


```

NN      NN      TTTTTTTTTT  000000  DDDDDDDD  AAAAAA  PPPPPPPP  CCCCCCCC  RRRRRRRR  CCCCCCCC
NN      NN      TTTTTTTTTT  000000  DDDDDDDD  AAAAAA  PPPPPPPP  CCCCCCCC  RRRRRRRR  CCCCCCCC
NN      NN      TT          00      00      DD      DD  AA      AA  PP      PP  CC          RR      RR  CC
NN      NN      TT          00      00      DD      DD  AA      AA  PP      PP  CC          RR      RR  CC
NNNN    NN      TT          00      0000  DD      DD  AA      AA  PP      PP  CC          RR      RR  CC
NNNN    NN      TT          00      0000  DD      DD  AA      AA  PP      PP  CC          RR      RR  CC
NN  NN  NN      TT          00      00      DD      DD  AA      AA  PPPPPPPP  CC          RRRRRRRR  CC
NN  NN  NN      TT          00      00      DD      DD  AA      AA  PPPPPPPP  CC          RRRRRRRR  CC
NA      NNNN    TT          0000      00      DD      DD  AAAAAAAAAA  PP          RR      RR  CC
NN      NNNN    TT          0000      00      DD      DD  AAAAAAAAAA  PP          RR      RR  CC
NN      NN      TT          00      00      DD      DD  AA      AA  PP          RR      RR  CC
NN      NN      TT          00      00      DD      DD  AA      AA  PP          RR      RR  CC
NN      NN      TT          000000  DDDDDDDD  AA      AA  PP          RR      RR  CC
NN      NN      TT          000000  DDDDDDDD  AA      AA  PP          RR      RR  CC

```

```

LL      I I I I I I  SSSSSSSS
LL      I I I I I I  SSSSSSSS
LL      II          SS
LL      II          SS
LL      II          SS
LL      II          SS
LL      II          SSSSSS
LL      II          SSSSSS
LL      II          SS
LL      II          SS
LL      II          SS
LL      II          SS
LLLLLLLLLLLL  I I I I I I  SSSSSSSS
LLLLLLLLLLLL  I I I I I I  SSSSSSSS

```

(2)	67
(3)	98
(4)	128
(5)	172

DECLARATIONS
NT\$CRC_TABLE - CRC POLYNOMIAL TABLE
NT\$CRC_INIT - INITIALIZE CRC COMPUTATION
NT\$CRC_LOGERR - LOG DAP CRC ERROR

```

0000 1          $BEGIN NTODAPCRC,000,NF$NETWORK,<DAP LEVEL CRC>,<LONG,NOWRT>
0000 2
0000 3
0000 4 *****
0000 5 *****
0000 6 *   COPYRIGHT (c) 1978, 1980, 1982, 1984 BY   *
0000 7 *   DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
0000 8 *   ALL RIGHTS RESERVED. *
0000 9 *
0000 10 *  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0000 11 *  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
0000 12 *  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
0000 13 *  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0000 14 *  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
0000 15 *  TRANSFERRED. *
0000 16 *
0000 17 *  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
0000 18 *  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
0000 19 *  CORPORATION. *
0000 20 *
0000 21 *  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
0000 22 *  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
0000 23 *
0000 24 *****
0000 25 *****
0000 26
0000 27
0000 28 **
0000 29 : Facility: RMS
0000 30
0000 31 : Abstract:
0000 32
0000 33 :       This module contains the CRC table for DAP CRC checksum computation
0000 34 :       plus routines related to CRC computation and error logging.
0000 35
0000 36 : Environment: VAX/VMS, executive mode
0000 37
0000 38 : Author: James A. Krycka,      Creation Date: 16-JUN-1979
0000 39
0000 40 : Modified By:
0000 41
0000 42
0000 43 :       V03-007 RAS0315      Ron Schaefer      22-Jun-1984
0000 44 :       Eliminate tiny data psect.
0000 45
0000 46 :       V03-006 JEJ0028      J E Johnson      11-Apr-1984
0000 47 :       Minor equate cleanup.
0000 48
0000 49 :       V03-005 KRM0054      K Malik      10-Aug-1982
0000 50 :       Removed nodename parsing code from NT$CRC_LOGERR
0000 51 :       (now done in NT$EXCH_CNf) & used new NWA$T_NODEBUF
0000 52 :       and NWA$B_NODBUFSIZ symbols.
0000 53
0000 54 :       V02-004 KRM0024      K R Malik      8-AUG-1981
0000 55 :       Fix bugs in NT$CRC_LOGERR.
0000 56
0000 57 :       V02-003 KRM0023      K R Malik      31-JUL-1981

```

B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z

0000 58 :
0000 59 :
0000 60 :
0000 61 :
0000 62 :
0000 63 :
0000 64 :
0000 65 :--

Add NT\$CRC_LOGERR routine to log DAP CRC errors to the
DECnet Event Logger (EVL).

