```
   AAAAAAAAA     NNN         NNN     AAAAAAAAA     LLL                YYY         YYY  ZZZZZZZZZZZZZZZ
   AAAAAAAAA     NNN         NNN     AAAAAAAAA     LLL                YYY         YYY  ZZZZZZZZZZZZZZZ
   AAAAAAAAA     NNN         NNN     AAAAAAAAA     LLL                YYY         YYY  ZZZZZZZZZZZZZZZ
AAA         AAA  NNN         NNN  AAA         AAA  LLL                YYY         YYY              ZZZ
AAA         AAA  NNN         NNN  AAA         AAA  LLL                YYY         YYY              ZZZ
AAA         AAA  NNN         NNN  AAA         AAA  LLL                YYY         YYY              ZZZ
AAA         AAA  NNNNNN      NNN  AAA         AAA  LLL                   YYY   YYY                ZZZ
AAA         AAA  NNNNNN      NNN  AAA         AAA  LLL                   YYY   YYY                ZZZ
AAA         AAA  NNNNNN      NNN  AAA         AAA  LLL                   YYY   YYY                ZZZ
AAA         AAA  NNN  NNN   NNN  AAA         AAA  LLL                      YYY                  ZZZ
AAA         AAA  NNN  NNN   NNN  AAA         AAA  LLL                      YYY                  ZZZ
AAA         AAA  NNN  NNN   NNN  AAA         AAA  LLL                      YYY                  ZZZ
AAAAAAAAAAAAAAA  NNN     NNNNNN  AAAAAAAAAAAAAAA  LLL                      YYY               ZZZ
AAAAAAAAAAAAAAA  NNN     NNNNNN  AAAAAAAAAAAAAAA  LLL                      YYY               ZZZ
AAAAAAAAAAAAAAA  NNN     NNNNNN  AAAAAAAAAAAAAAA  LLL                      YYY               ZZZ
AAA         AAA  NNN         NNN  AAA         AAA  LLL                      YYY            ZZZ
AAA         AAA  NNN         NNN  AAA         AAA  LLL                      YYY            ZZZ
AAA         AAA  NNN         NNN  AAA         AAA  LLL                      YYY            ZZZ
AAA         AAA  NNN         NNN  AAA         AAA  LLLLLLLLLLLLLLL          YYY            ZZZZZZZZZZZZZZZ
AAA         AAA  NNN         NNN  AAA         AAA  LLLLLLLLLLLLLLL          YYY            ZZZZZZZZZZZZZZZ
AAA         AAA  NNN         NNN  AAA         AAA  LLLLLLLLLLLLLLL          YYY            ZZZZZZZZZZZZZZZ
```

FILEID**OBJTIR

```
  000000    BBBBBBBB          JJ  TTTTTTTTTT    IIIIII    RRRRRRR
  000000    BBBBBBBB          JJ  TTTTTTTTTT    IIIIII    RRRRRRRR
00      00  BB      BB        JJ      TT          II      RR      RR
00      00  BB      BB        JJ      TT          II      RR      RR
00      00  BB      BB        JJ      TT          II      RR      RR
00      00  BB      BB        JJ      TT          II      RR      RR
00      00  BBBBBBBB          JJ      TT          II      RRRRRRRR
00      00  BBBBBBBB          JJ      TT          II      RRRRRRRR
00      00  BB      BB  JJ    JJ      TT          II      RR  RR
00      00  BB      BB  JJ    JJ      TT          II      RR  RR
00      00  BB      BB  JJ    JJ      TT          II      RR    RR        ....
00      00  BB      BB  JJ    JJ      TT          II      RR    RR        ....
  000000    BBBBBBBB      JJJJJJ      TT      IIIIIII     RR      RR      ....
  000000    BBBBBBBB      JJJJJJ      TT      IIIIIII     RR      RR      ....
```

```
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              SS
LL                II              SS
LL                II              SS
LL                II              SS
LLLLLLLLLL      IIIIII    SSSSSSSS
LLLLLLLLLL      IIIIII    SSSSSSSS
```

```
;     1          0001  0 %title 'OBJTIR - Analyze TIR/DBG/TBT Object Records'
;     2          0002  0        module objtir   (
;     3          0003  1                        ident='V04-000') = begin
;     4          0004  1
;     5          0005  1 !
;     6          0006  1 !*****************************************************************************
;     7          0007  1 !*                                                                           *
;     8          0008  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;     9          0009  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;    10          0010  1 !*  ALL RIGHTS RESERVED.                                                      *
;    11          0011  1 !*                                                                           *
;    12          0012  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
;    13          0013  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
;    14          0014  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
;    15          0015  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
;    16          0016  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
;    17          0017  1 !*  TRANSFERRED.                                                              *
;    18          0018  1 !*                                                                           *
;    19          0019  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
;    20          0020  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
;    21          0021  1 !*  CORPORATION.                                                              *
;    22          0022  1 !*                                                                           *
;    23          0023  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
;    24          0024  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;    25          0025  1 !*                                                                           *
;    26          0026  1 !*                                                                           *
;    27          0027  1 !*****************************************************************************
;    28          0028  1 !
;    29          0029  1
;    30          0030  1 !++
;    31          0031  1 ! Facility:     VAX/VMS Analyze Facility, Analyze TIR/DBG/TBT Object Records
;    32          0032  1 !
;    33          0033  1 ! Abstract:     This module is responsible for analyzing the TIR and
;    34          0034  1 !               associated object records:
;    35          0035  1 !                       DBG     Debugger Information Records
;    36          0036  1 !                       TBT     Traceback Information Records
;    37          0037  1 !                       TIR     Text Information/Relocation Records
;    38          0038  1 !
;    39          0039  1 !
;    40          0040  1 ! Environment:
10  41          0041  1 !
15  42          0042  1 ! Author: Paul C. Anagnostopoulos, Creation Date: 16 January 1980
;    43          0043  1 !
16  44          0044  1 ! Modified By:
;    45          0045  1 !
;    46          0046  1 !       V03-002 MCN0158         Maria del C. Nasr       22-Mar-1984
20  47          0047  1 !               Add OBJ$C_SYMSIZ to call to ANL$CHECK_SYMBOL.
;    48          0048  1 !
;    49          0049  1 !       V03-001 PCA1011         Paul C. Anagnostopoulos  1-Apr-1983
;    50          0050  1 !               Change the message prefix to ANLOBJ$_ to ensure that
;    51          0051  1 !               message symbols are unique across all ANALYZEs.  This
;    52          0052  1 !               is necessitated by the new merged message files.
;    53          0053  1 !--
```

```
;    55          0054  1 %sbttl 'Module Declarations'
;    56          0055  1 !
;    57          0056  1 ! Libraries and Requires:
;    58          0057  1 !
;    59          0058  1
;    60          0059  1 library 'starlet';
;    61          0060  1 require 'objexereq';
;    62          0496  1
;    63          0497  1 !
;    64          0498  1 ! Table of Contents:
;    65          0499  1 !
;    66          0500  1
;    67          0501  1 forward routine
;    68          0502  1         anl$object_tir: novalue,
;    69          0503  1         anl$object_tir_clean: novalue;
;    70          0504  1
;    71          0505  1 !
;    72          0506  1 ! Macro Definitions:
;    73          0507  1 !
;    74          0508  1 ! The following macro is used to initialize one entry in the TIR command table.
;    75          0509  1
;    76        M 0510  1 macro cmd_def(command,operand,stack) =
;    77        M 0511  1         %if %identical(command,reserved) %then
;    78        M 0512  1                 rep (stack-operand+1)*8 of byte (0)
;    79        M 0513  1         %else
;    80        M 0514  1                 uplit byte(%ascic %string(command)),
;    81        M 0515  1                 byte (%exactstring(3,%x'00',%string(operand))),
;    82        M 0516  1                 byte (stack)
;    83        M 0517  1         %fi
;    84          0518  1 %;
;    85          0519  1
;    86          0520  1 !
;    87          0521  1 ! External References:
;    88          0522  1 !
;    89          0523  1
;    90          0524  1 external routine
;    91          0525  1         anl$check_symbol,
;    92          0526  1         anl$format_error,
;    93          0527  1         anl$format_hex,
;    94          0528  1         anl$format_line,
;    95          0529  1         anl$object_argument_dsc,
;    96          0530  1         anl$object_env_ref,
;    97          0531  1         anl$object_psect_ref,
;    98          0532  1         anl$object_record_line,
;    99          0533  1         anl$report_line;
;   100          0534  1
;   101          0535  1 !
;   102          0536  1 ! Own Variables:
;   103          0537  1 !
;   104          0538  1 ! The following variable keeps track of the stack depth as we analyze TIR
;   105          0539  1 ! commands.  It is cleared after each object module.
;   106          0540  1
;   107          0541  1 own
;   108          0542  1         stack_depth: long initial (0);
;   109          0543  1
;   110          0544  1 ! The following bit vector is needed to keep track of which literals are
;   111          0545  1 ! defined with the OPR_DFLIT command.
```

```
;  112          0546  1
;  113          0547  1 own
;  114          0548  1         literal_def_bits: bitvector[256] initial(rep 256/8 of byte (0));
```

```
;  116          0549  1 %sbttl 'ANL$OBJECT_TIR - Analyze TIR & Associated Object Records'
;  117          0550  1 !++
;  118          0551  1 ! Functional Description:
;  119          0552  1 !       This routine is responsible for analyzing the TIR, DBG, and TBT
;  120          0553  1 !       object records.
;  121          0554  1 !
;  122          0555  1 ! Formal Parameters:
;  123          0556  1 !       record_number   The number of this object record.
;  124          0557  1 !       the_record      Address of descriptor of the object record.
;  125          0558  1 !
;  126          0559  1 ! Implicit Inputs:
;  127          0560  1 !       global data
;  128          0561  1 !
;  129          0562  1 ! Implicit Outputs:
;  130          0563  1 !       global data
;  131          0564  1 !
;  132          0565  1 ! Returned Value:
;  133          0566  1 !       none
;  134          0567  1 !
;  135          0568  1 ! Side Effects:
;  136          0569  1 !
;  137          0570  1 !--
;  138          0571  1
;  139          0572  1
;  140          0573  2 global routine anl$object_tir(record_number,the_record): novalue = begin
;  141          0574  2
;  142          0575  2 bind
;  143          0576  2         record_dsc = .the_record: descriptor;
```

```
;  145          0577  2 ! The following table defines all of the commands valid in these records
;  146          0578  2 ! (except the store immediate command).  Each entry includes the symbolic
;  147          0579  2 ! name of the command, the operand type (see below), and the number of
;  148          0580  2 ! longwords pushed on or popped from the stack.
