





Version: 'V04-000'

```

*****
*
* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
* ALL RIGHTS RESERVED.
*
* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
* TRANSFERRED.
*
* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
* CORPORATION.
*
* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
*
*****

```

MODULE: ISGENC.REQ

FACILITY: LINKER

ABSTRACT: REQUIRE FILE FOR I-SECTION GENERATION CONTROL

HISTORY:

AUTHOR: T.J. PORTER 01-APR-77

MODIFICATIONS:

- V03-001      JWT0152      Jim Teague      10-Feb-1984  
Change ISD\$C SIZE to reflect maximum size of new variable  
length global section names.
- V02-002      BLS0026      Benn Schreiber      28-Nov-1980  
Remove ascii name string from isect generation tables.
- V02-001      BLS0007      Benn Schreiber      15-Aug-1980  
Convert to MDL data structures.

--

TI

MA

LI

++

## functional description:

this is a require file which sets up all necessary data for generation of image sections. generation of image sections is driven by a table chosen depending upon the kind of link being performed.

exe\_isgentbl is used for normal user executable images

shr\_isgentbl is used for shareable images

sys\_isgentbl is used for system space images

the order of entries in these tables is the order in which the image sections are generated. there is one entry for each image section to be generated.

each entry provides the following information:

name of isection build routine

p-section attribute mask (the and mask)

p-section match pattern

i-section code which appears in the isd of header

size of the isd written to header

page fault cluster size

isd flags.

this file requires that sysdef be previously declared to define the isd's and the image header layout.

--

## COMPILETIME

NUM\_ISECTS=0;

