


```
PPPPPPPP  RRRRRRRR  EEEEEEEEE  FFFFFFFF  IIIIII  XX  XX
PPPPPPPP  RRRRRRRR  EEEEEEEEE  FFFFFFFF  IIIIII  XX  XX
PP  PP  RR  RR  EE  FF  II  XX  XX
PP  PP  RR  RR  EE  FF  II  XX  XX
PP  PP  RR  RR  EE  FF  II  XX  XX
PP  PP  RR  RR  EE  FF  II  XX  XX
PPPPPPPP  RRRRRRRR  EEEEEEEEE  FFFFFFFF  IIIIII  XX  XX
PPPPPPPP  RRRRRRRR  EEEEEEEEE  FFFFFFFF  IIIIII  XX  XX
PP  RR  RR  EE  FF  II  XX  XX
PP  RR  RR  EE  FF  II  XX  XX
PP  RR  RR  EE  FF  II  XX  XX
PP  RR  RR  EE  FF  II  XX  XX
PP  RR  RR  EE  FF  II  XX  XX
PP  RR  RR  EEEEEEEEE  FF  IIIIII  XX  XX
PP  RR  RR  EEEEEEEEE  FF  IIIIII  XX  XX
                                     ....
                                     ....
                                     ....
                                     ....
```

```
RRRRRRRR  EEEEEEEEE  QQQQQQ
RRRRRRRR  EEEEEEEEE  QQQQQQ
RR  RR  EE  QQ  QQ
RR  RR  EE  QQ  QQ
RR  RR  EE  QQ  QQ
RRRRRRRR  EEEEEEEEE  QQ  QQ  QQ
RRRRRRRR  EEEEEEEEE  QQ  QQ  QQ
RR  RR  EE  QQ  QQ  QQ
RR  RR  EE  QQ  QQ  QQ
RR  RR  EE  QQ  QQ  QQ
RR  RR  EE  QQ  QQ  QQ
RR  RR  EEEEEEEEE  QQQQ  QQ
RR  RR  EEEEEEEEE  QQQQ  QQ
```



```

++
FUNCTIONAL DESCRIPTION:

```

```

    THIS REQUIRE FILE DEFINES SOME GENERAL, LOCAL VALUES AND MACROS
    WHICH CANNOT OR ARE BEST NOT DEFINED GLOBALLY.
--

```

```

MACRO

```

```

SHORT = UNSIGNED(6)%,           ! SHORT LITERAL ATTRIBUTE
BYTLIT = UNSIGNED(8)%,         ! UNSIGNED BYTE ATTRIBUTE
WORDLIT = UNSIGNED(16)%,       ! UNSIGNED WORD ATTRIBUTE

```

```

MACRO THAT DESCRIBES A STRING

```

```

STRINGDESC(STRING) = %CHARCOUNT(STRING), UPLIT(%ASCII STRING)%,

```

```

MACRO TO GENERATE A QUADWORD STRING DESCRIPTOR

```

```

DESCRIPTOR(STRING) = VECTOR[2] INITIAL(STRINGDESC(STRING))%;

```

```

MACRO TO GENERATE A COUNTED STRING AS A BYTE VECTOR

```

```

CSTRING(STRING) = UPLIT BYTE(%CHARCOUNT(STRING),STRING)%,

```

```

MACRO TO ALLOCATE AN N-CHARACTER STORAGE VECTOR

```

```

CH$SEQUENCE(N) = VECTOR[CH$ALLOCATION(N)]%;

```

```

A SET OF MACROS FOR GENERATING STRUCTURE REFERENCE MACROS

```

```

COMPILETIME

```

```

XXXXXXXXSIZE = 0;

```

```

MACRO

```

```

BYTEBLOCKFIELDS(BLKNAM)[] =
MACRO %NAME(%STRING(BLKNAM), '%$C_START') =
    0,0,0,0,%QUOTE%;
XXXXXXXXBBF(BLKNAM,0,%REMAINING);
GLOBAL LITERAL %NAME(%STRING(BLKNAM), '%$C_SIZE') = XXXXXXXXSIZE : WEAK%,

```

```

XXXXXXXXBBF( BLKNAM, OFF, NAM, BYTS)[] =
MACRO %NAME( %STRING(BLKNAM), '%$', %STRING(NAM)) = OFF, 0, MIN(BYTS*8, 32), 0 %QUOTE %;
%ASSIGN(XXXXXXXXSIZE,OFF+BYTS)
XXXXXXXXBBF( BLKNAM, OFF+BYTS, %REMAINING)%;

```

```

LITERAL

```

```

SYM$C MAXLNG = 15,           ! MAXIMUM LENGTH OF STRINGS
CONTROL_REGION = %X'40000000', ! START OF CONTROL REGION
SYSTEM_SPACE = %X'80000000', ! START OF SYSTEM SPACE
MAX_ADDRESS = %X'C0000000'; ! MAXIMUM VIRTUAL ADDRESS

```

```

STRUCTURE

```

```

QUADVECTOR[I;N] =           ! A VECTOR OF N QUADWORDS
    [ N * 8 ]

```

(QUADVECTOR + 8*I);

THE FOLLOWING MACROS GENERATE ACCESSING MACROS FOR BLOCK
STRUCTURES WHICH HAVE FIELDS OF BYTE, WORD OR LONG WIDTH.

COMPILETIME

YYYYYYYYSIZE = 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') =
0,0,0,0%QUOTE%;

GENERATE THE ACCESSING MACROS FOR ALL THE FIELDS OF THE BLOCK

YYYYYYYYBBF(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') = YYYYYYYYYSIZE : 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), '_FML'))%,
!
! YYYFLDPREFIX(WIDTH) = %IF WIDTH EQL 1
!                       %THEN B
!                       %ELSE %IF WIDTH EQL 2
!                               %THEN W
!                               %ELSE L
!                               %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)%;
```