;  149          0581  2
;  150          0582  2 field cmd_table_fields = set
;  151          0583  2         command_name    = [0,0,32,0],
;  152          0584  2         operand_type    = [4,0,24,0],
;  153          0585  2         stack_affect    = [7,0, 8,1]
;  154          0586  2 tes;
;  155          0587  2
;  156          0588  2 own
;  157          0589  2         cmd_table: blockvector[128,8,byte] field(cmd_table_fields)
;  158          0590  2                 initial (
;  159          0591  2
;  160          0592  2 !                       COMMAND NAME    OPERAND TYPE    STACK AFFECT
;  161          0593  2 !                       ------------    ------------    ------------
;  162          0594  2
;  163          0595  2                 cmd_def(STA_GBL,        SYM,            +1),
;  164          0596  2                 cmd_def(STA_SB,         B,              +1),
;  165          0597  2                 cmd_def(STA_SW,         W,              +1),
;  166          0598  2                 cmd_def(STA_LW,         L,              +1),
;  167          0599  2                 cmd_def(STA_PB,         BPB,            +1),
;  168          0600  2                 cmd_def(STA_PW,         BPW,            +1),
;  169          0601  2                 cmd_def(STA_PL,         BPL,            +1),
;  170          0602  2                 cmd_def(STA_UB,         UB,             +1),
;  171          0603  2                 cmd_def(STA_UW,         UW,             +1),
;  172          0604  2                 cmd_def(STA_BFI,        NO,             +1),
;  173          0605  2                 cmd_def(STA_WFI,        NO,             +1),
;  174          0606  2                 cmd_def(STA_LFI,        NO,             +1),
;  175          0607  2                 cmd_def(STA_EPM,        SYM,            +1),
;  176          0608  2                 cmd_def(STA_CKARG,      ARG,            +1),
;  177          0609  2                 cmd_def(STA_WPB,        WPB,            +1),
;  178          0610  2                 cmd_def(STA_WPW,        WPW,            +1),
;  179          0611  2                 cmd_def(STA_WPL,        WPL,            +1),
;  180          0612  2                 cmd_def(STA_LSY,        ENS,            +1),
;  181          0613  2                 cmd_def(STA_LIT,        LTX,            +1),
;  182          0614  2                 cmd_def(reserved, 19,19),
;  183          0615  2
;  184          0616  2                 cmd_def(STO_SB,         NO,             -1),
;  185          0617  2                 cmd_def(STO_SW,         NO,             -1),
;  186          0618  2                 cmd_def(STO_LW,         NO,             -1),
;  187          0619  2                 cmd_def(STO_BD,         NO,             -1),
;  188          0620  2                 cmd_def(STO_WD,         NO,             -1),
;  189          0621  2                 cmd_def(STO_LD,         NO,             -1),
;  190          0622  2                 cmd_def(STO_LI,         NO,             -1),
;  191          0623  2                 cmd_def(STO_PIDR,       NO,             -1),
;  192          0624  2                 cmd_def(STO_PICR,       NO,             -1),
;  193          0625  2                 cmd_def(STO_RSB,        NO,             -2),
;  194          0626  2                 cmd_def(STO_RSW,        NO,             -2),
;  195          0627  2                 cmd_def(STO_RL,         NO,             -2),
;  196          0628  2                 cmd_def(STO_VPS,        VLD,            -1),
;  197          0629  2                 cmd_def(STO_USB,        NO,             -1),
;  198          0630  2                 cmd_def(STO_USW,        NO,             -1),
;  199          0631  2                 cmd_def(STO_RUB,        NO,             -2),
;  200          0632  2                 cmd_def(STO_RUW,        NO,             -2),
;  201          0633  2                 cmd_def(STO_B,          NO,             -1),
```

```
;  202     0634  2                    cmd_def(STO_W,          NO,          -1),
;  203     0635  2                    cmd_def(STO_RB,         NO,          -2),
;  204     0636  2                    cmd_def(STO_RW,         NO,          -2),
;  205     0637  2                    cmd_def(STO_RIVB,       TXT,         -1),
;  206     0638  2                    cmd_def(STO_PIRR,       NO,          -2),
;  207     0639  2                    cmd_def(reserved, 43,49),
;  208     0640  2
;  209     0641  2                    cmd_def(OPR_NOP,        NO,           0),
;  210     0642  2                    cmd_def(OPR_ADD,        NO,          -1),
;  211     0643  2                    cmd_def(OPR_SUB,        NO,          -1),
;  212     0644  2                    cmd_def(OPR_MUL,        NO,          -1),
;  213     0645  2                    cmd_def(OPR_DIV,        NO,          -1),
;  214     0646  2                    cmd_def(OPR_AND,        NO,          -1),
;  215     0647  2                    cmd_def(OPR_IOR,        NO,          -1),
;  216     0648  2                    cmd_def(OPR_EOR,        NO,          -1),
;  217     0649  2                    cmd_def(OPR_NEG,        NO,           0),
;  218     0650  2                    cmd_def(OPR_COM,        NO,           0),
;  219     0651  2                    cmd_def(OPR_INSV,       VLD,         -1),
;  220     0652  2                    cmd_def(OPR_ASH,        NO,          -1),
;  221     0653  2                    cmd_def(OPR_USH,        NO,          -1),
;  222     0654  2                    cmd_def(OPR_ROT,        NO,          -1),
;  223     0655  2                    cmd_def(OPR_SEL,        NO,          -2),
;  224     0656  2                    cmd_def(OPR_REDEF,      SYM,          0),
;  225     0657  2                    cmd_def(OPR_DFLIT,      LTX,         -1),
;  226     0658  2                    cmd_def(reserved, 67,79),
;  227     0659  2
;  228     0660  2                    cmd_def(CTL_SETRB,      NO,          -1),
;  229     0661  2                    cmd_def(CTL_AUGRB,      L,            0),
;  230     0662  2                    cmd_def(CTL_DFLOC,      NO,          -1),
;  231     0663  2                    cmd_def(CTL_STLOC,      NO,          -1),
;  232     0664  2                    cmd_def(CTL_STKDL,      NO,           0),
;  233     0665  2                    cmd_def(reserved, 85,127)
;  234     0666  2
;  235     0667  2                    );
;  236     0668  2
;  237     0669  2 ! The following list defines the operand type codes used in the table above.
;  238     0670  2 ! Each one specifies a different combination of operands that can follow
;  239     0671  2 ! a command operation code.
;  240     0672  2 !
;  241     0673  2 !       ARG     symbol, byte argument index, argument descriptor
;  242     0674  2 !       B       signed byte
;  243     0675  2 !       BP      byte psect number
;  244     0676  2 !       BPB     byte psect number, signed byte
;  245     0677  2 !       BPL     byte psect number, signed longword
;  246     0678  2 !       BPW     byte psect number, signed word
;  247     0679  2 !       ENS     word environment number, symbol
;  248     0680  2 !       L       signed longword
;  249     0681  2 !       LTX     byte literal index
;  250     0682  2 !       SYM     symbol
;  251     0683  2 !       TXT     ASCIC text string
;  252     0684  2 !       UB      unsigned byte
;  253     0685  2 !       UW      unsigned word
;  254     0686  2 !       VLD     byte position, byte size
;  255     0687  2 !       W       signed word
;  256     0688  2 !       WP      word psect number
;  257     0689  2 !       WPB     word psect number, signed byte
;  258     0690  2 !       WPL     word psect number, signed longword
```

```
;  259          0691  2 !       WPW     word psect number, signed word
```

```
;   261     0692  2 local
;   262     0693  2         scanp: ref block[,byte],
;   263     0694  2         record_type: byte,
;   264     0695  2         command_number: long,
;   265     0696  2         fit_ok: byte,
;   266     0697  2         command: signed byte,
;   267     0698  2         work_dsc: descriptor,
;   268     0699  2         literal_index: byte;
;   269     0700  2
;   270     0701  2
;   271     0702  2 ! We begin by printing a major line for the record.
;   272     0703  2
;   273     0704  2 scanp = .record_dsc[ptr];
;   274     0705  2 record_type = .scanp[obj$b_rectyp];
;   275     0706  3 anl$object_record_line((selectoneu .record_type of set
;   276     0707  3                         [obj$c_tir]:    anlobj$_objtirrec;
;   277     0708  3                         [obj$c_dbg]:    anlobj$_objdbgrec;
;   278     0709  3                         [obj$c_tbt]:    anlobj$_objtbtrec;
;   279     0710  2                         tes),
;   280     0711  2                         .record_number,record_dsc);
;   281     0712  2 increment (scanp);
;   282     0713  2
;   283     0714  2 ! Now we go into a loop processing the commands in the record.
;   284     0715  2 ! COMMAND_NUMBER will count them as we go.
;   285     0716  2 ! SCANP will advance along the various commands and fields of the record.
;   286     0717  2 ! FIT_OK will remain true unless a field spills off the end of the record.
;   287     0718  2
;   288     0719  2 command_number = 0;
;   289     0720  2 fit_ok = true;
;   290     0721  3 while (.scanp lssa (.record_dsc[ptr]+.record_dsc[len])) and .fit_ok do (
;   291     0722  3
;   292     0723  3         ! Count the command and prepare to print it nicely.
;   293     0724  3
;   294     0725  3         increment (command_number);
;   295     0726  3         anl$report_line(0);
;   296     0727  3
;   297     0728  3         ! We split up depending upon whether it is a store immediate
;   298     0729  3         ! command or some other one.
;   299     0730  3
;   300     0731  3         command = .scanp[0,0,8,1];
;   301     0732  3         increment (scanp);
;   302     0733  4         if .command lss 0 then (
;   303     0734  4
;   304     0735  4                 ! It's a store immediate.  Print a line for the command,
;   305     0736  4                 ! and then dump the text.
;   306     0737  4
;   307     0738  4                 anl$format_line(1,1,anlobj$_objtirstoim,.command_number,-.command);
;   308     0739  4                 build_descriptor(work_dsc,-.command,.scanp);
;   309     0740  4                 anl$format_hex(2,work_dsc);
;   310     0741  4                 scanp = .scanp + -.command;
;   311     0742  4
;   312     0743  4         ) else (
;   313     0744  4
;   314     0745  4                 ! It's some other command.  If it's invalid, just tell the
;   315     0746  4                 ! user.  We also have to ignore the rest of the
;   316     0747  4                 ! record since God only knows what it looks like.
;   317     0748  4
```

```
;  318     0749  5                    if .cmd_table[.command,command_name] eqlu 0 then (
;  319     0750  5                            anl$format_error(anlobj$_objtirres,.command);
;  320     0751  5                            build_descriptor(work_dsc,.record_dsc[len]-(.scanp-.record_dsc[ptr]),.record_dsc[ptr
;  321     0752  5                            anl$format_hex(2,work_dsc);
;  322     0753  5                            return;
;  323     0754  5                    ) else (
;  324     0755  5
;  325     0756  5                            ! It's a good command.  Adjust the stack and then
;  326     0757  5                            ! print a line for it.  We use a different line if
;  327     0758  5                            ! it changes the stack depth.
;  328     0759  5
;  329     0760  5                            stack_depth = .stack_depth + .cmd_table[.command,stack_affect];
;  330     0761  5                            anl$format_line(2,1,
;  331     0762  6                                    (if .cmd_table[.command,stack_affect] eqlu 0 then anlobj$_objtircmd
;  332     0763  5                                                                                 else anlobj$_objtircmdstk),
;  333     0764  5                                    .command_number,.cmd_table[.command,command_name],
;  334     0765  5                                    .command,.stack_depth);
;  335     0766  4                    );
```

```
;  337     0767  4                    ! Now we select on the operand type for this command.  This
;  338     0768  4                    ! will tell us how to print the operands, and also let us
;  339     0769  4                    ! adjust the scan pointer to the next command.  NOTE that we
;  340     0770  4                    ! use selectu so that operand types can make use of more
;  341     0771  4                    ! than one case.
;  342     0772  4
;  343     0773  4                    selectu .cmd_table[.command,operand_type] of set
;  344     0774  4
;  345     0775  4
;  346     0776  4                    ['BP',
;  347     0777  4                     'BPB',
;  348     0778  4                     'BPW',
;  349     0779  4                     'BPL']:        ! The command takes a byte psect number,
;  350     0780  4                                    ! perhaps followed by something else.  Make
;  351     0781  4                                    ! sure to record the psect reference.
;  352     0782  4
;  353     0783  5                                    (ensure_field_fit(0,0,8,0,record_dsc);
;  354     0784  6                                    if .fit_ok then (
;  355     0785  6                                            anl$format_line(0,2,anlobj$_objpsect,.scanp[0,0,8,0]);
;  356     0786  6                                            anl$object_psect_ref(.scanp[0,0,8,0]);
;  357     0787  6                                            increment (scanp);
;  358     0788  4                                    ););
;  359     0789  4
;  360     0790  4
;  361     0791  4                    ['WP',
;  362     0792  4                     'WPB',
;  363     0793  4                     'WPW',
;  364     0794  4                     'WPL']:        ! The command takes a word psect number,
;  365     0795  4                                    ! perhaps followed by something else.
;  366     0796  4
;  367     0797  5                                    (ensure_field_fit(0,0,16,0,record_dsc);
;  368     0798  6                                    if .fit_ok then (
;  369     0799  6                                            anl$format_line(0,2,anlobj$_objpsect,.scanp[0,0,16,0]);
;  370     0800  6                                            anl$object_psect_ref(.scanp[0,0,16,0]);
;  371     0801  6                                            scanp = .scanp + 2;
;  372     0802  4                                    ););
;  373     0803  4
;  374     0804  4
;  375     0805  4                    ['ENS']:       ! The command takes a word environment number,
;  376     0806  4                                    ! perhaps followed by something else.
;  377     0807  4
;  378     0808  5                                    (ensure_field_fit(0,0,16,0,record_dsc);
;  379     0809  6                                    if .fit_ok then (
;  380     0810  6                                            anl$format_line(0,2,anlobj$_objenv,.scanp[0,0,16,0]);
;  381     0811  6                                            anl$object_env_ref(.scanp[0,0,16,0]);
;  382     0812  6                                            scanp = .scanp + 2;
;  383     0813  4                                    ););
;  384     0814  4
;  385     0815  4                    ['B',
;  386     0816  4                     'BPB',
;  387     0817  4                     'WPB']:        ! The command takes a signed byte operand,
;  388     0818  4                                    ! perhaps preceeded by something else.
;  389     0819  4
;  390     0820  5                                    (ensure_field_fit(0,0,8,1,record_dsc);
;  391     0821  6                                    if .fit_ok then (
;  392     0822  6                                            anl$format_line(0,2,anlobj$_objvalue,.scanp[0,0,8,1]);
;  393     0823  6                                            increment (scanp);
```

```
;  394     0824  4                                    ););
;  395     0825  4
;  396     0826  4
;  397     0827  4                ['UB']:             ! The command takes an unsigned byte operand.
;  398     0828  4
;  399     0829  5                                    (ensure_field_fit(0,0,8,0,record_dsc);
;  400     0830  6                                    if .fit_ok then (
;  401     0831  6                                            anl$format_line(0,2,anlobj$_objuvalue,.scanp[0,0,8,0]);
;  402     0832  6                                            increment (scanp);
;  403     0833  4                                    ););
;  404     0834  4
;  405     0835  4
;  406     0836  4                ['LTX']:            ! The command takes a byte literal index.  We
;  407     0837  4                                    ! need to save it for later checking.
