



# 177

# MVS

*June 2001*

---

## **In this issue**

- 3 A DASD space monitor
  - 12 Automating SMP/E HOLDDATA analysis
  - 28 Speedy panel access
  - 29 Formatting internal system trace table entries – part 2
  - 71 A user problem tip
  - 72 MVS news
- 

© Xephon plc 2001

# update

# ***MVS Update***

---

## **Published by**

Xephon  
27-35 London Road  
Newbury  
Berkshire RG14 1JL  
England  
Telephone: 01635 33598  
From USA: 01144 1635 33598  
E-mail: Jaimek@xephon.com

## **North American office**

Xephon/QNA  
PO Box 350100,  
Westminster, CO 80035-0100  
USA  
Telephone: (303) 410 9344  
Fax: (303) 438 0290

## **Contributions**

Articles published in *MVS Update* are paid for at the rate of £170 (\$260) per 1000 words and £100 (\$160) per 100 lines of code for the first 200 lines of original material. The remaining code is paid for at the rate of £50 (\$80) per 100 lines. In addition, there is a flat fee of £30 (\$50) per article. To find out more about contributing an article, you can download a copy of our *Notes for Contributors* from [www.xephon.com/contnote.html](http://www.xephon.com/contnote.html).

## ***MVS Update* on-line**

Code from *MVS Update*, and complete issues in Acrobat PDF format, can be downloaded from our Web site at <http://www.xephon.com/mvsupdate.html>; you will need to supply a word from the printed issue.

## **Editor**

Jaime Kaminski

## **Disclaimer**

Readers are cautioned that, although the information in this journal is presented in good faith, neither Xephon nor the organizations or individuals that supplied information in this journal give any warranty or make any representations as to the accuracy of the material it contains. Neither Xephon nor the contributing organizations or individuals accept any liability of any kind howsoever arising out of the use of such material. Readers should satisfy themselves as to the correctness and relevance to their circumstances of all advice, information, code, JCL, EXECs, and other contents of this journal before making any use of it.

## **Subscriptions and back-issues**

A year's subscription to *MVS Update*, comprising twelve monthly issues, costs £340.00 in the UK; \$505.00 in the USA and Canada; £346.00 in Europe; £352.00 in Australasia and Japan; and £350.00 elsewhere. In all cases the price includes postage. Individual issues, starting with the January 1992 issue, are available separately to subscribers for £29.00 (\$43.00) each including postage.

---

© Xephon plc 2001. All rights reserved. None of the text in this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior permission of the copyright owner. Subscribers are free to copy any code reproduced in this publication for use in their own installations, but may not sell such code or incorporate it in any commercial product. No part of this publication may be used for any form of advertising, sales promotion, or publicity without the written permission of the publisher. Copying permits are available from Xephon in the form of pressure-sensitive labels, for application to individual copies. A pack of 240 labels costs \$36 (£24), giving a cost per copy of 15 cents (10 pence). To order, contact Xephon at any of the addresses above.

*Printed in England.*

# A DASD space monitor

## INTRODUCTION

The following ISPF application provides a comprehensive graphical view of DASD pool usage. The DASDMON REXX routine uses IDCAMS DCOLLECT to get information about DASD space utilization from all online disks. It then displays numeric information and a bar chart representing space usage utilization for each online disk. Individual disk information can be aggregated using a VOLSER mask to give a comprehensive view of your different DASD pools.

## DASDMON REXX

```
/* REXX                                                                    */
/* _____                                                                */
/* DASD SPACE MONITOR                                                    */
/* _____                                                                */
/* IN ORDER TO RUN THIS REXX UNDER TSO, YOU SHOULD MODIFY              */
/* IKJTS000 TO ADD IDCAMS IN AUTHPGM                                     */
/* _____                                                                */
                                                                    /*
                                                                    /* GET SYSTEM INFORMATION */
                                                                    /*
                                                                    /* POINTS TO PCCA */

CPU = STORAGE(208,4)
CPU = STORAGE(D2X(C2D(CPU) + 12),4) || "-" || ,
      STORAGE(D2X(C2D(CPU) + 08),4)
CVT = STORAGE(10,4)
CVTPROD = D2X(C2D(CVT) - C2D("28"X))
SPLVL = STORAGE(CVTPROD,07)
ADDRESS ISPEXEC «VGET (ZSYSID)»
SID = ZSYSID
                                                                    /*
                                                                    /* INITIALIZE VARIABLES */
                                                                    /*
                                                                    /* DEFAULT THRESHOLDS */

LO = 70
HI = 85
                                                                    /* DEFAULT MASK */

MASK = «'*****»
                                                                    /* DEFAULT SORT KEY */

SK   = «VOLUME»
SD   = «A»
                                                                    /* DEFAULT CALCULATION UNIT */
```

```

SU   = <<C>>
SU1  = <<CYL>>
SU2  = <<CYL>>
SU3  = <<CYL>>

/* COLOUR 3270 FIELDS */

BL = "21"X
GR = "22"X
YE = "23"X
RE = "24"X

/******
/* PREPARE IDCAMS SYSIN FOR DCOLLECT */
/******

<<ALLOC DD(SYSPRINT) DSN("NULLFILE") SHR REUSE>>
<<ALLOC DD(SYSIN) NEW LRECL(80) REUSE>>
QUEUE " DCOLLECT VOLUMES(*) OUTFILE(OUTDS) NODATAINFO"
QUEUE ""
<<EXECIO * DISKW SYSIN (FINIS)>>
/******
/* LOOP UNTIL PF3 */
/******

END = 0
DO WHILE (¬END)

/*=====*/
/* START DCOLLECT */
/*=====*/

<<ALLOC DD(OUTDS) NEW LRECL(264) RECFM(V B) REUSE>>
<<CALL "SYS1.LINKLIB(IDCAMS)">>
<<EXECIO * DISKR OUTDS (STEM X. FINIS)>>
<<FREE F(OUTDS)>>

/*=====*/
/* SELECT */
/* SPACE CALCULATION UNIT */
/*=====*/

SELECT
  WHEN SU = <<M>> THEN
    DO
      SUF = <<MB>>
      SUK = 1
    END
  WHEN SU = <<C>> THEN
    DO
      SUF = <<CYL>>
      SUK = 15 * 56664 / 1024 / 1024 /* 3390 DISK MODEL */
    END
  OTHERWISE
    DO
      SUF = <<MB>>
      SUK = 1
    END
END
END
SU1 = SUF

```

```

SU2 = SUF
SU3 = SUF

/*=====*/
/* ANALYSE DCOLLECT */
/* OUTPUT */
/*=====*/

POOL_NB = 0
DO I = 1 TO X.0
  VOLUME.I = SUBSTR(X.I,25,6)
  PCT_FREE.I = C2D(SUBSTR(X.I,36,1))
  KB_FREE.I = C2D(SUBSTR(X.I,37,4))
  KB_ALLOC.I = C2D(SUBSTR(X.I,41,4))
  TOT_CAPA.I = C2D(SUBSTR(X.I,45,4))
  TYPE.I = SUBSTR(X.I,69,8)
  STOGROUP_NM.I = RETURN_PATTERN(VOLUME.I)
  IF (RETURN_MATCH(VOLUME.I) = Y) THEN
    DO
      DO J = 0 TO POOL_NB
        IF STOGROUP_NM.I = GROUP.J
          THEN
            DO
              FREE.J = FREE.J + KB_FREE.I
              ALLOC.J = ALLOC.J + KB_ALLOC.I
              CAPA.J = CAPA.J + TOT_CAPA.I
              VOL_NB.J = VOL_NB.J + 1
            LEAVE
          END
        END
      END
      /*=====*/
      /* IF PATTERN IS NEW ADD */
      /* A NEW ENTRY */
      /*=====*/
    IF J = POOL_NB + 1
      THEN
        DO
          FREE.J = KB_FREE.I
          ALLOC.J = KB_ALLOC.I
          CAPA.J = TOT_CAPA.I
          GROUP.J = STOGROUP_NM.I
          VOL_NB.J = 1
          POOL_NB = POOL_NB + 1
        END
      END
    END
  END
  SGDATA = ""
  /*=====*/
  /* CREATE ISPF TABLE */
  /*=====*/
  «ISPEXEC TBERASE DASDMON»
  «ISPEXEC TBCREATE DASDMON NOWRITE
    NAMES(TVOL TUSED TFREE TTOTAL TPCT TLINE TLINEC)»

```

```

DO I = 1 TO POOL_NB
GROUP.I = LEFT(GROUP.I,06,' ')
PCT.I = 100 - 100*(FREE.I / CAPA.I)
PCT.I = FORMAT(PCT.I,3,0)
/*=====*/
/* MB CAPACITY UNIT */
/*=====*/

FREE.I = FORMAT(FREE.I/1024,5,0)
ALLOC.I = FORMAT(ALLOC.I/1024,5,0)
CAPA.I = FORMAT(CAPA.I/1024,06,0)
/*=====*/
/* CYL CAPACITY UNIT */
/*=====*/

FREEC.I = FORMAT(FREE.I / SUK ,5,0)
ALLOCC.I = FORMAT(ALLOC.I / SUK ,5,0)
CAPAC.I = FORMAT(CAPA.I / SUK ,6,0)
VOL_NB.I = FORMAT(VOL_NB.I,3)
/*=====*/
/* BAR COLOUR */
/*=====*/

IF PCT.I >= LO THEN CO = YE
ELSE CO = GR
IF PCT.I >= HI THEN CO = RE
STAR = COPIES("*",41)
BLNK = COPIES(" ",41)
BAR = (PCT.I)/2.5
BAR = FORMAT(BAR,2,0)
BAR = SUBSTR(STAR,1,BAR)
BAR = BAR || BL || BLNK
BAR = SUBSTR(BAR,1,40)
/*=====*/
/* MB CAPACITY UNIT */
/*=====*/

LNE.I = BL || GROUP.I || " " || VOL_NB.I || ,
" " || ALLOC.I || ,
" " || FREE.I || " " || CAPA.I || " " || ,
" " CO || PCT.I || BL || ,
" " || CO || BAR || " "
LNE.I = SUBSTR(LNE.I,1,80)
/*=====*/
/* CYL CAPACITY UNIT */
/*=====*/

LNEC.I = BL || GROUP.I || " " || VOL_NB.I || ,
" " || ALLOCC.I || ,
" " || FREEC.I || " " || CAPAC.I || " " || ,
" " CO || PCT.I || BL || ,
" " || CO || BAR || " "
LNEC.I = SUBSTR(LNEC.I,1,80)
TVOL = GROUP.I
TUSED = ALLOC.I
TFREE = FREE.I

```

```

TTOTAL = CAPA.I
TPCT   = PCT.I
TLINE  = LNE.I
TLINEC = LNEC.I
«ISPEXEC TBADD DASDMON»
END

```

```

/*=====*/
/* SORT FIELD      */
/*=====*/

```

```

SELECT
  WHEN SK = «VOLUME» THEN
    DO
      SF = «TVOL»
      ST = «C»
    END
  WHEN SK = «FREE» THEN
    DO
      SF = «TFREE»
      ST = «N»
    END
  WHEN SK = «USED» THEN
    DO
      SF = «TUSED»
      ST = «N»
    END
  WHEN SK = «TOTAL» THEN
    DO
      SF = «TTOTAL»
      ST = «N»
    END
  WHEN SK = «%» THEN
    DO
      SF = «TPCT»
      ST = «N»
    END
  OTHERWISE
    SF = «TVOL»
    ST = «N»
  DO
  END
END

```

```

/*=====*/
/* SORT DIRECTION  */
/*=====*/

```

```

SELECT
  WHEN SD = «A» THEN
    DO
      SO = «A»
    END
  WHEN SD = «D» THEN
    DO

```

```

        SO = <<D>>
    END
    OTHERWISE
    DO
        SO = <<A>>
    END
END
<<ISPEXEC TBTOP DASDMON>>
<<ISPEXEC TBSORT DASDMON FIELDS(<<SF>>,>>ST>>,>>SO>>)>>
<<ISPEXEC TBSKIP DASDMON>>
ALLDATA = ""
DO WHILE RC = 0
    <<ISPEXEC TBGET DASDMON>>
    SELECT
        /* CAPACITY UNIT = MB */
        WHEN SU = <<M>> THEN ALLDATA = ALLDATA || TLINE
        /* CAPACITY UNIT = CYL */
        WHEN SU = <<C>> THEN ALLDATA = ALLDATA || TLINEC
    END
    <<ISPEXEC TBSKIP DASDMON>>
END

PDATE = DATE("E") /* EUROPEAN FORMAT: DD/MM/YY */
PTIME = TIME()
ADDRESS ISPEXEC <<PQUERY PANEL(DASDMON) AREANAME(SGDATA) DEPTH(DEPTH)>>
AREALEN = DEPTH * 80
/*=====*/
/* SCROLLING MANAGEMENT */
/*=====*/

TOP = 0
RETCODE = 0
DO UNTIL (RETCODE = 0)
    SGDATA = SUBSTR(ALLDATA,(TOP*80)+1,AREALEN)
    ADDRESS ISPEXEC <<DISPLAY PANEL(DASDMON)>>
    RETCODE = RC
    ADDRESS ISPEXEC <<VGET (ZVERB ZSCROLLA ZSCROLLN)>>
    IF ZVERB = "" THEN LEAVE
    IF (POOL_NB > VDEPTH)
        THEN
            DO
                IF (ZVERB = "DOWN")
                    THEN
                        IF ZSCROLLA = "MAX"
                            THEN TOP = POOL_NB-VDEPTH+1
                            ELSE TOP = MIN(TOP+ZSCROLLN, POOL_NB-VDEPTH+1)
                        ELSE
                            IF (ZVERB = "UP")
                                THEN
                                    IF ZSCROLLA = "MAX"
                                        THEN TOP = 1
                                        ELSE TOP = MAX(TOP-ZSCROLLN,0)
                                END
                            END
                        END
                    END
                END
            END
        END
    END
END

```



```

                END
            END
            IF RETCODE >= 8 THEN END = 1
            <<ISPEXEC TBCLOSE  DASDMON>>
        END
        <<FREE F(SYSIN)>>
        EXIT

                                                    /*****/
                                                    /* VOLSER PATTERN          */
                                                    /*****/

RETURN_PATTERN:
    ARG VOLSER
    PATTERN = ""
    DO K = 1 TO LENGTH(MASK)
        IF SUBSTR(MASK,K,1) = <<%>> THEN
            DO
                PATTERN = PATTERN || "%"
            END
        ELSE
            DO
                PATTERN = PATTERN || SUBSTR(VOLSER,K,1)
            END
        END
    END
    RETURN PATTERN

                                                    /*****/
                                                    /* VOLSER FILTER          */
                                                    /*****/

RETURN_MATCH:
    ARG VOLSER
    MATCH = Y
    DO K = 1 TO LENGTH(MASK)
        SELECT
            WHEN SUBSTR(MASK,K,1) = <<%>> THEN
                DO
                    END
            WHEN SUBSTR(MASK,K,1) = <<+>> THEN
                DO
                    END
            OTHERWISE
                DO
                    IF SUBSTR(MASK,K,1) ≠ SUBSTR(VOLSER,K,1) THEN
                        DO
                            MATCH = N
                        END
                    END
                END
            END
        END
    END
    RETURN MATCH

```

## DASDMON ISPF PANEL

```
)Attr default(?+_ )
£ Type(output) Intens(high) Color(yellow) Just(left) Caps(off)
+ Type(text) Intens(high) Color(green) Caps(off)
# Type(input) Intens(high) Color(red) Caps(on)
Ø Area(dynamic) Scroll(on) Extend(on)
% Type(dataout) Intens(low) Color(blue)
* Type(dataout) Color(green) Hilite(reverse)
@ Type(dataout) Color(yellow) Hilite(reverse)
- Type(dataout) Color(red) Hilite(reverse)
/* _____ */
/* DASD SPACE MONITOR */
/* _____ */
)Body expand(@@)
+@-@ Dasd Space Monitor @-@+
?Command ==>_zcmd ?Scroll ==>_amt +
+CPU:£cpu + SYSID:£sid + MVS:£splvl + DATE:£pdate + TIME:£ptime +
Mask:#mask +Sort Key:#sk +A/D:#z+ Mb/Cyl:#z+ Hi-Threshold:#hi+%
?Volume Vol Used Free Total % + Low-Threshold:#lo+%
?Mask Cnt £su1+ £su2+ £su3+?Used Pool Pct used
£rline +
Øsgdata _____Ø
+@-@ PF1=HELP PF3=END PF7=UP PF8=DOWN ENTER=REFRESH @-@+
+
)Init
.cursor = zcmd
.zvars = '(sd,su)'
.help = dasdmonh
)Proc
&vdepth = 1vline(sgdata)
ver(&lo,range,Ø,1ØØ)
ver(&ht,range,Ø,1ØØ)
VER (&SK,LIST,VOLUME,USED,FREE,TOTAL,%)
VER (&SD,LIST,A,D)
VER (&SU,LIST,M,C)
)End
```

## DASDMONH ISPF HELP PANEL

```
)attr default(#_)
# type(text) intens(high) color(yellow)
$ type(text) intens(low) color(green)
¬ type(text) intens(low) color(green) hilite(reverse)
[ type(text) intens(high) color(yellow)
] type(text) intens(low) color(turq)
{ type(output) intens(high) color(yellow) caps(off)
} type(output) intens(high) color(turq) caps(off)
)body expand(||)
|-| DASD Space Monitor |-|
#Command ==>_ZCMD
```

This ISPF dialog gives a graphical view of your DASD Space utilization.  
The panel allows you to specify the:

```
#Mask:          [This mask is used to filter and aggregate DASD volser.

                + = Discrimination character
                % = Aggregation character

                Valid masks: IPL%++  RES+++  STR+%

#Sort Key:      [This field represents the sort key used in the display panel.
                Valid values are: Volume, Used, Free, Total, and Pct

#A/D:          [This field represents the sorting direction.
                Valid values are: (A)scending - (D)escending.

#Mb/Cyl:        [This field represents the capacity unit.
                Valid values are: (M)egabytes - (C)ylinders.

)init
)proc
)end
```

## IKJTSO00 UPDATE

In order to be able to use IDCAMS DCOLLECT under TSO, you should add IDCAMS to the authorized program list (AUTHPGM) in IKJTSO00 in SYS1.PARMLIB. This change can be activated dynamically using the following command:

```
TSO PARMLIB UPDATE(00)
```

The result of the DASDMON execution is as follows:

```
_____ DASD Space Monitor _____
Command ==>                               Scroll ==> PAGE
CPU: 9672-9910  SYSID: SMVS  MVS: SP6.0.9    DATE: 06/04/01  TIME: 12:08:50
Mask: W+++%%  Sort Key: VOLUME  A/D: A  Mb/Cyl: C          Hi-Threshold: 85 %
Volume Vol  Used  Free  Total    %          Low-Threshold: 70 %
Mask  Cnt  CYL  CYL  CYL  Used          Pool Pct used
_____ 0 _____ 50 _____ 100
WHR$%%  1  502  2836  3340  15  *****
WTRS%%  1  1669  1670  3340  50  *****
WYS$%%  2  223  6455  6678  3  *
WYSS%%  1  1778  449  2226  80  *****
_____ PF1=HELP  PF3=END  PF7=UP  PF8=DOWN  ENTER=REFRESH _____
The MASK field can be used to modify VOLSER filtering and aggregation. Special characters
are used to specify that mask:
+ = Discrimination character
% = Aggregation character
```

---

*Systems Programmer (UK)*

© Xephon 2001

---

# Automating SMP/E HOLDDATA analysis

## THE PROBLEM

A common problem when applying product maintenance using SMP/E is that PTFs have been held for any number of HOLDDATA reasons. When APPLYing this kind of PTF, any ++HOLD statements will typically be BYPASSED until the complete PREREQ chain has been established and an APPLY CHECK has run successfully. Once this has been done, the MVS systems programmer is left with the unenviable task of manually investigating all the HOLDDATA that has been BYPASSED. He has to browse, from the SMPPTS dataset, the text of each held PTF to analyse the corresponding hold reason.

