

[illegible]

val

[illegible]

```

LL               IIIIII               SSSSSSSS
LL               IIIIII               SSSSSSSS
LL               II                    SS
LL               II                    SS
LL               II                    SS
LL               II                    SS
LL               II                    SSSSSS
LL               II                    SSSSSS
LL               II                    S
LL               II                    S
LL               II                    S
LL               II                    S
LLLLLLLLLLLLLL  IIIIII               SSSSSSSS
LLLLLLLLLLLLLL  IIIIII               SSSSSSSS

```

```
1 0001 0 MODULE RECORDUTL(%TITLE 'Queue record manipulation utilities'
2 0002 0 IDENT = 'V04-000'
3 0003 0 ) =
4 0004 1 BEGIN
5 0005 1
6 0006 1
7 0007 1 *****
8 0008 1 *
9 0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
10 0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
11 0011 1 * ALL RIGHTS RESERVED.
12 0012 1 *
13 0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
14 0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
15 0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
16 0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
17 0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
18 0018 1 * TRANSFERRED.
19 0019 1 *
20 0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
21 0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
22 0022 1 * CORPORATION.
23 0023 1 *
24 0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
25 0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
26 0026 1 *
27 0027 1 *****
28 0028 1
29 0029 1
30 0030 1
31 0031 1 **
32 0032 1 FACILITY:
33 0033 1 Job controller.
34 0034 1
35 0035 1 ABSTRACT:
36 0036 1 This module contains utility routines to manipulate queue records.
37 0037 1
38 0038 1 ENVIRONMENT:
39 0039 1 VAX/VMS user and kernel mode.
40 0040 1 --
41 0041 1
42 0042 1 AUTHOR: M. Jack, CREATION DATE: 16-Feb-1982
43 0043 1
44 0044 1 MODIFIED BY:
45 0045 1
46 0046 1 V03-010 JAK0228 J A Krycka 20-Aug-1984
47 0047 1 Restrict total length of all parameter strings for a batch job.
48 0048 1
49 0049 1 V03-009 JAK0220 J A Krycka 18-Jul-1984
50 0050 1 Support SJCS_PAGINATE at the queue level in addition to the
51 0051 1 job and file levels.
52 0052 1
53 0053 1 V03-008 GRR0003 Gregory R. Robert 16-Sep-1983
54 0054 1 Remove delete access check.
55 0055 1
56 0056 1 V03-007 GRR0002 Gregory R. Robert 09-Sep-1983
57 0057 1 Make FIB non-local to repair asynchronous stack corruption.
```

58	0058	1	
59	0059	1	
60	0060	1	V03-006 MLJ0118 Martin L. Jack, 23-Aug-1983
61	0061	1	Change field name, implement page setup, fix file deletion bug.
62	0062	1	
63	0063	1	V03-005 MLJ0115 Martin L. Jack, 30-Jul-1983
64	0064	1	Changes for job controller baselevel.
65	0065	1	
66	0066	1	V03-004 MLJ0114 Martin L. Jack, 23-Jun-1983
67	0067	1	Split from QUEUEUTIL.
68	0068	1	
69	0069	1	V03-003 MLJ0113 Martin L. Jack, 26-May-1983
70	0070	1	Changes for job controller baselevel.
71	0071	1	
72	0072	1	V03-002 MLJ0112 Martin L. Jack, 29-Apr-1983
73	0073	1	Changes for job controller baselevel.
74	0074	1	
75	0075	1	V03-001 MLJ0109 Martin L. Jack, 14-Apr-1983
76	0076	1	Changes for job controller baselevel.
77	0077	1	**

```

79 0078 1 REQUIRE 'SRCS:JOBCTLDEF';
80 1119 1
81 1120 1
82 1121 1 FORWARD ROUTINE
83 1122 1 UPDATE_SF_X_ENTRY: NOVALUE,
84 1123 1 UPDATE_SF_X_ENTRY: NOVALUE,
85 1124 1 CREATE_SF_M_RECORD: L_OUTPUT_2,
86 1125 1 APPLY_SF_M_OPTIONS,
87 1126 1 DELETE_SF_M_RECORD: NOVALUE,
88 1127 1 CREATE_SF_M_RECORD: L_OUTPUT_2,
89 1128 1 APPLY_SF_M_OPTIONS,
90 1129 1 DELETE_SF_M_RECORD: NOVALUE,
91 1130 1 CREATE_SF_H_RECORD: L_OUTPUT_2,
92 1131 1 APPLY_SF_H_OPTIONS,
93 1132 1 DELETE_SF_H_RECORD: NOVALUE,
94 1133 1 CREATE_SF_Q_RECORD: L_OUTPUT_2,
95 1134 1 APPLY_SF_Q_OPTIONS,
96 1135 1 DELETE_SF_Q_RECORD: NOVALUE,
97 1136 1 DELETE_FILES: NOVALUE,
98 1137 1 DELETE_ONE_FILE: NOVALUE,
99 1138 1 DELETE_ONE_FILE_AST: NOVALUE;
100 1139 1
101 1140 1
102 1141 1 EXTERNAL ROUTINE
103 1142 1 ALLOCATE_ENTRY_NUMBER,
104 1143 1 ALLOCATE_MEMORY,
105 1144 1 ALLOCATE_RECORD: L_OUTPUT_2,
106 1145 1 CREATE_SF_Q_RECORD,
107 1146 1 DEALLOCATE_ENTRY_NUMBER: NOVALUE,
108 1147 1 DEALLOCATE_MEMORY: NOVALUE,
109 1148 1 DEALLOCATE_RECORD: NOVALUE,
110 1149 1 DEALLOCATE_VARIABLE_DATA: NOVALUE,
111 1150 1 LOCK_QUEUE_FILE: NOVALUE,
112 1151 1 READ_RECORD,
113 1152 1 REWRITE_RECORD: NOVALUE,
114 1153 1 SCAN_INCOMPLETE_SERVICES: NOVALUE,
115 1154 1 SCHEDULE_NONAST: NOVALUE,
116 1155 1 STORE_VARIABLE_DATA,
117 1156 1 STORE_VARIABLE_DATA_LIST,
118 1157 1 UNLOCK_QUEUE_FILE: NOVALUE;
119 1158 1
120 1159 1
121 1160 1 EXTERNAL
122 1161 1 SYSSGB_DEFPRI: BYTE ADDRESSING_MODE(GENERAL),
123 1162 1 SYSSGB_DEFQUEPRI: BYTE ADDRESSING_MODE(GENERAL);
124 1163 1
125 1164 1
126 1165 1 BUILTIN
127 1166 1 LOCC,
128 1167 1 TESTBITSS;

```

```

130 1168 1 GLOBAL ROUTINE UPDATE_SFX_ENTRY(SFE,SFM_N,SFM): NOVALUE=
131 1169 1
132 1170 1 ++
133 1171 1
134 1172 1 FUNCTIONAL DESCRIPTION:
135 1173 1 This routine updates an SFX entry from the corresponding SFM record.
136 1174 1
137 1175 1 INPUT PARAMETERS:
138 1176 1 SFE - Pointer to SFX entry.
139 1177 1 SFM_N - Record number of SFM.
140 1178 1 SFM - Pointer to SFM.
141 1179 1
142 1180 1 IMPLICIT INPUTS:
143 1181 1 NONE
144 1182 1
145 1183 1 OUTPUT PARAMETERS:
146 1184 1 NONE
147 1185 1
148 1186 1 IMPLICIT OUTPUTS:
149 1187 1 NONE
150 1188 1
151 1189 1 ROUTINE VALUE:
152 1190 1 NONE
153 1191 1
154 1192 1 SIDE EFFECTS:
155 1193 1 NONE
156 1194 1
157 1195 1 --
158 1196 1
159 1197 2 BEGIN
160 1198 2 MAP
161 1199 2 SFE: REF BBLOCK, ! Pointer to SFX entry
162 1200 2 SFM: REF BBLOCK; ! Pointer to SFM
163 1201 2
164 1202 2
165 1203 2 CH$MOVE(SFM$S_NAME, SFM[SFM$T_NAME], SFE[SFX$T_NAME]);
166 1204 2 SFE[SFX$S_LINK] = .SFM_N;
167 1205 2 SFE[SFX$S_NUMBER] = .SFM[SFM$S_NUMBER];
168 1206 1 END;

```

.TITLE RECORDUTL Queue record manipulation utilities
.IDENT \V04-000\

.PSECT COMMON,NOEXE, OVR,2

00000 DIAG_STORAGE BASE:
 .BLKB 0
00000 DIAG_TRACE:
 .BLKB 96
00060 DIAG_COUNT:
 .BLKB 96
000C0 DIAG_FLAGS:
 .BLKB 4
000C4 WORK_AREA:
 .BLKB 44
000F0 SNOJBC_COUNT:


```

00174 GETQUI_COUNT: .BLKB 132
0019C SNDACC_COUNT: .BLKB 40
001B8 SNDSMB_COUNT: .BLKB 28
00200 DIAG_STORAGE_END: .BLKB 72
00200 FLAGS: .BLKB 0
00204 IMAGE_DUMP_STSFLG: .BLKB 4
00208 THIS_SYSID: .BLKB 6
0020E .BLKB 2
00210 CUR_TIME: .BLKB 8
00218 HOURLY_TIME: .BLKB 8
00220 HOURLY_PARAMS: .BLKB 20
00234 SYMBIONT_COUNT: .BLKB 4
00238 QUEUE_REFERENCE_COUNT: .BLKB 4
0023C MBX_MESSAGE_COUNT: .BLKB 4
00240 MBX: .BLKB 4
00244 MBX_END: .BLKB 4
00248 MEMORY_FREE_QUEUES: .BLKB 40
00270 NONAST_WORK_QUEUE: .BLKB 8
00278 BCB_FREE_LIST: .BLKB 4
0027C BCB_ACTIVE_LIST: .BLKB 4
00280 GQL_FREE_LIST: .BLKB 4
00284 GQL_ACTIVE_LIST: .BLKB 4
00288 OPEN_GETQUI_LIST: .BLKB 4
0028C PROCESS_DATA_LIST: .BLKB 4
00290 SYMBIONT_CONTROL: .BLKB 4
00294 SPARE_AREA: .BLKB 12
002A0 REMOTE_REQUEST_LKSB: .BLKB 8
002A8 QUEUE_FILE_LKSB: .BLKB 8
002B0 QUEUE_LOCK_LKSB: .BLKB 8
002B8 RSP: .BLKB 8
002C0 JBC_PRIORITY:

```

002C4	JBC_PRIVILEGES:	.BLKB	4
002CC	JBC_QUOTAS:	.BLKB	8
0030E		.BLKB	66
00310	JBC_UIC:	.BLKB	2
00314	QUEUE_FAB:	.BLKB	4
00364	QUEUE_RAB:	.BLKB	80
003A8	QUEUE_NAM:	.BLKB	68
00408	QUEUE_XAB:	.BLKB	96
00460	QUEUE_RSA:	.BLKB	88
0055F		.BLKB	255
00560	QUEUE_ALQ:	.BLKB	1
00564	QUEUE_MBF:	.BLKB	4
00565		.BLKB	1
00568	ACCOUNTING_FABS:	.BLKB	3
00570	ACCOUNTING_RABS:	.BLKB	8
00578	ACCOUNT_FAB A:	.BLKB	8
005C8	ACCOUNT_RAB A:	.BLRB	80
0060C	ACCOUNT_NAM A:	.BLRB	68
0066C	ACCOUNT_RSA A:	.BLRB	96
0076B		.BLRB	255
0076C	ACCOUNT_FAB B:	.BLKB	1
007BC	ACCOUNT_RAB B:	.BLRB	80
00800	ACCOUNT_NAM B:	.BLRB	68
00860	ACCOUNT_RSA B:	.BLRB	96
0095F		.BLRB	255
00960	DIAG_FAB:	.BLKB	1
009B0	DIAG_RAB:	.BLKB	80
009F4	MBX_CHAN:	.BLKB	68
009F8	MBX_IOSB:	.BLKB	4
00A00	MBX_BUFFER:	.BLKB	8
00E00	VALUE_STORAGE_BASE:	.BLKB	1024
		.BLKB	0


```

00E00 ITEM_PRESENT:
      .BLKB 32
00E20 VALUE_GETQUI BASE:
      .BLKB 0
00E20 VALUE_ACCOUNTING MESSAGE:
      .BLKB 8
00E26 VALUE_ACCOUNTING TYPES:
      .BLKB 4
00E2A VALUE_AFTER TIME:
      .BLKB 8
00E32 VALUE_ALIGNMENT_PAGES:
      .BLKB 1
00E33 VALUE_BASE PRIORITY:
      .BLKB 1
00E34 VALUE_BATCH INPUT:
      .BLKB 6
00E3A VALUE_BATCH OUTPUT:
      .BLKB 10
00E44 VALUE_BUFFER COUNT:
      .BLKB 1
00E45 VALUE_CHARACTERISTIC_NAME:
      .BLKB 6
00E4B VALUE_CHARACTERISTIC_NUMBER:
      .BLKB 1
00E4C VALUE_CHARACTERISTICS:
      .BLKB 16
00E5C VALUE_CHECKPOINT DATA:
      .BLKB 8
00E62 VALUE_CLI:
      .BLKB 6
00E68 VALUE_CPU DEFAULT:
      .BLKB 4
00E6C VALUE_CPU LIMIT:
      .BLKB 4
00E70 VALUE_DESTINATION_QUEUE:
      .BLKB 8
00E78 VALUE_DEVICE NAME:
      .BLKB 6
00E7E VALUE_ENTRY NUMBER:
      .BLKB 4
00E82 VALUE_ENTRY NUMBER OUTPUT:
      .BLKB 10
00E8C VALUE_EXTEND QUANTITY:
      .BLKB 2
00E8E VALUE_FILE COPIES:
      .BLKB 1
00E8F VALUE_FILE IDENTIFICATION:
      .BLKB 36
00EB3 VALUE_FILE SETUP MODULES:
      .BLKB 8
00EB9 VALUE_FILE SPECIFICATION:
      .BLKB 6
00EBF VALUE_FIRST PAGE:
      .BLKB 4
00EC3 VALUE_FORM DESCRIPTION:
      .BLKB 6
00EC9 VALUE_FORM_LENGTH:

```

```

      .BLKB 1
00ECA VALUE_FORM MARGIN_BOTTOM:
      .BLKB 1
00ECB VALUE_FORM MARGIN_LEFT:
      .BLKB 2
00ECD VALUE_FORM MARGIN_RIGHT:
      .BLKB 2
00ECF VALUE_FORM MARGIN_TOP:
      .BLKB 1
00ED0 VALUE_FORM NAME:
      .BLKB 6
00ED6 VALUE_FORM NUMBER:
      .BLKB 4
00EDA VALUE_FORM:
      .BLKB 8
00EE2 VALUE_FORM SETUP_MODULES:
      .BLKB 8
00EE8 VALUE_FORM STOCK:
      .BLKB 6
00EEE VALUE_FORM WIDTH:
      .BLKB 2
00EFO VALUE_GENERIC_TARGET:
      .BLKB 996
012D4 VALUE_JOB_COPIES:
      .BLKB 1
012D5 VALUE_JOB_LIMIT:
      .BLKB 1
012D6 VALUE_JOB_NAME:
      .BLKB 6
012DC VALUE_JOB_RESET_MODULES:
      .BLKB 6
012E2 VALUE_JOB_SIZE_MAXIMUM:
      .BLKB 4
012E6 VALUE_JOB_SIZE_MINIMUM:
      .BLKB 4
012EA VALUE_JOB_STATUS_OUTPUT:
      .BLKB 10
012F4 VALUE_LAST_PAGE:
      .BLKB 4
012F8 VALUE_LIBRARY_SPECIFICATION:
      .BLKB 6
012FE VALUE_LOG_QUEUE:
      .BLKB 8
01306 VALUE_LOG_SPECIFICATION:
      .BLKB 6
0130C VALUE_NOTE:
      .BLKB 6
01312 VALUE_OPERATOR_REQUEST:
      .BLKB 6
01318 VALUE_OWNER_UID:
      .BLKB 4
0131C VALUE_PAGE_SETUP_MODULES:
      .BLKB 8
01322 VALUE_PARAMETER_1:
      .BLKB 6
01328 VALUE_PARAMETER_2:
      .BLKB 6

```

0132E VALUE_PARAMETER_3:
 .BLKB 6
01334 VALUE_PARAMETER_4:
 .BLKB 6
0133A VALUE_PARAMETER_5:
 .BLKB 6
01340 VALUE_PARAMETER_6:
 .BLKB 6
01346 VALUE_PARAMETER_7:
 .BLKB 6
0134C VALUE_PARAMETER_8:
 .BLKB 6
01352 VALUE_PRIORITY:
 .BLKB 1
01353 VALUE_PROCESSOR:
 .BLKB 6
01359 VALUE_PROTECTION:
 .BLKB 4
0135D VALUE_QUEUE:
 .BLKB 6
01363 VALUE_QUEUE_FILE_SPECIFICATION:
 .BLKB 8
01369 VALUE_RELATIVE_PAGE:
 .BLKB 4
0136D VALUE_RESERVED_INPUT_1:
 .BLKB 1
0136E VALUE_RESERVED_INPUT_2:
 .BLKB 2
01370 VALUE_RESERVED_INPUT_3:
 .BLKB 4
01374 VALUE_RESERVED_INPUT_4:
 .BLKB 6
0137A VALUE_RESERVED_OUTPUT_1:
 .BLKB 10
01384 VALUE_RESERVED_OUTPUT_2:
 .BLKB 10
0138E VALUE_SEARCH_STRING:
 .BLKB 6
01394 VALUE_SCNODE_NAME:
 .BLKB 6
0139A VALUE_WSDFAULT:
 .BLKB 2
0139C VALUE_WSEXTENT:
 .BLKB 2
0139E VALUE_WSQUOTA:
 .BLKB 2
013A0 VALUE_STORAGE_END:
 .BLKB 0

JBC\$_CLOSEOUT= 266328
JBC\$_NOCMKRNL= 272388
JBC\$_NOOPER= 272532
JBC\$_NOSYSNAM= 272404
JBC\$_OPENIN= 266392
JBC\$_OPENOUT= 266400
JBC\$_READERR= 266416
JBC\$_WRITEERR= 266448

```
.EXTRN ALLOCATE_ENTRY_NUMBER
.EXTRN ALLOCATE_MEMORY
.EXTRN ALLOCATE_RECORD
.EXTRN CREATE_SRO_RECORD
.EXTRN DEALLOCATE_ENTRY_NUMBER
.EXTRN DEALLOCATE_MEMORY
.EXTRN DEALLOCATE_RECORD
.EXTRN DEALLOCATE_VARIABLE_DATA
.EXTRN LOCK_QUEUE_FILE
.EXTRN READ_RECORD, REWRITE_RECORD
.EXTRN SCAN_INCOMPLETE_SERVICES
.EXTRN SCHEDULE_NONAST
.EXTRN STORE_VARIABLE_DATA
.EXTRN STORE_VARIABLE_DATA_LIST
.EXTRN UNLOCK_QUEUE_FILE
.EXTRN SYS$GB_DEFPR1, SYS$GB_DEFQUEPRI
```

```
.PSECT CODE, NOWRT, 2
```

```
.ENTRY UPDATE_SFX_ENTRY, Save R2,R3,R4,R5,R6,R7 : 1168
MOVL SFM, R7 : 1203
MOVL SFE, R6 :
MOVCL #32, 272(R7), (R6) :
MOVL SFM N, 36(R6) : 1204
MOVL 304(R7), 32(R6) : 1205
RET : 1206
```

```
00FC 00000
57 0C AC D0 00002
56 04 AC D0 00006
66 0110 C7 20 28 0000A
24 A6 08 AC D0 00010
20 A6 0130 C7 D0 00015
04 0001B
```

; Routine Size: 28 bytes, Routine Base: CODE + 0000

```

170 1207 1 GLOBAL ROUTINE UPDATE_SQX_ENTRY(SQE,SMQ_N,SMQ): NOVALUE=
171 1208 1
172 1209 1 ++
173 1210 1
174 1211 1 FUNCTIONAL DESCRIPTION:
175 1212 1 This routine updates an SQX entry from the corresponding SMQ record.
176 1213 1
177 1214 1 INPUT PARAMETERS:
178 1215 1 SQE - Pointer to SQX entry.
179 1216 1 SMQ_N - Record number of SMQ.
180 1217 1 SMQ - Pointer to SMQ.
181 1218 1
182 1219 1 IMPLICIT INPUTS:
183 1220 1 NONE
184 1221 1
185 1222 1 OUTPUT PARAMETERS:
186 1223 1 NONE
187 1224 1
188 1225 1 IMPLICIT OUTPUTS:
189 1226 1 NONE
190 1227 1
191 1228 1 ROUTINE VALUE:
192 1229 1 NONE
193 1230 1
194 1231 1 SIDE EFFECTS:
195 1232 1 NONE
196 1233 1
197 1234 1 --
198 1235 1
199 1236 2 BEGIN
200 1237 2 MAP
201 1238 2 SQE: REF BBLOCK, ! Pointer to SQX entry
202 1239 2 SMQ: REF BBLOCK; ! Pointer to SMQ
203 1240 2
204 1241 2
205 1242 2 CH$MOVE(SMQ$S_NAME, SMQ[SMQ$T_NAME], SQE[SQX$T_NAME]);
206 1243 2 SQE[SQX$S_QUEUE_LINK] = .SMQ_N;
207 1244 2 SQE[SQX$S_FLAGS] = 0;
208 1245 2 IF .SMQ[SMQ$S_BATCH]
209 1246 2 THEN SQE[SQX$S_BATCH] = TRUE;
210 1247 2 IF NOT .SMQ[SMQ$S_GENERIC_QUEUE] AND .SMQ[SMQ$S_ASSIGNED_QUEUE_LINK] EQL 0
211 1248 2 THEN SQE[SQX$S_EXECUTOR] = TRUE;
212 1249 2 IF .SMQ[SMQ$S_GENERIC_SELECTION]
213 1250 2 THEN SQE[SQX$S_GENERIC_SELECTION] = TRUE;
214 1251 1 END;

```

				00FC 00000	.ENTRY	UPDATE_SQX_ENTRY, Save R2,R3,R4,R5,R6,R7	: 1207
		57	0C	AC D0 00002	MOVL	SMQ, R7	: 1242
		56	04	AC D0 00006	MOVL	SQE, R6	
66	00B0	C7		20 28 0000A	MOV C3	#32, 176(R7), (R6)	
	24	A6	08	AC D0 00010	MOVL	SMQ_N, 36(R6)	: 1243
			20	A6 D4 00015	CLRL	32(R6)	: 1244
		04	0C	A7 E9 00018	BLBC	12(R7), 1\$	: 1245

09	20 0D	A6 A7		01 88 0001C 02 E0 00020 1\$: A7 D5 00025 04 12 00028	BISB2 #1, 32(R6) BBS #2, 13(R7), 2\$ TSTL 44(R7) BNEQ 2\$	:	1246 1247
			2C			:	
04	20 0D 20	A6 A7 A6		02 88 0002A 03 E1 0002E 2\$: 04 88 00033 04 00037 3\$:	BISB2 #2, 32(R6) BBC #3, 13(R7), 3\$ BISB2 #4, 32(R6) RET	:	1248 1249 1250 1251

; Routine Size: 56 bytes, Routine Base: CODE + 001C


```
216 1252 1 GLOBAL ROUTINE CREATE_SFM_RECORD(;SFM_N,SFM): L_OUTPUT_2=
217 1253 1
218 1254 1 ++
219 1255 1
220 1256 1 FUNCTIONAL DESCRIPTION:
221 1257 1 This routine allocates and initializes a form definition record.
222 1258 1
223 1259 1 INPUT PARAMETERS:
224 1260 1 NONE
225 1261 1
226 1262 1 I: LICIT INPUTS:
227 1263 1 NONE
228 1264 1
229 1265 1 OUTPUT PARAMETERS:
230 1266 1 SFM_N - Record number of SFM.
231 1267 1 SFM - Pointer to SFM.
232 1268 1
233 1269 1 IMPLICIT OUTPUTS:
234 1270 1 NONE
235 1271 1
236 1272 1 ROUTINE VALUE:
237 1273 1 Completion status.
238 1274 1
239 1275 1 SIDE EFFECTS:
240 1276 1 NONE
241 1277 1
242 1278 1 --
243 1279 1
244 1280 2 BEGIN
245 1281 2 MAP
246 1282 2 SFM: REF BBLOCK; ! Pointer to SFM
247 1283 2 LOCAL
248 1284 2 STATUS; ! Status return
249 1285 2
250 1286 2
251 1287 2 ! Allocate the queue record, and return if no more.
252 1288 2
253 1289 2 STATUS = ALLOCATE_RECORD( ; SFM_N, SFM);
254 1290 2 IF NOT .STATUS THEN RETURN .STATUS;
255 1291 2
256 1292 2
257 1293 2 ! Initialize the form definition record.
258 1294 2
259 1295 2 SFM[SYMSB_TYPE] = SYMSK_SFM;
260 1296 2 CH$WCHAR(
261 1297 2 .VALUE_FORM_NAME[SDSC_W_LENGTH],
262 1298 2 SFM[SFMST_NAME]);
263 1299 2 CH$COPY(
264 1300 2 .VALUE_FORM_NAME[SDSC_W_LENGTH],
265 1301 2 .VALUE_FORM_NAME[SDSC_A_POINTER],
266 1302 2 0,
267 1303 2 SFM$S_NAME-1,
268 1304 2 SFM[SFMST_NAME]+1);
269 1305 2 CH$COPY(
270 1306 2 SFM$S_NAME, SFM[SFMST_NAME],
271 1307 2 0,
272 1308 2 SFM$S_DESCRIPTION, SFM[SFMST_DESCRIPTION]);
```

```

: 273      1309 2 CH$MOVE(SFM$S NAME, SFM[SFM$T NAME], SFM[SFM$T_STOCK]);
: 274      1310 2 SFM[SFM$S_NUMBER] = .VALUE_FORM_NUMBER;
: 275      1311 2 SFM[SFM$S_LENGTH] = 66;
: 276      1312 2 SFM[SFM$S_WIDTH] = 132;
: 277      1313 2 SFM[SFM$S_MARGIN_BOTTOM] = 6;
: 278      1314 2 SFM[SFM$S_TRUNCATE] = TRUE;
: 279      1315 2
: 280      1316 2
: 281      1317 2 $$$_NORMAL
: 282      1318 1 END;

```

						00FC 00000	.ENTRY	CREATE SFM_RECORD, Save R2,R3,R4,R5,R6,R7	: 1252
						EF 9E 00002	MOVAB	VALUE_FORM_NAME, R7	
		00000000G	EF			00 FB 00009	CALLS	#0, A[LOCATE_RECORD	: 1289
			3A			50 E9 00010	BLBC	STATUS, 1\$	: 1290
		04	AB			02 90 00013	MOVB	#2, 4(SFM)	: 1295
			56	0110		CB 9E 00017	MOVAB	272(SFM), R6	: 1298
			66			67 90 0001C	MOVB	VALUE_FORM_NAME, (R6)	
	1F	00	02	B7		67 2C 0001F	MOVCS	VALUE_FORM_NAME, @VALUE_FORM_NAME+2, #0, -	: 1304
					0111	CB 00025		#31, 273(SFM)	
0100	8F	00		66		20 2C 00028	MOVCS	#32, (R6), #0, #256, 16(SFM)	: 1308
					10	AB 0002F			
		0134	CB	66		20 28 00031	MOVCS	#32, (R6), 308(SFM)	: 1309
		0130		CB	06	A7 D0 00037	MOVL	VALUE_FORM_NUMBER, 304(SFM)	: 1310
		0158		CB	06420084	8F D0 0003D	MOVL	#104988804, 344(SFM)	: 1312
		0C		AB		02 88 00046	BISB2	#2, 12(SFM)	: 1314
				50		01 D0 0004A	MOVL	#1, R0	: 1318
						04 0004D 1\$:	RET		