;  408     0838  4
;  409     0839  5                                    (ensure_field_fit(0,0,8,0,record_dsc);
;  410     0840  6                                    if .fit_ok then (
;  411     0841  6                                            anl$format_line(0,2,anlobj$_objlitindex,.scanp[0,0,8,0]);
;  412     0842  6                                            literal_index = .scanp[0,0,8,0];
;  413     0843  6                                            increment (scanp);
;  414     0844  4                                    ););
;  415     0845  4
;  416     0846  4
;  417     0847  4                ['W',
;  418     0848  4                 'BPW',
;  419     0849  4                 'WPW']:            ! The command takes a signed word operand,
;  420     0850  4                                    ! perhaps preceeded by something else.
;  421     0851  4
;  422     0852  5                                    (ensure_field_fit(0,0,16,1,record_dsc);
;  423     0853  6                                    if .fit_ok then (
;  424     0854  6                                            anl$format_line(0,2,anlobj$_objvalue,.scanp[0,0,16,1]);
;  425     0855  6                                            scanp = .scanp + 2;
;  426     0856  4                                    ););
;  427     0857  4
;  428     0858  4
;  429     0859  4                ['UW']:             ! The command takes an unsigned word operand.
;  430     0860  4
;  431     0861  4
;  432     0862  5                                    (ensure_field_fit(0,0,16,0,record_dsc);
;  433     0863  6                                    if .fit_ok then (
;  434     0864  6                                            anl$format_line(0,2,anlobj$_objuvalue,.scanp[0,0,16,0]);
;  435     0865  6                                            scanp = .scanp + 2;
;  436     0866  4                                    ););
;  437     0867  4
;  438     0868  4
;  439     0869  4                ['L',
;  440     0870  4                 'BPL',
;  441     0871  4                 'WPL']:            ! The command takes a signed longword operand,
;  442     0872  4                                    ! perhaps preceeded by somthing else.
;  443     0873  4
;  444     0874  4
;  445     0875  5                                    (ensure_field_fit(0,0,32,1,record_dsc);
;  446     0876  6                                    if .fit_ok then (
;  447     0877  6                                            anl$format_line(0,2,anlobj$_objvalue,.scanp[0,0,32,1]);
;  448     0878  6                                            scanp = .scanp + 4;
;  449     0879  4                                    ););
;  450     0880  4
```

```
;  451     0881  4
;  452     0882  4                 ['SYM',
;  453     0883  4                  'ARG',
;  454     0884  4                  'ENS']:         ! The command takes a symbol name, perhaps
;  455     0885  4                                  ! preceeded or followed by something else.
;  456     0886  4
;  457     0887  5                                  (ensure_ascic_fit(0,0,8,0,record_dsc,work_dsc);
;  458     0888  6                                  if .fit_ok then (
;  459     0889  6                                          anl$format_line(0,2,anlobj$_objsymbol,.work_dsc[len],.work_dsc[ptr])
;  460     0890  6                                          anl$check_symbol(work_dsc, obj$c_symsiz);
;  461     0891  6                                          scanp = .scanp + 1 + .work_dsc[len];
;  462     0892  4                                  ););
;  463     0893  4
;  464     0894  4
;  465     0895  4                 ['ARG']:        ! The command takes an argument descriptor.
;  466     0896  4                                  ! This consists of a symbol, which was analyzed
;  467     0897  4                                  ! above, an argument index, and an actual
;  468     0898  4                                  ! argument descriptor.
;  469     0899  4
;  470     0900  5                                  (ensure_field_fit(0,0,8,0,record_dsc);
;  471     0901  6                                  if .fit_ok then (
;  472     0902  6                                          anl$format_line(0,2,anlobj$_objtirargindex,.scanp[0,0,8,0]);
;  473     0903  6                                          increment (scanp);
;  474     0904  6                                          fit_ok = anl$object_argument_dsc(3,scanp,record_dsc);
;  475     0905  4                                  ););
;  476     0906  4
;  477     0907  4
;  478     0908  4                 ['TXT']:        ! The command takes a counted string with text.
;  479     0909  4
;  480     0910  5                                  (ensure_ascic_fit(0,0,8,0,record_dsc,work_dsc);
;  481     0911  6                                  if .fit_ok then (
;  482     0912  6                                          anl$format_hex(2,work_dsc);
;  483     0913  6                                          scanp = .scanp + 1 + .work_dsc[len];
;  484     0914  4                                  ););
;  485     0915  4
;  486     0916  4
;  487     0917  4                 ['VLD']:        ! The command takes a vield definition,
;  488     0918  4                                  ! consisting of a position and a size.
;  489     0919  4
;  490     0920  5                                  (ensure_field_fit(0,0,16,0,record_dsc);
;  491     0921  6                                  if .fit_ok then (
;  492     0922  6                                          anl$format_line(0,2,anlobj$_objtirvield,.scanp[0,0,8,0],.scanp[1,0,8
;  493     0923  6                                          if .scanp[0,0,8,0]+.scanp[1,0,8,0] gtru
;  494     0924  6                                             minu(31,(.scanp[0,0,8,0]+8)/8*8) then
;  495     0925  6                                                  anl$format_error(anlobj$_objbadvield);
;  496     0926  6                                          scanp = .scanp + 2;
;  497     0927  4                                  ););
;  498     0928  4
;  499     0929  4                 tes;
;  500     0930  4
;  501     0931  4                 ! Check the stack to ensure that we haven't popped too much.
;  502     0932  4
;  503     0933  4                 if .stack_depth lss 0 then
;  504     0934  4                         anl$format_error(anlobj$_objbadpop);
```

```
;   506          0935  4                    ! Now we have to perform some checks that are specific to
;   507          0936  4                    ! various commands.  Just select on the command and make
;   508          0937  4                    ! the check.
;   509          0938  4
;   510          0939  4                    selectoneu .command of set
;   511          0940  4                    [tir$c_sta_lit]:
;   512          0941  4
;   513          0942  4                            ! We have a stack literal value command.  Make sure
;   514          0943  4                            ! the literal has been defined.
;   515          0944  4
;   516          0945  4                            if not .literal_def_bits[.literal_index] then
;   517          0946  4                                    anl$format_error(anlobj$_objundeflit);
;   518          0947  4
;   519          0948  4                    [tir$c_opr_dflit]:
;   520          0949  4
;   521          0950  4                            ! We have a define literal command.  Set the defined
;   522          0951  4                            ! bit for the literal.
;   523          0952  4
;   524          0953  4                            literal_def_bits[.literal_index] = true;
;   525          0954  4
;   526          0955  4                    [tir$c_ctl_dfloc,
;   527          0956  4                     tir$c_ctl_stloc,
;   528          0957  4                     tir$c_ctl_stkdl]:
;   529          0958  4
;   530          0959  4                            ! We have a command that is not valid in TIR records,
;   531          0960  4                            ! only in DBG and TBT records.
;   532          0961  4
;   533          0962  4                            if .record_type eqlu obj$c_tir then
;   534          0963  4                                    anl$format_error(anlobj$_objnontircmd);
;   535          0964  4                    tes;
;   536          0965  4
;   537          0966  4            ! Analysis of command is complete.
;   538          0967  4
;   539          0968  3            );
;   540          0969  3
;   541          0970  3 ! Go on to next command.
;   542          0971  3
;   543          0972  2 );
;   544          0973  2
;   545          0974  2 return;
;   546          0975  2
;   547          0976  1 end;


                                                                .TITLE  OBJTIR OBJTIR - Analyze TIR/DBG/TBT Object Reco
;                                                                       rds
                                                                .IDENT  \V04-000\

                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

                                4C  42  47  5F  41  54  53  07  00000 P.AAA:   .ASCII  <7>\STA_GBL\                          ;
                                    42  53  5F  41  54  53  06  00008 P.AAB:   .ASCII  <6>\STA_SB\                           ;
                                    57  53  5F  41  54  53  06  0000F P.AAC:   .ASCII  <6>\STA_SW\                           ;
                                    57  4C  5F  41  54  53  06  00016 P.AAD:   .ASCII  <6>\STA_LW\                           ;
                                    42  50  5F  41  54  53  06  0001D P.AAE:   .ASCII  <6>\STA_PB\                           ;
                                    57  50  5F  41  54  53  06  00024 P.AAF:   .ASCII  <6>\STA_PW\                           ;
                                    4C  50  5F  41  54  53  06  0002B P.AAG:   .ASCII  <6>\STA_PL\                           ;
```

```
                                42  55  5F  41  54  53  06  00032 P.AAH:   .ASCII  <6>\STA_UB\
                                57  55  5F  41  54  53  06  00039 P.AAI:   .ASCII  <6>\STA_UW\
                            49  46  42  5F  41  54  53  07  00040 P.AAJ:   .ASCII  <7>\STA_BFI\
                            49  46  57  5F  41  54  53  07  00048 P.AAK:   .ASCII  <7>\STA_WFI\
                            49  46  4C  5F  41  54  53  07  00050 P.AAL:   .ASCII  <7>\STA_LFI\
                            4D  50  45  5F  41  54  53  07  00058 P.AAM:   .ASCII  <7>\STA_EPM\
                        47  52  41  4B  43  5F  41  54  53  09  00060 P.AAN:   .ASCII  <9>\STA_CKARG\
                            42  50  57  5F  41  54  53  07  0006A P.AAO:   .ASCII  <7>\STA_WPB\
                            57  50  57  5F  41  54  53  07  00072 P.AAP:   .ASCII  <7>\STA_WPW\
                            4C  50  57  5F  41  54  53  07  0007A P.AAQ:   .ASCII  <7>\STA_WPL\
                            59  53  4C  5F  41  54  53  07  00082 P.AAR:   .ASCII  <7>\STA_LSY\
                            54  49  4C  5F  41  54  53  07  0008A P.AAS:   .ASCII  <7>\STA_LIT\
                                42  53  5F  4F  54  53  06  00092 P.AAT:   .ASCII  <6>\STO_SB\
                                57  53  5F  4F  54  53  06  00099 P.AAU:   .ASCII  <6>\STO_SW\
                                57  4C  5F  4F  54  53  06  000A0 P.AAV:   .ASCII  <6>\STO_LW\
                                44  42  5F  4F  54  53  06  000A7 P.AAW:   .ASCII  <6>\STO_BD\
                                44  57  5F  4F  54  53  06  000AE P.AAX:   .ASCII  <6>\STO_WD\
                                44  4C  5F  4F  54  53  06  000B5 P.AAY:   .ASCII  <6>\STO_LD\
                                49  4C  5F  4F  54  53  06  000BC P.AAZ:   .ASCII  <6>\STO_LI\
                            52  44  49  50  5F  4F  54  53  08  000C3 P.ABA:   .ASCII  <8>\STO_PIDR\
                            52  43  49  50  5F  4F  54  53  08  000CC P.ABB:   .ASCII  <8>\STO_PICR\
                            42  53  52  5F  4F  54  53  07  000D5 P.ABC:   .ASCII  <7>\STO_RSB\
                            57  53  52  5F  4F  54  53  07  000DD P.ABD:   .ASCII  <7>\STO_RSW\
                                4C  52  5F  4F  54  53  06  000E5 P.ABE:   .ASCII  <6>\STO_RL\
                            53  50  56  5F  4F  54  53  07  000EC P.ABF:   .ASCII  <7>\STO_VPS\
                            42  53  55  5F  4F  54  53  07  000F4 P.ABG:   .ASCII  <7>\STO_USB\
                            57  53  55  5F  4F  54  53  07  000FC P.ABH:   .ASCII  <7>\STO_USW\
                            42  55  52  5F  4F  54  53  07  00104 P.ABI:   .ASCII  <7>\STO_RUB\
                            57  55  52  5F  4F  54  53  07  0010C P.ABJ:   .ASCII  <7>\STO_RUW\
                                    42  5F  4F  54  53  05  00114 P.ABK:   .ASCII  <5>\STO_B\
                                    57  5F  4F  54  53  05  0011A P.ABL:   .ASCII  <5>\STO_W\
                                42  52  5F  4F  54  53  06  00120 P.ABM:   .ASCII  <6>\STO_RB\
                                57  52  5F  4F  54  53  06  00127 P.ABN:   .ASCII  <6>\STO_RW\
                        42  56  49  52  5F  4F  54  53  08  0012E P.ABO:   .ASCII  <8>\STO_RIVB\
                        52  52  49  50  5F  4F  54  53  08  00137 P.ABP:   .ASCII  <8>\STO_PIRR\
                            50  4F  4E  5F  52  50  4F  07  00140 P.ABQ:   .ASCII  <7>\OPR_NOP\
                            44  44  41  5F  52  50  4F  07  00148 P.ABR:   .ASCII  <7>\OPR_ADD\