## A SOLUTION

The following REXX/ISPF application automates this process. It analyses the result of the SMPOUT SYSOUT of the APPLY CHECK to show a report of all held reasons bypassed during the SMP/E process.

## SMPEHD REXX EXEC

```
/* REXX */
x = msg(off)
/*****/
/* How are we invoked ? */
/*****/
se_edit = 0
ADDRESS ISREDIT "MACRO NOPROCESS"
if rc <> 0 then do
    idsn = ""
end
else
do
    address isredit "(idsn) = dataset"
    address isredit "(imember) = member"
    if imember <> "" then idsn = idsn("imember")
    if idsn = "" then
        do
            idsn = "<= Current edit session =>"
            se_edit = 1
        end
    end
end
```

```

end
end
/*****/
/* Display the options panel */
/*****/
address ispexec
"vget (smpout smppts sysoutc) profile"
if smpout = "" then smpout = "your.smpout.listing"
if smppts = "" then smppts = "your.smppts"
if sysoutc = "" then sysoutc = "A"
if idsn <> "" then smpout = idsn
smpcurs = "smpout"
lrc = 0
do while lrc = 0
  drc = 1
  do while drc <> 0
    "display panel(smpehd1) cursor("smpcurs")"
    drc = rc
    pfkey_01 = pfkey1
    field_01 = zcurfld
    if pfkey_01 = "PF03" then
      do
        zedsmg = "Session cancelled"
        zedlmsg = "SMP/E Hold Data processing cancelled"
        "SETMSG MSG(ISRZ001)"
        "vput (smpout smppts sysoutc) profile"
        exit
      end
    if (field_01 = "SMPPTS" & smppts = "") then
      do
        call manage_smppts
        smpcurs = "smppts"
        drc = 1
      end
    if pfkey_01 = " " then /* enter */
      do
        z = listdsi(smppts)
        if z <> 0 then
          do
            zedsmg = "SMPPTS Error"
            zedlmsg = "Error with SMPPTS dataset "smppts"
            "SETMSG MSG(ISRZ001)"
            smpcurs = "smppts"
            drc = 1
          end
        end
      end
    pfkey_01 = " "
  end
  function = "SMP/E sysout analysis..."
  "control display lock"
  "display panel(smpehd3)"
/*****/

```

```

/* Read the SMPDOUT dataset */
/*****
if se_edit = 0 then
  do
    address tso
    "allocate file(smpdd) da("smpout") shr reuse"
    "execio * diskr smpdd (finis stem smpoutf."
  end
else
  do
    address isredit
    "(lastl) = linenum .zlast"
    do i = 1 to lastl
      address isredit "(linev) = line " i
      smpoutf.i = linev
    end
    smpoutf.0 = lastl
  end
  address ispexec "tbcreate smptable" ,
                  "nowrite names(fmid action) ,
                  keys(ptfnum hreason) replace"
  ptfnum = ""
  do i = 1 to smpoutf.0
    select
      when word(smpoutf.i,1) = "GIM42001W" then
        do
          ptfnum = word(smpoutf.i,7)
        end
      when word(smpoutf.i,1) = "GIM35966I" then
        do
          hreason = word(smpoutf.i,4)

          x = outtrap('t')
          dsn = ""||strip(smpts,,"")||"("ptfnum)""
          address tso "listds "dsn
          zrc = rc
          x = outtrap('OFF')
          if zrc <> 0 then
            do
              address ispexec
              zedsmg = "SMPPTS Error"
              zedlmsg = "Error with SMPPTS dataset "smpts
              "SETMSG MSG(ISRZ001)"
            end
          else
            do
              call find_fmid
              address ispexec "tbadd smptable "
            end
          end
        end
      otherwise nop
    end
  end
end

```

```

end
address tso "free file(smpdd)"
address ispexec
"tbsort smptable fields(hreason,c,a,ptfnum,c,a)"
tbrc = 0
do while tbrc = 0
  "tbdispl smptable panel(smpehd2)"
  tbrc = rc
  tsel = translate(tsel)
  select
    when tsel = "B" then
      do
        call browse_ptf
        action = "PTF text browsed"
        "tbmod smptable"
      end
    when tsel = "S" then
      do
        action = "PTF ""++ HOLD"" element browsed"
        "tbmod smptable"
        outff.0 = 0
        call process_ptf
        call display_result
      end
    when tsel = "P" then
      do
        action = "PTF / HOLD element printed / Sysout class" sysoutc
        "tbmod smptable"
        outff.0 = 0
        call process_ptf
        call print_result
      end
    when tsel = "V" then
      do
        function = "Hold reason analysis..."
        "control display lock"
        "display panel(smpehd3)"
        action = "Hold reason / HOLD elements browsed"
        "tbmod smptable"
        outff.0 = 0
        call process_action
        call display_result
      end
    when tsel = "Q" then
      do
        function = "Hold reason analysis..."
        "control display lock"
        "display panel(smpehd3)"
        action = "Hold Reason / ++ Hold printed / Sysout class"sysoutc
        "tbmod smptable"
        outff.0 = 0
        call process_action

```

```

        call print_result
    end
    otherwise nop
end
end
"tbclose smptable"
end
/*****/
/* Sub-routines */
/*****/
browse_ptf:
    dsn = ""||strip(smppts,,"")||"("ptfnum)'"
    address ispexec
    "browse dataset("dsn)"
return
process_ptf:
    dsn = ""||strip(smppts,,"")||"("ptfnum)'"
    address tso "allocate file(ptf) da("dsn) shr reuse"
    address tso "execio * diskrt ptf (finis stem ptff."
    do i = 1 to ptff.0
        if substr(ptff.i,1,2) = "++" then
            do
                if substr(ptff.i,4,4) = "HOLD" then
                    do
                        if word(ptff.i,5) = "REASON("hreason)" then save = 1
                                                else save = 0
                    end
                else save = 0
            end
        if save = 1 then
            do
                outff.0 = outff.0 + 1
                j = outff.0
                outff.j = ptff.i
            end
        end
        outff.0 = outff.0 + 1
        j = outff.0
        outff.j = copies("=",80)
        address tso "free file(ptf)"
return
find_fmids:
    dsn = ""||strip(smppts,,"")||"("ptfnum)'"
    address tso "allocate file(ptf) da("dsn) shr reuse"
    address tso "execio * diskrt ptf (finis stem ptff."
    do k = 1 to ptff.0
        if substr(ptff.k,1,2) = "++" then
            do
                if substr(ptff.k,4,4) = "HOLD" then
                    do
                        if word(ptff.k,5) = "REASON("hreason)" then
                            do

```



```

                parse value word(ptff.k,4) with "FMID("fmid")"
            end
        end
    end
end
address tso "free file(ptf)"
return
display_result:
    address tso "allocate fi(outf) unit(vio) new
                lrecl(80) blksize(6160)
                recfm(f b) tracks space(5 1)"
    address tso "execio " outff.0 "diskw outf (finis stem outff."
    "lminit ddname(outf) dataid(outff)"
    "browse dataid(&outff)"
    address tso "free file(outf)"
return
print_result:
    address tso "allocate fi(outf) sysout("sysoutc")"
    address tso "execio " outff.0 "diskw outf (finis stem outff."
    address tso "free file(outf)"
return
process_action:
    outff.0 = 0
    address ispexec
    "tbtop smptable"
    "tbscan smptable arglist(hreason)"
    do while rc = 0
        call process_ptf
        "tbscan smptable arglist(hreason)"
    end
return
manage_smpts:
    address ispexec
    "tbopen tsmpts"
    select
        when rc = 00 then          /* OK                */
            do
            end
        when rc = 08 then          /* table does not exist */
            do
                "tbcreate tsmpts keys(product version level)
                names(dsname)"
                /* create a first record */
                PRODUCT = "0S390"
                VERSION = "029"
                level   = "0004"
                dsname  = "'SAMPLE.SMPPTS'"
                "tbadd tsmpts"
            end
        otherwise
            do
            end
    end
end

```

```

end
call sort_tsmpts
"addpop poploc(smpout) row(1) column(1)"
src = 1
do while src <> 0
  "tbdispl tsmpts panel(smpehd4)"
  if words(zcmd) >= 1 then
    do
      cmd      = word(zcmd,1)
      product = word(zcmd,2)
      select
        when cmd = 1 then
          do
            "tbttop tsmpts"
            "tbscan tsmpts arglist(product)
              condlist(ge)"
          end
        otherwise
          do
          end
        end
      end
    end
  else
    do
      pfkey_04 = pfkey4
      if pfkey_04 = "PF03" then
        do
          dsname = ""
          src = 0
          end
        select
          when act = S then
            do
              smppts = dsname
              src = 0
            end
          when act = I then
            do
              product = ""
              version = ""
              level   = ""
              dsname  = ""
              "addpop"
              "display panel(smpehd5)"
              if rc = 0 then
                do
                  "tbadd tsmpts"
                  call sort_tsmpts
                end
              end
              "rempop"
            end
          when act = R then

```

```

do
  "addpop"
  "display panel(smpehd5)"
  if rc = 0 then
    do
      "tbadd tsmpts"
      call sort_tsmpts
    end
  "rempop"
end
when act = D then
do
  "tbdelete tsmpts"
end
when act = U then
do
  "addpop"
  opro = product
  over = version
  olev = level
  odsn = dsname
  "display panel(smpehd5)"
  if rc = 0 then
    do
      npro = product
      nver = version
      nlev = level
      ndsn = dsname
      product = opro
      version = over
      level = olev
      dsname = odsn
      "tbdelete tsmpts"
      product = npro
      version = nver
      level = nlev
      dsname = ndsn
      "tbadd tsmpts"
      call sort_tsmpts
    end
  "rempop"
end
otherwise
do
end
end
end
end
"rempop"
"tbclose tsmpts"
return
sort_tsmpts:
  "tbsort tsmpts fields(product,c,a,version,c,a,level,c,a)"

```

return

## SMPEHD1

```
)panel
)Attr Default(%+_ )
£ type(text) intens(high) color(yellow)
@ type(text) intens(low) color(green)
¬ type(text) intens(low) color(green) hilite(reverse)
} type(text) intens(high) color(yellow)
> type(text) intens(low) color(turq)
] type(output) intens(high) color(yellow) caps(off)
[ type(output) intens(high) color(turq) caps(off)
) type(input) pas(on)
)body expand(??)
%?-?-SMP/E Hold Data Processing%-?
%Command ==>_ZCMD
+
>Before processing the SMP/E output, please verify the options below
>and hit the£ENTER>key to continue or hit>the£END>key to cancel.
+
@SMPOUT dataset to process:

+SMPOUT %==>_smpout +

@SMPPTS to be used to get information about hold data:

+SMPPTS %==>_smppts +

@SYSOUT Class for print request:

+SYSOUT Class %==>_z+

)init
.help = smpehd1h
.zvars = '(sysoutc)'
)proc
&pfkey1 = .pfkey
)end
```

## SMPEHD2

```
)Attr Default(%{_)
[ type(text) intens(low) color(green) hilite(reverse)
? type(output) intens(low) caps(off) color(turq)
¬ type(output) intens(high) caps(off) color(yellow)
£ type(output) intens(low) caps(off) color(green)
@ type(output) intens(high) color(green)
} type(text) intens(high) color(turq)
> type(text) intens(low) color(green)
] type(text) intens(low) color(red)
)Body Expand(??) Width(&ZSCREENW)
```

```

%-?-[Hold Reason items%-?-[
%Command ==>_ZCMD                    ? ?%Scroll ==>_amt {
%
]PTF "++HOLD" element actions: >S}Show >P}Print >B}Browse PTF text
]Hold Reason actions:                >V}View >Q}Print
{
  {PTF      FMID      Hold Reason Last action
%-  -----  -----  -----  -----
)Model
_Z-ptfnum  £fmid    £hreason    ?action
)Init
  &tset = ''
  .HELP = smpehd2h
  .ZVARS = '(TSEL)'
)Proc
  ver (&tset,list,S,s,B,b,V,v,p,P,q,Q)
)End

```

### SMPEHD3

```

)attr
£ TYPE(TEXT) INTENS(LOW) COLOR(TURQ)
{ TYPE(TEXT) INTENS(HIGH) COLOR(GREEN) HILITE(BLINK)
@ TYPE(TEXT) INTENS(LOW) COLOR(BLUE)
( TYPE(TEXT) INTENS(HIGH) COLOR(RED)
) TYPE(OUTPUT) INTENS(HIGH) COLOR(PINK) JUST(ASIS)
} type(text) intens(high) color(yellow)
> type(text) intens(low) color(turq)
)Body Expand(??) Width(&ZSCREENW)
{WW      WW 000000000000 RRRRRRRRRRRR KK      KK      }Time      ->&ZTIME
{WW      WW 000000000000 RRRRRRRRRRRR KK      KK      }Date      ->&ZDATE
{WW      WW 00      00 RR      RR KK      KK      }Julian    ->&ZJDATE
{WW      WW 00      00 RR      RR KK      KK      }System    ->&ZSYSID
{WW      WW 00      00 RRRRRRRRRRRR KKKKKKK      _____
{WW WW  WW 00      00 RRRRRRRRRRRR KKKKKKK      _____
{WW WWW WW 00      00 RR      RR      KK      KK
{WW WW WW WW 00      00 RR      RR      KK      KK
{WWW      WWW 00      00 RR      RR      KK      KK
{WWW      WWW 000000000000 RR      RR      KK      KK
{WW      WW 000000000000 RR      RR      KK      KK
{IIIIIIIII NN      NN      GGGGGGGGGG
{IIIIIIIII NNN      NN      GGGGGGGGGGGG
{II      NNNN      NN      GG      GG
(Please be patient, performing: {II      NN      NN      NN      GG
{II      NN      NN      NN      GG
{II      NN      NN      NN      GG
)FUNCTION {II      NN      NN      NN      GG      GGGGG
{II      NN      NN      NN      GG      GGGGG
{II      NN      NNNN      GG      GG
{II      NN      NNN      GG      GG
{IIIIIIIII NN      NN      GGGGGGGGGGGG
]SMP/E SMPOUT processing {IIIIIIIII NN      N      GGGGGGGGGG

```

```
)init
)proc
)end
```

## SMPEHD400

```
)ATTR
£ TYPE(PT)      /* PANEL TITLE */
_ TYPE(INPUT)   INTENS(HIGH) CAPS(ON)
+ TYPE(TEXT)    INTENS(LOW) color(turq)
' TYPE(TEXT)    INTENS(LOW) color(blue)
% TYPE(TEXT)    INTENS(high)
] TYPE(TEXT)    INTENS(high) color(yellow)
$ TYPE(OUTPUT)  INTENS(HIGH) color(green)
)BODY window(76 12)
£- - - - - SMPPTS Selection - - - - -
%COMMAND ==>_ZCMD          %SCROLL ==>_SAMT+
£- - - - -
]Action:+Select / Insert / Delete / Repeat / Update
£- - - - -
'   Product   Version Level CSI
£- - - - -
)MODEL clear(act)
_Z $z      + $z      + $z      +$z      +
)init
&act = ''
.ZVARS = '(ACT,product,version,level,DSNAME)'
.HELP = smpehd4h
)REINIT
  IF (.MSG=' ') &SELECT=' ' REFRESH(act)
)proc
VER(&ACT,LIST,S,I,D,R,U)
&pfkey4 = .pfkey
)END
SMPEHD5
)ATTR
£ TYPE(PT)      /* PANEL TITLE */
_ TYPE(INPUT)   INTENS(HIGH) CAPS(ON)
+ TYPE(TEXT)    INTENS(LOW)
% TYPE(TEXT)    INTENS(high)
$ TYPE(OUTPUT)  INTENS(HIGH)
)BODY window(60 07)
£- - - - - SMPPTS Entry - - - - -
+
+ Product:  _product+
+ Version:  _version+
+ Level:    _level +
+ SMPPTS:   _dsname      +
+
)init
.HELP = smpehd5h
```

```

)proc
&pfkey5 = .pfkey
)END

```

## SMPEHD1H

```

)attr default(%{_)
£ type(text) intens(high) color(yellow)
@ type(text) intens(low) color(green)
¬ type(text) intens(low) color(green) hilite(reverse)
} type(text) intens(high) color(yellow)
> type(text) intens(low) color(turq)
] type(output) intens(high) color(yellow) caps(off)
[ type(output) intens(high) color(turq) caps(off)
)body expand(??)
?-?-SMP/E Hold Data Processing%-?
%Command ==>_ZCMD 2 {
{
{This ISPF dialog automates the verification of a SMP/E@APPLY CHECK{output run
{by automating the analysis of@++ HOLD{elements from the SMPPTS dataset for
{PTFs which would be applied because of a BYPASS option.
{
{The selection panel allows you to specify the:
{
£SMPOUT: >The dataset that contains the output from the SMP/E
APPLY CHECK.
{
£SMPPTS: >The dataset name of the SMPPTS to obtain the HOLD
information from.
{
£SYSOUT Class: >The sysout class which is used by print requests
{
)init
)proc
)end

```

## SMPEHD2H

```

)attr default(%{_)
£ type(text) intens(high) color(yellow)
@ type(text) intens(low) color(green)
¬ type(text) intens(low) color(green) hilite(reverse)
} type(text) intens(high) color(yellow)
> type(text) intens(low) color(turq)
] type(text) intens(low) color(red)
)body expand(??)
?-?-Hold Reason items%-?
%Command ==>_ZCMD {
{
{This is a list of the PTFs that have hold conditions which you should
{investigate before applying. The options for the table rows are:
{

```

```

@ Selection based on]PTF number
{
£S: >Select the ++ HOLD information for this PTF
£P: >Print the ++ HOLD information for this PTF
£B: >Browse the PTF text
{
@ Selection based on]HOLD Reason
{
£V: >View all ++ HOLD informations for this HOLD reason
£Q: >Print all ++ HOLD informations for this HOLD reason
{
)init
)proc
)end