; Routine Size: 78 bytes, Routine Base: CODE + 0054


```
284 1319 1 GLOBAL ROUTINE APPLY_SFM_OPTIONS(SFM)=
285 1320 1
286 1321 1 ++
287 1322 1
288 1323 1 FUNCTIONAL DESCRIPTION:
289 1324 1 This routine applies options to a form definition record.
290 1325 1
291 1326 1 INPUT PARAMETERS:
292 1327 1 SFM - Pointer to SFM.
293 1328 1
294 1329 1 IMPLICIT INPUTS:
295 1330 1 Existing values in SFM.
296 1331 1
297 1332 1 OUTPUT PARAMETERS:
298 1333 1 NONE
299 1334 1
300 1335 1 IMPLICIT OUTPUTS:
301 1336 1 New values in SFM.
302 1337 1
303 1338 1 ROUTINE VALUE:
304 1339 1 Completion status.
305 1340 1
306 1341 1 SIDE EFFECTS:
307 1342 1 NONE
308 1343 1
309 1344 1 --
310 1345 1
311 1346 2 BEGIN
312 1347 2 MAP
313 1348 2 SFM: REF BBLOCK; ! Pointer to SFM
314 1349 2 LOCAL
315 1350 2 LSFM: BBLOCK[SYMSS_SYM], ! Saved copy of original SFM
316 1351 2 FINAL_STATUS; ! Final routine status
317 1352 2 LABEL
318 1353 2 PROTECTED_UPDATE;
319 1354 2
320 1355 2
321 1356 2 ! Set up for backing out the transaction in case of an error.
322 1357 2
323 1358 2 CH$MOVE(SYMSS_SYM, .SFM, LSFM);
324 1359 2 CH$FILL(0, SFM$$FORM_SETUP_MODULES, SFM[SFM$T_FORM_SETUP_MODULES]);
325 1360 2 CH$FILL(0, SFM$$PAGE_SETUP_MODULES, SFM[SFM$T_PAGE_SETUP_MODULES]);
326 1361 2 FINAL_STATUS = SSS_NORMAL;
327 1362 3 PROTECTED_UPDATE: BEGIN
328 1363 3
329 1364 3
330 1365 3 IF .ITEM_PRESENT[SJCS_FORM_DESCRIPTION]
331 1366 3 THEN
332 1367 4 BEGIN
333 1368 4 CH$WCHAR(
334 1369 4 .VALUE_FORM_DESCRIPTION[SDSC_W_LENGTH],
335 1370 4 SFM[SFM$T_DESCRIPTION]);
336 1371 4 CH$COPY(
337 1372 4 .VALUE_FORM_DESCRIPTION[SDSC_W_LENGTH],
338 1373 4 .VALUE_FORM_DESCRIPTION[SDSC_A_POINTER],
339 1374 4 0,
340 1375 4 SFM$$DESCRIPTION-1,
```

```
341 1376 4      SFM[SFM$T_DESCRIPTION]+1);
342 1377 3      END;
343 1378 3
344 1379 3
345 1380 3      IF .ITEM_PRESENT[SJCS_FORM_LENGTH]
346 1381 3      THEN
347 1382 3          SFM[SFM$B_LENGTH] = .VALUE_FORM_LENGTH;
348 1383 3
349 1384 3
350 1385 3      IF .ITEM_PRESENT[SJCS_FORM_MARGIN_BOTTOM]
351 1386 3      THEN
352 1387 3          SFM[SFM$B_MARGIN_BOTTOM] = .VALUE_FORM_MARGIN_BOTTOM;
353 1388 3
354 1389 3
355 1390 3      IF .ITEM_PRESENT[SJCS_FORM_MARGIN_LEFT]
356 1391 3      THEN
357 1392 3          SFM[SFM$W_MARGIN_LEFT] = .VALUE_FORM_MARGIN_LEFT;
358 1393 3
359 1394 3
360 1395 3      IF .ITEM_PRESENT[SJCS_FORM_MARGIN_RIGHT]
361 1396 3      THEN
362 1397 3          SFM[SFM$W_MARGIN_RIGHT] = .VALUE_FORM_MARGIN_RIGHT;
363 1398 3
364 1399 3
365 1400 3      IF .ITEM_PRESENT[SJCS_FORM_MARGIN_TOP]
366 1401 3      THEN
367 1402 3          SFM[SFM$B_MARGIN_TOP] = .VALUE_FORM_MARGIN_TOP;
368 1403 3
369 1404 3
370 1405 3      IF .ITEM_PRESENT[SJCS_FORM_NUMBER]
371 1406 3      THEN
372 1407 3          SFM[SFM$L_NUMBER] = .VALUE_FORM_NUMBER;
373 1408 3
374 1409 3
375 1410 3      IF .ITEM_PRESENT[SJCS_FORM_SETUP_MODULES]
376 1411 3      THEN
377 1412 4          BEGIN
378 1413 4              FINAL_STATUS = STORE_VARIABLE_DATA(
379 1414 4                  .SFM,
380 1415 4                  SFM$S_FORM_SETUP_MODULES,
381 1416 4                  SFM[SFM$T_FORM_SETUP_MODULES],
382 1417 4                  SYMSK_FORM_SETUP_MODULES,
383 1418 4                  .VALUE_FORM_SETUP_MODULES[SDSC_W_LENGTH],
384 1419 4                  .VALUE_FORM_SETUP_MODULES[SDSC_A_POINTER]);
385 1420 4              IF NOT .FINAL_STATUS THEN LEAVE PROTECTED_UPDATE;
386 1421 3          END;
387 1422 3
388 1423 3
389 1424 3      IF .ITEM_PRESENT[SJCS_FORM_SHEET_FEED]
390 1425 3      THEN
391 1426 3          SFM[SFM$V_SHEET_FEED] = TRUE;
392 1427 3
393 1428 3
394 1429 3      IF .ITEM_PRESENT[SJCS_NO_FORM_SHEET_FEED]
395 1430 3      THEN
396 1431 3          SFM[SFM$V_SHEET_FEED] = FALSE;
397 1432 3
```

```
398 1433 3
399 1434 3 IF .ITEM_PRESENT[SJC$_FORM_STOCK]
400 1435 3 THEN
401 1436 4 BEGIN
402 1437 4   CH$WCHAR(
403 1438 4     .VALUE_FORM_STOCK[SDSC_W_LENGTH],
404 1439 4     SFM[SFM$T_STOCK]);
405 1440 4   CH$COPY(
406 1441 4     .VALUE_FORM_STOCK[SDSC_W_LENGTH],
407 1442 4     .VALUE_FORM_STOCK[SDSC_A_POINTER],
408 1443 4     0,
409 1444 4     SFM$S_STOCK-1,
410 1445 4     SFM[SFM$T_STOCK]+1);
411 1446 3   END;
412 1447 3
413 1448 3 IF .ITEM_PRESENT[SJC$_FORM_TRUNCATE]
414 1449 3 THEN
415 1450 4 BEGIN
416 1451 4   SFM[SFM$V_TRUNCATE] = TRUE;
417 1452 4   SFM[SFM$V_WRAP] = FALSE;
418 1453 4   END;
419 1454 3
420 1455 3 IF .ITEM_PRESENT[SJC$_NO_FORM_TRUNCATE]
421 1456 3 THEN
422 1457 4   SFM[SFM$V_TRUNCATE] = FALSE;
423 1458 3
424 1459 3 IF .ITEM_PRESENT[SJC$_FORM_WIDTH]
425 1460 3 THEN
426 1461 4   SFM[SFM$W_WIDTH] = .VALUE_FORM_WIDTH;
427 1462 3
428 1463 3 IF .ITEM_PRESENT[SJC$_FORM_WRAP]
429 1464 3 THEN
430 1465 4 BEGIN
431 1466 4   SFM[SFM$V_TRUNCATE] = FALSE;
432 1467 4   SFM[SFM$V_WRAP] = TRUE;
433 1468 4   END;
434 1469 3
435 1470 3 IF .ITEM_PRESENT[SJC$_NO_FORM_WRAP]
436 1471 3 THEN
437 1472 4   SFM[SFM$V_WRAP] = FALSE;
438 1473 3
439 1474 3 IF .ITEM_PRESENT[SJC$_PAGE_SETUP_MODULES]
440 1475 3 THEN
441 1476 4 BEGIN
442 1477 4   FINAL STATUS = STORE_VARIABLE_DATA(
443 1478 4     .SFM,
444 1479 4     SFM$S_PAGE_SETUP_MODULES,
445 1480 4     SFM[SFM$T_PAGE_SETUP_MODULES],
446 1481 4     SYMSK_PAGE_SETUP_MODULES,
447 1482 4     .VALUE_PAGE_SETUP_MODULES[SDSC_W_LENGTH],
448 1483 4     .VALUE_PAGE_SETUP_MODULES[SDSC_A_POINTER]);
449 1484 4
450 1485 4
451 1486 4
452 1487 4
453 1488 4
454 1489 4
```

```
455 1490 4 IF NOT .FINAL_STATUS THEN LEAVE PROTECTED_UPDATE;
456 1491 3 END;
457 1492 3
458 1493 3
459 1494 3 IF .SFM[SFM$W_MARGIN_LEFT] + .SFM[SFM$W_MARGIN_RIGHT] GEQU .SFM[SFM$W_WIDTH]
460 1495 3 OR .SFM[SFM$B_MARGIN_TOP] + .SFM[SFM$B_MARGIN_BOTTOM] GEQU .SFM[SFM$B_LENGTH]
461 1496 3 THEN
462 1497 4 BEGIN
463 1498 4 FINAL_STATUS = JBC$ INCFORMPAR;
464 1499 4 LEAVE PROTECTED_UPDATE;
465 1500 3 END;
466 1501 3
467 1502 3
468 1503 2 END; ! block PROTECTED_UPDATE
469 1504 2
470 1505 2
471 1506 2 IF .FINAL_STATUS
472 1507 2 THEN
473 1508 3 BEGIN
474 1509 3 IF .ITEM_PRESENT[SJC$ FORM SETUP MODULES]
475 1510 3 OR .ITEM_PRESENT[SJC$ NO FORM SETUP MODULES]
476 1511 3 THEN
477 1512 3 DEALLOCATE VARIABLE DATA(
478 1513 3 SFM$ FORM SETUP MODULES,
479 1514 3 LSFMSFM$ FORM SETUP MODULES))
480 1515 3 ELSE
481 1516 3 CH$MOVE(
482 1517 3 SFM$ FORM SETUP MODULES,
483 1518 3 LSFMSFM$ FORM SETUP MODULES,
484 1519 3 SFMSFM$ FORM SETUP MODULES));
485 1520 3
486 1521 3
487 1522 3 IF .ITEM_PRESENT[SJC$ PAGE SETUP MODULES]
488 1523 3 OR .ITEM_PRESENT[SJC$ NO PAGE SETUP MODULES]
489 1524 3 THEN
490 1525 3 DEALLOCATE VARIABLE DATA(
491 1526 3 SFM$ PAGE SETUP MODULES,
492 1527 3 LSFMSFM$ PAGE SETUP MODULES))
493 1528 3 ELSE
494 1529 3 CH$MOVE(
495 1530 3 SFM$ PAGE SETUP MODULES,
496 1531 3 LSFMSFM$ PAGE SETUP MODULES,
497 1532 3 SFMSFM$ PAGE SETUP MODULES));
498 1533 3 END
499 1534 2 ELSE
500 1535 3 BEGIN
501 1536 3 DEALLOCATE VARIABLE DATA(
502 1537 3 SFM$ FORM SETUP MODULES,
503 1538 3 SFMSFM$ FORM SETUP MODULES));
504 1539 3 DEALLOCATE VARIABLE DATA(
505 1540 3 SFM$ PAGE SETUP MODULES,
506 1541 3 SFMSFM$ PAGE SETUP MODULES));
507 1542 3 CH$MOVE(SYM$ SYM, LSFMSFM);
508 1543 2 END;
509 1544 2
510 1545 2
511 1546 2 .FINAL_STATUS
```

: 512

1547 1 END;

					OFFC 00000				.ENTRY	APPLY_SFMM_OPTIONS, Save R2,R3,R4,R5,R6,R7,-	1319
										R8,R9,R10,R11	
									MOVAB	DEALLOCATE VARIABLE_DATA, R11	
									MOVAB	ITEM_PRESENT+7, R10	
									MOVAB	-512(SP), SP	
									MOVL	SFM, R6	1358
									MOVCS	#512, (R6), LSFMM	
									MOVAB	349(R6), R9	1359
									MOVCS	#0, (SP), #0, #6, (R9)	
									MOVAB	355(R6), R8	1360
									MOVCS	#0, (SP), #0, #6, (R8)	
									MOVL	#1, FINAL STATUS	1361
									BLBC	ITEM_PRESENT+6, 1\$	1365
									MOVAB	VALUE_FORM_DESCRIPTION, 16(R6)	1370
									MOVCS	VALUE_FORM_DESCRIPTION, -	1376
										@VALUE_FORM_DESCRIPTION+2, #0, #255, 17(R6)	
									BBC	#1, ITEM_PRESENT+6, 2\$	1380
									MOVAB	VALUE_FORM_LENGTH, 346(R6)	1382
									BBC	#2, ITEM_PRESENT+6, 3\$	1385
									MOVAB	VALUE_FORM_MARGIN_BOTTOM, 347(R6)	1387
									BBC	#3, ITEM_PRESENT+6, 4\$	1390
									MOVW	VALUE_FORM_MARGIN_LEFT, 340(R6)	1392
									BBC	#4, ITEM_PRESENT+6, 5\$	1395
									MOVW	VALUE_FORM_MARGIN_RIGHT, 342(R6)	1397
									BBC	#5, ITEM_PRESENT+6, 6\$	1400
									MOVAB	VALUE_FORM_MARGIN_TOP, 348(R6)	1402
									TSTB	ITEM_PRESENT+6	1405
									BGEQ	7\$	
									MOVL	VALUE_FORM_NUMBER, 304(R6)	1407
									BLBC	ITEM_PRESENT+7, 8\$	1410
									PUSHL	VALUE_FORM_SETUP_MODULES+2	1419
									MOVZWL	VALUE_FORM_SETUP_MODULES, -(SP)	1418
									PUSHL	#13	1416
									PUSHL	R9	
									PUSHL	#6	
									PUSHL	R6	
									CALLS	#6, STORE_VARIABLE_DATA	
									MOVL	R0, FINAL_STATUS	
									BLBC	FINAL_STATUS, 17\$	1420
									BBC	#2, ITEM_PRESENT+7, 9\$	1424
									BISB2	#1, 12(R6)	1426
									BBC	#3, ITEM_PRESENT+7, 10\$	1429
									BICB2	#1, 12(R6)	1431
									BBC	#4, ITEM_PRESENT+7, 11\$	1434
									MOVAB	VALUE_FORM_STOCK, 308(R6)	1439
									MOVCS	VALUE_FORM_STOCK, @VALUE_FORM_STOCK+2, #0, -	1445
										#31, 309(R6)	
									BBC	#5, ITEM_PRESENT+7, 12\$	1449
									BISB2	#2, 12(R6)	1452

04	0C	A6	04	8A	000E7	BICB2	#4, 12(R6)	1453	
		6A	06	E1	000EB	BBC	#6, ITEM_PRESENT+7, 13\$	1457	
	0C	A6	02	8A	000EF	BICB2	#2, 12(R6)	1459	
			6A	95	000F3	TSTB	ITEM_PRESENT+7	1462	
	0158	C6	07	18	000F5	BGEQ	14\$		
		08	CA	B0	000F7	MOVW	VALUE FORM WIDTH, 344(R6)	1464	
			AA	E9	000FE	BLBC	ITEM_PRESENT+8, 15\$	1467	
	0C	A6	02	8A	00102	BICB2	#2, T2(R6)	1470	
	0C	A6	04	88	00106	BISB2	#4, 12(R6)	1471	
04	01	AA	01	E1	0010A	BBC	#1, ITEM_PRESENT+8, 16\$	1475	
	0C	A6	04	8A	0010F	BICB2	#4, 12(R6)	1477	
1E	07	AA	03	E1	00113	BBC	#3, ITEM_PRESENT+14, 18\$	1480	
			CA	DD	00118	PUSHL	VALUE_PAGE_SETUP_MODULES+2	1489	
		7E	CA	3C	0011C	MOVZWL	VALUE_PAGE_SETUP_MODULES, -(SP)	1488	
			13	DD	00121	PUSHL	#19	1486	
			58	DD	00123	PUSHL	R8		
			06	DD	00125	PUSHL	#6		
			56	DD	00127	PUSHL	R6		
	00000000G	EF	C6	FB	00129	CALLS	#6, STORE_VARIABLE_DATA		
		57	50	DD	00130	MOVL	R0, FINAL_STATUS		
		6B	57	E9	00133	BLBC	FINAL_STATUS, 26\$	1490	
		50	C6	3C	00136	MOVZWL	340(R6), R0	1494	
		51	C6	3C	00138	MOVZWL	342(R6), R1		
		50	51	CO	00140	ADDL2	R1, R0		
50	0158	C6	10	00	ED	CMPZV	#0, #16, 344(R6), R0		
				16	1B	BLEQU	19\$		
		50	C6	9A	0014C	MOVZBL	348(R6), R0	1495	
		51	C6	9A	00151	MOVZBL	347(R6), R1		
		50	51	CO	00156	ADDL2	R1, R0		
50	015A	C6	08	00	ED	CMPZV	#0, #8, 346(R6), R0		
			57	07	1A	BGTRU	20\$		
			35	00	DD	00162	MOVL	#295272, FINAL_STATUS	1498
			04	57	E9	00169	BLBC	FINAL_STATUS, 26\$	1506
	0B	6A	01	6A	E8	0016C	BLBS	ITEM_PRESENT+7, 21\$	1509
			01	E1	0016F	BBC	#1, ITEM_PRESENT+7, 22\$	1510	
			CD	9F	00173	PUSHAB	LSFM+349	1514	
			06	DD	00177	PUSHL	#6		
		6B	02	FB	00179	CALLS	#2, DEALLOCATE_VARIABLE_DATA		
			06	11	0017C	BRB	23\$		
			06	28	0017E	MOV3	#6, LSFM+349, (R9)	1519	
69	FF5D	CD	03	E0	00184	BBS	#3, ITEM_PRESENT+14, 24\$	1522	
05	07	AA	04	E1	00189	BBC	#4, ITEM_PRESENT+14, 25\$	1523	
0B	07	AA	CD	9F	0018E	PUSHAB	LSFM+355	1527	
			06	DD	00192	PUSHL	#6		
		6B	02	FB	00194	CALLS	#2, DEALLOCATE_VARIABLE_DATA		
			1C	11	00197	BRB	27\$		
68	FF63	CD	06	28	00199	MOV3	#6, LSFM+355, (R8)	1532	
			14	11	0019F	BRB	27\$	1533	
			59	DD	001A1	PUSHL	R9	1538	
			06	DD	001A3	PUSHL	#6		
		6B	02	FB	001A5	CALLS	#2, DEALLOCATE_VARIABLE_DATA		
			58	DD	001A8	PUSHL	R8	1541	
			06	DD	001AA	PUSHL	#6		
		6B	02	FB	001AC	CALLS	#2, DEALLOCATE_VARIABLE_DATA		
66		6E	8F	28	001AF	MOV3	#512, LSFM, (R6)	1542	
		50	57	DD	001B5	MOVL	FINAL_STATUS, R0	1547	
			04	001B8	RET				

RECORDUTL
V04-000

Queue record manipulation utilities

D 14
16-Sep-1984 00:18:27
14-Sep-1984 12:37:12

VAX-11 Bliss-32 V4.0-742
[JOBCTL.SRC]RECORDUTL.B32;1

Page 21
(6)

RE
VO

; Routine Size: 441 bytes, Routine Base: CODE + 00A2

```

1548 1 GLOBAL ROUTINE DELETE_SFM_RECORD(SFM_N,SFM): NOVALUE=
1549 1
1550 1 ++
1551 1
1552 1 FUNCTIONAL DESCRIPTION:
1553 1 This routine deallocates a form definition record and its substructures.
1554 1
1555 1 INPUT PARAMETERS:
1556 1 SFM_N - Record number of SFM.
1557 1 SFM - Pointer to SFM.
1558 1
1559 1 IMPLICIT INPUTS:
1560 1 NONE
1561 1
1562 1 OUTPUT PARAMETERS:
1563 1 NONE
1564 1
1565 1 IMPLICIT OUTPUTS:
1566 1 NONE
1567 1
1568 1 ROUTINE VALUE:
1569 1 NONE
1570 1
1571 1 SIDE EFFECTS:
1572 1 NONE
1573 1
1574 1 --
1575 1
1576 2 BEGIN
1577 2 MAP
1578 2 SFM: REF BBLOCK; ! Pointer to SFM
1579 2
1580 2
1581 2 ! Deallocate the fixed/variable data areas, if they exist.
1582 2
1583 2 DEALLOCATE VARIABLE DATA(
1584 2 SFM$S FORM SETUP MODULES,
1585 2 SFM$SFM$T FORM SETUP MODULES));
1586 2 DEALLOCATE VARIABLE DATA(
1587 2 SFM$S PAGE SETUP MODULES,
1588 2 SFM$SFM$T_PAGE_SETUP_MODULES));
1589 2
1590 2
1591 2 ! Deallocate the form definition record.
1592 2
1593 2 DEALLOCATE_RECORD(.SFM_N);
1594 1 END;

```

```

0004 00000
7E      08      52 00000000G EF 9E 00002
          AC 0000015D 8F C1 00009
          62          06 DD 00012
          02 FB 00014

```

```

.ENTRY DELETE_SFM_RECORD, Save R2
MOVAB DEALLOCATE_VARIABLE_DATA, R2
ADDL3 #349, SFM, -(SP)
PUSHL #6
CALLS #2, DEALLOCATE_VARIABLE_DATA

```

```

: 1548
:
: 1585
:
:

```


RECORDUTL
V04-000

Queue record manipulation utilities

F 14

16-Sep-1984 00:18:27

14-Sep-1984 12:37:12

VAX-11 Bliss-32 V4.0-742

[JOBCTL.SRC]RECORDUTL.B32;1

Page 23
(7)

7E	08	AC	00000163	8F	C1	00017
				06	DD	00020
		62		02	FB	00022
			04	AC	DD	00025
00000000G	EF			01	FB	00028
				04	0002F	

ADDL3	#355, SFM, -(SP)
PUSHL	#6
CALLS	#2, DEALLOCATE_VARIABLE_DATA
PUSHL	SFM_N
CALLS	#1, DEALLOCATE_RECORD
RET	

: 1588
:
:
:
: 1593
:
: 1594

; Routine Size: 48 bytes, Routine Base: CODE + 025B

```
1595 1 GLOBAL ROUTINE CREATE_SMQ_RECORD(;SMQ_N,SMQ): L_OUTPUT_2=
1596 1
1597 1 ++
1598 1
1599 1 FUNCTIONAL DESCRIPTION:
1600 1 This routine allocates and initializes a queue header record.
1601 1
1602 1 INPUT PARAMETERS:
1603 1 NONE
1604 1
1605 1 IMPLICIT INPUTS:
1606 1 NONE
1607 1
1608 1 OUTPUT PARAMETERS:
1609 1 SMQ_N - Record number of SMQ.
1610 1 SMQ - Pointer to SMQ.
1611 1
1612 1 IMPLICIT OUTPUTS:
1613 1 NONE
1614 1
1615 1 ROUTINE VALUE:
1616 1 Completion status.
1617 1
1618 1 SIDE EFFECTS:
1619 1 NONE
1620 1
1621 1 --
1622 1
1623 2 BEGIN
1624 2 MAP
1625 2 SMQ: REF BBLOCK;
1626 2 LOCAL
1627 2 STATUS;
1628 2
1629 2
1630 2 ! Allocate the queue record, and return if no more.
1631 2 !
1632 2 STATUS = ALLOCATE_RECORD( ; SMQ_N, SMQ);
1633 2 IF NOT .STATUS THEN RETURN .STATUS;
1634 2
1635 2
1636 2 ! Initialize the queue header record.
1637 2 !
1638 2 SMQ[SYMSB_TYPE] = SYMSK_SMQ;
1639 2 SMQ[SMQSV_GENERIC_SELECTION] = TRUE;
1640 2 SMQ[SMQSV_JOB_SIZE_SCHEDULING] = TRUE;
1641 2 SMQ[SMQSV_PAGINATE] = TRUE;
1642 2 SMQ[SMQSV_STOPPED] = TRUE;
1643 2 SMQ[SMQSV_SWAP] = TRUE;
1644 2 CH$WCHAR(
1645 2 .VALUE_QUEUE[SDSC_W_LENGTH],
1646 2 SMQ[SMQST_NAME]);
1647 2 CH$COPY(
1648 2 .VALUE_QUEUE[SDSC_W_LENGTH],
1649 2 .VALUE_QUEUE[SDSC_A_POINTER],
1650 2 0,
1651 2 SMQSS_NAME-1, SMQ[SMQST_NAME]+1);
```

```

: 619      1652 2 SMQ[SMQ$OWNER UIC] = .JBC UIC;
: 620      1653 2 SMQ[SMQ$W_PROTECTION] = %B'T101111001111011'; ! S:E, O:D, G:R, W:W
: 621      1654 2 SMQ[SMQ$B_JOB_LIMIT] = 1;
: 622      1655 2 SMQ[SMQ$B_BASE_PRIORITY] = .SYS$GB_DEFPRI;
: 623      1656 2
: 624      1657 2
: 625      1658 2 SSS_NORMAL
: 626      1659 1 END;

```

			007C 00000	.ENTRY	CREATE SMQ RECORD, Save R2,R3,R4,R5,R6	: 1595
			EF 9E 00002	MOVAB	VALUE_QUEUE, R6	: 1632
	00000000G	EF	00 FB 00009	CALLS	#0, ALLOCATE_RECORD	: 1633
		3F	50 E9 00010	BLBC	STATUS, 1\$	: 1638
	04 AB		06 90 00013	MOVB	#6, 4(SMQ)	: 1641
	0D AB	0248	8F A8 00017	BISW2	#584, 13(SMQ)	: 1642
	0E AB	02000020	8F C8 0001D	BISL2	#33554464, 14(SMQ)	: 1646
1F	00B0 CB		66 90 00025	MOVB	VALUE_QUEUE, 176(SMQ)	: 1651
	02 B6		66 2C 0002A	MOVCS	VALUE_QUEUE, @VALUE_QUEUE+2, #0, #31, -	: 1652
		00B1 CB	00 00030		177(SMQ)	: 1653
	00D0 CB	EFB3 C6	D0 00033	MOVL	JBC UIC, 208(SMQ)	: 1654
	0104 CB	DE7B 8F	B0 0003A	MOVW	#-8581, 260(SMQ)	: 1655
	0116 CB		01 90 00041	MOVB	#1, 278(SMQ)	: 1659
	0114 CB	00000000G	00 90 00046	MOVB	SYS\$GB_DEFPRI, 276(SMQ)	
	50		01 D0 0004F	MOVL	#1, R0	
			04 00052 1\$:	RET		

; Routine Size: 83 bytes, Routine Base: CODE + 0288

```

628 1660 1 GLOBAL ROUTINE APPLY_SMQ_OPTIONS(SMQ)=
629 1661 1
630 1662 1 !++
631 1663 1
632 1664 1 FUNCTIONAL DESCRIPTION:
633 1665 1 This routine applies options to a queue header record.
634 1666 1
635 1667 1 INPUT PARAMETERS:
636 1668 1 SMQ - Pointer to SMQ.
637 1669 1
638 1670 1 IMPLICIT INPUTS:
639 1671 1 Existing values in SMQ.
640 1672 1
641 1673 1 OUTPUT PARAMETERS:
642 1674 1 NONE
643 1675 1
644 1676 1 IMPLICIT OUTPUTS:
645 1677 1 New values in SMQ.
646 1678 1
647 1679 1 ROUTINE VALUE:
648 1680 1 Completion status.
649 1681 1
650 1682 1 SIDE EFFECTS:
651 1683 1 NONE
652 1684 1
653 1685 1 --
654 1686 1
655 1687 2 BEGIN
656 1688 2 MAP
657 1689 2 SMQ: REF BBLOCK; ! Pointer to SMQ
658 1690 2 LOCAL
659 1691 2 LSMQ: BBLOCK[SYMSS_SYM], ! Saved copy of original SMQ
660 1692 2 FINAL_STATUS; ! Final routine status
661 1693 2 LABEL
662 1694 2 PROTECTED_UPDATE;
663 1695 2
664 1696 2
665 1697 2 ! Set up for backing out the transaction in case of an error.
666 1698 2
667 1699 2 CH$MOVE(SYMSS_SYM, SMQ, LSMQ);
668 1700 2 CH$FILL(0, SMQ$JOB_RESET_MODULES, SMQ[SMQ$JOB_RESET_MODULES]);
669 1701 2 SMQ[SMQ$GENERIC_TARGET] = 0;
670 1702 2 FINAL_STATUS = SS$NORMAL;
671 1703 3 PROTECTED_UPDATE: BEGIN
672 1704 3
673 1705 3
674 1706 3 IF TESTBITSS(SMQ[SMQ$V_BATCH_EXPLICIT])
675 1707 3 THEN
676 1708 4 IF (.ITEM_PRESENT[SJCS_BATCH] AND NOT .SMQ[SMQ$V_BATCH])
677 1709 4 OR (.ITEM_PRESENT[SJCS_NO_BATCH] AND .SMQ[SMQ$V_BATCH])
678 1710 3 THEN
679 1711 3 RETURN JBC$_INCQUETYP;
680 1712 3
681 1713 3
682 1714 3 IF .ITEM_PRESENT[SJCS_BASE_PRIORITY]
683 1715 3 THEN
684 1716 3 SMQ[SMQ$B_BASE_PRIORITY] = .VALUE_BASE_PRIORITY;
```

```

685 1717 3
686 1718 3
687 1719 3 IF .ITEM_PRESENT[SJCS_BATCH]
688 1720 3 THEN
689 1721 3     SMQ[SMQSV_BATCH] = TRUE;
690 1722 3
691 1723 3
692 1724 3 IF .ITEM_PRESENT[SJCS_CHARACTERISTIC_NAME]
693 1725 3 OR .ITEM_PRESENT[SJCS_CHARACTERISTIC_NUMBER]
694 1726 3 THEN
695 1727 3     CHSMOVE(
696 1728 3         SMQSS_CHARACTERISTICS,
697 1729 3         VALUE_CHARACTERISTICS,
698 1730 3         SMQ[SMQST_CHARACTERISTICS]);
699 1731 3
700 1732 3
701 1733 3 IF .ITEM_PRESENT[SJCS_NO_CHARACTERISTICS]
702 1734 3 THEN
703 1735 3     CHSFILL(
704 1736 3         0,
705 1737 3         SMQSS_CHARACTERISTICS,
706 1738 3         SMQ[SMQST_CHARACTERISTICS]);
707 1739 3
708 1740 3
709 1741 3 IF .ITEM_PRESENT[SJCS_GENERIC_SELECTION]
710 1742 3 THEN
711 1743 3     SMQ[SMQSV_GENERIC_SELECTION] = TRUE;
712 1744 3
713 1745 3
714 1746 3 IF .ITEM_PRESENT[SJCS_NO_GENERIC_SELECTION]
715 1747 3 THEN
716 1748 3     SMQ[SMQSV_GENERIC_SELECTION] = FALSE;
717 1749 3
718 1750 3
719 1751 3 IF .ITEM_PRESENT[SJCS_GENERIC_QUEUE]
720 1752 3 THEN
721 1753 3     SMQ[SMQSV_GENERIC_QUEUE] = TRUE;
722 1754 3
723 1755 3
724 1756 3 IF .ITEM_PRESENT[SJCS_NO_GENERIC_QUEUE]
725 1757 3 THEN
726 1758 3     SMQ[SMQSV_GENERIC_QUEUE] = FALSE;
727 1759 3
728 1760 3
729 1761 3 IF .ITEM_PRESENT[SJCS_GENERIC_TARGET]
730 1762 3 THEN
731 1763 4     BEGIN
732 1764 4         LOCAL
733 1765 4             P:          REF VECTOR,      ! Pointer to generic target value
734 1766 4             AUX_N,      ! Record number of auxiliary record
735 1767 4             AUX:        REF BBLOCK;      ! Pointer to auxiliary record
736 1768 4
737 1769 4
738 1770 4     FINAL_STATUS = ALLOCATE_RECORD( : AUX_N, AUX);
739 1771 4     IF NOT FINAL_STATUS THEN LEAVE PROTECTED_UPDATE;
740 1772 4     SMQ[SMQSL_GENERIC_TARGET] = .AUX_N;
741 1773 4
```

```

742 1774 4
743 1775 4 AUX[SYMSB TYPE] = SYMSK_GENERIC TARGET;
744 1776 4 VECTOR[AUX[SYMST DATA], 0] = .VALUE_GENERIC_TARGET[0];
745 1777 4 P = VALUE_GENERIC_TARGET[1] + 8 * (.VALUE_GENERIC_TARGET[0] - 1);
746 1778 4 INCR I FROM 1 TO .VALUE_GENERIC_TARGET[0] DO
747 1779 5 BEGIN
748 1780 5 LOCAL
749 1781 5 TSMQ: REF BBLOCK; ! Pointer to generic target SMQ
750 1782 5
751 1783 5 TSMQ = .P[1];
752 1784 5 IF .SMQ[SMQ$V_BATCH] NEQ .TSMQ[SMQ$V_BATCH]
753 1785 5 OR .TSMQ[SMQ$V_GENERIC_QUEUE]
754 1786 5 OR .TSMQ[SMQ$L_ASSIGNED_QUEUE_LINK] NEQ 0
755 1787 5 THEN
756 1788 6 BEGIN
757 1789 6 FINAL_STATUS = JBC$ INCDSTQUE;
758 1790 6 LEAVE_PROTECTED_UPDATE;
759 1791 5 END;
760 1792 5 VECTOR[AUX[SYMST DATA], .I] = .P[0];
761 1793 5 P = .P - 8;
762 1794 4 END;
763 1795 4 REWRITE_RECORD(.AUX_N);
764 1796 3 END;
765 1797 3
766 1798 3
767 1799 3 IF .ITEM_PRESENT[SJCS_OWNER_UIC]
768 1800 3 THEN
769 1801 3 SMQ[SMQ$L_OWNER_UIC] = .VALUE_OWNER_UIC;
770 1802 3
771 1803 3
772 1804 3 IF .ITEM_PRESENT[SJCS_PROTECTION]
773 1805 3 THEN
774 1806 3 SMQ[SMQ$W_PROTECTION] =
775 1807 3 (.SMQ[SMQ$W_PROTECTION] AND NOT .VALUE_PROTECTION<16,16>) OR
776 1808 3 (.VALUE_PROTECTION<0,16> AND .VALUE_PROTECTION<16,16>);
777 1809 3
778 1810 3
779 1811 3 IF .ITEM_PRESENT[SJCS_RETAIN_ALL_JOBS]
780 1812 3 THEN
781 1813 4 BEGIN
782 1814 4 SMQ[SMQ$V_RETAIN_ALL_JOBS] = TRUE;
783 1815 4 SMQ[SMQ$V_RETAIN_ERROR_JOBS] = FALSE;
784 1816 3 END;
785 1817 3
786 1818 3
787 1819 3 IF .ITEM_PRESENT[SJCS_RETAIN_ERROR_JOBS]
788 1820 3 THEN
789 1821 4 BEGIN
790 1822 4 SMQ[SMQ$V_RETAIN_ALL_JOBS] = FALSE;
791 1823 4 SMQ[SMQ$V_RETAIN_ERROR_JOBS] = TRUE;
792 1824 3 END;
793 1825 3
794 1826 3
795 1827 3 IF .ITEM_PRESENT[SJCS_NO_RETAIN_JOBS]
796 1828 3 THEN
797 1829 4 BEGIN
798 1830 4 SMQ[SMQ$V_RETAIN_ALL_JOBS] = FALSE;

