2,(R3)+				:STORE IT
125	003570	104302	001314	MOV	RCTBAD,R2				:RCT BAD BLOCKS
126	003572	100232		MOV	R2,(R3)+				:STORE IT
127	003573	104302	001315	MOV	DBBAD,R2				:DBN BAD BLOCKS
128	003575	100232		MOV	R2,(R3)+				:STORE IT
129	003576	104302	001316	MOV	XBBAD,R2				:XBN BAD BLOCKS
130	003600	100232		MOV	R2,(R3)+				:STORE IT
131	003601	104302	001470	MOV	RTYCNT,R2				:RETRIES
132	003603	100232		MOV	R2,(R3)+				:STORE IT
133	003604	104203	003163	MOV	#PARMTB,R3				:POINT BACK TO BEGINNING
134	003606	104242		MOV	(R4)+,R2				:GET LENGTH OF MESSAGE
135	003607	104245		SNDLP: MOV	(R4)+,R5				:GET END OF BUFFER ADDRESS (FOR CONVERT)
136	003610	023644		CALL	CLRBUF				:INITIALIZE THE BUFFER
137	003611	104047		MOV	R4,R0				:MOVE ADDRESS OF STRING TO R0
138	003612	103200	000400 001221	BIC	#FLIPON,FLAG1				:CLEAR FLAG (FOR CONVERT)
139	003615	104231		MOV	(R3)+,R1				:GET WORD TO CONVERT
140	003616	023660		CALL	DECASC				:CONVERT TO ASCII
141	003617	104021		MOV	R2,R1				:GET LENGTH IN R1
142	003620	060016		XFC	MAINTR				:SEND TO THE HOST
143	003621	105024		ADD	R2,R4				:POINT TO THE NEXT MESSAGE
144	003622	104242		MOV	(R4)+,R2				:GET LENGTH
145	003623	053607		BNE	SNDLP				:BRANCH IF NOT ZERO
146	003624	102200	002000 001220	BIT	#BSTGS,FLAG				:DID WE USE FCT ?
147	003627	013636		BEQ	USDFCT				:YES - PRINT THAT MESSAGE
148	003630	104207	003220	MOV	#FCTNOT,R0				:POINT TO NOT USED MESSAGE
149	003632	104201	000012	MOV	#LFCTNT,R1				:LENGTH OF MESSAGE
150	003634	060016		XFC	MAINTR				:SEND IT
151	003635	003643		BR	DONFCT				:EXIT
152	003636	104207	003204	USDFCT: MOV	#FCTUSD,R0				:POINT TO USED MESSAGE
153	003640	104201	000014	MOV	#LFCTUS,R1				:LENGTH OF MESSAGE
154	003642	060016		XFC	MAINTR				:SEND IT
155	003643	000000		DONFCT: RETURN					:RETURN

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 80
 RCT CLEANUP OVERLAY (G4)

1					
2					
3					
4					
5					
6	003644			CLRBUF:	PUSH R5,R2 ;SAVE R5
7	003646	104207	000003		MOV #3,R0 ;WORD COUNT
8	003650	104202	020040		MOV #BLANWD,R2 ;WORD OF 2 ASCII BLANKS
9	003652	100452		CLR:LP:	MOV R2,-(R5) ;STORE WORD IN BUFFER
10	003653	117407			DEC R0 ;DECREMENT COUNT
11	003654	053652			BNE CLR:LP ;LOOP TILL DONE
12	003655				POP R2,R5 ;RESTORE R5
13	003657	000000			RETURN

JDAFM - UDA FORMATTER CMACR X04.01 23-AUG-82 14:02:32 PAGE 81
 RCT CLEANUP OVERLAY (G4)

1									
2									
3									
4									
5									
6									
7	003660								
8	003664	104010	000736						
9	003666	114000	000737						
10	003670	104200	000012	000731					
11	003673	114000	000732						
12	003675	104204	000736						
13	003677	104203	000731						
14	003701	021565							
15	003702	104132							
16	003703	105202	000060						
17	003705	102200	000400	001221					
18	003710	053716							
19	003711	104027							
20	003712	101200	000400	001221					
21	003715	003724							
22	003716	110707			1\$:				
23	003717	101027							
24	003720	100457							
25	003721	103200	000400	001221					
26	003724	104141			2\$:				
27	003725	053664							
28	003726	102200	000400	001221					
29	003731	013736							
30	003732	110707							
31	003733	101207	000040						
32	003735	100457							
33	003736				3\$:				
34	003742	000000							

CONVERT FROM BINARY TO DECIMAL ASCII

R1 = VALUE TO BE CONVERTED
 R5 -> END OF BUFFER TO PUT RESULT

```

DECASC: PUSH    R0,R2,R3,R4      ;SAVE SOME REGS
DECALP: MOV     R1,TEMP          ;STORE VALUE TO BE CONVERTED
        CLR     TEMP+1          ;CLEAR HIGH ORDER
        MOV     #10,DDUMMY      ;FOR DIVIDE BY 10
        CLR     DDUMMY+1        ;CLEAR HIGH ORDER
        MOV     #TEMP,R4        ;DIVIDENT
        MOV     #DDUMMY,R3      ;DIVISOR
        CALL    DDIV            ;DO THE DIVIDE
        MOV     (R3),R2         ;GET REMAINDER (VALUE OF INTEREST)
        ADD     #'0,R2          ;MAKE IT ASCII
        BIT     #FLIPON,FLAG1   ;WHICH BYTE ARE WE ON ?
        BNE    1$              ;IF SET - THEN HIGH BYTE
        MOV     R2,RO           ;IF LOW BYTE - SAVE IN RO
        BIS     #FLIPON,FLAG1   ;SET NOW ON HIGH BYTE
        BR     2$              ;CONTINUE ON
1$:     SWAB    RO              ;GET FIRST DIGIT INTO HIGH BYTE
        BIS     R2,RO           ;OR IN LOW BYTE
        MOV     RO,-(R5)        ;STORE IT IN BUFFER
        BIC     #FLIPON,FLAG1   ;SET NOW ON LOW BYTE
        MOV     (R4),R1        ;GET QUOTIENT
        BNE    DECALP          ;IF NOT ZERO THEN CONTINUE
        BIT     #FLIPON,FLAG1   ;WHICH BYTE DID WE END ON ?
        BEQ    3$              ;IF CLEAR THEN - WE ARE FINISHED
        SWAB    RO              ;GET DIGIT IN HIGH ORDER
        BIS     #LOBL,RO        ;SET IN LOW ORDER BLANK
        MOV     RO,-(R5)        ;AND WRITE OUT LOW BYTE
        POP     R4,R3,R2,RO     ;RESTORE REGS
        RETURN
  
```

UDAFM - UCA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 82
 RCT CLEANUP OVERLAY (G4)

```

1
2
3
4 003743
5 003751 104204 006204
6 003753 104300 001476 000736
7 003756 104642 000002
8 003760 104643 000003
9 003762 101203 060000
10 003764 104205 007321
11 003766 104201 000200
12 003770 100252
13 003771 100253
14 003772 117401
15 003773 053770
16 003774 104203 001053
17 003776 104632 000011
18 004000 03202 177400
19 004002 104207 001424
20 004004 100672 000004
21 004006 104632 000001
22 004010 103202 007777
23 004012 100672 000001
24 004014 114002
25 004015 100672 000000
26 004017 024130
27 004020 104207 000721
28 004022 104203 122400
29 004024 104302 001113
30 004026 101023
31 004027 100673 000004
32 004031 104202 007321
33 004033 022600
34 004034 100623 000400
35 004036 100672 000001
36 004040 104143
37 004041 100673 000002
38 004043 104643 000001
39 004045 105303 001323
40 004047 100673 000003
41 004051 104203 000726
42 004053 100673 000005
43 004055 104040 000731
44 004057 104303 001321
45 004061 104304 001322
46 004063 104302 000740
47 004065 104207 000721
48 004067 101207 100000
49 004071 060012
50 004072 060003
51 004073 115001
52 004074 014104
53 004075 106200 000010 001501
54 004100 014104
55 004101 115400 001501
56 004103 004057
57 004104

:
: PROCESS REVECTOR BLOCK
:
CLEWRT: PUSHA
MOV #REVBUF,R4 ;POINT TO BUFFER
MOV COUNT,TEMP ;GET COUNT
CLHERE: MOV 2(R4),R2 ;GET LOW ORDER RBN
MOV 3(R4),R3 ;GET HIGH ORDER RBN
BIS #HD.RBN,R3 ;SET IN HDR CODE
MOV #CLBUF,R5 ;POINT TO BUFFER
MOV #RBNRPT,R1 ;INIT COUNTER
WLOOP: MOV R2,(R5)+ ;STORE LOW ORDER
MOV R3,(R5)+ ;STORE HIGH ORDER
DEC R1 ;DECREMENT COUNTER
BNE WLOOP ;CONTINUE TILL DONE
MOV #SCR,R3 ;POINT TO CHARACTERISTICS
MOV LBNTK(R3),R2 ;GET LBN/TRACK
BIC #HIBYTE,R2 ;CLEAR HIGH BYTE
MOV #CONBLK,R0 ;POINT TO CONVERT BLOCK
MOV R2,V3(R0) ;FOR CONVERT
MOV STCYL(R3),R2 ;STARTING CLYLINDER
BIC #LO,R2 ;CLEAR REST OF WORD
MOV R2,V1+1(R0) ;STORE
CLR R2 ;FOR STORE
MOV R2,V1(R0) ;LOW ALWAYS ZERO
CALL CS ;CONVERT AND SEEK
MOV #WRBLK,R0 ;POINT TO COMMAND BLOCK
MOV #WRCMD,R3 ;GET WRITE COMMAND
MOV CURTRK,R2 ;GET CURRENT TRACK
BIS R2,R3 ;SET TRACK FOR WRITE
MOV R3,RW.CMD(R0) ;STORE IN COMMAND BLOCK
MOV #CLBUF,R2 ;POINT TO BUFFER
CALL CEDC ;COMPUTE EDC - RETURNED IN R3
MOV R3,RW.EDC(R2) ;STORE IT
MOV R2,RW.BUF(R0) ;STICK IN COMMAND BLOCK
MOV (R4),R3 ;GET LOW ORDER HEADER
MOV R3,RW.LOW(R0) ;STORE IN WRITE BLOCK
MOV 1(R4),R3 ;GET HIGH ORDER
ADD ST.LBN,R3 ;ADD STARTING LBN BITS
MOV R3,RW.HI(R0) ;STORE IN WRITE BLOCK
MOV #HSLIM-1,R3 ;GET DUMMY SDI POINTER
MOV R3,RW.DUM(R0) ;POINT IN COMMAND BLOCK
MOV R4,DDUMMY ;SAVE R4
WRITE9: MOV HPREA,R3 ;GET HEADER PREAMBLE
MOV DPREA,R4 ;GET DATA PREAMBLE
MOV UNIT,R2 ;SET UNIT
MOV #WRBLK,R0 ;MAKE SURE POINTING AT BLOCK
BIS #BIT15,R0 ;SET NO REVECTORING
XFC SIP ;WAIT FOR SECTOR PULSE
XFC WRITE ;WRITE SECTOR
TST R1 ;ANY ERROR ?
BEQ 1$ ;NO - SKIP RETRY
CMP #MAXTRY, TMPTRY ;MAX ?
BEQ 1$ ;YES - GIVE UP
INC TMPTRY ;INC RETRY COUNT
BR WRITE9 ;DO RETRY
1$:

```


UDAFM - UDA FORMATTER DMA:R X04.01 23-AUG-82 14:02:32 PAGE 83
 RCT CLEANUP OVERLAY (G4)

