

ADDRESS LENGTH

BINARY CONTROL CARDS.

0 751
751

IDENT SUBRS
END

BLOCKS	TYPE	ADDRESS	LENGTH
PROGRAM*	LOCAL	0	745
AUTHS	LOCAL	745	4

ENTRY POINTS.

CANON	-	0	GETSPAC	-	153	DSTRY	-	214	HASH1	-	602
NAMSCAN	-	22	ADDWDS	-	157	NXTOWN	-	461	DEHASH	-	616
BEA	-	66	DELWDS	-	165	FOBSPEC	-	524	CLSDYNT	-	650
UNIQUE	-	143	COPY	-	174	NOBSPEC	-	527	SKPENT	-	675
CLOCKS	-	734	CPYNAME	-	177	HASH	-	601			

EXTERNAL SYMBOLS.

WINDOW	E.DUP	P.DYNT	EC.FILE	EC.ABLK	RETBUFF	EC:CIN	DE:CLO
CLKBUFF	E.NOSPC	SYSERR	CC.DIR	EN.RESV	L.DNT	DE:DSF	DE:DFIL
O.SYSTM	P.NAME	CL.DNT	M.WIN	C.T1	O.LOL	EC:IDSCP	DE:XOPN
O.SWPTM	P.KEY	CA.SBPD	EC.DRCT	C.T2	GETDNT	DE:IABSV	ERRLOC
O.USRTM	P.DIRECT	CA.DSKF	EN.OWNS	CL.DRCT	PUTDNT	DE:ZRM	IPLIST
E.BADNM	OPEN	P.ERRCL	EN.NODF	EF.DRCT	DSPCLKS	DE:CMRO	EC:CCAP
E.IMPOT	FLAPPER	P.ERRNO	EN.RSRV	RETDATA	EC:COU	DE:READ	CL:OPS

IDENT	SUBRS
ENTRY	CANON, NAMSCAN, BEA, UNIQUE, CLOCKS
ENTRY	GETSPAC, ADDWDS, DELWDS
EXT	WINDOW, CLKBUFF, O.SYSTM, O.SWPTM, O.USRTM
EXT	E.BADNM, E.IMPOT, E.DUP, E.NOSPC
EXT	P.NAME, P.KEY, P.DIRECT, OPEN, FLAPPER
EXT	P.DYNT, SYSERR
EXT	CL.DNT, CA.SBPD

0	MISCMAC	XTEXT	.. LINKAGE MACROS, ETC.
0	SYSMAC	XTEXT	.. SYSTEM CALLING MACROS
0	DIRDEFS	XTEXT	.. FIELD DEFINITIONS, TYPES, ETC.

* CANON SETS THE UPPER 4 BITS OF EACH WORD OF THE NAME BLOCK POINTED BY
 * B1 TO ZERO; IT SETS THE END-OF-NAME FLAG (=SIGN BIT) OF THE LAST NON-
 * BLANK WORD; AND IT DECREASES THE BLOCK HEADER WORD COUNT BY THE
 * NUMBER OF WORDS OF TRAILING BLANKS. IT EXITS TO E.BADNM IF THE
 * NAME IS EMPTY OR IF THERE ARE NON-GRAPHICS (OR NON-TRAILING BLANKS.)

* B1 - ADDRESS OF NAME-BLOCK HEADER
 * CALLR CANON,B6
 *

* SAVES: B1

0	5121000000		CANON	SA2	B1+0	.. BLOCK DATA HEADER [CONTAINS COUNT]
	10622			BX6	X2	
	43004			MX0	4	
1	0302000000 x		CAN.TRIM	ZR	X2,E.BADNM	.. ERROR - EMPTY NAME
	53321			SA3	B1+X2	
	15330			BX3	-X0*X3	.. CLEAR UPPER 4 BITS
2	7222777776			SX2	X2-1	
	0313000004 +			NZ	X2,CAN.LWD	.. FOUND THE LAST NON-BLANK WORD
3	7266777776			SX6	X2-1	.. COUNT DOWN THE TRUE LENGTH
	0200000001 +			JP	CAN.TRIM	
4	56610		CAN.LWD	SA6	B1	.. [POSSIBLY] UPDATED BLOCK HEADER
	43601			MX6	1	
	43773			MX7	59	.. =-1
	12663			BX6	X6+X3	.. SET END-OF-NAME FLAG
5	54630		CAN.LOOP	SA6	A3	
	0302000007 +			ZR	X2,GOODNAME	.. NOW DO THE SYNTAX CHECK
	53321			SA3	B1+X2	
6	36227			IX2	X2+X7	
	15630			BX6	-X0*X3	
	0200000005 +			JP	CAN.LOOP	
7	7100000177		GOODNAME	SX0	177B	.. ONE-CHARACTER MASK
	5121000001			SA2	B1+1	.. INITIALIZE A2, X2 TO FIRST NAME WORD
10	6120000070			SB2	8*7	.. INITIALIZE CHARACTER SHIFT COUNT
	6150000015 +			SB5	GN.ST1	.. INITIALIZE STATE = 1 = IN THE NAME
11	0420000013 +		GN.GETC	ZR	B2,GN.GETW	.. GET THE NEXT WORD OR QUIT
	6122777770			SB2	B2-7	
12	23322		GN.SHIFT	AX3	B2,X2	.. RIGHT-JUSTIFY THE NEXT CHARACTER
	11303			BX3	X0*X3	
	0250000000			JP	B5	.. JUMP ON STATE
13	0332000021 +		GN.GETW	NG	X2,GN.RET	.. NO MORE WORDS, NAME IS GOOD
	5022000001			SA2	A2+1	
14	6120000061			SB2	7*7	.. RESET SHIFT COUNT
	0200000012 +			JP	GN.SHIFT	
15	0303000017 +		GN.ST1	ZR	X2,GN.ET2	.. A BLANK => ENTER STATE 2
	6233777640			SB3	X2-137B	
16	0730000011 +			LT	B3,B0,GN.GETC	.. GRAPHIC
	0200000000 x			JP	E.BADNM	.. NONGRAPHIC
17	6150000020 +		GN.ET2	SB5	GN.ST2	.. CHANGE THE STATE ...

CANON - PUT A NAME IN CANONICAL FORM AND CHECK ITS SYNTAX

	0200000011 +		JP	GN.GETC	 AND KEEP SCANNING
20	0303000011 +	GN.ST2	ZR	X1,GN.GETC	.. TRAILING BLANKS ARE OK
	0200000000 x		JP	E.BADNM	.. ERROR - EMBEDDED BLANK
21	0260000000	GN.RET	JP	B4	.. NORMAL RETURN

* ↑NAMSCAN↑ SEARCHES FOR A DIRECTORY ENTRY WHOSE NAME IS THE SAME AS THAT
* POINTED TO BY B1. IT RETURNS THE ENTRY TYPE IN X0 AND THE
* STARTING FILE ADDRESS IN B1. ↑NAMSCAN↑ FAILS IF THE NAME IS NOT FOUND.