```

## SMPEHD4H

```

)attr default(%{_)
£ type(text) intens(high) color(yellow)
@ type(text) intens(low) color(green)
¬ type(text) intens(low) color(green) hilite(reverse)
} type(text) intens(high) color(yellow)
> type(text) intens(low) color(turq)
] type(text) intens(low) color(red)
)body expand(??)
?-?-SMPPTS Selection%?-?
%Command ==>_ZCMD {
{
{This is a list of pre-defined SMPPTS datasets.
{
{The options for the table rows are:
{
£S: >Select the SMPPTS dataset for the current analysis
£I: >Insert a new SMPPTS definition
£D: >Delete the SMPPTS definition
£R: >Repeat the SMPPTS definition
£U: >Update the SMPPTS definition
{
{
)init
)proc
)end

```

## SMPEHD5H

```

)attr default(%{_)
£ type(text) intens(high) color(yellow)
@ type(text) intens(low) color(green)
¬ type(text) intens(low) color(green) hilite(reverse)
} type(text) intens(high) color(yellow)
> type(text) intens(low) color(turq)
] type(text) intens(low) color(red)

```



```

)body expand(??)
?--SMPPTS Entry%?--?
%Command ==>_ZCMD
{
{
{This panel is used to define or update a SMPPTS definition.
{
{Product, Version and Level fields are used as comments for
{the SMPPTS definition.
{
)init
)proc
)end

```

## USING THE ISPF APPLICATION

The SMPEHD REXX EXEC can be called from a SMPOUT edit session initiated under SDSF with the SE command:

```

  Display Filter View Print Options Help
-----
SDSF JOB DATA SET DISPLAY - JOB U000189A (JOB05803)   LINE 1-5 (5)
COMMAND INPUT ==>                                     SCROLL ==> HALF
PREFIX=U000189*  DEST=(ALL)  OWNER=*
NP  DDNAME  STEPNAME PROCSTEP DSID OWNER   C DEST          REC-CNT PAGE
   JESMSGLG JES2             2 U000189 R LOCAL         20
   JESJCL   JES2             3 U000189 R LOCAL         27
   JESYSMSG JES2             4 U000189 R LOCAL         90
se  SMPOUT  HSCSMP   HSCSMP   103 U000189 R LOCAL         25
   SMPRPT  HSCSMP   HSCSMP   104 U000189 R LOCAL         373

```

In edit session, you can call SMPEHD:

```

SDSF EDIT      U000189E (JOB05803) SMPOUT              Session cancelled
Command ==> smpehd                                    Scroll ==> HALF
***** ***** Top of Data *****
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 1PAGE 0001 - NOW SET TO TARGET ZONE MVST100  DATE 10/08/00 TIME 13:54:
000002
000003 GIM42401I THE FOLLOWING PARAMETERS WERE SPECIFIED ON THE EXEC STATEM
000004     SET BDY(MVST100) .
000005 GIM20501I SET PROCESSING IS COMPLETE. THE HIGHEST RETURN CODE WAS 00
000006
000007
000008     APPLY S  (
000009     UW73524 /* PTF      RESOLVES AW45005 FOR UW68692 FMID(HBB6608) */
000010     UW70183 /* PTF      RESOLVES AW43836 FOR HDZ11E0 FMID(HDZ11E0) */
000011     UW73957 /* PTF      RESOLVES AW45267 FOR HDZ11E0 FMID(HDZ11E0) */
000012     UW73796 /* PTF      RESOLVES AW46082 FOR HDZ11E0 FMID(HDZ11E0) */
000013 /* UW70183   PTF      RESOLVES AW43836 FOR UW59033 FMID(HDZ11E0) */

```

```

000014      UQ46970 /* PTF      RESOLVES AQ41197 FOR HLE6609 FMID(HLE6609) */
000015      UW73616 /* PTF      RESOLVES AW46141 FOR HOPI280 FMID(HOPI280) */
000016      UW73824 /* PTF      RESOLVES AW46182 FOR HPRF310 FMID(HPRF310) */
000017      UQ44609 /* PTF      RESOLVES AQ38490 FOR HTCP380 FMID(HTCP380) */
000018 /* UQ44609      PTF      RESOLVES AQ38490 FOR UQ42015 FMID(HTCP380) */

```

Then you get the primary menu, where you should enter the name of the corresponding SMPPTS dataset:

```

----- SMP/E Hold Data Processing -----
Command ==>

Before processing the SMP/E output, please verify the options below
and hit the ENTER key to continue or hit the END key to cancel.

SMPDOUT dataset to process:
SMPDOUT ==> <= CURRENT EDIT SESSION =>

SMPPTS to be used to get information about hold data:
SMPPTS ==> YOUR.SMPPTS

SYSOUT Class for print request:
SYSOUT Class ==> A

```

If you enter a blank in the SMPPTS field, you will be able to access and to manage an ‘SMPPTS directory’ – you will not have to re-type the name of your common SMPPTS dataset each time you use the application.

```

----- SMP/E Hold Data Processing -----
Command ==>

Before processing the SMP/E output, please verify the options below
and hit the ENTER key to continue or hit the END key to cancel.

SMPDOUT dataset to process:
SMPDOUT ==> YOUR.SMPDOUT.LISTING

-----
| ----- SMPPTS Selection ----- Row 1 to 5 of 6 |
| COMMAND ==>                                SCROLL ==> PAGE |
| ----- |
| Action: Select / Insert / Delete / Repeat / Update |
| ----- |
|   Product   Version Level CSI |
| ----- |
|   CICS      120           'CICS.TS120.SMPPTS' |
|   DB2       510           'DB2.V510.SMPPTS' |
|   DB2       610           'DB2.V610.SMPPTS' |
|   OS390     027           'OS390R27.SMPPTS' |
| S OS390     029           'OS390R29.SMPPTS' |
| ----- |

```

Once you have selected the SMPPTS dataset, you can start the SMPOUT analysis. Then you get the following kind of display:

```

----- Hold Reason items ----- Row 1 to 11 of 11
Command ==>                               Scroll ==>

PTF "++HOLD" element actions:  S Show  P Print  B Browse PTF text
Hold Reason actions:          V View  Q Print

   PTF      FMID      Hold Reason Last action
-----
S UW68490  HPRF310  ACTION
  UW68615  HPRF310  ACTION
  UW68616  JPRF311  ACTION
  UW73796  HDZ11E0  ACTION
  UW67583  HPRF310  DOC
  UW68490  HPRF310  DOC
  UW68615  HPRF310  DOC
  UW68616  JPRF311  DOC
  UW69791  HPRF310  DOC
  UW70022  HPRF310  DOC
  UW71197  HPRF310  DOC
***** Bottom of data *****

```

Then you can select and browse the ++HOLD text:

```

Menu Utilities Compilers Help
-----
BROWSE   SYS00282.T142357.RA000.U000189.R0180894   Line 00000000 Col 001 080
Command ==>                               Scroll ==> HALF
***** Top of Data *****
++ HOLD(UW68490) SYS FMID(HPRF310) REASON(ACTION) DATE(00076)
COMMENT
  (If you have an exit 06, you are using it with channel
   attached printers, and you are testing the contents in exit 06
   of XTP6PROC, then you may not want to apply this PTF until
   you have modified exit 06. This PTF changes the value of
   XTP6PROC from DDffnnnn
   where DD is a constant,
   and ff is the FSA ID assigned to the FSA by JES,
   and nnnn is the 4 digit device address.
   to the label of the PRINTDEV statement. This makes the value
   of XTP6PROC consistent across all attachments.
   *****
   If you have modified your source code for APSGEXTP, save it
   before applying this PTF as this PTF will replace your
   copy of APSGEXTP. You will need to merge your changes
   into the new APSGEXTP.).

```

## Speedy panel access

Over the years I have often seen requests from ISPF dialog developers for methods of accessing the current screen name. With the release of OS/390 Version 2 Release 10 of ISPF a new in-built variable ZPANELID is available which provides users with the ISPF panel name.

What this means, for example, is that it is now very easy to create a command to jump immediately to the source of the current panel for editing/correction. This feature can be especially helpful during ISPF dialog development. The code shown below demonstrates how to do this:

```
/* REXX */  
ADDRESS ISPEXEC  
'VGET (ZPANELID)'  
'LMINIT DATAID(DID) DDNAME('ISPLIB')'  
'EDIT DATAID('DID') MEMBER('ZPANELID')
```

Interestingly I tried this code on my OS/390 Version 2 Release 8 system and it worked fine as well.

---

*Systems Programmer (UK)*

© Xephon 2001

---

### **Free weekly Enterprise IS News**

A weekly enterprise-oriented news service is available free from Xephon. Each week, subscribers receive an e-mail listing around 40 news items, with links to the full articles on our Web site. The articles are copyrighted by Xephon – they are not syndicated, and are not available from other sources.

To subscribe to this newsletter, send an e-mail to [news-list-request@xephon.com](mailto:news-list-request@xephon.com), with the word subscribe in the body of the message. You can also subscribe to this and other Xephon e-mail newsletters by visiting Xephon's home page at, <http://www.xephon.com>, which contains a simple subscription form.

## Formatting internal system trace table entries – part 2

*Please note, the author has informed us of an error in the May edition of MVS Update on page 68, line 34 that could cause a loop in the program's execution. The instruction in question immediately precedes the one named PCPGETXL:*

```
BO    PGNXTCDE          BRANCH IF NOT
```

*It must be changed to read:*

```
BO    PCNXTCDE          BRANCH IF NOT
```

*The code has been changed on the Xephon Web site.*

*This month we conclude our article about formatting internal system trace table entries.*

```

MVC   PPCDNSTC,=CL10'OLD-STATE' CONSTANT TO OUTPUT AREA
MVC   PPGHOLD,TTEPCADR    NEW PROGRAM STATE TO HOLD AREA
NI    PPGHOLD+3,255-1    RESET STATE BIT
MVC   PPCDSTAT,=CL4'PROB' ASSUME PROBLEM STATE AFTER PR
TM    TTEPCAB4,TTEPCPS   TEST ASSUMPTION
BO    PPGPCPRB           BRANCH IF VALID
MVC   PPCDSTAT,=CL4'SUPR' SHOW SUPERVISOR STATE AFTER PR
SPACE
PPGPCPRB MVC   PPCDRADC,=CL8'RET-ADR' CONSTANT TO OUTPUT AREA
UNPK  PPCDRAD(9),PPGHOLD(5) STOW ADDRESS OF NEW INSTRUCTION
TR    PPCDRAD,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI   PPCDRAD+8,C' '     REMOVE THE TRASH
SPACE
B     PPGBASR5           TRANSCRIBE DESCRIPTION OF DATA
SPACE
DROP  R1                FORGET PPGPCD
EJECT

*****
*          FORMAT A PR TRACE TABLE ENTRY          *
*****
SPACE
PPGPR  DS      0H
LA     R1,PRINTOUT
USING PPGPRD,R1          ESTABLISH PPGPRD ADDRESSABILITY
SPACE
MVC   PPRDKEYC,=CL8'PSW KEY' CONSTANT TO OUTPUT AREA
UNPK  PPRDKEY(3),TTEPRPKY(2) STOW PSW KEY IN OUTPUT AREA
TR    PPRDKEY,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI   PPRDKEY+1,C' '     REMOVE
MVI   PPRDKEY+2,C' '     THE TRASH

```

```

SPACE
MVC  PPRDNIDC,=CL9'NEW-ASID' CONSTANT TO OUTPUT AREA
UNPK PPRDNID(5),TTEPRASD(3) STO ADDRESS OF NEW ASID IN OUTPUT
TR   PPRDNID,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  PPRDNID+4,C' '      REMOVE THE TRASH
SPACE
MVC  PPRDNSTC,=CL10'NEW-STATE' CONSTANT TO OUTPUT AREA
MVC  PPGHOLD,TTEPERTAD      NEW PROGRAM STATE TO HOLD AREA
NI   PPGHOLD+3,255-1      RESET STATE BIT
MVC  PPRDSTAT,=CL4'PROB' ASSUME PROBLEM STATE AFTER PR
TM   TTEPTB4,TTEPRPS      TEST ASSUMPTION
BO   PPGPRPRB              BRANCH IF VALID
MVC  PPRDSTAT,=CL4'SUPR' SHOW SUPERVISOR STATE AFTER PR
SPACE
PPGPRPRB MVC  PPRDRADC,=CL8'NEW-ADR' CONSTANT TO OUTPUT AREA
UNPK PPRDRAD(9),PPGHOLD(5) STOW ADDRESS OF NEW INSTRUCTION
TR   PPRDRAD,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  PPRDRAD+8,C' '      REMOVE THE TRASH
SPACE
MVC  PPRDAADC,=CL14'ADR AFT OF PC' CONSTANT TO OUTPUT AREA
UNPK PPRDAAD(9),TTEPRFAD(5) STOW ADDRESS OF NEW INSTRUCTION
TR   PPRDAAD,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  PPRDAAD+8,C' '      REMOVE THE TRASH
SPACE
B    PPGBASR5              TRANSCRIBE DESCRIPTION OF DATA
SPACE
DROP R1                    FORGET PPGPRD
EJECT

*****
*          FORMAT A PT TRACE TABLE ENTRY          *
*****
SPACE
PPGPT  DS    0H
LA     R1,PRINTOUT
USING PPGPTD,R1            ESTABLISH PPGPTD ADDRESSABILITY
SPACE
MVC  PPTDKEYC,=CL8'PSW KEY' CONSTANT TO OUTPUT AREA
UNPK PPTDKEY(3),TTEPTPKY(2) STOW PSW KEY IN OUTPUT AREA
TR   PPTDKEY,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  PPTDKEY+1,C' '      REMOVE
MVI  PPTDKEY+2,C' '      THE TRASH
SPACE
MVC  PPTDNIDC,=CL9'NEW-ASID' CONSTANT TO OUTPUT AREA
UNPK PPTDNID(5),TTEPTASD(3) STO ADDRESS OF NEW ASID IN OUTPUT
TR   PPTDNID,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  PPTDNID+4,C' '      REMOVE THE TRASH
SPACE
MVC  PPTDNSTC,=CL10'NEW-STATE' CONSTANT TO OUTPUT AREA
MVC  PPGHOLD,TTEPTADR      NEW PROGRAM STATE TO HOLD AREA
NI   PPGHOLD+3,255-1      RESET STATE BIT
MVC  PPTDSTAT,=CL4'PROB' ASSUME PROBLEM STATE AFTER PT
TM   TTEPTAB4,TTEPTPS      TEST ASSUMPTION

```

```

BO      PPGPTRB          BRANCH IF VALID
MVC     PPTDSTAT,=CL4'SUPR' SHOW SUPERVISOR STATE AFTER PT
SPACE
PPGTPRB MVC PPTDRADC,=CL8'NEW-ADR' CONSTANT TO OUTPUT AREA
UNPK    PPTDRAD(9),PPGHOLD(5) STOW ADDRESS OF NEW INSTRUCTION
TR      PPTDRAD,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI     PPTDRAD+8,C' '   REMOVE THE TRASH
SPACE
B       PPGBASR5        TRANSCRIBE DESCRIPTION OF DATA
SPACE
DROP    R1              FORGET PPGPTD
EJECT

*****
*       FORMAT AN SSAR TRACE TABLE ENTRY                               *
*****

SPACE
PPGSSAR DS  0H
LA      R1,PRINTOUT
USING  PPGSSAD,R1      ESTABLISH PPGSSAD ADDRESSABILITY
SPACE
MVC     PSRDNIDC,=CL10'NEW-SASID' CONSTANT TO OUTPUT AREA
UNPK    PSRDNID(5),TTESSASD(3) STO ADDRESS OF NEW SECONDARY ASID
TR      PSRDNID,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI     PSRDNID+4,C' '   REMOVE THE TRASH
SPACE
PPGBASR5 BAS R5,PPGPUTLS TRANSCRIBE DESCRIPTION OF DATA
MVC     PRINTOUT,CLEAR  BLANK OUTPUT AREA
SPACE 1
B       PPG4MAT        FORMAT SSAR ENTRY
SPACE
DROP    R1              FORGET PPGSSAD
EJECT

*****
*       FORMAT A PORTION OF AN ENTRY FOR AN ALTERNATE CPU RECOVERY TTE *
*****

SPACE
PPGACR  DS  0H
LA      R1,PRINTOUT
USING  PPG017D,R1      ESTABLISH PPG017D ADDRESSABILITY
SPACE
MVC     P17DTCBC,=CL4'TCB' CONSTANT TO OUTPUT AREA
UNPK    P17DTCBC(9),TTE017TB(5) ADDRESS OF TCB TO OUTPUT AREA
TR      P17DTCBC,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI     P17DTCBC+8,C' '   REMOVE THE TRASH
SPACE
MVC     P17DHAC,=CL5'HASID' CONSTANT TO OUTPUT AREA
UNPK    P17DHID(5),TTE017HA(3) HOME ASID TO OUTPUT AREA
TR      P17DHID,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI     P17DHID+4,C' '   REMOVE THE TRASH
SPACE
MVC     P17CPUPA,=CL5'CPUPA' CONSTANT TO OUTPUT AREA
UNPK    P17CPID(5),TTE017LP(3) LOGICAL PROCESR ID TO OUTPUT AREA

```

```

TR    P17CPID,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI   P17CPID+4,C' '      REMOVE THE TRASH
SPACE
MVC   P17FLAG,=CL5'AFLAG' CONSTANT TO OUTPUT AREA
UNPK  P17FG(3),TTE017FG(2) ACR FLAG BYTE TO OUTPUT AREA
TR    P17FG,PATRANS-240  CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI   P17FG+2,C' '      REMOVE THE TRASH
SPACE
MVC   P17EPSWC,=CL5'EPSW' CONSTANT TO OUTPUT AREA
UNPK  P17EPSW(9),TTE017AD(5) ADDRESS OF FAILING PROCESS TO OUT
TR    P17EPSW,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI   P17EPSW+8,C' '    REMOVE THE TRASH
SPACE
MVC   P17SUPRC,=CL6'SUPER' CONSTANT TO OUTPUT AREA
UNPK  P17SUPER(9),TTE017PS(5) SUPER LOCK TO OUTPUT
TR    P17SUPER,PATRANS-240 CONVERT VALUE TO EBCDIC -LEGIBILITY
MVI   P17SUPER+8,C' '   REMOVE THE TRASH
SPACE
MVC   P17LOCLC,=CL6'LOCAL' CONSTANT TO OUTPUT AREA
UNPK  P17LOCAL(9),TTE017PL(5) ADDRESS OF LOCK LOCKS TO OUTPUT
TR    P17LOCAL,PATRANS-240 CONVERT VALUE TO EBCDIC -LEGIBILITY
MVI   P17LOCAL+8,C' '   REMOVE THE TRASH
SPACE
BAS   R5,PPGPUTLS          TRANSCRIBE ACR DATA
MVC   PRINTOUT,CLEAR      BLANK OUTPUT AREA
SPACE
CLI   PPGTNAME,C' '      TEST IF TASK SELECTED
BE    PPGSKIP             BRANCH IF NOT
SPACE 1
OI    PPGSW,2             DO NOT ATTEMPT COMPUTATION OF OFFSET
B     PPG4MAT             FORMAT ACR ENTRY
SPACE
DROP  R1                  FORGET PPG017D
EJECT

*****
*          FORMAT A PORTION OF AN ENTRY FOR AN EXTERNAL CALL          *
*          EXTERNAL INTERRUPT TRACE TABLE ENTRY                      *
*****
SPACE
PPGCALL DS    0H
SPACE
CLI   PPGTNAME,C' '      TEST IF TASK NAME IS PRESENT
BE    *+8                BRANCH IF NOT
NI    PPGSW,255-2        ALLOW COMPUTATION OF OFFSET IN PGM
SPACE
LA    R1,PRINTOUT
USING PPG303D,R1         ESTABLISH PPG303D ADDRESSABILITY
SPACE
BAS   R5,PPCPFTCB        FORMAT TCB, PSW, AND ASID'S
SPACE
MVC   P30DCPUC,=CL9'CPU-ADDR' CONSTANT TO OUTPUT AREA
UNPK  P30DCPU(5),TTE303CD(3) ISSUING PROCESSOR ADDR TO OUTPUT