1	004175	104204	006204		RBNWRT:	MOV	#REVBUFF,R4	:	POINT TO BUFFER
2	004177	104203	005567			MOV	#GOBLK,R3	:	POINT TO GOOD BLOCK
3	004201	104302	001446			MOV	EDC,R2	:	GET GOOD EDC
4	004203	100632	000400			MOV	R2,RW.EDC(R3)	:	STORE IT
5	004205	104203	001053			MOV	#SCR,R3	:	POINT TO CHARACTERISTICS
6	004207	104632	000004			MOV	HBNTRK(R3),R2	:	GET RBN/TRACK
7	004211	103202	177600			BIC	#HI1BYTE,R2	:	CLEAR HIGH GARBAGE
8	004213	104207	001424			MOV	#CONBLK,R0	:	POINT TO CONVERT BLOCK
9	004215	100672	000004			MOV	R2,V3(R0)	:	FOR CONVERT
10	004217	104632	000011			MOV	LBNTRK(R3),R2	:	GET LBN/TRACK
11	004221	103202	177400			BIC	#HI1BYTE,R2	:	CLEAR HIGH BYTE
12	004223	100672	000005			MOV	R2,V4(R0)	:	SET UP FOR RBN'S
13	004225	104632	000001			MOV	STCYL(R3),R2	:	STARTING CLYLINDER
14	004227	103202	007777			BIC	#LO,R2	:	CLEAR REST OF WORD
15	004231	100672	000001			MOV	R2,V1+1(R0)	:	STORE
16	004233	114002				CLR	R2	:	FOR STORE
17	004234	100672	000000			MOV	R2,V1(R0)	:	LOW ALWAYS ZERO
18	004236	104640	000002	000733	RNWHER:	MOV	2(R4),TEMP2	:	GET LOW ORDER RBN
19	004241	104640	000003	000734		MOV	3(R4),TEMP2+1	:	GET HIGH ORDER
20	004244	104040	000731			MOV	R4,DDUMMY	:	SAVE R4
21	004246	104204	000733			MOV	#TEMP2,R4	:	FOR CONVERT
22	004250	022665				CALL	CVTSK	:	CONVERT AND SEEK
23	004251	104207	000721			MOV	#WRBLK,R0	:	POINT TO COMMAND BLOCK
24	004253	104203	122400			MOV	#WRCMD,R3	:	GET WRITE COMMAND
25	004255	104302	001113			MOV	CURTRK,R2	:	GET CURRENT TRACK
26	004257	101023				BIS	R2,R3	:	SET TRACK FOR WRITE
27	004260	100673	000004			MOV	R3,RW.CMD(R0)	:	STORE IN COMMAND BLOCK
28	004262	104202	005567			MOV	#GOBLK,R2	:	POINT TO BLOCK
29	004264	100672	000001			MOV	R2,RW.BUF(R0)	:	STICK IN COMMAND BLOCK
30	004266	104143				MOV	(R4),R3	:	GET LOW ORDER HEADER
31	004267	100673	000002			MOV	R3,RW.LOW(R0)	:	STORE IN WRITE BLOCK
32	004271	104640	000001			MOV	1(R4),R3	:	GET HIGH ORDER
33	004273	105303	001324			ADD	ST.RBN,R3	:	ADD STARTING RBN BITS
34	004275	101203	060000			BIS	#HD.RBN,R3	:	GIVE RBN HEADER
35	004277	100673	000003			MOV	R3,RW.HI(R0)	:	STORE IN WRITE BLOCK
36	004301	104203	000726			MOV	#HSLIM-1,R3	:	GET DUMMY SDI POINTER
37	004303	100673	000005			MOV	R3,RW.DUM(R0)	:	POINT IN COMMAND BLOCK
38	004305	104303	001321		WRIT13:	MOV	HPREA,R3	:	GET HEADER PREAMBLE
39	004307	104304	001322			MOV	DPREA,R4	:	GET DATA PREAMBLE
40	004311	104302	000740			MOV	UNIT,R2	:	SET UNIT
41	004313	104207	000721			MOV	#WRBLK,R0	:	MAKE SURE POINTING AT BLOCK
42	004315	101207	100000			BIS	#BIT15,R0	:	SET NO REVECTORING
43	004317	060012				XFC	SIP	:	WAIT FOR SECTOR PULSE
44	004320	060003				XFC	WRITE	:	WRITE SECTOR
45	004321	115001				TST	R1	:	ANY ERROR ?
46	004322	014346				BEQ	2\$:	NO - SKIP RETRY
47	004323	106300	001477	001501		CMP	RETRY,IMPTRY	:	MAX ?
48	004326	014332				BEQ	1\$:	YES - TRY SOME RECOVERY
49	004327	115400	001501			INC	IMPTRY	:	INC RETRY COUNT
50	004331	004305				BR	WRIT13	:	DO RETRY
51	004332	104303	001502		1\$:	MOV	RECTMP,R3	:	GET CURRENT ERROR RECOVERY LEVEL
52	004334	074346				BMI	2\$:	IF NEGATIVE THEN FRIED
53	004335	115000	001500			TST	RECOV	:	IS THERE ONLY RECOVERY LEVEL 0 ?
54	004337	014341				BEQ	3\$:	YES - NO NEED TO ISSUE IT - JUST RETRY
55	004340	022571				CALL	ERRHND	:	TRY RECOVERY
56	004341	114000	001501		3\$:	CLR	IMPTRY	:	FOR INIT
57	004343	117400	001502			DEC	RECTMP	:	DECREMENT IT

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 83-1
 RCT CLEANUP OVERLAY (G4)

58	004345	004305		BR	WRIT13		;RETRY
59	004346					2\$:	
60	004346	114000	001501	CLR	TMPTRY		;RESET
61	004350	104300	001500	MOV	RECOV,RECTMP	001502	;DITTO
62	004353	104304	000731	MOV	DDUMMY,R4		;RESTORE R4
63	004355	105204	000004	ADD	#REVLEN,R4		;POINT TO NEXT ENTRY
64	004357	115400	000736	INC	TEMP		;INC COUNTER
65	004361	106200	000100	CMP	#64.,TEMP	000736	;DONE ?
66	004364	054236		BNE	RNWHÉR		;NO - REPEAT
67	004365	104207	001424	MOV	#CONBLK,RO		;FOR RESET
68	004367	114002		CLR	R2		;DITTO
69	004370	100672	000005	MOV	R2,V4(RO)		;RESET FOR NON-RBN'S
70	004372	000000		RETURN			

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 84-1
 RCT CLEANUP OVERLAY (G4)

58	004527	115400	001501		INC	TMPTRY		:INC RETRY COUNT
59	004531	004471			BR	READ8		:DO RETRY
60	004532	104303	001502	1\$:	MOV	RECTMP,R3		:GET CURRENT ERROR RECOVERY LEVEL
61	004534	074546			BMI	2\$:IF NEGATIVE THEN FRIED
62	004535	115000	001500		TST	RECOV		:IS THERE ONLY RECOVERY LEVEL 0 ?
63	004537	014541			BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY
64	004540	022571			CALL	ERRHND		:TRY RECOVERY
65	004541	114000	001501	3\$:	CLR	TMPTRY		:FOR INIT
66	004543	117400	001502		DEC	RECTMP		:DECREMENT IT
67	004545	004471			BR	READ8		:RETRY
68	004546			2\$:				
69	004546	115405			INC	R5		:INCREMENT BAD COUNTER
70	004547	106305	001245		CMP	FCTCPY,R5		:ALL BAD ?
71	004551	014705			BEQ	RFTL		:YUP - ALL OVER
72	004552	104203	001243		MOV	#RCTFMT,R3		:SIZE OF TABLE - R4 -> BLOCK NUMBER
73	004554	021503			CALL	DADD		:ADD TO POINT TO NEXT COPY
74	004555	114000	001501		CLR	TMPTRY		:RESET RETRY LEVEL
75	004557	104300	001500	001502	MOV	RECOV,RECTMP		:DITTO RECOVERY LEVELS
76	004562	004436			BR	RCL		:BRANCH BACK
77	004563			102\$:				
78	004563	114000	001501	RCD:	CLR	TMPTRY		:FOR RESET
79	004565	104300	001500	001502	MOV	RECOV,RECTMP		:GET RECOVERY LEVELS
80	004570	115005			TST	R5		:ANY ERRORS ?
81	004571	014676			BEQ	RLD		:NO - EXIT
82	004572	104203	001241		MOV	#FCTFMT,R3		:SIZE OF TABLE
83	004574	021521			CALL	DSUB		:GET BACK TO PREVIOUS COPY
84	004575	024130			CALL	CS		:CONVERT AND SEEK
85	004576	104207	000721		MOV	#WRBLK,R0		:POINT TO COMMAND BLOCK
86	004600	104203	122400		MOV	#WRCMD,R3		:GET WRITE COMMAND
87	004602	104302	001113		MOV	CURTRK,R2		:GET CURRENT TRACK
88	004604	101023			BIS	R2,R3		:SET TRACK FOR WRITE
89	004605	100673	000004		MOV	R3,RW,CMD(R0)		:STORE IN COMMAND BLOCK
90	004607	104203	006621		MOV	#RCTBUF,R3		:POINT TO BUFFER
91	004611	100673	000001		MOV	R3,RW,BUF(R0)		:STICK IN COMMAND BLOCK
92	004613	104143			MOV	(R4),R3		:GET LOW ORDER HEADER
93	004614	100673	000002		MOV	R3,RW,LOW(R0)		:STORE IN WRITE BLOCK
94	004616	104643	000001		MOV	1(R4),R3		:GET HIGH ORDER
95	004620	105303	001323		ADD	ST.LBN,R3		:ADD STARTING LBN BITS
96	004622	100673	000003		MOV	R3,RW,HI(R0)		:STORE IN WRITE BLOCK
97	004624	104203	000726		MOV	#HSLIM-1,R3		:GET DUMMY SDI POINTER
98	004626	100673	000005		MOV	R3,RW,DUM(R0)		:POINT IN COMMAND BLOCK
99	004630	104303	001321	WRIT10:	MOV	HPREA,R3		:GET HEADER PREAMBLE
100	004632	104304	001322		MOV	DPREA,R4		:GET DATA PREAMBLE
101	004634	104302	000740		MOV	UNIT,R2		:SET UNIT
102	004636	104207	000721		MOV	#WRBLK,R0		:MAKE SURE POINTING AT BLOCK
103	004640	101207	100000		BIS	#BIT15,R0		:SET NO REVECTORING
104	004642	060012			XFC	SIP		:WAIT FOR SECTOR PULSE
105	004643	060003			XFC	WRITE		:WRITE SECTOR
106	004644	115001			TST	R1		:ANY ERROR ?
107	004645	014671			BEQ	2\$:NO - SKIP RETRY
108	004646	106300	001477	001501	CMP	RETRY,TMPTRY		:MAX ?
109	004651	014655			BEQ	1\$:YES - TRY SOME RECOVERY
110	004652	115400	001501		INC	TMPTRY		:INC RETRY COUNT
111	004654	004630			BR	WRIT10		:DO RETRY
112	004655	104303	001502	1\$:	MOV	RECTMP,R3		:GET CURRENT ERROR RECOVERY LEVEL
113	004657	074671			BMI	2\$:IF NEGATIVE THEN FRIED
114	004660	115000	001500		TST	RECOV		:IS THERE ONLY RECOVERY LEVEL 0 ?

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 85
FINAL CHECK OVERLAY (H2)

1
2 004711
3
4
5
6
7
8
9
10 003023 104200 000066 001154
11 003026 023604
12 003027 023032
13 003030 023323
14 003031 000000

.....

.SBTTL FINAL CHECK OVERLAY (H2)
DMOVL H2,START

DO CHECK OF:

FCT
RCT
CORRECT HEAD ADDRESSING

FINCHK: MOV #H2,CUROVL
CALL VERHD
CALL FCTCK
CALL RCTCK
RETURN

;MAKE THIS THE CURRENT OVERLAY
;VERIFY ALL HEADS ACCESSIBLE
;VERIFY FCT
;VERIFY RCT

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 86-1
 :NAL CHECK OVERLAY (H2)

58	003202	104173				MOV	(R0),R3		:GET STATUS WORD
59	003203	102203	010000			BIT	#ECCF,R3		:ECC ERROR ?
60	003205	013211				BEQ	101\$:NOPE - VERIFY EDC
61	003206	023000				CALL	ECCCK		:CORRECT ECC
62	003207	115001				TST	R1		:TEST FLAG
63	003210	053217				BNE	100\$:UNCORRECTABLE
64	003211	104202	006204		101\$:	MOV	#PRMBUF,R2		:POINT TO BUFFER
65	003213	022600				CALL	CEDC		:COMPUTE EDC
66	003214	106623	000400			CMP	RW,EDC(R2),R3		:O.K. ?
67	003216	013242				BEQ	102\$:YUP - CONSIDER GOOD
68	003217	106300	001477	001501	100\$:	CMP	RETRY,TMPTRY		:MAX ?
69	003222	013226				BEQ	1\$:YES - TRY SOME RECOVERY
70	003223	115400	001501			INC	TMPTRY		:INC RETRY COUNT
71	003225	003164				BR	READ9		:DO RETRY
72	003226	104303	001502		1\$:	MOV	RECTMP,R3		:GET CURRENT ERROR RECOVERY LEVEL
73	003230	073243				BMI	YES		:IF NEGATIVE THEN FRIED
74	003231	115000	001500			TST	RECOV		:IS THERE ONLY RECOVERY LEVEL 0 ?
75	003233	013235				BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY
76	003234	022571				CALL	ERRHND		:TRY RECOVERY
77	003235	114000	001501		3\$:	CLR	TMPTRY		:FOR INIT
78	003237	117400	001502			DEC	RECTMP		:DECREMENT IT
79	003241	003164				BR	READ9		:RETRY
80	003242				102\$:				
81	003242				2\$:				
82	003242	115405				INC	R5		:NO - INCREMENT COUNTER
83	003243	115400	001246		YES:	INC	NEXT1		:INCREMENT IT
84	003245	106205	000002			CMP	#2,R5		:FOUND 2 GOOD ONES ?
85	003247	013267				BEQ	FCTCKD		:YUP - GO TO NEXT BLOCK
86	003250	104204	001114			MOV	#CURBN,R4		:FOR ADD
87	003252	104203	001241			MOV	#FCTFMT,R3		:FOR ADD
88	003254	021503				CALL	DADD		:POINT TO NEXT COPY
89	003255	114000	001501			CLR	TMPTRY		:RESET RETRY LEVEL
90	003257	104300	001500	001502		MOV	RECOV,RECTMP		:DITTO RECOVERY LEVELS
91	003262	106300	001245	001246		CMP	FCTCPY,NEXT1		:DONE THIS SECTOR ?
92	003265	053115				BNE	FCTCL1		:NO - WRITE NEXT FCT COPY
93	003266	003316				BR	FCTCKE		:2 NOT GOOD - TROUBLE
94	003267	102200	002000	001220	FCTCKD:	BIT	#BSTGS,FLAG		:BEST GUESS ?
95	003272	053315				BNE	4\$:YUP - ONLY CHECK FIRST BLOCK
96	003273	060022				XFC	UPDATE		:LET HOST KNOW STILL ALIVE
97	003274	115400	001257			INC	FCTCNT		:INCREMENT IT
98	003276					DUBINC	CURXBN		:INCREMENT IT
99	003303	104300	001116	001114		MOV	CURXBN,CURBN		:GET LOW ORDER
100	003306	104300	001117	001115		MOV	CURXBN+1,CURBN+1		:GET HIGH ORDER
101	003311	106300	001261	001257		CMP	FCTMPD,FCTCNT		:DONE ?
102	003314	053112				BNE	FCTCLP		:NOPE - DO NEXT SECTOR
103	003315	000000			4\$:	RETURN			
104	003316	104201	000011		FCTCKE:	MOV	#9,R1		:SIGNAL ERROR
105	003320	104302	001257			MOV	FCTCNT,R2		:BLOCK FAILED ON
106	003322	022542				CALL	ERRMNT		:ERROR RETURN