* B1↑NAME BLOCK PTR
* CALL NAMSCAN,ONFAILURE,B6
*

* SAVES: X4

22	43302	NAMSCAN	MX3	W.ENTTYP	.. ENTRY TYPE IS
	20373		LX3	60-S.ENTTYP	.. IN FIRST WORD OF NAME
23	5011000001	NS.NXT1	GETWD	NS.FAIL	.. FIRST WORD OF AN ENTRY
25	6120000000 x	NS.NXT2	GETADDR	B2	.. CALCULATE FILE ADDRESS OF THIS ENTRY
	11031		BX0	X3-X1	
26	20001		LX0	S.ENTTYP	.. ENTRY TYPE, LJZF
	15113		BX1	-X3-X1	.. CLEAR TYPE FROM FIRST WORD
	5121000001		SA2	B1+1	.. FIRST WORD OF TARGET NAME
27	13612	NS.LOOP	BX6	X1-X2	
	0316000035 +		NZ	X4-NS.DIFF	.. MISMATCH, MOVE TO NEXT ENTRY
30	0331000052 +		NG	X1-NS.MATCH	.. END-OF-NAME FLAG IS ON
	5022000001		SA2	A2+1	.. NEXT TARGET WORD
31	5011000001		GETWD	NSERR1	.. NEXT ENTRY WORD
	0200000027 +		JP	NS.LOOP	
33	5011000001	NS.SKPNM	GETWD	NSERR1	
35	0321000033 +	NS.DIFF	PL	X1-NS.SKPNM	.. NOT YET TO END OF ENTRY NAME
	5011000001		GETWD	NSERR2	.. FIRST WORD OF OBJECT PART
37	5011000001		GETWD	NSERR2	.. SECOND
41	5011000001	NS.ACCI	GETWD	NS.FAIL0	
	20103		LX1	S.APF	.. LEFT-JUSTIFY ACCESS-PAIR FLAG
43	0321000046 +		PL	X1-NS.SKPSF	.. NOT A PAIR
	5011000001		GETWD	NSERR3	.. SECOND WORD OF ACCESS-PAIR
45	0200000041 +		JP	NS.ACCL	.. SKIP REST OF ACCESS-LIST
46	20171	NS.SKPSF	LX1	60-S.APF	.. REPOSITION CURRENT DIRECTORY WORD
	0310000025 +		NZ	X0-NS.NXT2	.. NOT A SOFT-LINK, START NEW ENTRY
47	0331000023 +	NS.SFTL	NG	X1-NS.NXT1	.. LAST-WORD-OF-NAME FLAG
	5011000001		GETWD	NSERR4	
51	0200000047 +		JP	NS.SFTL	.. SKIP REST OF NAME
52	66120	NS.MATCH	B1	B3	.. SAVED STARTING FILE ADDRESS TO B1
	0260000001		JP	B4+1	.. NORMAL RETURN
53	0310000021 +	NS.FAIL0	NZ	X0-NS.FAIL	.. NOT A SOFTLINK, AND NO MORE ENTRIES
54	0100000000 x	NSERR4B	SYSERR		.. END OF DIRECTORY INSIDE SOFTLINK
	21 +	NS.FAIL	EQU	GN.RET	.. FAILURE RETURN
56	0100000000 x	NSERR1	SYSERR		.. UNEXPECTED END OF DIRECTORY
		*			.. DURING ENTRY NAME
60	0100000000 x	NSERR2	SYSERR		.. EOD IN OBJECT PART
62	0100000000 x	NSERR3	SYSERR		.. EOD IN ACCESS-PAIR
64	0100000000 x	NSERR4	SYSERR		.. EOD IN SOFTLINK NAME

```

*      X4 = A SET OF OPTIONS [LEFT-JUSTIFIED, ZERO-FILLED]
*      P.KEY = ACCESS-KEY-NUMBER OF C.KEY
*      CALL BEA,ONFAILURE
*
* RETURNS :      X0 = TYPE OF ENTRY [LUZF]
*                X4 = OPTIONS [LUZF]
*                B1 = FILE ADDRESS OF FIRST WORD
*
* AND IF HARD,   B2 = OBJECT TYPE INDEX [0 THRU 5]
*                X3 = OBJECT-DESCRIPTOR
* AND IF SOFT,   P.DIRCT = DIRECTORY-SPECIFIER
*
*                P.KEY = ACCESS-KEY-SPECIFIER
*                P.NAME = NAME

```

66	6110000000 X	BEA	SB1	P.NAME	.. PARAMETER FOR +NAMSCAN+
	6160000070 +		CALL	NAMSCAN,B7,B6	.. RETURNS ENTRY TYPE IN X0
					AND A1 AT LAST NAME WORD
					[FAIL IF +NAMSCAN+ FAILS]
71	43252		MX2	42	
	0300000077 +		ZR	X0,BEA.SOFT	.. THIS IS A SOFT-LINK ENTRY
72	5011000001		GETWD	BEAERR1	
	11221		BX2	X2*X1	.. IMPLICIT-OPTIONS
	63210		SB2	X1	.. OBJECT TYPE INDEX
74	5011000001		GETWD	BEAERR1	
	10311		BX3	X1	.. OBJECT-SPECIFIER
76	5150000000 X		SA5	P.KEY	
	0200000104 +		JP	BEA.UNL	.. NOW UNLOCK THE ENTRY
77	5011000001	BEA.SOFT	GETWD	BEAERR1	
	10611		BX6	X1	
101	5160000000 X		SA6	P.DIRCT	.. DIRECTORY-SPECIFIER
	5011000001		GETWD	BEAERR1	
103	5150000000 X		SA5	P.KEY	
	10611		BX6	X1	
	54650		SA6	A5	.. ACCESS-KEY-SPECIFIER
104	0305000113 +	BEA.UNL	ZR	X5,BEA.IMPL	.. IMPLICIT ACCESS [ZERO IS NULL KEY]
105	5011000001	BEA.EXPL	GETWD	E.IMPOT	
	20103		LX1	S.APF	.. LEFT-JUSTIFY ACCESS-PAIR FLAG
107	0321000000 X		PL	X1,E.IMPOT	.. NO ACCESS PERMITTED [END OF LIST]
	10211		BX2	X1	.. OPTIONS -> X2
110	5011000001		GETWD	BEAERR2	.. GET SECOND WORD OF PAIR
	13115		BX1	X1-X5	.. COMPARE WITH SEARCH KEY
112	0311000105 +		NZ	X1,BEA.EXPL	.. NOT THE ONE
	20217		LX2	S.OPTS-S.APF	.. LEFT-JUSTIFY THE OPTIONS
113	11442	BEA.IMPL	BX4	X4*X2	.. COMBINE OPTIONS WITH INPUT OPTIONS
114	5011000001	BEA.SKAL	GETWD	BEA.END0	.. FIRST WORD OF ACCESS-PAIR, OR...
	20103		LX1	S.APF	.. LEFT-JUSTIFY ACCESS-PAIR FLAG
116	0321000124 +		PL	X1,BEA.END1	.. NOT AN ACCESS-PAIR
	5011000001		GETWD	BEAERR2	.. SKIP 2 ND WORD OF PAIR
120	0200000114 +		JP	BEA.SKAL	.. SKIP REST OF ACCESS-LIST

121	0310000134 +	BEA.END0 NZ	X0, BEA.RET	.. NOT A SOFTLINK - END OF ENTRY
122	0100000000 x	BEAERR4 SYSERR		.. END OF DIRECTORY INSIDE SOFTLINK
124	0330000134 +	BEA.END1 NG	X0, BEA.RET	.. DONE IF NOT A SOFTLINK
	20171	LX1	60-S.APF	.. REPOSITION CURRENT WORD
125	6120000001 x	SB2	P.NAME+1	
126	10611	BEA.MVNM BX6	X1	
	56620	SA6	B2	
	6122000001	SB2	B2+1	
127	0331000132 +	NG	X1, BEA.RET0	.. END OF NAME
	5011000001	GETWD	BFAERR3	.. NEXT WORD OF NAME2
131	0200000126 +	JP	BEA.MVNM	.. COPY REST OF NAME
132	5011000001	BEA.RET0 GETWD	BEA.RET	.. PICK UP NEXT WORD
134	0270000001	BEA.RET JP	B7+1	.. NORMAL RETURN
135	0100000000 x	BEAERR1 SYSERR		.. END OF DIRECTORY IN OBJECT PART
137	0100000000 x	BEAERR2 SYSERR		.. EOD IN ACCESS PAIR
141	0100000000 x	BEAERR3 SYSERR		.. EOD IN SOFT NAME

* ↑UNIQUE↑ CHECKS THAT P.NAME IS A LEGAL NAME; IT OPENS THE DIRECTORY
 * FILE [WRITE ENABLE] AND SCANS IT FOR THIS NAME - IF THE NAME IS
 * NOT ALREADY IN THE DIRECTORY, IT RETURNS.