V02-002 JAK0058 J A Krycka 01-JUN-1981
This module was created from code previously residing in
NTOCREATE.

```
0000 67      .SBTTL  DECLARATIONS
0000 68
0000 69      :
0000 70      : Include Files:
0000 71      :
0000 72
0000 73      $DAPPLGDEF      ; Define DAP prologue symbols
0000 74      $DAPCNFDEF     ; Define DAP Configuration message
0000 75      $DAPCRCDEF     ; Define DAP CRC checksum symbols
0000 76      $EVCDEF        ; Define event class symbols
0000 77      $IFBDEF        ; Define IFAB symbols
0000 78      $IODEF        ; Define IOS_ symbols
0000 79      $NFBDEF       ; Define network function definition symbols
0000 80      $NMADEF       ; Define network management symbols
0000 81      $NWADEF       ; Define network work area symbols
0000 82      $RAWDEF       ; Define raw event record format
0000 83
0000 84      :
0000 85      : Macros:
0000 86      :
0000 87      :     None
0000 88      :
0000 89      : Equated Symbols:
0000 90      :
0000 91      :     None
0000 92      :
0000 93      : Own Storage:
0000 94      :
0000 95      :     None
0000 96      :
```

```
0000 98 .SBTTL NT$CRC_TABLE - CRC POLYNOMIAL TABLE
0000 99
0000 100 :++
0000 101 : This is the CRC table for use in DAP CRC checksum computation.
0000 102 : The CRC polynomial function (order 16) used is:
0000 103 :
0000 104 : X**16 + X**15 + X**13 + X**7 + X**4 + X**2 + X**1 + 1
0000 105 :--
0000 106
0000 107 .ALIGN LONG
0000 108
0000 109 NT$CRC_TABLE:: : CRC polynomial table
00000000 0000 110 .LONG DAP$K_CRC_TBL0 : Table entry 0
000053E3 0004 111 .LONG DAP$K_CRC_TBL1 : Table entry 1
0000A7C6 0008 112 .LONG DAP$K_CRC_TBL2 : Table entry 2
0000F425 000C 113 .LONG DAP$K_CRC_TBL3 : Table entry 3
00009D87 0010 114 .LONG DAP$K_CRC_TBL4 : Table entry 4
0000CE64 0014 115 .LONG DAP$K_CRC_TBL5 : Table entry 5
00003A41 0018 116 .LONG DAP$K_CRC_TBL6 : Table entry 6
000069A2 001C 117 .LONG DAP$K_CRC_TBL7 : Table entry 7
0000E905 0020 118 .LONG DAP$K_CRC_TBL8 : Table entry 8
0000BAE6 0024 119 .LONG DAP$K_CRC_TBL9 : Table entry 9
00004EC3 0028 120 .LONG DAP$K_CRC_TBLA : Table entry 10
00001D20 002C 121 .LONG DAP$K_CRC_TBLB : Table entry 11
00007482 0030 122 .LONG DAP$K_CRC_TBLC : Table entry 12
00002761 0034 123 .LONG DAP$K_CRC_TBLD : Table entry 13
0000D344 0038 124 .LONG DAP$K_CRC_TBLE : Table entry 14
000080A7 003C 125 .LONG DAP$K_CRC_TBLF : Table entry 15
0040 126
```



```

0053 172      .SBTTL  NT$CRC_LOGERR - LOG DAP CRC ERROR
0053 173
0053 174 :++
0053 175 : NT$CRC_LOGERR - generates an appropriate message to the DECnet Event Logger
0053 176 : (EVL) to log the occurrence of a DAP level CRC checksum error.
0053 177 :
0053 178 : Calling Sequence:
0053 179 :
0053 180 :     BSBW  NT$CRC_LOGERR
0053 181 :
0053 182 : Input Parameters:
0053 183 :
0053 184 :     R7      NWA (=DAP) address
0053 185 :     R8      RAB address
0053 186 :     R9      IRAB address
0053 187 :     R10     IFAB address
0053 188 :     R11     Impure area address
0053 189 :
0053 190 : Implicit Inputs:
0053 191 :
0053 192 :     NWA$T_NODEBUF
0053 193 :     NWA$B_NODBUFSIZ
0053 194 :
0053 195 : Output Parameters:
0053 196 :
0053 197 :     None
0053 198 :
0053 199 : Implicit Outputs:
0053 200 :
0053 201 :     None
0053 202 :
0053 203 : Completion Codes:
0053 204 :
0053 205 :     None
0053 206 :
0053 207 : Side Effects:
0053 208 :
0053 209 :     An attempt is made to log a DAP CRC error to the DECnet Event Logger.
0053 210 :     Return status of the request is neither checked nor returned.
0053 211 :
0053 212 :--
0053 213
0053 214 NT$CRC_LOGERR::
3F  BB 0053 215     PUSHR  #*M<R0,R1,R2,R3,R4,R5> ; Entry point
0053 216     ; Save registers
0053 217 :
0053 218 : Get the name of the partner node
0053 219 :
52 0168 C7 90 0053 219     MOVB   NWA$B_NODBUFSIZ(R7),R2 ; Get size of nodename
53 0169 C7 9E 005A 220     MOVAB  NWA$T_NODEBUF(R7),R3   ; Get address of node spec list
005F 221
005F 222 :
005F 223 : Build the NFB descriptor, the NFB, the event buffer descriptor and
005F 224 : the event buffer itself.
005F 225 :
005F 226 :
55 0120 C7 9E 005F 227     MOVAB  NWA$T_TEMP(R7),R5      ; Get address to build event buffer
85 05 9A 0064 228     MOVZBL #5,(R5)+             ; NFB desc count field