```



```

TR    P30DCPU,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   P30DCPU+4,C' '      REMOVE THE TRASH
SPACE
MVC   P30DXICC,=CL6'ICODE' CONSTANT TO OUTPUT AREA
UNPK  P30DXIC(5),TTE303CD+2(3) EXT INTERRUPT CODE TO OUTPUT
TR    P30DXIC,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   P30DXIC+4,C' '      REMOVE THE TRASH
SPACE
MVC   P30DSIGC,=CL5'SIGP' CONSTANT TO OUTPUT AREA
UNPK  P30DSIG(3),TTE303PB(2) EXT SIGP VALUE TO OUTPUT AREA
TR    P30DSIG,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   P30DSIG+2,C' '      REMOVE THE TRASH
SPACE
MVC   P30DLOKC,=CL8'CPU-LOK' CONSTANT TO OUTPUT AREA
UNPK  P30DLOK(9),TTE303LH(5) CPU LOCKS TO OUTPUT AREA
TR    P30DLOK,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   P30DLOK+8,C' '      REMOVE THE TRASH
SPACE
BAS   R5,PPGPUTLS          TRANSCRIBE CALL DATA
MVC   PRINTOUT,CLEAR       BLANK OUTPUT AREA
SPACE
MVC   PATGILL,TTE303P2     STOW RIGHT HALF OF INTERRUPTED PSW
B     PPG4MAT              FORMAT CALL ENTRY
SPACE
DROP  R1                  FORGET PPG303D
EJECT

*****
*          FORMAT A PORTION OF AN ENTRY FOR A CLOCK COMPARATOR          *
*          EXTERNAL INTERRUPT TRACE TABLE ENTRY                        *
*****
SPACE
PPGCLKC DS  0H
SPACE
CLI   PPGTNAME,C' '      TEST IF TASK NAME IS PRESENT
BE    *+8                BRANCH IF NOT
NI    PPGSW,255-2        ALLOW COMPUTATION OF OFFSET IN PGM
SPACE
LA    R1,PRINTOUT
USING PPG403D,R1         ESTABLISH PPG403D ADDRESSABILITY
SPACE
BAS   R5,PPCPFTCB        FORMAT TCB, PSW, AND ASID'S
SPACE
MVC   P40DXICC,=CL6'ICODE' CONSTANT TO OUTPUT AREA
UNPK  P40DXIC(5),TTE403CD+2(3) EXT INTERRUPT CODE TO OUTPUT
TR    P40DXIC,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   P40DXIC+4,C' '      REMOVE THE TRASH
SPACE
MVC   P40DLOKC,=CL7'CPULOK' CONSTANT TO OUTPUT AREA
UNPK  P40DLOK(9),TTE403LH(5) CPU LOCKS TO OUTPUT AREA
TR    P40DLOK,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   P40DLOK+8,C' '      REMOVE THE TRASH
SPACE

```

```

MVC P40DTQTC,=CL5'QTCB ' CONSTANT TO OUTPUT AREA
UNPK P40DTTCB(9),TTE403TT(5) ADDRES OF TQE TCB TO OUTPUT AREA
TR P40DTTCB,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P40DTTCB+8,C' ' REMOVE THE TRASH
SPACE
MVC P40DTQAC,=CL4'QID' CONSTANT TO OUTPUT AREA
UNPK P40DTAID(5),TTE403TA(3) TQE-TCB'S ASID TO OUTPUT AREA
TR P40DTAID,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P40DTAID+4,C' ' REMOVE THE TRASH
SPACE
BAS R5,PPGPUTLS TRANSCRIBE CLKC DATA
MVC PRINTOUT,CLEAR BLANK OUTPUT AREA
SPACE
MVC PATGILL,TTE403P2 STOW RIGHT HALF OF INTERRUPTED PSW
B PPG4MAT FORMAT CLKC ENTRY
SPACE
DROP R1 FORGET PPG403D
EJECT

```

```

*****
*          FORMAT A PORTION OF AN ENTRY FOR A TASK DISPATCH TTE          *
*****

```

```

SPACE
PPGDSP DS 0H
SPACE
CLI PPGTNAME,C' ' TEST IF TASK NAME IS PRESENT
BE *+8 BRANCH IF NOT
NI PPGSW,255-2 ALLOW COMPUTATION OF OFFSET IN PGM
SPACE
LA R1,PRINTOUT
USING PPG00FD,R1 ESTABLISH PPG00FD ADDRESSABILITY
SPACE
BAS R5,PPCPFTCB FORMAT TCB, PSW, AND ASID'S
SPACE
MVC P0FDG0C,=CL5'GPR0' CONSTANT TO OUTPUT AREA
UNPK P0FDG0(9),TTE00FG0(5) CONTENTS OF GPR0 OUTPUT AREA
TR P0FDG0,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI P0FDG0+8,C' ' REMOVE THE TRASH
SPACE
MVC P0FDG1C,=CL5'GPR1' CONSTANT TO OUTPUT AREA
UNPK P0FDG1(9),TTE00FG1(5) CONTENTS OF GPR1 OUTPUT AREA
TR P0FDG1,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI P0FDG1+8,C' ' REMOVE THE TRASH
SPACE
MVC P0FSUPRC,=CL6'LOCAL' CONSTANT TO OUTPUT AREA
UNPK P0FSUPER(9),TTE00FPL(5) LOCAL LOCK TO OUTPUT
TR P0FSUPER,PATRANS-240 CONVERT VALUE TO EBCDIC -LEGIBILITY
MVI P0FSUPER+8,C' ' REMOVE THE TRASH
SPACE
BAS R5,PPGPUTLS TRANSCRIBE DSP DATA
MVC PRINTOUT,CLEAR BLANK OUTPUT AREA
SPACE
MVC PATGILL,TTE00FP2 STOW RIGHT HALF OF INTERRUPTED PSW

```

```

      B      PPG4MAT          FORMAT DSP ENTRY
      SPACE
      DROP  R1              FORGET PPG00FD
      EJECT
*****
*      FORMAT A PORTION OF AN ENTRY FOR AN EXTERNAL INTERRUPT      *
*****
      SPACE
PPGEXT DS      0H
      SPACE
      CLI  PPGTNAME,C' '    TEST IF TASK NAME IS PRESENT
      BE   *+8              BRANCH IF NOT
      NI   PPGSW,255-2      ALLOW COMPUTATION OF OFFSET IN PGM
      SPACE
      LA   R1,PRINTOUT
      USING PPG003D,R1      ESTABLISH PPG003D ADDRESSABILITY
      SPACE
      BAS  R5,PPCPFTCB      FORMAT TCB, ASID'S, AND PSW
      SPACE
      MVC  P03DXICC,=CL9'INT-CODE' CONSTANT TO OUTPUT AREA
      UNPK P03DXIC(9),TTE003CD(5) EXT INTERRUPT CODE TO OUTPUT AREA
      TR   P03DXIC,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
      MVI  P03DXIC+8,C' '    REMOVE THE TRASH
      SPACE
      MVC  P03DLCLC,=CL5'LOCAL' CONSTANT TO OUTPUT AREA
      UNPK P03DLOCL(9),TTE003PL(5) LOCAL LOCKS TO OUTPUT AREA
      TR   P03DLOCL,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
      MVI  P03DLOCL+8,C' '    REMOVE THE TRASH
      SPACE
      BAS  R5,PPGPUTLS      TRANSCRIBE EXT DATA
      MVC  PRINTOUT,CLEAR    BLANK OUTPUT AREA
      SPACE
      MVC  PATGILL,TTE003P2  STOW RIGHT HALF OF INTERRUPTED PSW
      B    PPG4MAT          FORMAT EXT ENTRY
      SPACE
      DROP  R1              FORGET PPG00BD
      EJECT
*****
*      FORMAT A PORTION OF AN ENTRY FOR AN I/O INTERRUPT TTE      *
*****
      SPACE
PPGIO  DS      0H
      SPACE
      CLI  PPGTNAME,C' '    TEST IF TASK NAME IS PRESENT
      BE   *+8              BRANCH IF NOT
      NI   PPGSW,255-2      ALLOW COMPUTATION OF OFFSET IN PGM
      SPACE
      LA   R1,PRINTOUT
      USING PPG00BD,R1      ESTABLISH PPG00BD ADDRESSABILITY
      SPACE
      BAS  R5,PPCPFTCB      FORMAT TCB, ASID'S, AND PSW
      SPACE

```

```

MVC P0BDADRC,=CL6'UCB-AD' CONSTANT TO OUTPUT AREA
UNPK P0BDADR(9),TTE00BUB(5) ADDRESS OF UCB TO OUTPUT AREA
TR P0BDADR,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P0BDADR+8,C' ' REMOVE THE TRASH
SPACE
MVC P0BDNUMC,=CL4'DEV#' CONSTANT TO OUTPUT AREA
UNPK P0BDNUM(5),TTE00BDN(3) DEVICE NUMBER TO OUTPUT AREA
TR P0BDNUM,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P0BDNUM+4,C' ' REMOVE THE TRASH
SPACE
MVC P0BDSTAC,=CL4'STAT' CONSTANT TO OUTPUT AREA
UNPK P0BDSTAT(5),TTE00BDS(3) DEVICE AND CHAN STAT TO OUT AREA
TR P0BDSTAT,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P0BDSTAT+4,C' ' REMOVE THE TRASH
SPACE
MVC P0BRESDC,=CL8'RESIDUAL' CONSTANT TO OUTPUT AREA
UNPK P0BRSDUL(5),TTE00BCT(3) RESIDUAL COUNT TO OUTPUT AREA
TR P0BRSDUL,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P0BRSDUL+4,C' ' REMOVE THE TRASH
SPACE
BAS R5,PPGPUTLS TRANSCRIBE I/O DATA
MVC PRINTOUT,CLEAR BLANK OUTPUT AREA
SPACE
MVC PATGILL,TTE00BP2 STOW RIGHT HALF OF INTERRUPTED PSW
B PPG4MAT FORMAT I/O ENTRY
SPACE
DROP R1 FORGET PPG00BD
EJECT

```

```

*****
*          FORMAT A PORTION OF AN ENTRY FOR A PROGRAM INTERRUPT TTE          *
*****

```

```

SPACE
PPGPGM DS 0H
SPACE
CLI PPGTNAME,C' ' TEST IF TASK NAME IS PRESENT
BE *+8 BRANCH IF NOT
NI PPGSW,255-2 ALLOW COMPUTATION OF OFFSET IN PGM
SPACE
LA R1,PRINTOUT
USING PPG007D,R1 ESTABLISH PPG007D ADDRESSABILITY
SPACE
BAS R5,PPCPFTCB FORMAT TCB, ASID'S, AND PSW
SPACE
MVC P07DILC,=CL8'INST-LEN' CONSTANT TO OUTPUT AREA
UNPK P07DIL(5),TTE007IL(3) INSTRUCTION LENGTH TO OUTPUT AREA
TR P07DIL,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P07DIL+4,C' ' REMOVE THE TRASH
SPACE
MVC P07DICC,=CL9'INT-CODE' CONSTANT TO OUTPUT AREA
UNPK P07DCODE(5),TTE007CD(3) INTERRUPT CODE TO OUTPUT AREA
TR P07DCODE,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P07DCODE+4,C' ' REMOVE THE TRASH

```

```

SPACE
BAS  R5,PPGPUTLS          TRANSCRIBE PROGRAM INTERRUPT DATA
MVC  PRINTOUT,CLEAR      BLANK OUTPUT AREA
SPACE
MVC  PATGILL,TTE007P2    STOW RIGHT HALF OF INTERRUPTED PSW
B    PPG4MAT             FORMAT PGM ENTRY
SPACE
DROP R1                  FORGET PPG007D
EJECT
*****
*          FORMAT A PORTION OF AN ENTRY FOR A RECOVERY EVENT TTE          *
*****
PPGRCVY DS    0H
LA     R1,PRINTOUT
USING PPG01DD,R1          ESTABLISH PPG01DD ADDRESSABILITY
SPACE
MVC   P1DDTCBC,=CL4'TCB'  CONSTANT TO OUTPUT AREA
UNPK  P1DDTCB(9),TTE01DTB(5) ADDRESS OF TCB TO OUTPUT AREA
TR    P1DDTCB,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI   P1DDTCB+8,C' '      REMOVE THE TRASH
SPACE
MVC   P1DDHAC,=CL5'HASID' CONSTANT TO OUTPUT AREA
UNPK  P1DDHID(5),TTE01DHA(3) HOME ASID TO OUTPUT AREA
TR    P1DDHID,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI   P1DDHID+4,C' '      REMOVE THE TRASH
SPACE
MVC   P1DDSI,=CL9'SUB-CODE' CONSTANT TO OUTPUT AREA
UNPK  P1DDSI(5),TTE01DSI(3) SUBTYPE CODE TO OUTPUT AREA
TR    P1DDSI,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   P1DDSI+4,C' '      REMOVE THE TRASH
SPACE
MVC   P1DDLHC,=CL10'CPU-LOCKS' CONSTANT TO OUTPUT AREA
UNPK  P1DDLH(9),TTE01DSI(5) CPU LOCKS HELD TO OUTPUT AREA
TR    P1DDLH,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   P1DDLH+8,C' '      REMOVE THE TRASH
SPACE
MVC   P1DDW1C,=CL6'WORD1 ' CONSTANT TO OUTPUT AREA
UNPK  P1DDW1(9),TTE01DU1(5) 1ST UNIQUE WORD TO OUTPUT AREA
TR    P1DDW1,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   P1DDW1+8,C' '      REMOVE THE TRASH
SPACE
MVC   P1DDW2C,=CL6'WORD2 ' CONSTANT TO OUTPUT AREA
UNPK  P1DDW2(9),TTE01DU2(5) 2ND UNIQUE WORD TO OUTPUT AREA
TR    P1DDW2,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   P1DDW2+8,C' '      REMOVE THE TRASH
SPACE
MVC   P1DDW8C,=CL6'WORD8 ' CONSTANT TO OUTPUT AREA
UNPK  P1DDW8(9),TTE01DU8(5) 8TH UNIQUE WORD TO OUTPUT AREA
TR    P1DDW8,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   P1DDW8+8,C' '      REMOVE THE TRASH
SPACE

```

```

PCP4MAT  BAS   R5,PPGPUTLS           TRANSCRIBE PROSE FOR TTE
          MVC   PRINTOUT,CLEAR       BLANK OUTPUT AREA
          SPACE
          OI    PPGSW,2              DO NOT ATTEMPT COMPUTATION OF OFFSET
          B     PPG4MAT              FORMAT TTE
          SPACE
          DROP  R1                   FORGET PPG01DD
          EJECT
*****
*          FORMAT A PORTION OF AN ENTRY FOR A RESUME SUBCHANNEL TTE          *
*****
          SPACE
PPGRSCH  DS     0H
          LA    R1,PRINTOUT
          USING PPG401D,R1           ESTABLISH PPG401D ADDRESSABILITY
          SPACE
          BAS   R5,PPCPFSC           FORMAT COMMON SUBCHANNEL SEGMENT
          SPACE
          MVC   P41DIOSC,=CL8'IOSB-AD' CONSTANT TO OUTPUT AREA
          UNPK  P41DIOS(9),TTE401IO(5) ADDRESS OF IOSB TO OUTPUT AREA
          TR    P41DIOS,PATRANS-240  CONVERT VALUE TO EBCDIC-LEGIBILITY
          MVI   P41DIOS+8,C' '       REMOVE THE TRASH
          B     PCP4MAT              TRANSCRIBE RSCH DATA
          SPACE
          DROP  R1                   FORGET PPG401D
          EJECT
*****
*          FORMAT A PORTION OF AN ENTRY FOR AN SRB DISPATCH INTERRUPT        *
*****
          SPACE
PPGSRB   DS     0H
          SPACE
          CLI   PPGTNAME,C' '        TEST IF TASK NAME IS PRESENT
          BE    *+8                   BRANCH IF NOT
          NI    PPGSW,255-2          ALLOW COMPUTATION OF OFFSET IN PGM
          SPACE
          LA    R1,PRINTOUT
          USING PPG10FD,R1           ESTABLISH PPG10FD ADDRESSABILITY
          SPACE
          MVC   PAFDWRKC,=CL5'WORK'  CONSTANT TO OUTPUT AREA
          UNPK  PAFDWORK(9),TTE10FTB(5) WORK UNIT ADDRESS TO OUTPUT AREA
          TR    PAFDWORK,PATRANS-240  CONVERT VALUE TO EBCDIC-LEGIBILITY
          MVI   PAFDWORK+8,C' '       REMOVE THE TRASH
          SPACE
          MVC   PAFDHAC,=CL5'HASID'  CONSTANT TO OUTPUT AREA
          UNPK  PAFDHID(5),TTE10FHA(3) HOME ASID TO OUTPUT AREA
          TR    PAFDHID,PATRANS-240  CONVERT VALUE TO EBCDIC - LEGIBILITY
          MVI   PAFDHID+4,C' '        REMOVE THE TRASH
          SPACE
          MVC   PAFDRAC,=CL10'RLATD-ASID' CONSTANT TO OUTPUT AREA
          UNPK  PAFDRID(5),TTE10FAP(3) RELATED ASID TO OUTPUT AREA
          TR    PAFDRID,PATRANS-240  CONVERT VALUE TO EBCDIC - LEGIBILITY