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 87-1
FINAL CHECK OVERLAY (M2)

58	003470	053477				BNE	100\$:UNCORRECTABLE
59	003471	104202	006204		101\$:	MOV	#PRMBUF,R2		:POINT TO BUFFER
60	003473	022600				CALL	CEDC		:COMPUTE EDC
61	003474	106623	000400			CMP	RW.EDC(R2),R3		:O.K. ?
62	003476	013522				BEQ	102\$:YUP - CONSIDER GOOD
63	003477	106300	001477	001501	100\$:	CMP	RETRY,TMPTRY		:MAX ?
64	003502	013506				BEQ	1\$:YES - TRY SOME RECOVERY
65	003503	115400	001501			INC	TMPTRY		:INC RETRY COUNT
66	003505	003445				BR	READ10		:DO RETRY
67	003506	104303	U01502		1\$:	MOV	RECTMP,R3		:GET CURRENT ERROR RECOVERY LEVEL
68	003510	073523				BMI	RCTNGD		:IF NEGATIVE SKIP GOOD INCREMENT
69	003511	115000	001500			TST	RECOV		:IS THERE ONLY RECOVERY LEVEL 0 ?
70	003513	013515				BEQ	3\$:YES - NO NEED TO ISSUE IT - JUST RETRY
71	003514	022571				CALL	ERRHND		:TRY RECOVERY
72	003515	114000	001501		3\$:	CLR	TMPTRY		:FOR INIT
73	003517	117400	001502			DEC	RECTMP		:DECREMENT IT
74	003521	003445				BR	READ10		:RETRY
75	003522				102\$:				
76	003522				2\$:				
77	003522	115405				INC	R5		:YUP - INCREMENT COUNTER
78	003523	115400	001246		RCTNGD:	INC	NEXT1		:INCREMENT IT
79	003525	114000	001501			CLR	TMPTRY		:FOR RESET
80	003527	104300	001500	001502		MOV	RECOV,RECTMP		:GET RECOVERY LEVELS
81	003532	106205	000002			CMP	#2,R5		:FOUND 2 GOOD ONES ?
82	003534	013554				BEQ	RCTCKD		:YUP - GO TO NEXT BLOCK
83	003535	104204	001114			MOV	#CURBN,R4		:FOR ADD
84	003537	104203	001243			MOV	#RCTFMT,R3		:FOR ADD
85	003541	021503				CALL	DADD		:POINT TO NEXT COPY
86	003542	114000	001501			CLR	TMPTRY		:RESET RETRY LEVEL
87	003544	104300	001500	001502		MOV	RECOV,RECTMP		:DITTO RECOVERY LEVELS
88	003547	106300	001245	001246		CMP	FCTCPY,NEXT1		:DONE THIS SECTOR ?
89	003552	053362				BNE	RCTCL1		:NO - READ NEXT FCT COPY
90	003553	003577				BR	RCTCKE		:2 NOT GOOD - TROUBLE
91	003554	060022			RCTCKD:	XFC	UPDATE		:LET HOST KNOW STILL ALIVE
92	003555	115400	001476			INC	COUNT		:INCREMENT IT
93	003557					DUBINC	CURLBN		:INCREMENT IT
94	003564	104300	001116	001114		MOV	CURLBN,CURBN		:GET LOW ORDER
95	003567	104300	001117	001115		MOV	CURLBN+1,CURBN+1		:GET HIGH ORDER
96	003572	106300	001262	001476		CMP	RCTLBN,COUNT		:DONE ?
97	003575	053357				BNE	RCTCLP		:NOPE - DO NEXT SECTOR
98	003576	000000				RETURN			
99	003577	104201	000013		RCTCKE:	MOV	#11.,R1		:SET ERROR CODE
100	003601	104302	001476			MOV	COUNT,R2		:RCT BLOCK FAILED ON
101	003603	022542				CALL	ERRHND		:ERROR RETURN

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 88-1
FINAL CHECK OVERLAY (H2)

58	003760	100673	000003		MOV	R3,RW.HI(R0)	:STORE IT BACK	
59	003762	101200	010000	001221	BIS	#BDTST,FLAG1	:SET THAT WE TRIED BAD HEADER CODE	
60	003765	003720			BR	5\$:AND TRY READ AGAIN	
61	003766	103200	010000	001221	1\$:	BIC	#BDTST,FLAG1	:CLEAR BAD HEADER TESTED FLAG
62	003771				DUBINC	CURXBN	:INCREMENT CURRENT SECTOR	
63	003776	117400	001451		DEC	SECCNT	:DECREMENT SECTOR/TRACK COUNT	
64	004000	053674			BNE	2\$:TRY IT AGAIN WITH NEXT SECTOR	
65	004001	104307	001471		MOV	CURPNT,R0	:POINTER TO INFO BLOCK	
66	004003	104303	001120		MOV	STASEC,R3	:STORE STARTING SECTOR NUMBER	
67	004005	100273			MOV	R3,(R0)+	:IN INFO BLOCK	
68	004006	104303	001121		MOV	STASEC+1,R3	:GET HIGH ORDER	
69	004010	100273			MOV	R3,(R0)+	:STORE IT	
70	004011	104303	001460		MOV	CURGRP,R3	:GET CURRENT GROUP	
71	004013	100273			MOV	R3,(R0)+	:STORE IT	
72	004014	104303	001113		MOV	CURTRK,R3	:GET CURRENT TRACK	
73	004016	100273			MOV	R3,(R0)+	:STORE IT	
74	004017	104070	001471		MOV	R0,CURPNT	:STORE POINT BACK	
75	004021	115400	001450		INC	ERRCNT	:INCREMENT ERROR COUNT	
76	004023	103200	010000	001221	3\$:	BIC	#BDTST,FLAG1	:CLEAR FLAG IN CASE IT WAS SET
77	004026	115400	001113		INC	CURTRK	:DO NEXT TRACK	
78	004030	104300	001130	0G0736	MOV	SECTRK,TEMP	:FOR ADD	
79	004033	114000	000737		CLR	TEMP+1	:DITTO	
80	004035	104203	000736		MOV	#TEMP,R3	:SET UP	
81	004037	104204	001120		MOV	#STASEC,R4	:DITTO	
82	004041	021503			CALL	DADD	:GET NEW STARTING SECTOR	
83	004042	104300	001120	001116	MOV	STASEC,CURXBN	:RESET CURRENT SECTOR	
84	004045	104300	001121	001117	MOV	STASEC+1,CURXBN+1	:RESET HIGH ORDER	
85	004050	104300	001130	001451	MOV	SECTRK,SECCNT	:RESET SECTORS/TRACK	
86	004053	117400	001462		DEC	TRKCNT	:DECREMENT COUNT OF TRACKS	
87	004055	053674			BNE	2\$:IF NOT DONE - DO NEXT TRACK	
88	004056	060022			XFC	UPDATE	:LET HOST KNOW WE'RE ALIVE	
89	004057	115400	001460		INC	CURGRP	:ELSE INCREMENT GROUP NUMBER	
90	004061	104207	001053		MOV	#SCR,R0	:POINT TO CHARACTERISTICS BLOCK	
91	004063	104673	000003		MOV	TRKGRP(R0),R3	:LOAD TRACKS/GROUP	
92	004065	103203	177400		BIC	#HIBYTE,R3	:CLEAR OUT HIGH GARBAGE	
93	004067	104030	001462		MOV	R3,TRKCNT	:STORE IN COUNTER	
94	004071	114000	001113		CLR	CURTRK	:RESET TRACK NUMBER	
95	004073	117400	001461		DEC	GRPCNT	:DONE ALL GROUPS ?	
96	004075	053660			BNE	4\$:NO - DO NEXT GROUP	
97	004076	115000	001450		TST	ERRCNT	:ANY PROBLEMS ?	
98	004100	014305			BEQ	15\$:NOPE - FINISHED	
99	004101	104205	006621		MOV	#BDLST,R5	:POINT TO BAD LIST	
100	004103	104250	001120	14\$:	MOV	(R5)+,STASEC	:GET STARTING SECTOR NUMBER	
101	004105	104250	001121		MOV	(R5)+,STASEC+1	:GET HIGH ORDER	
102	004107	104250	001460		MOV	(R5)+,CURGRP	:GET GROUP NUMBER	
103	004111	104250	001113		MOV	(R5)+,CURTRK	:GET TRACK NUMBER	
104	004113	104203	001053		MOV	#SCR,R3	:POINT TO CHARACTERISTICS	
105	004115	104670	000021	001455	MOV	XBNCYL(R0),CNTCYL	:GET NUMBER OF XBN CYLINDERS	
106	004120	117400	001455		DEC	CNTCYL	:DECREMENT FOR ONE ALREADY DONE	
107	004122	104300	001132	000736	13\$:	MOV	SECTCY,TEMP	:FOR ADD
108	004125	114000	000737		CLR	TEMP+1	:CLEAR HIGH ORDER	
109	004127	104203	000736		MOV	#TEMP,R3	:SET UP FOR ADD	
110	004131	104204	001120		MOV	#STASEC,R4	:DITTO	
111	004133	021503			CALL	DADD	:GET SECTOR NUMBER ON NEXT CYLINDER	
112	004134	104300	001130	001451	MOV	SECTRK,SECCNT	:RESET SECTOR COUNT	
113	004137	104300	001120	001116	MOV	STASEC,CURXBN	:RESET SECTOR NUMBER	
114	004142	104300	001121	001117	MOV	STASEC+1,CURXBN+1	:RESET HIGH ORDER	

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 88-2
FINAL CHECK OVERLAY (H2)

115	004145				DUBINC	CYLNUM	: INCREMENT CYLINDER NUMBER
116	004152	104300	001126	001077	MOV	CYLNUM, ISEEK+1	: GET LO ORDER WORD OF CYLINDER NUMBER
117	004155	104300	001127	001100	MOV	CYLNUM+1, ISEEK+2	: LOAD HIGH ORDER WORD OF CYL NUM
118	004160	104300	001460	001101	MOV	CURGRP, ISEEK+3	: LOAD GROUP NUMBER
119	004163	022242			CALL	SEEK	: SEEK TO CURRENT CYL NUM
120	004164	115001			TST	R1	: ANY ERRORS ?
121	004165	073334			BMI	SKERR	: YUP - QUIT
122	004166	104302	000740		MOV	UNIT, R2	: SDI INTERCONNECT
123	004170	104207	000721		MOV	#RDBLK, R0	: POINT TO READ COMMAND BLOCK
124	004172	104203	006204		MOV	#PRMBUF, R3	: BUFFER POINTER
125	004174	100673	000001		MOV	R3, RW.BUF(R0)	: STORE IT
126	004176	104303	001116		MOV	CURXBN, R3	: GET LOW RORDER BLOCK NUMBER
127	004200	100673	000002		MOV	R3, RW.LOW(R0)	: STORE IN COMMAND BLOCK
128	004202	104303	001117		MOV	CURXBN+1, R3	: LOAD HIGH ORDER BLOCK NUMBER
129	004204	105303	001325		ADD	ST.XBN, R3	: ADD STARTING XBN BITS
130	004206	101203	120000		BIS	#HD.XBN, R3	: SET IN XBN HEADER CODE
131	004210	100673	000003		MOV	R3, RW.HI(R0)	: STORE IN COMMAND BLOCK
132	004212	104203	013400		MOV	#RWCMD, R3	: GET READ COMMAND
133	004214	101303	001113		BIS	CURTRK, R3	: SET IN TRACK NUMBER
134	004216	100673	000004		MOV	R3, RW.CMD(R0)	: STORE IN COMMAND BLOCK
135	004220	104203	000726		MOV	#HSLIM-1, R3	: POINTER TO DUMMY SDI BLOCK
136	004222	100673	000005		MOV	R3, RW.DUM(R0)	: STORE IT IN READ BLOCK
137	004224	104207	100721		MOV	#<BIT15!RDBLK>, R0	: MAKE SURE POINTING AT BLOCK
138	004226	104203	100000		MOV	#RDCMD, R3	: RESET STATUS POINTER
139	004230	100673	000000		MOV	R3, RW.STAT(R0)	: STORE IT BACK
140	004232	060012			XFC	SIP	: SYNCH WITH SECTOR/INDEX PULSE
141	004233	060002			XFC	READ	: READ 1 SECTOR
142	004234	115001			TST	R1	: ANY ERROR ?
143	004235	014277			BEQ	10\$: NO - THIS HEAD O.K.
144	004236	102200	010000	001221	BIT	#BDTST, FLAG1	: HAVE WE TESTED BAD HEADER CODE ?
145	004241	054260			BNE	17\$: YUP - DON'T TRY AGAIN
146	004242	104207	000721		MOV	#RDBLK, R0	: POINT TO READ CONTROL BLOCK
147	004244	104673	000003		MOV	RW.HI(R0), R3	: GET HEADER WORD
148	004246	103203	170000		BIC	#HD.CLR, R3	: CLEAR HEADER
149	004250	101203	110000		BIS	#HD.BAD, R3	: SET IN BAD HEADER CODE
150	004252	100673	000003		MOV	R3, RW.HI(R0)	: STORE IT BACK
151	004254	101200	010000	001221	BIS	#BDTST, FLAG1	: SET THAT WE TRIED BAD HEADER CODE
152	004257	004212			BR	16\$: AND TRY READ AGAIN
153	004260	103200	010000	001221	BIC	#BDTST, FLAG1	: CLEAR BAD HEADER TESTED FLAG
154	004263				DUBINC	CURXBN	: INCREMENT CURRENT SECTOR
155	004270	117400	001451		DEC	SECCNT	: DECREMENT SECTOR/TRACK COUNT
156	004272	054166			BNE	11\$: TRY IT AGAIN WITH NEXT SECTOR
157	004273	117400	001455		DEC	CNTCYL	: DECREMENT CYLINDER COUNT
158	004275	014306			BEQ	12\$: FAILED ON ALL SECTORS - SEND WARNING
159	004276	004122			BR	13\$: TRY NEXT CYLINDER
160	004277	103200	006621	001221	BIC	#BDLST, FLAG1	: CLEAR FLAG IN CASE IT WAS SET
161	004302	117400	001450		DEC	ERRCNT	: DECREMENT ERROR COUNT
162	004304	054103			BNE	14\$: CONTINUE TILL DONE ALL
163	004305	000000			RETURN		: ALL DONE RETURN
164	004306	104207	003232		MOV	#WRN, R0	: ADDRESS OF WARNING
165	004310	104201	000040		MOV	#WRNLN, R1	: LENGTH OF WARNING
166	004312	060016			XFC	MAINTR	: SEND IT
167	004313	004305			BR	15\$: RETURN