! VARIABLE FOR TABLE GENERATION

DEFINE GENERIC I-SECTION TYPE PARAMETERS:-

SIZE = NUMBER OF BYTES IN ISD OF HEADER

CODE = IMAGE SECTION TYPE CODE IN ISD

PFC = PAGE FAULT CLUSTER SIZE

GBL = INDICATOR FOR WHETHER POTENTIALLY GLOBAL

TRUE OR FALSE

DZRO = DEMAND ALLOCATE ZERO FILL IMAGE SECTION

TYPE INDICATOR - TRUE OR FALSE

## LITERAL

SUFFIX\_SIZE = 4,

! Size of "\_nnn" global isd suffix

NULL\_SIZE = 0,

! NULL ISD TYPE

NULL\_CODE = 0,

NULL\_PFC = 0,

NULL\_GBL = FALSE,

NULL\_DZRO = FALSE,

NORMAL\_GBL = FALSE,

NORMAL\_DZRO = FALSE,

NORMAL\_CODE = ISD\$K\_NORMAL,

NORMAL\_SIZE = ISD\$K\_LENPRIV,

NORMAL\_PFC = NULL\_PFC,

! NORMAL ISD'S TYPE

! NORMAL ISD SIZE

! NORMALLY NULL

```

STACK_CODE = ISD$K_USRSTACK,           ! STACK ISD TYPE CODE
STACK_SIZE = ISD$K_LENDRZRO,          ! DEMAND ZERO ISD
STACK_PFC = NULL_PFC,                 ! DEFAULT IS NULL
STACK_GBL = FALSE,
STACK_DZRO = TRUE,

SHRFXD_CODE = ISL$C_SHRFXD,           ! SHARABLE AT FIXED PLACE
SHRFXD_SIZE = ISD$K_LENPRIV,          ! IN THE ADDRESS SPACE
SHRFXD_PFC = NULL_PFC,
SHRFXD_GBL = FALSE,
SHRFXD_DZRO = FALSE,

PRVFXD_CODE = ISL$C_PRVFXD,           ! PRIVATE IN FIXED PLACE OF ADDRESS
PRVFXD_SIZE = ISD$K_LENPRIV,
PRVFXD_PFC = NULL_PFC,
PRVFXD_GBL = FALSE,
PRVFXD_DZRO = FALSE,

SHRPIC_CODE = ISL$C_SHRPIC,           ! SHAREABLE POSITION INDEPENDENT
SHRPIC_SIZE = ISD$K_LENPRIV,
SHRPIC_PFC = NULL_PFC,
SHRPIC_GBL = FALSE,
SHRPIC_DZRO = FALSE,

PRVPIC_CODE = ISL$C_PRVVIC,           ! PRIVATE PIC SECTIONS
PRVPIC_SIZE = ISD$K_LENPRIV,
PRVPIC_PFC = NULL_PFC,
PRVPIC_GBL = FALSE,
PRVPIC_DZRO = FALSE,

RDWRTEXEREL = GPSS$M_WRT OR GPSS$M_EXE OR GPSS$M_REC, ! MASK FOR READ,
SHRPICMASK = RDWRTEXEREL OR GPSS$M_SHR OR GPSS$M_PIC, ! WRITE EXECUTE AND RELOCATABLE
! SEARCH MASK FOR PIC AND SHAREABLE P-SECTIONS
! WHICH ARE READ, WRITE EXECUTABLE
! AND RELOCATABLE
RDWRTEXERELVEC = RDWRTEXEREL OR GPSS$M_VEC, ! MASK FOR VECTOR IMAGE SECTIONS
SHRPICVEC = SHRPICMASK OR GPSS$M_VEC; ! MASK FOR SHAREABLE VECTOR IMAGE SECTIONS

DEFINE THE HEADER FILLING

GLOBAL LITERAL
ISD$C_SIZE = ISD$K_MAXLENGLBL + ISL$C_SIZE : WEAK, ! SIZE OF FULL IMAGE SECTION DESCRIPTOR INCLUDING LINKER PART
HDR$K_FILLCHR = 255 : BYTLIT WEAK, ! FILL CHARACTER
IHDS$K_SHR = IHDS$K_LIM : WEAK, ! JUST A DIFFERENT NAME
IHDS$K_ACTIVOFF = IHDS$K_LENGTH : WEAK, ! OFFSET TO ACTIVATION AREA
IHDS$K_SYMDBGOFF = IHDS$K_ACTIVOFF + IHASK_LENGTH : WEAK, ! OFFSET TO SYMBOL TABLE DATA
IHDS$K_IMGIDOFF = IHDS$K_SYMDBGOFF + IHSSK_LENGTH : WEAK, ! OFFSET TO IMAGE IDENTIFICATION AREA
IHDS$K_PATCHOFF = IHDS$K_IMGIDOFF + IHISK_LENGTH : WEAK, ! OFFSET TO PATCH CONTROL AREA
IHDS$K_MAXLENGTH = IHDS$K_PATCHOFF : WEAK, ! MAXIMUM HEADER LENGTH WITHOUT PATCH AREA
HDR$K_MINFILL = 2 : WEAK; ! MINIMUM FILL MUST EQUAL WIDTH
! OF ISD SIZE FIELD

```

THE FOLLOWING MACROS GENERATE ACCESSING MACROS FOR BLOCK

STRUCTURES WHICH HAVE FIELDS OF BYTE, WORD OR LONG WIDTH.

COMPILETIME

YYYYYYYSIZE = 0;

! VARIABLE WHICH IS RESULTANT BLOCK LENGTH

MACRO

GENBLOCKFIELDS(BLKNAM)[] =

CALLING SEQUENCE:

GENBLOCKFIELDS(BLKNAM,  
FLDNAM1,FLDWIDTH1,  
FLDNAM2,FLDWIDTH2,  
.....

ETC,

FLDNAMN,FLDWIDTHN);

WHERE:

FLDNAMI = THE NAME OF FIELD I IN THE BLOCK  
FLDWIDTHI = WIDTH IN BYTES OF FIELD I

THE MACRO GENERATES FIELD ACCESS MACROS OF THE FORM  
BLKNAM\$X FLDNAM WHERE X IS B, W, L FOR FIELD WIDTHS OF  
1, 2, 4 BYTES RESPECTIVELY.

IN ADDITION A MACRO OF FORM INIT BLKNAM IS  
GENERATED FOR STATIC INITIALIZATION OF BLOCK. FOR EXAMPLE  
INITIAL (INIT\_BLKNAM(VALUE1,VALUE2,....VALUEN))

GENERATE A MACRO TO REFERENCE THE START OF THE BLOCK

MACRO %NAME(%STRING(BLKNAM),'\$C\_START') =  
0,0,0,%QUOTE%;

GENERATE THE ACCESSING MACROS FOR ALL THE FIELDS OF THE BLOCK

YYYYYYYBBF(BLKNAM,0,%REMAINING);

DEFINE A GLOBAL LITERAL (FOR LINK TIME ERROR DETECTION) WHOSE  
VALUE IS NUMBER OF BYTES IN BLOCK

GLOBAL LITERAL %NAME(%STRING(BLKNAM),'\$C\_SIZE') = YYYYYYSIZE : WEAK;

NOW DEFINE A MACRO THAT CAN BE USED TO STATICALLY INITIALIZE  
ALL THE FIELDS OF THIS BLOCK

MACRO NAME IS INIT\_BLKNAM

YYCREBINIMACRO(%NAME('INIT\_',%STRING(BLKNAM)), (ZZZBBF(%REMAINING)), (XXXBBF(%REMAINING)))%,

AUXILIARY MACROS:-

YYCREBINIMACRO(NAME, FORMAL, BODY) =  
MACRO NAME FORMAL =%REMOVE BODY%QUOTE%.