                            42  55  53  5F  52  50  4F  07  00150 P.ABS:   .ASCII  <7>\OPR_SUB\
                            4C  55  4D  5F  52  50  4F  07  00158 P.ABT:   .ASCII  <7>\OPR_MUL\
                            56  49  44  5F  52  50  4F  07  00160 P.ABU:   .ASCII  <7>\OPR_DIV\
                            44  4E  41  5F  52  50  4F  07  00168 P.ABV:   .ASCII  <7>\OPR_AND\
                            52  4F  49  5F  52  50  4F  07  00170 P.ABW:   .ASCII  <7>\OPR_IOR\
                            52  4F  45  5F  52  50  4F  07  00178 P.ABX:   .ASCII  <7>\OPR_EOR\
                            47  45  4E  5F  52  50  4F  07  00180 P.ABY:   .ASCII  <7>\OPR_NEG\
                            4D  4F  43  5F  52  50  4F  07  00188 P.ABZ:   .ASCII  <7>\OPR_COM\
                        56  53  4E  49  5F  52  50  4F  08  00190 P.ACA:   .ASCII  <8>\OPR_INSV\
                            48  53  41  5F  52  50  4F  07  00199 P.ACB:   .ASCII  <7>\OPR_ASH\
                            48  53  55  5F  52  50  4F  07  001A1 P.ACC:   .ASCII  <7>\OPR_USH\
                            54  4F  52  5F  52  50  4F  07  001A9 P.ACD:   .ASCII  <7>\OPR_ROT\
                            4C  45  53  5F  52  50  4F  07  001B1 P.ACE:   .ASCII  <7>\OPR_SEL\
                46  45  44  45  52  5F  52  50  4F  09  001B9 P.ACF:   .ASCII  <9>\OPR_REDEF\
                54  49  4C  46  44  5F  52  50  4F  09  001C3 P.ACG:   .ASCII  <9>\OPR_DFLIT\
                42  52  54  45  53  5F  4C  54  43  09  001CD P.ACH:   .ASCII  <9>\CTL_SETRB\
                42  52  47  55  41  5F  4C  54  43  09  001D7 P.ACI:   .ASCII  <9>\CTL_AUGRB\
                43  4F  4C  46  44  5F  4C  54  43  09  001E1 P.ACJ:   .ASCII  <9>\CTL_DFLOC\
                43  4F  4C  54  53  5F  4C  54  43  09  001EB P.ACK:   .ASCII  <9>\CTL_STLOC\
                4C  44  4B  54  53  5F  4C  54  43  09  001F5 P.ACL:   .ASCII  <9>\CTL_STKDL\
```

```
                                        .PSECT  $OWN$,NOEXE,2

  00000000  00000 STACK_DEPTH:
                                        .LONG   0                                       ;
        00#  00004 LITERAL_DEF_BITS:
                                        .BYTE   0[32]                                   ;
  00000000'  00024 CMD_TABLE:
                                        .ADDRESS P.AAA
4D  59  53  00028                       .ASCII  \SYM\
        01  0002B                       .BYTE   1
  00000000' 0002C                       .ADDRESS P.AAB
00  00  42  00030                       .ASCII  \B\<0><0>
        01  00033                       .BYTE   1
  00000000' 00034                       .ADDRESS P.AAC
00  00  57  00038                       .ASCII  \W\<0><0>
        01  0003B                       .BYTE   1
  00000000' 0003C                       .ADDRESS P.AAD
00  00  4C  00040                       .ASCII  \L\<0><0>
        01  00043                       .BYTE   1
  00000000' 00044                       .ADDRESS P.AAE
42  50  42  00048                       .ASCII  \BPB\
        01  0004B                       .BYTE   1
  00000000' 0004C                       .ADDRESS P.AAF
57  50  42  00050                       .ASCII  \BPW\
        01  00053                       .BYTE   1
  00000000' 00054                       .ADDRESS P.AAG
4C  50  42  00058                       .ASCII  \BPL\
        01  0005B                       .BYTE   1
  00000000' 0005C                       .ADDRESS P.AAH
00  42  55  00060                       .ASCII  \UB\<0>
        01  00063                       .BYTE   1
  00000000' 00064                       .ADDRESS P.AAI
00  57  55  00068                       .ASCII  \UW\<0>
        01  0006B                       .BYTE   1
  00000000' 0006C                       .ADDRESS P.AAJ
00  4F  4E  00070                       .ASCII  \NO\<0>
        01  00073                       .BYTE   1
  00000000' 00074                       .ADDRESS P.AAK
00  4F  4E  00078                       .ASCII  \NO\<0>
        01  0007B                       .BYTE   1
  00000000' 0007C                       .ADDRESS P.AAL
00  4F  4E  00080                       .ASCII  \NO\<0>
        01  00083                       .BYTE   1
  00000000' 00084                       .ADDRESS P.AAM
4D  59  53  00088                       .ASCII  \SYM\
        01  0008B                       .BYTE   1
  00000000' 0008C                       .ADDRESS P.AAN
47  52  41  00090                       .ASCII  \ARG\
        01  00093                       .BYTE   1
  00000000' 00094                       .ADDRESS P.AAO
42  50  57  00098                       .ASCII  \WPB\
        01  0009B                       .BYTE   1
  00000000' 0009C                       .ADDRESS P.AAP
57  50  57  000A0                       .ASCII  \WPW\
        01  000A3                       .BYTE   1
  00000000' 000A4                       .ADDRESS P.AAQ
```

```
4C 50 57   000A8      .ASCII  \WPL\
       01  000AB      .BYTE   1
 00000000' 000AC      .ADDRESS P.AAR
53 4E 45   000B0      .ASCII  \ENS\
       01  000B3      .BYTE   1
 00000000' 000B4      .ADDRESS P.AAS
58 54 4C   000B8      .ASCII  \LTX\
       01  000BB      .BYTE   1
      00#  000BC      .BYTE   0[8]
 00000000' 000C4      .ADDRESS P.AAT
00 4F 4E   000C8      .ASCII  \NO\<0>
       FF  000CB      .BYTE   -1
 00000000' 000CC      .ADDRESS P.AAU
00 4F 4E   000D0      .ASCII  \NO\<0>
       FF  000D3      .BYTE   -1
 00000000' 000D4      .ADDRESS P.AAV
00 4F 4E   000D8      .ASCII  \NO\<0>
       FF  000DB      .BYTE   -1
 00000000' 000DC      .ADDRESS P.AAW
00 4F 4E   000E0      .ASCII  \NO\<0>
       FF  000E3      .BYTE   -1
 00000000' 000E4      .ADDRESS P.AAX
00 4F 4E   000E8      .ASCII  \NO\<0>
       FF  000EB      .BYTE   -1
 00000000' 000EC      .ADDRESS P.AAY
00 4F 4E   000F0      .ASCII  \NO\<0>
       FF  000F3      .BYTE   -1
 00000000' 000F4      .ADDRESS P.AAZ
00 4F 4E   000F8      .ASCII  \NO\<0>
       FF  000FB      .BYTE   -1
 00000000' 000FC      .ADDRESS P.ABA
00 4F 4E   00100      .ASCII  \NO\<0>
       FF  00103      .BYTE   -1
 00000000' 00104      .ADDRESS P.ABB
00 4F 4E   00108      .ASCII  \NO\<0>
       FF  0010B      .BYTE   -1
 00000000' 0010C      .ADDRESS P.ABC
00 4F 4E   00110      .ASCII  \NO\<0>
       FE  00113      .BYTE   -2
 00000000' 00114      .ADDRESS P.ABD
00 4F 4E   00118      .ASCII  \NO\<0>
       FE  0011B      .BYTE   -2
 00000000' 0011C      .ADDRESS P.ABE
00 4F 4E   00120      .ASCII  \NO\<0>
       FE  00123      .BYTE   -2
 00000000' 00124      .ADDRESS P.ABF
44 4C 56   00128      .ASCII  \VLD\
       FF  0012B      .BYTE   -1
 00000000' 0012C      .ADDRESS P.ABG
00 4F 4E   00130      .ASCII  \NO\<0>
       FF  00133      .BYTE   -1
 00000000' 00134      .ADDRESS P.ABH
00 4F 4E   00138      .ASCII  \NO\<0>
       FF  0013B      .BYTE   -1
 00000000' 0013C      .ADDRESS P.ABI
00 4F 4E   00140      .ASCII  \NO\<0>
       FE  00143      .BYTE   -2
```

```
                  00000000'  00144          .ADDRESS P.ABJ
                  00  4F  4E  00148          .ASCII  \NO\<0>
                      FE     0014B          .BYTE   -2
                  00000000'  0014C          .ADDRESS P.ABK
                  00  4F  4E  00150          .ASCII  \NO\<0>
                      FF     00153          .BYTE   -1
                  00000000'  00154          .ADDRESS P.ABL
                  00  4F  4E  00158          .ASCII  \NO\<0>
                      FF     0015B          .BYTE   -1
                  00000000'  0015C          .ADDRESS P.ABM
                  00  4F  4E  00160          .ASCII  \NO\<0>
                      FE     00163          .BYTE   -2
                  00000000'  00164          .ADDRESS P.ABN
                  00  4F  4E  00168          .ASCII  \NO\<0>
                      FE     0016B          .BYTE   -2
                  00000000'  0016C          .ADDRESS P.ABO
                  54  58  54  00170          .ASCII  \TXT\
                      FF     00173          .BYTE   -1
                  00000000'  00174          .ADDRESS P.ABP
                  00  4F  4E  00178          .ASCII  \NO\<0>
                      FE     0017B          .BYTE   -2
                      00#    0017C          .BYTE   0[56]
                  00000000'  001B4          .ADDRESS P.ABQ
                  00  4F  4E  001B8          .ASCII  \NO\<0>
                      00     001BB          .BYTE   0
                  00000000'  001BC          .ADDRESS P.ABR
                  00  4F  4E  001C0          .ASCII  \NO\<0>
                      FF     001C3          .BYTE   -1
                  00000000'  001C4          .ADDRESS P.ABS
                  00  4F  4E  001C8          .ASCII  \NO\<0>
                      FF     001CB          .BYTE   -1
                  00000000'  001CC          .ADDRESS P.ABT
                  00  4F  4E  001D0          .ASCII  \NO\<0>
                      FF     001D3          .BYTE   -1
                  00000000'  001D4          .ADDRESS P.ABU
                  00  4F  4E  001D8          .ASCII  \NO\<0>
                      FF     001DB          .BYTE   -1
                  00000000'  001DC          .ADDRESS P.ABV
                  00  4F  4E  001E0          .ASCII  \NO\<0>
                      FF     001E3          .BYTE   -1
                  00000000'  001E4          .ADDRESS P.ABW
                  00  4F  4E  001E8          .ASCII  \NO\<0>
                      FF     001EB          .BYTE   -1
                  00000000'  001EC          .ADDRESS P.ABX
                  00  4F  4E  001F0          .ASCII  \NO\<0>
                      FF     001F3          .BYTE   -1
                  00000000'  001F4          .ADDRESS P.ABY
                  00  4F  4E  001F8          .ASCII  \NO\<0>
                      00     001FB          .BYTE   0
                  00000000'  001FC          .ADDRESS P.ABZ
                  00  4F  4E  00200          .ASCII  \NO\<0>
                      00     00203          .BYTE   0
                  00000000'  00204          .ADDRESS P.ACA
                  44  4C  56  00208          .ASCII  \VLD\
                      FF     0020B          .BYTE   -1
                  00000000'  0020C          .ADDRESS P.ACB
                  00  4F  4E  00210          .ASCII  \NO\<0>
```

```
                 FF   00213             .BYTE    -1
          00000000'   00214             .ADDRESS P.ACC
         00  4F  4E   00218             .ASCII   \NO\<0>
                 FF   0021B             .BYTE    -1
          00000000'   0021C             .ADDRESS P.ACD
         00  4F  4E   00220             .ASCII   \NO\<0>
                 FF   00223             .BYTE    -1
          00000000'   00224             .ADDRESS P.ACE
         00  4F  4E   00228             .ASCII   \NO\<0>
                 FE   0022B             .BYTE    -2
          00000000'   0022C             .ADDRESS P.ACF
         4D  59  53   00230             .ASCII   \SYM\
                 00   00233             .BYTE    0
          00000000'   00234             .ADDRESS P.ACG
         58  54  4C   00238             .ASCII   \LTX\
                 FF   0023B             .BYTE    -1
                00#   0023C             .BYTE    0[104]
          00000000'   002A4             .ADDRESS P.ACH
         00  4F  4E   002A8             .ASCII   \NO\<0>
                 FF   002AB             .BYTE    -1
          00000000'   002AC             .ADDRESS P.ACI
         00  00  4C   002B0             .ASCII   \L\<0><0>
                 00   002B3             .BYTE    0
          00000000'   002B4             .ADDRESS P.ACJ
         00  4F  4E   002B8             .ASCII   \NO\<0>
                 FF   002BB             .BYTE    -1
          00000000'   002BC             .ADDRESS P.ACK
         00  4F  4E   002C0             .ASCII   \NO\<0>
                 FF   002C3             .BYTE    -1
          00000000'   002C4             .ADDRESS P.ACL
         00  4F  4E   002C8             .ASCII   \NO\<0>
                 00   002CB             .BYTE    0
                00#   002CC             .BYTE    0[344]

                                        .EXTRN   ANLOBJ$_OK, ANLOBJ$_ANYTHING
                                        .EXTRN   ANLOBJ$_DATATYPE
                                        .EXTRN   ANLOBJ$_ERRORCOUNT
                                        .EXTRN   ANLOBJ$_ERRORNONE
                                        .EXTRN   ANLOBJ$_ERRORS, ANLOBJ$_EXEFIXA
                                        .EXTRN   ANLOBJ$_EXEFIXAIMAGE
                                        .EXTRN   ANLOBJ$_EXEFIXALINE
                                        .EXTRN   ANLOBJ$_EXEFIXCOUNT
                                        .EXTRN   ANLOBJ$_EXEFIXEXTRA
                                        .EXTRN   ANLOBJ$_EXEFIXFIXED
                                        .EXTRN   ANLOBJ$_EXEFIXFLAGS
                                        .EXTRN   ANLOBJ$_EXEFIXG
                                        .EXTRN   ANLOBJ$_EXEFIXGIMAGE
                                        .EXTRN   ANLOBJ$_EXEFIXGLINE
                                        .EXTRN   ANLOBJ$_EXEFIXLIST
                                        .EXTRN   ANLOBJ$_EXEFIXNAME
                                        .EXTRN   ANLOBJ$_EXEFIXNAMEO
                                        .EXTRN   ANLOBJ$_EXEFIXP
                                        .EXTRN   ANLOBJ$_EXEFIXPSECT
                                        .EXTRN   ANLOBJ$_EXEFIXUP
                                        .EXTRN   ANLOBJ$_EXEFIXUPNONE
                                        .EXTRN   ANLOBJ$_EXEGST, ANLOBJ$_EXEHDR