```

```

MVI   PAFDRID+4,C' '      REMOVE THE TRASH
SPACE
MVC   PAFDP1C,=CL5'PSW1'  CONSTANT TO OUTPUT AREA
UNPK  PAFDPSW1(9),TTE10FP1(5) LEFT HALF OF PSW OUTPUT AREA
TR    PAFDPSW1,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   PAFDPSW1+8,C' '      REMOVE THE TRASH
SPACE
MVC   PAFDP2C,=CL5'PSW2'  CONSTANT TO OUTPUT AREA
UNPK  PAFDPSW2(9),TTE10FP2(5) RIGHT HALF OF PSW OUTPUT AREA
TR    PAFDPSW2,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   PAFDPSW2+8,C' '      REMOVE THE TRASH
EJECT
MVC   PAFDG0C,=CL5'GPR0'  CONSTANT TO OUTPUT AREA
UNPK  PAFDG0(9),TTE10FG0(5) CONTENTS OF GPR0 OUTPUT AREA
TR    PAFDG0,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI   PAFDG0+8,C' '      REMOVE THE TRASH
SPACE
MVC   PAFDG1C,=CL5'GPR1'  CONSTANT TO OUTPUT AREA
UNPK  PAFDG1(9),TTE10FG1(5) CONTENTS OF GPR1 OUTPUT AREA
TR    PAFDG1,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI   PAFDG1+8,C' '      REMOVE THE TRASH
SPACE
MVC   PAFCPUPA,=CL6'CPU-AF' CONSTANT TO OUTPUT AREA
UNPK  PAFCPID(5),TTE10FFN(3) CPU AFFINITY TO OUTPUT AREA
TR    PAFCPID,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   PAFCPID+4,C' '      REMOVE THE TRASH
SPACE
MVC   PAFFLAGC,=CL5'FLAGS' CONSTANT TO OUTPUT AREA
UNPK  PAFFLAG(5),TTE10FSF(3) FLAG + HLHI CODE TO OUTPUT AREA
TR    PAFFLAG,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI   PAFFLAG+4,C' '      REMOVE THE TRASH
SPACE
BAS   R5,PPGPUTLS
MVC   PRINTOUT,CLEAR      BLANK OUTPUT AREA
SPACE
MVC   PATGILL,TTE10FP2    STOW RIGHT HALF OF INTERRUPTED PSW
B     PPG4MAT             FORMAT SRB ENTRY
SPACE
DROP  R1                 FORGET PPG10FD
EJECT
*****
*           FORMAT A PORTION OF AN ENTRY FOR A CLEAR SUBCHANNEL TTE           *
*           AND A SUBCHANNEL TYPE MAJOR ID                                     *
*****
SPACE
PPGTSCH DS   0H
PPGCSCH DS   0H
LA     R1,PRINTOUT
USING PPG301D,R1         ESTABLISH PPG301D ADDRESSABILITY
SPACE
BAS   R5,PPCPFSCHE      FORMAT COMMON SUBCHANNEL SEGMENT
SPACE

```

```

MVC P31DIOQC,=CL7'IOQ-ADR' CONSTANT TO OUTPUT AREA
UNPK P31DIOQ(9),TTE301IQ(5) ADDRESS OF IOQ TO OUTPUT AREA
TR P31DIOQ,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P31DIOQ+8,C' ' REMOVE THE TRASH
SPACE
MVC P31DIOSC,=CL8'IOSB-AD' CONSTANT TO OUTPUT AREA
UNPK P31DIOS(9),TTE301IO(5) ADDRESS OF IOSB TO OUTPUT AREA
TR P31DIOS,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P31DIOS+8,C' ' REMOVE THE TRASH
B PCP4MAT TRANSCRIBE CSCH DATA
SPACE
DROP R1 FORGET PPG301D
EJECT
*****
* FORMAT A PORTION OF AN ENTRY FOR A HALT SUBCHANNEL TTE *
*****
SPACE
PPGHSCH DS 0H
LA R1,PRINTOUT
USING PPG201D,R1 ESTABLISH PPG201D ADDRESSABILITY
SPACE
BAS R5,PPCPFSCH FORMAT COMMON SUBCHANNEL SEGMENT
SPACE
MVC P21DIOQC,=CL7'IOQ-ADR' CONSTANT TO OUTPUT AREA
UNPK P21DIOQ(5),TTE201IQ(3) ADDRESS OF IOQ TO OUTPUT AREA
TR P21DIOQ,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P21DIOQ+4,C' ' REMOVE THE TRASH
SPACE
MVC P21DIOSC,=CL8'IOSB-AD' CONSTANT TO OUTPUT AREA
UNPK P21DIOS(9),TTE201IO(5) ADDRESS OF IOSB TO OUTPUT AREA
TR P21DIOS,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P21DIOS+8,C' ' REMOVE THE TRASH
B PCP4MAT TRANSCRIBE HSCH DATA
SPACE
DROP R1 FORGET PPG201D
EJECT
*****
* FORMAT A PORTION OF AN ENTRY FOR A MODIFY SUBCHANNEL TTE *
*****
SPACE
PPGMSCH DS 0H
LA R1,PRINTOUT
USING PPG101D,R1 ESTABLISH PPG101D ADDRESSABILITY
SPACE
BAS R5,PPCPFSCH FORMAT COMMON SUBCHANNEL SEGMENT
SPACE
MVC P11DORBC,=CL8'ORB-FLAG' CONSTANT TO OUTPUT AREA
UNPK P11DORB(5),TTE101O2(3) ORB FLAGS FROM SCHIB TO OUT AREA
TR P11DORB,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P11DORB+4,C' ' REMOVE THE TRASH
SPACE
MVC P11DLPMC,=CL10'LPATH-MASK' CONSTANT TO OUTPUT AREA

```



```

UNPK P11DLPM(3),TTE101LM(2) LOGICAL PATH MASK FROM SCHIB
TR P11DLPM,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P11DLPM+2,C' ' REMOVE THE TRASH
SPACE
MVC P11DPOMC,=CL10'OPATH-MASK' CONSTANT TO OUTPUT AREA
UNPK P11DPOM(3),TTE101LM(2) PATH OPERATIONAL MASK FROM SCHIB
TR P11DPOM,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P11DPOM+2,C' ' REMOVE THE TRASH
SPACE
MVC P11DIOSC,=CL8'IOSB-AD' CONSTANT TO OUTPUT AREA
UNPK P11DIOS(9),TTE101IO(5) ADDRESS OF IOSB TO OUTPUT AREA
TR P11DIOS,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P11DIOS+8,C' ' REMOVE THE TRASH
B PCP4MAT TRANSCRIBE MSCH DATA
SPACE
DROP R1 FORGET PPG101D
EJECT

```

```

*****
*          FORMAT A PORTION OF AN ENTRY FOR A START SUBCHANNEL TTE          *
*****

```

```

SPACE
PPGSSCH DS 0H
LA R1,PRINTOUT
USING PPG001D,R1 ESTABLISH PPG001D ADDRESSABILITY
SPACE
BAS R5,PPCPFSCH FORMAT COMMON SUBCHANNEL SEGMENT
SPACE
MVC P01DOR2C,=CL6'ORB-W2' CONSTANT TO OUTPUT AREA
UNPK P01DOR2(9),TTE001O2(5) ORB WORD TWO TO OUTPUT AREA
TR P01DOR2,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P01DOR2+8,C' ' REMOVE THE TRASH
SPACE
MVC P01DOR3C,=CL6'ORB-W3' CONSTANT TO OUTPUT AREA
UNPK P01DOR3(9),TTE001O3(5) ORB WORD THREE TO OUTPUT AREA
TR P01DOR3,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P01DOR3+8,C' ' REMOVE THE TRASH
SPACE
MVC P01DCAPC,=CL6'UCB-AD' CONSTANT TO OUTPUT AREA
UNPK P01DCAP(9),TTE001CU(5) CAPTURE UCB ADDRESS TO OUT AREA
TR P01DCAP,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P01DCAP+8,C' ' REMOVE THE TRASH
SPACE
MVC P01DIOSC,=CL8'IOSB-AD' CONSTANT TO OUTPUT AREA
UNPK P01DIOS(9),TTE001IO(5) ADDRESS OF IOSB TO OUTPUT AREA
TR P01DIOS,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P01DIOS+8,C' ' REMOVE THE TRASH
B PCP4MAT TRANSCRIBE SSCH DATA
SPACE
DROP R1 FORGET PPG001D
EJECT
*          FORMAT A PORTION OF AN ENTRY FOR A PC OR BRANCH ENTERED          *

```

```

*          SYSTEM SERVICE TRACE TABLE ENTRY          *
SPACE
PPGSSRV DS   ØH
SPACE
CLI  PPGTNAME,C' '      TEST IF TASK NAME IS PRESENT
BE   *+8                BRANCH IF NOT
NI   PPGSW,255-2        ALLOW COMPUTATION OF OFFSET IN PGM
SPACE
LA   R1,PRINTOUT
USING PPG2Ø5D,R1        ESTABLISH PPG2Ø5D ADDRESSABILITY
SPACE
MVC  P25DTCBC,=CL4'TCB' CONSTANT TO OUTPUT AREA
UNPK P25DTCB(9),TTE2Ø5TB(5) ADDRESS OF TCB TO OUTPUT AREA
TR   P25DTCB,PATRANS-24Ø CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  P25DTCB+8,C' '     REMOVE THE TRASH
SPACE
MVC  P25DSVRC,=CL9'SERV-ID' CONSTANT TO OUTPUT AREA
UNPK P25DSVRN(5),TTE2Ø5SI(3) SERVER ID NO. TO OUTPUT AREA
TR   P25DSVRN,PATRANS-24Ø CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI  P25DSVRN+4,C' '    REMOVE THE TRASH
SPACE
MVC  P25DHAC,=CL5'HASID' CONSTANT TO OUTPUT AREA
UNPK P25DHID(5),TTE2Ø5HA(3) HOME ASID TO OUTPUT AREA
TR   P25DHID,PATRANS-24Ø CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  P25DHID+4,C' '    REMOVE THE TRASH
SPACE
MVC  P25RETC,=CL8'RET-ADR' CONSTANT TO OUTPUT AREA
UNPK P25RETA(9),TTE2Ø5CI(5) CALLER'S RETURN ADDRESS TO OUTPUT
TR   P25RETA,PATRANS-24Ø CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  P25RETA+8,C' '    REMOVE THE TRASH
SPACE
MVC  P25DW1C,=CL6'WORD1' CONSTANT TO OUTPUT AREA
UNPK P25DWD1(9),TTE2Ø5U1(5) 1ST WORD OF INFO UNIQUE TO SRV-ID
TR   P25DWD1,PATRANS-24Ø CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  P25DWD1+8,C' '    REMOVE THE TRASH
SPACE
MVC  P25DW2C,=CL6'WORD2' CONSTANT TO OUTPUT AREA
UNPK P25DWD2(9),TTE2Ø5U2(5) 2ND WORD OF INFO UNIQUE TO SRV-ID
TR   P25DWD2,PATRANS-24Ø CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  P25DWD2+8,C' '    REMOVE THE TRASH
SPACE
MVC  P25DW3C,=CL6'WORD3' CONSTANT TO OUTPUT AREA
UNPK P25DWD3(9),TTE2Ø5U3(5) 3RD WORD OF INFO UNIQUE TO SRV-ID
TR   P25DWD3,PATRANS-24Ø CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  P25DWD3+8,C' '    REMOVE THE TRASH
SPACE
MVC  P25DW4C,=CL6'WORD4' CONSTANT TO OUTPUT AREA
UNPK P25DWD4(9),TTE2Ø5U4(5) 4TH WORD OF INFO UNIQUE TO SRV-ID
TR   P25DWD4,PATRANS-24Ø CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  P25DWD4+8,C' '    REMOVE THE TRASH
EJECT
*****

```

```

*          TRANSLATE A SERVICE IDENTIFIER INTO THE NAME OF ITS MACRO          *
*****
SPACE
L      R15,=A(PPGSSID)      FETCH ADDRESS OF MACRO TABLE
LA     R14,PPGSSID#        SET LOOP COUNT
PGLOOPV CLC  TTE205SI,0(R15)  TEST FOR A MATCHING SERVICE IDENT.
BE     PGLSETM              BRANCH WHEN ONE IS FOUND
BL     PGLSETU              BRANCH WHEN ENTRY IS LESS THAN TABLE
LA     R15,PPGSSIDL(R15)   ELSE POINT TO NEXT ENTRY
BCT    R14,PGLOOPV         AND CONTINUE SEARCH
PGLSETU MVC  P25DSVRM,=CL8'UNKNOWN' SERVICE IDENT. IS A NEW ONE ON ME
B      PGLWRITE            CONTINUE...
SPACE
PGLSETM MVC  P25DSVRM,2(R15) CONVERT SERVICE IDENT. TO MACRO NAME
SPACE
PGLWRITE BAS  R5,PPGPUTLS    TRANSCRIBE SSRV DATA
MVC     PRINTOUT,CLEAR      BLANK OUTPUT AREA
SPACE
MVC     PATGILL,TTE205CI    STOW RET ADDRESS OF SERVICE INVOKER
CLC     PATGILL,PPGF0      TEST IF ADDRESS IS AVAILABLE
BNE     PPG4MAT            BRANCH IF SO
SPACE
OI      PPGSW,2             ELSE DO NOT COMPUTE OFFSET
B       PPG4MAT            FORMAT SSRV ENTRY
SPACE
DROP   R1                  FORGET PPG205D
EJECT
*****
*          FORMAT A PORTION OF AN ENTRY FOR A WAIT DISPATCH TTE          *
*****
SPACE
PPGWAIT DS   0H
LA     R1,PRINTOUT
USING  PPGF0FD,R1          ESTABLISH PPGF0FD ADDRESSABILITY
SPACE
MVC    PF0DTCBC,=CL4'TCB'  CONSTANT TO OUTPUT AREA
UNPK   PF0DTCB(9),TTEF0FTB(5) ADDRESS OF TCB TO OUTPUT AREA
TR     PF0DTCB,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI    PF0DTCB+8,C' '      REMOVE THE TRASH
SPACE
MVC    PF0DHAC,=CL5'HASID' CONSTANT TO OUTPUT AREA
UNPK   PF0DHID(5),TTEF0FHA(3) HOME ASID TO OUTPUT AREA
TR     PF0DHID,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI    PF0DHID+4,C' '      REMOVE THE TRASH
SPACE
MVC    PF0DTDC,=CL10'TOD-CLOCK' CONSTANT TO OUTPUT AREA
UNPK   PF0DTOD(13),TTETOD(7) TOD CLOCK TO OUTPUT AREA
TR     PF0DTOD,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI    PF0DTOD+12,C' '     REMOVE THE TRASH
B      PCP4MAT              TRANSCRIBE WAIT DATA
SPACE
DROP   R1                  FORGET PPGF0FD

```

```

EJECT
*****
*          FORMAT A BRANCH IN SUBSPACE GROUP SYSTEM TRACE ENTRY          *
*****
SPACE
PPGBSG DS    ØH
        LA    R1,PRINTOUT
        USING PPGBSGD,R1          ESTABLISH PPGBSGD ADDRESSABILITY
SPACE
MVC    PBSADARC,=CL11'TARGET-ADR' CONSTANT TO OUTPUT AREA
UNPK   PBSADADR(9),TTEBSADR(5) BSG'S TARGET ADDRESS TO OUTPUT
TR     PBSADADR,PATRANS-24Ø CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI    PBSADADR+8,C' '          REMOVE THE TRASH
SPACE
MVC    PBSDALEC,=CL12'ACCESS-LIST' CONSTANT TO OUTPUT AREA
MVC    PBSDALE,=CL11'NOT/APPLIC' ASSUME NOT APPLICABLE
CLC    TTEBSALT,PPGFØ          TEST ASSUMPTION
BE     PPGBASR5                BRANCH IF VALID
CLC    TTEBSALT,PPHONE+1      TEST IT TWICE
BE     PPGBASR5                BRANCH IF VALID
SPACE
MVC    PPGHOLD(3),TTEBSALT MOVE ALET TO WORK AREA
NI     PPGHOLD,255-8          REMOVE HIGH-ORDER BIT
MVC    PBSDASNC,=CL5'SEQ#' CONSTANT TO OUTPUT AREA
UNPK   PBSDASN(3),PPGHOLD(2) ACCESS-LIST SEQUENCE NUMBER TO OUT
TR     PBSDASN,PATRANS-24Ø CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI    PBSDASN+2,C' '          REMOVE THE TRASH
SPACE
MVC    PBSDNUMC,=CL7'NUMBER' CONSTANT TO OUTPUT AREA
UNPK   PBSDNUM(5),PPGHOLD+1(3) ACCESS-LIST ENTRY NUMBER TO OUT
TR     PBSDNUM,PATRANS-24Ø CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI    PBSDNUM+4,C' '          REMOVE THE TRASH
SPACE
MVC    PBSDALE,=CL11'PRIM-SPACE' ASSUME PRIMARY-SPACE ACCES LST
TM     TTEBSALT,TTEBSAP      TEST ASSUMPTION
BO     PPGBASR5                BRANCH IF VALID
MVC    PBSDALE,=CL11'DISP-UNIT' SHOW DISPATCHABLE-UNIT ACS LIST
B      PPGBASR5                TRANSCRIBE DESCRIPTION OF DATA
SPACE
DROP   R1                      FORGET PPGBSGD
EJECT
*****
*          ESTABLISH THE LENGTH OF A USR TTE, THEN FORMAT IT          *
*****
SPACE
PPGUSR DS    ØH
        CLI   PPGTNAME,C' '      TEST IF TASK NAME IS PRESENT
        BE    PPGUSRLR          BRANCH IF NOT
SPACE
CLC    PPGTASID,TTEØ7FHA      TEST IF CORRECT TASK
BNE    PPGUSRNT              BRANCH IF NOT
SPACE