*
 * CALL UNIQUE

* ERROR EXITS: E.BADNAM [THRU ↑CANON↑], OPENING ERRORS, E.DUP

143	6110000000 x	UNIQUE	SB1	P.NAME		
	6160000145 +		CALL	CANON,,B6	.. PUT NAME IN CANONICAL FORM	
145	7110000001		SX1	1	.. OPEN DIRECTORY FILE	
	6160000147 +		CALL	OPEN,,B6	.. FOR WRITING	
147	6110000000 x		SB1	P.NAME		
	6160000151 +		CALL	NAMSCAN,B7,B6	.. RETURN IF NO EXISTING OCCURRENCE	
152	0200000000 x		JP	E.DUP	.. ERROR	


```

*
*   RESERVE SPACE IN A DIRECTORY
*
*   X1 = N, NUMBER OF WORDS TO RESERVE
*   CALL GETSPAC
*
*   SAVES: B1->R7, X1
153 5120000000 X      GETSPAC SA2      WINDOW      .. BLOCK SIZE, FREE POINTER
      10322          73330      BX3      X2
      21222          73220      SX3      X3      .. FREE POINTER IS DIRDSR[43:60]
154 36431            37224      IX4      X3+X1      .. F.P+N
      21222          73220      AX2      18      .. RIGHT JUSTIFY BLOCK SIZE
      73220          37224      SX2      X2
      37224          37224      IX2      X2-X4
155 0332000000 X      NG      X2-E.NOSPC      .. ERROR - F.P+N>B.S [=MAX.LENGTH]
156 0270000000      GS.RET JP      B7      .. SPACE EXISTS, RETURN

*
*   MAKE A GAP IN A DIRECTORY FILE
*
*   X1 -> WHERE
*   X2 -> NUMBER
*   CALL ADDWDS
      [X1 AND X2 ARE NOT
      CHANGED BY ADDWDS]
157 5140000000 X      ADDWDS SA4      WINDOW
      36642          54640      IX6      X4+X2      .. UPDATE FREE POINTER FIELD IN
      54640          54640      SA6      A4      .. THE DIRECTORY DESCRIPTOR WORD
160 73340          37031      SX3      X4      .. FREE POINTER IS DIRDSR[43:60]
      37031          0300000164 *      IX0      X3-X1      .. WORD COUNT := F.P-WHERE
      0300000164 *      ZR      X0,AW.RET      .. NOTHING TO MOVE
161 6211777776 X      SB1      WINDOW-I+X1      .. SOURCE BASE REGISTER = WHERE - I
      63221          43573      SB2      B1+X2      .. SINK BASE REGISTER = SOURCE + N
      43573          43573      MX5      50      .. =-I
162 53301          10633      AW.LOOP SA3      B1+X0      .. MOVE A WORD
      10633          53662      BX6      X3      .. FROM ADDRESS I
      53662          36005      SA6      B2+X0      .. TO ADDRESS I+N
      36005          36005      IX0      X0+X5      .. REDUCE I BY ONE
163 0310000162 *      NZ      X0,AW.LOOP      .. REPEAT UNTIL I<WHERE [5 MICR/WORD]
164 0270000000      AW.RET JP      B7      .. RETURN

*
*   CLOSE A GAP IN A DIRECTORY FILE
*
*   X1 -> WHERE
*   X2 -> NUMBER
*   CALL DELWDS
*
*   SAVES: X1, X2
165 5150000000 X      DELWDS SA5      WINDOW      .. UPDATE FREE POINTER
      37752          54750      IX7      X5-X2
      54750          54750      SA7      A5

```

```
166 63150          SB1      X5          .. NUMBER OF WDS PREVIOUSLY IN USE
      36312          IX3      X1+X2      .. FWA OF AREA FOLLOWING GAP
      63330          SB3      X3
      67313          SB3      B1-B3      .. NUMBER OF WORDS TO SLIDE
167 0430000173 +    ZR        B3-DW.RET .. NONE ...
      6242000000    SB4      X2+0
170 6111000000 X    SB1      WINDOW+B1 .. MAKE B1 AN ABSOLUTE CM ADDRESS
      6150000001    SB5      1
171 57313          DW.LOOP SA3      B1-B3
      10633          BX6      X3
      55624          SA6      A3-B4
      67335          SB3      B3-B5
172 0530000171 +    NZ        B3-DW.LOOP
173 0270000000    DW.RET    JP        B7          .. RETURN
*
* COPY WORDS (INTO DIRECTORY FILE)
*
* B1 -> FROM
* B2 -> TO
* B3 -> NUMBER OF WORDS TO COPY (FROM AND TO AREAS SHOULD BE DISJOINT)
* CALL COPY
*
* CHANGES: B3, A1-X1, A6-X6
*
ENTRY COPY
174 6133777776    COPY     SB3      B3-1          .. +COPY+ ASSUMES NUMBER>=1
      56113          SA1      B1+B3
      10611          BX6      X1
175 56623          SA6      B2+B3
      0530000174 +    NZ        B3-COPY
176 0270000000    JP        B7          .. RETURN
```

* A1 → FIRST WORD OF PRINTNAME
 * B1 → ABSOLUTE ADDRESS OF RESULT-BUFFER
 * CALL CPYNAME

* ↑CPYNAME↑ COPIES THE PRINTNAME CONSISTING OF EIGHT-CHARACTER WORD,
 * THE LEFTMOST 4 BITS OF EACH FILLED WITH ZEROS, INTO THE FIRST
 * WORD OF THE BUFFER IS PLACED THE LENGTH OF THE PRINTNAME, IN
 * CHARACTERS. IN B1 IS RETURNED THE NUMBER OF WORDS PLACED INTO THE
 * BUFFER (INCLUDING THE COUNT WORD.)

		ENTRY	CPYNAME	
177	6121000001	CPYNAME SB2	B1+1	.. ADDRESS FOR FIRST CHARACTER-WORD
	7170000010	SX7	8	.. INITIAL CHARACTER COUNT (SEE NOTE BELOW AT =CPYNAME3=)
200	43002	MX0	W.ENTTYP	.. MAKE MASK
	20073	LX0	60-S.ENTTYP	.. FOR ENTRY TYPE FIELD
	15110	BX1	-X0*X1	.. REMOVE ENTRY TYPE FROM FIRST NAME WORD
201	0331000205 +	CPYNAME1 NG	X1,CPYNAME2	.. THIS IS THE LAST NAME WORD
	7277000010	SX7	X7+8	.. 1 FULL WORD = 8 CHARACTERS
202	10611	BX6	X1	
	56620	SA6	B2	.. STORE INTO BUFFER
	6122000001	SB2	B2+1	.. ADVANCE BUFFER POINTER
203	5011000001	GETWD	CPYNAME1	.. (SYSERR IF NO MORE WORDS)
	0200000201 +	JP	CPYNAME1	.. COPY THE REST
205	43001	CPYNAME2 MX0	1	.. REMOVE FLAG
	15610	LX0	60-S.EONF	.. SYMBOL = 0
	56620	BX6	-X0*X1	.. FROM LAST WORD
	43065	SA6	B2	
		MX0	60-7	.. (COMPLEMENT OF) ONE-CHARACTER MASK
* NOW IT IS TIME TO ADJUST THE CHARACTER COUNT, IN X7, SINCE IT MAY * BE AS MUCH AS 7 TOO LARGE.				
206	15210	CPYNAME3 BX2	-X0*X1	.. LOOK FOR TRAILING BLANK
	21107	AX1	7	
	0312000210 +	NZ	X2,CPYNAME4	.. THERE ARE NO MORE
207	7277777776	SX7	X7-1	.. DECREMENT CHARACTER COUNT
	0200000206 +	JP	CPYNAME3	
210	56710	CPYNAME4 SA7	B1	.. STORE CHARACTER COUNT
	67121	SB1	B2-B1	
	6111000001	SB1	B1+1	.. NUMBER OF WORDS IN BUFFER
211	0270000000	JP	B7	.. RETURN
212	0100000000 x	CPYNAME4 SYSERR		.. UNEXPECTED END OF DIRECTORY

* B2 - OBJECT TYPE INDEX, HOPEFULLY 0, 1, OR 2
* X3 = OBJECT DESCRIPTOR: DISK ADDRESS, UNIQUE NAME
* CALL DSTRY, ON FAILURE FAILS (WITH APPROPRIATE ERROR CLASS,
* NUMBER IN X1, X2) IF CAN'T DESTROY
* OTHERWISE, RETURNS WITH B6
* ZERO/NOTZERO AS IT DIDNT/DID
* CLOBBER M.WIN, THE WINDOW MAP ENTRY
*
* SAVES: B1, B3

ENTRY DSTRY

EXT CA.DSKF, P.ERRCL, P.ERRNO, EC.FILE, CC.DIR, M.WIN
EXT EC.DRCT, EN.OWNS, EN.NODF, EN.RSRV, EC.ABLK, EN.RESV
EXT C.T1, C.T2, CL.DRCT, EF.DRCT
EXT RETDATA, RETBUFF

214 0720000451 * DSTRY NG B2, DSERR0 ** NOT A LEGAL TYPE **
6140000000 X SB4 CA.SBPD
215 0742000451 * GT B2, B4, DSERR0 ** NOT ALLOWED TO DESTROY **
7160000000 X CRCAP C.T1, (CL.DRCT, CA.DSKF), (X3) ** CREATE FILE CAP
226 6160000000 SB6 0 ** INITIALIZE -GOOD M.WIN- FLAG
7160000250 + DS.ERR1 ** FILE OPENING ERROR PROCESSOR
7160000000 X XOPEN C.T1, DS.ERR0, C.T2 ** EXCLUSIVE OPEN
235 13666 OFF.ERR
236 0520000265 + NE B2, B0, DSTRYDIR ** OBJECT IS DIRECT, OR SUSP. DESCR
7160000256 + ON.ERR DS.ERR2 ** FILE DESTRUCTION ERROR PROCESSOR
7160000000 X DFIL C.T1 ** ATTEMPT TO DESTROY FILE
244 13666 OFF.ERR ** IT WORKED
245 0270000001 RETDSTR JP B7+1 ** NORMAL RETURN
246 7110000000 X DS.ERR0 SX1 EC.DRCT ** DIRECTORY TO BE
7120000000 X SX2 EN.DSKF ** DESTROYED IS (ALREADY)
247 0270000000 JP B7 ** EXCLUSIVELY OPEN

* OPEN ERROR PROCESSOR: IF OBJECT IS GONE, JUST RETURN (NORMALLY)

250 5110000000 X DS.ERR1 SA1 P.ERRCL
5120000000 X SA2 P.ERRNO
251 7130000000 X SX3 EC.FILE
37413 IX4 X1-X3
252 0314000255 + NZ X4, DS.ERR1A ** NOT A FILE ERROR CLASS, FRETURN WITH IT
73220 SX2 X2
253 7130000000 X SX3 EN.NODF
37423 IX4 X2-X3
254 0304000245 + ZR X4, RETDSTR ** FILE GONE, RETURN NORMALLY
255 0270000000 DS.ERR1A JP B7 ** FRETURN WITH ERROR CLASS, NUMBER

* FILE DESTRUCTION ERROR PROCESSOR

256 5110000000 X DS.ERR2 SA1 P.ERRCL
5120000000 X SA2 P.ERRNO
257 7160000000 X CLOSE C.T2 ** CLOSE THE FILE

264 0270000000

JP

B7

.. FRETURN

* BEFORE WE CAN DESTROY A DIRECTORY, IT MUST CONTAIN NO OWNERSHIP ENTRIES
* AND ITS DIRECTLY ASSOCIATED ACCOUNTING BLOCK (IF IT HAS
* ONE) MUST BE EMPTY (AND FUNDING NO PROCESSES).