```

```

85 04 A5 DE 0067 229 MGVAL 4(R5),(R5)+ ; NFB desc addr field
85 85 1C 90 006B 230 MOVB #NFB$C_LOGEVENT,(R5)+ ; NFB
85 52 22 C1 0070 232 ADDL3 #RAWST_DATA+4,R2,(R5)+ ; Event buffer desc count
85 04 A5 DE 0074 233 MOVAL 4(R5),(R5)+ ; Event buffer desc address
85 F8 A5 B0 0078 234 MOVW -8(R5),(R5)+ ; Put event count in event buffer
007C 235 $GETTIM_S - ; Get the system time
007C 236 TIMADR=(R5) ; Put time in the event buffer
85 55 08 C0 0085 237 ADDL2 #8,R5 ; Bump the event buffer pointer
85 2001 8F B0 0088 238 MOVW #EVC$C_VMS_DPC,(R5)+ ; Put in the event code
85 FF 8F 90 008D 239 MOVB #EVC$C_SRC_NON,(R5)+ ; Put in the source type
55 11 C0 0091 240 ADDL2 #17,R5 ; Skip the source ID field
85 85 00 B0 0094 241 MOVW #EVC$C_VMS_PNOD,(R5)+ ; Remote node name
85 40 8F 90 0097 242 MOVB #NMASM_PTY_ASC,(R5)+ ; Data type
85 85 52 90 009B 243 MOVW R2,(R5)+ ; Put in the nodename count
65 63 52 28 009E 244 MOVCL R2,(R3),(R5) ; Put in the nodename
55 0120 C7 9E 00A2 245 MOVAB N$WAST_TEMP(R7),R5 ; Get address of TEMP
54 0D A5 9E 00A7 246 MOVAB 13(R5),R4
00AB 247 $QIO_S - ;
00AB 248 CHAN=IFB$W_CHNL(R10),- ; Use already assigned channel
00AB 249 FUNC=#IOS_ACPCONTROL,- ; Function code
00AB 250 P1=(R5),- ; Adr of descriptor of NFB
00AB 251 P2=R4 ; Adr of descriptor of event buffer
00C7 252
00C7 253
00C7 254 ; Do not bother to check the return status since we wish to preserve the
00C7 255 ; original DAP CRC error status in R0 (and not the success or failure of
00C7 256 ; our attempt to log it).
00C7 257
00C7 258
3F BA 00C7 259 30$. POPR #^M<R0,R1,R2,R3,R4,R5> ; Restore registers
05 05 00C9 260 RSB ; Exit
00CA 261
00CA 262 .END ; End of module