```

```

PPGUSRLR MVC PPGUSER#(1),TTEXPSID SET USER SUB-TYPE IDENTIFICATION
OI PPGUSER#,240 TURN ON ALL ZONE BITS
TR PPGUSER#(1),PATRANS-240 CONVERT IT TO SOMETHING READABLE
SPACE
LA R6,PPGUSERX POINT TO DUMMY USER ENTRY
SR R5,R5 CLEAR LENGTH REGISTER
IC R5,TTETYPE FETCH TYPE
N R5,PPG000F ISOLATE NUMBER OF ADDITIONAL WORDS
SH R5,=H'4' REDUCE BY STANDARD SIZE
MH R5,=H'4' COMPUTE ADDITIONAL LENGTH
LA R5,32(R5) ADD STANDARD SIZE FOR USER ENTRY
OI PPGSW,2 DO NOT ATTEMPT COMPUTATION OF OFFSET
B PPGNANC FORMAT USR ENTRY
SPACE
PPGUSRNT DS 0H PROVIDE TARGET FOR BRANCH INST.
SR R5,R5 CLEAR LENGTH REGISTER
IC R5,TTETYPE FETCH TYPE
N R5,PPG000F ISOLATE NUMBER OF ADDITIONAL WORDS
SH R5,=H'4' REDUCE BY STANDARD SIZE
MH R5,=H'4' COMPUTE ADDITIONAL LENGTH
LA R5,32(R5) ADD STANDARD SIZE FOR USER ENTRY
LR R9,R5 SET LENGTH OF CURRENT ENTRY
B PPGSKIP GO AND PROCESS NEXT TRACE TABLE NTRY
EJECT
* TRACE TABLE ENTRIES FOR WHICH NO PARTIAL FORMATTING OF *
* DATA IS DONE *
SPACE
PPGALTR DS 0H
OI PPGSW,2 DO NOT ATTEMPT COMPUTATION OF OFFSET
B PPG4MAT FORMAT ALTR ENTRY
SPACE
PPGBR DS 0H
B PPG4MAT FORMAT AN ENTRY
SPACE
PPGEMS DS 0H
MVC PATGILL,TTE103P2 STOW RIGHT HALF OF INTERRUPTED PSW
B PPG4MAT FORMAT EMS ENTRY
SPACE
PPGMCH DS 0H
MVC PATGILL,TTE013P2 STOW RIGHT HALF OF INTERRUPTED PSW
B PPG4MAT FORMAT MCH ENTRY
SPACE
PPGRST DS 0H
MVC PATGILL,TTE015P2 STOW RIGHT HALF OF INTERRUPTED PSW
B PPG4MAT FORMAT RST ENTRY
SPACE
PPGSPER DS 0H
MVC PATGILL,TTE009P2 STOW RIGHT HALF OF INTERRUPTED PSW
B PPG4MAT FORMAT SPER ENTRY
SPACE
PPGSS DS 0H
MVC PATGILL,TTE203P2 STOW RIGHT HALF OF INTERRUPTED PSW

```

```

      B      PPG4MAT          FORMAT SS ENTRY
      SPACE
PPGSSRB DS      ØH
      MVC    PATGILL,TTE2ØFP2  STOW RIGHT HALF OF INTERRUPTED PSW
      B      PPG4MAT          FORMAT SSRB ENTRY
      SPACE
PPGSUSP DS      ØH
      MVC    PATGILL,TTEØ19RT  STOW RET ADDRESS OF SERVICE INVOKER
      B      PPG4MAT          FORMAT SUSP ENTRY
      SPACE
PPGSVCE DS      ØH
      MVC    PATGILL,TTEFØ5P2  STOW RIGHT HALF OF INTERRUPTED PSW
      B      PPG4MAT          FORMAT SVCE ENTRY
      EJECT
*      FORMAT A PORTION OF AN ENTRY FOR SVC INTERRUPT AND          *
*      AND SVC RETURN TRACE TABLE ENTRIES                        *
      SPACE
PPGSVC  DS      ØH
      SPACE
      CLI    PPGTNAME,C' '    TEST IF TASK NAME IS PRESENT
      BE     *+8              BRANCH IF NOT
      NI     PPGSW,255-2      ALLOW COMPUTATION OF OFFSET IN PGM
      SPACE
      BAS    R2,PPGPRNTS     TRANSCRIBE SVC DATA
      B      PPG4MAT          FORMAT SVC ENTRY
      SPACE
PPGSVCR DS      ØH
      SPACE
      CLI    PPGTNAME,C' '    TEST IF TASK NAME IS PRESENT
      BE     *+8              BRANCH IF NOT
      NI     PPGSW,255-2      ALLOW COMPUTATION OF OFFSET IN PGM
      SPACE
      BAS    R2,PPGPRNTS     TRANSCRIBE SVCR DATA
      MVC    PATGILL,TTE1Ø5P2 STOW RIGHT HALF OF INTERRUPTED PSW
      B      PPG4MAT          FORMAT SVCR ENTRY
      SPACE 3
*****
*      TRANSCRIBE A LINE OF DATA OF BEHALF OF EACH TRACE TABLE ENTRY *
*****
      SPACE
PPGPUTLS DS      ØH
      PUT    PATOUT,CLEAR     TRANSCRIBE DATA
      BR     R5              RETURN TO CALLER
      EJECT
*****
*      PROVIDE READILY UNDERSTANDABLE PROSE FOR AN SVC ENTRY      *
*****
      SPACE
PPGPRNTS LA     R1,PRINTOUT
      USING PPGØØ5D,R1       ESTABLISH PPGØØ5D ADDRESSABILITY
      SPACE

```

```

MVC P05DTCBC,=CL4'TCB' CONSTANT TO OUTPUT AREA
UNPK P05DTCB(9),TTE005TB(5) ADDRESS OF TCB TO OUTPUT AREA
TR P05DTCB,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI P05DTCB+8,C' ' REMOVE THE TRASH
SPACE
MVC P05DSVCC,=CL5'SVC #' CONSTANT TO OUTPUT AREA
UNPK P05DSVCN(5),TTE005SN(3) SVC NUMBER TO OUTPUT AREA
TR P05DSVCN,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P05DSVCN+4,C' ' REMOVE THE TRASH
SPACE
MVC P05DHAC,=CL5'HASID' CONSTANT TO OUTPUT AREA
UNPK P05DHID(5),TTE005HA(3) HOME ASID TO OUTPUT AREA
TR P05DHID,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI P05DHID+4,C' ' REMOVE THE TRASH
SPACE
MVC P05DP1C,=CL5'PSW1' CONSTANT TO OUTPUT AREA
UNPK P05DPSW1(9),TTE005P1(5) LEFT HALF OF PSW OUTPUT AREA
TR P05DPSW1,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P05DPSW1+8,C' ' REMOVE THE TRASH
SPACE
MVC P05DP2C,=CL5'PSW2' CONSTANT TO OUTPUT AREA
UNPK P05DPSW2(9),TTE005P2(5) RIGHT HALF OF PSW OUTPUT AREA
TR P05DPSW2,PATRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI P05DPSW2+8,C' ' REMOVE THE TRASH
SPACE
MVC P05DG0C,=CL5'GPR0' CONSTANT TO OUTPUT AREA
UNPK P05DG0(9),TTE005G0(5) CONTENTS OF GPR0 OUTPUT AREA
TR P05DG0,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI P05DG0+8,C' ' REMOVE THE TRASH
SPACE
MVC P05DG1C,=CL5'GPR1' CONSTANT TO OUTPUT AREA
UNPK P05DG1(9),TTE005G1(5) CONTENTS OF GPR1 OUTPUT AREA
TR P05DG1,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI P05DG1+8,C' ' REMOVE THE TRASH
SPACE
MVC P05DGFC,=CL5'GPRF' CONSTANT TO OUTPUT AREA
UNPK P05DGF(9),TTE005GF(5) CONTENTS OF GPRF OUTPUT AREA
TR P05DGF,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI P05DGF+8,C' ' REMOVE THE TRASH
EJECT

```

```

*****
* LOCATE NAME OF TASK WITHIN AN ADDRESS SPACE *
*****

```

```

SPACE 1
MVC P05DJNAM,=CL8'UNKNOWN' JUST IN CASE...
SPACE 1
L R5,CVTPTR ADDR OF COMMUNICATIONS VECTOR TABLE
USING CVT,R5 ESTABLISH CVT ADDRESSABILITY
SPACE 1
L R15,CVTASVT FETCH ADDRESS OF ASVT
USING ASVT,R15 ESTABLISH ASVT ADDRESSABILITY
DROP R5 FORGET CVT

```

```

SPACE 1
LH R8,TTE005HA      FETCH ASID ASSOCIATED WITH ENTRY
C R8,ASVTMAXU      TEST IF IT EXCEEDS LIMITS
BH PPGOUT          BRANCH IF SO
SPACE 1
SLL R8,2           MULTIPLY ASID BY FOUR
L R14,ASVTFRST(R8) ADDRESS OF ASCB
USING ASCB,R14     ESTABLISH ASCB ADDRESSABILITY
SPACE 1
TM ASVTENTY,ASVTAVAL TEST IF ASCB IS IN USE
BO PPGOUT          BRANCH IF NOT
SPACE 1
SR R5,R5           CLEAR THE TRASH
ICM R5,7,ASCBJNI+1 POINTER TO INITIATED JOBNAME
BZ PATJBNS         IF NONE, TRY STARTED TASK
MVC P05DJNAM,0(R5) STOW NAME OF JOB IN OUTPUT AREA
B PPGSVCN          CONTINUE...
SPACE 1
PATJBNS EQU *
SPACE 1
ICM R5,7,ASCBJNI+1 POINTER TO START/MOUNT/LOGON TASK
BZ PPGOUT          FORGET IT
MVC P05DJNAM,0(R5) STOW NAME OF JOB IN DATA AREA
EJECT
*****
* TRANSLATE AN SVC NUMBER INTO THE NAME OF ITS MACRO *
*****
SPACE 1
PPGSVCN DS 0H
SPACE
LH R5,TTE005SN      RETRIEVE NUMBER OF SVC
C R5,PPGSVCHI      TEST IF WITHIN RANGE
BH PPGOUT          BRANCH IF NOT
MH R5,=H'8'        COMPUTE OFFSET INTO TABLE
LA R5,PPGSVCTB(R5) POINT TO ENTRY
MVC P05DSVNM,0(R5) MOVE NAME OF SVC TO OUTPUT AREA
SPACE
CLC P05DSVNM(3),GETMAIN TEST IF GETMAIN/FREEMAIN
BNE PPGOUT
MVC P05DSVNM,GETMAIN ASSUME GETMAIN
CLC TTE005G1,PPGF0 TEST IF PERFECT ASSUMPTION
BE PPGOUT          BRANCH IF SO
TM TTE005P1+2,X'20' TEST IF CC = 2
BNO PPGOUT          BRANCH IF NOT
MVC P05DSVNM,FREEMAIN OTHERWISE SHOW FREEMAIN
SPACE
PPGOUT DS 0H
SPACE
DROP R1,R14,R15    FORGET ASCB, CVT, AND ASVT
SPACE
BAS R5,PPGPUTLS
MVC PRINTOUT,CLEAR BLANK OUTPUT AREA

```



```

SPACE
MVC  PATGILL,TTE005P2    STOW RIGHT HALF OF INTERRUPTED PSW
BR   R2                  RETURN TO CALLER
EJECT
*   FORMAT COMMON SECTION OF SUBCHANNEL TRACE TABLE ENTRIES      *
*   NOTE: A MODIFY SUBCHANNEL TTE CONTAINS NO DRIVER ID EVEN      *
*   THOUGH IT IS FORMATTED AS SUCH, IT'S JUST A LITTLE WHITE LIE *
SPACE 1
PPCPFSCH LA  R1,PRINTOUT    POINT TO COMMON OUTPUT AREA
        USING PPG301D,R1    ESTABLISH PPG301D ADDRESSABILITY
SPACE
MVC  P31DTCBC,=CL4'TCB'   CONSTANT TO OUTPUT AREA
UNPK P31DTCB(9),TTE301TB(5) ADDRESS OF TCB TO OUTPUT AREA
TR   P31DTCB,PATTRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  P31DTCB+8,C' '      REMOVE THE TRASH
SPACE
MVC  P31DIAC,=CL9'IOS-ASID' CONSTANT TO OUTPUT AREA
UNPK P31DIID(5),TTE301AD(3) IOS' ASID TO OUTPUT AREA
TR   P31DIID,PATTRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  P31DIID+4,C' '      REMOVE THE TRASH
SPACE
MVC  P31DHAC,=CL5'HASID'  CONSTANT TO OUTPUT AREA
UNPK P31DHID(5),TTE301HA(3) HOME ASID TO OUTPUT AREA
TR   P31DHID,PATTRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  P31DHID+4,C' '      REMOVE THE TRASH
SPACE
MVC  P31DCCC,=CL9'COND-CODE' CONSTANT TO OUTPUT AREA
UNPK P31DCC(3),TTE301CC(2) CONDITION CODE TO OUTPUT AREA
TR   P31DCC,PATTRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  P31DCC+2,C' '      REMOVE THE TRASH
SPACE
MVC  P31DVRC,=CL9'DVR-ID'  CONSTANT TO OUTPUT AREA
UNPK P31DVR(3),TTE301DI(2) DRIVER IDENT TO OUTPUT AREA
TR   P31DVR,PATTRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI  P31DVR+2,C' '      REMOVE THE TRASH
SPACE
MVC  P31DADRC,=CL6'UCB-AD' CONSTANT TO OUTPUT AREA
UNPK P31DADR(9),TTE301UB(5) ADDRESS OF UCB TO OUTPUT AREA
TR   P31DADR,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI  P31DADR+8,C' '      REMOVE THE TRASH
SPACE
MVC  P31DNUMC,=CL4'DEV#'   CONSTANT TO OUTPUT AREA
UNPK P31DNUM(5),TTE301DN(3) DEVICE NUMBER TO OUTPUT AREA
TR   P31DNUM,PATTRANS-240 CONVERT VALUE TO EBCDIC-LEGIBILITY
MVI  P31DNUM+4,C' '      REMOVE THE TRASH
SPACE
OI   PPGSW,2             DO NOT ATTEMPT COMPUTATION OF OFFSET
BR   R5                  RETURN TO CALLER
SPACE
DROP R1                  FORGET PPG301D
EJECT
*   FORMAT ADDRESS OF TCB, ASID'S, AND PSW                          *

```

```

SPACE 1
PPCPFTCB LA R1,PRINTOUT POINT TO COMMON OUTPUT AREA
USING PPG007D,R1 ESTABLISH PPG007D ADDRESSABILITY
SPACE
MVC P07DTCBC,=CL4'TCB' CONSTANT TO OUTPUT AREA
UNPK P07DTCB(9),TTE007TB(5) ADDRESS OF TCB TO OUTPUT AREA
TR P07DTCB,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI P07DTCB+8,C' ' REMOVE THE TRASH
SPACE
MVC P07DHAC,=CL5'HASID' CONSTANT TO OUTPUT AREA
UNPK P07DHID(5),TTE007HA(3) HOME ASID TO OUTPUT AREA
TR P07DHID,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI P07DHID+4,C' ' REMOVE THE TRASH
SPACE
MVC P07DPAC,=CL5'PASID' CONSTANT TO OUTPUT AREA
UNPK P07DPID(5),TTE007PA(3) PRIMARY ASID TO OUTPUT AREA
TR P07DPID,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI P07DPID+4,C' ' REMOVE THE TRASH
SPACE
MVC P07DSAC,=CL5'SASID' CONSTANT TO OUTPUT AREA
UNPK P07DSID(5),TTE007SA(3) SECONDARY ASID TO OUTPUT AREA
TR P07DSID,PATRANS-240 CONVERT VALUE TO EBCDIC - LEGIBILITY
MVI P07DSID+4,C' ' REMOVE THE TRASH
SPACE
MVC P07DP1C,=CL5'PSW1' CONSTANT TO OUTPUT AREA
UNPK P07DPSW1(9),TTE007P1(5) LEFT HALF OF PSW OUTPUT AREA
TR P07DPSW1,PATRANS-240 CONVERT VALUE TO EBCDIC- LEGIBILITY
MVI P07DPSW1+8,C' ' REMOVE THE TRASH
SPACE
MVC P07DP2C,=CL5'PSW2' CONSTANT TO OUTPUT AREA
UNPK P07DPSW2(9),TTE007P2(5) RIGHT HALF OF PSW OUTPUT AREA
TR P07DPSW2,PATRANS-240 CONVERT VALUE TO EBCDIC- LEGIBILITY
MVI P07DPSW2+8,C' ' REMOVE THE TRASH
SPACE
BR R5 RETURN TO CALLER
SPACE
DROP R1 FORGET PPG007D
EJECT
* CONSTANTS AND OTHER JUNK *
SPACE 1
PPGSAFE DS D
PATDOUBL DS D
PPGSAVE DS 18F
PATH DS F
PPGHOLDC DS F
PPGHOLDA DS F
PPGHOLD DS F
PPGBUFNO DS F
PATGILL DS F
PPGBUFS DS F
PPGTASCB DS F
PPGSASID DS F

```

```

PCAR      DS      F
PPGTASID DS      H
PPGSW     DC      X'00'
PPGF0     DC      F'0'
PPHONE    DC      F'1'
PPCEND    DC      A(4090)
PPCURSOR  DC      F'0'
PPG000F   DC      XL4'0000000F'
PAT0FFF   DC      XL4'00FFFFFF'
PAT7FFF   DC      XL4'7FFFFFFF'
PATCR12   DC      XL4'7FFFFFFC'
PPGNTBVT  DC      H'0'
PPGNTBUF  DC      H'0'
PPGTRBUF  DC      H'0'
PPG4096   DC      H'4096'
          SPACE
PPGMVTNM  MVC      PPGTNAME(*-*),2(R2) ***** EXECUTE ONLY *****
MEBCDIC   MVC      EBCDIC(0),0(R7)
PATTPART  TR      0(0,R10),PATRANS-240
PATRANS   DC      C'0123456789ABCDEF'
          SPACE
          DROP   R7                      FORGET TTE
          SPACE
PATOUT    DCB      DDNAME=SYSPRINT,LRECL=133,BLKSIZE=133,RECFM=FA,DSORG=PS,L
          MACRF=PM,DCBE=PATEDCB
PATEDCB   DCBE
          SPACE
PPGSTTID  DC      CL4'STT'
PPGJNAME  DC      CL8'TRACE'
PPGTNAME  DC      CL8' '
          SPACE
CLEAR     DC      C' '
PRINTOUT  DS      CL133
EBCDIC    EQU      PRINTOUT+85
          SPACE
PPGHOLDL  DS      CL40
          EJECT
          DS      0F
*          BRANCH ENTRIES
PPGBREN   DC      AL1(00),AL2(0000),X'04',A(PPGBR),CL4'BR'
PPGSIZE   EQU      *-PPGBREN
          SPACE 2
*          ADDRESS SPACE ENTRIES
PPGADREN  DC      AL1(TTETPC),XL2'0',AL1(08),A(PPGPC),CL4'PC'
          DC      AL1(TTETPR),XL2'0',AL1(12),A(PPGPR),CL4'PR'
          DC      AL1(TTETPT),XL2'0',AL1(08),A(PPGPT),CL4'PT'
          DC      AL1(TTETBSG),XL2'0',AL1(08),A(PPGBSG),CL4'BSG'
          DC      AL1(TTETSSAR),AL2(0),AL1(04),A(PPGSSAR),CL4'SSAR'
PPGADRNO  EQU      ((*-PPGADREN)/PPGSIZE)
          SPACE 2
PPGUSERX  DC      XL1'00',AL2(TTETUSR0),AL1(00),A(PPGUSR),CL4'USR '
PPGUSER#  EQU      *-1