265 7160000000 x

DSTRYDIR READ

C.T2,0,RETDATA,1,DSERR1 .. SHOULD NOT FRETURN

276 5110000000 x

SA1

RETDATA

277 7160000000 x

LX1

60-18

.. RIGHT JUSTIFY DIRECTORY SIZE

313 6160000001

CHGMAPRO

(CL.DRCT,CC.DIR),M.WIN,C.T2,0,WINDOW,X1,EF.DRCT

314 6241000000 x

SB6

1

.. SIGNAL THAT M.WIN IS CLOBBERED

315 5011000002

SA1

WINDOW

316 5011000001

SB4

X1+WINDOW

320 10211

PL

X1,DSDR1

321 6160000322 +

DSDR1

GETWD

DSDR2

.. SKIP THE SUCCESSOR LINK

322 0200000320 +

DSDR15

BX2

X1

.. GET FIRST WORD OF FIRST ENTRY

323 10211

LX2

2

.. ENTRY TYPE IS BITS 2,3. BIT 3 MEANS

324 5140000000 x

NG

X2,DSDR.E1

.. THIS IS AN OWNERSHIP ENTRY, CANNOT

325 10544

CALLFR

SKPENT,DSDR2,B6 .. SKIP TO NEXT ENTRY OR FRETURN

326 20202

JP

DSDR15

.. DESTROY CONTAINING DIRECTORY

327 0332000354 +

*

* NOW, IF THE DIRECTORY HAS ITS OWN ACCOUNTING BLOCK WE TRY TO DESTROY IT.

328 7160000000 x

DSDR2

SA4

WINDOW

.. DESCR[6:6] IS THE HAS-OWN-ACCOUNTING-

329 10544

ZEROMAP

(CL.DRCT,CC.DIR),M.WIN,C.T2

BLOCK FLAG

330 20405

BX5

X4

.. SAVE FIRST WORD

331 21451

LX4

5

.. NO ACCOUNTING BLOCK TO DESTROY

332 7160000373 +

PL

X4,DSDR3

.. RIGHT JUSTIFY ACC.PTR

333 7160000000 x

AX4

5+18+18

.. DESTROY ACC. BLOCK

334 13666

ON.ERR

DS.ERR3

.. DESTROY DIRECTORY FILE

335 7160000000 x

ABSV

ARSV,DES,X4,*,* .. NORMAL RETURN

336 7160000000 x

OFF.ERR

C.T1

.. DESTROY DIRECTORY FILE

337 7160000000 x

DFIL

B7+1

.. NORMAL RETURN

338 0270000001

JP

B7+1

.. NORMAL RETURN

339 7160000000 x

DSDR3

JP

B7+1

.. NORMAL RETURN

340 7160000000 x

DSDR.E1

ZEROMAP

(CL.DRCT,CC.DIR),M.WIN,C.T2

.. DESTROY DIRECTORY FILE

341 7160000000 x

CLOSE

C.T2

.. NORMAL RETURN

342 7110000000 x

SX1

EC.DRCT

.. DIRECTORY OWNS SOMETHING

343 7120000000 x

SX2

EN.OWNS

.. DIRECTORY OWNS SOMETHING

344 0270000000

JP

B7

.. DIRECTORY OWNS SOMETHING

* ACCOUNTING BLOCK DESTRUCTION ERROR PROCESSING

345 5110000000 x

DS.ERR3

SA1

P.ERRCL

.. ONLY ALLOCATION ERRORS EXPECTED

346 7160000000 x

SA2

P.ERRNO

.. ONLY ALLOCATION ERRORS EXPECTED

347 7160000000 x

SX6

EC.ABLK

.. ONLY ALLOCATION ERRORS EXPECTED

348 0316000457 +

IX6

X1-X6

.. ONLY ALLOCATION ERRORS EXPECTED

349 7170000000 x

NZ

X6,DSERR3

.. ONLY ALLOCATION ERRORS EXPECTED

350 73620

SX7

EN.RESV

.. ONLY ALLOCATION ERRORS EXPECTED

351 73620

SX6

X2

.. ONLY ALLOCATION ERRORS EXPECTED

```

      37767
      0307000405 +
377  7160000000 X
404  0270000000
405  7160000000 X      DS.ERR3A
412  10177
      7160000000 X
420  5110000006 X      43273
      20273
421  7160000000 X
431  7160000442 +
432  7160000000 X
440  13666
441  0200000346 +
442  5110000000 X      DS.ERR3B
      5120000000 X
443  7160000000 X
450  0270000000
      *
      *
451  0100000000 X      DSERR0  SYSERR
453  0100000000 X      DSERR1  SYSERR
455  0100000000 X      DSERR2  SYSERR
457  0100000000 X      DSERR3  SYSERR

```

IX7 X6-X7
 ZR X7,DS.ERR3A .. DAA HAS RESERVED SPACE IN IT
 CLOSE C.T2
 JP B7 .. FRETURN
 DISPCAP C.T2
 BX1 X7
 DSPDF (,X1),(RETRUFF,11)
 SA1 RETBUFF+6 *** SHOULD BE SYMBOLIC NAME ***
 MX2 50 .. SET X2 TO
 LX2 50 .. 2*59 - 1
 ABSV ABSV,MSF,X1,X4,(X5) .. MOVE SPACE OUT, IGNORE FRETURN
 ON.ERR DS.ERR3B
 ABSV ABSV,DES,X4,*,* .. TRY AGAIN
 OFF.ERR
 JP DSDR3
 SA1 P.ERRCL
 SA2 P.ERRNO
 CLOSE C.T2
 JP B7 .. FRETURN
 .. UNEXPECTED ERROR CLASS

* CALL NEXTOWN, ON FAILURE

* RETURNS IN X2,X3 THE OBJECT PART OF THE NEXT OWNERSHIP ENTRY
* AND IN R1 THE DIRECTORY FILE ADDRESS OF THE START OF THIS ENTRY
*