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 90
FCT WRITE OVERLAY (F9)

```

1          .SBTTL FCT WRITE OVERLAY (F9)
2 004361   DMOVLY F9,START
3          :
4          :
5          :
6          :
7 003C23   104200 000030 001154 FCTWRT: MOV    #F9,CUROVL   :FOR INIT
8 003026   114005          CLR     R5          :CLEAR ERROR COUNTER
9 003027   104050 001246          MOV    R5,NEXT1     :INIT NEXT COPY COUNTER
10 003031   104204 001257          MOV    #FCTCNT,R4   :POINT TO FCT BLOCK NUMBER
11 003033   104203 001463          MOV    #ONE,R3      :FOR SUB
12 003035   021521          CALL   CSJUB        :SUB TO GET RIGHT ONE
13 003036   104305 001254          MOV    BUFPNT,R5    :GET BUFFER POINTER
14 003040   104303 001144          FCTRLP: MOV   LBNCYL,R3  :GET LBN CYLINDERS
15 003042   104207 001424          MOV    #CONBLK,R0   :POINT TO CONVERT BLOCK
16 003044   100673 000000          MOV    R3,V1(R0)    :STORE FOR CONVERT
17 003046   104303 001145          MOV    LBNCYL+1,R3  :GET HIGH ORDER
18 003050   100673 000001          MOV    R3,V1+1(R0)  :STORE IT
19 003052   104303 001130          MOV    SECTRK,R3    :GET SECTORS/TRACK
20 003054   100673 000004          MOV    R3,V3(R0)    :STORE IT
21 003056   022665          CALL   CVTSK        :CONVERT AND SEEK
22 003057   104207 000721          MOV    #WRBLK,R0    :POINT TO COMMAND BLOCK
23 003061   104203 122400          MOV    #WRCMD,R3    :GET WRITE COMMAND
24 003063   104302 001113          MOV    CURTRK,R2    :GET CURRENT TRACK
25 003065   101023          BIS    R2,R3        :SET TRACK FOR WRITE
26 003066   100673 000004          MOV    R3,RW.CMD(R0):STORE IN COMMAND BLOCK
27 003070   104052          MOV    R5,R2        :POINT TO BUFFER
28 003071   022600          CALL   CEDC         :COMPUTE EDC - RETURNED IN R3
29 003072   100623 000400          MOV    R3,RW.EDC(R2):STORE IT
30 003074   100672 000001          MOV    R2,RW.BUF(R0):STICK IN COMMAND BLOCK
31 003076   104143          MOV    (R1),R3      :GET LOW ORDER HEADER
32 003077   100673 000002          MOV    R3,RW.LOW(R0):STORE IN WRITE BLOCK
33 003101   104643 000001          MOV    1(R4),R3     :GET HIGH ORDER
34 003103   105303 001325          ADD    ST.XBN,R3    :ADD STARTING XBN BITS
35 003105   101203 120000          BIS    #HD.XBN,R3   :SET HEADER
36 003107   100673 000003          MOV    R3,RW.HI(R0) :STORE IN WRITE BLOCK
37 003111   104203 000726          MOV    #HSLIM-1,R3  :GET DUMMY SDI POINTER
38 003113   100673 000005          MOV    R3,RW.DUM(R0):POINT IN COMMAND BLOCK
39 003115   104303 001321          WRITES: MOV   HPREA,R3 :GET HEADER PREAMBLE
40 003117   104304 001322          MOV    DPREA,R4     :GET DATA PREAMBLE
41 003121   104302 000740          MOV    UNIT,R2      :SET UNIT
42 003123   104207 000721          MOV    #WRBLK,R0    :MAKE SURE POINTING AT BLOCK
43 003125   101207 100000          BIS    #BIT15,R0    :SET NO RVECTORING
44 003127   060012          XFC    SIP          :WAIT FOR SECTOR PULSE
45 003130   060003          XFC    WRITE        :WRITE SECTOR
46 003131   115001          TST    R1           :ANY ERROR ?
47 003132   013157          BEQ    FVGOOD       :NOPE
48 003133   106300 001477 001501          CMP    RETRY,IMPTRY :MAX ?
49 003136   013142          BEQ    1$           :YES - TRY SOME RECOVERY
50 003137   115400 001501          INC    IMPTRY       :INC RETRY COUNT
51 003141   003115          BR     WRITES       :DO RETRY
52 003142   104303 001502          1$:  MOV    RECTMP,R3   :GET CURRENT ERROR RECOVERY LEVEL
53 003144   073156          BMI    2$           :IF NEGATIVE THEN FRIED
54 003145   115000 001500          TST    RECOV        :IS THERE ONLY RECOVERY LEVEL 0 ?
55 003147   013151          BEQ    3$           :YES - NO NEED TO ISSUE IT - JUST RETRY
56 003150   022571          CALL   ERRHND       :TRY RECOVERY
57 003151   114000 001501          3$:  CLR    IMPTRY     :FOR INIT

```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 91
 PBN->D,X,L,RBN CONVERSION OVERLAY (G5)

```

1          .SBTTL PBN->D,X,L,RBN CONVERSION OVERLAY (G5)
2 003227  DMOVLY G5,START
3          :
4          :
5          :
6 003023  103200  170000  001112  PCON:  BIC      #HD.CLR,CURPBN+1  ;CLEAR THE HEADER
7 003026  104203  001155          MOV      #HGHPBN,R3    ;HIGHEST PBN IN LBN AREA
8 003030  104204  001111          MOV      #CURPBN,R4    ;CURRENT PBN
9 003032  021623          CALL     CCMP           ;IS IT LBN OR RBN ?
10 003033  033117          BPL     LRBN           ;YUP - GO COMPUTE IT
11 003034  021521          CALL     DSUB          ;SUBTRACT HIGH LBN PBN
12 003035  104203  001140          MOV      #XBNSEC,R3   ;TOTAL XBN SECTORS
13 003037  021623          CALL     DCMP          ;IS IT AN XBN ?
14 003040  013042          BEQ     DBNFND        ;NOPE - THE FIRST DBN
15 003041  033073          BPL     XBNFND        ;YUP - GO FIXIT
16 003042  104204  001111          DBNFND: MOV      #CURPBN,R4    ;FALSE DBN - GET VALUE
17 003044  104203  001140          MOV      #XBNSEC,R3   ;TOTAL XBN SECTORS
18 003046  021521          CALL     DSUB          ;SUBTRACT TO GET RELATIVE DBN
19 003047  023250          CALL     CONGRP        ;COMPUTE THE "OFFSET" SECTOR
20 003050  104203  001130          MOV      #SECTRK,R3   ;SECTORS/TRACK
21 003052  104204  001111          MOV      #CURPBN,R4   ;TRACK
22 003054  021537          CALL     DMUL          ;MULTIPLY TO GET STARTING PBN ON TRACK
23 003055  104203  000731          MOV      #DDUMMY,R3   ;SECTOR ON TRACK
24 003057  021503          CALL     DADD          ;ADD TO GET ACTUAL PBN
25 003060  104641  000001          MOV      1(R4),R1     ;GET HIGH ORDER
26 003062  105301  001326          ADD     ST.DBN,R1     ;ADD TO GET ABSOLUTE DBN
27 003064  103201  170000          BIC     #HD.CLR,R1    ;CLEAR THE HEADER
28 003066  101201  140000          BIS     #HD.DBN,R1    ;MARK AS DBN
29 003070  104010  001112          MOV     R1,CURPBN+1   ;STORE BACK
30 003072  003241          BR      PDONE         ;CLEAN UP AND RETURN
31 003073  023250          XBNFND: CALL     CONGRP        ;COMPUTE THE "OFFSET" SECTOR
32 003074  104203  001130          MOV      #SECTRK,R3   ;SECTORS/TRACK
33 003076  104204  001111          MOV      #CURPBN,R4   ;TRACK
34 003100  021537          CALL     DMUL          ;MULTIPLY TO GET STARTING PBN ON TRACK
35 003101  104203  000731          MOV      #DDUMMY,R3   ;SECTOR ON TRACK
36 003103  021503          CALL     DADD          ;ADD TO GET ACTUAL PBN
37 003104  104641  000001          MOV     1(R4),R1     ;GET HIGH ORDER
38 003106  105301  001325          ADD     ST.XBN,R1     ;ADD TO GET ABSOLUTE XBN
39 003110  103201  170000          BIC     #HD.CLR,R1    ;CLEAR HEADER
40 003112  101201  120000          BIS     #HD.XBN,R1    ;MARK AS XBN
41 003114  104010  001112          MOV     R1,CURPBN+1   ;STORE BACK
42 003116  003241          BR      PDONE         ;CLEAN UP AND RETURN
43          :
44          :
45 003117  023250          LRBN:  CALL     CONGRP        ;COMPUTE THE OFFSET SECTOR
46 003120  104205  001053          MOV      #SCR,R5      ;POINT TO CHARACTERISTICS
47 003122  104653  000011          MOV      LBNTRK(R5),R3 ;GET LBN/TRK
48 003124  103203  177400          BIC     #HI BYTE,R3   ;CLEAR HIGH BYTE
49 003126  104201  000731          MOV      #DDUMMY,R1   ;POINT TO REMAINDER
50 003130  104114          MOV     (R1),R4       ;GET IT
51 003131  106043          CMP     R4,R5         ;COMPARE
52 003132  073205          BMI     LBNFND        ;IF MINUS THEN LBN
53 003133  104653  000004          MOV     RBNTRK(R5),R3 ;GET RBN/TRACK
54 003135  103203  177600          BIC     #HI1 BYTE,R3  ;CLEAR OUT GARBAGE
55 003137  104030  000736          MOV     R3,TEMP       ;STORE IT
56 003141  114000  000737          CLR    TEMP+1        ;FOR STORE
57 003143  104203  000736          MOV     #TEMP,R3     ;FOR MULTIPLY

```

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 91-1
 PBN->D,X,L,RBN CONVERSION OVERLAY (G5)

