

0001 0
0002 0
0003 0
0004 0
0005 0
0006 0
0007 0
0008 0
0009 0
0010 0
0011 0
0012 0
0013 0
0014 0
0015 0
0016 0
0017 0
0018 0
0019 0
0020 0
0021 0
0022 0
0023 0
0024 0
0025 0
0026 0
0027 0
0028 0
0029 0
0030 0
0031 0
0032 0
0033 0
0034 0
0035 0
0036 0
0037 0
0038 0
0039 0
0040 0
0041 0
0042 0

STRUCDEF -- DECLARATION FILE FOR DATA STRUCTURE DEFINITION
AND ACCESS MACROS USED IN THE VAX DEBUGGER

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.
*

WRITTEN BY
Bert Beander August, 1981.

MODULE FUNCTION:
This REQUIRE file contains all macros used in defining and accessing
data structures (BLISS BLOCKs) in the VAX Debugger. These symbolic
names should always be used in BLISS Field-References.

DATA STRUCTURE DEFINITION AND ACCESS

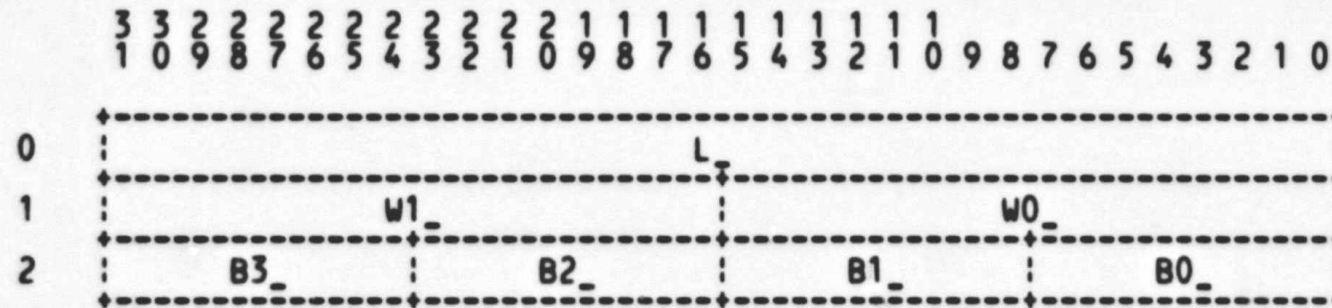
The following macros must be used in defining field names for all data structures in the Debugger. These macros supply the position, size, and sign-extension values when used in FIELD declarations for BLOCK and BLOCKVECTOR data structures. The various generic forms (as specified by the letters in the names) are as follows:

- A Materialized address
- L Longword
- W Zero-extended word
- B Zero-extended byte
- V Zero-extended bit field
- SW Sign-extended word
- SB Sign-extended byte
- SV Sign-extended bit field

The "A" form should be used whenever the field being defined is such that only the address of the field may be materialized in a structure reference; that is, fetch and store operations on the field are not valid. An example of such a field is an ASCII string.

Each of the "V" and "SV" forms take one or two parameters. The first parameter is the bit position within the longword (or byte) and the second is the field size in bits. The second parameter is optional; if omitted, it defaults to 1. Thus V_(5) means bit 5 while V_(5,3) means the 3-bit field starting at bit 5 and ending at bit 7. Bit positions are counted from the low-order (least significant) end of the longword, starting at zero.

The following data structure picture shows the locations of the various fields that can be specified. Note how the bit positions are numbered along the top of the illustration.



 MACRO

- A_ = 0, 0, 0 % ! Address of a longword
- A0_ = 0, 0, 0 % ! Address of byte 0
- A1_ = 8, 0, 0 % ! Address of byte 1
- A2_ = 16, 0, 0 % ! Address of byte 2