			ENTRY	NXTOWN	
461	43402	NXTOWN	MX4	W.ENTTYP	.. CONSTRUCT MASK
	20473		LX4	60-S.ENTTYP	.. TO EXTRACT ENTRY TYPES
462	5011000001	NO.0	GETWD	NO.FAIL	.. DIRECTORY WORD TO X1
464	6110000000 x	NO.1	GETADDR	B1	.. SAVE THIS FILE ADDRESS
	11041		BX0	X4*X1	.. SAVE ENTRY TYPE
465	20003		LX0	S.ENTTYP+W.ENTTYP	.. RJZF THE ENTRY TYPE
	0331000470 +		NG	X1,NO.3	.. END-OF-NAME FLAG IS SET
466	5011000001	NO.2	GETWD	NOERR1	
	0321000466 +		PL	X1,NO.2	.. SKIP REMAINDER OF NAME
470	5011000001	NO.3	GETWD	NOERR2	
	10211		BX2	X1	.. SAVE 1-ST WORD OF OBJECT PART
472	5011000001		GETWD	NOERR2	
	10311		BX3	X1	.. SAVE 2-ND WORD
474	5011000001	NO.4	GETWD	NO.45	.. POSSIBLY THE FIRST WORD OF AN ACCESS-PAIR
	20103		LX1	S.APF	.. LEFT-JUSTIFY THE ACCESS-PAIR FLAG
476	0321000503 +		PL	X1,NO.5	
	5011000001		GETWD	NOERR3	
500	0200000474 +		JP	NO.4	.. SKIP REMAINDER OF ACCESS-LIST
501	0300000520 +	NO.45	ZR	X0,NOERR4	.. SOFTLINK ENDED TOO SOON
	7200777774		SX0	X0-ET.OWN	
502	0310000511 +		NZ	X0,NO.FAIL	.. NO MORE OWNERSHIP ENTRIES
	0200000505 +		JP	NO.55	
503	20171	NO.5	LX1	60-S.APF	.. REPOSITION LAST DIRECTORY WORD
	0300000506 +		ZR	X0,NO.6	.. SOFTLINK, GO SKIP NAME2
504	7200777774		SX0	X0-ET.OWN	
	0310000464 +		NZ	X0,NO.1	.. KEEP LOOKING FOR AN OWNERSHIP
505	5011777776	NO.55	SA1	A1-1	.. RESET WINDOW POINTER
	0270000001		JP	B7+1	.. NORMAL RETURN
506	0331000462 +	NO.6	NG	X1,NO.0	.. END-OF-NAME FLAG OF NAME2
	5011000001		GETWD	NOERR5	
510	0200000506 +		JP	NO.6	.. SKIP REST OF NAME2
511	0270000000	NO.FAIL	JP	B7	.. FAILURE RETURN
512	0100000000 x	NOERR1	SYSERR		.. END OF DIRECTORY ENCOUNTERED IN
		*			ENTRY NAME
514	0100000000 x	NOERR2	SYSERR		.. EOD IN OBJECT PART
516	0100000000 x	NOERR3	SYSERR		.. EOD IN ACCESS LIST
520	0100000000 x	NOERR4	SYSERR		.. EOD IN SOFTLINK BEFORE SOFTNAME
522	0100000000 x	NOERR5	SYSERR		.. END OF DIRECTORY ENCOUNTERED IN
		*			SOFTLINK NAME

```

*      CALL F FOBSPEC,ONFAILURE
*
*      RETURNS: A1-X1 SET TO FIRST OBJECT SPECIFIER, B1 SET TO
*               PROPER STATE FOR SUBSEQUENT CALL TO NOBSPEC
*
*      CALL F NOBSPEC,ONFAILURE
*
*      RETURNS: A1-X1 SET TO NEXT OBJECT SPECIFIER. NOTE THAT B1 MUST
*               BE PRESERVED BETWEEN CALLS TO NOBSPEC, BECAUSE IT
*               CONTAINS A SORT OF COROUTINE LINKAGE.
*
*      SAVES: B6
*      ENTRY  FOBSPEC,NOBSPEC
*
524  5120000000 X      FOBSPEC  SA2      WINDOW      .. FETCH DESCRIPTOR WORD
                                0322000530 +      PL      X2,NOS.ST0      .. NO SUCCESSOR LINK, START RIGHT IN
525  5110000002 X      SA1      WINDOW+2      .. SUCCESSOR LINK OBJECT SPECIFIER
                                6110000530 +      SB1      NOS.ST0      .. STATE 0 = BEFORE ENTRY
526  0270000001      JP      B7+1      .. NORMAL RETURN
*
527  0210000000      NOBSPEC  JP      B1      .. BRANCH ON STATE
*
530  5011000001      NOS.ST0  GETWD      NOS.FAIL      .. NEXT DIRECTORY WORD
532  43002      NOS.ST0A  MX0      W.ENTTYP      .. MASK FOR
                                20073      LX0      60-S.ENTTYP      .. ENTRY TYPE
                                11001      BX0      X0*X1      .. SAVE ENTRY TYPE
533  0331000536 +      NG      X1,NOS.ST0C      .. BIT 1 IS END -OF-NAME FLAG
534  5011000001      NOS.ST0B  GETWD      NOSERR1      .. SKIP REST OF
                                0321000534 +      PL      X1,NOS.ST0R      .. ENTRY NAME
536  5011000001      NOS.ST0C  GETWD      NOSERR2      .. FIRST WORD OF OBJECT PART
                                0300000554 +      ZR      X0,NOS.ST2A      .. THIS IS A SOFTLINK ENTRY
540  63210      SB2      X1      .. OBJECT TYPE INDEX OF THIS ENTRY
                                6130000000 X      SB3      CA,SBPD      .. HIGHEST NUMBERED LLD-OBJECT SPECIFIER
541  0732000544 +      GT      B2,B3,NOS.ST0D      .. NOT A LLD-OBJECT
                                5011000001      GETWD      NOSERR2      .. 2ND WORD OF OBJECT PART, A LLD-
*                                   OBJECT SPECIFIER
543  6110000546 +      SB1      NOS.ST1      .. STATE 1 = IN HARD ENTRY
                                0270000001      JP      B7+1      .. NORMAL RETURN
*
544  5011000001      NOS.ST0D  GETWD      NOSERR2      .. SKIP OBJECT PART - NAME TAG OR
*                                   ACCESS-KEY
546  5011000001      NOS.ST1  GETWD      NOS.FAIL      .. FIRST WORD OF ACCESS-PAIR OR...
                                20103      LX1      S.APF      .. LEFT JUSTIFY THE ACCESS PAIR FLAG
550  0321000553 +      PL      X1,NOS.ST1A      .. NOT A PAIR
                                5011000001      GETWD      NOSERR3      .. SKIP REST OF
552  0200000546 +      JP      NOS.ST1      .. ACCESS LIST
553  20171      NOS.ST1A  LX1      60-S.APF      .. REPOSITION CURRENT WORD
                                0200000532 +      JP      NOS.ST0A      .. START NEW ENTRY
*
554  6110000555 +      NOS.ST2A  SB1      NOS.ST2      .. STATE 2 = INSIDE A SOFTLINK
                                0270000001      JP      B7+1      .. NORMAL RETURN
555  5011000001      NOS.ST2  GETWD      NOSERR2      .. ACCESS-KEY SPECIFIER
  
```


557	5011000001	NOS.ST2B	GETWD	NOSERR4	.. FIRST WORD OF ACCESS-PAIR, OR...
	20103		LX1	S.APF	
561	0321000564 +		PL	X1,NOS.ST2C	.. NOT A PAIR
	5011000001		GETWD	NOSERR3	.. SKIP REST OF
563	0200000557 +		JP	NOS.ST2B	.. ACCESS LIST
564	20171	NOS.ST2C	LX1	60-S.APF	.. REPOSITION CURRENT WORD
565	0331000530 +	NOS.ST2D	NG	X1,NOS.ST0	.. BIT 1 IS END-OF-NAME FLAG
	5011000001		GETWD	NOSERR4	.. SKIP REST OF
567	0200000565 +		JP	NOS.ST2D	.. LINK NAME
570	0270000000	NOS.FAIL	JP	B7	.. FAILURE RETURN
571	0100000000 X	NOSERR1	SYSERR		.. END OF DIRECTORY ENCOUNTERED IN
		*			ENTRY NAME
573	0100000000 X	NOSERR2	SYSERR		.. EOD IN OBJECT PART
575	0100000000 X	NOSERR3	SYSERR		.. EOD IN ACCESS PAIR
577	0100000000 X	NOSERR4	SYSERR		.. END OF DIRECTORY ENCOUNTERED IN
		*			SOFTLINK NAME

* CALL FR HASH.ONFAILURE,B5

*
* ↑HASH↑ EXPECTS TO FIND A 42-BIT UNIQUE NAME [LJZF] STORED AT P.DYNT;
* IT RETURNS WITH A1,X1 SET TO THE APPROPRIATE HASH TABLE ENTRY (AND
* THUS X1=0 IF THE ENTRY IS EMPTY) AND B1 SET TO THE CORRESPONDING
* CLIST INDEX. ↑HASH↑ FAILS IF THE HASH TABLE IS FULL.
*

* SAVES: B4 AND X5

			ENTRY	HASH,HASH1	
			EXT	L.DNT	
601	5110000000 X		HASH SA1	P.DYNT	.. 42-BIT UNIQUE NAME, LJZF
602	10411		HASH1 BX4	X1	.. COPY IT FOR USE LATER
	20152		LX1	60-18	.. RIGHT JUSTIFY THE UNIQUE NAME
	27101		PX1	X1	.. FLOAT(DIVIDEND)
603	7120000000 X		SX2	L.DNT	.. LENGTH OF HASH TABLE
	63220		SB2	X2	.. SAVE FOR LATER
	27202		PX2	X2	.. X2 ← FLOAT(DIVISOR)
604	24002		NX0	X2	.. DIVISION REQUIRES NORMALIZED DIVISOR
	44310		FX3	X1/X0	.. FLOATING POINT QUOTIENT
	26313		UX3	B1,X3	
	22313		LX3	B1,X3	.. [TRUNCATED] INTEGRAL QUOTIENT
605	27303		PX3	X3	
	42223		DX2	X2*X3	
	31112		FX1	X1-X2	
	26101		UX1	X1	.. REMAINDER
606	63110		SB1	X1	.. B1 WILL SCAN THROUGH THE TABLE
	63310		SB3	X1	.. B3 WILL MARK THE END-OF-SCAN
	43052		MX0	60-18	.. = HOLE 18
607	5111000000 X		HSH.LOOP SA1	WINDOW+B1	
	0301000615 +		ZR	X1,HSH.RET	.. EMPTY ENTRY
610	13241		BX2	X4-X1	.. COMPARE UNIQUE NAME
	11202		BX2	X0*X2	.. CLEAR OPEN COUNT FIELD
	0302000615 +		ZR	X2,HSH.RET	.. MATCHING ENTRY
611	6111000001		SB1	B1+1	
	0413000614 +		EQ	B1,B3,HSH.FAIL	.. TABLE FULL
612	0712000607 +		LT	B1,B2,HSH.LOOP	.. KEEP ON TRUCKING
	6110000000		SB1	0	
613	0513000607 +		NE	B1,B3,HSH.LOOP	.. DITTO HERE
614	0250000000		HSH.FAIL JP	B5	.. SORRY, NO ROOM
615	0250000001		HSH.RET JP	B5+1	.. NORMAL RETURN