ZZZBBF[FLDNAM, FLDWIDTH] = %NAME(%STRING(FLDNAM),'\_FML')%,

XXXBBF[FLDNAM, FLDWIDTH] = %IF FLDWIDTH EQL 1 %THEN BYTE

```

        %ELSE %IF FLDWIDTH EQL 2 %THEN WORD
        %ELSE LONG
        %FI
    %FI
        (%NAME(%STRING(FLDNAM),'_FHL'))%,
!
YYYFLDPREFIX(WIDTH) = %IF WIDTH EQL 1
    %THEN B
    %ELSE %IF WIDTH EQL 2
        %THEN W
        %ELSE %IF WIDTH EQL 4
            %THEN L
            %ELSE T
            %FI
        %FI
    %FI%,
!
YYYYYYYYBBF( BLKNAM, OFF, NAM, BYTS)[ ] =
    MACRO %NAME( %STRING(BLKNAM), '$', %STRING(YYYFLDPREFIX(BYTS)), '-', %STRING(NAM)) =
        OFF, 0, MIN(BYTS*8, 32), 0 %QUOTE%;
    %ASSIGN(YYYYYYYYSIZE,OFF+BYTS)
    YYYYYYYYBBF( BLKNAM, OFF+BYTS, %REMAINING)%;

```

```

DEFINE ACCESS MACROS FOR I-SECTION GENERATION CONTROL
TABLE ENTRIES

```

```

GENBLOCKFIELDS(ISC,
    MASK,2,
    MATCH,2,
    CODE,1,
    SIZE,1,
    PFC,1,
    FLAGS,1,
    MATCTL,1);
! NAMES ARE ISCSX YY...
! P-SECTION AND MASK
! P-SECTION ATTRIBUTE MATCH
! I-SECT TYPE
! ISD SIZE
! PAGE FAULT CLUSTER
! ISD FLAGS
! MATCH CONTROL FIELD OF ISD

```

```
MACRO
```

```
ISECTDEF(NAME,TYPE,MATCH)=
```

```
THIS MACRO DEFINES AN IMAGE SECTION NAME AND THE PARAMETERS
OF SUCH AN IMAGE SECTION.
```

```
NAME IS A NAME TO BE ASSOCIATED WITH THE IMAGE SECTION
TYPE IS ONE OF THE GENERIC TYPES LISTED ABOVE.
MATCH IS THE PATTERN OF P-SECTION ATTRIBUTES TO BE
MATCHED IN THE SEARCH
```

```
LITERAL
```

```

%NAME(%STRING(NAME),'_MATCH') = MATCH OR GPSSM_REL,
%NAME(%STRING(NAME),'_ISDCODE') = %NAME(%STRING(TYPE),'_CODE'),
%NAME(%STRING(NAME),'_ISDSIZ') = %NAME(%STRING(TYPE),'_SIZE'),
%NAME(%STRING(NAME),'_PFC') = %NAME(%STRING(TYPE),'_PFC'),
%NAME(%STRING(NAME),'_GBL') = %NAME(%STRING(TYPE),'_GBL'),
%NAME(%STRING(NAME),'_CRF') = %IF(%NAME(%STRING(NAME),'_MATCH')
    AND (GPSSM_WRT OR
    GPSSM_SHR) EQL GPSSM_WRT
    %THEN TRUE
! SET MATCH MASK ALWAYS RELOCATABLE
! ISD TYPE CODE
! SIZE OF ISD
! AND CLUSTER SIZE
! GLOBAL FLAG
! SET THE COPY ON REFERENCE
! FLAG IFF P-SECTION
! WRITABLE AND NOT SHAREABLE
! IS WRITABLE

```

```

                                %ELSE FALSE
                                %FI
%NAME(%STRING(NAME),'_DZRO') = %NAME(%STRING(TYPE),'_DZRO'),      ! DEMAND ZERO FLAG
%NAME(%STRING(NAME),'_WRT') = %IF(%NAME(%STRING(NAME),'_MATCH')   ! WRITABLE FLAG IF P-SECTION
                                AND GPSSM_WRT) NEQ 0              ! IS WRITABLE
                                %THEN TRUE
                                %ELSE %IF %NAME(NAME,'_DZRO')      ! OR IF THE IMAGE SECTION
                                %THEN TRUE                          ! IS A DEMAND ZERO
                                %ELSE FALSE
                                %FI
                                %FI
%NAME(%STRING(NAME),'_MTCLFLD') = ISD$K_MATALL;%                ! MATCH CONTROL FIELD OF ISD'S

```

```
ISGENTBL[ISNAME,SRCHMASK] = INIT_ISC(
```

THIS MACRO GENERATES ISECTION GENERATION CONTROL TABLE ENTRIES, INITIALIZED WITH PARAMETERS ASSOCIATED WITH THE PARTICULAR IMAGE SECTIONS.