```

NTODAPCRC
Symbol table

DAP LEVEL CRC

H 16

15-SEP-1984 23:53:33 VAX/VMS Macro V04-00
5-SEP-1984 16:20:27 [RMS.SRC]NTODAPCRC.MAR;1

Page 8
(5)

```

SS.PSECT_EP           = 00000000
SSRMSTEST             = 0000001A
SSRMS_PBUGCHK         = 00000010
SSRMS_TBUGCHK         = 00000008
SSRMS_UMODE           = 00000004
SST1                  = 00000001
DAPSB_DCODE_FID      = 00000019
DAPSB_DCODE_MAC      = 0000001B
DAPSB_DCODE_MSG      = 0000001A
DAPSB_DECVER         = 00000047
DAPSB_ECONUM         = 00000045
DAPSB_FILESYS        = 00000043
DAPSB_OSTYPE         = 00000042
DAPSB_USRNUM         = 00000046
DAPSB_USRVER         = 00000048
DAPSB_VERNUM         = 00000044
DAPSB_X_FIELD        = 00000024
DAPSC_BCN            = 000000C0
DAPSK_BLN            = 000000C0
DAPSK_CRC_INIT       = 0000FFFF
DAPSK_CRC_TBLO       = 00000000
DAPSK_CRC_TBL1       = 000053E3
DAPSK_CRC_TBL2       = 0000A7C6
DAPSK_CRC_TBL3       = 0000F425
DAPSK_CRC_TBL4       = 00009D87
DAPSK_CRC_TBL5       = 0000CE64
DAPSK_CRC_TBL6       = 00003A41
DAPSK_CRC_TBL7       = 000069A2
DAPSK_CRC_TBL8       = 0000E905
DAPSK_CRC_TBL9       = 0000BAE6
DAPSK_CRC_TBLA       = 00004EC3
DAPSK_CRC_TBLB       = 00001D20
DAPSK_CRC_TBLC       = 00007482
DAPSK_CRC_TBLD       = 00002761
DAPSK_CRC_TBLE       = 0000D344
DAPSK_CRC_TBLF       = 000080A7
DAPSL_CMWA          = 00000030
DAPSL_CRC_RSLT       = 00000020
DAPSL_DCODE_STS      = 00000018
DAPSL_MSG_MASK       = 0000C01C
DAPSL_SSPQA         = 00000080
DAPSL_TEMP          = 00000090
DAPSQ_DCODE_FLG     = 00000000
DAPSQ_MSG_BUF1      = 00000008
DAPSQ_MSG_BUF2      = 00000010
DAPSQ_SYSCAP        = 00000028
DAPSV_DAPCRC        = 00000015
DAPSW_BUFSIZ        = 00000040
DAPSW_PARTNER       = 00000006
DAPSW_VERSION       = 00000004
EVCSC_SRC_NON       = 000000FF
EVCSC_VMS_DPC       = 00002001
EVCSC_VMS_PNOD      = 00000000
IFBSW_CHNC          = 00000020
IOS_ACPCONTROL      = 00000038
NFBSC_LOGEVENT      = 0000001C
NMASH_PTY_ASC       = 00000040