```



```
799 1831 4      SMQ[SMQ$V_RETAIN_ERROR_JOBS] = FALSE;
800 1832 3      END;
801 1833 3
802 1834 3
803 1835 3      IF .ITEM_PRESENT[SJCS_SCSNODE_NAME]
804 1836 3      THEN
805 1837 3          COPY_SYSID(VALUE_SCSNODE_NAME, SMQ[SMQ$T_SYSID]);
806 1838 3
807 1839 3
808 1840 3      IF .SMQ[SMQ$V_BATCH]
809 1841 3      THEN
810 1842 4          BEGIN
811 1843 4              IF .ITEM_PRESENT[SJCS_CPU_DEFAULT]
812 1844 4              THEN
813 1845 5                  BEGIN
814 1846 5                      SMQ[SMQ$V_CPU_DEFAULT] = TRUE;
815 1847 5                      SMQ[SMQ$L_CPU_DEFAULT] = .VALUE_CPU_DEFAULT;
816 1848 4                  END;
817 1849 4
818 1850 4
819 1851 4              IF .ITEM_PRESENT[SJCS_NO_CPU_DEFAULT]
820 1852 4              THEN
821 1853 5                  BEGIN
822 1854 5                      SMQ[SMQ$V_CPU_DEFAULT] = FALSE;
823 1855 5                      SMQ[SMQ$L_CPU_DEFAULT] = 0;
824 1856 4                  END;
825 1857 4
826 1858 4
827 1859 4              IF .ITEM_PRESENT[SJCS_CPU_LIMIT]
828 1860 4              THEN
829 1861 5                  BEGIN
830 1862 5                      SMQ[SMQ$V_CPU_MAXIMUM] = TRUE;
831 1863 5                      SMQ[SMQ$L_CPU_MAXIMUM] = .VALUE_CPU_LIMIT;
832 1864 4                  END;
833 1865 4
834 1866 4
835 1867 4              IF .ITEM_PRESENT[SJCS_NO_CPU_LIMIT]
836 1868 4              THEN
837 1869 5                  BEGIN
838 1870 5                      SMQ[SMQ$V_CPU_MAXIMUM] = FALSE;
839 1871 5                      SMQ[SMQ$L_CPU_MAXIMUM] = 0;
840 1872 4                  END;
841 1873 4
842 1874 4
843 1875 4              IF .ITEM_PRESENT[SJCS_JOB_LIMIT]
844 1876 4              THEN
845 1877 4                  SMQ[SMQ$B_JOB_LIMIT] = .VALUE_JOB_LIMIT;
846 1878 4
847 1879 4
848 1880 4              IF .ITEM_PRESENT[SJCS_SWAP]
849 1881 4              THEN
850 1882 4                  SMQ[SMQ$V_SWAP] = TRUE;
851 1883 4
852 1884 4
853 1885 4              IF .ITEM_PRESENT[SJCS_NO_SWAP]
854 1886 4              THEN
855 1887 4                  SMQ[SMQ$V_SWAP] = FALSE;
```

```

: 856      1888  4
: 857      1889  4
: 858      1890  4      IF .ITEM_PRESENT[SJCS_WSDEFAULT]
: 859      1891  4      THEN
: 860      1892  5          BEGIN
: 861      1893  5              SMQ[SMQ$V_WSDEFAULT] = TRUE;
: 862      1894  5              SMQ[SMQ$W_WSDEFAULT] = .VALUE_WSDEFAULT;
: 863      1895  4              END;
: 864      1896  4
: 865      1897  4
: 866      1898  4      IF .ITEM_PRESENT[SJCS_NO_WSDEFAULT]
: 867      1899  4      THEN
: 868      1900  5          BEGIN
: 869      1901  5              SMQ[SMQ$V_WSDEFAULT] = FALSE;
: 870      1902  5              SMQ[SMQ$W_WSDEFAULT] = 0;
: 871      1903  4              END;
: 872      1904  4
: 873      1905  4
: 874      1906  4      IF .ITEM_PRESENT[SJCS_WSEXTENT]
: 875      1907  4      THEN
: 876      1908  5          BEGIN
: 877      1909  5              SMQ[SMQ$V_WSEXTENT] = TRUE;
: 878      1910  5              SMQ[SMQ$W_WSEXTENT] = .VALUE_WSEXTENT;
: 879      1911  4              END;
: 880      1912  4
: 881      1913  4
: 882      1914  4      IF .ITEM_PRESENT[SJCS_NO_WSEXTENT]
: 883      1915  4      THEN
: 884      1916  5          BEGIN
: 885      1917  5              SMQ[SMQ$V_WSEXTENT] = FALSE;
: 886      1918  5              SMQ[SMQ$W_WSEXTENT] = 0;
: 887      1919  4              END;
: 888      1920  4
: 889      1921  4
: 890      1922  4      IF .ITEM_PRESENT[SJCS_WSQUOTA]
: 891      1923  4      THEN
: 892      1924  5          BEGIN
: 893      1925  5              SMQ[SMQ$V_WSQUOTA] = TRUE;
: 894      1926  5              SMQ[SMQ$W_WSQUOTA] = .VALUE_WSQUOTA;
: 895      1927  4              END;
: 896      1928  4
: 897      1929  4
: 898      1930  4      IF .ITEM_PRESENT[SJCS_NO_WSQUOTA]
: 899      1931  4      THEN
: 900      1932  5          BEGIN
: 901      1933  5              SMQ[SMQ$V_WSQUOTA] = FALSE;
: 902      1934  5              SMQ[SMQ$W_WSQUOTA] = 0;
: 903      1935  4              END;
: 904      1936  4
: 905      1937  4
: 906      1938  4      IF .SMQ[SMQ$V_CPU_MAXIMUM] AND .SMQ[SMQ$V_CPU_DEFAULT]
: 907      1939  4      AND .SMQ[SMQ$[_CPU_MAXIMUM] - 1 LSSU .SMQ[SMQ$L_CPU_DEFAULT] - 1
: 908      1940  4      THEN
: 909      1941  4          SMQ[SMQ$L_CPU_DEFAULT] = .SMQ[SMQ$L_CPU_MAXIMUM];
: 910      1942  4
: 911      1943  4
: 912      1944  4      IF .SMQ[SMQ$V_WSEXTENT] AND .SMQ[SMQ$V_WSQUOTA]
```



```

913 1945 4 AND .SMQ[SMQ$W_WSEXTENT] LSSU .SMQ[SMQ$W_WSQUOTA]
914 1946 4 THEN
915 1947 4     SMQ[SMQ$W_WSQUOTA] = .SMQ[SMQ$W_WSEXTENT];
916 1948 4
917 1949 4
918 1950 4 IF .SMQ[SMQ$V_WSQUOTA] AND .SMQ[SMQ$V_WSDEFAULT]
919 1951 4 AND .SMQ[SMQ$W_WSQUOTA] LSSU .SMQ[SMQ$W_WSDEFAULT]
920 1952 4 THEN
921 1953 4     SMQ[SMQ$W_WSDEFAULT] = .SMQ[SMQ$W_WSQUOTA];
922 1954 4
923 1955 4
924 1956 4 IF .SMQ[SMQ$V_WSEXTENT] AND .SMQ[SMQ$V_WSDEFAULT]
925 1957 4 AND .SMQ[SMQ$W_WSEXTENT] LSSU .SMQ[SMQ$W_WSDEFAULT]
926 1958 4 THEN
927 1959 4     SMQ[SMQ$W_WSDEFAULT] = .SMQ[SMQ$W_WSEXTENT];
928 1960 4 END
929 1961 3 ELSE
930 1962 4 BEGIN
931 1963 4 IF .ITEM_PRESENT[SJCS_DEVICE_NAME]
932 1964 4 THEN
933 1965 5 BEGIN
934 1966 5     CH$WCHAR(
935 1967 5         .VALUE_DEVICE_NAME[SDSC_W_LENGTH],
936 1968 5         SMQ[SMQ$T_DEVICE_NAME]);
937 1969 5     CH$COPY(
938 1970 5         .VALUE_DEVICE_NAME[SDSC_W_LENGTH],
939 1971 5         .VALUE_DEVICE_NAME[SDSC_A_POINTER],
940 1972 5         0,
941 1973 5         SMQ$S_DEVICE_NAME-1,
942 1974 5         SMQ[SMQ$T_DEVICE_NAME]+1);
943 1975 4 END;
944 1976 4
945 1977 4
946 1978 4 IF .ITEM_PRESENT[SJCS_FILE_BURST]
947 1979 4 THEN
948 1980 5 BEGIN
949 1981 5     SMQ[SMQ$V_FILE_BURST] = TRUE;
950 1982 5     SMQ[SMQ$V_FILE_BURST_ONE] = FALSE;
951 1983 4 END;
952 1984 4
953 1985 4
954 1986 4 IF .ITEM_PRESENT[SJCS_FILE_BURST_ONE]
955 1987 4 THEN
956 1988 5 BEGIN
957 1989 5     SMQ[SMQ$V_FILE_BURST] = FALSE;
958 1990 5     SMQ[SMQ$V_FILE_BURST_ONE] = TRUE;
959 1991 4 END;
960 1992 4
961 1993 4
962 1994 4 IF .ITEM_PRESENT[SJCS_NO_FILE_BURST]
963 1995 4 THEN
964 1996 5 BEGIN
965 1997 5     SMQ[SMQ$V_FILE_BURST] = FALSE;
966 1998 5     SMQ[SMQ$V_FILE_BURST_ONE] = FALSE;
967 1999 4 END;
968 2000 4
969 2001 4
```

```

: 970      2002  4      IF .ITEM_PRESENT[SJCS_FILE_FLAG]
: 971      2003  4      THEN
: 972      2004  5          BEGIN
: 973      2005  5              SMQ[SMQSV_FILE_FLAG] = TRUE;
: 974      2006  5              SMQ[SMQSV_FILE_FLAG_ONE] = FALSE;
: 975      2007  4          END;
: 976      2008  4
: 977      2009  4
: 978      2010  4      IF .ITEM_PRESENT[SJCS_FILE_FLAG_ONE]
: 979      2011  4      THEN
: 980      2012  5          BEGIN
: 981      2013  5              SMQ[SMQSV_FILE_FLAG] = FALSE;
: 982      2014  5              SMQ[SMQSV_FILE_FLAG_ONE] = TRUE;
: 983      2015  4          END;
: 984      2016  4
: 985      2017  4
: 986      2018  4      IF .ITEM_PRESENT[SJCS_NO_FILE_FLAG]
: 987      2019  4      THEN
: 988      2020  5          BEGIN
: 989      2021  5              SMQ[SMQSV_FILE_FLAG] = FALSE;
: 990      2022  5              SMQ[SMQSV_FILE_FLAG_ONE] = FALSE;
: 991      2023  4          END;
: 992      2024  4
: 993      2025  4
: 994      2026  4      IF .ITEM_PRESENT[SJCS_FILE_TRAILER]
: 995      2027  4      THEN
: 996      2028  5          BEGIN
: 997      2029  5              SMQ[SMQSV_FILE_TRAILER] = TRUE;
: 998      2030  5              SMQ[SMQSV_FILE_TRAILER_ONE] = FALSE;
: 999      2031  4          END;
1000      2032  4
1001      2033  4
1002      2034  4      IF .ITEM_PRESENT[SJCS_FILE_TRAILER_ONE]
1003      2035  4      THEN
1004      2036  5          BEGIN
1005      2037  5              SMQ[SMQSV_FILE_TRAILER] = FALSE;
1006      2038  5              SMQ[SMQSV_FILE_TRAILER_ONE] = TRUE;
1007      2039  4          END;
1008      2040  4
1009      2041  4
1010      2042  4      IF .ITEM_PRESENT[SJCS_NO_FILE_TRAILER]
1011      2043  4      THEN
1012      2044  5          BEGIN
1013      2045  5              SMQ[SMQSV_FILE_TRAILER] = FALSE;
1014      2046  5              SMQ[SMQSV_FILE_TRAILER_ONE] = FALSE;
1015      2047  4          END;
1016      2048  4
1017      2049  4
1018      2050  4      IF .ITEM_PRESENT[SJCS_FORM_NAME]
1019      2051  4      OR .ITEM_PRESENT[SJCS_FORM_NUMBER]
1020      2052  4      THEN
1021      2053  4          SMQ[SMQSL_FORM_LINK] = .VALUE_FORM[0];
1022      2054  4
1023      2055  4
1024      2056  4      IF .ITEM_PRESENT[SJCS_JOB_BURST]
1025      2057  4      THEN
1026      2058  4          SMQ[SMQSV_JOB_BURST] = TRUE;
```

```
1027 2059 4
1028 2060 4
1029 2061 4 IF .ITEM_PRESENT[SJCS_NO_JOB_BURST]
1030 2062 4 THEN
1031 2063 4     SMQ[SMQSV_JOB_BURST] = FALSE;
1032 2064 4
1033 2065 4
1034 2066 4 IF .ITEM_PRESENT[SJCS_JOB_FLAG]
1035 2067 4 THEN
1036 2068 4     SMQ[SMQSV_JOB_FLAG] = TRUE;
1037 2069 4
1038 2070 4
1039 2071 4 IF .ITEM_PRESENT[SJCS_NO_JOB_FLAG]
1040 2072 4 THEN
1041 2073 4     SMQ[SMQSV_JOB_FLAG] = FALSE;
1042 2074 4
1043 2075 4
1044 2076 4 IF .ITEM_PRESENT[SJCS_JOB_RESET_MODULES]
1045 2077 4 THEN
1046 2078 5     BEGIN
1047 2079 5         FINAL STATUS = STORE_VARIABLE_DATA(
1048 2080 5             .SMQ,
1049 2081 5             SMQ$S_JOB_RESET_MODULES,
1050 2082 5             SMQ[SMQST_JOB_RESET_MODULES],
1051 2083 5             SYMSK_JOB_RESET_MODULES,
1052 2084 5             .VALUE_JOB_RESET_MODULES[SDSC_W_LENGTH],
1053 2085 5             .VALUE_JOB_RESET_MODULES[SDSC_A_POINTER]);
1054 2086 5         IF NOT .FINAL_STATUS THEN LEAVE PROTECTED_UPDATE;
1055 2087 4     END;
1056 2088 4
1057 2089 4
1058 2090 4 IF .ITEM_PRESENT[SJCS_JOB_SIZE_MAXIMUM]
1059 2091 4 THEN
1060 2092 4     SMQ[SMQSL_JOB_SIZE_MAXIMUM] = .VALUE_JOB_SIZE_MAXIMUM;
1061 2093 4
1062 2094 4
1063 2095 4 IF .ITEM_PRESENT[SJCS_NO_JOB_SIZE_MAXIMUM]
1064 2096 4 THEN
1065 2097 4     SMQ[SMQSL_JOB_SIZE_MAXIMUM] = 0;
1066 2098 4
1067 2099 4
1068 2100 4 IF .ITEM_PRESENT[SJCS_JOB_SIZE_MINIMUM]
1069 2101 4 THEN
1070 2102 4     SMQ[SMQSL_JOB_SIZE_MINIMUM] = .VALUE_JOB_SIZE_MINIMUM;
1071 2103 4
1072 2104 4
1073 2105 4 IF .ITEM_PRESENT[SJCS_NO_JOB_SIZE_MINIMUM]
1074 2106 4 THEN
1075 2107 4     SMQ[SMQSL_JOB_SIZE_MINIMUM] = 0;
1076 2108 4
1077 2109 4
1078 2110 4 IF .ITEM_PRESENT[SJCS_JOB_SIZE_SCHEDULING]
1079 2111 4 THEN
1080 2112 4     SMQ[SMQSV_JOB_SIZE_SCHEDULING] = TRUE;
1081 2113 4
1082 2114 4
1083 2115 4 IF .ITEM_PRESENT[SJCS_NO_JOB_SIZE_SCHEDULING]
```

```
1084 2116 4 THEN
1085 2117 4     SMQ[SMQSV_JOB_SIZE_SCHEDULING] = FALSE;
1086 2118 4
1087 2119 4
1088 2120 4 IF .ITEM_PRESENT[SJCS_JOB_TRAILER]
1089 2121 4 THEN
1090 2122 4     SMQ[SMQSV_JOB_TRAILER] = TRUE;
1091 2123 4
1092 2124 4
1093 2125 4 IF .ITEM_PRESENT[SJCS_NO_JOB_TRAILER]
1094 2126 4 THEN
1095 2127 4     SMQ[SMQSV_JOB_TRAILER] = FALSE;
1096 2128 4
1097 2129 4
1098 2130 4 IF .ITEM_PRESENT[SJCS_LIBRARY_SPECIFICATION]
1099 2131 4 THEN
1100 2132 5     BEGIN
1101 2133 5         CH$WCHAR(
1102 2134 5             .VALUE_LIBRARY_SPECIFICATION[SDSC_W_LENGTH],
1103 2135 5             SMQ[SMQST_LIBRARY]);
1104 2136 5         CH$COPY(
1105 2137 5             .VALUE_LIBRARY_SPECIFICATION[SDSC_W_LENGTH],
1106 2138 5             .VALUE_LIBRARY_SPECIFICATION[SDSC_A_POINTER],
1107 2139 5             0,
1108 2140 5             SMQSS_LIBRARY-1,
1109 2141 5             SMQ[SMQST_LIBRARY]+1);
1110 2142 4     END;
1111 2143 4
1112 2144 4
1113 2145 4 IF .ITEM_PRESENT[SJCS_NO_LIBRARY_SPECIFICATION]
1114 2146 4 THEN
1115 2147 4     CH$FILL(
1116 2148 4         0,
1117 2149 4         SMQSS_LIBRARY,
1118 2150 4         SMQ[SMQST_LIBRARY]);
1119 2151 4
1120 2152 4
1121 2153 4 IF .ITEM_PRESENT[SJCS_PAGINATE]
1122 2154 4 THEN
1123 2155 4     SMQ[SMQSV_PAGINATE] = TRUE;
1124 2156 4
1125 2157 4
1126 2158 4 IF .ITEM_PRESENT[SJCS_NO_PAGINATE]
1127 2159 4 THEN
1128 2160 4     SMQ[SMQSV_PAGINATE] = FALSE;
1129 2161 4
1130 2162 4
1131 2163 4 IF .ITEM_PRESENT[SJCS_PROCESSOR]
1132 2164 4 THEN
1133 2165 5     BEGIN
1134 2166 5         CH$WCHAR(
1135 2167 5             .VALUE_PROCESSOR[SDSC_W_LENGTH],
1136 2168 5             SMQ[SMQST_PROCESSOR]);
1137 2169 5         CH$COPY(
1138 2170 5             .VALUE_PROCESSOR[SDSC_W_LENGTH],
1139 2171 5             .VALUE_PROCESSOR[SDSC_A_POINTER],
1140 2172 5             0,
```

```
1141 2173 5          SMOSS_PROCESSOR-1,
1142 2174 5          SMQ[SMOST_PROCESSOR]+1);
1143 2175 4          END;
1144 2176 4
1145 2177 4
1146 2178 4          IF .ITEM_PRESENT[SJCS_NO_PROCESSOR]
1147 2179 4          THEN
1148 2180 4              CH$FILL(
1149 2181 4                  0,
1150 2182 4                  SMOSS_PROCESSOR,
1151 2183 4                  SMQ[SMOST_PROCESSOR]);
1152 2184 4
1153 2185 4
1154 2186 4          IF .ITEM_PRESENT[SJCS_TERMINAL]
1155 2187 4          THEN
1156 2188 4              SMQ[SMQSV_TERMINAL] = TRUE;
1157 2189 4
1158 2190 4
1159 2191 4          IF .ITEM_PRESENT[SJCS_NO_TERMINAL]
1160 2192 4          THEN
1161 2193 4              SMQ[SMQSV_TERMINAL] = FALSE;
1162 2194 3          END;
1163 2195 3
1164 2196 3
1165 2197 2          END;      ! block PROTECTED_UPDATE
1166 2198 2
1167 2199 2
1168 2200 2          IF .FINAL_STATUS
1169 2201 2          THEN
1170 2202 3              BEGIN
1171 2203 3                  IF .ITEM_PRESENT[SJCS_GENERIC_QUEUE]
1172 2204 3                  OR .ITEM_PRESENT[SJCS_NO_GENERIC_QUEUE]
1173 2205 3                  OR .ITEM_PRESENT[SJCS_GENERIC_TARGET]
1174 2206 3                  THEN
1175 2207 4                      BEGIN
1176 2208 4                          IF .LSMQ[SMQSL_GENERIC_TARGET] NEQ 0
1177 2209 4                          THEN
1178 2210 4                              DEALLOCATE_RECORD(.LSMQ[SMQSL_GENERIC_TARGET]);
1179 2211 4                          END
1180 2212 3                  ELSE
1181 2213 3                      SMQ[SMQSL_GENERIC_TARGET] = .LSMQ[SMQSL_GENERIC_TARGET];
1182 2214 3
1183 2215 3
1184 2216 3          IF .ITEM_PRESENT[SJCS_JOB_RESET_MODULES]
1185 2217 3          OR .ITEM_PRESENT[SJCS_NO_JOB_RESET_MODULES]
1186 2218 3          THEN
1187 2219 3              DEALLOCATE_VARIABLE_DATA(
1188 2220 3                  SMOSS_JOB_RESET_MODULES,
1189 2221 3                  LSMQ[SMOST_JOB_RESET_MODULES])
1190 2222 3          ELSE
1191 2223 3              CH$MOVE(
1192 2224 3                  SMOSS_JOB_RESET_MODULES,
1193 2225 3                  LSMQ[SMOST_JOB_RESET_MODULES],
1194 2226 3                  SMQ[SMOST_JOB_RESET_MODULES]);
1195 2227 3          END
1196 2228 2          ELSE
1197 2229 3              BEGIN
```

```
1198 2230 3 IF .SMQ[SMQSL GENERIC TARGET] NEQ 0
1199 2231 3 THEN DEALLOCATE_RECORD(.SMQ[SMQSL_GENERIC_TARGET]);
1200 2232 3 DEALLOCATE_VARIABLE_DATA(
1201 2233 3 SMQSS JOB_RESET_MODULES,
1202 2234 3 SMQ[SMQST_JOB_RESET_MODULES]);
1203 2235 3 CHSMOVE(SYMSS_SYM, LSMQ, .SMQ);
1204 2236 3 END;
1205 2237 2
1206 2238 2
1207 2239 2 .FINAL_STATUS
1208 2240 1 END;
```

			OFFC	00000	.ENTRY	APPLY_SMQ_OPTIONS, Save R2,R3,R4,R5,R6,R7,-		
		5E	FE00	CE 7E	00002	R8,R9,R10,R11	1660	
		56	04	AC D0	00007	-512(SP), SP		
06	6E	66	0200	8F 28	0000B	SMQ, R6	1699	
	00	59	0118	C6 9E	00011	#512, (R6), LSMQ		
		6E		00 2C	00016	280(R6), R9	1700	
				69	0001B	#0, (SP), #0, #6, (R9)		
			74	A6 D4	0001C	CLRL 116(R6)	1701	
		58		01 D0	0001F	MOVL #1, FINAL_STATUS	1702	
		57	0C	A6 9E	00022	MOVAB 12(R6), R7	1706	
	1D	67		01 E3	00026	BBCS #1, (R7), 3\$		
		03	00000000'	EF E9	0002A	BLBC ITEM_PRESENT+1, 1\$	1708	
		09		67 E9	00031	BLBC (R7), 2\$		
	0B	08	00000000'	EF 01	E1 00034 1\$:	BBC #1, ITEM_PRESENT+1, 3\$	1709	
		08		67 E9	0003C	BLBC (R7), 3\$		
		50	00048118	8F D0	0003F 2\$:	MOVL #295192, R0	1711	
				04	00046	RET		
			00000000'	EF 95	00047 3\$:	TSTB ITEM_PRESENT	1714	
				09 18	0004D	BGEQ 4\$		
		0114	00000000'	EF 90	0004F	MOVAB VALUE_BASE_PRIORITY, 276(R6)	1716	
		03	00000000'	EF E9	00058 4\$:	BLBC ITEM_PRESENT+1, 5\$	1719	
		67		01 88	0005F	BISB2 #1, (R7)	1721	
	08	08	00000000'	EF 04	E0 00062 5\$:	BBS #4, ITEM_PRESENT+1, 6\$	1724	
	09	09	00000000'	EF 05	E1 0006A	BBC #5, ITEM_PRESENT+1, 7\$	1725	
	A6	0A	00000000'	EF 10	28 00072 6\$:	MOVAB #16, VALUE_CHARACTERISTICS, 48(R6)	1730	
	07	07	00000000'	EF 06	E1 0007B 7\$:	BBC #6, ITEM_PRESENT+1, 8\$	1733	
10	00	6E		00 2C	00083	MOVAB #0, (SP), #0, #16, 48(R6)	1738	
			30	A6	00088			
	04	04	00000000'	EF 04	E1 0008A 8\$:	BBC #4, ITEM_PRESENT+8, 9\$	1741	
	01	A7		08 88	00092	BISB2 #8, 1(R7)	1743	
	04	04	00000000'	EF 05	E1 00096 9\$:	BBC #5, ITEM_PRESENT+8, 10\$	1746	
	01	A7		08 8A	0009E	BICB2 #8, 1(R7)	1748	
	04	04	00000000'	EF 02	E1 000A2 10\$:	BBC #2, ITEM_PRESENT+8, 11\$	1751	
	01	A7		04 88	000AA	BISB2 #4, 1(R7)	1753	
	04	04	00000000'	EF 03	E1 000AE 11\$:	BBC #3, ITEM_PRESENT+8, 12\$	1756	
	01	A7		04 8A	000B6	BICB2 #4, 1(R7)	1758	
	64	04	00000000'	EF 06	E1 000BA 12\$:	BBC #6, ITEM_PRESENT+8, 18\$	1761	
			00000000G	EF 00	FB 000C2	CALLS #0, ALLOCATE_RECORD	1770	
				58	D0 000C9	MOVL R0, FINAL_STATUS		
				03	58 E8	000CC	BLBS FINAL_STATUS, 13\$	1771

			048C	31	0G0CF	BRW	82\$			
	74	A6	5A	D0	000D2	13\$:	MOVL	AUX_N, 116(R6)	1772	
	04	AB	0E	90	000D6		MOVB	#14, 4(AUX)	1775	
		52	00000000'	EF	D0	000DA	MOVL	VALUE_GENERIC_TARGET, R2	1776	
	0C	AB	52	D0	000E1		MOVL	R2, 12(AUX)		
		53	00000000'	EF	42	7E	000E5	MOVAQ	VALUE_GENERIC_TARGET-4[R2], P	1777
				51	D4	000ED	CLRL	I	1778	
				28	11	000EF	BRB	17\$		
		50	04	A3	D0	000F1	14\$:	MOVL	4(P), TSMQ	1783
54	0C	A0		67	8D	000F5	XORB3	(R7), 12(TSMQ), R4	1784	
		0A		54	E8	000FA	BLBS	R4, 15\$		
05	0D	A0		02	E0	000FD	BBS	#2, 13(TSMQ), 15\$	1785	
			2C	A0	D5	00102	TSTL	44(TSMQ)	1786	
				0A	13	00105	BEQL	16\$		
		58	00048120	8F	D0	00107	15\$:	MOVL	#295200, FINAL_STATUS	1789
				03F7	31	0010E	BRW	76\$	1790	
	0C	AB41		63	D0	00111	16\$:	MOVL	(P), 12(AUX)[1]	1792
		53		08	C2	00116	SUBL2	#8, P	1793	
D4		51		52	F3	00119	17\$:	AOBLEQ	R2, I, 14\$	1778
				5A	DD	0011D	PUSHL	AUX_N	1795	
	00000000G	EF		01	FB	0011F	CALLS	#1, REWRITE RECORD		
		09	00000000'	EF	E9	00126	18\$:	BLBC	ITEM_PRESENT+14, 19\$	1799
	00D0	C6	00000000'	EF	D0	0012D		MOVL	VALUE_OWNER_UIC, 208(R6)	1801
29	00000000'	EF		05	E1	00136	19\$:	BBC	#5, ITEM_PRESENT+16, 20\$	1804
		51	0104	C6	3C	0013E		MOVZWL	260(R6), R1	1807
		50	00000000'	EF	3C	00143		MOVZWL	VALUE_PROTECTION+2, R0	
		51		50	CA	0014A	BICL2	R0, RT		
		50	00000000'	EF	3C	0014D		MOVZWL	VALUE_PROTECTION, R0	1808
		52	00000000'	EF	3C	00154		MOVZWL	VALUE_PROTECTION+2, R2	
		52		52	D2	0015B	MCOML	R2, R2		
		50		52	CA	0015E	BICL2	R2, R0		
0104	C6	50		51	A9	00161	BISW3	R1, R0, 260(R6)		
	08	00000000U	EF	04	E1	00167	20\$:	BBC	#4, ITEM_PRESENT+17, 21\$	1811
		02		04	88	0016F		BISB2	#4, 2(R7)	1814
		02		08	8A	00173		BICB2	#8, 2(R7)	1815
	08	00000000'	EF	05	E1	00177	21\$:	BBC	#5, ITEM_PRESENT+17, 22\$	1819
		02		04	8A	0017F		BICB2	#4, 2(R7)	1822
		02		08	88	00183		BISB2	#8, 2(R7)	1823
	04	00000000'	EF	06	E1	00187	22\$:	BBC	#6, ITEM_PRESENT+17, 23\$	1827
		02		0C	8A	0018F		BICB2	#12, 2(R7)	1831
			00000000'	EF	95	00193	23\$:	TSTB	ITEM_PRESENT+17	1835
				12	18	00199	BGEQ	24\$		
	0106	C6	00000000'	EF	D0	0019B		MOVL	VALUE_SCSNODE_NAME, 262(R6)	1837
	010A	C6	00000000'	EF	B0	001A4		MOVW	VALUE_SCSNODE_NAME+4, 266(R6)	
		03		67	E8	001AD	24\$:	BLBS	(R7), 25\$	1840
				0142	31	001B0		BRW	43\$	
	08	00000000'	EF	03	E1	001B3	25\$:	BBC	#3, ITEM_PRESENT+2, 26\$	1843
		67		04	88	001BB		BISB2	#4, (R7)	1846
		40	00000000'	EF	D0	001BE		MOVL	VALUE_CPU_DEFAULT, 64(R6)	1847
06	00000000'	EF		04	E1	001C6	26\$:	BBC	#4, ITEM_PRESENT+2, 27\$	1851
		67		04	8A	001CE		BICB2	#4, (R7)	1854
			40	A6	D4	001D1		CLRL	64(R6)	1855
	08	00000000'	EF	05	E1	001D4	27\$:	BBC	#5, ITEM_PRESENT+2, 28\$	1859
		67		08	88	001DC		BISB2	#8, (R7)	1862
		44	00000000'	EF	D0	001DF		MOVL	VALUE_CPU_LIMIT, 68(R6)	1863
06	00000000'	EF		06	E1	001E7	28\$:	BBC	#6, ITEM_PRESENT+2, 29\$	1867
		67		08	8A	001EF		BICB2	#8, (R7)	1870