58	003145	104204	001111	MOV	#CURPBN,R4	;DITTO - NUM OF TRACKS
59	003147	021537		CALL	DMUL	;MULTIPLY BY TRACK NUMBER
60	003150	104204	000731	MOV	#DDUMMY,R4	;FOR SUBTRACT
61	003152	104653	000011	MOV	LBNTRK(R5),R3	;GET LBN/TRK
62	003154	103203	177400	BIC	#HIBYTE,R3	;CLEAR HIGH BYTE
63	003156	104030	000736	MOV	R3,TEMP	;STORE IT
64	003160	114000	000737	CLR	TEMP+1	;FOR CLEAR
65	003162	104203	000736	MOV	#TEMP,R3	;POINT FOR SUBTRACT
66	003164	021521		CALL	DSUB	;SUBTRACT TO GET RESIDUE RBN
67	003165	104204	001111	MOV	#CURPBN,R4	;TO GET RBN NUMBER
68	003167	104203	000731	MOV	#DDUMMY,R3	;DITTO
69	003171	021503		CALL	DADD	;GIVES RELATIVE RBN
70	003172	104641	000001	MOV	1(R4),R1	;GET HIGH ORDER
71	003174	105301	001324	ADD	ST.RBN,R1	;ADD TO GET ABSOLUTE RBN
72	003176	103201	170000	BIC	#HD.CLR,R1	;CLEAR TH EHADER
73	003200	101201	060000	BIS	#HD.RBN,R1	;SET AS A RBN
74	003202	104010	001112	MOV	R1,CURPBN+1	;STORE BACK
75	003204	003241		BR	PDONE	;CLEAN UP AND RETURN
76						
77						
78	003205	104204	001111	LBNFND: MOV	#CURPBN,R4	;MULT NUM OF TRACKS
79	003207	104653	000011	MOV	LBNTRK(R5),R3	;GET LBN/TRK
80	003211	103203	177400	BIC	#HIBYTE,R3	;CLEAR HIGH BYTE
81	003213	104030	000736	MOV	R3,TEMP	;STORE IT
82	003215	114000	000737	CLR	TEMP+1	;FOR CLEAR
83	003217	104203	000736	MOV	#TEMP,R3	;POINT FOR MULT
84	003221	021537		CALL	DMUL	;GET LBN'S
85	003222	104203	000731	MOV	#DDUMMY,R3	;PLUS RESIDUE
86	003224	021503		CALL	DADD	;GIVES LBN NUMBER
87	003225	104207	001053	MOV	#SCR,R0	;POINT TO CHARACTERISTICS
88	003227	104641	000001	MOV	1(R4),R1	;GET HIGH ORDER
89	003231	105301	001323	ADD	ST.LBN,R1	;ADD TO GET ABSOLUTE LBN
90	003233	103201	170000	BIC	#HD.CLR,R1	;CLEAR HEADER
91	003235	101201	000000	BIS	#HD.LBN,R1	;SET AS LBN
92	003237	104010	001112	MOV	R1,CURPBN+1	;STORE BACK
93	003241	104300	001111	001114 PDONE: MOV	CURPBN,CURBN	;GET LOW ORDER
94	003244	104300	001112	001115 MOV	CURPBN+1,CURBN+1	;HIGH ORDER
95	003247	000000		RETURN		

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 92
 PBN->D,X,L,RBN CONVERSION OVERLAY (G5)

```

1
2
3
4 003250 104300 001130 000731 CONGRP: MOV SECTRK,DDUMMY ;GET SECTORS/TRACK
5 003253 114000 000732 CLR DDUMMY+1 ;CLEAR HIGH ORDER
6 003255 104203 000731 MOV #DDUMMY,R3 ;FOR DIVIDE
7 003257 104204 001111 MOV #CURPBN,R4 ;DITTO
8 003261 021565 CALL DDIV ;DIVIDE PBN/SECTRK TO GET TRACK #
9 003262 104141 MOV (R4),R1 ;GET CURRENT TRACK
10 003263 PUSH R1 ;SAVE IT ON STACK
11 003264 104642 000001 MOV 1(R4),R2 ;DITTO HIGH ORDER
12 003266 PUSH R2 ;SAVE AGAIN
13 003267 115001 TST R1 ;IS LOW ORDER TRACK 0 ?
14 003270 053273 BNE 1$ ;NOPE - TRACK # CAN'T BE 0
15 003271 115002 TST R2 ;IS HIGH ORDER TRACK 0 ?
16 003272 013357 BEQ 5$ ;YES - NO OFFSET OR NEED TO GO ON
17 003273 104131 1$: MOV (R3),R1 ;GET CURRENT SECTOR NUMBER
18 003274 PUSH R1 ;SAVE IT
19 003275 104207 001053 MOV #SCR,R0 ;POINT TO CHARACTERISTICS BLOCK
20 003277 104671 000003 MOV TRKGRP(R0),R1 ;LOAD TRACKS/GROUP
21 003301 103201 177400 BIC #HIBYTE,R1 ;CLEAR OUT HIGH GARBAGE
22 003303 100131 MOV R1,(R3) ;STORE TRACK/GROUP IN DDUMMY
23 003304 021565 CALL DDIV ;DIVIDE TO GET GROUP NUMBER
24 003305 104141 MOV (R4),R1 ;GET ABSOLUTE GROUP NUMBER
25 003306 053312 BNE 3$ ;CAN'T BE 0 - CONTINUES
26 003307 104641 000001 MOV 1(R4),R1 ;GET HIGH ORDER
27 003311 013365 BEQ 4$ ;GROUP IS 0 - NO OFFSET
28 003312 104207 001053 3$: MOV #SCR,R0 ;POINT TO CHARACTERISTICS
29 003314 104671 000002 MOV GRPCYL(R0),R1 ;GET GROUPS/CYLINDER
30 003316 103201 177400 BIC #HIBYTE,R1 ;CLEAR OUT GARBAGE
31 003320 100131 MOV R1,(R3) ;STORE GROUPS/CYLINDER
32 003321 021565 CALL DDIV ;DIVIDE TO GET RELATIVE GROUP
33 003322 104131 MOV (R3),R1 ;GET GROUP NUMBER
34 003323 013365 BEQ 4$ ;IF ZERO THEN DONE - NO OFFSET
35 003324 104207 001053 MOV #SCR,R0 ;POINT TO CHARACTERISTICS
36 003326 104672 000011 MOV OFFS(R0),R2 ;GET GROUP OFFSET
37 003330 110702 SWAB R2 ;GET INTO LOWBYTE
38 003331 103202 177400 BIC #HIBYTE,R2 ;CLEAR HIGH GARBAGE
39 003333 115002 TST R2 ;ANY OFFSET ?
40 003334 013365 BEQ 4$ ;NO - NO NEED TO FIX UP
41 003335 100142 MOV R2,(R4) ;STORE FOR MULT
42 003336 114002 CLR R2 ;CLEAR FOR STORE
43 003337 100642 000001 MOV R2,1(R4) ;CLEAR HIGH ORDER
44 003341 021537 CALL DMUL ;MULTIPLY GROUP*OFFSET
45 003342 POP R2 ;RESTORE ORIGINAL SECTOR
46 003343 104145 MOV (R4),R5 ;GET TOTAL OFFSET
47 003344 105025 ADD R2,R5 ;ADD TO GET SECTOR NUMBER
48 003345 100145 MOV R5,(R4) ;STORE IT
49 003346 114002 CLR R2 ;FOR STROE
50 003347 100642 000001 MOV R2,1(R4) ;CLEAR HIGH ORDER
51 003351 104301 001130 MOV SECTRK,R1 ;SECTOR'S/TRACK
52 003353 100131 MOV R1,(R3) ;FOR MOD FUNCTION
53 003354 100632 000001 MOV R2,1(R3) ;CLEAR HIGH ORDER
54 003356 021565 CALL DDIV ;REMAINDER IS NEW SECTOR NUMBER
55 003357 100641 000001 5$: POP R1 ;RESTORE TRACK NUMBER LOW
56 003360 MOV R1,1(R4) ;STORE IT
57 003362 POP R1 ;RESTORE TRACK NUMBER HIGH

```


UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 93
 ERROR MESSAGE OVERLAY (G6)

1					.SBTTL	ERROR MESSAGE OVERLAY (G6)
2					:	
3					:	ERROR MESSAGE OVERLAY
4					:	
5	003373				DMOVLY	G6,START
6					:	
7	003023	104207	001264		MOV	#DMBUF,R0 ;POINT TO BUFFER
8	003025	104671	000001		MOV	1(R0),R1 ;GET SUB CODE
9	003027	104172			MOV	(R0),R2 ;GET ERROR NUMBER
10	003030	117402			DEC	R2 ;MAKE RELATIVE TO 0
11	003031	110202			ROL	R2 ;MULTIPLY BY 2
12	003032	105202	003441		ADD	#ERRTBL,R2 ;OFFSET TO MESSAGE ADDRESS
13	003034	104127			MOV	(R2),R0 ;MESSAGE TO SEND
14	003035	104621	000001		MOV	1(R2),R1 ;LENGTH OF MESSAGE
15	003037	060016			XFC	MAINTR ;SEND IT
16	003040	000000			RETURN	;GO BACK AND DIE
17						
18					.ENABL	LC
19	003041	050001			M.ER1:	.WORD 50001
20	003042	107	105	124	.asciz	"GET STATUS failure"
21					;	*****
22	003054	050002			M.ER2:	.WORD 50002
23	003055	123	104	111	.asciz	"SDI send error"
24					;	*****
25	003065	050003			M.ER3:	.WORD 50003
26	003066	125	156	163	.asciz	"Unsuccessful SDI command"
27					;	*****
28	003103	050004			M.ER4:	.WORD 50004
29	003104	123	104	111	.asciz	"SDI receive error"
30					;	*****
31	003115	050005			M.ER5:	.WORD 50005
32	003116	125	116	111	.asciz	"UNIBUS I/O error"
33					;	*****
34	003127	050006			M.ER6:	.WORD 50006
35	003130	106	157	162	.asciz	"Formatter initialization error"
36					;	*****
37	003150	050007			M.ER7:	.WORD 50007
38	003151	116	157	156	.asciz	"Non-existent unit number"
39					;	*****
40	003166	050010			M.ER8:	.WORD 50010
41	003167	104	102	116	.asciz	"DBN/XBN format error (FORMAT XFC failed)"
42					;	*****
43	003214	050011			M.ER9:	.WORD 50011
44	003215	106	103	124	.asciz	"FCT check error"
45					;	*****
46	003225	050012			M.ER10:	.WORD 50012
47	003226	123	105	105	.asciz	"SEEK error"
48					;	*****
49	003234	050013			M.ER11:	.WORD 50013
50	003235	122	103	124	.asciz	"RCT check error"
51					;	*****
52	003245	050014			M.ER12:	.WORD 50014
53	003246	114	102	116	.asciz	"LBN format error (FORMAT XFC failed)"
54					;	*****
55	003271	050015			M.ER13:	.WORD 50015
56	003272	106	103	124	.asciz	"FCT write error"
57					;	*****

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 93-1
 ERROR MESSAGE OVERLAY (G6)

58	003302	050016			M.ER14: .WORD 50016
59	003303	122	103	124	.asciz 'RCT read error'
60					*****
61	003313	050017			M.ER15: .WORD 50017
62	003314	122	103	124	.asciz 'RCT write error'
63					*****
64	003324	050020			M.ER16: .WORD 50020
65	003325	122	103	124	.asciz 'RCT full'
66					*****
67	003332	050021			M.ER17: .WORD 50021
68	003333	106	103	124	.asciz 'FCT read error'
69					*****
70	003343	050022			M.ER18: .WORD 50022
71	003344	106	103	124	.asciz 'FCT non-existent'
72					*****
73	003355	050023			M.ER19: .WORD 50023
74	003356	106	103	124	.asciz 'FCT Down Line Load error (FCT block not avbl.)'
75					*****
76	003406	050024			M.ER20: .WORD 50024
77	003407	104	162	151	.asciz 'Drive init timeout'
78					*****
79	003421	050025			M.ER21: .WORD 50025
80	003422	111	156	166	.asciz 'Invalid response to question'
81					*****
82	003441				M.ER22:
83					.DSABL LC

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 94
 ERROR MESSAGE OVERLAY (G6)

1				
2				
3				
4	003441	003041	ERRTBL: .WORD	M.ER1
5	003442	000013	.WORD	M.ER2-M.ER1
6	003443	003054	.WORD	M.ER2
7	003444	000011	.WORD	M.ER3-M.ER2
8	003445	003065	.WORD	M.ER3
9	003446	000016	.WORD	M.ER4-M.ER3
10	003447	003103	.WORD	M.ER4
11	003450	000012	.WORD	M.ER5-M.ER4
12	003451	003115	.WORD	M.ER5
13	003452	000012	.WORD	M.ER6-M.ER5
14	003453	003127	.WORD	M.ER6
15	003454	000021	.WORD	M.ER7-M.ER6
16	003455	003150	.WORD	M.ER7
17	003456	000016	.WORD	M.ER8-M.ER7
18	003457	003166	.WORD	M.ER8
19	003460	000026	.WORD	M.ER9-M.ER8
20	003461	003214	.WORD	M.ER9
21	003462	000011	.WORD	M.ER10-M.ER9
22	003463	003225	.WORD	M.ER10
23	003464	000007	.WORD	M.ER11-M.ER10
24	003465	003234	.WORD	M.ER11
25	003466	000011	.WORD	M.ER12-M.ER11
26	003467	003245	.WORD	M.ER12
27	003470	000024	.WORD	M.ER13-M.ER12
28	003471	003271	.WORD	M.ER13
29	003472	000011	.WORD	M.ER14-M.ER13
30	003473	003302	.WORD	M.ER14
31	003474	000011	.WORD	M.ER15-M.ER14
32	003475	003313	.WORD	M.ER15
33	003476	000011	.WORD	M.ER16-M.ER15
34	003477	003324	.WORD	M.ER16
35	003500	000006	.WORD	M.ER17-M.ER16
36	003501	003332	.WORD	M.ER17
37	003502	000011	.WORD	M.ER18-M.ER17
38	003503	003343	.WORD	M.ER18
39	003504	000012	.WORD	M.ER19-M.ER18
40	003505	003355	.WORD	M.ER19
41	003506	000031	.WORD	M.ER20-M.ER19
42	003507	003406	.WORD	M.ER20
43	003510	000013	.WORD	M.ER21-M.ER20
44	003511	003421	.WORD	M.ER21
45	003512	000020	.WORD	M.ER22-M.ER21
46	003513		DMEND	
47		000001	.END	