                                        .EXTRN   ANLOBJ$_EXEHDRACTIVE
```

```
.EXTRN  ANLOBJ$_EXEHDRBLKCOUNT
.EXTRN  ANLOBJ$_EXEHDRCHANCOUNT
.EXTRN  ANLOBJ$_EXEHDRCHANDEF
.EXTRN  ANLOBJ$_EXEHDRDECECO
.EXTRN  ANLOBJ$_EXEHDRDMT
.EXTRN  ANLOBJ$_EXEHDRDST
.EXTRN  ANLOBJ$_EXEHDRFILEID
.EXTRN  ANLOBJ$_EXEHDRFIXED
.EXTRN  ANLOBJ$_EXEHDRFLAGS
.EXTRN  ANLOBJ$_EXEHDRGBLIDENT
.EXTRN  ANLOBJ$_EXEHDRGST
.EXTRN  ANLOBJ$_EXEHDRIDENT
.EXTRN  ANLOBJ$_EXEHDRIMAGEID
.EXTRN  ANLOBJ$_EXEHDRISD
.EXTRN  ANLOBJ$_EXEHDRISDBASE
.EXTRN  ANLOBJ$_EXEHDRISDCOUNT
.EXTRN  ANLOBJ$_EXEHDRISDFLAGS
.EXTRN  ANLOBJ$_EXEHDRISDGBLNAM
.EXTRN  ANLOBJ$_EXEHDRISDNUM
.EXTRN  ANLOBJ$_EXEHDRISDPFCDEF
.EXTRN  ANLOBJ$_EXEHDRISDPFCSIZ
.EXTRN  ANLOBJ$_EXEHDRISDTYPE
.EXTRN  ANLOBJ$_EXEHDRISDVBN
.EXTRN  ANLOBJ$_EXEHDRLINKID
.EXTRN  ANLOBJ$_EXEHDRMATCH
.EXTRN  ANLOBJ$_EXEHDRNAME
.EXTRN  ANLOBJ$_EXEHDRNOPATCH
.EXTRN  ANLOBJ$_EXEHDRPAGECOUNT
.EXTRN  ANLOBJ$_EXEHDRPAGEDEF
.EXTRN  ANLOBJ$_EXEHDRPATCH
.EXTRN  ANLOBJ$_EXEHDRPATCHDATE
.EXTRN  ANLOBJ$_EXEHDRPRIV
.EXTRN  ANLOBJ$_EXEHDRROPATCH
.EXTRN  ANLOBJ$_EXEHDRRWPATCH
.EXTRN  ANLOBJ$_EXEHDRSYMDBG
.EXTRN  ANLOBJ$_EXEHDRSYSVER
.EXTRN  ANLOBJ$_EXEHDRTEXTVBN
.EXTRN  ANLOBJ$_EXEHDRTIME
.EXTRN  ANLOBJ$_EXEHDRTYPEEXE
.EXTRN  ANLOBJ$_EXEHDRTYPELIM
.EXTRN  ANLOBJ$_EXEHDRUSERECO
.EXTRN  ANLOBJ$_EXEHDRXFER1
.EXTRN  ANLOBJ$_EXEHDRXFER2
.EXTRN  ANLOBJ$_EXEHDRXFER3
.EXTRN  ANLOBJ$_EXEHEADING
.EXTRN  ANLOBJ$_EXEPATCH
.EXTRN  ANLOBJ$_FLAG, ANLOBJ$_HEXDATA
.EXTRN  ANLOBJ$_HEXHEADING1
.EXTRN  ANLOBJ$_HEXHEADING2
.EXTRN  ANLOBJ$_INDMSGSEC
.EXTRN  ANLOBJ$_INTERACT
.EXTRN  ANLOBJ$_MASK, ANLOBJ$_OBJCPRREC
.EXTRN  ANLOBJ$_OBJDBGREC
.EXTRN  ANLOBJ$_OBJENV, ANLOBJ$_OBJEOMFLAGS
.EXTRN  ANLOBJ$_OBJEOMREC
.EXTRN  ANLOBJ$_OBJEOMSEVABT
.EXTRN  ANLOBJ$_OBJEOMSEVERR
```

```
.EXTRN  ANLOBJ$_OBJEOMSEVIGN
.EXTRN  ANLOBJ$_OBJEOMSEVRES
.EXTRN  ANLOBJ$_OBJEOMSEVSUC
.EXTRN  ANLOBJ$_OBJEOMSEVWRN
.EXTRN  ANLOBJ$_OBJEOMWREC
.EXTRN  ANLOBJ$_OBJFADPASSMECH
.EXTRN  ANLOBJ$_OBJGSDENV
.EXTRN  ANLOBJ$_OBJGSDENVFLAGS
.EXTRN  ANLOBJ$_OBJGSDENVPAR
.EXTRN  ANLOBJ$_OBJGSDEPM
.EXTRN  ANLOBJ$_OBJGSDEPMW
.EXTRN  ANLOBJ$_OBJGSDIDC
.EXTRN  ANLOBJ$_OBJGSDIDCENT
.EXTRN  ANLOBJ$_OBJGSDIDCFLAGS
.EXTRN  ANLOBJ$_OBJGSDIDCMATCH
.EXTRN  ANLOBJ$_OBJGSDIDCOBJ
.EXTRN  ANLOBJ$_OBJGSDIDCVALA
.EXTRN  ANLOBJ$_OBJGSDIDCVALB
.EXTRN  ANLOBJ$_OBJGSDLEPM
.EXTRN  ANLOBJ$_OBJGSDLPRO
.EXTRN  ANLOBJ$_OBJGSDLSY
.EXTRN  ANLOBJ$_OBJGSDPRO
.EXTRN  ANLOBJ$_OBJGSDPROW
.EXTRN  ANLOBJ$_OBJGSDPSC
.EXTRN  ANLOBJ$_OBJGSDPSCALIGN
.EXTRN  ANLOBJ$_OBJGSDPSCALLOC
.EXTRN  ANLOBJ$_OBJGSDPSCBASE
.EXTRN  ANLOBJ$_OBJGSDPSCFLAGS
.EXTRN  ANLOBJ$_OBJGSDREC
.EXTRN  ANLOBJ$_OBJGSDSPSC
.EXTRN  ANLOBJ$_OBJGSDSYM
.EXTRN  ANLOBJ$_OBJGSDSYMW
.EXTRN  ANLOBJ$_OBJGTXREC
.EXTRN  ANLOBJ$_OBJHDRIGNREC
.EXTRN  ANLOBJ$_OBJHEADING
.EXTRN  ANLOBJ$_OBJLITINDEX
.EXTRN  ANLOBJ$_OBJLNKREC
.EXTRN  ANLOBJ$_OBJLNMREC
.EXTRN  ANLOBJ$_OBJMHDCREATE
.EXTRN  ANLOBJ$_OBJMHDNAME
.EXTRN  ANLOBJ$_OBJMHDPATCH
.EXTRN  ANLOBJ$_OBJMHDREC
.EXTRN  ANLOBJ$_OBJMHDRECSIZ
.EXTRN  ANLOBJ$_OBJMHDSTRLVL
.EXTRN  ANLOBJ$_OBJMHDVERSION
.EXTRN  ANLOBJ$_OBJMTCCORRECT
.EXTRN  ANLOBJ$_OBJMTCINPUT
.EXTRN  ANLOBJ$_OBJMTCNAME
.EXTRN  ANLOBJ$_OBJMTCREC
.EXTRN  ANLOBJ$_OBJMTCSEQNUM
.EXTRN  ANLOBJ$_OBJMTCUIC
.EXTRN  ANLOBJ$_OBJMTCVERSION
.EXTRN  ANLOBJ$_OBJMTCWHEN
.EXTRN  ANLOBJ$_OBJPROARGCOUNT
.EXTRN  ANLOBJ$_OBJPROARGNUM
.EXTRN  ANLOBJ$_OBJPSECT
.EXTRN  ANLOBJ$_OBJSRCREC
```

```
                .EXTRN  ANLOBJ$_OBJSTATHEADING1
                .EXTRN  ANLOBJ$_OBJSTATHEADING2
                .EXTRN  ANLOBJ$_OBJSTATLINE
                .EXTRN  ANLOBJ$_OBJSTATTOTAL
                .EXTRN  ANLOBJ$_OBJSYMBOL
                .EXTRN  ANLOBJ$_OBJSYMFLAGS
                .EXTRN  ANLOBJ$_OBJTIRARGINDEX
                .EXTRN  ANLOBJ$_OBJTIRCMD
                .EXTRN  ANLOBJ$_OBJTIRCMDSTK
                .EXTRN  ANLOBJ$_OBJTBTREC
                .EXTRN  ANLOBJ$_OBJTIRREC
                .EXTRN  ANLOBJ$_OBJTIRSTOIM
                .EXTRN  ANLOBJ$_OBJTIRVIELD
                .EXTRN  ANLOBJ$_OBJTTLREC
                .EXTRN  ANLOBJ$_OBJVALUE
                .EXTRN  ANLOBJ$_OBJUVALUE
                .EXTRN  ANLOBJ$_PROTECTION
                .EXTRN  ANLOBJ$_SEVERITY
                .EXTRN  ANLOBJ$_TEXT, ANLOBJ$_TEXTHDR
                .EXTRN  ANLOBJ$_NOSUCHMOD
                .EXTRN  ANLOBJ$_BADDATE
                .EXTRN  ANLOBJ$_BADHDRBLKCOUNT
                .EXTRN  ANLOBJ$_BADSEVERITY
                .EXTRN  ANLOBJ$_BADSYM1ST
                .EXTRN  ANLOBJ$_BADSYMCHAR
                .EXTRN  ANLOBJ$_BADSYMLEN
                .EXTRN  ANLOBJ$_EXEBADFIXUPEND
                .EXTRN  ANLOBJ$_EXEBADFIXUPISD
                .EXTRN  ANLOBJ$_EXEBADFIXUPVBN
                .EXTRN  ANLOBJ$_EXEBADISDS1
                .EXTRN  ANLOBJ$_EXEBADISDTYPE
                .EXTRN  ANLOBJ$_EXEBADMATCH
                .EXTRN  ANLOBJ$_EXEBADPATCHLEN
                .EXTRN  ANLOBJ$_EXEBADOBJ
                .EXTRN  ANLOBJ$_EXEBADTYPE
                .EXTRN  ANLOBJ$_EXEBADXFERO
                .EXTRN  ANLOBJ$_EXEHDRISDLONG
                .EXTRN  ANLOBJ$_EXEHDRLONG
                .EXTRN  ANLOBJ$_EXEISDLENDZRO
                .EXTRN  ANLOBJ$_EXEISDLENGBL
                .EXTRN  ANLOBJ$_EXEISDLENPRIV
                .EXTRN  ANLOBJ$_EXENOTNATIVE
                .EXTRN  ANLOBJ$_EXTRABYTES
                .EXTRN  ANLOBJ$_FIELDFIT
                .EXTRN  ANLOBJ$_FLAGERROR
                .EXTRN  ANLOBJ$_NOTOK, ANLOBJ$_OBJBADIDCMATCH
                .EXTRN  ANLOBJ$_OBJBADNUM
                .EXTRN  ANLOBJ$_OBJBADPOP
                .EXTRN  ANLOBJ$_OBJBADPUSH
                .EXTRN  ANLOBJ$_OBJBADTYPE
                .EXTRN  ANLOBJ$_OBJBADVIELD
                .EXTRN  ANLOBJ$_OBJEOMBADSEV
                .EXTRN  ANLOBJ$_OBJEOMMISSING
                .EXTRN  ANLOBJ$_OBJFADBADAVC
                .EXTRN  ANLOBJ$_OBJFADBADRBC
                .EXTRN  ANLOBJ$_OBJGSDBADALIGN
                .EXTRN  ANLOBJ$_OBJGSDBADSUBTYP
```

```
                                                        .EXTRN  ANLOBJ$_OBJHDRRES
                                                        .EXTRN  ANLOBJ$_OBJMHDBADRECSIZ
                                                        .EXTRN  ANLOBJ$_OBJMHDBADSTRLVL
                                                        .EXTRN  ANLOBJ$_OBJMHDMISSING
                                                        .EXTRN  ANLOBJ$_OBJNONTIRCMD
                                                        .EXTRN  ANLOBJ$_OBJNOPSC
                                                        .EXTRN  ANLOBJ$_OBJNULLREC
                                                        .EXTRN  ANLOBJ$_OBJPOSPACE
                                                        .EXTRN  ANLOBJ$_OBJPROMINMAX
                                                        .EXTRN  ANLOBJ$_OBJPSCABSLEN
                                                        .EXTRN  ANLOBJ$_OBJRECTOOBIG
                                                        .EXTRN  ANLOBJ$_OBJTIRRES
                                                        .EXTRN  ANLOBJ$_OBJUNDEFENV
                                                        .EXTRN  ANLOBJ$_OBJUNDEFLIT
                                                        .EXTRN  ANLOBJ$_OBJUNDEFPSC
                                                        .EXTRN  ANALYZE$_FACILITY
                                                        .EXTRN  ANL$CHECK_SYMBOL
                                                        .EXTRN  ANL$FORMAT_ERROR
                                                        .EXTRN  ANL$FORMAT_HEX, ANL$FORMAT_LINE
                                                        .EXTRN  ANL$OBJECT_ARGUMENT_DSC
                                                        .EXTRN  ANL$OBJECT_ENV_REF
                                                        .EXTRN  ANL$OBJECT_PSECT_REF
                                                        .EXTRN  ANL$OBJECT_RECORD_LINE
                                                        .EXTRN  ANL$REPORT_LINE

                                                        .PSECT  $CODE$,NOWRT,2

                                    0FFC 00000          .ENTRY  ANL$OBJECT_TIR, Save R2,R3,R4,R5,R6,R7,R8,- ; 0573
                                                                R9,R10,R11                                  ;
                                5E          08  C2 00002        SUBL2   #8, SP                              ;
                                56      08  AC  D0 00005        MOVL    THE_RECORD, R6                      ; 0576
                                57      04  A6  9E 00009        MOVAB   4(R6), R7                           ; 0704
                                            67  DD 0000D        PUSHL   (R7)                                ;
                                58      00  BE  90 0000F        MOVB    @SCANP, RECORD_TYPE                 ; 0705
                                            56  DD 00013        PUSHL   R6                                  ; 0706
                                        04  AC  DD 00015        PUSHL   RECORD_NUMBER                       ; 0711
                                02          58  91 00018        CMPB    RECORD_TYPE, #2                     ; 0707
                                            08  12 0001B        BNEQ    1$                                  ;
                                00000000G   8F  DD 0001D        PUSHL   #ANLOBJ$_OBJTIRREC                  ;
                                            1D  11 00023        BRB     4$                                  ;
                                04          58  91 00025 1$:    CMPB    RECORD_TYPE, #4                     ; 0708
                                            08  12 00028        BNEQ    2$                                  ;
                                00000000G   8F  DD 0002A        PUSHL   #ANLOBJ$_OBJDBGREC                  ;
                                            10  11 00030        BRB     4$                                  ;
                                05          58  91 00032 2$:    CMPB    RECORD_TYPE, #5                     ; 0709
                                            05  13 00035        BEQL    3$                                  ;
                                7E          01  CE 00037        MNEGL   #1, -(SP)                           ;
                                            06  11 0003A        BRB     4$                                  ;
                                00000000G   8F  DD 0003C 3$:    PUSHL   #ANLOBJ$_OBJTBTREC                  ;
                        0000G   CF          03  FB 00042 4$:    CALLS   #3, ANL$OBJECT_RECORD_LINE          ; 0706
                                            6E  D6 00047        INCL    SCANP                               ; 0712
                                            59  D4 00049        CLRL    COMMAND_NUMBER                      ; 0719
                                54          01  90 0004B        MOVB    #1, FIT_OK                          ; 0720
                                55          66  3C 0004E 5$:    MOVZWL  (R6), R5                            ; 0721
                                55          67  C0 00051        ADDL2   (R7), R5                            ;
                                55          6E  D1 00054        CMPL    SCANP, R5                           ;
                                            01  1F 00057        BLSSU   6$                                  ;
```

```
                                 04 00059          RET
                01               54 E8 0005A 6$:   BLBS    FIT_OK, 7$
                                 04 0005D          RET
                                59 D6 0005E 7$:    INCL    COMMAND_NUMBER                          ; 0725
                                7E D4 00060        CLRL    -(SP)                                   ; 0726
        0000G   CF               01 FB 00062       CALLS   #1, ANL$REPORT_LINE
                5B         00    BE 90 00067       MOVB    @SCANP, COMMAND                         ; 0731
                               6E D6 0006B         INCL    SCANP                                   ; 0732
                53              5B 98 0006D        CVTBL   COMMAND, R3                             ; 0733
                               2D 18 00070         BGEQ    8$
                52              53 CE 00072        MNEGL   R3, R2                                  ; 0738
                               52 DD 00075         PUSHL   R2
                               59 DD 00077         PUSHL   COMMAND_NUMBER
                    00000000G  8F DD 00079         PUSHL   #ANLOBJ$_OBJTIRSTOIM
                               01 DD 0007F         PUSHL   #1
                               01 DD 00081         PUSHL   #1
        0000G   CF              05 FB 00083       CALLS   #5, ANL$FORMAT_LINE
          04    AE              52 D0 00088       MOVL    R2, WORK_DSC                            ; 0739
          08    AE              6E D0 0008C       MOVL    SCANP, WORK_DSC+4
                          04    AE 9F 00090       PUSHAB  WORK_DSC                                ; 0740
                               02 DD 00093         PUSHL   #2
        0000G   CF              02 FB 00095       CALLS   #2, ANL$FORMAT_HEX
                6E              52 C0 0009A       ADDL2   R2, SCANP                               ; 0741
                               AF 11 0009D         BRB     5$                                      ; 0733
                    0000'CF43  7F 0009F 8$:       PUSHAQ  CMD_TABLE[R3]                           ; 0749
                52              9E D0 000A4       MOVL    @(SP)+, R2