1F

09 00000000'	EF	44	A6	D4 001F2	CLRL	68(R6)	1871
0116	C6	00000000'	06	E1 001F5	BBC	#6, ITEM PRESENT+9, 30\$	1875
04 00000000'	EF		01	E1 001FD	MOVB	VALUE JOB LIMIT, 278(R6)	1877
02	A7		20	E1 00206	BBC	#1, ITEM PRESENT+18, 31\$	1880
04 00000000'	EF		02	E1 0020E	BISB2	#32, 2(R7)	1882
02	A7		20	E1 00212	BBC	#2, ITEM PRESENT+18, 32\$	1885
		00000000'	02	8A 0021A	BICB2	#32, 2(R7)	1887
			0E	95 0021E	TSTB	ITEM_PRESENT+18	1890
02	A7	80	8F	88 00224	BGEQ	33\$	1893
010E	C6	00000000'	EF	88 00226	BISB2	#128, 2(R7)	1894
02	09	00000000'	EF	B0 0022B	MOVW	VALUE WSDEFAULT, 270(R6)	1898
	A7	80	8F	E9 00234	BLBC	ITEM PRESENT+19, 34\$	1901
		010E	C6	8A 0023B	BICB2	#128, 2(R7)	1902
0D 00000000'	EF		01	B4 00240	CLRW	270(R6)	1906
03	A7		01	E1 00244	BBC	#1, ITEM PRESENT+19, 35\$	1909
0110	C6	00000000'	01	88 0024C	BISB2	#1, 3(R7)	1910
08 00000000'	EF		02	B0 00250	MOVW	VALUE WSEXTENT, 272(R6)	1914
03	A7		02	E1 00259	BBC	#2, ITEM PRESENT+19, 36\$	1917
		0110	01	8A 00261	BICB2	#1, 3(R7)	1918
0D 00000000'	EF		C6	B4 00265	CLRW	272(R6)	1922
03	A7		03	E1 00269	BBC	#3, ITEM PRESENT+19, 37\$	1925
0112	C6	00000000'	02	88 00271	BISB2	#2, 3(R7)	1926
08 00000000'	EF		EF	B0 00275	MOVW	VALUE WSQUOTA, 274(R6)	1930
03	A7		04	E1 0027E	BBC	#4, ITEM PRESENT+19, 38\$	1933
		0112	02	8A 00286	BICB2	#2, 3(R7)	1934
18	67		C6	B4 0028A	CLRW	274(R6)	1938
14	67		03	E1 0028E	BBC	#3, (R7), 39\$	1939
51	44		02	E1 00292	BBC	#2, (R7), 39\$	
50	40		01	C3 00296	SUBL3	#1, 68(R6), R1	
			01	C3 00298	SUBL3	#1, 64(R6), R0	
			51	D1 002A0	CMPL	R1, R0	
			05	1E 002A3	BGEQU	39\$	
	40	44	A6	D0 002A5	MOVL	68(R6), 64(R6)	1941
28	14	03	A7	E9 002AA	BLBC	3(R7), 40\$	1944
	67		19	E1 002AE	BBC	#25, (R7), 41\$	
0112	C6	0110	C6	B1 002B2	CMPL	272(R6), 274(R6)	1945
			07	1E 002B9	BGEQU	40\$	
0112	C6	0110	C6	B0 002BB	MOVW	272(R6), 274(R6)	1947
14	67		19	E1 002C2	BBC	#25, (R7), 41\$	1950
10	67		17	E1 002C6	BBC	#23, (R7), 41\$	
010E	C6	0112	C6	B1 002CA	CMPL	274(R6), 270(R6)	1951
			07	1E 002D1	BGEQU	41\$	
010E	C6	0112	C6	B0 002D3	MOVW	274(R6), 270(R6)	1953
	14	03	A7	E9 002DA	BLBC	3(R7), 42\$	1956
10	67		17	E1 002DE	BBC	#23, (R7), 42\$	
010E	C6	0110	C6	B1 002E2	CMPL	272(R6), 270(R6)	1957
			07	1E 002E9	BGEQU	42\$	
010E	C6	0110	C6	B0 002EB	MOVW	272(R6), 270(R6)	1959
			0213	31 002F2	BRW	76\$	1840
17 00000000'	EF		03	E1 002F5	BBC	#3, ITEM PRESENT+3, 44\$	1963
50	A6	00000000'	EF	90 002FD	MOVB	VALUE_DEVICE_NAME, 80(R6)	1968
00 00000000'	FF	00000000'	EF	2C 00305	MOVCS	VALUE_DEVICE_NAME, @VALUE_DEVICE_NAME+2, -	1974
		51	A6	00312		#0, #31, 81(R6)	
	06	00000000'	EF	E9 00314	BLBC	ITEM PRESENT+4, 45\$	1978
	67		10	88 0031B	BISB2	#16, (R7)	1981
	67		20	8A 0031E	BICB2	#32, (R7)	1982
06 00000000'	EF		01	E1 00321	BBC	#1, ITEM_PRESENT+4, 46\$	1986

03 00000000'	67	10	8A 00329	BICB2	#16, (R7)	1989
	67	20	88 0032C	BISB2	#32, (R7)	1990
	EF	02	E1 0032F 46\$:	BBC	#2, ITEM PRESENT+4, 47\$	1994
08 00000000'	67	30	8A 00337	BICB2	#48, (R7)	1998
	EF	04	E1 0033A 47\$:	BBC	#4, ITEM PRESENT+4, 48\$	2002
	67	40	8F 88 00342	BISB2	#64, (R7)	2005
	67	80	8F 8A 00346	BICB2	#128, (R7)	2006
08 00000000'	EF	05	E1 0034A 48\$:	BBC	#5, ITEM PRESENT+4, 49\$	2010
	67	40	8F 8A 00352	BICB2	#64, (R7)	2013
	67	80	8F 88 00356	BISB2	#128, (R7)	2014
04 00000000'	EF	06	E1 0035A 49\$:	BBC	#6, ITEM PRESENT+4, 50\$	2018
	67	C0	8F 8A 00362	BICB2	#192, (R7)	2022
08 00000000'	EF	03	E1 00366 50\$:	BBC	#3, ITEM PRESENT+5, 51\$	2026
	01	01	88 0036E	BICB2	#1, 1(R7)	2029
	01	02	8A 00372	BICB2	#2, 1(R7)	2030
08 00000000'	EF	04	E1 00376 51\$:	BBC	#4, ITEM PRESENT+5, 52\$	2034
	01	01	8A 0037E	BICB2	#1, 1(R7)	2037
	01	02	88 00382	BISB2	#2, 1(R7)	2038
04 00000000'	EF	05	E1 00386 52\$:	BBC	#5, ITEM PRESENT+5, 53\$	2042
	01	03	8A 0038E	BICB2	#3, 1(R7)	2046
08 00000000'	EF	06	E0 00392 53\$:	BBS	#6, ITEM PRESENT+6, 54\$	2050
	00000000'	EF	95 0039A	TSTB	ITEM_PRESENT+6	2051
		08	18 003A0	BGEQ	55\$	
04 00000000'	70	EF	D0 003A2 54\$:	MOVL	VALUE FORM, 112(R6)	2053
	01	01	E1 003AA 55\$:	BBC	#1, ITEM PRESENT+9, 56\$	2056
	01	10	88 003B2	BISB2	#16, 1(R7)	2058
04 00000000'	EF	02	E1 003B6 56\$:	BBC	#2, ITEM PRESENT+9, 57\$	2061
	01	10	8A 003BE	BICB2	#16, 1(R7)	2063
04 00000000'	EF	04	E1 003C2 57\$:	BBC	#4, ITEM PRESENT+9, 58\$	2066
	01	20	88 003CA	BISB2	#32, 1(R7)	2068
04 00000000'	EF	05	E1 003CE 58\$:	BBC	#5, ITEM PRESENT+9, 59\$	2071
	01	20	8A 003D6	BICB2	#32, 1(R7)	2073
	25	EF	E9 003DA 59\$:	BLBC	ITEM PRESENT+10, 60\$	2076
	00000000'	EF	DD 003E1	PUSHL	VALUE_JOB_RESET_MODULES+2	2085
	7E	EF	3C 003E7	MOVZWL	VALUE_JOB_RESET_MODULES, -(SP)	2084
		0F	DD 003EE	PUSHL	#15	2082
		59	DD 003F0	PUSHL	R9	
		06	DD 003F2	PUSHL	#6	
		56	DD 003F4	PUSHL	R6	
00000000G	EF	06	FB 003F6	CALLS	#6, STORE_VARIABLE_DATA	
	58	50	D0 003FD	MOVL	R0, FINAL_STATUS	
	03	58	E8 00400	BLBS	FINAL_STATUS, 60\$	2086
		0158	31 00403	BRW	82\$	
09 00000000'	EF	02	E1 00406 60\$:	BBC	#2, ITEM PRESENT+10, 61\$	2090
	0080	EF	D0 0040E	MOVL	VALUE_JOB_SIZE_MAXIMUM, 128(R6)	2092
04 00000000'	EF	03	E1 00417 61\$:	BBC	#3, ITEM PRESENT+10, 62\$	2095
	0080	C6	D4 0041F	CLRL	128(R6)	2097
09 00000000'	EF	04	E1 00423 62\$:	BBC	#4, ITEM PRESENT+10, 63\$	2100
	0084	EF	D0 0042B	MOVL	VALUE_JOB_SIZE_MINIMUM, 132(R6)	2102
04 00000000'	EF	05	E1 00434 63\$:	BBC	#5, ITEM PRESENT+10, 64\$	2105
	0084	C6	D4 0043C	CLRL	132(R6)	2107
05 00000000'	EF	06	E1 00440 64\$:	BBC	#6, ITEM PRESENT+10, 65\$	2110
	01	8F	88 00448	BISB2	#64, 1(R7)	2112
	00000000'	EF	95 0044D 65\$:	TSTB	ITEM_PRESENT+10	2115
		05	18 00453	BGEQ	66\$	
	01	8F	8A 00455	BICB2	#64, 1(R7)	2117
05 00000000'	EF	01	E1 0045A 66\$:	BBC	#1, ITEM_PRESENT+11, 67\$	2120

	01	A7	80	8F	88	00462	BISB2	#128, 1(R7)	2122
	05 00000000'	EF		02	E1	00467 67\$:	BBC	#2, ITEM_PRESENT+11, 68\$	2125
	01	A7	80	8F	8A	0046F	BICB2	#128, 1(R7)	2127
	19 00000000'	EF		05	E1	00474 68\$:	BBC	#5, ITEM_PRESENT+11, 69\$	2130
	0088	C6	00000000'	EF	90	0047C	MOVB	VALUE_LIBRARY_SPECIFICATION, 136(R6)	2135
27	00 00000000'	FF	00000000'	EF	2C	00485	MOVCS	VALUE_LIBRARY_SPECIFICATION, -	2141
			0089	C6		00492		@VALUE_LIBRARY_SPECIFICATION+2, #0, #39, -	
								137(R6)	
	08 00000000'	EF		06	E1	00495 69\$:	BBC	#6, ITEM_PRESENT+11, 70\$	2145
28	00	6E		00	2C	0049D	MOVCS	#0, (SP), #0, #40, 136(R6)	2150
			0088	C6		004A2			
	04 00000000'	EF		05	E1	004A5 70\$:	BBC	#5, ITEM_PRESENT+14, 71\$	2153
	02	A7		02	88	004AD	BISB2	#2, 2(R7)	2155
	04 00000000'	EF		06	E1	004B1 71\$:	BBC	#6, ITEM_PRESENT+14, 72\$	2158
	02	A7		02	8A	004B9	BICB2	#2, 2(R7)	2160
	19 00000000'	EF		03	E1	004BD 72\$:	BBC	#3, ITEM_PRESENT+16, 73\$	2163
	00D4	C6	00000000'	EF	90	004C5	MOVB	VALUE_PROCESSOR, 212(R6)	2168
27	00 00000000'	FF	00000000'	EF	2C	004CE	MOVCS	VALUE_PROCESSOR, @VALUE_PROCESSOR+2, #0, -	2174
			00D5	C6		004DB		#39, 213(R6)	
	08 00000000'	EF		04	E1	004DE 73\$:	BBC	#4, ITEM_PRESENT+16, 74\$	2178
28	00	6E		00	2C	004E6	MOVCS	#0, (SP), #0, #40, 212(R6)	2183
			00D4	C6		004EB			
	05 00000000'	EF		03	E1	004EE 74\$:	BBC	#3, ITEM_PRESENT+18, 75\$	2186
	02	A7	40	8F	88	004F6	BISB2	#64, 2(R7)	2188
	05 00000000'	EF		04	E1	004FB 75\$:	BBC	#4, ITEM_PRESENT+18, 76\$	2191
	02	A7	40	8F	8A	00503	BICB2	#64, 2(R7)	2193
		53	74	58	E9	00508 76\$:	BLBC	FINAL_STATUS, 82\$	2200
		50		AE	D0	0050B	MOVL	LSMQ+T16, R0	2208
	10 00000000'	EF		02	E0	0050F	BBS	#2, ITEM_PRESENT+8, 77\$	2203
	08 00000000'	EF		03	E0	00517	BBS	#3, ITEM_PRESENT+8, 77\$	2204
	0D 00000000'	EF		06	E1	0051F	BBC	#6, ITEM_PRESENT+8, 78\$	2205
				0F	13	00527 77\$:	BEQL	79\$	2208
				50	DD	00529	PUSHL	R0	2210
	00000000G	EF		01	FB	0052B	CALLS	#1, DEALLOCATE_RECORD	
				04	11	00532	BRB	79\$	2203
	74	A6		50	D0	00534 78\$:	MOVL	R0, 116(R6)	2213
		08	00000000'	EF	E8	00538 79\$:	BLBS	ITEM_PRESENT+10, 80\$	2216
	0F 00000000'	EF		01	E1	0053F	BBC	#1, ITEM_PRESENT+10, 81\$	2217
			FF18	CD	9F	00547 80\$:	PUSHAB	LSMQ+280	2221
				06	DD	0054B	PUSHL	#6	
	00000000G	EF		02	FB	0054D	CALLS	#2, DEALLOCATE_VARIABLE_DATA	
				28	11	00554	BRB	84\$	
	69	FF18	CD	06	28	00556 81\$:	MOVCS	#6, LSMQ+280, (R9)	2226
				20	11	0055C	BRB	84\$	2200
			74	A6	D5	0055E 82\$:	TSTL	116(R6)	2230
				0A	13	00561	BEQL	83\$	
			74	A6	DD	00563	PUSHL	116(R6)	2231
	00000000G	EF		01	FB	00566	CALLS	#1, DEALLOCATE_RECORD	
				59	DD	0056D 83\$:	PUSHL	R9	2234
				06	DD	0056F	PUSHL	#6	
	00000000G	EF		02	FB	00571	CALLS	#2, DEALLOCATE_VARIABLE_DATA	
	66	6E	0200	8F	28	00578	MOVCS	#512, LSMQ, (R6)	2235
		50		58	D0	0057E 84\$:	MOVL	FINAL_STATUS, R0	2240
				04	00581		RET		

RECORDUTL
V04-000

Queue record manipulation utilities

K 15
16-Sep-1984 00:18:27
14-Sep-1984 12:37:12

VAX-11 Bliss-32 V4.0-742
[JOBCTL.SRC]RECORDUTL.B32;1

Page 41
(9)

```
: 1210 2241 1 GLOBAL ROUTINE DELETE_SMQ_RECORD(SMQ_N,SMQ): NOVALUE=
: 1211 2242 1
: 1212 2243 1 ++
: 1213 2244 1
: 1214 2245 1 FUNCTIONAL DESCRIPTION:
: 1215 2246 1 This routine deallocates a queue header and its substructures.
: 1216 2247 1
: 1217 2248 1 INPUT PARAMETERS:
: 1218 2249 1 SMQ_N - Record number of SMQ.
: 1219 2250 1 SMQ - Pointer to SMQ.
: 1220 2251 1
: 1221 2252 1 IMPLICIT INPUTS:
: 1222 2253 1 NONE
: 1223 2254 1
: 1224 2255 1 OUTPUT PARAMETERS:
: 1225 2256 1 NONE
: 1226 2257 1
: 1227 2258 1 IMPLICIT OUTPUTS:
: 1228 2259 1 NONE
: 1229 2260 1
: 1230 2261 1 ROUTINE VALUE:
: 1231 2262 1 NONE
: 1232 2263 1
: 1233 2264 1 SIDE EFFECTS:
: 1234 2265 1 NONE
: 1235 2266 1
: 1236 2267 1 --
: 1237 2268 1
: 1238 2269 2 BEGIN
: 1239 2270 2 MAP
: 1240 2271 2 SMQ: REF BBLOCK; ! Pointer to SMQ
: 1241 2272 2
: 1242 2273 2
: 1243 2274 2 ! If there are services outstanding for this queue, cancel them.
: 1244 2275 2
: 1245 2276 2 IF .SMQ[SMQ$V_PAUSING]
: 1246 2277 2 OR .SMQ[SMQ$V_RESETTING]
: 1247 2278 2 OR .SMQ[SMQ$V_RESUMING]
: 1248 2279 2 OR .SMQ[SMQ$V_STARTING]
: 1249 2280 2 OR .SMQ[SMQ$V_STOPPING]
: 1250 2281 2 THEN
: 1251 2282 2 SCAN_INCOMPLETE_SERVICES(ISRV_K_PURGE_SMQ, .SMQ_N);
: 1252 2283 2
: 1253 2284 2
: 1254 2285 2 ! Deallocate the generic target record, if it exists.
: 1255 2286 2
: 1256 2287 2 IF .SMQ[SMQ$L_GENERIC_TARGET] NEQ 0
: 1257 2288 2 THEN
: 1258 2289 2 DEALLOCATE_RECORD(.SMQ[SMQ$L_GENERIC_TARGET]);
: 1259 2290 2
: 1260 2291 2
: 1261 2292 2 ! Deallocate the fixed/variable data areas, if they exist.
: 1262 2293 2
: 1263 2294 2 DEALLOCATE_VARIABLE_DATA(
: 1264 2295 2 SMQ$S_JOB_RESET_MODULES,
: 1265 2296 2 SMQ[SMQ$T_JOB_RESET_MODULES]);
: 1266 2297 2
```

```

: 1267
: 1268
: 1269
: 1270
: 1271

2298 2
2299 2 : Finally, deallocate the queue header record.
2300 2
2301 2 DEALLOCATE_RECORD(.SMQ_N);
2302 1 END;

```

			000C 00000	.ENTRY DELETE_SMQ_RECORD, Save R2,R3	: 2241
	53	00000000G	EF 9E 00002	MOVAB DEALLOCATE_RECORD, R3	
	52	08	AC D0 00009	MOVL SMQ, R2	: 2276
	50	10	A2 9E 0000D	MOVAB 16(R2), R0	
10	60		03 E0 00011	BBS #3, (R0), 1\$	
OC	60		05 E0 00015	BBS #5, (R0), 1\$	: 2277
08	60		06 E0 00019	BBS #6, (R0), 1\$	: 2278
	04	01	A0 E8 0001D	BLBS 1(R0), 1\$	: 2279
OC	60		0A E1 00021	BBC #10, (R0), 2\$	: 2280
		04	AC DD 00025 1\$:	PUSHL SMQ_N	: 2282
		04	DD 00028	PUSHL #4	
00000000G	EF		02 FB 0002A	CALLS #2, SCAN_INCOMPLETE_SERVICES	
		74	A2 D5 00031 2\$:	TSTL 116(R2)	: 2287
			06 13 00034	BEQL 3\$	
		74	A2 DD 00036	PUSHL 116(R2)	: 2289
	63		01 FB 00039	CALLS #1, DEALLOCATE_RECORD	
		0118	C2 9F 0003C 3\$:	PUSHAB 280(R2)	: 2296
			06 DD 00040	PUSHL #6	
00000000G	EF		02 FB 00042	CALLS #2, DEALLOCATE_VARIABLE_DATA	
		04	AC DD 00049	PUSHL SMQ_N	: 2301
	63		01 FB 0004C	CALLS #1, DEALLOCATE_RECORD	
			04 0004F	RET	: 2302

; Routine Size: 80 bytes, Routine Base: CODE + 0860

```
1273 2303 1 GLOBAL ROUTINE CREATE_SJH_RECORD(SMQ_N,SMQ;SJH_N,SJH): L_OUTPUT_2=
1274 2304 1
1275 2305 1 ++
1276 2306 1
1277 2307 1 FUNCTIONAL DESCRIPTION:
1278 2308 1 This routine allocates and initializes a job header record.
1279 2309 1
1280 2310 1 INPUT PARAMETERS:
1281 2311 1 SMQ_N - Record number of SMQ.
1282 2312 1 SMQ - Pointer to SMQ.
1283 2313 1
1284 2314 1 IMPLICIT INPUTS:
1285 2315 1 MBX - Pointer to buffered mailbox message.
1286 2316 1
1287 2317 1 OUTPUT PARAMETERS:
1288 2318 1 SJH_N - Record number of SJH.
1289 2319 1 SJH - Pointer to SJH.
1290 2320 1
1291 2321 1 IMPLICIT OUTPUTS:
1292 2322 1 NONE
1293 2323 1
1294 2324 1 ROUTINE VALUE:
1295 2325 1 Completion status.
1296 2326 1
1297 2327 1 SIDE EFFECTS:
1298 2328 1 NONE
1299 2329 1
1300 2330 1 --
1301 2331 1
1302 2332 2 BEGIN
1303 2333 2 MAP
1304 2334 2 SMQ: REF BBLOCK,
1305 2335 2 SJH: REF BBLOCK;
1306 2336 2 LOCAL
1307 2337 2 STATUS;
1308 2338 2
1309 2339 2
1310 2340 2 ! Allocate the queue record, and return if no more.
1311 2341 2
1312 2342 2 STATUS = ALLOCATE_RECORD( ; SJH_N, SJH);
1313 2343 2 IF NOT .STATUS THEN RETURN .STATUS;
1314 2344 2
1315 2345 2
1316 2346 2 ! Initialize the queue record.
1317 2347 2
1318 2348 2 SJH[SYMSB_TYPE] = SYMSK_SJH;
1319 2349 2 STATUS = ALLOCATE_ENTRY_NUMBER(SJH[SYMSL_ENTRY_NUMBER]);
1320 2350 2 IF NOT .STATUS
1321 2351 2 THEN
1322 2352 3 BEGIN
1323 2353 3 DEALLOCATE_RECORD(.SJH_N);
1324 2354 3 RETURN .STATUS;
1325 2355 2 END;
1326 2356 2
1327 2357 2
1328 2358 2 IF .SMQ[SMQ$V_BATCH]
1329 2359 2 THEN
```



```
1330 2360 3 BEGIN
1331 2361 3 SJH[SJHSV_LOG_DELETE] = TRUE;
1332 2362 3 SJH[SJHSV_LOG_SPOOL] = TRUE;
1333 2363 3 END
1334 2364 2 ELSE
1335 2365 2 BEGIN
1336 2366 2 SJH[SJHSV_RESTART] = TRUE;
1337 2367 2 SJH[SJH$B_JOB_COPIES] = 1;
1338 2368 2 END;
1339 2369 2
1340 2370 2
1341 2371 2 SJH[SJHSV_OPEN] = TRUE;
1342 2372 2 SJH[SJH$L_QUEUE_LINK] = .SMQ_N;
1343 2373 2 SJH[SJH$B_PRIORITY] = .SYS$GB_DEFQUEPRI;
1344 2374 2 IF SJH[SJH$B_PRIORITY] EQL 0 THEN SJH[SJH$B_PRIORITY] = .MBX[ACM$B_PROCPRI];
1345 2375 2 SJH[SJH$L_PID] = .MBX[ACM$L_PID];
1346 2376 2 COPY SYSID(THIS SYSID, SJH[SJH$T_SYSID]);
1347 2377 2 SJH[SJH$L_UIC] = .MBX[ACM$L_UIC];
1348 2378 2 CH$MOVE(SJH$S_USERNAME, MBX[ACM$T_USERNAME], SJH[SJH$T_USERNAME]);
1349 2379 2 CH$MOVE(SJH$S_ACCOUNT, MBX[ACM$T_ACCOUNT], SJH[SJH$T_ACCOUNT]);
1350 2380 2 COPY TIME(CUR_TIME, SJH[SJH$Q_TIME]);
1351 2381 2 COPY TIME(CUR_TIME, SJH[SJH$Q_AFTER_TIME]);
1352 2382 2 SS$ NORMAL
1353 2383 1 END;
```

			00FC 00000	.ENTRY	CREATE SJH_RECORD, Save R2,R3,R4,R5,R6,R7	2303
			EF 9E 00032	MOVAB	MBX, R7	
00000000G	EF		00 FB 00009	CALLS	#0, ALLOCATE_RECORD	2342
	52		50 D0 00010	MOVL	R0, STATUS	
	1D		52 E9 00013	BLBC	STATUS, 1\$	2345
04	AB		07 90 00016	MOVB	#7, 4(SJH)	2348
		08	AB 9F 0001A	PUSHAB	8(SJH)	2349
00000000G	EF		01 FB 0001D	CALLS	#1, ALLOCATE_ENTRY_NUMBER	
	52		50 D0 00024	MOVL	R0, STATUS	
	0D		52 E8 00027	BLBS	STATUS, 2\$	2350
			5A DD 0002A	PUSHL	SJH_N	2353
00000000G	EF		01 FB 0002C	CALLS	#1, DEALLOCATE_RECORD	
	50		52 D0 00033 1\$:	MOVL	STATUS, R0	2354
			04 00036	RET		
	50	08	AC D0 00037 2\$:	MOVL	SMQ, R0	2358
	06	0C	A0 E9 0003B	BLBC	12(R0), 3\$	
0D	AB		14 88 0003F	BISB2	#20, 13(SJH)	2362
			09 11 00043	BRB	4\$	2358
0E	AB		02 88 00045 3\$:	BISB2	#2, 14(SJH)	2366
017A	CB		01 90 00049	MOVB	#1, 378(SJH)	2367
10	AB	40	8F 88 0004E 4\$:	BISB2	#64, 16(SJH)	2371
0134	CB	04	AC D0 00053	MOVL	SMQ_N, 308(SJH)	2372
	51	017D	CB 9E 00059	MOVAB	381(SJH), R1	2373
	61	00000000G	07 90 0005E	MOVB	SYS\$GB_DEFQUEPRI, (R1)	
			07 12 00065	BNEQ	5\$	2374
	50		67 D0 00067	MOVL	MBX, R0	
	61	24	A0 90 0006A	MOVB	36(R0), (R1)	
	56		67 D0 0006E 5\$:	MOVL	MBX, R6	2375

RECORDUTL
V04-007

Queue record manipulation utilities

C 16
16-Sep-1984 00:18:27 VAX-11 Bliss-32 V4.0-742
14-Sep-1984 12:37:12 [JOBCTL.SRC]RECORDUTL.B32;1

Page 46
(11)

	0130	CB	28	A6	D0	00071	MOVL	40(R6), 304(SJH)	:	
	016C	CB	CB	A7	D0	00077	MOVL	THIS-SYSID, 364(SJH)	:	2376
	0170	CB	CC	A7	B0	0007D	MOVW	THIS-SYSID+4, 368(SJH)	:	
	0144	CB	0C	A6	D0	00083	MOVL	12(R6), 324(SJH)	:	2377
0148	10	A6		0C	28	00089	MOVC3	#12, 16(R6), 328(SJH)	:	2378
14	1C	A6		08	28	00090	MOVC3	#8, 28(R6), 20(SJH)	:	2379
	013C	CB	D0	A7	7D	00096	MOVQ	CUR_TIME, 316(SJH)	:	2380
	0098	CB	D0	A7	7D	0009C	MOVQ	CUR_TIME, 152(SJH)	:	2381
		50		01	D0	000A2	MOVL	#1, R0	:	2383
				04	D0	000A5	RET		:	