ISNAME IS THE NAME OF THE IMAGE SECTION TYPE  
SRCHMASK IS THE MASK OF P-SECTION ATTRIBUTES TO BE CONSIDERED IN SEARCH FOR PATTERNS

```

SRCHMASK,                                ! STORE AND MASK
%NAME(%STRING(ISNAME),'_MATCH'),          ! SET MATCH PATTERN
%NAME(%STRING(ISNAME),'_ISDCODE'),        ! CODE FOR ISD
%NAME(%STRING(ISNAME),'_ISDSIZ'),        ! SIZE OF ISD
%NAME(%STRING(ISNAME),'_PFC'),           ! AND PAGE FAULT CLUSTER
(ISD$M_GBL*%NAME(%STRING(ISNAME),'_GBL')) OR ! SET ISECT
(ISD$M_WRT*%NAME(%STRING(ISNAME),'_WRT')) OR ! DESCRIPTOR
(ISD$M_CRF*%NAME(%STRING(ISNAME),'_CRF')) OR ! FLAGS
(ISD$M_DZRO*%NAME(%STRING(ISNAME),'_DZRO')),
%NAME(%STRING(ISNAME),'_MTCLFLD')
)%

```

THIS MACRO GENERATES AN IMAGE SECTION GENERATION CONTROL TABLE.

```

ISGENTBL(TBLNAM)[]=
%ASSIGN(NUM_ISECTS,(%LENGTH-1)/2)        ! GET NUMBER IN
OWN                                       ! TABLE AND SET
%NAME(%STRING(TBLNAM),'_ISECTS') :       ! IN VARIABLE
    BYTE INITIAL (NUM_ISECTS),
%NAME(%STRING(TBLNAM),'_ISGENTBL') :    ! GENERATE AND
    BLOCKVECTOR[NUM_ISECTS,ISC$C_SIZE,BYTE] ! INITIALIZE ALL
    INITIAL(ISGENTBL[ENT(%REMAINING)]);% ! ENTRIES OF TABLE

```

DEFINE INTERESTING IMAGE SECTION TYPES

```

ISECTDEF(_U_R,NORMAL,GPSSM_REL);        ! USER RO DATA
ISECTDEF(_U_RW,NORMAL,GPSSM_WRT OR GPSSM_REL); ! USER R/W DATA
ISECTDEF(_U_X,NORMAL,GPSSM_EXE OR GPSSM_REL); ! USER EXECUTE ONLY
ISECTDEF(_U_XW,NORMAL,GPSSM_WRT OR      ! USER MODIFIABLE
                                GPSSM_EXE OR GPSSM_REL); ! EXECUTABLE CODE
ISECTDEF(_U_RV,NORMAL,GPSSM_REL OR GPSSM_VEC); ! RO VECTOR
ISECTDEF(_U_RWV,NORMAL,GPSSM_WRT OR GPSSM_VEC ! RW VECTOR
                                OR GPSSM_REC);