                               28 12 000A7         BNEQ    9$
                               53 DD 000A9         PUSHL   R3                                      ; 0750
                    00000000G  8F DD 000AB         PUSHL   #ANLOBJ$_OBJTIRRES
                        0000G   CF 02 FB 000B1     CALLS   #2, ANL$FORMAT_ERROR
              50                67 6E C3 000B6     SUBL3   SCANP, (R7), R0                         ; 0751
                                51 66 3C 000BA     MOVZWL  (R6), R1
        04    AE                50 51 C1 000BD     ADDL3   R1, R0, WORK_DSC
                          08    AE 67 D0 000C2     MOVL    (R7), WORK_DSC+4
                                04 AE 9F 000C6     PUSHAB  WORK_DSC                                ; 0752
                               02 DD 000C9         PUSHL   #2
                        0000G   CF 02 FB 000CB     CALLS   #2, ANL$FORMAT_HEX
                               04 000D0          RET                                             ; 0749
                    0000'CF43  7F 000D1 9$:       PUSHAQ  CMD_TABLE+7[R3]                         ; 0760
                50              9E 98 000D6       CVTBL   @(SP)+, R0
                        0000'   CF 50 C0 000D9     ADDL2   R0, STACK_DEPTH
                            0000'   CF DD 000DE    PUSHL   STACK_DEPTH                             ; 0765
                               0C BB 000E2         PUSHR   #^M<R2,R3>                              ; 0764
                               59 DD 000E4         PUSHL   COMMAND_NUMBER
                    0000'CF43  7F 000E6           PUSHAQ  CMD_TABLE+7[R3]                         ; 0762
                               9E 95 000EB         TSTB    @(SP)+
                               08 12 000ED         BNEQ    10$
                    00000000G  8F DD 000EF         PUSHL   #ANLOBJ$_OBJTIRCMD
                               06 11 000F5         BRB     11$
                    00000000G  8F DD 000F7 10$:    PUSHL   #ANLOBJ$_OBJTIRCMDSTK
                               01 DD 000FD 11$:    PUSHL   #1                                      ; 0761
                               02 DD 000FF         PUSHL   #2
                        0000G   CF 07 FB 00101     CALLS   #7, ANL$FORMAT_LINE
                    0000'CF43  7F 00106           PUSHAQ  CMD_TABLE+4[R3]                         ; 0773
52                9E              18 00 EF 0010B   EXTZV   #0, #24, @(SP)+, R2
                  00005042    8F    52 D1 00110   CMPL    R2, #20546                              ; 0776
                                  1B 13 00117     BEQL    12$
```

```
                00425042   8F              52  D1 00119          CMPL    R2, #4345922                                    ;
                                           12  13 00120          BEQL    12$                                             ;
                004C5042   8F              52  D1 00122          CMPL    R2, #5001282                                    ;
                                           09  13 00129          BEQL    12$                                             ;
                00575042   8F              52  D1 00012B         CMPL    R2, #5722178                                    ;
                                           3A  12 00132          BNEQ    14$                                             ;
                           37              54  E9 00134 12$:     BLBC    FIT_OK, 14$                                     ; 0783
          50               6E              01  C1 00137         ADDL3   #1, SCANP, R0                                   ;
                           55              50  D1 0013B         CMPL    R0, R5                                          ;
                                           0D  1B 0013E         BLEQU   13$                                             ;
                       00000000G   8F      DD 00140          PUSHL   #ANLOBJ$_FIELDFIT                               ;
                 0000G CF                  01  FB 00146         CALLS   #1, ANL$FORMAT_ERROR                            ;
                                           54  94 0014B         CLRB    FIT_OK                                          ;
                           1E              54  E9 0014D 13$:     BLBC    FIT_OK, 14$                                     ; 0784
                           7E        00    BE  9A 00150         MOVZBL  @SCANP, -(SP)                                   ; 0785
                       00000000G   8F      DD 00154          PUSHL   #ANLOBJ$_OBJPSECT                               ;
                                           02  DD 0015A         PUSHL   #2                                              ;
                                           7E  D4 0015C         CLRL    -(SP)                                           ;
                 0000G CF                  04  FB 0015E         CALLS   #4, ANL$FORMAT_LINE                             ;
                           7E        00    BE  9A 00163         MOVZBL  @SCANP, -(SP)                                   ; 0786
                 0000G CF                  01  FB 00167         CALLS   #1, ANL$OBJECT_PSECT_REF                        ;
                                           6E  D6 0016C         INCL    SCANP                                           ; 0787
                00005057   8F              52  D1 0016E 14$:     CMPL    R2, #20567                                      ; 0791
                                           1B  13 00175         BEQL    15$                                             ;
                00425057   8F              52  D1 00177         CMPL    R2, #4345943                                    ;
                                           12  13 0017E         BEQL    15$                                             ;
                004C5057   8F              52  D1 00180         CMPL    R2, #5001303                                    ;
                                           09  13 00187         BEQL    15$                                             ;
                00575057   8F              52  D1 00189         CMPL    R2, #5722199                                    ;
                                           3B  12 00190         BNEQ    17$                                             ;
                           38              54  E9 00192 15$:     BLBC    FIT_OK, 17$                                     ; 0797
          50               6E              02  C1 00195         ADDL3   #2, SCANP, R0                                   ;
                           55              50  D1 00199         CMPL    R0, R5                                          ;
                                           0D  1B 0019C         BLEQU   16$                                             ;
                       00000000G   8F      DD 0019E          PUSHL   #ANLOBJ$_FIELDFIT                               ;
                 0000G CF                  01  FB 001A4         CALLS   #1, ANL$FORMAT_ERROR                            ;
                                           54  94 001A9         CLRB    FIT_OK                                          ;
                           1F              54  E9 001AB 16$:     BLBC    FIT_OK, 17$                                     ; 0798
                           7E        00    BE  3C 001AE         MOVZWL  @SCANP, -(SP)                                   ; 0799
                       00000000G   8F      DD 001B2          PUSHL   #ANLOBJ$_OBJPSECT                               ;
                                           02  DD 001B8         PUSHL   #2                                              ;
                                           7E  D4 001BA         CLRL    -(SP)                                           ;
                 0000G CF                  04  FB 001BC         CALLS   #4, ANL$FORMAT_LINE                             ;
                           7E        00    BE  3C 001C1         MOVZWL  @SCANP, -(SP)                                   ; 0800
                 0000G CF                  01  FB 001C5         CALLS   #1, ANL$OBJECT_PSECT_REF                        ;
                           6E              02  C0 001CA         ADDL2   #2, SCANP                                       ; 0801
             00534E45   8F                 52  D1 001CD 17$:     CMPL    R2, #5459525                                    ; 0805
                                           3B  12 001D4         BNEQ    19$                                             ;
                           38              54  E9 001D6         BLBC    FIT_OK, 19$                                     ; 0808
          50               6E              02  C1 001D9         ADDL3   #2, SCANP, R0                                   ;
                           55              50  D1 001DD         CMPL    R0, R5                                          ;
                                           0D  1B 001E0         BLEQU   18$                                             ;
                       00000000G   8F      DD 001E2          PUSHL   #ANLOBJ$_FIELDFIT                               ;
                 0000G CF                  01  FB 001E8         CALLS   #1, ANL$FORMAT_ERROR                            ;
                                           54  94 001ED         CLRB    FIT_OK                                          ;
                           1F              54  E9 001EF 18$:     BLBC    FIT_OK, 19$                                     ; 0809
                           7E        00    BE  3C 001F2         MOVZWL  @SCANP, -(SP)                                   ; 0810
```

```
                        00000000G  8F    DD 001F6         PUSHL   #ANLOBJ$_OBJENV                         :
                                   02    DD 001FC         PUSHL   #2                                      :
                                   7E    D4 001FE         CLRL    -(SP)                                   :
           0000G  CF               04    FB 00200         CALLS   #4, ANL$FORMAT_LINE                     :
                  7E          00   BE    3C 00205         MOVZWL  @SCANP, -(SP)                           : 0811
           0000G  CF               01    FB 00209         CALLS   #1, ANL$OBJECT_ENV_REF                  :
                  6E               02    C0 0020E         ADDL2   #2, SCANP                               : 0812
        00000042  8F               52    D1 00211  19$:   CMPL    R2, #66                                 : 0815
                  12               13       00218         BEQL    20$                                     :
        00425042  8F               52    D1 0021A         CMPL    R2, #4345922                            :
                  09               13       00221         BEQL    20$                                     :
        00425057  8F               52    D1 00223         CMPL    R2, #4345943                            :
                  31               12       0022A         BNEQ    22$                                     :
                  2E               54    E9 0022C  20$:   BLBC    FIT_OK, 22$                             : 0820
   50             6E               01    C1 0022F         ADDL3   #1, SCANP, R0                           :
                  55               50    D1 00233         CMPL    R0, R5                                  :
                                   0D    1B 00236         BLEQU   21$                                     :
                        00000000G  8F    DD 00238         PUSHL   #ANLOBJ$_FIELDFIT                       :
           0000G  CF               01    FB 0023E         CALLS   #1, ANL$FORMAT_ERROR                    :
                                   54    94 00243         CLRB    FIT_OK                                  :
                  15               54    E9 00245  21$:   BLBC    FIT_OK, 22$                             : 0821
                  7E          00   BE    98 00248         CVTBL   @SCANP, -(SP)                           : 0822
                        00000000G  8F    DD 0024C         PUSHL   #ANLOBJ$_OBJVALUE                       :
                                   02    DD 00252         PUSHL   #2                                      :
                                   7E    D4 00254         CLRL    -(SP)                                   :
           0000G  CF               04    FB 00256         CALLS   #4, ANL$FORMAT_LINE                     :
                                   6E    D6 0025B         INCL    SCANP                                   : 0823
        00004255  8F               52    D1 0025D  22$:   CMPL    R2, #16981                              : 0827
                  31               12       00264         BNEQ    24$                                     :
                  2E               54    E9 00266         BLBC    FIT_OK, 24$                             : 0829
   50             6E               01    C1 00269         ADDL3   #1, SCANP, R0                           :
                  55               50    D1 0026D         CMPL    R0, R5                                  :
                                   0D    1B 00270         BLEQU   23$                                     :
                        00000000G  8F    DD 00272         PUSHL   #ANLOBJ$_FIELDFIT                       :
           0000G  CF               01    FB 00278         CALLS   #1, ANL$FORMAT_ERROR                    :
                                   54    94 0027D         CLRB    FIT_OK                                  :