; Routine Size: 166 bytes, Routine Base: CODE + 0880


```
1355 2384 1 GLOBAL ROUTINE APPLY_SJH_OPTIONS(SJH)=
1356 2385 1
1357 2386 1 **
1358 2387 1
1359 2388 1 FUNCTIONAL DESCRIPTION:
1360 2389 1 This routine applies options to a job header record.
1361 2390 1
1362 2391 1 INPUT PARAMETERS:
1363 2392 1 SJH - Pointer to SJH.
1364 2393 1
1365 2394 1 IMP_LICIT INPUTS:
1366 2395 1 Existing values in SJH.
1367 2396 1
1368 2397 1 OUTPUT PARAMETERS:
1369 2398 1 NONE
1370 2399 1
1371 2400 1 IMPLICIT OUTPUTS:
1372 2401 1 New values in SJH.
1373 2402 1
1374 2403 1 ROUTINE VALUE:
1375 2404 1 Completion status.
1376 2405 1
1377 2406 1 SIDE EFFECTS:
1378 2407 1 NONE
1379 2408 1
1380 2409 1 --
1381 2410 1
1382 2411 2 BEGIN
1383 2412 2 MAP
1384 2413 2 SJH: REF BBLOCK;
1385 2414 2 LOCAL
1386 2415 2 LSJH: BBLOCK[SYMSS_SYM], ! Saved copy of original SJH
1387 2416 2 FINAL_STATUS; ! Final routine status
1388 2417 2 LABEL
1389 2418 2 PROTECTED_UPDATE;
1390 2419 2
1391 2420 2
1392 2421 2 ! Set up for backing out the transaction in case of an error.
1393 2422 2
1394 2423 2 CHSMOVE(SYMSS_SYM, SJH, LSJH);
1395 2424 2 CHSFILL(0, SJHSS_CHECKPOINT, SJH[SJHST_CHECKPOINT]);
1396 2425 2 CHSFILL(0, SJHSS_LOG_SPECIFICATION, SJH[SJHST_LOG_SPECIFICATION]);
1397 2426 2 CHSFILL(0, SJHSS_NOTE, SJH[SJHST_NOTE]);
1398 2427 2 CHSFILL(0, SJHSS_PARAMETERS, SJH[SJHST_PARAMETERS]);
1399 2428 2 CHSFILL(0, SJHSS_OPERATOR_REQUEST, SJH[SJHST_OPERATOR_REQUEST]);
1400 2429 2 FINAL_STATUS = SS$NORMAL;
1401 2430 3 PROTECTED_UPDATE: BEGIN
1402 2431 3
1403 2432 3
1404 2433 3 IF .ITEM_PRESENT[SJCS_AFTER_TIME]
1405 2434 3 THEN
1406 2435 4 BEGIN
1407 2436 4 COPY_TIME(VALUE_AFTER_TIME, SJH[SJH$Q_AFTER_TIME]);
1408 2437 5 IF TIME_GEQU(CUR_TIME, SJH[SJH$Q_AFTER_TIME])
1409 2438 4 THEN
1410 2439 4 COPY_TIME(CUR_TIME, SJH[SJH$Q_AFTER_TIME]);
1411 2440 3 END;
```

```
1412 2441 3
1413 2442 3
1414 2443 3 IF .ITEM_PRESENT[SJCS_NO_AFTER_TIME]
1415 2444 3 THEN
1416 2445 3     COPY_TIME(CUR_TIME, SJH[SJH$Q_AFTER_TIME]);
1417 2446 3
1418 2447 3
1419 2448 3 IF .ITEM_PRESENT[SJCS_CHARACTERISTIC_NAME]
1420 2449 3 OR .ITEM_PRESENT[SJCS_CHARACTERISTIC_NUMBER]
1421 2450 3 THEN
1422 2451 3     CH$MOVE(
1423 2452 3         SJH$S_CHARACTERISTICS,
1424 2453 3         VALUE_CHARACTERISTICS,
1425 2454 3         SJH[SJH$T_CHARACTERISTICS]);
1426 2455 3
1427 2456 3
1428 2457 3 IF .ITEM_PRESENT[SJCS_NO_CHARACTERISTICS]
1429 2458 3 THEN
1430 2459 3     CH$FILL(
1431 2460 3         0,
1432 2461 3         SJH$S_CHARACTERISTICS,
1433 2462 3         SJH[SJH$T_CHARACTERISTICS]);
1434 2463 3
1435 2464 3
1436 2465 3 IF .ITEM_PRESENT[SJCS_CHECKPOINT_DATA]
1437 2466 3 THEN
1438 2467 3     BEGIN
1439 2468 3         FINAL_STATUS = STORE_VARIABLE_DATA(
1440 2469 3             .SJH,
1441 2470 3             SJH$S_CHECKPOINT,
1442 2471 3             SJH[SJH$T_CHECKPOINT],
1443 2472 3             SYMSK_CHECKPOINT,
1444 2473 3             .VALUE_CHECKPOINT_DATA[SDSC_W_LENGTH],
1445 2474 3             .VALUE_CHECKPOINT_DATA[SDSC_A_POINTER]);
1446 2475 3     IF NOT .FINAL_STATUS THEN LEAVE_PROTECTED_UPDATE;
1447 2476 3     END;
1448 2477 3
1449 2478 3
1450 2479 3 IF .ITEM_PRESENT[SJCS_NO_CHECKPOINT_DATA]
1451 2480 3 THEN
1452 2481 3     BEGIN
1453 2482 3         SJH[SJH$S_COMPLETED_BLOCKS] = 0;
1454 2483 3         SJH[SJH$S_CURRENT_FILE_CHKPT] = 0;
1455 2484 3         SJH[SJH$B_JOB_COPIES_CHKPT] = 0;
1456 2485 3         SJH[SJH$B_FILE_COPIES_CHKPT] = 0;
1457 2486 3     END;
1458 2487 3
1459 2488 3
1460 2489 3 IF .ITEM_PRESENT[SJCS_CLI]
1461 2490 3 THEN
1462 2491 3     BEGIN
1463 2492 3         CH$WCHAR(
1464 2493 3             .VALUE_CLI[SDSC_W_LENGTH],
1465 2494 3             SJH[SJH$T_CLI]);
1466 2495 3         CH$COPY(
1467 2496 3             .VALUE_CLI[SDSC_W_LENGTH],
1468 2497 3             .VALUE_CLI[SDSC_A_POINTER],
```

```

: 1469      2498  4      0,
: 1470      2499  4      SJH$S_CLI-1,
: 1471      2500  4      SJH[SJH$T_CLI]+1);
: 1472      2501  3      END;
: 1473      2502  3
: 1474      2503  3
: 1475      2504  3      IF .ITEM_PRESENT[SJC$_NO_CLI]
: 1476      2505  3      THEN
: 1477      2506  3          CH$FILL(
: 1478      2507  3              0,
: 1479      2508  3              SJH$S_CLI,
: 1480      2509  3              SJH[SJH$T_CLI]);
: 1481      2510  3
: 1482      2511  3
: 1483      2512  3      IF .ITEM_PRESENT[SJC$_CPU_LIMIT]
: 1484      2513  3      THEN
: 1485      2514  4          BEGIN
: 1486      2515  4              SJH[SJH$V_CPU_MAXIMUM] = TRUE;
: 1487      2516  4              SJH[SJH$L_CPU_MAXIMUM] = .VALUE_CPU_LIMIT;
: 1488      2517  3          END;
: 1489      2518  3
: 1490      2519  3
: 1491      2520  3      IF .ITEM_PRESENT[SJC$_NO_CPU_LIMIT]
: 1492      2521  3      THEN
: 1493      2522  4          BEGIN
: 1494      2523  4              SJH[SJH$V_CPU_MAXIMUM] = FALSE;
: 1495      2524  4              SJH[SJH$L_CPU_MAXIMUM] = 0;
: 1496      2525  3          END;
: 1497      2526  3
: 1498      2527  3
: 1499      2528  3      IF .ITEM_PRESENT[SJC$_DESTINATION_QUEUE]
: 1500      2529  3      THEN
: 1501      2530  3          SJH[SJH$L_QUEUE_LINK] = .VALUE_DESTINATION_QUEUE[0];
: 1502      2531  3
: 1503      2532  3
: 1504      2533  3      IF .ITEM_PRESENT[SJC$_FILE_BURST]
: 1505      2534  3      THEN
: 1506      2535  4          BEGIN
: 1507      2536  4              SJH[SJH$V_FILE_BURST_ONE] = FALSE;
: 1508      2537  4              SJH[SJH$V_FILE_BURST] = TRUE;
: 1509      2538  4              SJH[SJH$V_FILE_BURST_EXPLICIT] = TRUE;
: 1510      2539  3          END;
: 1511      2540  3
: 1512      2541  3
: 1513      2542  3      IF .ITEM_PRESENT[SJC$_FILE_BURST_ONE]
: 1514      2543  3      THEN
: 1515      2544  4          BEGIN
: 1516      2545  4              SJH[SJH$V_FILE_BURST] = FALSE;
: 1517      2546  4              SJH[SJH$V_FILE_BURST_ONE] = TRUE;
: 1518      2547  4              SJH[SJH$V_FILE_BURST_EXPLICIT] = TRUE;
: 1519      2548  3          END;
: 1520      2549  3
: 1521      2550  3
: 1522      2551  3      IF .ITEM_PRESENT[SJC$_NO_FILE_BURST]
: 1523      2552  3      THEN
: 1524      2553  4          BEGIN
: 1525      2554  4              SJH[SJH$V_FILE_BURST] = FALSE;
```

```
: 1526      2555  4      SJH[SJH$V_FILE_BURST_ONE] = FALSE;
: 1527      2556  4      SJH[SJH$V_FILE_BURST_EXPLICIT] = TRUE;
: 1528      2557  3      END;
: 1529      2558  3
: 1530      2559  3
: 1531      2560  3  IF .ITEM_PRESENT[SJCS_FILE_FLAG]
: 1532      2561  3  THEN
: 1533      2562  4      BEGIN
: 1534      2563  4      SJH[SJH$V_FILE_FLAG_ONE] = FALSE;
: 1535      2564  4      SJH[SJH$V_FILE_FLAG] = TRUE;
: 1536      2565  4      SJH[SJH$V_FILE_FLAG_EXPLICIT] = TRUE;
: 1537      2566  3      END;
: 1538      2567  3
: 1539      2568  3
: 1540      2569  3  IF .ITEM_PRESENT[SJCS_FILE_FLAG_ONE]
: 1541      2570  3  THEN
: 1542      2571  4      BEGIN
: 1543      2572  4      SJH[SJH$V_FILE_FLAG] = FALSE;
: 1544      2573  4      SJH[SJH$V_FILE_FLAG_ONE] = TRUE;
: 1545      2574  4      SJH[SJH$V_FILE_FLAG_EXPLICIT] = TRUE;
: 1546      2575  3      END;
: 1547      2576  3
: 1548      2577  3
: 1549      2578  3  IF .ITEM_PRESENT[SJCS_NO_FILE_FLAG]
: 1550      2579  3  THEN
: 1551      2580  4      BEGIN
: 1552      2581  4      SJH[SJH$V_FILE_FLAG] = FALSE;
: 1553      2582  4      SJH[SJH$V_FILE_FLAG_ONE] = FALSE;
: 1554      2583  4      SJH[SJH$V_FILE_FLAG_EXPLICIT] = TRUE;
: 1555      2584  3      END;
: 1556      2585  3
: 1557      2586  3
: 1558      2587  3  IF .ITEM_PRESENT[SJCS_FILE_TRAILER]
: 1559      2588  3  THEN
: 1560      2589  4      BEGIN
: 1561      2590  4      SJH[SJH$V_FILE_TRAILER_ONE] = FALSE;
: 1562      2591  4      SJH[SJH$V_FILE_TRAILER] = TRUE;
: 1563      2592  4      SJH[SJH$V_FILE_TRAILER_EXPLICIT] = TRUE;
: 1564      2593  3      END;
: 1565      2594  3
: 1566      2595  3
: 1567      2596  3  IF .ITEM_PRESENT[SJCS_FILE_TRAILER_ONE]
: 1568      2597  3  THEN
: 1569      2598  4      BEGIN
: 1570      2599  4      SJH[SJH$V_FILE_TRAILER] = FALSE;
: 1571      2600  4      SJH[SJH$V_FILE_TRAILER_ONE] = TRUE;
: 1572      2601  4      SJH[SJH$V_FILE_TRAILER_EXPLICIT] = TRUE;
: 1573      2602  3      END;
: 1574      2603  3
: 1575      2604  3
: 1576      2605  3  IF .ITEM_PRESENT[SJCS_NO_FILE_TRAILER]
: 1577      2606  3  THEN
: 1578      2607  4      BEGIN
: 1579      2608  4      SJH[SJH$V_FILE_TRAILER] = FALSE;
: 1580      2609  4      SJH[SJH$V_FILE_TRAILER_ONE] = FALSE;
: 1581      2610  4      SJH[SJH$V_FILE_TRAILER_EXPLICIT] = TRUE;
: 1582      2611  3      END;
```

```
: 1583 2612 3
: 1584 2613 3
: 1585 2614 3 IF .ITEM_PRESENT[SJ($_FORM_NAME)]
: 1586 2615 3 OR .ITEM_PRESENT[SJ($_FORM_NUMBER)]
: 1587 2616 3 THEN
: 1588 2617 3     SJH[SJH$_FORM_LINK] = .VALUE_FORM[0];
: 1589 2618 3
: 1590 2619 3
: 1591 2620 3 IF .ITEM_PRESENT[SJ($_HOLD)]
: 1592 2621 3 THEN
: 1593 2622 3     SJH[SJH$_HOLDING] = TRUE;
: 1594 2623 3
: 1595 2624 3
: 1596 2625 3 IF .ITEM_PRESENT[SJ($_NO_HOLD)]
: 1597 2626 3 THEN
: 1598 2627 4 BEGIN
: 1599 2628 4     COPY TIME(CUR TIME, SJH[SJH$_AFTER_TIME]);
: 1600 2629 4     SJH[SJH$_HOLDING] = FALSE;
: 1601 2630 4     SJH[SJH$_REFUSED] = FALSE;
: 1602 2631 4     SJH[SJH$_RETAINED] = FALSE;
: 1603 2632 3 END;
: 1604 2633 3
: 1605 2634 3
: 1606 2635 3 IF .ITEM_PRESENT[SJ($_JOB_COPIES)]
: 1607 2636 3 THEN
: 1608 2637 3     SJH[SJH$_JOB_COPIES] = .VALUE_JOB_COPIES;
: 1609 2638 3
: 1610 2639 3
: 1611 2640 3 IF .ITEM_PRESENT[SJ($_JOB_NAME)]
: 1612 2641 3 THEN
: 1613 2642 4 BEGIN
: 1614 2643 4     CH$WCHAR(
: 1615 2644 4         .VALUE_JOB_NAME[SDSC_W_LENGTH],
: 1616 2645 4         SJH[SJH$_NAME]);
: 1617 2646 4     CH$COPY(
: 1618 2647 4         .VALUE_JOB_NAME[SDSC_W_LENGTH],
: 1619 2648 4         .VALUE_JOB_NAME[SDSC_A_POINTER],
: 1620 2649 4         0,
: 1621 2650 4         SJH$_NAME-1,
: 1622 2651 4         SJH[SJH$_NAME]+1);
: 1623 2652 3 END;
: 1624 2653 3
: 1625 2654 3
: 1626 2655 3 IF .ITEM_PRESENT[SJ($_LOG_DELETE)]
: 1627 2656 3 THEN
: 1628 2657 3     SJH[SJH$_LOG_DELETE] = TRUE;
: 1629 2658 3
: 1630 2659 3
: 1631 2660 3 IF .ITEM_PRESENT[SJ($_NO_LOG_DELETE)]
: 1632 2661 3 THEN
: 1633 2662 3     SJH[SJH$_LOG_DELETE] = FALSE;
: 1634 2663 3
: 1635 2664 3
: 1636 2665 3 IF .ITEM_PRESENT[SJ($_LOG_SPECIFICATION)]
: 1637 2666 3 THEN
: 1638 2667 4 BEGIN
: 1639 2668 4     FINAL_STATUS = STORE_VARIABLE_DATA(
```

```
: 1640      2669  4      .SJH
: 1641      2670  4      SJH$.LOG_SPECIFICATION,
: 1642      2671  4      SJH[SJH$.LOG_SPECIFICATION],
: 1643      2672  4      SYMSK.LOG_SPECIFICATION,
: 1644      2673  4      .VALUE_LOG_SPECIFICATION[SDSC-W-LENGTH],
: 1645      2674  4      .VALUE_LOG_SPECIFICATION[SDSC-A-POINTER]);
: 1646      2675  4      IF NOT .FINAL_STATUS THEN LEAVE PROTECTED_UPDATE;
: 1647      2676  4      SJH[SJH$.LOG_NULL] = FALSE;
: 1648      2677  3      END;
: 1649      2678  3
: 1650      2679  3
: 1651      2680  3      IF .ITEM_PRESENT[SJC$.NO_LOG_SPECIFICATION]
: 1652      2681  3      THEN
: 1653      2682  3          SJH[SJH$.LOG_NULL] = TRUE;
: 1654      2683  3
: 1655      2684  3
: 1656      2685  3      IF .ITEM_PRESENT[SJC$.LOG_SPOOL]
: 1657      2686  3      THEN
: 1658      2687  3          SJH[SJH$.LOG_SPOOL] = TRUE;
: 1659      2688  3
: 1660      2689  3
: 1661      2690  3      IF .ITEM_PRESENT[SJC$.NO_LOG_SPOOL]
: 1662      2691  3      THEN
: 1663      2692  4          BEGIN
: 1664      2693  4              SJH[SJH$.LOG_QUEUE_LINK] = 0;
: 1665      2694  4              SJH[SJH$.LOG_SPOOL] = FALSE;
: 1666      2695  4              IF NOT .ITEM_PRESENT[SJC$.LOG_DELETE] THEN SJH[SJH$.LOG_DELETE] = FALSE;
: 1667      2696  3          END;
: 1668      2697  3
: 1669      2698  3
: 1670      2699  3      IF .ITEM_PRESENT[SJC$.LOG_QUEUE]
: 1671      2700  3      THEN
: 1672      2701  4          BEGIN
: 1673      2702  4              IF .BBLOCK[.VALUE_LOG_QUEUE[1], SMQ$.BATCH]
: 1674      2703  4              THEN
: 1675      2704  5                  BEGIN
: 1676      2705  5                      FINAL_STATUS = JBC$.INCDSTQUE;
: 1677      2706  5                      LEAVE PROTECTED_UPDATE;
: 1678      2707  4                  END;
: 1679      2708  4              SJH[SJH$.LOG_QUEUE_LINK] = .VALUE_LOG_QUEUE[0];
: 1680      2709  4              SJH[SJH$.LOG_SPOOL] = TRUE;
: 1681      2710  3          END;
: 1682      2711  3
: 1683      2712  3
: 1684      2713  3      IF .ITEM_PRESENT[SJC$.LOWERCASE]
: 1685      2714  3      THEN
: 1686      2715  3          SJH[SJH$.LOWERCASE] = TRUE;
: 1687      2716  3
: 1688      2717  3
: 1689      2718  3      IF .ITEM_PRESENT[SJC$.NO_LOWERCASE]
: 1690      2719  3      THEN
: 1691      2720  3          SJH[SJH$.LOWERCASE] = FALSE;
: 1692      2721  3
: 1693      2722  3
: 1694      2723  3      IF .ITEM_PRESENT[SJC$.NOTE
: 1695      2724  3      THEN
: 1696      2725  4          BEGIN
```



```
1697 2726 4 FINAL_STATUS = STORE_VARIABLE_DATA(
1698 2727 4 .SJH,
1699 2728 4 SJH$$ NOTE,
1700 2729 4 SJH[SJH$T NOTE],
1701 2730 4 SYMSK NOTE,
1702 2731 4 .VALUE_NOTE[SDSC_W_LENGTH],
1703 2732 4 .VALUE_NOTE[SDSC_A_POINTER]);
1704 2733 4 IF NOT .FINAL_STATUS THEN LEAVE PROTECTED_UPDATE;
1705 2734 3 END;
1706 2735 3
1707 2736 3
1708 2737 3 IF .ITEM_PRESENT[SJC$_NOTIFY]
1709 2738 3 THEN
1710 2739 3 SJH[SJH$V_NOTIFY] = TRUE;
1711 2740 3
1712 2741 3
1713 2742 3 IF .ITEM_PRESENT[SJC$_NO_NOTIFY]
1714 2743 3 THEN
1715 2744 3 SJH[SJH$V_NOTIFY] = FALSE;
1716 2745 3
1717 2746 3
1718 2747 3 IF .ITEM_PRESENT[SJC$_PAGINATE]
1719 2748 3 THEN
1720 2749 4 BEGIN
1721 2750 4 SJH[SJH$V_PAGINATE] = TRUE;
1722 2751 4 SJH[SJH$V_PAGINATE_EXPLICIT] = TRUE;
1723 2752 3 END;
1724 2753 3
1725 2754 3
1726 2755 3 IF .ITEM_PRESENT[SJC$_NO_PAGINATE]
1727 2756 3 THEN
1728 2757 4 BEGIN
1729 2758 4 SJH[SJH$V_PAGINATE] = FALSE;
1730 2759 4 SJH[SJH$V_PAGINATE_EXPLICIT] = TRUE;
1731 2760 3 END;
1732 2761 3
1733 2762 3
1734 2763 3 IF .BBLOCK[ITEM_PRESENT, 0, SJC$_PARAMETER_1, 8, 0] NEQ 0
1735 2764 3 THEN
1736 2765 4 BEGIN
1737 2766 5 IF (.VALUE_PARAMETER_1[SDSC_W_LENGTH] +
1738 2767 5 .VALUE_PARAMETER_2[SDSC_W_LENGTH] +
1739 2768 5 .VALUE_PARAMETER_3[SDSC_W_LENGTH] +
1740 2769 5 .VALUE_PARAMETER_4[SDSC_W_LENGTH] +
1741 2770 5 .VALUE_PARAMETER_5[SDSC_W_LENGTH] +
1742 2771 5 .VALUE_PARAMETER_6[SDSC_W_LENGTH] +
1743 2772 5 .VALUE_PARAMETER_7[SDSC_W_LENGTH] +
1744 2773 4 .VALUE_PARAMETER_8[SDSC_W_LENGTH]) GTRU JBC$K_MAXPARSIZ_ALL
1745 2774 4 THEN FINAL_STATUS = JBC$ INVPARLEN
1746 2775 4 ELSE FINAL_STATUS = STORE_VARIABLE_DATA_LIST(
1747 2776 4 .SJH,
1748 2777 4 SJH$$ PARAMETERS,
1749 2778 4 SJH[SJH$T PARAMETERS],
1750 2779 4 SYMSK PARAMETERS,
1751 2780 4 .VALUE_PARAMETER_1[SDSC_W_LENGTH], .VALUE_PARAMETER_1[SDSC_A_POINTER],
1752 2781 4 .VALUE_PARAMETER_2[SDSC_W_LENGTH], .VALUE_PARAMETER_2[SDSC_A_POINTER],
1753 2782 4 .VALUE_PARAMETER_3[SDSC_W_LENGTH], .VALUE_PARAMETER_3[SDSC_A_POINTER],
```

```

: 1754      2783  4      .VALUE_PARAMETER_4[SDSC_W_LENGTH], .VALUE_PARAMETER_4[SDSC_A_POINTER],
: 1755      2784  4      .VALUE_PARAMETER_5[SDSC_W_LENGTH], .VALUE_PARAMETER_5[SDSC_A_POINTER],
: 1756      2785  4      .VALUE_PARAMETER_6[SDSC_W_LENGTH], .VALUE_PARAMETER_6[SDSC_A_POINTER],
: 1757      2786  4      .VALUE_PARAMETER_7[SDSC_W_LENGTH], .VALUE_PARAMETER_7[SDSC_A_POINTER],
: 1758      2787  4      .VALUE_PARAMETER_8[SDSC_W_LENGTH], .VALUE_PARAMETER_8[SDSC_A_POINTER]);
: 1759      2788  4      IF NOT .FINAL_STATUS THEN LEAVE PROTECTED_UPDATE;
: 1760      2789  3      END;
: 1761      2790  3
: 1762      2791  3
: 1763      2792  3      IF .ITEM_PRESENT[SJC$_PRIORITY]
: 1764      2793  3      THEN
: 1765      2794  3          SJH[SJH$_PRIORITY] = .VALUE_PRIORITY;
: 1766      2795  3
: 1767      2796  3
: 1768      2797  3      IF .ITEM_PRESENT[SJC$_OPERATOR_REQUEST]
: 1769      2798  3      THEN
: 1770      2799  4          BEGIN
: 1771      2800  4              FINAL_STATUS = STORE_VARIABLE_DATA(
: 1772      2801  4                  .SJH,
: 1773      2802  4                  SJH$_OPERATOR_REQUEST,
: 1774      2803  4                  SJH[SJH$_OPERATOR_REQUEST],
: 1775      2804  4                  SYMSK_OPERATOR_REQUEST,
: 1776      2805  4                  .VALUE_OPERATOR_REQUEST[SDSC_W_LENGTH],
: 1777      2806  4                  .VALUE_OPERATOR_REQUEST[SDSC_A_POINTER]);
: 1778      2807  4              IF NOT .FINAL_STATUS THEN LEAVE PROTECTED_UPDATE;
: 1779      2808  3              END;
: 1780      2809  3
: 1781      2810  3
: 1782      2811  3      IF .ITEM_PRESENT[SJC$_RESTART]
: 1783      2812  3      THEN
: 1784      2813  3          SJH[SJH$_RESTART] = TRUE;
: 1785      2814  3
: 1786      2815  3
: 1787      2816  3      IF .ITEM_PRESENT[SJC$_NO_RESTART]
: 1788      2817  3      THEN
: 1789      2818  3          SJH[SJH$_RESTART] = FALSE;
: 1790      2819  3
: 1791      2820  3
: 1792      2821  3      IF .ITEM_PRESENT[SJC$_WSDEFAULT]
: 1793      2822  3      THEN
: 1794      2823  4          BEGIN
: 1795      2824  4              SJH[SJH$_WSDEFAULT] = TRUE;
: 1796      2825  4              SJH[SJH$_WSDEFAULT] = .VALUE_WSDEFAULT;
: 1797      2826  3          END;
: 1798      2827  3
: 1799      2828  3
: 1800      2829  3      IF .ITEM_PRESENT[SJC$_NO_WSDEFAULT]
: 1801      2830  3      THEN
: 1802      2831  4          BEGIN
: 1803      2832  4              SJH[SJH$_WSDEFAULT] = FALSE;
: 1804      2833  4              SJH[SJH$_WSDEFAULT] = 0;
: 1805      2834  3          END;
: 1806      2835  3
: 1807      2836  3
: 1808      2837  3      IF .ITEM_PRESENT[SJC$_WSEXTENT]
: 1809      2838  3      THEN
: 1810      2839  4          BEGIN
```



```
1811 2840 4    SJH[SJH$V_WSEXTENT] = TRUE;
1812 2841 4    SJH[SJH$W_WSEXTENT] = .VALUE_WSEXTENT;
1813 2842 3    END;
1814 2843 3
1815 2844 3
1816 2845 3    IF .ITEM_PRESENT[SJCS_NO_WSEXTENT]
1817 2846 3    THEN
1818 2847 4        BEGIN
1819 2848 4            SJH[SJH$V_WSEXTENT] = FALSE;
1820 2849 4            SJH[SJH$W_WSEXTENT] = 0;
1821 2850 3        END;
1822 2851 3
1823 2852 3
1824 2853 3    IF .ITEM_PRESENT[SJCS_WSQUOTA]
1825 2854 3    THEN
1826 2855 4        BEGIN
1827 2856 4            SJH[SJH$V_WSQUOTA] = TRUE;
1828 2857 4            SJH[SJH$W_WSQUOTA] = .VALUE_WSQUOTA;
1829 2858 3        END;
1830 2859 3
1831 2860 3
1832 2861 3    IF .ITEM_PRESENT[SJCS_NO_WSQUOTA]
1833 2862 3    THEN
1834 2863 4        BEGIN
1835 2864 4            SJH[SJH$V_WSQUOTA] = FALSE;
1836 2865 4            SJH[SJH$W_WSQUOTA] = 0;
1837 2866 3        END;
1838 2867 3
1839 2868 3
1840 2869 2    END;      ! block PROTECTED_UPDATE
1841 2870 2
1842 2871 2
1843 2872 2    IF .FINAL_STATUS
1844 2873 2    THEN
1845 2874 3        BEGIN
1846 2875 3            IF .ITEM_PRESENT[SJCS_CHECKPOINT_DATA]
1847 2876 3            OR .ITEM_PRESENT[SJCS_NO_CHECKPOINT_DATA]
1848 2877 3            THEN
1849 2878 3                DEALLOCATE VARIABLE DATA(
1850 2879 3                    SJH$S_CHECKPOINT,
1851 2880 3                    LSJH[SJH$T_CHECKPOINT])
1852 2881 3            ELSE
1853 2882 3                CH$MOVE(
1854 2883 3                    SJH$S_CHECKPOINT,
1855 2884 3                    LSJH[SJH$T_CHECKPOINT],
1856 2885 3                    SJH[SJH$T_CHECKPOINT]);
1857 2886 3
1858 2887 3
1859 2888 3            IF .ITEM_PRESENT[SJCS_LOG_SPECIFICATION]
1860 2889 3            OR .ITEM_PRESENT[SJCS_NO_LOG_SPECIFICATION]
1861 2890 3            THEN
1862 2891 3                DEALLOCATE VARIABLE DATA(
1863 2892 3                    SJH$S_LOG_SPECIFICATION,
1864 2893 3                    LSJH[SJH$T_LOG_SPECIFICATION])
1865 2894 3            ELSE
1866 2895 3                CH$MOVE(
1867 2896 3                    SJH$S_LOG_SPECIFICATION,
```

```
1868 2897 3      LSJH[SJH$T_LOG_SPECIFICATION],
1869 2898 3      SJH[SJH$T_LOG_SPECIFICATION]);
1870 2899 3
1871 2900 3
1872 2901 3      IF .ITEM_PRESENT[SJCS_NOTE]
1873 2902 3      OR .ITEM_PRESENT[SJCS_NO_NOTE]
1874 2903 3      THEN
1875 2904 3          DEALLOCATE VARIABLE_DATA(
1876 2905 3              SJH$S_NOTE,
1877 2906 3              LSJH[SJH$T_NOTE])
1878 2907 3      ELSE
1879 2908 3          CH$MOVE(
1880 2909 3              SJH$S_NOTE,
1881 2910 3              LSJH[SJH$T_NOTE],
1882 2911 3              SJH[SJH$T_NOTE]);
1883 2912 3
1884 2913 3
1885 2914 3      IF .BBLOCK[ITEM_PRESENT, 0, SJCS_PARAMETER_1, 8, 0] NEQ 0
1886 2915 3      OR .ITEM_PRESENT[SJCS_NO_PARAMETERS]
1887 2916 3      THEN
1888 2917 3          DEALLOCATE VARIABLE_DATA(
1889 2918 3              SJH$S_PARAMETERS,
1890 2919 3              LSJH[SJH$T_PARAMETERS])
1891 2920 3      ELSE
1892 2921 3          CH$MOVE(
1893 2922 3              SJH$S_PARAMETERS,
1894 2923 3              LSJH[SJH$T_PARAMETERS],
1895 2924 3              SJH[SJH$T_PARAMETERS]);
1896 2925 3
1897 2926 3
1898 2927 3      IF .ITEM_PRESENT[SJCS_OPERATOR_REQUEST]
1899 2928 3      OR .ITEM_PRESENT[SJCS_NO_OPERATOR_REQUEST]
1900 2929 3      THEN
1901 2930 3          DEALLOCATE VARIABLE_DATA(
1902 2931 3              SJH$S_OPERATOR_REQUEST,
1903 2932 3              LSJH[SJH$T_OPERATOR_REQUEST])
1904 2933 3      ELSE
1905 2934 3          CH$MOVE(
1906 2935 3              SJH$S_OPERATOR_REQUEST,
1907 2936 3              LSJH[SJH$T_OPERATOR_REQUEST],
1908 2937 3              SJH[SJH$T_OPERATOR_REQUEST]);
1909 2938 3      END
1910 2939 3  ELSE
1911 2940 3      BEGIN
1912 2941 3          DEALLOCATE VARIABLE_DATA(
1913 2942 3              SJH$S_CHECKPOINT,
1914 2943 3              SJH[SJH$T_CHECKPOINT]);
1915 2944 3          DEALLOCATE VARIABLE_DATA(
1916 2945 3              SJH$S_LOG_SPECIFICATION,
1917 2946 3              SJH[SJH$T_LOG_SPECIFICATION]);
1918 2947 3          DEALLOCATE VARIABLE_DATA(
1919 2948 3              SJH$S_NOTE,
1920 2949 3              SJH[SJH$T_NOTE]);
1921 2950 3          DEALLOCATE VARIABLE_DATA(
1922 2951 3              SJH$S_PARAMETERS,
1923 2952 3              LSJH[SJH$T_PARAMETERS]);
1924 2953 3          DEALLOCATE VARIABLE_DATA(
```

```
: 1925      2954 3      SJH$$ OPERATOR REQUEST,  
: 1926      2955 3      SJH[SJH$T_OPERATOR REQUEST]);  
: 1927      2956 3      CH$MOVE(SYM$$_SYM, LSJH, .SJH);  
: 1928      2957 2      END;  
: 1929      2958 2  
: 1930      2959 2  
: 1931      2960 2      .FINAL_STATUS  
: 1932      2961 1      END;
```

				OFFC 00000	.ENTRY	APPLY_SJH_OPTIONS, Save R2,R3,R4,R5,R6,R7,-	
				5E FDF8 CE 9E 00002	MOVAB	R8,R9,R10,R11	2384
				56 04 AC D0 00007	MOVL	-520(SP), SP	
	08	AE		66 0200 8F 28 0000B	MOVAB	SJH, R6	2423
				5B 0180 C6 9E 00012	MOVAB	#512, (R6), LSJH	
20		00		6E 00 2C 00017	MOVAB	384(R6), R11	2424
				6B 0001C	MOVAB	#0, (SP), #0, #32, (R11)	
				5A 01A0 C6 9E 0001D	MOVAB	416(R6), R10	2425
06		00		6E 00 2C 00022	MOVAB	#0, (SP), #0, #6, (R10)	
				6A 00027	MOVAB	422(R6), R9	2426
				59 01A6 C6 9E 00028	MOVAB	#0, (SP), #0, #6, (R9)	
06		00		6E 00 2C 0002D	MOVAB	434(R6), 4(SP)	2427
				69 00032	MOVAB	#0, (SP), #0, #32, @4(SP)	
20		00	04	AE 01B2 C6 9E 00033	MOVAB	428(R6), R8	2428
				6E 00 2C 00039	MOVAB	#0, (SP), #0, #6, (R8)	
				04 BE 0003F	MOVAB	#1, FINAL_STATUS	2429
06		00		5B 01AC C6 9E 00040	MOVAB	#3, ITEM_PRESENT, 2\$	2433
				6E 00 2C 00045	MOVAB	152(R6), R2	2436
				68 0004A	MOVAB	VALUE_AFTER_TIME, (R2)	
				57 01 D0 0004B	MOVAB	156(R6), R0	
				EF 03 E1 0004E	MOVAB	VALUE_AFTER_TIME+4, (R0)	
				52 0098 C6 9E 00056	MOVAB	CUR_TIME+4, R1	2437
				62 00000000' EF D0 0005B	MOVAB	R1, (R0)	
				50 009C C6 9E 00062	MOVAB	1\$	
				60 00000000' EF D0 00067	MOVAB	2\$	
				51 00000000' EF D0 0006E	MOVAB	CUR_TIME, (R2)	
				60 51 D1 00075	MOVAB	2\$	
				0B 1A 00078	MOVAB	CUR_TIME, (R2)	
				13 12 0007A	MOVAB	R1, (R0)	
				62 00000000' EF D1 0007C	MOVAB	#4, ITEM_PRESENT, 3\$	2439
				0A 1F 00083	MOVAB	CUR_TIME, 152(R6)	
				62 00000000' EF D0 00085	MOVAB	#4, ITEM_PRESENT+1, 4\$	2443
				60 51 D0 0008C	MOVAB	#4, ITEM_PRESENT+1, 5\$	2448
				09 00000000' EF 04 E1 0008F	MOVAB	#5, ITEM_PRESENT+1, 6\$	2454
				0098 C6 00000000' EF 7D 00097	MOVAB	#6, ITEM_PRESENT+1, 6\$	2457
				08 00000000' EF 04 E0 000A0	MOVAB	#0, (SP), #0, #16, 160(R6)	2462
				0A 00000000' EF 05 E1 000A8	MOVAB	ITEM_PRESENT+1	2465
				06 00000000' EF 10 28 000B0	MOVAB	7\$	
				08 00000000' EF 06 E1 000BA	MOVAB		
10		00		6E 00 2C 000C2	MOVAB		
				00A0 C6 000C7	MOVAB		
				00000000' EF 95 000CA	MOVAB		
				25 18 000D0	MOVAB		