*
* B1 → DNT INDEX OF ENTRY TO DELETE
* CALL DEHASH,B6
** USES: C.T1, ALL REGISTERS
*

ENTRY DEHASH

```

616 13666          5161000000 X      DEHASH  BX6      X6-X6      .. FIRST ZERO OUT
                                     SA6      WINDOW+B1 .. THE ENTRY TO BE DELETED

* B4 WILL BE USED TO HOLD THE INDEX OF THE LAST ENTRY REHASHED. NOTE
* THAT ↑HASH↑ SAVES R4.

617 6144000001    66410          DEHASHLP SB4      B1
                                     SB4      B4+1
620 0743000621    6130000000 X      SB3      L DNT
                                     LT      B4,B3,DEHASH1 .. NOT TO END OF TABLE YET
                                     SB4      0          .. WRAP AROUND

* X5 WILL HOLD THE ENTRY WHICH IS CURRENTLY BEING REHASHED.
* NOTE THAT ↑HASH↑ SAVES X5.

621 5154000000 X      DEHASH1 SA5      WINDOW+B4      .. FETCH NEXT ENTRY
                                     ZR      X5,RETDEHSH .. EMPTY ENTRY, FINISHED
622 43052          0905000645 +      MX0      42
                                     BX6      X0-X5
                                     SA6      P DYNT      .. PUT UNIQUE NAME WHERE
623 6150000624 +      CALL      HASH,DEHSHERR,B5 .. SYSTEM ERROR ON -TABLE FULL- REPLY
625 0414000617 +      EQ      B1,B4,DEHASHLP .. THIS ENTRY DID NOT MOVE
                                     BX6      X5          .. (↑HASH↑ SAVED X5)
                                     BX7      X7-X7
626 5161000000 X      SA6      WINDOW+B1      .. MOVE WINDOW[B4] TO WINDOW[B1]
                                     SA7      WINDOW+B4 .. ZERO OUT OLD LOCATION
627 7160000000 X      CAPIN      (CL,DRCT,CL,DNT),B4,C,T1 .. MOVE CORRESPONDING
637 7160000000 X      CAPOUT     *,B1,* .. CAPABILITY
644 0200000617 +      JP      DEHASHLP
645 0260000000      RETDEHSH JP      B4          .. RETURN
646 0100000000 X      DEHSHERR SYSERR .. UNEXPECTED REPLY FROM ↑HASH↑
  
```

```

ENTRY      CLSDYNT
EXT         0.LOL,GETDNT,PUTDNT
EXT         0.LOL

*
*      B1 -> INDEX IN LOCAL OPEN LIST OF ENTRY TO CLOSE
*      B2 -> ZERO/NOT ZERO MEANS RESPECTIVELY LOCAL/GLOBAL ACTION
*      CALL CLSDYNT,ONFAILURE
*
* IF CLOSING THE ENTRY CAUSES ITS GLOBAL OPEN COUNT TO GO TO ZERO, THEN
* CLSDYNT RETURNS VIA THE -FRETURN- RETURN.

650  5110000000 X          CLSDYNT SA1      0.LOL      .. LOCAL OPEN LIST DESCRIPTOR WORD
      20130                LX1      60-18-18    .. RIGHT JUSTIFY START ADDRESS FIELD
      53211                SA2      X1+B1      .. FETCH ENTRY TO BE CLOSED
651  43573                MX5      59          .. =-1
      36625                IX6      X2+X5      .. DECREMENT LOCAL OPEN COUNT
      43053                MX0      60-17
      15360                BX3      -X0*X6      .. EXTRACT NEW L.O.C.
652  0303000654 +          ZR      X3,CLSDYNTG .. DELETE LOCAL ENTRY AND PERFORM GLOBAL ACTION
      0520000654 +          NZ      B2,CLSDYNTG .. (DITTO)
653  54620                SA6      A2          .. UPDATE THE LOL ENTRY
      0270000001            JP      B7+1      .. NORMAL RETURN

*
654  73310                CLSDYNTG SX3      X1      .. SAVE LOL START ADDRESS
      20144                LX1      1R+18      .. REPOSITION DESCRIPTOR (IN USE FIELD AT RIGHT)
      36615                IX6      X1+X5      .. DECREMENT IN USE FIELD
      54610                SA6      A1          .. UPDATE THE DESCRIPTOR
655  63260                SB2      X6

* IF THE ENTRY BEING DELETED IS NOT THE LAST ONE IN THE LOL, COMPACT THE
* LIST BY MOVING ITS LAST ENTRY TO THE NEWLY VACATED ENPOSITION.

      0412000657 +          EQ      B1,B2,CLDNG1 .. THE LAST LOL ENTRY IS BEING DELETED
      53432                SA4      X3+B2      .. LAST ENTRY
656  10644                BX6      X4          .. NEWLY VACATED POSITION
      53631                SA6      X3+B1      .. UNIQUE NAME OF DELETED ENTRY
657  43052                CLDNG1 MX0      42      .. SAVE FOR *HASH* TO USE
      11602                BX6      X0*X2      .. LOCK AND MAP IN THE
      5160000000 X          SA6      P.DYNT      .. DYNAMIC NAME TABLE, WITH WRITE ACCESS
660  7110000001            SX1      1          .. SYSTEM ERROR ON *TABLE FULL* REPLY
      6160000662 +          CALLR    GETDNT,B6 .. (DITTO FOR *EMPTY ENTRY*)
662  6150000663 +          CALLR    HASH,CLDNGERR,B5
664  0301000673 +          ZR      X1,CLDNGERR .. =-1
      43573                MX5      59          .. DECREMENT GLOBAL OPEN COUNT
      36615                IX6      X1+X5      .. EXTRACT COUNT FILED
665  73260                SX2      X6          .. GO REMOVE ENTRY FROM DNT
      0302000670 +          ZR      X2,CLDNG2 .. UPDATE DNT ENTRY
      54610                SA6      A1          .. MAP OUT AND UNLOCK THE DNT
666  6160000667 +          CALLR    PUTDNT,B6 .. NORMAL RETURN
667  0270000001            JP      B7+1

```

SUBRS : MISCELLANEOUS SUBROUTINES
CLSDYNT - LOCAL/GLOBAL DYNAMIC NAME TAG CLOSING

COMPASS - VER 2.

09/07/71 16.42.12.

PAGE 21

670 6160000671 +
671 6160000672 +
672 0270000000

CLDNG2

CALLR
CALLR
JP

DEHASH.B6
PUTDNT.B6
B7

.. B1 POINTS TO ENTRY TO DELETE
.. RELEASE THE DNT
.. -ABNORMAL- RETURN

673 0100000000 X

CLDNGERR SYSERR

.. UNEXPECTED REPLY FROM ↑HASH↑

```
*
*      A1,X1 → START OF A DIRECTORY ENTRY
*      CALL FR SKPENT, ON FAILURE, B6
*
*      ↑SKPENT↑ RETURNS WITH A1,X1 SET TO THE START OF THE NEXT DIRECTORY ENTRY, OR
*      FAILS IF THERE ARE NO MORE ENTRIES.
```