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 94-1
 SYMBOL TABLE

ACC	001025	CLELP	003061	CITXBN	001116	DMBUFL=	000016	FCTAVL=	000001
ACCESS	002226	CLELP2	003051	CLLOF	001236	DMUL	001537	FCTBAD=	000004
AGAIN	003476	CLESKP	003117	CVT	= 000020	DOLBN	003033	FCTCK	003032
ALLOVR	002562	CLEWRT	003743	CVTERR	002721	DONDLL	003424	FCTCKD	003267
ALLOV1	002570	CLHERE	003756	CVTSK	002665	DONE	= 000021	FCTCKE	003316
AOUT	002353	CLRBUF	003644	CYL	= 000006	DONFCT	003643	FCTCLP	003112
ATTN	= 000002	CLRLP	003652	CYLBN	= 000000	DONONE	003547	FCTCL1	003115
ATTN1	002344	CLSKP2	003140	CYLNUM	001126	DPBN	003277	FCTCNT	001257
BADEDC	001447	CLSKP3	003154	C512	= 000016	DPREA	001322	FCTCPY	001245
BADPBN	001224	CLSKP4	003153	DADD	001503	DSUB	001521	FCTEMT=	000002
BADRBN	003616	CMDBUF=	006621	DADD1	001512	DUPOVL=	007774	FCTFLG=	000025
BD	= 020000	CMPDAT=	000006	DASH	= 000055	DWRD	001445	FCTFMT	001241
BDHD	= 000040	CNLP	004033	DATA	= 000005	DXBN	003035	FCTNOT	003220
BDIRCT	004101	CNLP1	004031	DATAGN	003625	DXCH	003432	FCTNPD	001261
BDLST	= 006621	CNT	001250	DATBUF	003243	DXCHEC	003431	FCTPTR	001256
BDTST	= 010000	CNTCYL	001455	DATCON	003464	DXERR	003340	FCTRCT	003166
BIT0	= 000001	CONBAD	004261	DATE	001302	DXFCPG	003341	FCTREV	001312
BIT1	= 000002	CONBLK	001424	DATERR	003702	DXFCP1	003360	FCTRLP	003040
BIT10	= 002000	CONDON	004113	DATLP	003476	DXFORM	003070	FCTSKP	003437
BIT11	= 004000	CONDO1	004115	DATLP1	003567	DXTRK	003026	FCTSK1	003431
BIT12	= 010000	CONERR	004116	DATLP2	003612	EAGAIN	002607	FCTSLP	003615
BIT13	= 020000	CONER1	004121	DATRET	003623	ECC	= 000015	FCTSP	003733
BIT14	= 040000	CONEXT	004272	DATRT1	003624	ECCCK	003000	FCTSZ	= 000010
BIT15	= 100000	CONGRP	003250	DATVER	003636	ECCF	= 010000	FCTUSD	003204
BIT2	= 000004	CONINT	003036	DATVL1	003672	ECHO	= 000010	FCTWRT	003023
BIT3	= 000010	CONLOW	004107	DATVL2	003701	EDC	001446	FCWERR	003223
BIT4	= 000020	CONON	003427	DAYS	= 036031	EIMAGE	001232	FDAT	= 000012
BIT5	= 000040	CONTEX	004275	DBBAD	001315	EMAX	001226	FDLL	003246
BIT6	= 000100	COUNT	001476	DBBUF=	003430	ENTRY	000714	FERR	003330
BIT7	= 000200	CR	001040	DBLEN	= 000034	EORCT	003147	FIDANS	004220
BIT8	= 000400	CR.ACC	000766	DBN	= 000010	ERCV	= 000002	FIDMUL	004233
BIT9	= 001000	CR.CLR	000772	DBNCYL=	000022	ERDN	= 000010	FILLIT	003637
BLANWD	= 020040	CR.DIS	000756	DBNFND	003042	ERECOV	001103	FINCHK	003023
BLKFND	003157	CR.ERV	001006	DCLR	001027	ERFLAG	001222	FINDON	004216
BMAX	= 007775	CR.GCR	000746	DCMP	001623	ERLEN	= 000002	FINI	= 040000
BOTTOM	003276	CR.GSR	000752	DCMP1	001646	ERPNT	001253	FINLEN	004201
BRBN	003774	CR.GST	000742	DCMP2	001641	ERR	001227	FINLN1	004203
BREAK	= 000000	CR.ONL	001012	DCMP3	001654	ERRBUF	001225	FINMSG	003172
BSTGS	= 002000	CR.RCL	001002	DCMP4	001634	ERRCNT	001450	FIXBLK	003445
BUFMSK=	007777	CR.RUN	000762	DDIV	001565	ERRHND	002571	FIXFCT	003766
BUFPNT	001254	CR.SEK	000776	DDUMMY	000731	ERRMNT	002542	FIXIT	004001
BUF1	= 004535	CS	004130	DEAD	= 000020	ERROR	003440	FIXLP	003462
BUF2	= 005152	CSKIP	003454	DECALP	003664	ERRSYM=	000002	FKIP1	004051
BUF3	= 005567	CSKIP1	003466	DECASC	003660	ERRT	002073	FKIP10	004075
BUF4	= 006204	CSKIP2	003465	DINIT	= 000011	ERRTBL	003441	FKIP2	004035
BUF5	= 006621	CSKIP3	003726	DIS	001022	ER1	003706	FKIP9	004420
BUF6	= 007321	CSKIP6	003672	DISCON	002315	EXTFCT	004236	FKP1	004066
CBUF	001441	CSKIP7	003733	DLERR	003062	EXTRET	004257	FLAG	001220
CDONE	003777	CS1	004314	DLERT	003443	EXTRT1	004260	FLAG1	001221
CEDC	002600	CURBN	001114	DLL	= 000400	FBDHD	= 010000	FLIP	003112
CHAR	001765	CURGRP	001460	DLLDN	003776	FBEGIN	003252	FLIPON=	000400
CHRDNE=	010000	CURLBN	001116	DLLDN1	003777	FBEG2	003265	FLKIP1	004374
CKR	004166	CUROVL	001154	DLLFLE	003762	FCFXLP	003202	FLKIP2	004360
CKR1	004352	CURPBN	001111	DLLNO	004000	FCL	= 177760	FLKP1	004411
CLBUF	= 007321	CURPNT	001471	DLLRET	003421	FCLMSG	001317	FMSTL	= 000010
CLEAR	002234	CURRBN	001107	DLLRT1	003405	FCNT	001240	FMTSTA	003224
CLEDON	003147	CURTRK	001113	DMBUF	001264	FPCG	003573	FODONE	003206

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 94-2

SYMBOL TABLE

FOLOOP	003034	HD.PRIV=	050000	LFCTNT=	000012	LSKIP2	003432	M.ER6	003127
FORERR	003366	HD.RBN=	060000	LFCTUS=	000014	LSKIP3	004166	M.ER7	003150
FORMAT=	000001	HD.REV=	030000	LFERR	003376	LSKIP4	003170	M.ER8	003166
FPRIM	020000	HD.XBN=	120000	LFINMS=	000012	LSKIP6	003641	M.ER9	003214
FRCPY	000001	HEAD	000005	LFIXIT	004324	LSKIP7	004173	M512	= 126736
FRDONE	003721	HERE	003602	LFORM	003023	LSKIP8	003565	M576	= 074161
FRSKP	003715	HGHPBN	001155	LHERE	003547	LSKIP9	004153	N	001452
FSER	000002	HIBYTE=	177400	LKIP10	003520	LSND	004130	NDLL	= 004000
FT.BUF	000000	HI1BYT=	177600	LKIP12	003330	LTO	001251	NEXT	002403
FT.HI	000002	HI2BYT=	177700	LKIP2	003051	LTRK	003023	NEXT1	001246
FT.LOW=	000001	HKIP	003031	LKIP20	003137	MAINTR=	000016	NEXT5	002410
FWGOOD	003157	HKIP1	003055	LKIP21	003146	MAINTW=	000017	NGD	003224
FWRD	001442	HOLD	001230	LKIP22	003263	MANU	= 000200	NGD1	003234
FWTDON	003215	HOLDBN	001120	LKIP23	003237	MARBAD	003465	NN1	001454
F1	= 000000	HOLDPN	001124	LKIP24	003247	MASK	= 000000	NO	003322
F2	= 000003	HOLRBN	001122	LKIP25	003343	MAXTRY=	000010	NOCERR	004063
F3	= 000006	HPREA	001321	LKIP27	003273	MENTLN=	000002	NOCRY	002621
F4	= 000011	HSLIM	000727	LKIP28	003523	MLEN	003246	MODLL	003070
F5	= 000014	HSTHI	= 000002	LKIP29	003507	MNCNT	001263	NOERR	003565
F6	= 000017	HSTLO	= 000001	LKIP30	003556	MORE	004015	NOFCT	= 100000
F7	= 000022	H1	= 000055	LKIP31	003561	MSGLOP	003352	NOGOOD	003265
F8	= 000025	H2	= 000066	LKIP33	003541	MSGLP2	003414	NOINC	003120
F9	= 000030	IMAGE	= 007321	LKIP4	003500	MSGLP3	003421	NOLBN	003461
GCR	001017	IMLEN	= 000003	LKIP5	003444	MSGLP4	003412	NOSEK	003332
GDBLK	= 005567	IMSTAR	001320	LKIP7	003540	MSGOFF=	000001	NOTHER	004307
GDECC	003022	INDSEC=	000013	LKIP8	003131	MSGTBL	003024	NOTR	003202
GDONE	004274	INI	001247	LKIP9	003115	MSG1	002030	NOTY	004234
GENCON	004015	INIRCT=	020000	LIEN	= 000015	MSG1LN=	000016	NUM	001440
GETUNT	004134	INITDD	002377	LMORE	004340	MSG2LN=	000022	NUMDBN	003355
GOBAD	= 000020	INITIT	002331	LNOERR	003532	MSG4LN=	000022	NUMLBN	003060
GOVER	004147	INITL	004330	LO	= 007777	MSG5LN=	000026	NUMRBN	003120
GOVER1	004143	INITPT	002362	LOAD	002152	MSG6LN=	000035	NUMXBN	003405
GRP	= 000010	INITP1	002367	LOADER	002222	MSG7LN=	000016	N1	001453
GRPCNT	001461	INITS	002340	LOAD2	002177	MULDN	004136	OCDONE	003207
GRPCYL=	000002	INPEL	= 000007	LOAD3	002203	MULERR	004137	OCLoop	003062
GSKIP	004252	INPERR	003234	LOAD4	002217	MULER1	004176	OERR	002417
XIP1	004167	INST	= 000001	LOAD5	002157	MULPC	000733	OERR2	002426
GSKIP2	004205	IRECAL	001102	LOBL	= 000040	MULT10	004142	OFATAL	003331
GSKIP3	004224	ISEEK	001076	LOBYTE=	000377	MULT2	004126	OFFS	= 000011
GSR	001020	ISKIP	003603	LOK	004163	M.ER1	003041	OFFSET	000736
GST	001016	LAGAIN	003443	LONGTO=	000001	M.ER10	003225	OK	003723
GSTATS	001657	LAST	= 100000	LOOP	003120	M.ER11	003234	OLDONE	003326
GTFLAG=	002000	LBNBAD	001313	LOOPP	003201	M.ER12	003245	ONE	001463
G1	= 000060	LBNCYL	001144	LOOPP2	003153	M.ER13	003271	ONLIN	002323
G2	= 000033	LBNFND	003205	LOOP1	002012	M.ER14	003302	ONLINE	001105
G3	= 000036	LBNHOS=	000012	LOOP2	002036	M.ER15	003313	OQUIT	003352
G4	= 000041	LBNLBN	001134	LOOP3	003431	M.ER16	003324	ORFAL	003331
G5	= 000044	LBNPCY	001146	LOVER	004223	M.ER17	003332	OVCNT	= 000023
G6	= 000063	LBNTRK=	000011	LOVER1	004320	M.ER18	003343	OVER	003500
G7	= 000047	LBUFE	= 003300	LOVER2	004264	M.ER19	003355	OVER1	003774
G8	= 000052	LCHEC	003377	LOVER3	004255	M.ER2	003054	OVER2	003743
HASH	003024	LDONE	004323	LOVER4	004243	M.ER20	003406	OVE.F1=	003023
HD.BAL=	110000	LEN	= 000000	LPBN	003160	M.ER21	003421	OVE.F2=	003023
HD.CLR=	170000	LERR	003515	LRBN	003117	M.ER22	003441	OVE.F3=	003023
HD.CUR	001457	LER1	003656	LREDO	004407	M.ER3	003065	OVE.F4=	003023
HD.DBN=	140000	LER2	003665	LSKIP	003421	M.ER4	003103	OVE.F5=	003023
HD.LBN=	000000	LER3	003653	LSKIP1	003433	M.ER5	003115	OVE.F6=	003023