		00000000'	EF	DD	000D2	PUSHL	VALUE CHECKPOINT_DATA+2	2474			
	7E	00000000'	EF	3C	000D8	MOVZWL	VALUE CHECKPOINT_DATA, -(SP)	2473			
			0B	DD	000DF	PUSHL	#11	2471			
			5B	DD	000E1	PUSHL	R11				
			20	DD	000E3	PUSHL	#32				
			56	DD	000E5	PUSHL	R6				
	00000000G	EF	06	FB	000E7	CALLS	#6, STORE_VARIABLE_DATA				
		57	50	D0	000FE	MOVL	R0, FINAL_STATUS				
		03	57	E8	000F1	BLBS	FINAL_STATUS, 7\$	2475			
			04F6	31	000F4	BRW	75\$				
		10	00000000'	EF	E9	000F7	7\$: BLBC	ITEM_PRESENT+2, 8\$	2479		
			00D8	C6	D4	000FE	CLRL	216(R6)	2482		
			00EC	C6	D4	00102	CLRL	236(R6)	2483		
			017B	C6	94	00106	CLRB	379(R6)	2484		
			0178	C6	94	0010A	CLRB	376(R6)	2485		
	19	00000000'	EF	01	E1	0010E	8\$: BBC	#1, ITEM_PRESENT+2, 9\$	2489		
		00B0	C6	0C	00000000'	EF	90	00116	MOVBL	VALUE_CLI, 176(R6)	2494
27	00	00000000'	FF	00000000'	EF	2C	0011F	MOVCS	VALUE_CLI, @VALUE_CLI+2, #0, #39, 177(R6)	2500	
				00B1	C6		0012C				
	08	00000000'	EF	02	E1	0012F	9\$: BBC	#2, ITEM_PRESENT+2, 10\$	2504		
28	00		6E	00	2C	00137	MOVCS	#0, (SP), #0, #40, 176(R6)	2509		
				00B0	C6		0013C				
	0D	00000000'	EF	05	E1	0013F	10\$: BBC	#5, ITEM_PRESENT+2, 11\$	2512		
		0C	A6	01	88	00147	BISB2	#1, 12(R6)	2515		
		00E8	C6	00000000'	EF	D0	0014B	MOVL	VALUE_CPU_LIMIT, 232(R6)	2516	
	08	00000000'	EF	06	E1	00154	11\$: BBC	#6, ITEM_PRESENT+2, 12\$	2520		
		0C	A6	01	8A	0015C	BICB2	#1, 12(R6)	2523		
				00E8	C6	D4	00160	CLRL	232(R6)	2524	
	09	00000000'	EF	02	E1	00164	12\$: BBC	#2, ITEM_PRESENT+3, 13\$	2528		
		0134	C6	00000000'	EF	D0	0016C	MOVL	VALUE_DESTINATION_QUEUE, 308(R6)	2530	
			08	00000000'	EF	E9	00175	13\$: BLBC	ITEM_PRESENT+4, 14\$	2533	
		0C	A6	08	8A	0017C	BICB2	#8, 12(R6)	2536		
		0C	A6	06	88	00180	BISB2	#6, 12(R6)	2538		
	08	00000000'	EF	01	E1	00184	14\$: BBC	#1, ITEM_PRESENT+4, 15\$	2542		
		0C	A6	02	8A	0018C	BICB2	#2, 12(R6)	2545		
		0C	A6	0C	88	00190	BISB2	#12, 12(R6)	2547		
	08	00000000'	EF	02	E1	00194	15\$: BBC	#2, ITEM_PRESENT+4, 16\$	2551		
		0C	A6	0A	8A	0019C	BICB2	#10, 12(R6)	2555		
		0C	A6	04	88	001A0	BISB2	#4, 12(R6)	2556		
	09	00000000'	EF	04	E1	001A4	16\$: BBC	#4, ITEM_PRESENT+4, 17\$	2560		
		0C	A6	40	8F	8A	001AC	BICB2	#64, 12(R6)	2563	
		0C	A6	30	88	001B1	BISB2	#48, 12(R6)	2565		
	09	00000000'	EF	05	E1	001B5	17\$: BBC	#5, ITEM_PRESENT+4, 18\$	2569		
		0C	A6	10	8A	001BD	BICB2	#16, 12(R6)	2572		
		0C	A6	60	8F	88	001C1	BISB2	#96, 12(R6)	2574	
	09	00000000'	EF	06	E1	001C6	18\$: BBC	#6, ITEM_PRESENT+4, 19\$	2578		
		0C	A6	50	8F	8A	001CE	BICB2	#80, 12(R6)	2582	
		0C	A6	20	88	001D3	BISB2	#32, 12(R6)	2583		
	0A	00000000'	EF	03	E1	001D7	19\$: BBC	#3, ITEM_PRESENT+5, 20\$	2587		
		0D	A6	02	8A	001DF	BICB2	#2, 13(R6)	2590		
		0C	A6	0180	8F	A8	001E3	BISW2	#384, 12(R6)	2592	
	09	00000000'	EF	04	E1	001E9	20\$: BBC	#4, ITEM_PRESENT+5, 21\$	2596		
		0C	A6	80	8F	8A	001F1	BICB2	#128, 12(R6)	2599	
		0D	A6	03	88	001F6	BISB2	#3, 13(R6)	2601		
	0A	00000000'	EF	05	E1	001FA	21\$: BBC	#5, ITEM_PRESENT+5, 22\$	2605		
		0C	A6	0280	8F	AA	00202	BICW2	#640, 12(R6)	2609	
		0D	A6	01	88	00208	BISB2	#1, 13(R6)	2610		

08	00000000'	EF	00000000'	06	E0 0020C	22\$:	BBS	#6, ITEM_PRESENT+6, 23\$	2614
				EF	95 0C214		TSTB	ITEM_PRESENT+6	2615
				09	18 0021A		BGEQ	24\$	
		C6	00000000'	EF	D0 0021C	23\$:	MOVL	VALUE_FORM, 252(R6)	2617
			00000000'	EF	95 00225	24\$:	TSTB	ITEM_PRESENT+8	2620
				04	18 0022B		BGEQ	25\$	
	10	A6		20	88 0022D		BISB2	#32, 16(R6)	2622
		OF	00000000'	EF	E9 00231	25\$:	BLBC	ITEM_PRESENT+9, 26\$	2625
	0098	C6	00000000'	EF	7D 00238		MOVQ	CUR_TIME, 152(R6)	2628
	10	A6	08A0	8F	AA 00241		BICW2	#2208, 16(R6)	2631
09	00000000'	EF		03	E1 00247	26\$:	BBC	#3, ITEM_PRESENT+9, 27\$	2635
	017A	C6	00000000'	EF	90 0024F		MOVB	VALUE_JOB_COPIES, 378(R6)	2637
			00000000'	EF	95 00258	27\$:	TSTB	ITEM_PRESENT+9	2640
				19	18 0025E		BGEQ	28\$	
	0108	C6	00000000'	EF	90 00260		MOVB	VALUE_JOB_NAME, 264(R6)	2645
27	00 00000000'	FF	00000000'	EF	2C 00269		MOVCS	VALUE_JOB_NAME, @VALUE_JOB_NAME+2, #0, -	2651
			0109	C6	00276			#39, 265(R6)	
			00000000'	EF	95 00279	28\$:	TSTB	ITEM_PRESENT+11	2655
				04	18 0027F		BGEQ	29\$	
	0D	A6		04	88 00281		BISB2	#4, 13(R6)	2657
		04	00000000'	EF	E9 00285	29\$:	BLBC	ITEM_PRESENT+12, 30\$	2660
	0D	A6		04	8A 0028C		BICB2	#4, 13(R6)	2662
29	00000000'	EF		02	E1 00290	30\$:	BBC	#2, ITEM_PRESENT+12, 32\$	2665
			00000000'	EF	DD 00298		PUSHL	VALUE_LOG_SPECIFICATION+2	2674
		7E	00000000'	EF	3C 0029E		MOVZWL	VALUE_LOG_SPECIFICATION, -(SP)	2673
				10	DD 002A5		PUSHL	#16	2671
				5A	DD 002A7		PUSHL	R10	
				06	DD 002A9		PUSHL	#6	
				56	DD 002AB		PUSHL	R6	
	00000000G	EF		06	FB 002AD		CALLS	#6, STORE_VARIABLE_DATA	
		57		50	D0 002B4		MOVL	R0, FINAL_STATUS	
		03		57	E8 002B7		BLBS	FINAL_STATUS, 31\$	2675
				0330	31 002BA		BRW	75\$	
	0D	A6		08	8A 002BD	31\$:	BICB2	#8, 13(R6)	2676
04	00000000'	EF		03	E1 002C1	32\$:	BBC	#3, ITEM_PRESENT+12, 33\$	2680
	0D	A6		08	88 002C9		BISB2	#8, 13(R6)	2682
04	00000000'	EF		04	E1 002CD	33\$:	BBC	#4, ITEM_PRESENT+12, 34\$	2685
	0D	A6		10	88 002D5		BISB2	#16, 13(R6)	2687
14	00000000'	EF		05	E1 002D9	34\$:	BBC	#5, ITEM_PRESENT+12, 35\$	2690
			0104	C6	D4 002E1		CLRL	260(R6)	2693
	0D	A6		10	8A 002E5		BICB2	#16, 13(R6)	2694
			00000000'	EF	95 002E9		TSTB	ITEM_PRESENT+11	2695
				04	19 002EF		BLSS	35\$	
	0D	A6		04	8A 002F1		BICB2	#4, 13(R6)	
22	00000000'	EF		01	E1 002F5	35\$:	BBC	#1, ITEM_PRESENT+12, 37\$	2699
		50	00000000'	EF	50 002FD		MOVL	VALUE_LOG_QUEUE+4, R0	2702
		0A	0C	A0	E9 00304		BLBC	12(R0), 38\$	
		57	00048120	8F	D0 00308		MOVL	#295200, FINAL_STATUS	2705
				0221	31 0030F		BRW	59\$	2706
	0104	C6	00000000'	EF	D0 00312	36\$:	MOVL	VALUE_LOG_QUEUE, 260(R6)	2708
	0D	A6		10	88 0031B		BISB2	#16, 13(R6)	2709
04	00000000'	EF		06	E1 0031F	37\$:	BBC	#6, ITEM_PRESENT+12, 38\$	2713
	0D	A6		20	88 00327		BISB2	#32, 13(R6)	2715
			00000000'	EF	95 0032B	38\$:	TSTB	ITEM_PRESENT+12	2718
				04	18 00331		BGEQ	39\$	
	0D	A6		20	8A 00333		BICB2	#32, 13(R6)	2720
25	00000000'	EF		02	E1 00337	39\$:	BBC	#2, ITEM_PRESENT+13, 40\$	2723

	00000000'	EF	DD	0033F	PUSHL	VALUE_NOTE+2	2732		
7E	00000000'	EF	3C	00345	MOVZWL	VALUE_NOTE, -(SP)	2731		
		11	DD	0034C	PUSHL	#17	2729		
		59	DD	0034E	PUSHL	R9			
		06	DD	00350	PUSHL	#6			
		56	DD	00352	PUSHL	R6			
00000000G	EF	06	FB	00354	CALLS	#6, STORE_VARIABLE_DATA			
	57	50	DD	0035B	MOVL	R0, FINAL_STATUS			
	03	57	E8	0035E	BLBS	FINAL_STATUS, 40\$	2733		
		0289	31	00361	BRW	75\$			
05 00000000'	EF	04	E1	00364	BBC	#4, ITEM_PRESENT+13, 41\$	2737		
	OD	A6	40	8F	88	0036C	BISB2	#64, 13(R6)	2739
05 00000000'	EF	05	E1	00371	BBC	#5, ITEM_PRESENT+13, 42\$	2742		
	OD	A6	40	8F	8A	00379	BICB2	#64, 13(R6)	2744
06 00000000'	EF	05	E1	0037E	BBC	#5, ITEM_PRESENT+14, 43\$	2747		
	OD	A6	0180	8F	A8	00386	BISW2	#384, 13(R6)	2751
09 00000000'	EF	06	E1	0038C	BBC	#6, ITEM_PRESENT+14, 44\$	2755		
	OD	A6	80	8F	8A	00394	BICB2	#128, 13(R6)	2758
	OE	A6		01	88	00399	BISB2	#1, 14(R6)	2759
7F80	8F	00000000'	EF	B3	0039D	44\$:	BITW	ITEM_PRESENT+14, #32640	2763
			03	12	003A6		BNEQ	45\$	
			00C4	31	003A8		BRW	48\$	
	50	00000000'	EF	3C	003AB	45\$:	MOVZWL	VALUE_PARAMETER_1, R0	2767
	51	00000000'	EF	3C	003B2		MOVZWL	VALUE_PARAMETER_2, R1	
	50		51	CO	003B9		ADDL2	R1, R0	
	55	00000000'	EF	3C	003BC		MOVZWL	VALUE_PARAMETER_3, R5	2768
	50		55	CO	003C3		ADDL2	R5, R0	2767
	54	00000000'	EF	3C	003C6		MOVZWL	VALUE_PARAMETER_4, R4	2769
	50		54	CO	003CD		ADDL2	R4, R0	2768
	53	00000000'	EF	3C	003D0		MOVZWL	VALUE_PARAMETER_5, R3	2770
	50		53	CO	003D7		ADDL2	R3, R0	2769
	52	00000000'	EF	3C	003DA		MOVZWL	VALUE_PARAMETER_6, R2	2771
	50		52	CO	003E1		ADDL2	R2, R0	2770
	51	00000000'	EF	3C	003E4		MOVZWL	VALUE_PARAMETER_7, R1	2772
	50		51	CO	003EB		ADDL2	R1, R0	2771
	6E	00000000'	EF	3C	003EE		MOVZWL	VALUE_PARAMETER_8, (SP)	2773
	50		6E	CO	003F5		ADDL2	(SP), R0	
000001E0	8F		50	D1	003F8		CMPL	R0, #480	
			09	1B	003FF		BLEQU	46\$	
	57	00048090	8F	DD	00401		MOVL	#295056, FINAL_STATUS	2774
			62	11	00408		BRB	47\$	
	00000000'	EF	DD	0040A	46\$:		PUSHL	VALUE_PARAMETER_8+2	2787
7E	00000000'	EF	3C	00410			MOVZWL	VALUE_PARAMETER_8, -(SP)	
	00000000'	EF	DD	00417			PUSHL	VALUE_PARAMETER_7+2	2786
			51	DD	0041D		PUSHL	R1	
	00000000'	EF	DD	0041F			PUSHL	VALUE_PARAMETER_6+2	2785
			52	DD	00425		PUSHL	R2	
	00000000'	EF	DD	00427			PUSHL	VALUE_PARAMETER_5+2	2784
			53	DD	0042D		PUSHL	R3	
	00000000'	EF	DD	0042F			PUSHL	VALUE_PARAMETER_4+2	2783
			54	DD	00435		PUSHL	R4	
	00000000'	EF	DD	00437			PUSHL	VALUE_PARAMETER_3+2	2782
			55	DD	0043D		PUSHL	R5	
	00000000'	EF	DD	0043F			PUSHL	VALUE_PARAMETER_2+2	2781
7E	00000000'	EF	3C	00445			MOVZWL	VALUE_PARAMETER_2, -(SP)	
	00000000'	EF	DD	0044C			PUSHL	VALUE_PARAMETER_1+2	2780
7E	00000000'	EF	3C	00452			MOVZWL	VALUE_PARAMETER_1, -(SP)	

			14	DD	00459	PUSHL	#20	2778
		48	AE	DD	0045B	PUSHL	72(SP)	
			20	DD	0045E	PUSHL	#32	
			56	DD	00460	PUSHL	R6	
	00000000G	EF	14	FB	00462	CALLS	#20, STORE_VARIABLE_DATA_LIST	
		57	50	DD	00469	MOVL	R0, FINAL_STATUS	
		38	57	E9	0046C	BLBC	FINAL_STATUS, 50\$	2788
09	00000000'	EF	02	E1	0046F	BBC	#2, ITEM_PRESENT+16, 49\$	2792
	017D	C6	90	00477	MOVB	VALUE_PRIORITY, 381(R6)		2794
25	00000000'	EF	05	E1	00480	BBC	#6, ITEM_PRESENT+13, 51\$	2797
			EF	DD	00488	PUSHL	VALUE_OPERATOR_REQUEST+2	2806
		7E	EF	3C	0048E	MOVZWL	VALUE_OPERATOR_REQUEST, -(SP)	2805
			12	DD	00495	PUSHL	#18	2803
			58	DD	00497	PUSHL	R8	
			06	DD	00499	PUSHL	#6	
			56	DD	0049B	PUSHL	R6	
	00000000G	EF	06	FB	0049D	CALLS	#6, STORE_VARIABLE_DATA	
		57	50	DD	004A4	MOVL	R0, FINAL_STATUS	
		03	57	E8	004A7	BLBS	FINAL_STATUS, 51\$	2807
			0140	31	004AA	BRW	75\$	
04	00000000'	EF	02	E1	004AD	BBC	#2, ITEM_PRESENT+17, 52\$	2811
	OE	A6	02	88	004B5	BISB2	#2, 14(R6)	2813
04	00000000'	EF	03	E1	004B9	BBC	#3, ITEM_PRESENT+17, 53\$	2816
	OE	A6	02	8A	004C1	BICB2	#2, 14(R6)	2818
			EF	95	004C5	TSTB	ITEM_PRESENT+18	2821
			0D	18	004CB	BGEQ	54\$	
	OE	A6	04	88	004CD	BISB2	#4, 14(R6)	2824
	0172	C6	EF	B0	004D1	MOVW	VALUE_WSDFAULT, 370(R6)	2825
		08	EF	E9	004DA	BLBC	ITEM_PRESENT+19, 55\$	2829
	OE	A6	04	8A	004E1	BICB2	#4, 14(R6)	2832
		0172	C6	B4	004E5	CLRW	370(R6)	2833
0D	00000000'	EF	01	E1	004E9	BBC	#1, ITEM_PRESENT+19, 56\$	2837
	OE	A6	08	88	004F1	BISB2	#8, 14(R6)	2840
	0174	C6	EF	B0	004F5	MOVW	VALUE_WSEXTENT, 372(R6)	2841
08	00000000'	EF	02	E1	004FE	BBC	#2, ITEM_PRESENT+19, 57\$	2845
	OE	A6	08	8A	00506	BICB2	#8, 14(R6)	2848
		0174	C6	B4	0050A	CLRW	372(R6)	2849
0D	00000000'	EF	03	E1	0050E	BBC	#3, ITEM_PRESENT+19, 58\$	2853
	OE	A6	10	88	00516	BISB2	#16, 14(R6)	2856
	0176	C6	EF	B0	0051A	MOVW	VALUE_WSQUOTA, 374(R6)	2857
08	00000000'	EF	04	E1	00523	BBC	#4, ITEM_PRESENT+19, 59\$	2861
	OE	A6	10	8A	0052B	BICB2	#16, 14(R6)	2864
		0176	C6	B4	0052F	CLRW	374(R6)	2865
			57	E8	00533	BLBS	FINAL_STATUS, 60\$	2872
			00B4	31	00536	BRW	75\$	
			EF	95	00539	TSTB	ITEM_PRESENT+1	2875
			07	19	0053F	BLSS	61\$	
		OE	EF	E9	00541	BLBC	ITEM_PRESENT+2, 62\$	2876
		80	AD	9F	00548	PUSHAB	LSJH+384	2880
			20	DD	0054B	PUSHL	#32	
	00000000G	EF	02	FB	0054D	CALLS	#2, DEALLOCATE_VARIABLE_DATA	
			05	11	00554	BRB	63\$	
6B	80	AD	20	28	00556	MOV3	#32, LSJH+384, (R11)	2885
08	00000000'	EF	02	E0	0055B	BBS	#2, ITEM_PRESENT+12, 64\$	2888
0E	00000000'	EF	03	E1	00563	BBC	#3, ITEM_PRESENT+12, 65\$	2889
			AD	9F	0056B	PUSHAB	LSJH+416	2893
		A0	06	DD	0056E	PUSHL	#6	

00000000G	EF		02	FB	00570	CALLS	#2, DEALLOCATE_VARIABLE_DATA			
			05	11	00577	BRB	66\$			
6A	A0	AD	06	28	00579	65\$:	MOV C3	#6, LSJH+416, (R10)	2898	
08	00000000'	EF	02	E0	0057E	66\$:	BBS	#2, ITEM_PRESENT+13, 67\$	2901	
0E	00000000'	EF	03	E1	00586		BBC	#3, ITEM_PRESENT+13, 68\$	2902	
		A6	AD	9F	0058E	67\$:	PUSHAB	LSJH+422	2906	
			06	DD	00591		PUSHL	#6		
00000000G	EF		02	FB	00593	CALLS	#2, DEALLOCATE_VARIABLE_DATA			
			05	11	0059A	BRB	69\$			
69	A6	AD	06	28	0059C	68\$:	MOV C3	#6, LSJH+422, (R9)	2911	
	7F80	8F	00000000'	EF	B3	005A1	69\$:	BITW	ITEM_PRESENT+14, #32640	2914
			08	12	005AA		BNEQ	70\$		
			00000000'	EF	95	005AC		TSTB	ITEM_PRESENT+15	2915
			0E	18	005B2		BGEQ	71\$		
		B2	AD	9F	005B4	70\$:	PUSHAB	LSJH+434	2919	
			20	DD	005B7		PUSHL	#32		
00000000G	EF		02	FB	005B9	CALLS	#2, DEALLOCATE_VARIABLE_DATA			
			06	11	005C0	BRB	72\$			
04	BE	B2	AD	20	28	005C2	71\$:	MOV C3	#32, LSJH+434, @4(SP)	2924
	08	00000000'	EF	06	E0	005C8	72\$:	BBS	#6, ITEM_PRESENT+13, 73\$	2927
		00000000'	EF	95	005D0		TSTB	ITEM_PRESENT+13	2928	
			0E	18	005D6		BGEQ	74\$		
		AC	AD	9F	005D8	73\$:	PUSHAB	LSJH+428	2932	
			06	DD	005DB		PUSHL	#6		
00000000G	EF		02	FB	005DD	CALLS	#2, DEALLOCATE_VARIABLE_DATA			
			46	11	005E4	BRB	76\$			
68	AC	AD	06	28	005E6	74\$:	MOV C3	#6, LSJH+428, (R8)	2937	
			3F	11	005EB		BRB	76\$	2872	
			58	DD	005ED	75\$:	PUSHL	R11	2943	
			20	DD	005EF		PUSHL	#32		
00000000G	EF		02	FB	005F1	CALLS	#2, DEALLOCATE_VARIABLE_DATA			
			5A	DD	005F8		PUSHL	R10	2946	
			06	DD	005FA		PUSHL	#6		
00000000G	EF		02	FB	005FC	CALLS	#2, DEALLOCATE_VARIABLE_DATA			
			59	DD	00603		PUSHL	R9	2949	
			06	DD	00605		PUSHL	#6		
00000000G	EF		02	FB	00607	CALLS	#2, DEALLOCATE_VARIABLE_DATA			
		B2	AD	9F	0060E		PUSHAB	LSJH+434	2952	
			20	DD	00611		PUSHL	#32		
00000000G	EF		02	FB	00613	CALLS	#2, DEALLOCATE_VARIABLE_DATA			
			58	DD	0061A		PUSHL	R8	2955	
			06	DD	0061C		PUSHL	#6		
00000000G	EF		02	FB	0061E	CALLS	#2, DEALLOCATE_VARIABLE_DATA			
66	08	AE	0200	8F	28	00625	76\$:	MOV C3	#512, LSJH, (R8)	2956
		50		57	D0	0062C		MOVL	FINAL_STATUS, R0	2961
				04	0062F		RET			

; Routine Size: 1584 bytes, Routine Base: CODE + 0956


```

: 1934      2962 1 GLOBAL ROUTINE DELETE_SJH_RECORD(SJH_N,SJH): NOVALUE=
: 1935      2963 1
: 1936      2964 1  ++
: 1937      2965 1
: 1938      2966 1  FUNCTIONAL DESCRIPTION:
: 1939      2967 1      This routine deallocates a job header and its substructures.
: 1940      2968 1
: 1941      2969 1  INPUT PARAMETERS:
: 1942      2970 1      SJH_N          - Record number of SJH.
: 1943      2971 1      SJH          - Pointer to SJH.
: 1944      2972 1
: 1945      2973 1  IMPLICIT INPUTS:
: 1946      2974 1      NONE
: 1947      2975 1
: 1948      2976 1  OUTPUT PARAMETERS:
: 1949      2977 1      NONE
: 1950      2978 1
: 1951      2979 1  IMPLICIT OUTPUTS:
: 1952      2980 1      NONE
: 1953      2981 1
: 1954      2982 1  ROUTINE VALUE:
: 1955      2983 1      NONE
: 1956      2984 1
: 1957      2985 1  SIDE EFFECTS:
: 1958      2986 1      NONE
: 1959      2987 1
: 1960      2988 1  --
: 1961      2989 1
: 1962      2990 2 BEGIN
: 1963      2991 2 MAP
: 1964      2992 2      SJH:          REF BBLOCK;      ! Pointer to SJH
: 1965      2993 2 LOCAL
: 1966      2994 2      SQR_N,          ! Record number of SQR
: 1967      2995 2      SQR_NS,        ! Record number of successor of SQR
: 1968      2996 2      SQR,          REF BBLOCK,      ! Pointer to SQR
: 1969      2997 2      DELETIONS;      ! List of deletions to process
: 1970      2998 2
: 1971      2999 2
: 1972      3000 2 ! If there are services outstanding for this job, cancel them.
: 1973      3001 2
: 1974      3002 2 IF .SJH[SJH$V_ABORTING]
: 1975      3003 2 OR .SJH[SJH$V_STARTING]
: 1976      3004 2 THEN
: 1977      3005 2     SCAN_INCOMPLETE_SERVICES(ISRV_K_PURGE_SJH, .SJH_N);
: 1978      3006 2
: 1979      3007 2
: 1980      3008 2 ! Deallocate the fixed/variable data areas, if they exist.
: 1981      3009 2
: 1982      3010 2 DEALLOCATE VARIABLE DATA(
: 1983      3011 2     SJH$S_CHECKPOINT,
: 1984      3012 2     SJH[SJH$T_CHECKPOINT]);
: 1985      3013 2 DEALLOCATE VARIABLE DATA(
: 1986      3014 2     SJH$S_LOG_SPECIFICATION,
: 1987      3015 2     SJH[SJH$T_LOG_SPECIFICATION]);
: 1988      3016 2 DEALLOCATE VARIABLE DATA(
: 1989      3017 2     SJH$S_NOTE,
: 1990      3018 2     SJH[SJH$T_NOTE]);
```

```
: 1991 3019 2 DEALLOCATE VARIABLE DATA(
: 1992 3020 2     SJHSS PARAMETERS,
: 1993 3021 2     SJH[SJHST_PARAMETERS]);
: 1994 3022 2 DEALLOCATE VARIABLE DATA(
: 1995 3023 2     SJHSS REFUSAL REASON,
: 1996 3024 2     SJH[SJHST_REFUSAL_REASON]);
: 1997 3025 2 DEALLOCATE VARIABLE DATA(
: 1998 3026 2     SJHSS OPERATOR REQUEST,
: 1999 3027 2     SJH[SJHST_OPERATOR_REQUEST]);
: 2000 3028 2
: 2001 3029 2
: 2002 3030 2 ! Deallocate the file record list.
: 2003 3031 2 !
: 2004 3032 2 DELETIONS = 0;
: 2005 3033 2 SQR_N = .SJH[SJHSL_FILE_LIST];
: 2006 3034 2 WHILE .SQR_N NEQ 0 DO
: 2007 3035 3 BEGIN
: 2008 3036 3     SQR = READ_RECORD(.SQR_N);
: 2009 3037 3     SQR_NS = .SQR[SYMSL_LINK];
: 2010 3038 3
: 2011 3039 3
: 2012 3040 3 ! Cancel the delete option for an open job, or for a job that is deleted
: 2013 3041 3 ! before execution, unless the file is not in a directory.
: 2014 3042 3 !
: 2015 3043 3 IF
: 2016 3044 4     (.SJH[SJH$V_DELETED]
: 2017 3045 4     AND NOT .SJH[SJH$V_RETAINED]
: 2018 3046 4     AND .SQR[SQR$W_FILE_ID_DID_NUM] NEQ 0)
: 2019 3047 3 OR
: 2020 3048 3     .SJH[SJH$V_OPEN]
: 2021 3049 3 THEN
: 2022 3050 3     SQR[SQR$V_DELETE_FILE] = FALSE;
: 2023 3051 3
: 2024 3052 3
: 2025 3053 3 DELETE_SQR_RECORD(.SQR_N, .SQR, DELETIONS);
: 2026 3054 3 SQR_N = .SQR_NS;
: 2027 3055 2 END;
: 2028 3056 2
: 2029 3057 2
: 2030 3058 2 ! Execute file deletions, if any.
: 2031 3059 2 !
: 2032 3060 2 IF .DELETIONS NEQ 0
: 2033 3061 2 THEN
: 2034 3062 3     IF SYSID_NEQ(THIS_SYSID, SJH[SJHST_SYSID])
: 2035 3063 2     THEN
: 2036 3064 2         CREATE_SQR_RECORD(SQR$K_DELETE_FILES, .SJH, .DELETIONS)
: 2037 3065 2     ELSE
: 2038 3066 2         DELETE_FILES(.DELETIONS);
: 2039 3067 2
: 2040 3068 2
: 2041 3069 2 ! Deallocate the job entry number.
: 2042 3070 2 !
: 2043 3071 2 DEALLOCATE_ENTRY_NUMBER(.SJH[SYMSL_ENTRY_NUMBER]);
: 2044 3072 2
: 2045 3073 2
: 2046 3074 2 ! Finally, deallocate the job header record.
: 2047 3075 2 !
```

: 2048
: 20493076 2 DEALLOCATE_RECORD(.SJH_N);
3077 1 END;