			ENTRY	SKPENT	
675	0301000722 + 43002 20073	SKPENT	ZR MX0 LX0 BX0 LX0 NG	X1,SKPERR0 W,ENTTYP 60,S,ENTTYP X0,X1 S,ENTTYP+W,ENTTYP X1,SKPENT2	** CONSISTENCY CHECK ** .. MAKE MASK .. FOR ENTRY TYPE FIELD .. EXTRACT ENTRY TYPE .. AND RIGHT JUSTIFY IT .. END OF ENTRY NAME
676	11001 20003 0331000701 +	SKPENT1	GETWD PL	SKPERR1 X1,SKPENT1	.. NEXT WORD OF ENTRY NAME .. NOT END, KEEP LOOKING
677	5011000001 0321000677 +	SKPENT2	GETWD GETWD	SKPERR2 SKPERR2	.. FIRST WORD OF OBJECT PART .. SECOND WORD OF OBJECT PART
701	5011000001	SKPENT3	GETWD LX1 PL	SKPFAIL0 S,APF X1,SKPENT4	.. PART OF AN ACCESS-PAIR, OR... .. LEFT JUSTIFY ACCESS-PAIR FLAG .. NOT A PAIR
704	5011000001 20103		GETWD JP	SKPERR3 SKPENT3	.. SKIP THE SECOND WORD OF THE ACCESS-PAIR .. SKIP REST OF ACCESS-LIST
706	0321000711 + 5011000001	SKPENT4	LX1 NZ	60,S,APF X0,SKPRET	.. REPOSITION CURRENT DIRECTORY WORD .. NOT A SOFTLINK, NEXT ENTRY FOUND
710	0200000704 +	SKPENT5	NG GETWD JP	X1,SKPRET0 SKPERR4 SKPENT5	.. NEXT OWRD OF SOFTLINK NAME
711	20171 0310000717 +	SKPRET0	GETWD	SKPFAIL	.. FIRST WORD OF NEXT ENTRY, OR .. END OF DIRECTORY
712	0331000715 + 5011000001	*			.. NORMAL RETURN
714	0200000712 +	SKPRET	JP	B6+1	
715	5011000001	SKPFAIL0	ZR	X0,SKPERR4	.. SOFTLINK SHOULD NOT END YET
717	0260000001	SKPFAIL	JP	B6	.. FAILURE RETURN
720	0300000732 +	*			
721	0260000000	*			
		*			
		* DISASTERS:			
722	0100000000 X	SKPERR5	SYSERR		• THERE WAS NOTHING TO SKIP
724	0100000000 X	SKPERR1	SYSERR		• END OF DIRECTORY (EOD) IN ENTRY NAME
726	0100000000 X	SKPERR2	SYSERR		• EOD IN THE OBJECT PART OF A DIRECTORY ENTRY
730	0100000000 X	SKPERR3	SYSERR		• EOD IN THE ACCESS-LIST OF AN ENTRY
732	0100000000 X	SKPERR4	SYSERR		• EOD IN SOFTLINK NAME

```

*          CALL      CLOCKS          .. CLKBUFF MUST CONTAIN INITIAL VALUES
*
* ↑CLOCKS↑ DISPLAYS THE CLOCKS AGAIN, SUBTRACTS INITIAL FROM
* FINAL VALUES, AND ADDS THE DIFFERENCES TO THE TOTALS.

734  5110000002 x          CLOCKS  SA1      C:KBUFF+SYSTM  .. PICK UP
                                5120000003 x  SA2      C:KBUFF+SWPTIME .. INITIAL TIMES
735  5130000001 x          SA3      C:KBUFF+USRTIME  .. FROM BUFFER
736  0130000000 x          SYSCALL  D:PCCLKS  .. GET CURRENT TIMES
737  5140000002 x          SA4      C:KBUFF+SYSTM  .. NOW UPDATE TOTALS
                                5150000000 x  SA5      O:SYSTM
740  37441                IX4      X4-X1          .. DELTA(SYSTEM TIME)
                                36654         IX6      X5+X4          .. NEW TOTAL
                                5140000003 x  SA4      C:KBUFF+SWPTIME
741  54650                SA6      A5
                                5150000000 x  SA5      O:SWPTM
                                37442         IX4      X4-X2          .. DELTA(SWAP TIME)
742  36654                IX6      X5+X4          .. NEW TOTAL
                                54650         SA6      A5
                                5140000001 x  SA4      C:KBUFF+USRTIME
743  5150000000 x          SA5      O:USRTM
                                37443         IX4      X4-X3          .. DELTA(USER TIME)
                                36654         IX6      X5+X4          .. NEW TOTAL
744  54650                SA6      A5
                                0270000000    JP      B7          .. RETURN

                                2  SYSTM  EQU      2          .. DEFINE POSITIONS IN
                                3  SWPTM  EQU      3          .. BUFFER RETURNED BY DISPLAY
                                1  USRTM  EQU      1          .. CLOCK OPERATION

751                                END

```

36675 STORAGE USED
6600 ASSEMBLY

2079 STATEMENTS
10.313 SECONDS

233 SYMBOLS
630 REFERENCES

000061 INVENTED SYMBOLS

ABSV.DES	2		13/35	14/14					
ABSV.MSF	5		14/12						
ADDWDS	157	PROGRAM*	2/03 E	9/26 L					
AW.LOOP	162	PROGRAM*	9/35 L	9/39					
AW.RET	164	PROGRAM*	9/31	9/40 L					
BEA	66	PROGRAM*	2/02 E	6/16 L					
BEAERR1	135	PROGRAM*	6/23	6/26	6/31	6/34	7/17 L		
BEAERR2	137	PROGRAM*	6/43	6/52	7/18 L				
BEAERR3	141	PROGRAM*	7/12	7/19 L					
BEAERR4	122	PROGRAM*	7/02 L						
BEA.END0	121	PROGRAM*	6/49	7/01 L					
BEA.END1	124	PROGRAM*	6/50	7/04 L					
BEA.EXPL	105	PROGRAM*	6/38 L	6/44					
BEA.IMPL	113	PROGRAM*	6/37	6/47 L					
BEA.MVNM	126	PROGRAM*	7/07 L	7/12					
BEA.RET	134	PROGRAM*	7/01	7/04	7/15	7/15 L			
BEA.RETO	132	PROGRAM*	7/10	7/14 L					
BEA.SKAL	114	PROGRAM*	6/48 L	6/52					
BEA.SOFT	77	PROGRAM*	6/21	6/30 L					
BEA.UNL	104	PROGRAM*	6/28	6/37 L					
CANON	0	PROGRAM*	2/02 E	3/12 L	8/11				
CAN.LOOP	5	PROGRAM*	3/26 L	3/31					
CAN.LWD	4	PROGRAM*	3/19	3/22 L					
CAN.TRIM	1	PROGRAM*	3/15 L	3/21					
CA.DSKF	0	EXTERNAL*	12/22						
CA.SBPD	0	EXTERNAL*	12/19	16/33					
CC.DIR	0	EXTERNAL*	13/11	13/29	13/40				
CLDNGERR	673	PROGRAM*	20/45	20/45	21/05 L				
CLDNG1	657	PROGRAM*	20/35	20/39 L					
CLDNG2	670	PROGRAM*	20/49	21/01 L					
CLKBUFF	0	EXTERNAL*	23/06	23/07	23/08	23/10	23/14	23/20	
CLOCKS	734	PROGRAM*	2/02 E	23/06 L					
CLSDYNT	650	PROGRAM*	20/01 E	20/14 L					
CLSDYNTG	654	PROGRAM*	20/21	20/22	20/26 L				
CL.DNT	0	EXTERNAL*	19/28						
CL.DRCT	0	EXTERNAL*	12/22	13/11	13/29	13/40	19/28		
CL.OPs	0	EXTERNAL*	12/21	12/53	13/28	13/39	14/05	14/13	19/28
			12/24	13/07	13/34	13/40	14/07	14/12	
			12/28	13/10	13/36	14/03	14/11	19/27	
COPY	174	PROGRAM*	10/25 E	10/27 L	10/31				
CPYNAME	177	PROGRAM*	11/11 E	11/13 L					
CPYNAME1	201	PROGRAM*	11/19 L	11/25					
CPYNAME2	205	PROGRAM*	11/19	11/26 L					
CPYNAME3	206	PROGRAM*	11/35 L	11/39					
CPYNAME4	210	PROGRAM*	11/37	11/41 L					
CPYNMERR	212	PROGRAM*	11/25	11/46 L					
C.T1	0	EXTERNAL*	12/22	12/25	12/29	13/37	19/28		
C.T2	0	EXTERNAL*	12/25	13/08	13/29	13/41	14/06		
			13/01	13/11	13/40	14/04	14/20		
DEHASH	616	PROGRAM*	18/52 E	19/01 L	21/02				
DEHASHLP	617	PROGRAM*	19/08 L	19/22	19/29				
DEHASH1	621	PROGRAM*	19/10	19/16 L					
DESHERR	646	PROGRAM*	19/22	19/33 L					