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 94-3
SYMBOL TABLE

OVS.F7=	003023	OVS.H2=	037742	RCLP6	003377	RETAD	003374	SEEK2	002301
OVS.F8=	003023	OVS.MN=	001040	RCTBAD	001314	RETRY	001477	SEEK3	002312
OVS.F9=	003023	OV... =	022567	RCTBUF=	006621	REVBUF=	006204	SEEK4	002273
OVS.G1=	003023	PAERR	002501	RCTCK	003323	REVCNT	001255	SEEK5	002300
OVS.G2=	003023	PAGAIN	003073	RCTCKD	003554	REVECT=	000100	SEEK6	002272
OVS.G3=	003023	PAGE	002435	RCTCKE	003577	REVLEN=	000004	SEEK7	002267
OVS.G4=	003023	PAGER	002463	RCTCLP	003357	REVRBN	001152	SEND =	000004
OVS.G5=	003023	PALP1	002445	RCTCL1	003362	REVSEC=	000007	SERBD	003736
OVS.G6=	003023	PALP2	002452	RCTCNT	001474	RFTL	004705	SERCON	003723
OVS.G7=	003023	PALP3	002500	RCTERR	003565	RLD	004676	SERNUM	001306
OVS.G8=	003023	PALP4	002473	RCTFMT	001243	RLDONE	003322	SEROK	003745
OVS.H1=	003023	PARITY=	000200	RCTINI	003023	RLOOP	003131	SERRT	003761
OVS.H2=	003023	PARIT1=	000400	RCTLBN	001262	RLOOP1	003164	SETOVL	003437
OVS.MN=	000714	PARMTB	003163	RCTNGD	003523	RNWHER	004236	SFTRPT	004271
OVLBLK	001420	PBNBUF=	005152	RCTRLP	004034	ROVER	003055	SHORTO=	000000
OVLBN =	000003	PBUFE =	003317	RCTSZ =	000014	ROVER1	003072	SIP =	000012
OVLTL	001327	PCNT	001475	RCTTOT	001236	RPRIM =	000004	SKERR	003334
OVL.F1=	001057	PCON	003023	RCTUPD	003023	RPT	003423	SKIP1	003154
OVL.F2=	001402	PDONE	003241	RCTWLP	003402	RPT1	003745	SKIP19	003135
OVL.F3=	000422	PERR	003162	RCTWRT	004027	RQUIT	003243	SKIP3	003230
OVL.F4=	000652	PHYSA =	001000	RCTWT	003375	RRC	004373	SKIP4	003140
OVL.F5=	001202	PLEN =	000021	RCV =	000005	RRERR	003550	SKIP5	003071
OVL.F6=	000334	PNGBLK	003070	RCVMNT	002532	RRPL	003540	SKIP6	003264
OVL.F7=	000450	PNGPG	003571	RCVRDY=	000001	RSER =	000000	SKIP7	003233
OVL.F8=	000573	PNGPNG	003066	RCWERR	004220	RTDON	003563	SKPCNT	001234
OVL.F9=	000205	PNGRD	003125	RCXLP	003550	RTRY =	000001	SLAS =	000057
OVL.G1=	001317	PRET	003161	RC.FRE=	000000	RTY =	100000	SLEEK	003164
OVL.G2=	000045	PRIM =	001000	RC.NUL=	100000	RTYCNT	001470	SLEEK2	003201
OVL.G3=	000225	PRIMRB	002730	RC.PRIV=	020000	RTYDN =	000002	SLEN =	000026
OVL.G4=	001667	PRMBUF=	006204	RC.SND=	030000	RUN	001024	SLOOP	003456
OVL.G5=	000351	PRMY =	100000	RC.UNU=	040000	RWCMD =	013400	SND	004116
OVL.G6=	000471	PROD =	000000	RDBLK	000721	RWGD	003525	SNDCNT	001467
OVL.G7=	000364	QUESDN=	000200	RDBUF =	004535	RWGOOD	004161	SNDLP	003607
OVL.G8=	000224	RBNBUF=	006621	RDCMD =	100000	RWRDY =	100000	SNDMNT	002522
OVL.H1=	000313	RBNLBN	001136	RDLEN =	000004	RWTDON	004217	SNDRES	003546
OVL.H2=	001337	RBNPCY	001150	RDONE	003214	RW.BUF=	000001	SRCK	001740
OVL.MN=	003540	RBNRPT=	000200	RDONE1	003220	RW.CMD=	000004	ST	001031
OVL... =	022147	RBNTRK=	000004	RDONE2	003233	RW.DAT=	000000	STACK	001216
OVRLAY	002510	RBNWRT	004175	READ =	000002	RW.DUM=	000005	STARIT	001233
OVS.F1=	013176	RB.CMD=	000002	READ1	003525	RW.EDC=	000400	START	003023
OVS.F2=	016304	RB.HI =	000001	READ10	003445	RW.ER1=	000000	START2	003166
OVS.F3=	023346	RB.IM =	000003	READ11	003115	RW.ER2=	000401	START3	003344
OVS.F4=	024412	RB.LOW=	000000	READ2	003651	RW.HI =	000003	STASEC	001120
OVS.F5=	026764	RCBUFE=	003372	READ3	003472	RW.LOW=	000002	STATFR	002661
OVS.F6=	032510	RCD	004563	READ4	003620	RW.STA=	000000	STATRE	002645
OVS.F7=	031370	RCFIX	003421	READ7	003112	SBUFE =	003342	STATRT	002660
OVS.F8=	021310	RCFXLP	004204	READ8	004471	SCR	001053	STATST	001661
OVS.F9=	042640	RCINDN=	000100	READ9	003164	SECCNT	001451	STATUS=	000007
OVS.G1=	010340	RCINER	003441	REBUFE=	003524	SECNDY	003256	STATVL	002642
OVS.G2=	033400	RCINIT=	000040	RECAL	002146	SECSI6=	000400	STCKSV	001217
OVS.G3=	033512	RCINLP	003157	RECIR =	040000	SECSI8=	000440	STCLR =	170377
OVS.G4=	034164	RCL	004436	RECOV	001500	SECTCY	001132	STCYL =	000001
OVS.G5=	043252	RCLN =	000034	RECTMP	001502	SECTRK	001130	STDBN =	000003
OVS.G6=	044174	RCLP	003033	REDO	004064	SEEK	002242	STDIAG	001752
OVS.G7=	015334	RCLP2	003162	RELEN =	000025	SEEKER	003372	STFLAG=	004000
OVS.G8=	022676	RCLP3	003431	REPEAT=	00100C	SEEK0	002244	STFORM	001745
OVS.H1=	026136	RCLP4	003372	RESTAB	003272	SEEK1	002253	STLBN =	000002

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 94-4
SYMBOL TABLE

STO	001252	TALKP	002070	TKIP9	003140	UREAD =	000013	XAGAIN	003577
STPNIC	002005	TALKRT	002062	TMPTRY	001501	USDFCT	003636	XBBAD	001316
STRBM =	000003	TATTN1	002141	TOTRCT	001472	UWRITE =	000014	XBUFE =	003466
STSC =	000012	TBLK	001235	TRK =	000011	VAXTME	004277	XBLEM =	000034
STSKP	002002	TBUFF	003247	TRKCNT	001462	VERHD	003604	XBNCYL =	000021
STSK1	001677	TBUFFL =	000051	TRKCYL	001142	VLD =	000004	XBNFND	003073
STWLK	001757	TCLEAR	002075	TRKGRP =	000003	VLD1 =	000010	XBNIT	003145
STXBM =	000002	TEMP	000736	TWO	003575	V1 =	000000	XBNSEC	001140
ST.DB =	001000	TEMP2	000733	TWOB =	000006	V2 =	000002	XDONE	003353
ST.DBM	001326	TERR	002135	TWOC	001465	V3 =	000004	XEORCT	003654
ST.DF =	000020	THREB =	000011	TWRD	001444	V4 =	000005	XFLIP	003616
ST.DR =	003040	TILOP	004027	UDAFM =	001000 G	WLOOP	003770	XGNBLK	003574
ST.FRB =	000002	TILOP1	004034	UERR	002432	WP =	000001	XNOINC	003624
ST.ERR =	000374	TIMER	002627	UHASH	003333	WRBLK	000721	XPBN	003331
ST.FD =	002000	TIMLOP	003767	UHKIP	003340	WRCMD =	122400	XPERR	003670
ST.IN =	000004	TIMLO1	003763	UHKIP1	003364	WRFLG	001223	XPNGRD	003631
ST.LBN	001323	TIMLP	002632	UID =	000000	WRITE =	000003	XPRET	003666
ST.PS =	000002	TIMTBL	003330	UNIT	000740	WRITE1	003260	XSKIP1	003035
ST.RBN	001324	TIMVAL =	100000	UNITBD	003714	WRITE2	003465	XSKIP2	003137
ST.RU =	000001	TKIP1	003045	UNITCN	003705	WRITE3	004117	XSKIP3	003134
ST.SR =	000020	TKIP11	003242	UNITRT	003713	WRITE4	003225	XSKIP4	003211
ST.WE =	000010	TKIP12	003165	UNNO	000741	WRITES	003115	XSKIP5	003237
ST.WP =	170000	TKIP13	003272	UNSEC =	000175	WRITE8	003260	XSKIP6	003234
ST.XBN	001325	TKIP14	003267	UN.ERI	000717	WRITE9	004057	XSLEEK	003071
SWAP	003231	TKIP2	003046	UN.ERR	000715	WRIT10	004630	XSLEK2	003105
SWRD	001443	TKIP3	003252	UN.ERT	000716	WRIT12	003254	XYZ	003670
TALIP1	002111	TKIP4	003233	UN.SEK	000720	WRIT13	004305	XYZ1	003671
TALK	002010	TKIP5	003154	UPDATE =	000022	WRN	003232	Y =	000131
TALKDN	002052	TKIP7	003212	UPDPNT	001471	WRNLN =	000040	YES	003243
TALKIP	002107	TKIP8	003130						

. ABS. 045356 000
000000 001

ERRORS DETECTED: 0

VIRTUAL MEMORY USED: 4256 WORDS (17 PAGES)
DYNAMIC MEMORY AVAILABLE FOR 71 PAGES
.B:UDAFML/C=\$DMACRO,B:UDAFML

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 3-8
 CRCSS REFERENCE TABLE (CREF V04.00)

HASH	69-4#	71-54	71-119											
HD.BAD	3-144#	47-157	49-78	54-189	56-169	63-48	88-57	88-149						
HD.CLR	3-146#	38-67	38-87	44-36	44-64	47-156	49-26	49-69	49-77	52-18	54-118	54-163	54-180	54-188
	54-206	56-21	56-57	56-106	56-131	56-161	56-168	59-7	60-7	63-30	63-52	63-68	63-82	63-92
	63-104	71-79	71-108	71-115	71-136	71-142	71-156	71-174	71-182	77-68	78-31	82-75	88-56	88-148
	89-9	91-6	91-27	91-39	91-72	91-90								
HD.CUR	6-32#	46-49*	46-52*	47-117	49-27									
HD.DBN	3-147#	46-49	48-16	91-28										
HD.LBN	3-141#	54-69	55-14	56-22	65-31	71-47	72-32	73-71	87-40	91-91				
HD.PRV	3-145#	54-124	54-158	54-177	56-132	63-72								
HD.RBN	3-142#	54-71	54-115	54-134	54-244	55-16	56-58	56-116	56-142	71-112	82-9	83-34	91-73	
HD.REV	3-143#	54-164	54-181	54-196	54-207	56-162	78-32							
HD.XBN	3-148#	46-52	48-14	62-90	75-27	75-96	86-45	88-38	88-130	90-35	91-40			
HEAD	3-94#	41-93												
HERE	47-112#	47-163	47-168											
HGHPBN	5-125#	40-29*	40-30*	77-59	77-70	91-7								
HI1BYT	3-125#	20-20	38-51	38-98	52-95	56-41	60-15	61-18	83-7	91-54				
HI2BYT	3-124#	41-37												
HI BYTE	3-122#	12-18	20-14	25-9	30-16	33-13	34-14	38-23	38-27	38-39	38-100	38-139	39-9	41-29
	41-92	41-95	43-6	46-34	46-61	46-72	47-19	49-36	52-27	52-39	52-86	56-9	56-71	59-15
	61-12	65-9	67-18	72-11	73-47	82-18	83-11	84-15	87-19	88-12	88-16	88-92	91-48	91-62
	91-80	92-21	92-30	92-38										
HKIP	69-8#	69-23												
HKIP1	69-5	69-20#												
HOLD	5-150#	63-18	63-19*	63-21	63-37	66-37	70-35	71-27*	71-28*	71-30	71-59	71-124	73-34*	73-35*
	73-37	73-39	73-40	73-41	73-42	84-20	87-5*	87-6*	87-8	87-10	87-11	87-12	87-13	
HOLDBN	5-103#	46-55*	46-57*	46-97	46-100	46-101	52-83	52-91	52-92	58-13*	58-15*	71-50		
HOLDPN	5-107#	56-95	56-96	56-206	61-28*	61-29*								
HOLRBN	5-105#	52-93	52-100	52-101	58-19*	58-20*								
HPREA	5-191#	41-96*	46-79	47-172	52-46	54-253	62-94	65-35	67-98	72-36	73-75	75-100	82-44	83-38
	84-99	90-39												
HSLIM	5-29#	47-80	47-122	54-32	54-76	62-92	65-33	67-23	67-96	72-34	73-73	75-21	75-98	82-41
	83-36	84-24	84-97	86-47	87-42	88-43	88-135	90-37						
HSTHI	3-191#	16-8												
HSTLO	3-190#	16-7	24-12											
IMAGE	3-348#	32-12*	32-13*	32-14*	32-15*	32-18	32-21	32-24	32-27	38-220	41-10	46-83	47-111	47-176
	48-8	48-35	48-37	49-13	49-37	49-88	52-50	54-63	54-155	54-240	54-257	55-8	55-35	55-37
	56-11	56-67	56-176	71-236	71-239	71-244								
IMLEN	3-306#	41-4	47-166	48-40	49-53	49-85	49-95	54-160	54-219	54-246	55-40	56-24	56-60	56-87
	56-173	56-195												
IMSTAR	5-190#	46-80	47-173	49-60*	49-93	49-102*	52-47	54-254	56-94*	56-193	56-202*			
INDSEC	3-263#													
INI	5-166#	18-7												
INIRCT	3-228#	52-71	52-74	56-30										
INIT5	13-190#	13-201												
INITDD	13-213	13-215	13-219#											
INITIT	13-186#	45-6												
INITL	38-6	45-5#												
INITP1	13-212#	13-217												
INITPT	13-158	13-209#	46-90	47-183	52-57	54-264								
INPEL	22-61#	23-24												
INPERR	22-59#	22-61	23-23											
INST	3-277#	62-36*	77-35	77-37*										
IRECAL	5-56	5-84#												
ISEEK	5-55	5-80#	19-13*	19-14*	19-15*	46-64*	46-65*	46-66*	52-31*	52-32*	52-33*	54-266*	63-113*	63-116*
	63-121*	71-218*	71-221*	71-224*	73-139*	73-142*	73-145*	82-80*	82-81*	82-82*	88-24*	88-25*	88-26*	88-116*