			00FC 00000	.ENTRY	DELETE SJH_RECORD, Save R2,R3,R4,R5,R6,R7	2962
	57	00000000G	EF 9E 00002	MOVAB	DEALLOCATE_VARIABLE_DATA, R7	
	5E		04 C2 00009	SUBL2	#4, SP	
	52	08	AC D0 0000C	MOVL	SJH, R2	3002
	55	10	A2 9E 00010	MOVAB	16(R2), R5	
04	65		01 E0 00014	BBS	#1, (R5), 1\$	
0C	65		0C E1 00018	BBC	#12, (R5), 2\$	3003
		04	AC DD 0001C	PUSHL	SJH_N	3005
			05 DD 0001F	PUSHL	#5	
	00000000G	EF	02 FB 00021	CALLS	#2, SCAN_INCOMPLETE_SERVICES	
		0180	C2 9F 00028	PUSHAB	384(R2)	3012
			20 DD 0002C	PUSHL	#32	
	67		02 FB 0002E	CALLS	#2, DEALLOCATE_VARIABLE_DATA	
		01A0	C2 9F 00031	PUSHAB	416(R2)	3015
			06 DD 00035	PUSHL	#6	
	67		02 FB 00037	CALLS	#2, DEALLOCATE_VARIABLE_DATA	
		01A6	C2 9F 0003A	PUSHAB	422(R2)	3018
			06 DD 0003E	PUSHL	#6	
	67		02 FB 00040	CALLS	#2, DEALLOCATE_VARIABLE_DATA	
		01B2	C2 9F 00043	PUSHAB	434(R2)	3021
			20 DD 00047	PUSHL	#32	
	67		02 FB 00049	CALLS	#2, DEALLOCATE_VARIABLE_DATA	
		01D2	C2 9F 0004C	PUSHAB	466(R2)	3024
			06 DD 00050	PUSHL	#6	
	67		02 FB 00052	CALLS	#2, DEALLOCATE_VARIABLE_DATA	
		01AC	C2 9F 00055	PUSHAB	428(R2)	3027
			06 DD 00059	PUSHL	#6	
	67		02 FB 0005B	CALLS	#2, DEALLOCATE_VARIABLE_DATA	
			6E D4 0005E	CLRL	DELETIONS	3032
	54	00F4	C2 D0 00060	MOVL	244(R2), SQR_N	3033
			34 13 00065	BEQL	7\$	3034
			54 DD 00067	PUSHL	SQR_N	3036
	00000000G	EF	01 FB 00069	CALLS	#1, READ_RECORD	
		53	50 D0 00070	MOVL	R0, SQR	
		56	63 D0 00073	MOVL	(SQR), SQR_NS	3037
09	65		02 E1 00076	BBC	#2, (R5), 2\$	3044
05	65		0B E0 0007A	BBS	#11, (R5), 4\$	3045
		32	A3 B5 0007E	TSTW	50(SQR)	3046
			04 12 00081	BNEQ	5\$	
04	65		06 E1 00083	BBC	#6, (R5), 6\$	3048
	0C	A3	01 8A 00087	BICB2	#1, 12(SQR)	3050
		4008	8F BB 0008B	PUSHR	#*M<R3,SP>	3053
			54 DD 0008F	PUSHL	SQR_N	
	0000V	CF	03 FB 00091	CALLS	#3, DELETE_SQR_RECORD	
		54	56 D0 00096	MOVL	SQR_NS, SQR_N	3054
			CA 11 00099	BRB	3\$	3034
		50	6E D0 0009B	MOVL	DELETIONS, R0	3060
			2C 13 0009E	BEQL	10\$	
	016C	C2 00000000'	EF D1 000A0	CMPL	THIS_SYSID, 364(R2)	3062
			0B 12 000A9	BNEQ	8\$	

0170	C2	00000000'	EF	B1	000AB	CMPW	THIS_SYSID+4, 368(R2)	:	
			0F	13	000B4	BEQL	9\$	:	
			50	DD	000B6	8\$: PUSH	R0	:	3064
			52	DD	000B8	PUSH	R2	:	
			08	DD	000BA	PUSH	#11	:	
00000000G	EF		03	FB	000BC	CALLS	#3, CREATE_SRQ_RECORD	:	
			07	11	000C3	BRB	10\$	:	
			50	DD	000C5	9\$: PUSH	R0	:	3066
0000V	CF		01	FB	000C7	CALLS	#1, DELETE_FILES	:	
		08	A2	DD	000CC	10\$: PUSH	8(R2)	:	3071
00000000G	EF		01	FB	000CF	CALLS	#1, DEALLOCATE_ENTRY_NUMBER	:	
		04	AC	DD	000D6	PUSH	SJH_N	:	3076
00000000G	EF		01	FB	000D9	CALLS	#1, DEALLOCATE_RECORD	:	
			04	000E0		RET		:	3077

; Routine Size: 225 bytes, Routine Base: CODE + 0F86

; F

```
2051 3078 1 GLOBAL ROUTINE CREATE_SQR_RECORD(SJH;SQR_N,SQR): L_OUTPUT_2=
2052 3079 1
2053 3080 1 ++
2054 3081 1
2055 3082 1 FUNCTIONAL DESCRIPTION:
2056 3083 1 This routine allocates and initializes a job file record.
2057 3084 1
2058 3085 1 INPUT PARAMETERS:
2059 3086 1 SJH - Pointer to SJH.
2060 3087 1
2061 3088 1 IMPLICIT INPUTS:
2062 3089 1 NONE
2063 3090 1
2064 3091 1 OUTPUT PARAMETERS:
2065 3092 1 SQR_N - Record number of SQR.
2066 3093 1 SQR - Pointer to SQR.
2067 3094 1
2068 3095 1 IMPLICIT OUTPUTS:
2069 3096 1 NONE
2070 3097 1
2071 3098 1 ROUTINE VALUE:
2072 3099 1 Completion status.
2073 3100 1
2074 3101 1 SIDE EFFECTS:
2075 3102 1 NONE
2076 3103 1
2077 3104 1 --
2078 3105 1
2079 3106 2 BEGIN
2080 3107 2 MAP
2081 3108 2 SJH: REF BBLOCK, ! Pointer to SJH
2082 3109 2 SQR: REF BBLOCK; ! Pointer to SQR
2083 3110 2 LOCAL
2084 3111 2 STATUS;
2085 3112 2
2086 3113 2
2087 3114 2 ! Allocate the queue record, and return if no more.
2088 3115 2
2089 3116 2 STATUS = ALLOCATE_RECORD( ; SQR_N, SQR);
2090 3117 2 IF NOT .STATUS THEN RETURN .STATUS;
2091 3118 2
2092 3119 2
2093 3120 2 ! Initialize the queue record.
2094 3121 2
2095 3122 2 SQR[SYMSB_TYPE] = SYMSK_SQR;
2096 3123 2 SQR[SYMSL_ENTRY_NUMBER] = .SJH[SYMSL_ENTRY_NUMBER];
2097 3124 2 SQR[SQR$V_PAGINATE] = TRUE;
2098 3125 2 CH$MOVE(
2099 3126 2 SQR$S_FILE_IDENTIFICATION,
2100 3127 2 VALUE_FILE_IDENTIFICATION,
2101 3128 2 SQR[SQR$T_FILE_IDENTIFICATION]);
2102 3129 2 SQR[SQR$L_FILE_SIZE] = .VALUE_FILE_IDENTIFICATION[28,0,32,0];
2103 3130 2 SQR[SQR$B_FILE_COPIES] = 1;
2104 3131 2 CH$WCHAR(
2105 3132 2 .VALUE_FILE_SPECIFICATION[SDSC_W_LENGTH],
2106 3133 2 SQR[SQR$T_FILE_SPECIFICATION]);
2107 3134 2 CH$COPY(
```

```
2108 3135 2 .VALUE_FILE_SPECIFICATION[SDSC_W_LENGTH],
2109 3136 2 .VALUE_FILE_SPECIFICATION[SDSC_A_POINTER],
2110 3137 2 0,
2111 3138 2 SQRSS_FILE_SPECIFICATION-1,
2112 3139 2 SQR[SQRST_FILE_SPECIFICATION]+1);
2113 3140 2
2114 3141 2
2115 3142 2 IF .ITEM_PRESENT[SJCS_BATCH_INPUT]
2116 3143 2 THEN
2117 3144 3 BEGIN
2118 3145 3 LOCAL
2119 3146 3 P; REF BBLOCK, ! Cursor for item list
2120 3147 3 P_END; ! Pointer past end of item list
2121 3148 3
2122 3149 3
2123 3150 3 ! Pick up a pointer to the item list and one to the last item.
2124 3151 3 !
2125 3152 3 P = .VALUE_BATCH_INPUT[SDSC_A_POINTER];
2126 3153 3 P_END = .P + .VALUE_BATCH_INPUT[SDSC_W_LENGTH] - 4;
2127 3154 3
2128 3155 3
2129 3156 3 ! Loop over the items.
2130 3157 3 !
2131 3158 3 WHILE .P LSSA .P_END DO
2132 3159 4 BEGIN
2133 3160 4 LOCAL
2134 3161 4 TYPE, ! Item type
2135 3162 4 SIZE; ! Item size
2136 3163 4
2137 3164 4
2138 3165 4 ! Get and advance over the item type and size.
2139 3166 4 !
2140 3167 4 TYPE = .P[DJISW_ITEM_CODE];
2141 3168 4 SIZE = .P[DJISW_ITEM_SIZE];
2142 3169 4 P = .P + DJISS_ITEM_HEADER;
2143 3170 4
2144 3171 4
2145 3172 4 ! Process the item.
2146 3173 4 !
2147 3174 4 CASE .TYPE FROM DJISK_CONDITION_VECTOR TO DJISK_FILE_SPECIFICATION OF
2148 3175 4 SET
2149 3176 4
2150 3177 4
2151 3178 4 [OUTRANGE]:
2152 3179 4 EXITLOOP;
2153 3180 4
2154 3181 4
2155 3182 4 [DJISK_CONDITION_VECTOR]:
2156 3183 5 BEGIN
2157 3184 5 CH$COPY(
2158 3185 5 .SIZE, .P,
2159 3186 5 0,
2160 3187 5 SQRSS_CONDITION_VECTOR, SQR[SQRSL_CONDITION_1]);
2161 3188 4 END;
2162 3189 4
2163 3190 4
2164 3191 4 [DJISK_FILE_SPECIFICATION]:
```



```

2165 3192 5 BEGIN
2166 3193 5 CH$WCHAR(
2167 3194 5 .SIZE,
2168 3195 5 SQR[SQR$T_FILE_SPECIFICATION]);
2169 3196 5 CH$COPY(
2170 3197 5 .SIZE, .P,
2171 3198 5 0,
2172 3199 5 SQR$S_FILE_SPECIFICATION-1,
2173 3200 5 SQR[SQR$T_FILE_SPECIFICATION]+1);
2174 3201 4 END;
2175 3202 4
2176 3203 4
2177 3204 4 TES;
2178 3205 4
2179 3206 4
2180 3207 4 ! Advance to the next item.
2181 3208 4 !
2182 3209 4 P = .P + .SIZE;
2183 3210 3 END;
2184 3211 2 END;
2185 3212 2
2186 3213 2
2187 3214 2 ! Default the job name from the file name of the first file.
2188 3215 2 !
2189 3216 2 IF .SJH[SJH$L_FILE_LIST] EQL 0 AND CH$RCHAR(SJH[SJH$T_NAME]) EQL 0
2190 3217 2 THEN
2191 3218 3 BEGIN
2192 3219 3 LOCAL
2193 3220 3 DESC: VECTOR[2]; ! Descriptor for file specification
2194 3221 3 ITMLST: BBLOCK[12]; ! $FILESCAN item list
2195 3222 3
2196 3223 3
2197 3224 3 DESC[0] = CH$RCHAR(SQR[SQR$T_FILE_SPECIFICATION]);
2198 3225 3 DESC[1] = SQR[SQR$T_FILE_SPECIFICATION] + 1;
2199 3226 3 ITMLST[0,0,16,0] = 0;
2200 3227 3 ITMLST[2,0,16,0] = FSCNS_NAME;
2201 3228 3 ITMLST[4,0,32,0] = 0;
2202 3229 3 ITMLST[8,0,32,0] = 0;
2203 3230 3 $FILESCAN(SRCSTR=DESC, VALUELST=ITMLST);
2204 3231 3 IF .ITMLST[0,0,16,0] GTRU SJH$S_NAME-1 THEN ITMLST[0,0,16,0] = SJH$S_NAME-1;
2205 3232 3 CH$WCHAR(
2206 3233 3 .ITMLST[0,0,16,0],
2207 3234 3 SJH[SJH$T_NAME]);
2208 3235 3 CH$COPY(
2209 3236 3 .ITMLST[0,0,16,0], .ITMLST[4,0,32,0],
2210 3237 3 0,
2211 3238 3 SJH$S_NAME-1, SJH[SJH$T_NAME]+1);
2212 3239 2 END;
2213 3240 2
2214 3241 2
2215 3242 2 SS$ NORMAL
2216 3243 1 END;
```

.EXTRN SY\$FILESCAN

				03FC 00000	.ENTRY	CREATE_SQR_RECORD, Save R2,R3,R4,R5,R6,R7,-	
			SE	14 C2 00002	SUBL2	R8,R9	3078
		00000000G	EF	00 FB 00005	CALLS	#20, SP	
			01	50 E8 0000C	BLBS	#0, ALLOCATE_RECORD	3116
				04 0000F	RET	STATUS, 1\$	3117
		04	AB	08 90 00010 1\$:	MOVB	#8, 4(SQR)	3122
			57	AC D0 00014	SJH, R7		3123
		08	AB	A7 D0 00018	MOVL	8(R7), 8(SQR)	
		0D	AB	04 88 0001D	BISB2	#4, 1\$ (SQR)	3124
1C	AB	00000000'	EF	1C 28 00021	MOVCS	#28, VALUE_FILE_IDENTIFICATION, 28(SQR)	3128
		38	AB	00000000' EF D0 0002A	MOVL	VALUE_FILE_IDENTIFICATION+28, 56(SQR)	3129
		44	AB	01 90 00032	MOVB	#1, 68(SQR)	3130
		4B	AB	00000000' EF 90 00036	MOVB	VALUE_FILE_SPECIFICATION, 75(SQR)	3133
00FF	8F	00 00000000'	FF	00000000' EF 2C 0003E	MOVCS	VALUE_FILE_SPECIFICATION, -	3139
				4C AB 0004D		@VALUE_FILE_SPECIFICATION+2, #0, #255, -	
		4B 00000000'	EF	02 E1 0004F	BBC	#2, ITEM_PRESENT+1, 7\$	3142
			56	00000000' EF D0 00057	MOVL	VALUE_BATCH_INPUT+2, P	3152
			50	00000000' EF 3C 0005E	MOVZWL	VALUE_BATCH_INPUT, R0	3153
			59	FC A046 9E 00065	MOVAB	-4(R0)[P], P_END	
			59	56 D1 0006A 2\$:	CPL	P, P_END	3158
				33 1E 0006D	BGEQU	7\$	
			50	02 A6 3C 0006F	MOVZWL	2(P), TYPE	3167
			58	86 3C 00073	MOVZWL	(P)+, SIZE	3168
			56	02 C0 00076	ADDL2	#2, P	3169
		01 00008002	8F	50 CF 00079	CASEL	TYPE, #32770, #1	3174
			000F	0006 00081 3\$:	.WORD	4\$-3\$, -	
						5\$-3\$	
				1B 11 00085	BRB	7\$	3179
		00	66	58 2C 00087 4\$:	MOVCS	SIZE, (P), #0, #12, 16(SQR)	3187
				10 AB 0008C			
				0D 11 0008E	BRB	6\$	3174
		4B	AB	58 90 00090 5\$:	MOVB	SIZE, 75(SQR)	3195
00FF	8F	00	66	58 2C 00094	MOVCS	SIZE, (P), #0, #255, 76(SQR)	3200
				4C AB 0009B			
			56	58 C0 0009D 6\$:	ADDL2	SIZE, P	3209
				C8 11 000A0	BRB	2\$	3158
				00F4 C7 D5 000A2 7\$:	TSTL	244(R7)	3216
				3F 12 000A6	BNEQ	9\$	
				0108 C7 95 000A8	TSTB	264(R7)	
				39 12 000AC	BNEQ	9\$	
		0C	AE	4B AB 9A 000AE	MOVZBL	75(SQR), DESC	3224
		10	AE	4C AB 9E 000B3	MOVAB	76(SQR), DESC+4	3225
			6E	00060000 8F D0 000B8	MOVL	#393216, ITMLST	3226
				04 AE 7C 000BF	CLRQ	ITMLST+4	3228
				7E D4 000C2	CLRL	-(SP)	3230
				04 AE 9F 000C4	PUSHAB	ITMLST	
			14	AE 9F 000C7	PUSHAB	DESC	
		00000000G	00	03 FB 000CA	CALLS	#3, SYSS\$FILESCAN	
			27	6E B1 000D1	CMPW	ITMLST, #39	3231
				23 1B 000D4	BLEQU	8\$	
			6E	27 B0 000D6	MOVW	#39, ITMLST	
		0108	C7	6E 90 000D9 8\$:	MOVB	ITMLST, 264(R7)	3234
27		04	BE	6E 2C 000DE	MOVCS	ITMLST, @ITMLST+4, #0, #39, 265(R7)	3238
				0109 C7 000E4			
			50	01 D0 000E7 9\$:	MOVL	#1, R0	3243

Queue record manipulation utilities

16-Sep-1984 00:18:27
14-Sep-1984 12:37:12

VAX-11 Bliss-32 V4.0-742
[JOBCTL.SRC]RECORDUTL.B32;1

Page 71
(14)

REC
V04.

04 000EA

RET

```
; Routine Size: 235 bytes,    Routine Base: CODE + 1067
```

[illegible]

```
: 2218 3244 1 GLOBAL ROUTINE APPLY_SQR_OPTIONS(SJH,SQR)=
: 2219 3245 1
: 2220 3246 1 |++
: 2221 3247 1 |
: 2222 3248 1 | FUNCTIONAL DESCRIPTION:
: 2223 3249 1 |     This routine applies options to a job file record.
: 2224 3250 1 |
: 2225 3251 1 | INPUT PARAMETERS:
: 2226 3252 1 |     SJH             - Pointer to SJH.
: 2227 3253 1 |     SQR             - Pointer to SQR.
: 2228 3254 1 |
: 2229 3255 1 | IMPLICIT INPUTS:
: 2230 3256 1 |     Existing values in SQR.
: 2231 3257 1 |
: 2232 3258 1 | OUTPUT PARAMETERS:
: 2233 3259 1 |     NONE
: 2234 3260 1 |
: 2235 3261 1 | IMPLICIT OUTPUTS:
: 2236 3262 1 |     New values in SQR.
: 2237 3263 1 |
: 2238 3264 1 | ROUTINE VALUE:
: 2239 3265 1 |     Completion status.
: 2240 3266 1 |
: 2241 3267 1 | SIDE EFFECTS:
: 2242 3268 1 |     NONE
: 2243 3269 1 |
: 2244 3270 1 | --
: 2245 3271 1 |
: 2246 3272 2 BEGIN
: 2247 3273 2 MAP
: 2248 3274 2     SJH:          REF BBLOCK,      ! Pointer to SJH
: 2249 3275 2     SQR:          REF BBLOCK;      ! Pointer to SQR
: 2250 3276 2 LOCAL
: 2251 3277 2     LSQR:          BBLOCK[SYM$$_SYM],    ! Saved copy of original SQR
: 2252 3278 2     FINAL_STATUS;      ! Final routine status
: 2253 3279 2 LABEL
: 2254 3280 2     PROTECTED_UPDATE;
: 2255 3281 2
: 2256 3282 2
: 2257 3283 2 ! Set up for backing out the transaction in case of an error.
: 2258 3284 2
: 2259 3285 2 CH$MOVE(SYM$$_SYM, .SQR, LSQR);
: 2260 3286 2 CH$FILL(0, SQR$$ FILE_SETUP_MODULES, SQR[SQR$T_FILE_SETUP_MODULES]);
: 2261 3287 2 FINAL_STATUS = S$$_NORMAL;
: 2262 3288 3 PROTECTED_UPDATE: BEGIN
: 2263 3289 3
: 2264 3290 3
: 2265 3291 3 IF .ITEM_PRESENT[SJCS_DELETE_FILE]
: 2266 3292 3 THEN
: 2267 3293 3     SQR[SQR$V_DELETE_FILE] = TRUE;
: 2268 3294 3
: 2269 3295 3
: 2270 3296 3 IF .ITEM_PRESENT[SJCS_NO_DELETE_FILE]
: 2271 3297 3 THEN
: 2272 3298 3     SQR[SQR$V_DELETE_FILE] = FALSE;
: 2273 3299 3
: 2274 3300 3
```

```
2275 3301 3 IF .ITEM_PRESENT[SJCS_DOUBLE_SPACE]
2276 3302 3 THEN
2277 3303 3   SQR[SQR$V_DOUBLE_SPACE] = TRUE;
2278 3304 3
2279 3305 3
2280 3306 3 IF .ITEM_PRESENT[SJCS_NO_DOUBLE_SPACE]
2281 3307 3 THEN
2282 3308 3   SQR[SQR$V_DOUBLE_SPACE] = FALSE;
2283 3309 3
2284 3310 3
2285 3311 3 IF .ITEM_PRESENT[SJCS_FILE_BURST]
2286 3312 3 THEN
2287 3313 4   BEGIN
2288 3314 4     SQR[SQR$V_FILE_BURST] = TRUE;
2289 3315 4     SQR[SQR$V_FILE_BURST_EXPLICIT] = TRUE;
2290 3316 3   END;
2291 3317 3
2292 3318 3
2293 3319 3 IF .ITEM_PRESENT[SJCS_NO_FILE_BURST]
2294 3320 3 THEN
2295 3321 4   BEGIN
2296 3322 4     SQR[SQR$V_FILE_BURST] = FALSE;
2297 3323 4     SQR[SQR$V_FILE_BURST_EXPLICIT] = TRUE;
2298 3324 3   END;
2299 3325 3
2300 3326 3
2301 3327 3 IF .ITEM_PRESENT[SJCS_FILE_COPIES]
2302 3328 3 THEN
2303 3329 3   SQR[SQR$B_FILE_COPIES] = .VALUE_FILE_COPIES;
2304 3330 3
2305 3331 3
2306 3332 3 IF .ITEM_PRESENT[SJCS_FILE_FLAG]
2307 3333 3 THEN
2308 3334 4   BEGIN
2309 3335 4     SQR[SQR$V_FILE_FLAG] = TRUE;
2310 3336 4     SQR[SQR$V_FILE_FLAG_EXPLICIT] = TRUE;
2311 3337 3   END;
2312 3338 3
2313 3339 3
2314 3340 3 IF .ITEM_PRESENT[SJCS_NO_FILE_FLAG]
2315 3341 3 THEN
2316 3342 4   BEGIN
2317 3343 4     SQR[SQR$V_FILE_FLAG] = FALSE;
2318 3344 4     SQR[SQR$V_FILE_FLAG_EXPLICIT] = TRUE;
2319 3345 3   END;
2320 3346 3
2321 3347 3
2322 3348 3 IF .ITEM_PRESENT[SJCS_FILE_SETUP_MODULES]
2323 3349 3 THEN
2324 3350 4   BEGIN
2325 3351 4     FINAL STATUS = STORE_VARIABLE_DATA(
2326 3352 4       .SQR,
2327 3353 4       SQR$S_FILE_SETUP_MODULES,
2328 3354 4       SQR[SQR$T_FILE_SETUP_MODULES],
2329 3355 4       SYM$K_FILE_SETUP_MODULES,
2330 3356 4       .VALUE_FILE_SETUP_MODULES[SDSC_W_LENGTH],
2331 3357 4       .VALUE_FILE_SETUP_MODULES[SDSC_A_POINTER]);
```

```

2332 3358 4 IF NOT .FINAL_STATUS THEN LEAVE PROTECTED_UPDATE;
2333 3359 3 END;
2334 3360 3
2335 3361 3
2336 3362 3 IF .ITEM_PRESENT[SJCS_FILE_TRAILER]
2337 3363 3 THEN
2338 3364 4 BEGIN
2339 3365 4 SQR[SQR$V_FILE_TRAILER] = TRUE;
2340 3366 4 SQR[SQR$V_FILE_TRAILER_EXPLICIT] = TRUE;
2341 3367 4 END;
2342 3368 3
2343 3369 3
2344 3370 3 IF .ITEM_PRESENT[SJCS_NO_FILE_TRAILER]
2345 3371 3 THEN
2346 3372 4 BEGIN
2347 3373 4 SQR[SQR$V_FILE_TRAILER] = FALSE;
2348 3374 4 SQR[SQR$V_FILE_TRAILER_EXPLICIT] = TRUE;
2349 3375 4 END;
2350 3376 3
2351 3377 3
2352 3378 3 IF .ITEM_PRESENT[SJCS_FIRST_PAGE]
2353 3379 3 THEN
2354 3380 3 SQR[SQR$L_FIRST_PAGE] = .VALUE_FIRST_PAGE;
2355 3381 3
2356 3382 3
2357 3383 3 IF .ITEM_PRESENT[SJCS_NO_FIRST_PAGE]
2358 3384 3 THEN
2359 3385 3 SQR[SQR$L_FIRST_PAGE] = 0;
2360 3386 3
2361 3387 3
2362 3388 3 IF .ITEM_PRESENT[SJCS_LAST_PAGE]
2363 3389 3 THEN
2364 3390 3 SQR[SQR$L_LAST_PAGE] = .VALUE_LAST_PAGE;
2365 3391 3
2366 3392 3
2367 3393 3 IF .ITEM_PRESENT[SJCS_NO_LAST_PAGE]
2368 3394 3 THEN
2369 3395 3 SQR[SQR$L_LAST_PAGE] = 0;
2370 3396 3
2371 3397 3
2372 3398 3 IF .ITEM_PRESENT[SJCS_PAGE_HEADER]
2373 3399 3 THEN
2374 3400 3 SQR[SQR$V_PAGE_HEADER] = TRUE;
2375 3401 3
2376 3402 3
2377 3403 3 IF .ITEM_PRESENT[SJCS_NO_PAGE_HEADER]
2378 3404 3 THEN
2379 3405 3 SQR[SQR$V_PAGE_HEADER] = FALSE;
2380 3406 3
2381 3407 3
2382 3408 3 IF .ITEM_PRESENT[SJCS_PAGINATE]
2383 3409 3 THEN
2384 3410 4 BEGIN
2385 3411 4 SQR[SQR$V_PAGINATE] = TRUE;
2386 3412 4 SQR[SQR$V_PAGINATE_EXPLICIT] = TRUE;
2387 3413 4 END;
2388 3414 3
```

```
: 2389      3415 3
: 2390      3416 3 IF .ITEM_PRESENT[SJCS_NO_PAGINATE]
: 2391      3417 3 THEN
: 2392      3418 4 BEGIN
: 2393      3419 4   SQR[SQR$V_PAGINATE] = FALSE;
: 2394      3420 4   SQR[SQR$V_PAGINATE_EXPLICIT] = TRUE;
: 2395      3421 3   END;
: 2396      3422 3
: 2397      3423 3
: 2398      3424 3 IF .ITEM_PRESENT[SJCS_PASSALL]
: 2399      3425 3 THEN
: 2400      3426 3   SQR[SQR$V_PASSALL] = TRUE;
: 2401      3427 3
: 2402      3428 3
: 2403      3429 3 IF .ITEM_PRESENT[SJCS_NC_PASSALL]
: 2404      3430 3 THEN
: 2405      3431 3   SQR[SQR$V_PASSALL] = FALSE;
: 2406      3432 3
: 2407      3433 3
: 2408      3434 2 END;      ! block PROTECTED_UPDATE
: 2409      3435 2
: 2410      3436 2
: 2411      3437 2 IF .FINAL_STATUS
: 2412      3438 2 THEN
: 2413      3439 3 BEGIN
: 2414      3440 3   IF .ITEM_PRESENT[SJCS_FILE_SETUP_MODULES]
: 2415      3441 3   OR .ITEM_PRESENT[SJCS_NO_FILE_SETUP_MODULES]
: 2416      3442 3   THEN
: 2417      3443 3       DEALLOCATE VARIABLE DATA(
: 2418      3444 3           SQR$S_FILE_SETUP_MODULES,
: 2419      3445 3           LSQR[SQR$T_FILE_SETUP_MODULES])
: 2420      3446 3   ELSE
: 2421      3447 3       CH$MOVE(
: 2422      3448 3           SQR$S_FILE_SETUP_MODULES,
: 2423      3449 3           LSQR[SQR$T_FILE_SETUP_MODULES],
: 2424      3450 3           SQR[SQR$T_FILE_SETUP_MODULES]);
: 2425      3451 3   END
: 2426      3452 2 ELSE
: 2427      3453 3 BEGIN
: 2428      3454 3   DEALLOCATE VARIABLE DATA(
: 2429      3455 3       SQR$S_FILE_SETUP_MODULES,
: 2430      3456 3       SQR[SQR$T_FILE_SETUP_MODULES]);
: 2431      3457 3   CH$MOVE(SYM$S_SYM, LSQR, .SQR);
: 2432      3458 2   END;
: 2433      3459 2
: 2434      3460 2
: 2435      3461 2 SJH[SJH$L_JOB_SIZE] = .SJH[SJH$L_JOB_SIZE] +
: 2436      3462 2   .SQR[SQR$[FILE_SIZE] * .SQR[SQR$B_FILE_COPIES] * .SJH[SJH$B_JOB_COPIES];
: 2437      3463 2
: 2438      3464 2
: 2439      3465 2 .FINAL_STATUS
: 2440      3466 1 END;
```