NAME	COUNT	TYPE	DATE	TIME	DATE	TIME	DATE	TIME	DATE	TIME
DELWDS	165	PROGRAM*	2/03 E	9/51 L						
DF:ABSV	0	EXTERNAL*	13/35	14/12	14/14					
DF:CLO	0	EXTERNAL*	13/01	13/41	14/04	14/20				
DF:CMQO	0	EXTERNAL*	13/11							
DF:DFIL	0	EXTERNAL*	12/29	13/37						
DF:DSF	0	EXTERNAL*	14/08							
DF:READ	0	EXTERNAL*	13/08							
DF:XOPN	0	EXTERNAL*	12/25							
DF:ZRM	0	EXTERNAL*	13/29	13/40						
DSDR1	316	PROGRAM*	13/14	13/16 L						
DSDR15	320	PROGRAM*	13/17 L	13/22						
DSDR2	324	PROGRAM*	13/17	13/22	13/26 L					
DSDR3	346	PROGRAM*	13/31	13/36 L	14/15					
DSDR.E1	354	PROGRAM*	13/19	13/39 L						
DSERR0	451	PROGRAM*	12/18	12/20	14/23 L					
DSERR1	453	PROGRAM*	13/08	14/24 L						
DSERR2	455	PROGRAM*	14/25 L							
DSERR3	457	PROGRAM*	13/51	14/26 L						
DSPCLKS	0	EXTERNAL*	23/09							
DSTRY	214	PROGRAM*	12/11 E	12/18 L						
DSTRYDIR	265	PROGRAM*	12/26	13/07 L						
DS.ERR0	246	PROGRAM*	12/25	12/32 L						
DS.ERR1	250	PROGRAM*	12/23	12/38 L						
DS.ERR1A	255	PROGRAM*	12/42	12/47 L						
DS.ERR2	256	PROGRAM*	12/27	12/51 L						
DS.ERR3	373	PROGRAM*	13/33	13/47 L						
DS.ERR3A	405	PROGRAM*	14/02	14/05 L						
DS.ERR3B	442	PROGRAM*	14/12	14/17 L						
DW.LOOP	171	PROGRAM*	10/09 L	10/13						
DW.RET	173	PROGRAM*	10/05	10/14 L						
EC.ABLK	0	EXTERNAL*	13/49							
EC.DRCT	0	EXTERNAL*	12/32	13/41						
EC.FILE	0	EXTERNAL*	12/40							
EC.CCAP	0	EXTERNAL*	12/22							
EC.CIN	0	EXTERNAL*	19/28							
EC.COUT	0	EXTERNAL*	19/29							
EC.DSCP	0	EXTERNAL*	14/06							
EF.DRCT	0	EXTERNAL*	13/11							
EN.NODF	0	EXTERNAL*	12/44							
EN.OWNS	0	EXTERNAL*	13/42							
EN.RESV	0	EXTERNAL*	13/52							
EN.RSRV	0	EXTERNAL*	12/33							
ERRLOC	0	EXTERNAL*	12/24 S	12/26 S	12/28 S	12/30 S	13/34 S	13/36 S	14/13 S	14/15 S
ET.OWN	3		15/28	15/34						
E.BADNM	0	EXTERNAL*	3/15	3/51	4/04					
E.DUP	0	EXTERNAL*	8/15							
E.IMPOT	0	EXTERNAL*	6/39	6/40						
E.NOSPC	0	EXTERNAL*	9/16							
FLAPPER	0	EXTERNAL*								
FORSPEC	524	PROGRAM*	16/12 E	16/14 L						
GETDNT	0	EXTERNAL*	20/44							
GETSPAC	153	PROGRAM*	2/03 E	9/09 L						
GN.EST2	17	PROGRAM*	3/48	3/53 L						

[illegible]

SUBRS : MISCELLANEOUS SUBROUTINES
SYMBOLIC REFERENCE TABLE.

COMPASS - VER 2.

09/07/71 16.42.15.

PAGE 27

NO.3	470	PROGRAM*	15/14	15/17 L			
NO.4	474	PROGRAM*	15/21 L	15/25			
NO.45	501	PROGRAM*	15/22	15/27 L			
NO.5	503	PROGRAM*	15/23	15/32 L			
NO.55	505	PROGRAM*	15/30	15/36 L			
NO.6	506	PROGRAM*	15/33	15/39 L	15/41		
NSERR1	56	PROGRAM*	5/23	5/26	5/49 L		
NSERR2	60	PROGRAM*	5/28	5/29	5/51 L		
NSERR3	62	PROGRAM*	5/33	5/52 L			
NSERR4	64	PROGRAM*	5/39	5/53 L			
NSERR4B	54	PROGRAM*	5/45 L				
NS.ACCL	41	PROGRAM*	5/29 L	5/33			
NS.DIFF	35	PROGRAM*	5/19	5/26 L			
NS.FAIL	21	PROGRAM*	5/13	5/44	5/47 D		
NS.FAIL0	53	PROGRAM*	5/30	5/44 L			
NS.LOOP	27	PROGRAM*	5/18 L	5/23			
NS.MATCH	52	PROGRAM*	5/20	5/41 L			
NS.NXT1	23	PROGRAM*	5/12 L	5/37			
NS.NXT2	25	PROGRAM*	5/13 L	5/36			
NS.SFTL	47	PROGRAM*	5/37 L	5/39			
NS.SKPNM	33	PROGRAM*	5/25 L	5/26			
NS.SKPSF	46	PROGRAM*	5/31	5/35 L			
NXTOWN	461	PROGRAM*	15/06 E	15/08 L			
OPEN	0	EXTERNAL*	8/13				
O.LOL	0	EXTERNAL*	20/14				
O.SWPTM	0	EXTERNAL*	23/16				
O.SYSTM	0	EXTERNAL*	23/11				
O.USRTM	0	EXTERNAL*	23/21				
PUTDNT	0	EXTERNAL*	20/52	21/03			
P.DIRCT	0	EXTERNAL*	6/32 S				
P.DYNT	0	EXTERNAL*	18/14	19/20 S	20/41 S		
P.ERRCL	0	EXTERNAL*	12/38	12/51	13/47	14/17	
P.ERRNO	0	EXTERNAL*	12/39	12/52	13/48	14/18	
P.KEY	0	EXTERNAL*	6/27	6/34			
P.NAME	0	EXTERNAL*	6/16	7/06	8/09	8/13	
RETBUFF	0	EXTERNAL*	14/08	14/08			
RETDATA	0	EXTERNAL*	13/08	13/08			
RETDEHSH	645	PROGRAM*	19/17	19/31 L			
RETDSTR	245	PROGRAM*	12/30 L	12/46			
SKPENT	675	PROGRAM*	13/22	22/07 E	22/09 L		
SKPENT1	677	PROGRAM*	22/15 L	22/16			
SKPENT2	701	PROGRAM*	22/14	22/17 L			
SKPENT3	704	PROGRAM*	22/19 L	22/23			
SKPENT4	711	PROGRAM*	22/21	22/24 L			
SKPENT5	712	PROGRAM*	22/26 L	22/28			
SKPERR0	722	PROGRAM*	22/09	22/40 L			
SKPERR1	724	PROGRAM*	22/16	22/41 L			
SKPERR2	726	PROGRAM*	22/18	22/19	22/42 L		
SKPERR3	730	PROGRAM*	22/23	22/43 L			
SKPERR4	732	PROGRAM*	22/28	22/33	22/44 L		
SKPFAIL	721	PROGRAM*	22/30	22/34 L			
SKPFAIL0	720	PROGRAM*	22/20	22/33 L			
SKPRET	717	PROGRAM*	22/25	22/31 L			

SUBRS : MISCELLANEOUS SUBROUTINES
SYMBOLIC REFERENCE TABLE.

COMPASS - VER 2.

09/07/71 16.42.15.

PAGE 28

SKPRETO	715	PROGRAM*	22/26	22/29 L						
SWPTIME	3		23/07	23/14	23/28 D					
SYSERR	0	EXTERNAL*	5/45	5/53	7/19	14/25	15/48	17/16	21/05	22/43
			5/49	7/02	11/46	14/26	15/49	17/17	22/40	22/44
			5/51	7/17	14/23	15/45	15/50	17/18	22/41	
			5/52	7/18	14/24	15/47	17/14	19/33	22/42	
SYSTIME	2		23/06	23/10	23/27 D					
S.APF	3		5/30	6/39	6/49	15/22	16/43	17/02	22/20	
			5/35	6/45	7/05	15/32	16/47	17/07	22/24	
S.ENTTYP	1		5/11	5/15	11/17	15/09	15/13	16/25	22/11	22/13
S.OPTS	22		6/45							
UNIQUE	143	PROGRAM*	2/02 E	8/09 L						
USRTIME	1		23/08	23/20	23/29 D					
WINDOW	0	EXTERNAL*	5/13	9/32	13/11	13/26	16/16	19/16		
			9/09	9/51	13/12	15/11	18/32	19/25 S		
			9/26	10/07	13/13	16/14	19/02 S	19/26 S		
W.ENTTYP	2		5/10	11/16	15/08	15/13	16/24	22/10	22/13	