```



```

ISECTDEF(_U_X_V, NORMAL, GPSSM_EXE OR GPSSM_REL           ! EXE ONLY VECTOR
          OR GPSSM_VEC);
ISECTDEF(_U_XWV, NORMAL, GPSSM_WRT OR GPSSM_EXE           ! MODIFIABLE
          OR GPSSM_VEC OR GPSSM_REL);                   ! EXECUTABLE VECTOR
ISECTDEF(STACK, STACK, 0);                               ! USER STACK
ISECTDEF(S_ALONE, NORMAL, GPSSM_REL);                   ! STAND ALONE IMAGE

    THE SET FOR SHAREABLE IMAGES

ISECTDEF(_S_R, SHRFXD, _U_R_MATCH OR GPSSM_SHR);        ! SHAREABLE NON-PIC IMAGE SECTIONS
ISECTDEF(_S_RQ, SHRFXD, _U_RQ_MATCH OR GPSSM_SHR);
ISECTDEF(_S_X, SHRFXD, _U_X_MATCH OR GPSSM_SHR);
ISECTDEF(_S_XQ, SHRFXD, _U_XQ_MATCH OR GPSSM_SHR);

ISECTDEF(_R, PRVFXD, _U_R_MATCH);                       ! PRIVATE NON-PIC SET
ISECTDEF(_RQ, PRVFXD, _U_RQ_MATCH);
ISECTDEF(_X, PRVFXD, _U_X_MATCH);
ISECTDEF(_XQ, PRVFXD, _U_XQ_MATCH);

ISECTDEF(_SPR, SHRPIX, _U_R_MATCH OR GPSSM_SHR OR GPSSM_PIX); ! SHAREABLE PIC SET
ISECTDEF(_SPRQ, SHRPIX, _U_RQ_MATCH OR GPSSM_SHR OR GPSSM_PIX);
ISECTDEF(_SPX, SHRPIX, _U_X_MATCH OR GPSSM_SHR OR GPSSM_PIX);
ISECTDEF(_SPXQ, SHRPIX, _U_XQ_MATCH OR GPSSM_SHR OR GPSSM_PIX);

ISECTDEF(_PR, PRVPIC, _U_R_MATCH OR GPSSM_PIX);         ! PRIVATE PIC SET
ISECTDEF(_PRQ, PRVPIC, _U_RQ_MATCH OR GPSSM_PIX);
ISECTDEF(_PX, PRVPIC, _U_X_MATCH OR GPSSM_PIX);
ISECTDEF(_PXQ, PRVPIC, _U_XQ_MATCH OR GPSSM_PIX);

ISECTDEF(_S_R_V, SHRFXD, _U_R_V_MATCH OR GPSSM_SHR);    ! SHAREABLE NON-PIC VECTOR SET
ISECTDEF(_S_RQV, SHRFXD, _U_RQV_MATCH OR GPSSM_SHR);
ISECTDEF(_S_X_V, SHRFXD, _U_X_V_MATCH OR GPSSM_SHR);
ISECTDEF(_S_XQV, SHRFXD, _U_XQV_MATCH OR GPSSM_SHR);

ISECTDEF(_R_V, PRVFXD, _U_R_V_MATCH);                   ! PRIVATE NON-PIC VECTOR SET
ISECTDEF(_RQV, PRVFXD, _U_RQV_MATCH);
ISECTDEF(_X_V, PRVFXD, _U_X_V_MATCH);
ISECTDEF(_XQV, PRVFXD, _U_XQV_MATCH);

ISECTDEF(_SPR_V, SHRPIX, _U_R_V_MATCH OR GPSSM_SHR OR GPSSM_PIX); ! SHAREABLE PIC VECTOR SET
ISECTDEF(_SPRQV, SHRPIX, _U_RQV_MATCH OR GPSSM_SHR OR GPSSM_PIX);
ISECTDEF(_SPX_V, SHRPIX, _U_X_V_MATCH OR GPSSM_SHR OR GPSSM_PIX);
ISECTDEF(_SPXQV, SHRPIX, _U_XQV_MATCH OR GPSSM_SHR OR GPSSM_PIX);

ISECTDEF(_PR_V, PRVPIC, _U_R_V_MATCH OR GPSSM_PIX);    ! PRIVATE PIC VECTOR SET
ISECTDEF(_PRQV, PRVPIC, _U_RQV_MATCH OR GPSSM_PIX);
ISECTDEF(_PX_V, PRVPIC, _U_X_V_MATCH OR GPSSM_PIX);
ISECTDEF(_PXQV, PRVPIC, _U_XQV_MATCH OR GPSSM_PIX);

```

This image displays a grid of 100 small thumbnail images, each representing a different software utility or its output. The thumbnails are arranged in a 10x10 grid. Many of the thumbnails contain text-based data, such as lists, tables, and command-line outputs. Some thumbnails are labeled with the names of the utilities they represent, including:

- LINKER LIS
- STRUNDEQ LIS
- STRPOSEXT LIS
- STRREPLAC LIS
- STRSRCHM LIS
- STRRIGHT LIS
- STRTRIM LIS
- LINK MAP
- STRUPCASE LIS
- STRTRANSL LIS
- DATBAS MDL
- STRPREFIX LIS
- STRPREFIX LIS
- LINKER
- PREFIX REQ
- ISDSORT LIS
- TIRAU REQ
- ISGENC REQ
- DATBAS LIS

The thumbnails show various types of data, including lists of files, command-line outputs, and graphical representations of data. The overall appearance is that of a comprehensive catalog of software utilities available on the VAX/VMS system.