				03FC 00000	.ENTRY	APPLY_SQR_OPTIONS, Save R2,R3,R4,R5,R6,R7,-	3244
		59	00000000G	EF 9E 00002	MOVAB	R8,R9	
		58	000000000	EF 9E 00009	MOVAB	DEALLOCATE VARIABLE_DATA, R9	
		5E	FE00	CE 9E 00010	MOVAB	ITEM_PRESENT+5, R8	
		56	08	AC D0 00015	MOVL	-512(SP), SP	
	6E	66	0200	8F 28 00019	MOVCL	SQR, R6	3285
06	00	6E		00 2C 0001F	MOVCL	#512, (R6), LSQR	
			45	A6 00024	MOVCL	#0, (SP), #0, #6, 69(R6)	3286
		57		01 D0 00026	MOVL	#1, FINAL STATUS	3287
		04	FE	A8 E9 00029	BLBC	ITEM_PRESENT+3, 1\$	3291
	OC	A6		01 88 0002D	BISB2	#1, 12(R6)	3293
04	FE	A8		01 E1 00031 1\$:	BBC	#1, ITEM_PRESENT+3, 2\$	3296
	OC	A6		01 8A 00036	BICB2	#1, 12(R6)	3298
04	FE	A8		04 E1 0003A 2\$:	BBC	#4, ITEM_PRESENT+3, 3\$	3301
	OC	A6		04 88 0003F	BISB2	#4, 12(R6)	3303
04	FE	A8		05 E1 00043 3\$:	BBC	#5, ITEM_PRESENT+3, 4\$	3306
	OC	A6		04 8A 00048	BICB2	#4, 12(R6)	3308
		04	FF	A8 E9 0004C 4\$:	BLBC	ITEM_PRESENT+4, 5\$	3311
	OC	A6		18 88 00050	BISB2	#24, 12(R6)	3315
08	FF	A8		02 E1 00054 5\$:	BBC	#2, ITEM_PRESENT+4, 6\$	3319
	OC	A6		08 8A 00059	BICB2	#8, 12(R6)	3322
	OC	A6		10 88 0005D	BISB2	#16, 12(R6)	3323
06	FF	A8		03 E1 00061 6\$:	BBC	#3, ITEM_PRESENT+4, 7\$	3327
	44	A6	0089	C8 90 00066	MOVCL	VALUE_FILE_COPIES, 68(R6)	3329
05	FF	A8		04 E1 0006C 7\$:	BBC	#4, ITEM_PRESENT+4, 8\$	3332
	OC	A6	60	8F 88 00071	BISB2	#96, 12(R6)	3336
09	FF	A8		06 E1 00076 8\$:	BBC	#6, ITEM_PRESENT+4, 9\$	3340
	OC	A6		20 8A 0007B	BICB2	#32, 12(R6)	3343
	OC	A6	40	8F 88 0007F	BISB2	#64, 12(R6)	3344
		1F		68 E9 00084 9\$:	BLBC	ITEM_PRESENT+5, 10\$	3348
			00B0	C8 DD 00087	PUSHL	VALUE_FILE_SETUP_MODULES+2	3357
		7E	00AE	C8 3C 0008B	MOVZWL	VALUE_FILE_SETUP_MODULES, -(SP)	3356
				OC DD 00090	PUSHL	#12	3354
			45	A6 9F 00092	PUSHAB	69(R6)	
				06 DD 00095	PUSHL	#6	
				56 DD 00097	PUSHL	R6	
	00000000G	EF		06 FB 00099	CALLS	#6, STORE_VARIABLE_DATA	
		57		50 D0 000A0	MOVL	R0, FINAL STATUS	
		74		57 E9 000A3	BLBC	FINAL STATUS, 22\$	3358
06		68		03 E1 000A6 10\$:	BBC	#3, ITEM_PRESENT+5, 11\$	3362
	OC	A6	0180	8F A8 000AA	BISW2	#384, 12(R6)	3366
09		68		05 E1 000B0 11\$:	BBC	#5, ITEM_PRESENT+5, 12\$	3370
	OC	A6	80	8F 8A 000B4	BICB2	#128, 12(R6)	3373
	OD	A6		01 88 000B9	BISB2	#1, 13(R6)	3374
06		68		06 E1 000BD 12\$:	BBC	#6, ITEM_PRESENT+5, 13\$	3378
	3C	A6	00BA	C8 D0 000C1	MOVL	VALUE_FIRST_PAGE, 60(R6)	3380
				68 95 000C7 13\$:	TSTB	ITEM_PRESENT+5	3383
				03 18 000C9	BGEQ	14\$	
			3C	A6 D4 000CB	CLRL	60(R6)	3385
06	06	A8		03 E1 000CE 14\$:	BBC	#3, ITEM_PRESENT+11, 15\$	3388
	40	A6	04EF	C8 D0 000D3	MOVL	VALUE_LAST_PAGE, 64(R6)	3390
03	06	A8		04 E1 000D9 15\$:	BBC	#4, ITEM_PRESENT+11, 16\$	3393
			40	A6 D4 000DE	CLRL	64(R6)	3395
04	09	A8		01 E1 000E1 16\$:	BBC	#1, ITEM_PRESENT+14, 17\$	3398
	OD	A6		02 88 000E6	BISB2	#2, 13(R6)	3400
04	09	A8		02 E1 000EA 17\$:	BBC	#2, ITEM_PRESENT+14, 18\$	3403

	0D	A6		02	8A	000EF		BICB2	#2, 13(R6)	:	3405
04	09	A8		05	E1	000F3	18\$:	BBC	#5, ITEM_PRESENT+14, 19\$	:	3408
	0D	A6		0C	88	000F8		BISB2	#12, 13(R6)	:	3412
08	09	A8		06	E1	000FC	19\$:	BBC	#6, ITEM_PRESENT+14, 20\$	:	3416
	0D	A6		04	8A	00101		BICB2	#4, 13(R6)	:	3419
	0D	A6		08	88	00105		BISB2	#8, 13(R6)	:	3420
		04	0B	A8	E9	00109	20\$:	BLBC	ITEM_PRESENT+10, 21\$	:	3424
	0D	A6		10	88	0010D		BISB2	#16, 13(R6)	:	3426
04	0B	A8		01	E1	00111	21\$:	BBC	#1, ITEM_PRESENT+16, 22\$	:	3429
	0D	A6		10	8A	00116		BICB2	#16, 13(R6)	:	3431
		19		57	E9	0011A	22\$:	BLBC	FINAL_STATUS, 25\$	:	3437
		04		68	E8	0011D		BLBS	ITEM_PRESENT+5, 23\$	:	3440
0A		68		01	E1	00120		BBC	#1, ITEM_PRESENT+5, 24\$	:	3441
			45	AE	9F	00124	23\$:	PUSHAB	LSQR+69	:	3445
				06	DD	00127		PUSHL	#6	:	
		69		02	FB	00129		CALLS	#2, DEALLOCATE_VARIABLE_DATA	:	
45	A6			16	11	0012C		BRB	26\$	:	
		45	AE	06	28	0012E	24\$:	MOV3	#6, LSQR+69, 69(R6)	:	3450
				0E	11	00134		BRB	26\$	:	3437
			45	A6	9F	00136	25\$:	PUSHAB	69(R6)	:	3456
				06	DD	00139		PUSHL	#6	:	
		69		02	FB	0013B		CALLS	#2, DEALLOCATE_VARIABLE_DATA	:	
66		6E	0200	8F	28	0013E		MOV3	#512, LSQR, (R6)	:	3457
		50	04	AC	D0	00144	26\$:	MOVL	SJH, R0	:	3461
		51	44	A6	9A	00148		MOVZBL	68(R6), R1	:	3462
56	38	A6		51	C5	0014C		MULL3	R1, 56(R6), R6	:	
		52	017A	C0	9A	00151		MOVZBL	378(R0), R2	:	
		56		52	C4	00156		MULL2	R2, R6	:	
	0100	C0		56	C0	00159		ADDL2	R6, 256(R0)	:	
		50		57	D0	0015E		MOVL	FINAL_STATUS, R0	:	3466
				04	00161			RET		:	

; Routine Size: 354 bytes, Routine Base: CODE + 1152

```
: 2442 3467 1 GLOBAL ROUTINE DELETE_SQR_RECORD(SQR_N,SQR,DELETIONS): NOVALUE=
: 2443 3468
: 2444 3469 1 !++
: 2445 3470 1
: 2446 3471 1 FUNCTIONAL DESCRIPTION:
: 2447 3472 1 This routine deallocates a job file record and its substructures.
: 2448 3473 1
: 2449 3474 1 INPUT PARAMETERS:
: 2450 3475 1 SQR_N - Record number of SQR.
: 2451 3476 1 SQR - Pointer to SQR.
: 2452 3477 1 DELETIONS - (Optional) Pointer to list head for deletions.
: 2453 3478 1
: 2454 3479 1 IMPLICIT INPUTS:
: 2455 3480 1 NONE
: 2456 3481 1
: 2457 3482 1 OUTPUT PARAMETERS:
: 2458 3483 1 NONE
: 2459 3484 1
: 2460 3485 1 IMPLICIT OUTPUTS:
: 2461 3486 1 NONE
: 2462 3487 1
: 2463 3488 1 ROUTINE VALUE:
: 2464 3489 1 NONE
: 2465 3490 1
: 2466 3491 1 SIDE EFFECTS:
: 2467 3492 1 NONE
: 2468 3493 1
: 2469 3494 1 --
: 2470 3495 1
: 2471 3496 2 BEGIN
: 2472 3497 2 MAP
: 2473 3498 2 SQR: REF BBLOCK; ! Pointer to SQR
: 2474 3499 2 BUILTIN
: 2475 3500 2 ACTUALCOUNT;
: 2476 3501 2
: 2477 3502 2
: 2478 3503 2 ! Deallocate the fixed/variable data areas, if they exist.
: 2479 3504 2
: 2480 3505 2 DEALLOCATE VARIABLE DATA(
: 2481 3506 2 SQR$S_FILE_SETUP_MODULES,
: 2482 3507 2 SQR[SQR$T_FILE_SETUP_MODULES]);
: 2483 3508 2
: 2484 3509 2
: 2485 3510 2 ! Delete files as requested.
: 2486 3511 2
: 2487 3512 2 IF .SQR[SQR$V_DELETE_FILE]
: 2488 3513 2 AND CH$RCHAR(SQR[SQR$T_FILE_ID_DVI]) NEQ 0
: 2489 3514 2 AND ACTUALCOUNT() GEQU 3
: 2490 3515 2 THEN
: 2491 3516 3 BEGIN
: 2492 3517 3 SQR[SYMSL_LINK] = ..DELETIONS;
: 2493 3518 3 .DELETIONS = .SQR_N;
: 2494 3519 3 REWRITE_RECORD(.SQR_N);
: 2495 3520 3 END
: 2496 3521 2 ELSE
: 2497 3522 2 DEALLOCATE_RECORD(.SQR_N);
: 2498 3523 1 END;
```


			0004 00000	.ENTRY	DELETE_SQR_RECORD, Save R2	:	3467
	52	08	AC D0 00002	MOVL	SQR, R2	:	3507
		45	A2 9F 00006	PUSHAB	69(R2)	:	
			06 DD 00009	PUSHL	#6	:	
00000000G	EF		02 FB 0000B	CALLS	#2, DEALLOCATE_VARIABLE_DATA	:	
	1E	0C	A2 E9 00012	BLBC	12(R2), 1\$	:	3512
		1C	A2 95 00016	TS*B	28(R2)	:	3513
			19 13 00019	BEQL	1\$	:	
	03		6C 91 0001B	CMPB	(AP), #3	:	3514
			14 1F 0001E	BLSSU	1\$	:	
	62	0C	BC D0 00020	MOVL	@DELETIONS, (R2)	:	3517
	0C	04	AC D0 00024	MOVL	SQR_N, @DELETIONS	:	3518
		04	AC DD 00029	PUSHL	SQR_N	:	3519
00000000G	EF		01 FB 0002C	CALLS	#1, REWRITE_RECORD	:	
			04 00033	RET		:	3512
		04	AC DD 00034 1\$:	PUSHL	SQR_N	:	3522
00000000G	EF		01 FB 00037	CALLS	#1, DEALLOCATE_RECORD	:	
			04 0003E	RET		:	3523

; Routine Size: 63 bytes, Routine Base: CODE + 12B4

```

: 2500 3524 1 GLOBAL ROUTINE DELETE_FILES(SQR_N): NOVALUE=
: 2501 3525 1
: 2502 3526 1 ++
: 2503 3527 1
: 2504 3528 1 FUNCTIONAL DESCRIPTION:
: 2505 3529 1 This routine schedules the deletion of files in a completed job.
: 2506 3530 1
: 2507 3531 1 INPUT PARAMETERS:
: 2508 3532 1 SQR_N - Record number of first SQR in list.
: 2509 3533 1
: 2510 3534 1 IMPLICIT INPUTS:
: 2511 3535 1 NONE
: 2512 3536 1
: 2513 3537 1 OUTPUT PARAMETERS:
: 2514 3538 1 NONE
: 2515 3539 1
: 2516 3540 1 IMPLICIT OUTPUTS:
: 2517 3541 1 NONE
: 2518 3542 1
: 2519 3543 1 ROUTINE VALUE:
: 2520 3544 1 NONE
: 2521 3545 1
: 2522 3546 1 SIDE EFFECTS:
: 2523 3547 1 NONE
: 2524 3548 1
: 2525 3549 1 --
: 2526 3550 1
: 2527 3551 2 BEGIN
: 2528 3552 2 LOCAL
: 2529 3553 2 P: REF VECTOR; ! Pointer to file deletion context
: 2530 3554 2
: 2531 3555 2
: 2532 3556 2 ! Allocate the file deletion context area, initialize it with the list head,
: 2533 3557 2 ! and schedule execution of the file deletion routine.
: 2534 3558 2
: 2535 3559 2 P = ALLOCATE_MEMORY();
: 2536 3560 2 P[0] = SQR_N;
: 2537 3561 2 SCHEDULE_NONAST(DELETE_ONE_FILE, 4, P);
: 2538 3562 1 END;

```

			0000 0000	.ENTRY	DELETE_FILES, Save nothing	: 3524
	SE		04 C2 00002	SUBL2	#4, SP	: 3559
00000000G	EF		00 FB 00005	CALLS	#0, ALLOCATE_MEMORY	: 3560
	6E		50 D0 0000C	MOVL	R0, P	: 3561
00	BE	04	AC D0 0000F	MOVL	SQR_N, @P	: 3562
			5E DD 00014	PUSHL	SP	
			04 DD 00016	PUSHL	#4	
		0000V	CF 9F 00018	PUSHAB	DELETE ONE_FILE	
00000000G	EF		03 FB 0001C	CALLS	#3, SCHEDULE_NONAST	
			04 00023	RET		: 3562

; Routine Size: 36 bytes, Routine Base: CODE + 12F3

RECORDUTL
V04-000

Queue record manipulation utilities

M 2
16-Sep-1984 00:18:27
14-Sep-1984 12:37:12

VAX-11 B1155-32 V4.0-742
[JOBCTL.SRC]RECORDUTL.B32;1

Page 81
(17)

RES
V04

```

: 2540 3563 1 ROUTINE DELETE_ONE_FILE(REF_P): NOVALUE=
: 2541 3564 1
: 2542 3565 1 ++
: 2543 3566 1
: 2544 3567 1 FUNCTIONAL DESCRIPTION:
: 2545 3568 1 This routine deletes one file in a completed job.
: 2546 3569 1
: 2547 3570 1 INPUT PARAMETERS:
: 2548 3571 1 P - Pointer to pointer to file deletion context area.
: 2549 3572 1
: 2550 3573 1 IMPLICIT INPUTS:
: 2551 3574 1 NONE
: 2552 3575 1
: 2553 3576 1 OUTPUT PARAMETERS:
: 2554 3577 1 NONE
: 2555 3578 1
: 2556 3579 1 IMPLICIT OUTPUTS:
: 2557 3580 1 NONE
: 2558 3581 1
: 2559 3582 1 ROUTINE VALUE:
: 2560 3583 1 NONE
: 2561 3584 1
: 2562 3585 1 SIDE EFFECTS:
: 2563 3586 1 NONE
: 2564 3587 1
: 2565 3588 1 --
: 2566 3589 1
: 2567 3590 2 BEGIN
: 2568 3591 2 LOCAL
: 2569 3592 2 P: REF VECTOR; ! Pointer to file deletion context.
: 2570 3593 2
: 2571 3594 2
: 2572 3595 2 IF .QUEUE_FAB[FAB$W_IF1] NEQ 0
: 2573 3596 2 THEN
: 2574 3597 3 BEGIN
: 2575 3598 3
: 2576 3599 3 ! Point to file deletion context.
: 2577 3600 3
: 2578 3601 3 P = ..REF_P;
: 2579 3602 3
: 2580 3603 3
: 2581 3604 3 ! Lock the queue file.
: 2582 3605 3
: 2583 3606 3 LOCK_QUEUE_FILE();
: 2584 3607 3
: 2585 3608 3
: 2586 3609 3 ! Loop until a deletion is successfully submitted or no files remain.
: 2587 3610 3
: 2588 3611 3 WHILE TRUE DO
: 2589 3612 4 BEGIN
: 2590 3613 4 BIND
: 2591 3614 4 FIB = P[2] : BBLOCK[FIB$C_LENGTH]; ! FIB
: 2592 3615 4
: 2593 3616 4 LOCAL
: 2594 3617 4 SQR_N, ! Record number of SQR
: 2595 3618 4 SQR: REF BBLOCK, ! Pointer to SQR
: 2596 3619 4 DEVICE_DESC: VECTOR[2], ! Descriptor for device name
```

```
: 2597      3620  4      FIB_DESC:      VECTOR[2],      ! Descriptor for FIB
: 2598      3621  4      FNA_DESC:      VECTOR[2],      ! Descriptor for filename
: 2599      3622  4      ITMLST:      BBLOCK[28];      ! $FILESCAN item list
: 2600      3623  4
: 2601      3624  4
: 2602      3625  4      ! If no more files, exit the loop.
: 2603      3626  4      !
: 2604      3627  4      IF .P[0] EQL 0
: 2605      3628  4      THEN
: 2606      3629  5          BEGIN
: 2607      3630  5              DEALLOCATE_MEMORY(.P);
: 2608      3631  5              EXITLOOP;
: 2609      3632  4              END;
: 2610      3633  4
: 2611      3634  4
: 2612      3635  4      ! Read the SQR record and link to next.
: 2613      3636  4      !
: 2614      3637  4      SQR = READ_RECORD(SQR N = .P[0]);
: 2615      3638  4      P[0] = .SQR[SYMSL_LINR];
: 2616      3639  4
: 2617      3640  4
: 2618      3641  4      ! Extract the name, type, and version from the file specification.
: 2619      3642  4      !
: 2620      3643  4      FNA_DESC[0] = CH$RCHAR(SQR[SQR$T_FILE_SPECIFICATION]);
: 2621      3644  4      FNA_DESC[1] = SQR[SQR$T_FILE_SPECIFICATION] + 1;
: 2622      3645  4      ITMLST[0,0,16,0] = 0;
: 2623      3646  4      ITMLST[2,0,16,0] = FSCNS_NAME;
: 2624      3647  4      ITMLST[4,0,32,0] = 0;
: 2625      3648  4      ITMLST[8,0,16,0] = 0;
: 2626      3649  4      ITMLST[10,0,16,0] = FSCNS_TYPE;
: 2627      3650  4      ITMLST[12,0,32,0] = 0;
: 2628      3651  4      ITMLST[16,0,16,0] = 0;
: 2629      3652  4      ITMLST[18,0,16,0] = FSCNS_VERSION;
: 2630      3653  4      ITMLST[20,0,32,0] = 0;
: 2631      3654  4      ITMLST[24,0,32,0] = 0;
: 2632      3655  4      $FILESCAN(SRCSTR=FNA_DESC, VALUELST=ITMLST);
: 2633      3656  4
: 2634      3657  4
: 2635      3658  4      ! Initialize for the delete.
: 2636      3659  4      !
: 2637      3660  4      DEVICE_DESC[0] = CH$RCHAR(SQR[SQR$T_FILE_ID_DVI]);
: 2638      3661  4      DEVICE_DESC[1] = SQR[SQR$T_FILE_ID_DVI] + 1;
: 2639      3662  4      CH$FIL[0, FIB$C_LENGTH, FIB);
: 2640      3663  4      FIB[FIB$W_FID_NUM] = .SQR[SQR$W_FILE_ID_FID_NUM];
: 2641      3664  4      FIB[FIB$W_FID_SEQ] = .SQR[SQR$W_FILE_ID_FID_SEQ];
: 2642      3665  4      FIB[FIB$W_FID_RVN] = .SQR[SQR$W_FILE_ID_FID_RVN];
: 2643      3666  4      FIB[FIB$W_DID_NUM] = .SQR[SQR$W_FILE_ID_DID_NUM];
: 2644      3667  4      FIB[FIB$W_DID_SEQ] = .SQR[SQR$W_FILE_ID_DID_SEQ];
: 2645      3668  4      FIB[FIB$W_DID_RVN] = .SQR[SQR$W_FILE_ID_DID_RVN];
: 2646      3669  4      FIB_DESC[0] = FIB$C_LENGTH;
: 2647      3670  4      FIB_DESC[1] = FIB;
: 2648      3671  4      FNA_DESC[0] = .ITMLST[0,0,16,0] + .ITMLST[8,0,16,0] + .ITMLST[16,0,16,0];
: 2649      3672  4      FNA_DESC[1] = .ITMLST[4,0,32,0];
: 2650      3673  4
: 2651      3674  4
: 2652      3675  4      ! Assign a channel to the device, and issue the delete QIO. Te
: 2653      3676  4      ! cleanly if anything fails.
```

```
: 2654      3677  4      !
: 2655      3678  5      IF $ASSIGN(DEVNAM=DEVICE_DESC, CHAN=P[1])
: 2656      3679  4      THEN
: 2657      3680  4      IF $QIO(
: 2658      3681  4      FUNC=IOS_DELETE OR IOSM_DELETE,
: 2659      3682  4      CHAN=.P[1],
: 2660      3683  4      ASTADR=DELETE_ONE_FILE_AST,
: 2661      3684  4      ASTPRM=.P,
: 2662      3685  4      P1=FIB_DESC,
: 2663      3686  5      P2=FNA_DESC)
: 2664      3687  4      THEN
: 2665      3688  5      BEGIN
: 2666      3689  5      DEALLOCATE_RECORD(.SQR_N);
: 2667      3690  5      EXITLOOP;
: 2668      3691  5      END
: 2669      3692  4      ELSE
: 2670      3693  5      BEGIN
: 2671      3694  5      DEALLOCATE_RECORD(.SQR_N);
: 2672      3695  5      $DASSGN(CHAN=.P[1]);
: 2673      3696  5      END
: 2674      3697  4      ELSE
: 2675      3698  4      DEALLOCATE_RECORD(.SQR_N);
: 2676      3699  3      END;
: 2677      3700  3
: 2678      3701  3
: 2679      3702  3      ! Unlock the queue file.
: 2680      3703  3      !
: 2681      3704  3      UNLOCK_QUEUE_FILE();
: 2682      3705  2      END;
: 2683      3706  1      END;
```

```
.EXTRN S/$$ASSIGN, SY$$QIO
.EXTRN SY$$DASSGN

07FC 00000 DELETE_ONE_FILE:
      SA 00000000G EF 9E 00002 .WORD Save R2,R3,R4,R5,R6,R7,R8,R9,R10 : 3563
      SE 00000000' 34 C2 00009 MOVAB DEALLOCATE_RECORD, R10
      EF 85 0000C SUBL2 #52, SP
      01 12 00012 TSTW QUEUE_FAB+2 : 3595
      04 00014 BNEQ 1$
      56 04 BC D0 00015 1$: MOVL @REF P, P : 3601
      00000000G EF 00 FB 00019 CALLS #0, [O]CK_QUEUE_FILE : 3606
      58 08 A6 9E 00020 MOVAB 8(R6), R8 : 3614
      66 D5 00024 2$: TSTL (P) : 3627
      0C 12 00026 BNEQ 3$
      56 DD 00028 PUSHL P : 3630
      00000000G EF 01 FB 0002A CALLS #1, DEALLOCATE_MEMORY : 3629
      00E0 31 00031 BRW 7$ : 3637
      59 66 D0 00034 3$: MOVL (P), SQR_N
      59 DD 00037 PUSHL SQR_N
      00000000G EF 01 FB 00039 CALLS #1, READ_RECORD
      57 50 D0 00040 MOVL R0, SQR
      66 67 D0 00043 MOVL (SQR), (P) : 3638
      1C AE 4B A7 9A 00046 MOVZBL 75(SQR), FNA_DESC : 3643
```


20	AE	4C	A7	9E	0004B	MOVAB	76(R7), FNA_DESC+4	3644	
	6E	00060000	8F	D0	00050	MOVL	#393216, ITMLST	3645	
		04	AE	D4	00057	CLRL	ITMLST+4	3647	
08	AE	00070000	8F	D0	0005A	MOVL	#458752, ITMLST+8	3648	
		0C	AE	D4	00062	CLRL	ITMLST+12	3650	
10	AE	00080000	8F	D0	00065	MOVL	#524288, ITMLST+16	3651	
		14	AE	7C	0006D	CLRL	ITMLST+20	3653	
			7E	D4	00070	CLRL	-(SP)	3655	
		04	AE	9F	00072	PUSHAB	ITMLST		
		24	AE	9F	00075	PUSHAB	FNA_DESC		
00000000G	00		03	FB	00078	CALLS	#3, SYSS\$FILESCAN		
2C	AE	1C	A7	9A	0007F	MOVZBL	28(SQR), DEVICE_DESC	3660	
30	AE	1D	A7	9E	00084	MOVAB	29(R7), DEVICE_DESC+4	3661	
	6E		00	2C	00089	MOVCS	#0, (SP), #0, #64, (R8)	3662	
			68		00090				
04	A8	2C	A7	7D	0009	MOVQ	44(SQR), 4(R8)	3663	
0C	A8	34	A7	D0	00096	MOVL	52(SQR), 12(R8)	3667	
24	AE	40	8F	9A	0009B	MOVZBL	#64, FIB_DESC	3669	
28	AE		58	D0	000A0	MOVL	R8, FIB_DESC+4	3670	
	50		6E	3C	000A4	MOVZWL	ITMLST, R0	3671	
	51	08	AE	3C	000A7	MOVZWL	ITMLST+8, R1		
	50		51	C0	000AB	ADDL2	R1, R0		
	52	10	AE	3C	000AE	MOVZWL	ITMLST+16, R2		
1C AE	50		52	C1	000B2	ADDL3	R2, R0, FNA_DESC		
	20	AE	04	AE	D0	000B7	MOVL	ITMLST+4, FNA_DESC+4	3672
			7E	7C	000BC	CLRL	-(SP)	3678	
		04	A6	9F	000BE	PUSHAB	4(P)		
		38	AE	9F	000C1	PUSHAB	DEVICE_DESC		
00000000G	00		04	FB	000C4	CALLS	#4, SYSS\$ASSIGN		
	3E		50	E9	000CB	BLBC	R0, 5\$		
			7E	7C	000CE	CLRL	-(SP)	3686	
			7E	7C	000D0	CLRL	-(SP)		
		2C	AE	9F	000D2	PUSHAB	FNA_DESC		
		38	AE	9F	000D5	PUSHAB	FIB_DESC		
			56	DD	000D8	PUSHL	P		
		0000V	CF	9F	000DA	PUSHAB	DELETE_ONE_FILE_AST		
			7E	D4	000DE	CLRL	-(SP)		
	7E	0135	8F	3C	000E0	MOVZWL	#309, -(SP)		
		04	A6	DD	000E5	PUSHL	4(P)		
			7E	D4	000E8	CLRL	-(SP)		
00000000G	00		0C	FB	000EA	CALLS	#12, SYSS\$QIO		
	07		50	E9	000F1	BLBC	R0, 4\$		
			59	DD	000F4	PUSHL	SQR_N	3689	
	6A		01	FB	000F6	CALLS	#1, DEALLOCATE_RECORD		
			19	11	000F9	BRB	7\$	3688	
			59	DD	000FB	PUSHL	SQR_N	3694	
	6A		01	FB	000FD	CALLS	#1, DEALLOCATE_RECORD		
		04	A6	DD	00100	PUSHL	4(P)	3695	
00000000G	00		01	FB	00103	CALLS	#1, SYSS\$DASSGN		
			05	11	0010A	BRB	6\$	3680	
			59	DD	0010C	PUSHL	SQR_N	3698	
	6A		01	FB	0010E	CALLS	#1, DEALLOCATE_RECORD		
		FF10	31	00111	6\$:	BRW	2\$	3611	
00000000G	EF		00	FB	00114	CALLS	#0, UNLOCK_QUEUE_FILE	3704	
			04	0011B	7\$:	RET		3706	

; Routine Size: 284 bytes, Routine Base: CODE + 1317

RECORDUTL
V04-000

Queue record manipulation utilities

E 3
16-Sep-1984 00:18:27
14-Sep-1984 12:37:12

VAX-11 Bliss-32 V4.0-742
[JOBCTL.SRC]RECORDUTL.B32;1

Page 86
(18)

RES
V04


```
: 2685 3707 1 ROUTINE DELETE_ONE_FILE_AST(P): NOVALUE=
: 2686 3708 1
: 2687 3709 1 ++
: 2688 3710 1
: 2689 3711 1 FUNCTIONAL DESCRIPTION:
: 2690 3712 1 This routine is the completion AST for deletion of one file in a
: 2691 3713 1 completed job. It reschedules execution of the file deletion routine,
: 2692 3714 1 or terminates the operation if it is complete.
: 2693 3715 1
: 2694 3716 1 INPUT PARAMETERS:
: 2695 3717 1 P - Pointer to file deletion context area.
: 2696 3718 1
: 2697 3719 1 IMPLICIT INPUTS:
: 2698 3720 1 NONE
: 2699 3721 1
: 2700 3722 1 OUTPUT PARAMETERS:
: 2701 3723 1 NONE
: 2702 3724 1
: 2703 3725 1 IMPLICIT OUTPUTS:
: 2704 3726 1 NONE
: 2705 3727 1
: 2706 3728 1 ROUTINE VALUE:
: 2707 3729 1 NONE
: 2708 3730 1
: 2709 3731 1 SIDE EFFECTS:
: 2710 3732 1 NONE
: 2711 3733 1
: 2712 3734 1 --
: 2713 3735 1
: 2714 3736 2 BEGIN
: 2715 3737 2 MAP
: 2716 3738 2 P: REF VECTOR; ! Pointer to file deletion context.
: 2717 3739 2
: 2718 3740 2
: 2719 3741 2 ! Deassign the channel, and reschedule the file deletion routine. If the
: 2720 3742 2 operation is complete, release the file context area.
: 2721 3743 2
: 2722 3744 2 $DASSGN(CHAN=.P[1]);
: 2723 3745 2 .P[0] EQL 0
: 2724 3746 2 THEN DEALLOCATE MEMORY(.P)
: 2725 3747 2 ELSE SCHEDULE_NONAST(DELETE_ONE_FILE, 4, P);
: 2726 3748 1 END;
```

```
0004 00000 DELETE_ONE_FILE_AST:
      52      04 AC D0 00002      .WORD Save R2
      00000000G 00      04 A2 DD 00006      MOVL P, R2
      01 FB 00009      PUSH 4(R2)
      62 D5 00010      CALLS #1, SYSSDASSGN
      0A 12 00012      TSTL (R2)
      52 DD 00014      BNEQ 1$
      00000000G EF      01 FB 00016      PUSH R2
      04 0001D      CALLS #1, DEALLOCATE_MEMORY
      RET
```

```
: 3707
: 3744
:
: 3745
: 3746
:
:
```

RECORDUTL
V04-000

Queue record manipulation utilities

G 3
14-Sep-1984 00:18:27
14-Sep-1984 12:37:12

VAX-11 Bliss-32 V4.0-742
[JOBCTL.SRC]RECORDUTL.B32;1

Page 88
(19)

RES
V04

	04	AC	9F	0001E	18:	PUSHAB	P	
		04	DD	0001		PUSHL	#4	
		CF	9F	00023		PUSHAB	DELETE ONE FILE	
00000000G	EF	03	FB	00027		CALLS	#3, SCHEDULE_NONAST	
		04	0002E			RET		

; 3747
;
;
;
;
; 3748

; Routine Size: 47 bytes, Routine Base: CODE + 1433

; R

RECORDUTL
V04-000

Queue record manipulation utilities

H 3
16-Sep-1984 00:18:27
14-Sep-1984 12:37:12

VAX-11 Bliss-32 V4.0-742
[JOBCTL.SRC]RECORDUTL.B32;1

Page 89
(20)

RESI
V04.

: 2728 3749 1 END
: 2729 3750 0 ELUDOM

PSECT SUMMARY

Name	Bytes	Attributes
COMMON	5024	NOVEC, WRT, RD, NOEXE, NOSHR, LCL, REL, OVR, NOPIC, ALIGN(2)
CODE	5218	NOVEC, NOWRT, RD, EXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	195	1	1000	00:01.4

COMMAND QUALIFIERS

: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:RECORDUTL/OBJ=OBJ\$:RECORDUTL MSRC\$:RECORDUTL/UPDATE=(ENH\$:RECORDUTL)

: Size: 5218 code + 5024 data bytes
: Run Time: 01:08.5
: Elapsed Time: 04:16.2
: Lines/CPU Min: 3283
: Lexemes/CPU-Min: 34849
: Memory Used: 588 pages
: Compilation Complete

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	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

0194 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

RESTRICT
LIS

RESPONSE
LIS

SNOOZE
LIS

SCHEDULER
LIS