```

```

NTSCRC_INIT          00000040 RG 01
NTSCRC_LOGERR        00000053 RG 01
NTSCRC_TABLE         00000000 RG 01
NWSB_ALLXABCNT      0000011C
NWSB_DAP_RAC        000000C9
NWSB_FILESYS        000000C5
NWSB_KEYXABCNT      0000011D
NWSB_NETSTRSIZ      0000016F
NWSB_NODBUFSIZ     00000168
NWSB_ORG            000000C6
NWSB_OSTYPE         000000C4
NWSB_RFM            000000C7
NWSB_RMS_RAC        000000C8
NWSB_BLN            00000800
NWSK_BLN            00000800
NWSL_ALLXABADR      00000100
NWSL_DATXABADR      00000104
NWSL_DEV            000000C0
NWSL_FHCXABADR      00000108
NWSL_KEYXABADR      0000010C
NWSL_MSG_MASK       000000D4
NWSL_PROXABADR      00000110
NWSL_RDXABADR       00000114
NWSL_SAVE_FLGS      00000128
NWSL_SUMXABADR      00000118
NWSL_THREAD         000000FC
NWSL_XLTATTR        00000238
NWSL_XLTBUFFLG      0000022C
NWSL_XLTCNT         00000228
NWSL_XLTMAXINDX     00000234
NWSL_XLTSIZ         00000230
NWSQ_ACS            00000244
NWSQ_BIGBUF         00000170
NWSQ_BLD            000000F0
NWSQ_FLG            00000000
NWSQ_INODE          0000025C
NWSQ_IOSB           000000D8
NWSQ_LINKDEF        00000160
NWSQ_LOGNAME        0000023C
NWSQ_NCB            00000264
NWSQ_RCV            000000E0
NWSQ_SAVE_DESC      00000120
NWSQ_XLTBUF1        0000024C
NWSQ_XLTBUF2        00000254
NWSQ_XMT            000000E8
NWSW_ACSBUF         0000026C
NWSW_AUXBUF         000005E0
NWSW_DAP            00000000
NWSW_INODEBUF       000004AC
NWSW_ITM_ATTR       00000200
NWSW_ITM_END        00000224
NWSW_ITM_LST        00000200
NWSW_ITM_MAXINDX    00000218
NWSW_ITM_STRING     0000020C
NWSW_NCBBUF         0000052C
NWSW_NODEBUF        00000169
NWSW_RCVBUF         000001A0

```

```

NFAST_SCAN      00000100
NFAST_TEMP      00000120
NFAST_XLTBUF1   000002AC
NFAST_XLTBUF2   000003AC
NFAST_XMTBUF    000003C0
NFASTW_BUILD    000000D2
NFASTW_DAPBUFSIZ 000000CA
NFASTW_DIR_OFF  000000CC
NFASTW_DISPLAY  000000D0
NFASTW_FIL_OFF  000000CE
NFASTW_JNLXABJOP 0000011E
RAWSC_SRCTYP    0000000C
RAWSC_SIZE      0000001F
RAWSC_SIZE      0000001F
RAWST_DATA      0000001E
RAWST_SRCID     0000000D
RAWST_SYSTM     00000002
RAWSW_BYTES     00000000
RAWSW_EVTCODE   0000000A
SYSSGETTIM     ***** GX 01
SYSSQIO        ***** GX 01
    
```

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes												
. ABS	00000000 (0.)	00 (0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE		
NFSNETWORK	000000CA (202.)	01 (1.)	PIC	USR	CON	REL	LCL	NOSHR	EXE	RD	NOWRT	NOVEC	LONG		
\$ABSS	00000800 (2048.)	02 (2.)	NOPIC	USR	CON	ABS	LCL	NOSHR	EXE	RD	WRT	NOVEC	BYTE		

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	34	00:00:00.10	00:00:00.95
Command processing	161	00:00:01.15	00:00:06.08
Pass 1	624	00:00:22.02	00:01:00.88
Symbol table sort	0	00:00:03.37	00:00:06.64
Pass 2	84	00:00:03.18	00:00:08.96
Symbol table output	19	00:00:00.17	00:00:00.23
Psect synopsis output	3	00:00:00.03	00:00:00.16
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	928	00:00:30.03	00:01:23.91

The working set limit was 1650 pages.
 114649 bytes (224 pages) of virtual memory were used to buffer the intermediate code.
 There were 130 pages of symbol table space allocated to hold 2403 non-local and 2 local symbols.
 262 source lines were read in Pass 1, producing 13 object records in Pass 2.
 25 pages of virtual memory were used to define 24 macros.

! Macro library statistics !

Macro library name	Macros defined
-----	-----
-\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	2
-\$255\$DUA28:[SHRLIB]NMALIBRY.MLB;1	0
-\$255\$DUA28:[SHRLIB]EVCDEF.MLB;1	2
-\$255\$DUA28:[RMS.OBJ]RMS.MLB;1	7
-\$255\$DUA28:[SYSLIB]STARLET.MLB;2	9
TOTALS (all libraries)	20

2635 GETS were required to define 20 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LISS:NTODAPCRC/OBJ=OBJ\$:NTODAPCRC MSRC\$:NTODAPCRC/UPDATE=(ENH\$:NTODAPCRC)+LIB\$:RMS/LIB+SHRLIB\$:EVCDEF/LIB+SHRLIB\$:NMALIBRY