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE 5-10
 CROSS REFERENCE TABLE (CREF V04.00)

LOAD2	13-94	13-99#		
LOAD3	13-101#	13-108		
LOAD4	13-106	13-110#		
LOAD5	13-81	13-88#	13-98	
LOADER	13-97	13-109	13-113#	
LOBL	3-315#	81-31		
LOBYTE	3-123#	21-8	33-17	
LOK	54-106	54-214#		
LONGTO	3-73#	41-47		
LOOP	62-23	62-40#	62-140	
LOOP1	13-15#	13-75		
LOOP2	13-26	13-28#		
LOOP3	62-152#	62-154		
LOOPP	62-58	62-70#	62-124	
LOOPP2	62-39	62-55#	62-156	
LOVER	54-231#	54-235		
LOVER1	54-260	54-269#		
LOVER2	54-239	54-252#	54-268	
LOVER3	54-245	54-248#		
LOVER4	54-242#	54-247		
LPBN	58-21	61-5#		
LRBN	91-10	91-45#		
LREDO	55-32	55-37#		
LSKIP	54-9	54-14#		
LSKIP1	54-18#	54-60		
LSKIP2	54-13	54-17#		
LSKIP3	54-193	54-213	54-216#	
LSKIP4	52-53	52-61#		
LSKIP6	54-96	54-100#		
LSKIP7	54-66	54-72	54-218#	
LSKIP8	54-70	54-73#		
LSKIP9	54-205	54-209#		
LSND	54-151	54-192	54-199#	
LTO	5-168#	13-100	41-63*	41-64*
LTRK	56-6#			
M.ER1	93-19#	94-4	94-5	
M.ER10	93-46#	94-21	94-22	94-23
M.ER11	93-49#	94-23	94-24	94-25
M.ER12	93-52#	94-25	94-26	94-27
M.ER13	93-55#	94-27	94-28	94-29
M.ER14	93-58#	94-29	94-30	94-31
M.ER15	93-61#	94-31	94-32	94-33
M.ER16	93-64#	94-33	94-34	94-35
M.ER17	93-67#	94-35	94-36	94-37
M.ER18	93-70#	94-37	94-38	94-39
M.ER19	93-73#	94-39	94-40	94-41
M.ER2	93-22#	94-5	94-6	94-7
M.ER20	93-76#	94-41	94-42	94-43
M.ER21	93-79#	94-43	94-44	94-45
M.ER22	93-82#	94-45		
M.ER3	93-25#	94-7	94-8	94-9
M.ER4	93-28#	94-9	94-10	94-11
M.ER5	93-31#	94-11	94-12	94-13
M.ER6	93-34#	94-13	94-14	94-15
M.ER7	93-37#	94-15	94-16	94-17
M.ER8	93-40#	94-17	94-18	94-19

UDAFM - UDA FORMATTER: DMACR X04.01 23-AUG-82 14:02:32 PAGE 5-18
 CROSS REFERENCE TABLE (CREF V04.00)

ST.IN	3-177#													
ST.LBN	5-193#	20-11	40-28	42-10*	59-12	61-9	61-25	63-81	63-91	63-103	65-30	67-28	67-94	71-78
	71-181	72-31	73-70	82-39	84-29	84-95	87-39	91-89						
ST.PS	3-171#	12-32												
ST.RBN	5-194#	42-16*	54-133	56-141	60-12	64-22	69-22	83-33	91-71					
ST.RU	3-170#	12-29												
ST.SR	3-174#	12-44												
ST.WE	3-178#	12-57												
ST.WP	3-173#													
ST.XBN	5-195#	42-26*	47-43	50-39	62-9	62-10	62-77	75-26	75-95	86-16	86-17	86-33	88-37	88-129
	90-34	91-38												
STACK	5-134#	23-3												
STARIT	5-153#	48-11*	48-38	48-41*	55-11*	55-38	55-41*							
START	5-4	14-10	14-16	15-14	15-23	15-32	22-5#	38-5	38-5	46-8	46-8	49-10	49-10	52-5
	52-5	56-2	56-2	58-6	58-6	62-5	62-5	63-2	63-2	67-2	67-2	68-7	68-7	73-2
	73-2	75-2	75-2	76-7	76-7	77-7	77-7	78-5	78-5	85-2	85-2	90-2	90-2	91-2
	91-2	93-5	93-5											
START2	68-11	71-7#												
START3	22-5	23-3#												
STASEC	5-102#	88-7*	88-8*	88-66	88-68	88-81	88-83	88-84	88-100*	88-101*	88-110	88-113	88-114	
STATFR	18-45	18-47	18-53#											
STATRE	18-43#	18-49												
STATRT	18-52#	18-55												
STATST	12-14#	12-43	12-47	12-51	12-55	12-60								
STATUS	3-55#	18-43												
STATVL	13-62	13-66	13-141	13-193	13-212	18-41#	44-10							
STCKSV	5-135#	13-53*	13-55	25-5*	25-55									
STCLR	3-128#	42-9	42-15	42-25	42-35									
STCYL	3-83#	38-145	65-12	67-11	72-13	73-49	82-21	83-13	84-8	87-22				
STDBN	3-82#	42-30												
STDIAG	12-49	12-52#												
STFLAG	3-248#	25-7	25-16	25-16	25-16	25-25								
STFORM	12-45	12-48#												
STLBN	3-79#	42-8												
STO	5-169#	13-140	41-83*											
STPNIC	12-27	12-30	12-33	12-40	12-71#									
STRBN	3-80#	42-14												
STSC	3-262#													
STSK1	12-20	12-23#												
STSKP	12-62	12-68#												
STWLK	12-53	12-56#												
STXBN	3-81#	42-20												
SWAP	63-77	63-86#												
SWRD	6-14#	38-203	54-229											
TALIP1	13-54	13-56#												
TALK	12-15	12-64	12-66	13-13#	13-121	13-129	13-139	13-172	13-180	17-27				
TALKDN	13-33	13-37#												
TALKIP	13-51	13-55#												
TALKP	13-43	13-45#												
TALKRT	13-38	13-42#												
TATTN1	13-69	13-73#												
TBLK	5-155#	39-17*	39-20*	48-30	55-30									
TBUFF	22-72#	22-75	23-12	23-28										
TBUFFL	22-75#	23-13	23-29											
TCLEAR	13-24	13-36	13-41	13-50#										
TEMP	5-37#	20-8*	20-10	30-15*	30-16*	30-17*	30-19*	30-21*	30-22	30-36*	30-38	30-40*	37-14*	37-15*

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE S-19
 CROSS REFERENCE TABLE (CREF V04.00)

	37-18	37-21	37-22	37-79*	37-81*	37-84	38-28*	38-29*	38-31	38-40*	38-41*	38-42	38-52*	38-53*
	38-54	38-65*	38-68*	38-69	38-85*	38-88*	38-89	38-117*	38-118*	38-119	38-128*	38-131*	38-132	38-156*
	38-157*	38-160	38-172*	38-173*	38-176	38-181	38-216*	38-217*	38-218	39-10*	39-11*	39-12	39-15*	40-20*
	40-21*	40-22	41-4*	41-5*	41-6	41-9	41-14*	41-15*	41-18	41-21*	41-56*	41-57*	41-60	47-20*
	47-21*	47-22	47-37*	47-38*	47-39	49-42*	49-43*	49-46	49-53*	49-54*	49-55	50-8*	50-9*	50-10
	50-31*	50-32*	50-33	56-76*	56-77*	56-80	56-87*	56-88*	56-89	59-5*	59-6*	59-7*	59-8	60-5*
	60-6*	60-7*	60-8	61-5*	61-6*	61-7	62-73*	62-74*	62-75	64-8*	64-9*	64-13	69-8*	69-9*
	69-13	73-27*	73-28*	73-29	73-109	74-8	81-8*	81-9*	81-12	82-6*	83-64*	83-65*	86-30	86-31*
	86-32*	86-33*	88-78*	88-79*	88-80	88-107*	88-108*	88-109	91-55*	91-56*	91-57	91-63*	91-64*	91-65
	91-81*	91-82*	91-83											
TEMP2	5-33*	63-50*	63-51*	63-58*	63-64	63-86*	63-87*	63-96	71-73*	71-74*	71-104	71-140*	71-141*	71-152
	83-18*	83-19*	83-21											
TERR	13-63	13-67	13-70#											
THREB	3-295#	39-17												
TILOP	41-76#	41-78												
TILOP1	41-80#	41-82												
TIMER	13-64	13-65	18-25#											
TIMLO1	41-49	41-51#												
TIMLOP	41-53#	41-55												
TIMLP	18-27#	18-32												
TIMTBL	22-98#	26-22	37-32											
TIMVAL	3-312#	18-26												
TKIP1	49-17	49-20#												
TKIP11	49-87	49-89#												
TKIP12	49-64#	49-91												
TKIP13	49-101	49-103#												
TKIP14	49-97	49-102#												
TKIP2	49-15	49-19	49-21#	49-32										
TKIP3	49-62	49-93#												
TKIP4	49-67	49-71	49-82	49-85#										
TKIP5	49-39	49-41	49-60#											
TKIP7	49-74	49-76#												
TKIP8	49-49#	49-52												
TKIP9	49-50	49-53#												
TMPTRY	6-51#	41-40*	62-103*	62-105*	62-112*	62-118*	65-43*	65-45*	65-52*	65-58*	67-55*	67-57*	67-64*	67-73*
	67-77*	67-107*	67-109*	67-116*	72-45*	72-47*	72-54*	72-60*	73-83*	73-85*	73-92*	73-98*	75-54*	75-56*
	75-63*	75-73*	75-77*	75-109*	75-111*	75-118*	82-53*	82-55*	82-58*	83-47*	83-49*	83-56*	83-60*	84-56*
	84-58*	84-65*	84-74*	84-78*	84-108*	84-110*	84-117*	86-68*	86-70*	86-77*	86-89*	87-63*	87-65*	87-72*
	87-79*	87-86*	90-48*	90-50*	90-57*	90-63*								
TOTRCT	6-44#	38-78*	38-79*	52-22	63-20	71-29	73-36	87-7						
TRK	3-261#	19-12	82-79	89-13										
TRKCNT	6-35#	46-73*	46-102*	52-40*	52-102*	88-17*	88-86*	88-93*						
TRKCYL	5-118#	38-33*	38-34*	38-43	38-55	38-110								
TRKGRP	3-90#	38-22	46-71	52-38	88-15	88-91	92-20							
TWO	39-16	39-20#												
TWOB	3-294#	39-20												
TWOC	6-38#	71-234												
TWRD	6-15#	38-205	54-230											
UDAFM	5-3#													
UERR	14-29#	16-12	16-21	16-32										
UHASH	63-32	64-4#												
UHKIP	64-8#	64-23												
UHKIP1	64-5	64-20#												
UID	3-323#	44-35	44-63											
UN.ERI	5-12#	13-86*	13-95*	13-96*	13-99*	44-19*	44-20*	44-23*	44-82*	46-87*	46-88*	46-94*	47-180*	47-181*
	47-187*	52-54*	52-55*	52-61*	54-261*	54-262*	54-269*							

UDAFM - UDA FORMATTER DMACR X04.01 23-AUG-82 14:02:32 PAGE S-21
 CROSS REFERENCE TABLE (CREF V04.00)

XNOINC	66-21	66-23#			
XPBN	49-20	50-31#			
XPERR	66-44	66-52#			
XPNGRD	66-31#	66-47			
XPRET	66-17	66-50#			
XSKIP1	52-10	52-13#			
XSKIP2	52-45#	52-60			
XSKIP3	52-43#	52-103			
XSKIP4	52-65	52-70#			
XSKIP5	52-72	52-79	52-82#		
XSKIP6	52-76	52-80#			
XSLEEK	52-25#	52-115	52-117		
XSLEK2	52-31#	52-106			
XYZ	71-11	71-139	71-166	71-177	71-186
XYZ1	71-52	71-113	71-192#		71-191#
Y	22-76#	34-17			
YES	86-73	86-83#			