0043 0
 0044 0
 0045 0
 0046 0
 0047 0
 0048 0
 0049 0
 0050 0
 0051 0
 0052 0
 0053 0
 0054 0
 0055 0
 0056 0
 0057 0
 0058 0
 0059 0
 0060 0
 0061 0
 0062 0
 0063 0
 0064 0
 0065 0
 0066 0
 0067 0
 0068 0
 0069 0
 0070 0
 0071 0
 0072 0
 0073 0
 0074 0
 0075 0
 0076 0
 0077 0
 0078 0
 0079 0
 0080 0
 0081 0
 0082 0
 0083 0
 0084 0
 0085 0
 0086 0
 0087 0
 0088 0
 0089 0
 0090 0
 0091 0
 0092 0
 0093 0
 0094 0
 0095 0
 0096 0
 0097 0
 0098 0
 0099 0

```
0100 0 A3_ = 24, 0, 0 %, ! Address of byte 3
0101 0
0102 0 L_ = 0, 32, 0 %, ! Longword
0103 0 W_ = 0, 16, 0 %, ! Word, zero-extended
0104 0 B_ = 0, 8, 0 %, ! Byte, zero-extended
0105 0
0106 0 W0_ = 0, 16, 0 %, ! Word 0 zero-extended
0107 0 W1_ = 16, 16, 0 %, ! Word 1 zero-extended
0108 0
0109 0 B0_ = 0, 8, 0 %, ! Byte 0 zero-extended
0110 0 B1_ = 8, 8, 0 %, ! Byte 1 zero-extended
0111 0 B2_ = 16, 8, 0 %, ! Byte 2 zero-extended
0112 0 B3_ = 24, 8, 0 %, ! Byte 3 zero-extended
0113 0
0114 0 V_(P,S) = P, %IF %NULL(S) %THEN 1 %ELSE S %FI, 0 %, ! Unsigned bit field
0115 0
0116 0 V0_(P,S) = P, %IF %NULL(S) %THEN 1 %ELSE S %FI, 0 %, ! Bits in B0_
0117 0 V1_(P,S) = (P+8), %IF %NULL(S) %THEN 1 %ELSE S %FI, 0 %, ! Bits in B1_
0118 0 V2_(P,S) = (P+16), %IF %NULL(S) %THEN 1 %ELSE S %FI, 0 %, ! Bits in B2_
0119 0 V3_(P,S) = (P+24), %IF %NULL(S) %THEN 1 %ELSE S %FI, 0 %, ! Bits in B3_
0120 0
0121 0 SW_ = 0, 16, 1 %, ! Word, sign-extended
0122 0 SB_ = 0, 8, 1 %, ! Byte, sign-extended
0123 0
0124 0 SW0_ = 0, 16, 1 %, ! Word 0 sign-extended
0125 0 SW1_ = 16, 16, 1 %, ! Word 1 sign-extended
0126 0
0127 0 SB0_ = 0, 8, 1 %, ! Byte 0 sign-extended
0128 0 SB1_ = 8, 8, 1 %, ! Byte 1 sign-extended
0129 0 SB2_ = 16, 8, 1 %, ! Byte 2 sign-extended
0130 0 SB3_ = 24, 8, 1 %, ! Byte 3 sign-extended
0131 0
0132 0 SV_(P,S) = P, %IF %NULL(S) %THEN 1 %ELSE S %FI, 1 %, ! Signed bit field
0133 0
0134 0 SV0_(P,S) = P, %IF %NULL(S) %THEN 1 %ELSE S %FI, 1 %, ! Bits in B0_
0135 0 SV1_(P,S) = (P+8), %IF %NULL(S) %THEN 1 %ELSE S %FI, 1 %, ! Bits in B1_
0136 0 SV2_(P,S) = (P+16), %IF %NULL(S) %THEN 1 %ELSE S %FI, 1 %, ! Bits in B2_
0137 0 SV3_(P,S) = (P+24), %IF %NULL(S) %THEN 1 %ELSE S %FI, 1 %, ! Bits in B3_
0138 0
0139 0
0140 0 ! END OF STRUCDEF.REQ
```

COMMAND QUALIFIERS

BLISS/LIBRARY=LIB\$:STRUCDEF.L32/LIST=LIS\$:STRUCDEF.LIS SRC\$:STRUCDEF.REQ

: Run Time: 00:01.3
: Elapsed Time: 00:03.6
: Lines/CPU Min: 6412
: Lexemes/CPU-Min: 25740
: Memory Used: 12 pages

; Library Precompilation Complete

DEL

LIT

MAC

F

MAC

1

1