                  15               54    E9 0027F  23$:   BLBC    FIT_OK, 24$                             : 0830
                  7E          00   BE    9A 00282         MOVZBL  @SCANP, -(SP)                           : 0831
                        00000000G  8F    DD 00286         PUSHL   #ANLOBJ$_OBJUVALUE                      :
                                   02    DD 0028C         PUSHL   #2                                      :
                                   7E    D4 0028E         CLRL    -(SP)                                   :
           0000G  CF               04    FB 00290         CALLS   #4, ANL$FORMAT_LINE                     :
                                   6E    D6 00295         INCL    SCANP                                   : 0832
        0058544C  8F               52    D1 00297  24$:   CMPL    R2, #5788748                            : 0836
                  35               12       0029E         BNEQ    26$                                     :
                  32               54    E9 002A0         BLBC    FIT_OK, 26$                             : 0839
   50             6E               01    C1 002A3         ADDL3   #1, SCANP, R0                           :
                  55               50    D1 002A7         CMPL    R0, R5                                  :
                                   0D    1B 002AA         BLEQU   25$                                     :
                        00000000G  8F    DD 002AC         PUSHL   #ANLOBJ$_FIELDFIT                       :
           0000G  CF               01    FB 002B2         CALLS   #1, ANL$FORMAT_ERROR                    :
                                   54    94 002B7         CLRB    FIT_OK                                  :
                  19               54    E9 002B9  25$:   BLBC    FIT_OK, 26$                             : 0840
                  7E          00   BE    9A 002BC         MOVZBL  @SCANP, -(SP)                           : 0841
                        00000000G  8F    DD 002C0         PUSHL   #ANLOBJ$_OBJLITINDEX                    :
                                   02    DD 002C6         PUSHL   #2                                      :
                                   7E    D4 002C8         CLRL    -(SP)                                   :
```

```
          0000G  CF          04  FB 002CA         CALLS   #4, ANL$FORMAT_LINE                    ;
                 5A      00  BE  90 002CF         MOVB    @SCANP, LITERAL_INDEX                  ; 0842
                             6E  D6 002D3         INCL    SCANP                                  ; 0843
       00000057  8F          52  D1 002D5  26$:   CMPL    R2, #87                                ; 0847
                             12  13 002DC         BEQL    27$                                    ;
       00575042  8F          52  D1 002DE         CMPL    R2, #5722178                           ;
                             09  13 002E5         BEQL    27$                                    ;
       00575057  8F          52  D1 002E7         CMPL    R2, #5722199                           ;
                             32  12 002EE         BNEQ    29$                                    ;
                 2F          54  E9 002F0  27$:   BLBC    FIT_OK, 29$                            ; 0852
    50           6E          02  C1 002F3         ADDL3   #2, SCANP, R0                          ;
                 55          50  D1 002F7         CMPL    R0, R5                                 ;
                             0D  1B 002FA         BLEQU   28$                                    ;
                 00000000G   8F  DD 002FC         PUSHL   #ANLOBJ$_FIELDFIT                      ;
          0000G  CF          01  FB 00302         CALLS   #1, ANL$FORMAT_ERROR                   ;
                             54  94 00307         CLRB    FIT_OK                                 ;
                 16          54  E9 00309  28$:   BLBC    FIT_OK, 29$                            ; 0853
                 7E      00  BE  32 0030C         CVTWL   @SCANP, -(SP)                          ; 0854
                 00000000G   8F  DD 00310         PUSHL   #ANLOBJ$_OBJVALUE                      ;
                             02  DD 00316         PUSHL   #2                                     ;
                             7E  D4 00318         CLRL    -(SP)                                  ;
          0000G  CF          04  FB 0031A         CALLS   #4, ANL$FORMAT_LINE                    ;
                 6E          02  C0 0031F         ADDL2   #2, SCANP                              ; 0855
       00005755  8F          52  D1 00322  29$:   CMPL    R2, #22357                             ; 0859
                             32  12 00329         BNEQ    31$                                    ;
                 2F          54  E9 0032B         BLBC    FIT_OK, 31$                            ; 0862
    50           6E          02  C1 0032E         ADDL3   #2, SCANP, R0                          ;
                 55          50  D1 00332         CMPL    R0, R5                                 ;
                             0D  1B 00335         BLEQU   30$                                    ;
                 00000000G   8F  DD 00337         PUSHL   #ANLOBJ$_FIELDFIT                      ;
          0000G  CF          01  FB 0033D         CALLS   #1, ANL$FORMAT_ERROR                   ;
                             54  94 00342         CLRB    FIT_OK                                 ;
                 16          54  E9 00344  30$:   BLBC    FIT_OK, 31$                            ; 0863
                 7E      00  BE  3C 00347         MOVZWL  @SCANP, -(SP)                          ; 0864
                 00000000G   8F  DD 0034B         PUSHL   #ANLOBJ$_OBJUVALUE                     ;
                             02  DD 00351         PUSHL   #2                                     ;
                             7E  D4 00353         CLRL    -(SP)                                  ;
          0000G  CF          04  FB 00355         CALLS   #4, ANL$FORMAT_LINE                    ;
                 6E          02  C0 0035A         ADDL2   #2, SCANP                              ; 0865
       0000004C  8F          52  D1 0035D  31$:   CMPL    R2, #76                                ; 0869
                             12  13 00364         BEQL    32$                                    ;
       004C5042  8F          52  D1 00366         CMPL    R2, #5001282                           ;
                             09  13 0036D         BEQL    32$                                    ;
       004C5057  8F          52  D1 0036F         CMPL    R2, #5001303                           ;
                             31  12 00376         BNEQ    34$                                    ;
                 2E          54  E9 00378  32$:   BLBC    FIT_OK, 34$                            ; 0875
    50           6E          04  C1 0037B         ADDL3   #4, SCANP, R0                          ;
                 55          50  D1 0037F         CMPL    R0, R5                                 ;
                             0D  1B 00382         BLEQU   33$                                    ;
                 00000000G   8F  DD 00384         PUSHL   #ANLOBJ$_FIELDFIT                      ;
          0000G  CF          01  FB 0038A         CALLS   #1, ANL$FORMAT_ERROR                   ;
                             54  94 0038F         CLRB    FIT_OK                                 ;
                 15          54  E9 00391  33$:   BLBC    FIT_OK, 34$                            ; 0876
                         00  BE  DD 00394         PUSHL   @SCANP                                 ; 0877
                 00000000G   8F  DD 00397         PUSHL   #ANLOBJ$_OBJVALUE                      ;
                             02  DD 0039D         PUSHL   #2                                     ;
                             7E  D4 0039F         CLRL    -(SP)                                  ;
```

```
                    0000G  CF              04 FB 003A1          CALLS   #4, ANL$FORMAT_LINE
                           6E              04 C0 003A6          ADDL2   #4, SCANP                          ;0878
                00475241   8F              52 D1 003A9 34$:     CMPL    R2, #4674113                     ;0882
                                           12 13 003B0          BEQL    35$
                004D5953   8F              52 D1 003B2          CMPL    R2, #5069139
                                           09 13 003B9          BEQL    35$
                00534E45   8F              52 D1 003BB          CMPL    R2, #5459525
                                           75 12 003C2          BNEQ    38$
                           72              54 E9 003C4 35$:     BLBC    FIT_OK, 38$                      ;0887
              50           6E              01 C1 003C7          ADDL3   #1, SCANP, R0
                           55              50 D1 003CB          CMPL    R0, R5
                                           0D 1B 003CE          BLEQU   36$
                                00000000G  8F DD 003D0          PUSHL   #ANLOBJ$_FIELDFIT
                    0000G  CF              01 FB 003D6          CALLS   #1, ANL$FORMAT_ERROR
                                           54 94 003DB          CLRB    FIT_OK
                           59              54 E9 003DD 36$:     BLBC    FIT_OK, 38$
                        04 AE          00  BE 9A 003E0          MOVZBL  @SCANP, WORK_DSC
           08   AE          6E              01 C1 003E5          ADDL3   #1, SCANP, WORK_DSC+4
                           4C              54 E9 003EA          BLBC    FIT_OK, 38$
                           50          04  AE 3C 003ED          MOVZWL  WORK_DSC, R0
                           50              08 C6 003F1          DIVL2   #8, R0
                           50              6E C0 003F4          ADDL2   SCANP, R0
                                           50 D6 003F7          INCL    R0
                           55              50 D1 003F9          CMPL    R0, R5
                                           0D 1B 003FC          BLEQU   37$
                                00000000G  8F DD 003FE          PUSHL   #ANLOBJ$_FIELDFIT
                    0000G  CF              01 FB 00404          CALLS   #1, ANL$FORMAT_ERROR
                                           54 94 00409          CLRB    FIT_OK
                           2B              54 E9 0040B 37$:     BLBC    FIT_OK, 38$                      ;0888
                           7E          08  AE DD 0040E          PUSHL   WORK_DSC+4                       ;0889
                           7E          08  AE 3C 00411          MOVZWL  WORK_DSC, -(SP)
                                00000000G  8F DD 00415          PUSHL   #ANLOBJ$_OBJSYMBOL
                                           02 DD 0041B          PUSHL   #2
                                           7E D4 0041D          CLRL    -(SP)
                    0000G  CF              05 FB 0041F          CALLS   #5, ANL$FORMAT_LINE
                                           1F DD 00424          PUSHL   #31                              ;0890
                                       08  AE 9F 00426          PUSHAB  WORK_DSC
                    0000G  CF              02 FB 00429          CALLS   #2, ANL$CHECK_SYMBOL
                           50          04  AE 3C 0042E          MOVZWL  WORK_DSC, R0                     ;0891
                           50              6E C0 00432          ADDL2   SCANP, R0
                           6E          01  A0 9E 00435          MOVAB   1(R0), SCANP
                00475241   8F              52 D1 00439 38$:     CMPL    R2, #4674113                     ;0895
                                           40 12 00440          BNEQ    40$
                           3D              54 E9 00442          BLBC    FIT_OK, 40$                      ;0900
              50           6E              01 C1 00445          ADDL3   #1, SCANP, R0
                           55              50 D1 00449          CMPL    R0, R5
                                           0D 1B 0044C          BLEQU   39$
                                00000000G  8F DD 0044E          PUSHL   #ANLOBJ$_FIELDFIT
                    0000G  CF              01 FB 00454          CALLS   #1, ANL$FORMAT_ERROR
                                           54 94 00459          CLRB    FIT_OK
                           24              54 E9 0045B 39$:     BLBC    FIT_OK, 40$                      ;0901
                           7E          00  BE 9A 0045E          MOVZBL  @SCANP, -(SP)                    ;0902
                                00000000G  8F DD 00462          PUSHL   #ANLOBJ$_OBJTIRARGINDEX
                                           02 DD 00468          PUSHL   #2
                                           7E D4 0046A          CLRL    -(SP)
                    0000G  CF              04 FB 0046C          CALLS   #4, ANL$FORMAT_LINE
                                           6E D6 00471          INCL    SCANP                            ;0903
```

```
                                56 DD 00473          PUSHL   R6                                  ; 0904
                        04  AE  9F 00475          PUSHAB  SCANP                               ;
                                03 DD 00478          PUSHL   #3                                  ;
               0000G  CF        03 FB 0047A          CALLS   #3, ANL$OBJECT_ARGUMENT_DSC         ;
                      54        50 90 0047F          MOVB    R0, FIT_OK                          ;