```

```

EJECT
* EXPLICIT ENTRIES
PPGEXPEN DC XL1'7A',AL2(TTETSPER),AL1(56),A(PPGSPER),CL4'SPER'
DC XL1'7A',AL2(TTETRCVY),AL1(52),A(PPGRCVY),CL4'RCVY'
DC XL1'7A',AL2(TTETEMS),AL1(56),A(PPGEMS),CL4'EMS'
DC XL1'7A',AL2(TTETSVCE),AL1(56),A(PPGSVCE),CL4'SVCE'
DC XL1'7A',AL2(TTETCLKC),AL1(56),A(PPGCLKC),CL4'CCXI'
DC XL1'7B',AL2(TTETIO),AL1(60),A(PPGIO),CL4'I/O'
DC XL1'7B',AL2(TTETRCVY),AL1(56),A(PPGRCVY),CL4'RCVY'
DC XL1'7B',AL2(TTETRST),AL1(60),A(PPGRST),CL4'RST'
DC XL1'7B',AL2(TTETEMS),AL1(60),A(PPGEMS),CL4'EMSB'
DC XL1'7C',AL2(TTETRCVY),AL1(60),A(PPGRCVY),CL4'RCVY'
DC XL1'7C',AL2(TTETSS),AL1(64),A(PPGSS),CL4'SS'
DC XL1'7D',AL2(TTETIO),AL1(68),A(PPGIO),CL4'I-0'
DC XL1'7D',AL2(TTETRCVY),AL1(64),A(PPGRCVY),CL4'RCVY'
DC XL1'71',AL2(TTETWAIT),AL1(20),A(PPGWAIT),CL4'WAIT'
DC XL1'74',AL2(TTETRCVY),AL1(28),A(PPGRCVY),CL4'RCVY'
DC XL1'74',AL2(TTETRSCHE),AL1(32),A(PPGRSCH),CL4'RSCH'
DC XL1'75',AL2(TTETRCVY),AL1(32),A(PPGRCVY),CL4'RCVY'
DC XL1'75',AL2(TTETRSCHE),AL1(32),A(PPGRSCH),CL4'RSC5'
DC XL1'76',AL2(TTETSSCH),AL1(40),A(PPGSSCH),CL4'SSCH'
DC XL1'76',AL2(TTETSVC),AL1(40),A(PPGSVC),CL4'SVC'
DC XL1'76',AL2(TTETALTR),AL1(40),A(PPGALTR),CL4'ALTR'
DC XL1'76',AL2(TTETRCVY),AL1(36),A(PPGRCVY),CL4'RCVY'
DC XL1'76',AL2(TTETSRB),AL1(40),A(PPGSRB),CL4'SRB'
DC XL1'76',AL2(TTETMSCH),AL1(40),A(PPGMSCH),CL4'MSCH'
DC XL1'76',AL2(TTETSVCR),AL1(40),A(PPGSVCR),CL4'SVCR'
DC XL1'76',AL2(TTETHSCH),AL1(40),A(PPGHSCH),CL4'HSCH'
DC XL1'76',AL2(TTETSSRV),AL1(40),A(PPGSSRV),CL4'SSRV'
DC XL1'76',AL2(TTETCSCH),AL1(40),A(PPGCSCH),CL4'CSCH'
DC XL1'77',AL2(TTETEXT),AL1(44),A(PPGEXT),CL4'EXT'
DC XL1'77',AL2(TTETSRB),AL1(44),A(PPGSRB),CL4'SRB7'
DC XL1'77',AL2(TTETRCVY),AL1(40),A(PPGRCVY),CL4'RCVY'
DC XL1'77',AL2(TTETSSRB),AL1(44),A(PPGSSRB),CL4'SSRB'
DC XL1'78',AL2(TTETEXT),AL1(48),A(PPGEXT),CL4'GEXI'
DC XL1'78',AL2(TTETSSRB),AL1(48),A(PPGSSRB),CL4'SSBD'
DC XL1'78',AL2(TTETMSCH),AL1(48),A(PPGTSCH),CL4'TSCH'
DC XL1'78',AL2(TTETPGM),AL1(48),A(PPGPGM),CL4'PGM'
DC XL1'78',AL2(TTETRCVY),AL1(44),A(PPGRCVY),CL4'RCVY'
DC XL1'78',AL2(TTETACR),AL1(48),A(PPGACR),CL4'ACR'
DC XL1'78',AL2(TTETSUSP),AL1(48),A(PPGSUSP),CL4'SUSP'
DC XL1'78',AL2(TTETCALL),AL1(48),A(PPGCALL),CL4'CALL'
DC XL1'79',AL2(TTETPGM),AL1(52),A(PPGPGM),CL4'PGMI'
DC XL1'79',AL2(TTETSUSP),AL1(52),A(PPGSUSP),CL4'SULS'
DC XL1'79',AL2(TTETCALL),AL1(52),A(PPGCALL),CL4'XCXI'
DC XL1'79',AL2(TTETDSP),AL1(52),A(PPGDSP),CL4'DSP'
DC XL1'79',AL2(TTETRCVY),AL1(48),A(PPGRCVY),CL4'RCVY'
DC XL1'79',AL2(TTETMCH),AL1(52),A(PPGMCH),CL4'MCH'
DC XL1'79',AL2(TTETCLKC),AL1(52),A(PPGCLKC),CL4'CLKC'
PPGEXPNO EQU ((*-PPGEXPEN)/PPGSIZE)
EJECT
LTOrg

```

```

EJECT
PPGSVCHI DC F'144'
PPGSVCTB DC CL8'EXCP/DAP'
DC CL8'WAIT/R/O'
DC CL8'POST '
DC CL8'EXIT '
GETMAIN DC CL8'GETMAIN '
FREEMAIN DC CL8'FREEMAIN'
DC CL8'LINK '
DC CL8'XCTL '
DC CL8'LOAD '
DC CL8'DELETE '
DC CL8'GET/FREE'
DC CL8'TIME '
DC CL8'SYNCH '
DC CL8'ABEND '
DC CL8'SPIE '
DC CL8'ERREXCP '
DC CL8'PURGE '
DC CL8'RESTORE '
DC CL8'BLDL/FND'
DC CL8'OPEN '
DC CL8'CLOSE '
DC CL8'STOW '
DC CL8'OPEN=J '
DC CL8'CLOSE=J '
DC CL8'DEVTYPE '
DC CL8'TRKBAL '
DC CL8'CAT/LOC '
DC CL8'OBTAIN '
DC CL8'RES-1C '
DC CL8'SCRATCH '
DC CL8'RENAME '
DC CL8'FEOV '
DC CL8'ALLOC '
DC CL8'IOHALT '
DC CL8'MGCR/QED'
DC CL8'WTO/WTOR'
DC CL8'WTL '
DC CL8'SEGLD/WT'
DC CL8'RES-26 '
DC CL8'LABEL '
DC CL8'EXTRACT '
DC CL8'IDENTIFY'
DC CL8'ATTACH '
DC CL8'CIRB '
DC CL8'CHAP '
DC CL8'OVLYBRCH'
DC CL8'TTIMER '
DC CL8'STIMER '
DC CL8'DEQ '
DC CL8'RES-31 '

```

DC CL8'RES-32 '  
DC CL8'SNAP/SDP'  
DC CL8'RESTART '  
DC CL8'RELEX '  
DC CL8'DISABLE '  
DC CL8'E0V '  
DC CL8'ENQ/RESV'  
DC CL8'FREEDBUF'  
DC CL8'REL/QBUF'  
DC CL8'OLTEP '  
DC CL8'STAE/STI'  
DC CL8'IKJEGS6A'  
DC CL8'DETACH '  
DC CL8'CHKPT '  
DC CL8'RDJCFB '  
DC CL8'RES-41 '  
DC CL8'BTAMTEST'  
DC CL8'RES-43 '  
DC CL8'SYNADAF '  
DC CL8'BSP '  
DC CL8'GSERV '  
DC CL8'ASGNBFR '  
DC CL8'NO MAC48'  
DC CL8'SPAR '  
DC CL8'DAR '  
DC CL8'DQUEUE '  
DC CL8'IFBSTAT '  
DC CL8'RES-4D '  
DC CL8'LSPACE '  
DC CL8'STATUS '  
DC CL8'RES-50 '  
DC CL8'SETPRT '  
DC CL8'RES-52 '  
DC CL8'SMFWTM '  
DC CL8'GRAPHICS'  
DC CL8'DDRSWAP '  
DC CL8'ATLAS '  
DC CL8'DOM '  
DC CL8'RES-58 '  
DC CL8'RES-59 '  
DC CL8'RES-5A '  
DC CL8'VOLSTAT '  
DC CL8'TCPEXCP '  
DC CL8'TGET/PUT'  
DC CL8'SEESVC94'  
DC CL8'SYSEVENT'  
DC CL8'STAX '  
DC CL8'IKJEGS9G'  
DC CL8'PROTECT '  
DC CL8'DYNALLOC'  
DC CL8'IKJEFFIB'  
DC CL8'QTIP '

```

DC      CL8'AQCTL      '
DC      CL8'XLATE     '
DC      CL8'TOPCTL    '
DC      CL8'IMGLIB    '
DC      CL8'RES-6A    '
DC      CL8'MODESET   '
DC      CL8'RES-6C    '
DC      CL8'C SVC109'
DC      CL8'RES-6E    '
DC      CL8'NO MAC6F'
DC      CL8'PGRLSE    '
DC      CL8'PGFIX/RE'
DC      CL8'EXCPVR    '
DC      CL8'RES-73    '
DC      CL8'C SVC116'
DC      CL8'DEBCHK    '
DC      CL8'RES-76    '
DC      CL8'TESTAUTH'
DC      CL8'GET/FR78'
DC      CL8'VSAM      '
DC      CL8'C SVC122'
DC      CL8'PURGEDQ   '
DC      CL8'TPIO      '
DC      CL8'EVENTS    '
DC      CL8'MSSICBSV'
DC      CL8'RES-7F    '
DC      CL8'RES-80    '
DC      CL8'RES-81    '
DC      CL8'RACHECK   '
DC      CL8'RACINIT   '
DC      CL8'RACLIST   '
DC      CL8'RACDEF    '
DC      CL8'RES-86    '
DC      CL8'RES-87    '
DC      CL8'RES-88    '
DC      CL8'ESR       '
DC      CL8'PGSER     '
DC      CL8'CVAFDSM   '
DC      CL8'RES-8C    '
DC      CL8'RES-8D    '
DC      CL8'RES-8E    '
DC      CL8'CIPHER    '
DC      CL8'NO MAC90'

```

EJECT

```

*****
*          SERVICE IDENTIFIER NUMBERS WITH THEIR CORRESPONDING MACROS          *
*****

```

```

SPACE
PPGSSID DC      XL2'0001',CL8'WAIT      '
PPGSSIDL EQU    *-PPGSSID
SPACE
DC      XL2'0002',CL8'POST      '

```

DC XL2'0004',CL8'GETMAIN '  
 DC XL2'0005',CL8'FREEMAIN '  
 DC XL2'000A',CL8'GET/FREE '  
 DC XL2'005F',CL8'SYSEVENT '  
 DC XL2'0078',CL8'GET/FREE '  
 DC XL2'007A',CL8'SPI/INT '  
 DC XL2'0100',CL8'ETCON '  
 DC XL2'0101',CL8'ETCRE '  
 DC XL2'0102',CL8'ATSET '  
 DC XL2'0103',CL8'AXSET '  
 DC XL2'0104',CL8'AXEXT '  
 DC XL2'0105',CL8'AXFRE '  
 DC XL2'0106',CL8'AXRES '  
 DC XL2'0107',CL8'ETDES '  
 DC XL2'0108',CL8'ETDIS '  
 DC XL2'0109',CL8'LXFRE '  
 DC XL2'010A',CL8'LXRES '  
 DC XL2'010E',CL8'SUSPEND '  
 DC XL2'010F',CL8'RESUME '  
 DC XL2'0110',CL8'SCHEDULE '  
 DC XL2'0111',CL8'SCHEDULE '  
 DC XL2'0112',CL8'SCHEDULE '  
 DC XL2'0113',CL8'DSGNL '  
 DC XL2'0114',CL8'RISGNL '  
 DC XL2'0115',CL8'RPSGNL '  
 DC XL2'0116',CL8'SCHEDULE '  
 DC XL2'0117',CL8'SCHEDULE '  
 DC XL2'0118',CL8'SUSPEND '  
 DC XL2'0119',CL8'RESUME '  
 DC XL2'011A',CL8'RESUME '  
 DC XL2'011B',CL8'RESUME '  
 DC XL2'011C',CL8'SCHEDULE '  
 DC XL2'0128',CL8'WAIT '  
 DC XL2'0129',CL8'POST '  
 DC XL2'012A',CL8'POST '  
 DC XL2'012B',CL8'POST '  
 DC XL2'012C',CL8'ASCBCHAP '  
 DC XL2'012D',CL8'STATUS '  
 DC XL2'012E',CL8'STATUS '  
 DC XL2'0132',CL8'STOR-GET '  
 DC XL2'0133',CL8'STOR-REL '  
 DC XL2'0146',CL8'SPI/INT '

SPACE

PPGSSID# EQU ((\*-PPGSSID)/PPGSSIDL)

EJECT

PPGDSECT DSECT

PPGTYPE DS X

PPGEXPL DS XL2

PPGLEN DS X

PPGAPGM DS A

PPGNAME DS CL4

SPACE 3



```

PATDSECT DSECT
          DS      C
PATINTC  DS      CL13
PATDPSW2 DS      CL8
PATC1    DS      C
          DS      CL7
PATDPGM  DS      CL8
PATC2    DS      C
          DS      CL7
PATDOFF  DS      CL8
PATC3    DS      C
          SPACE
          YREGS
          SPACE
          IHAPSA
          SPACE
          IHATBVT
          EJECT
          IHATTE AS=YES,                                X
                BR=YES,  ALL BRANCH TRACING ENTRY MAPPINGS      X
                EX=YES,  ALL EXPLICIT TRACING ENTRY MAPPINGS    X
                ACR=YES,  ACR ENTRY MAPPING                      X
                ALTR=YES, TRACE OPTIONS ALTERATION ENTRY MAPPING X
                DSP=YES,  ALL DISPATCHER ENTRY MAPPINGS          X
                EXT=YES,  ALL EXTERNAL INTERRUPT ENTRY MAPPINGS X
                IO=YES,   I/O INTERRUPT ENTRY MAPPING            X
                MCH=YES,  MACHINE CHECK INTERRUPT ENTRY MAPPING X
                PGM=YES,  PROGRAM INTERRUPT ENTRY MAPPING        X
                RCYV=YES, RECOVERY ENTRY MAPPING                 X
                RST=YES,  RESTART INTERRUPT ENTRY MAPPING        X
                SCH=YES,  ALL SUBCHANNEL ENTRY MAPPINGS          X
                SIGA=YES, SIGNAL ADAPTER MAPPING                 X
                SPER=YES, SLIP/PER EVENT ENTRY MAPPING           X
                SSRV=YES, PC OR BR ENTERED SYSTEM SERVICE ENTRY MAPPING X
                SUSP=YES, SUSPENSION ENTRY MAPPING                X
                SVC=YES,  ALL SUPERVISOR CALL ENTRY MAPPING       X
                TIME=YES, TIMER SERVICES ENTRY MAPPING           X
                USRN     USER EVENT TRACE ENTRY MAPPING
          EJECT
          CVT    DSECT=YES
          SPACE 2
          IHAECVT LIST=YES
          SPACE 2
          IEEBASEA
          SPACE 2
          IEEZB806
          SPACE 2
          IEZMTPRM
          SPACE
          IHAASCB
          SPACE
          IHAASVT

```

```

SPACE
IKJTBC
SPACE
IHAXTLST
SPACE
IHACDE
SPACE
IHALPDE
SPACE
IHATRV
SPACE
IHATOB
SPACE
IHACSD
SPACE
* IEANUCMP
NUCMEN DSECT
NUCMNAME DS CL8
NUCMADDR DS F
NUCMFLAG DS X
NUCMLEN DS XL3
EJECT
*****
* DSECT FOR AN I/O INTERRUPT SYSTEM TRACE TABLE ENTRY *
*****
SPACE
PPG00BD DSECT
P0BDTCBC DS CL4
P0BDTCB DS CL8
DS 2C
P0BDHAC DS CL5
DS C
P0BDHID DS CL4
DS 2C
P0BDPAC DS CL5
DS C
P0BDPID DS CL4
DS 2C
P0BDSAC DS CL5
DS C
P0BDSID DS CL4
DS 2C
P0BDP1C DS CL5
P0BDPSW1 DS CL8
DS 2C
P0BDP2C DS CL5
P0BDPSW2 DS CL8
DS 2C
P0BDADRC DS CL6
DS C
P0BDADR DS CL8
DS 2C

```

```

PØBDNUMC DS    CL4
          DS    C
PØBDNUM  DS    CL4
          DS    2C
PØBDSTAC DS    CL4
          DS    C
PØBDSTAT DS    CL4
          DS    2C
PØBRESDC DS    CL8
          DS    C
PØBRSDUL DS    CL4
          EJECT

```

```

*****
*          DSECT FOR A TASK DISPATCH SYSTEM TRACE TABLE ENTRY          *
*****

```

```

          SPACE
PPGØØFD  DSECT
PØFDTCBC DS    CL4
PØFDTCB  DS    CL8
          DS    2C
PØFDHAC  DS    CL5
          DS    C
PØFDHID  DS    CL4
          DS    2C
PØFDPAC  DS    CL5
          DS    C
PØFDPID  DS    CL4
          DS    2C
PØFDSAC  DS    CL5
          DS    C
PØFDSID  DS    CL4
          DS    2C
PØFDP1C  DS    CL5
PØFDPSW1 DS    CL8
          DS    2C
PØFDP2C  DS    CL5
PØFDPSW2 DS    CL8
          DS    2C
PØFDGØC  DS    CL5
PØFDGØ   DS    CL8
          DS    2C
PØFDG1C  DS    CL5
PØFDG1   DS    CL8
          DS    2C
PØFSUPRC DS    CL6
PØFSUPER DS    CL8
          EJECT

```

```

*****
*          DSECT FOR A GENERAL EXTERNAL INTERRUPT SYSTEM TRACE TABLE ENTRY *
*****

```

```

          SPACE
PPGØØ3D  DSECT

```

P03DTCBC	DS	CL4
P03DTCB	DS	CL8
	DS	2C
P03DHAC	DS	CL5
	DS	C
P03DHID	DS	CL4
	DS	2C
P03DPAC	DS	CL5
	DS	C
P03DPID	DS	CL4
	DS	2C
P03DSAC	DS	CL5
	DS	C
P03DSID	DS	CL4
	DS	2C
P03DP1C	DS	CL5
P03DPSW1	DS	CL8
	DS	2C
P03DP2C	DS	CL5
P03DPSW2	DS	CL8
	DS	2C
P03DXICC	DS	CL9
	DS	C
P03DXIC	DS	CL8
	DS	2C
P03DLCLC	DS	CL5
	DS	C
P03DLOCL	DS	CL8

EJECT

\*\*\*\*\*  
 \* DSECT FOR AN SVC INTERRUPT SYSTEM TRACE TABLE ENTRY \*  
 \*\*\*\*\*

	SPACE	
PPG005D	DSECT	
P05DTCBC	DS	CL4
P05DTCB	DS	CL8
	DS	C
P05DJNAM	DS	CL8
	DS	C
P05DSVCC	DS	CL5
	DS	C
P05DSVCN	DS	CL4
	DS	C
P05DSVNM	DS	CL8
	DS	2C
P05DHAC	DS	CL5
	DS	C
P05DHID	DS	CL4
	DS	2C
P05DP1C	DS	CL5
P05DPSW1	DS	CL8
	DS	2C

```

P05DP2C DS CL5
P05DPSW2 DS CL8
          DS 2C
P05DG0C DS CL5
P05DG0 DS CL8
          DS 2C
P05DG1C DS CL5
P05DG1 DS CL8
          DS 2C
P05DGFC DS CL5
P05DGF DS CL8
          EJECT

```

```

*****
*          DSECT FOR A PROGRAM INTERRUPT SYSTEM TRACE TABLE ENTRY          *
*****

```

```

          SPACE
PPG007D DSECT
P07DTCBC DS CL4
P07DTCB DS CL8
          DS 2C
P07DHAC DS CL5
          DS C
P07DHID DS CL4
          DS 2C
P07DPAC DS CL5
          DS C
P07DPID DS CL4
          DS 2C
P07DSAC DS CL5
          DS C
P07DSID DS CL4
          DS 2C
P07DP1C DS CL5
P07DPSW1 DS CL8
          DS 2C
P07DP2C DS CL5
P07DPSW2 DS CL8
          DS 2C
P07DILC DS CL8
          DS C
P07DIL DS CL4
          DS 2C
P07DICC DS CL9
          DS C
P07DCODE DS CL4
          EJECT

```

```

*****
*          DSECT FOR AN ALTERNATE CPU RECOVERY SYSTEM TRACE TABLE ENTRY    *
*****

```

```

          SPACE
PPG017D DSECT
P17DTCBC DS CL4

```

P17DTCB	DS	CL8
	DS	2C
P17DHAC	DS	CL5
	DS	C
P17DHID	DS	CL4
	DS	2C
P17CPUPA	DS	CL5
	DS	C
P17CPID	DS	CL4
	DS	2C
P17FLAG	DS	CL5
	DS	C
P17FG	DS	CL2
	DS	2C
P17EPSWC	DS	CL5
P17EPSW	DS	CL8
	DS	2C
P17SUPRC	DS	CL6
P17SUPER	DS	CL8
	DS	2C
P17LOCLC	DS	CL6
P17LOCAL	DS	CL8
	DS	2C

EJECT

\*\*\*\*\*

\* DSECT FOR AN SRB DISPATCH SYSTEM TRACE TABLE ENTRY \*

\*\*\*\*\*

SPACE

PPG10FD	DSECT	
PAFDWRKC	DS	CL5
PAFDWORK	DS	CL8
	DS	2C
PAFDHAC	DS	CL5
	DS	C
PAFDHID	DS	CL4
	DS	2C
PAFDRAC	DS	CL10
	DS	C
PAFDRID	DS	CL4
	DS	2C
PAFCPUPA	DS	CL5
	DS	C
PAFCPID	DS	CL4
	DS	2C
PAFFLAGC	DS	CL5
	DS	C
PAFFLAG	DS	CL4
	DS	2C
PAFDP1C	DS	CL5
PAFDPSW1	DS	CL8
	DS	2C
PAFDP2C	DS	CL5

```

PAFDPSW2 DS CL8
          DS 2C
PAFDG0C DS CL5
PAFDG0 DS CL8
          DS 2C
PAFDG1C DS CL5
PAFDG1 DS CL8
          EJECT

```

```

*****
* DSECT FOR A PC OR BRANCH-ENTERED SYSTEM SERVICE TRACE TABLE ENTRY *
*****

```

```

          SPACE
PPG205D DSECT
P25DTCBC DS CL4
P25DTCB DS CL8
          DS 2C
P25DSVRC DS CL8
          DS C
P25DSVRN DS CL4
          DS C
P25DSVRM DS CL8
          DS 2C
P25DHAC DS CL5
          DS C
P25DHID DS CL4
          DS 2C
P25RETC DS CL8
P25RETA DS CL8
          DS 2C
P25DW1C DS CL6
P25DWD1 DS CL8
          DS 2C
P25DW2C DS CL6
P25DWD2 DS CL8
          DS 2C
P25DW3C DS CL6
P25DWD3 DS CL8
          DS 2C
P25DW4C DS CL6
P25DWD4 DS CL8
          DS 2C
          EJECT

```

```

*****
* DSECT FOR A RECOVERY EVENT SYSTEM TRACE TABLE ENTRY *
*****

```

```

          SPACE
PPG01DD DSECT
P1DDTCBC DS CL4
P1DDTCB DS CL8
          DS 2C
P1DDHAC DS CL5
          DS C

```

P1DDHID	DS	CL4
	DS	2C
P1DDSI	DS	CL9
P1DDSI	DS	CL4
	DS	2C
P1DDLHC	DS	CL10
P1DDLH	DS	CL8
	DS	2C
P1DDW1C	DS	CL6
P1DDW1	DS	CL8
	DS	2C
P1DDW2C	DS	CL6
P1DDW2	DS	CL8
	DS	2C
P1DDW8C	DS	CL6
P1DDW8	DS	CL8

EJECT

\*\*\*\*\*

\* DSECT FOR A WAIT DISPATCH SYSTEM TRACE TABLE ENTRY \*

\*\*\*\*\*

SPACE

PPGF0FD	DSECT	
PF0DTCBC	DS	CL4
PF0DTCB	DS	CL8
	DS	2C
PF0DHAC	DS	CL5
	DS	C
PF0DHID	DS	CL4
	DS	2C
PF0DTDC	DS	CL10
	DS	C
PF0DTOD	DS	CL12

EJECT

\*\*\*\*\*

\* DSECT FOR AN EXTERNAL CALL SYSTEM TRACE TABLE ENTRY \*

\*\*\*\*\*

SPACE

PPG303D	DSECT	
P30DTCBC	DS	CL4
P30DTCB	DS	CL8
	DS	2C
P30DHAC	DS	CL5
	DS	C
P30DHID	DS	CL4
	DS	2C
P30DPAC	DS	CL5
	DS	C
P30DPID	DS	CL4
	DS	2C
P30DSAC	DS	CL5
	DS	C
P30DSID	DS	CL4



	DS	2C
P30DP1C	DS	CL5
P30DPSW1	DS	CL8
	DS	2C
P30DP2C	DS	CL5
P30DPSW2	DS	CL8
	DS	2C
P30DCPUC	DS	CL9
P30DCPU	DS	CL4
	DS	2C
P30DXICC	DS	CL6
P30DXIC	DS	CL4
	DS	2C
P30DSIGC	DS	CL5
P30DSIG	DS	CL1
	DS	2C
P30DLOKC	DS	CL8
P30DLOK	DS	CL8
	DS	2C

EJECT

\*\*\*\*\*  
 \* DSECT FOR A CLOCK COMPARATOR EXTERNAL INTERRUPT TTE \*  
 \*\*\*\*\*

SPACE

PPG403D	DSECT	
P40DTCBC	DS	CL4
P40DTCB	DS	CL8
	DS	2C
P40DHAC	DS	CL5
	DS	C
P40DHID	DS	CL4
	DS	2C
P40DPAC	DS	CL5
	DS	C
P40DPID	DS	CL4
	DS	2C
P40DSAC	DS	CL5
	DS	C
P40DSID	DS	CL4
	DS	2C
P40DP1C	DS	CL5
P40DPSW1	DS	CL8
	DS	2C
P40DP2C	DS	CL5
P40DPSW2	DS	CL8
	DS	2C
P40DXICC	DS	CL6
P40DXIC	DS	CL4
	DS	2C
P40DLOKC	DS	CL7
P40DLOK	DS	CL8
	DS	2C

P40DTQTC DS CL5  
P40DTTCB DS CL8  
DS 2C  
P40DTQAC DS CL4  
P40DTAID DS CL4

EJECT

\*\*\*\*\*

\* DSECT FOR A START SUBCHANNEL SYSTEM TRACE TABLE ENTRY \*

\*\*\*\*\*

SPACE

PPG001D DSECT

P01DTCBC DS CL4

P01DTCB DS CL8

DS 2C

P01DIAC DS CL5

DS C

P01DIID DS CL4

DS 2C

P01DHAC DS CL5

DS C

P01DHID DS CL4

DS 2C

P01DCCC DS CL9

DS C

P01DCC DS CL2

DS 2C

P01DVRC DS CL7

DS C

P01DVR DS CL2

DS 2C

P01DADRC DS CL6

DS C

P01DADR DS CL8

DS 2C

P01DNUMC DS CL4

DS C

P01DNUM DS CL4

DS 2C

P01DIOSC DS CL8

P01DIOS DS CL8

DS 2C

P01DCAPC DS CL10

DS C

P01DCAP DS CL8

DS 2C

P01DOR2C DS CL6

DS C

P01DOR2 DS CL8

DS 2C

P01DOR3C DS CL6

DS C

P01DOR3 DS CL8

EJECT

\*\*\*\*\*  
\* DSECT FOR A MODIFY SUBCHANNEL SYSTEM TRACE TABLE ENTRY \*  
\*\*\*\*\*

SPACE

PPG101D DSECT  
P11DTCBC DS CL4  
P11DTCB DS CL8  
DS 2C  
P11DIAC DS CL5  
DS C  
P11DIID DS CL4  
DS 2C  
P11DHAC DS CL5  
DS C  
P11DHID DS CL4  
DS 2C  
P11DCCC DS CL9  
DS C  
P11DCC DS CL2  
DS 2C  
P11DVRC DS CL7  
DS C  
P11DVR DS CL2  
DS 2C  
P11DADRC DS CL6  
DS C  
P11DADR DS CL8  
DS 2C  
P11DNUMC DS CL4  
DS C  
P11DNUM DS CL4  
DS 2C  
P11DORBC DS CL8  
DS C  
P11DORB DS CL4  
DS 2C  
P11DLPMC DS CL10  
DS C  
P11DLPM DS CL2  
DS 2C  
P11DPOMC DS CL10  
DS C  
P11DPOM DS CL2  
DS 2C  
P11DIOSC DS CL8  
P11DIOS DS CL8

EJECT

\*\*\*\*\*  
\* DSECT FOR A HALT SUBCHANNEL SYSTEM TRACE TABLE ENTRY \*  
\*\*\*\*\*

SPACE

```

PPG201D  DSECT
P21DTCBC DS    CL4
P21DTCB  DS    CL8
          DS    2C
P21DIAC  DS    CL5
          DS    C
P21DIID  DS    CL4
          DS    2C
P21DHAC  DS    CL5
          DS    C
P21DHID  DS    CL4
          DS    2C
P21DCCC  DS    CL9
          DS    C
P21DCC   DS    CL2
          DS    2C
P21DVRC  DS    CL7
          DS    C
P21DVR   DS    CL2
          DS    2C
P21DADRC DS    CL6
          DS    C
P21DADR  DS    CL8
          DS    2C
P21DNUMC DS    CL4
          DS    C
P21DNUM  DS    CL4
          DS    2C
P21DIOQC DS    CL7
          DS    C
P21DIOQ  DS    CL8
          DS    2C
P21DIOSC DS    CL8
P21DIOS  DS    CL8

```

EJECT

```

*****
*          DSECT FOR A CLEAR SUBCHANNEL SYSTEM TRACE TABLE ENTRY          *
*****

```

SPACE

```

PPG301D  DSECT
P31DTCBC DS    CL4
P31DTCB  DS    CL8
          DS    2C
P31DIAC  DS    CL5
          DS    C
P31DIID  DS    CL4
          DS    2C
P31DHAC  DS    CL5
          DS    C
P31DHID  DS    CL4
          DS    2C
P31DCCC  DS    CL9
          DS    C

```

P31DCC	DS	CL2
	DS	2C
P31DVRC	DS	CL7
	DS	C
P31DVR	DS	CL2
	DS	2C
P31DADRC	DS	CL6
	DS	C
P31DADR	DS	CL8
	DS	2C
P31DNUMC	DS	CL4
	DS	C
P31DNUM	DS	CL4
	DS	2C
P31DIOQC	DS	CL7
	DS	C
P31DIOQ	DS	CL8
	DS	2C
P31DIOSC	DS	CL8
P31DIOS	DS	CL8
	EJECT	

\*\*\*\*\*  
 \* DSECT FOR A RESUME SUBCHANNEL SYSTEM TRACE TABLE ENTRY \*  
 \*\*\*\*\*

	SPACE	
PPG401D	DSECT	
P41DTCBC	DS	CL4
P41DTCB	DS	CL8
	DS	2C
P41DIAC	DS	CL5
	DS	C
P41DIID	DS	CL4
	DS	2C
P41DHAC	DS	CL5
	DS	C
P41DHID	DS	CL4
	DS	2C
P41DCCC	DS	CL9
	DS	C
P41DCC	DS	CL2
	DS	2C
P41DVRC	DS	CL7
	DS	C
P41DVR	DS	CL2
	DS	2C
P41DADRC	DS	CL6
	DS	C
P41DADR	DS	CL8
	DS	2C
P41DNUMC	DS	CL4
	DS	C
P41DNUM	DS	CL4

```

        DS      2C
P41DIO SC DS      CL8
P41DIO S DS      CL8
        EJECT
*****
*          DSECT FOR A SET SECONDARY ASN SYSTEM TRACE TABLE ENTRY          *
*****
        SPACE
PPGSSAD DSECT
PSRDNIDC DS      CL10
PSRDNID  DS      CL4
        EJECT
*****
*          DSECT FOR A PROGRAM CALL SYSTEM TRACE TABLE ENTRY              *
*****
        SPACE
PPGPCD  DSECT
PPCDKEYC DS      CL8
PPCDKEY  DS      CL1
          DS      CL2
PPCDNUMC DS      CL4
PPCDNUM  DS      CL3
          DS      CL2
PPCDRADC DS      CL8
PPCDRAD  DS      CL8
          DS      CL2
PPCDNSTC DS      CL10
PPCDSTAT DS      CL4
        EJECT
*****
*          DSECT FOR A PROGRAM TRANSFER SYSTEM TRACE TABLE ENTRY          *
*****
        SPACE
PPGPTD  DSECT
PPTDKEYC DS      CL8
PPTDKEY  DS      CL1
          DS      CL2
PPTDNIDC DS      CL9
PPTDNID  DS      CL4
          DS      CL2
PPTDRADC DS      CL8
PPTDRAD  DS      CL8
          DS      CL2
PPTDNSTC DS      CL10
PPTDSTAT DS      CL4
        EJECT
*          DSECT FOR A PROGRAM RETURN SYSTEM TRACE TABLE ENTRY            *
        SPACE
PPGPRD  DSECT
PPRDKEYC DS      CL8
PPRDKEY  DS      CL1
          DS      CL2

```

```

PPRDNIDC DS    CL9
PPRDNID  DS    CL4
          DS    CL2
PPRDRADC DS    CL8
PPRDRAD  DS    CL8
          DS    CL2
PPRDNSTC DS    CL10
PPRDSTAT DS    CL4
          DS    CL2
PPRDAADC DS    CL14
PPRDAAD  DS    CL8
          EJECT
*****
*          DSECT FOR A BRANCH IN SUBSPACE GROUP TRACE TABLE ENTRY          *
*****
          SPACE
PPGBSGD  DSECT
PBSDADRC DS    CL11
PBSDADR  DS    CL8
          DS    CL2
PBSDALEC DS    CL12
PBSDALE  DS    CL11
          DS    CL2
PBSDASNC DS    CL5
PBSDASN  DS    CL2
          DS    CL2
PBSDNUMC DS    CL7
PBSDNUM  DS    CL4
          SPACE
          END

```

---

*John Gizelar*  
*Systems Programmer (USA)*

© Xephon 2001

---

## A user problem tip

When dealing with users' problems, it can often be useful to see the current state of their screen for diagnostic purposes. A useful way of doing this, if they are using a Windows PC as a terminal, is to get them to press the 'PrtSc' (print screen) key and then open up a package such as Microsoft Word. The screen can then be pasted into Microsoft Word and e-mailed as an attachment back to the support area.

---

*Systems Programmer (UK)*

© Xephon 2001

---

Level 8 Systems has announced Geneva AppBuilder 2.0, an enhanced version of the product formerly known as SeerHPS. Geneva AppBuilder 2.0 enables application developers to quickly build and deploy enterprise-scale, multi-tier e-business applications across multiple platforms including OS/390.

Geneva AppBuilder 2.0 adds support for creating J2EE applications from specifications stored in the Geneva AppBuilder repository by generating Java, HTML/Java Servlets and Enterprise Java Beans (EJBs), technologies critical to creating enterprise e-business applications. Using Geneva AppBuilder's HTML/Java Servlet and Java generation capability, both thin HTML and graphically rich downloadable Java clients can be created from the same specification.

AppBuilder allows the Java environments to take advantage of OS/390 by connecting the Enterprise Java Bean and HTML/Servlet applications to Geneva AppBuilder-created applications on OS/390, which is based on CICS or IMS COBOL.

For further information contact:

Level 8 Systems, 8000 Regency Parkway,  
Cary, NC 27511, USA  
Tel: (919) 380-5000  
Fax: (919) 469-1910

Level 8 Systems, Harman House, Ground  
Floor, George Street, Uxbridge, Middlesex,  
UB8 1QQ, UK  
Tel: (01895) 206 700  
Fax: (01895) 206 740  
<http://www.level8.com>

\* \* \*

Software AG has announced that its Tamino XML database product has been extended to include support for both OS/390 and AIX. The product is designed to allow the use of XML to gain access to existing applications running on these platforms. This support adds to existing Tamino support on Linux, Solaris, UnixWare, and Windows NT and 2000.

For further information contact:

Software AG, Uhlandstr. 12, 64297  
Darmstadt, Germany.  
Tel: +49 6151 92 0  
Fax: +49 6151 92 1191  
<http://www.softwareag.com>

\* \* \*

Xephon will be holding its annual *MVS 2001* conference at the Radisson SAS Portman Hotel in London, on 5-6 June 2001. *MVS 2001* is designed specifically for technical managers, systems programmers, strategic planners, and other system specialists at MVS/ESA, OS/390, and z/OS installations.

In a separate conference on 7 June 2001, also at the Portman Hotel, Xephon will provide a complete analysis of *z/OS: IBM and ISV Mainframe Software Pricing*.

For *MVS Update* subscribers the attendance fee for each event is £570.00 plus £66.50 VAT. For further information, please telephone the registrar, Toni Brown, on (01635) 33823.

<http://www.xephon.com/zevent.html>

\* \* \*