            00545854  8F        52 D1 00482 40$:     CMPL    R2, #5527636                        ; 0908
                                6B 12 00489          BNEQ    43$                                 ;
                      68        54 E9 0048B          BLBC    FIT_OK, 43$                         ; 0910
          50          6E        01 C1 0048E          ADDL3   #1, SCANP, R0                       ;
                      55        66 3C 00492          MOVZWL  (R6), R5                            ;
                      55        67 C0 00495          ADDL2   (R7), R5                            ;
                      55        50 D1 00498          CMPL    R0, R5                              ;
                                0D 1B 0049B          BLEQU   41$                                 ;
                 00000000G      8F DD 0049D          PUSHL   #ANLOBJ$_FIELDFIT                   ;
               0000G  CF        01 FB 004A3          CALLS   #1, ANL$FORMAT_ERROR                ;
                                54 94 004A8          CLRB    FIT_OK                              ;
                      49        54 E9 004AA 41$:     BLBC    FIT_OK, 43$                         ;
               04     AE   00   BE 9A 004AD          MOVZBL  @SCANP, WORK_DSC                    ;
     08   AE          6E        01 C1 004B2          ADDL3   #1, SCANP, WORK_DSC+4               ;
                      3C        54 E9 004B7          BLBC    FIT_OK, 43$                         ;
                      50   04   AE 3C 004BA          MOVZWL  WORK_DSC, R0                        ;
                      50        08 C6 004BE          DIVL2   #8, R0                              ;
                      50        6E C0 004C1          ADDL2   SCANP, R0                           ;
                                50 D6 004C4          INCL    R0                                  ;
                      55        66 3C 004C6          MOVZWL  (R6), R5                            ;
                      55        67 C0 004C9          ADDL2   (R7), R5                            ;
                      55        50 D1 004CC          CMPL    R0, R5                              ;
                                0D 1B 004CF          BLEQU   42$                                 ;
                 00000000G      8F DD 004D1          PUSHL   #ANLOBJ$_FIELDFIT                   ;
               0000G  CF        01 FB 004D7          CALLS   #1, ANL$FORMAT_ERROR                ;
                                54 94 004DC          CLRB    FIT_OK                              ;
                      15        54 E9 004DE 42$:     BLBC    FIT_OK, 43$                         ; 0911
                           04   AE 9F 004E1          PUSHAB  WORK_DSC                            ; 0912
                                02 DD 004E4          PUSHL   #2                                  ;
               0000G  CF        02 FB 004E6          CALLS   #2, ANL$FORMAT_HEX                  ;
                      50   04   AE 3C 004EB          MOVZWL  WORK_DSC, R0                        ; 0913
                      50        6E C0 004EF          ADDL2   SCANP, R0                           ;
                      6E   01   A0 9E 004F2          MOVAB   1(R0), SCANP                        ;
            00444C56  8F        52 D1 004F6 43$:     CMPL    R2, #4475990                        ; 0917
                                6C 12 004FD          BNEQ    47$                                 ;
                      69        54 E9 004FF          BLBC    FIT_OK, 47$                         ; 0920
          50          6E        02 C1 00502          ADDL3   #2, SCANP, R0                       ;
                      55        66 3C 00506          MOVZWL  (R6), R5                            ;
                      55        67 C0 00509          ADDL2   (R7), R5                            ;
                      55        50 D1 0050C          CMPL    R0, R5                              ;
                                0D 1B 0050F          BLEQU   44$                                 ;
                 00000000G      8F DD 00511          PUSHL   #ANLOBJ$_FIELDFIT                   ;
               0000G  CF        01 FB 00517          CALLS   #1, ANL$FORMAT_ERROR                ;
                                54 94 0051C          CLRB    FIT_OK                              ;
                      4A        54 E9 0051E 44$:     BLBC    FIT_OK, 47$                         ; 0921
                      52        6E D0 00521          MOVL    SCANP, R2                           ; 0922
                      7E   01   A2 9A 00524          MOVZBL  1(R2), -(SP)                        ;
                      7E        62 9A 00528          MOVZBL  (R2), -(SP)                         ;
                 00000000G      8F DD 0052B          PUSHL   #ANLOBJ$_OBJTIRVIELD                ;
                                02 DD 00531          PUSHL   #2                                  ;
                                7E D4 00533          CLRL    -(SP)                               ;
               0000G  CF        05 FB 00535          CALLS   #5, ANL$FORMAT_LINE                 ;
```

```
                                51          62  9A 0053A          MOVZBL  (R2), R1                                        ; 0923
                                50       01 A2  9A 0053D          MOVZBL  1(R2), R0
                                51          50  C0 00541          ADDL2   R0, R1
                                50          62  9A 00544          MOVZBL  (R2), R0                                        ; 0924
                                50          08  C0 00547          ADDL2   #8, R0
                                50          08  C6 0054A          DIVL2   #8, R0
                                50          08  C4 0054D          MULL2   #8, R0
                                1F          50  D1 00550          CMPL    R0, #31
                                            03  1B 00553          BLEQU   45$
                                50          1F  D0 00555          MOVL    #31, R0
                                50          51  D1 00558 45$:     CMPL    R1, R0
                                            0B  1B 0055B          BLEQU   46$
                                   00000000G 8F  DD 0055D          PUSHL   #ANLOBJ$_OBJBADVIELD                            ; 0925
                             0000G CF          01  FB 00563          CALLS   #1, ANL$FORMAT_ERROR
                                6E          02  C0 00568 46$:     ADDL2   #2, SCANP                                       ; 0926
                                      0000' CF  D5 0056B 47$:     TSTL    STACK_DEPTH                                     ; 0933
                                            0B  18 0056F          BGEQ    48$
                                   00000000G 8F  DD 00571          PUSHL   #ANLOBJ$_OBJBADPOP                              ; 0934
                             0000G CF          01  FB 00577          CALLS   #1, ANL$FORMAT_ERROR
                                12          53  91 0057C 48$:     CMPB    R3, #18                                         ; 0940
                                            11  12 0057F          BNEQ    49$
                                50          5A  9A 00581          MOVZBL  LITERAL_INDEX, R0                               ; 0945
                          35 0000' CF       50  E0 00584          BBS     R0, LITERAL_DEF_BITS, 52$
                                   00000000G 8F  DD 0058A          PUSHL   #ANLOBJ$_OBJUNDEFLIT                            ; 0946
                                            28  11 00590          BRB     51$
                             42 8F          53  91 00592 49$:     CMPB    R3, #66                                         ; 0948
                                            0B  12 00596          BNEQ    50$
                                50          5A  9A 00598          MOVZBL  LITERAL_INDEX, R0                               ; 0953
                          1E 0000' CF       50  E2 0059B          BBSS    R0, LITERAL_DEF_BITS, 52$
                                            1C  11 005A1          BRB     52$
                             52 8F          53  91 005A3 50$:     CMPB    R3, #82                                         ; 0955
                                            16  1F 005A7          BLSSU   52$
                             54 8F          53  91 005A9          CMPB    R3, #84
                                            10  1A 005AD          BGTRU   52$
                                02          58  91 005AF          CMPB    RECORD_TYPE, #2                                 ; 0962
                                            0B  12 005B2          BNEQ    52$
                                   00000000G 8F  DD 005B4          PUSHL   #ANLOBJ$_OBJNONTIRCMD                           ; 0963
                             0000G CF          01  FB 005BA 51$:     CALLS   #1, ANL$FORMAT_ERROR
                                         FA8C 31  005BF 52$:     BRW     5$                                              ; 0721
                                            04  005C2          RET                                                     ; 0976

; Routine Size:  1475 bytes,    Routine Base:  $CODE$ + 0000
```

```
;   549          0977  1 %sbttl 'ANL$OBJECT_TIR_CLEAN - Check TIR Errors and Clean Up'
;   550          0978  1 !++
;   551          0979  1 ! Functional Description:
;   552          0980  1 !       This routine is called at the end of each module to check for any
;   553          0981  1 !       global TIR errors (e.g., stack not empty).  It also cleans up for
;   554          0982  1 !       the next module.
;   555          0983  1 !
;   556          0984  1 ! Formal Parameters:
;   557          0985  1 !       none
;   558          0986  1 !
;   559          0987  1 ! Implicit Inputs:
;   560          0988  1 !       global data
;   561          0989  1 !
;   562          0990  1 ! Implicit Outputs:
;   563          0991  1 !       global data
;   564          0992  1 !
;   565          0993  1 ! Returned Value:
;   566          0994  1 !       none
;   567          0995  1 !
;   568          0996  1 ! Side Effects:
;   569          0997  1 !
;   570          0998  1 !--
;   571          0999  1
;   572          1000  1
;   573          1001  2 global routine anl$object_tir_clean: novalue = begin
;   574          1002  2
;   575          1003  2
;   576          1004  2 ! If the stack isn't clean, issue an error message.
;   577          1005  2
;   578          1006  2 if .stack_depth gtr 0 then
;   579          1007  2         anl$format_error(anlobj$_objbadpush,.stack_depth);
;   580          1008  2
;   581          1009  2 ! Now just clean it out anyway.
;   582          1010  2
;   583          1011  2 stack_depth = 0;
;   584          1012  2
;   585          1013  2 ! Clear all the literal definition bits.
;   586          1014  2
;   587          1015  2 ch$fill(%x'00', %allocation(literal_def_bits),literal_def_bits);
;   588          1016  2
;   589          1017  2 return;
;   590          1018  2
;   591          1019  1 end;


                                        003C 00000          .ENTRY  ANL$OBJECT_TIR_CLEAN, Save R2,R3,R4,R5      ; 1001
                                50  0000'  CF   D0 00002          MOVL    STACK_DEPTH, R0                     ;
                                          0D   15 00007          BLEQ    1$                                  ;
                                          50   DD 00009          PUSHL   R0                                  ; 1007
                                    00000000G  8F   DD 0000B          PUSHL   #ANLOBJ$_OBJBADPUSH                 ;
                                        0000G  CF       02   FB 00011          CALLS   #2, ANL$FORMAT_ERROR        ;
                                               0000'    CF   D4 00016 1$:      CLRL    STACK_DEPTH                 ; 1011
                    20          00             6E       00   2C 0001A          MOVC5   #0, (SP), #0, #32, LITERAL_DEF_BITS ; 1015
                                               0000'    CF       0001F                                          ;
```

```
                                                04 00022        RET                                     ; 1019

; Routine Size:  35 bytes,    Routine Base:  $CODE$ + 05C3

;  592          1020  1
;  593          1021  0 end eludom



;                                   PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  $OWN$                          1060  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $PLIT$                          511  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                         1510  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)



;                               Library Statistics
;                                        -------- Symbols --------      Pages       Processing
;       File                             Total    Loaded   Percent      Mapped      Time
;
;  _$255$DUA28:[SYSLIB]STARLET.L32;1      9776        20         0         581        00:01.0



;                                       COMMAND QUALIFIERS

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

; Size:          1510 code + 1571 data bytes
; Run Time:        00:30.6
; Elapsed Time:    01:25.6
; Lines/CPU Min:    2004
; Lexemes/CPU-Min: 24691
; Memory Used:  442 pages
; Compilation Complete
```

OB MISC
LIS

RMS21DX
LIS

RMS31DX
LIS

RMS
LIS

OBJUTIR
LIS