


```

000000  TTTTTTTTTT  SSSSSSSS  PPPPPPPP  000000  WW  WW  CCCCCCCC  DDDDDDDD  CCCCCCCC
000000  TTTTTTTTTT  SSSSSSSS  PPPPPPPP  000000  WW  WW  CCCCCCCC  DDDDDDDD  CCCCCCCC
00 00  TT  SS  SS  SS  PP  PP  PP  00 00  WW  WW  CC  DD  DD  CC
00 00  TT  SS  SS  SS  PP  PP  PP  00 00  WW  WW  CC  DD  DD  CC
00 00  TT  SS  SS  SS  PP  PP  PP  00 00  WW  WW  CC  DD  DD  CC
00 00  TT  SS  SS  SS  PP  PP  PP  00 00  WW  WW  CC  DD  DD  CC
00 00  TT  SS  SS  SS  PP  PP  PP  00 00  WW  WW  CC  DD  DD  CC
00 00  TT  SS  SS  SS  PP  PP  PP  00 00  WW  WW  CC  DD  DD  CC
00 00  TT  SS  SS  SS  PP  PP  PP  00 00  WWW WWW  CC  DD  DD  CC
00 00  TT  SS  SS  SS  PP  PP  PP  00 00  WWW WWW  CC  DD  DD  CC
000000  TTT  SSSSSSSS  PPP  000000  WW  WW  CCCCCCCC  DDDDDDDD  CCCCCCCC
000000  TTT  SSSSSSSS  PPP  000000  WW  WW  CCCCCCCC  DDDDDDDD  CCCCCCCC

```

```

LL  IIIIII  SSSSSSSS
LL  IIIIII  SSSSSSSS
LL  II  SS
LL  II  SS
LL  II  SS
LL  II  SS
LL  II  SSSSSS
LL  II  SSSSSS
LL  II  SS
LL  II  SS
LL  II  SS
LLLLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLLLL  IIIIII  SSSSSSSS

```

(2) 46
(3) 78

DECLARATIONS
OTSSPOWCDCD_R3 - D COMPLEX*16 ** D COMPLEX*16

```
0000 1 .TITLE OTSSPOWCDCD - D COMPLEX*16 ** D COMPLEX*16 routine
0000 2 .IDENT /1-002/ ; File: OTSPOWCDC.MAR Edit: SBL1002
0000 3
0000 4
0000 5 *****
0000 6
0000 7 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
0000 8 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
0000 9 * ALL RIGHTS RESERVED. *
0000 10
0000 11 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0000 12 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
0000 13 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
0000 14 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0000 15 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
0000 16 * TRANSFERRED. *
0000 17
0000 18 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
0000 19 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
0000 20 * CORPORATION. *
0000 21
0000 22 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
0000 23 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
0000 24
0000 25
0000 26 *****
0000 27
0000 28
0000 29 :++
0000 30 : FACILITY: Language support library - user callable
0000 31
0000 32 : ABSTRACT:
0000 33
0000 34 : D COMPLEX*16 base to D COMPLEX*16 power giving D COMPLEX*16 result.
0000 35
0000 36 : ENVIRONMENT: User Mode, AST Reentrant
0000 37
0000 38 :--
0000 39 : AUTHOR: Steven B. Lionel, CREATION DATE: 20-July-1979
0000 40
0000 41 : MODIFIED BY:
0000 42
0000 43 : 1-001 - Original. Adapted from OTSSPOWCC version 1-003. SBL 20-Jul-1979
0000 44 : 1-002 - Use general mode addressing. SBL 30-Nov-1981
```

OTS
Syn
BAS
DON
EVE
EXP
MTH
MTH
OTS
OTS
POW
REC
SQL
SQL
UNC
PSE
--
_01
Pha
--
Ini
Con
Pas
Syn
Pas
Syn
Pse
Crc
Ass
The
317
The
219
0 p

Mac
--
_S
0
The

```
0000 46 .SBTTL DECLARATIONS
0000 47 :
0000 48 : INCLUDE FILES:
0000 49 :
0000 50 :
0000 51 :
0000 52 : EXTERNAL DECLARATIONS:
0000 53 :
0000 54 .DSABL GBL
0000 55 .EXTRN MTH$CDEXP ; Complex exponentiation
0000 56 .EXTRN MTH$CDLOG ; Complex logarithm
0000 57 .EXTRN OTSSMULCD_R3 ; Complex multiplication
0000 58 :
0000 59 : MACROS:
0000 60 :
0000 61 :
0000 62 :
0000 63 : EQUATED SYMBOLS:
0000 64 :
00000004 0000 65 base = 4 ; base input - by value
00000014 0000 66 exp = 20 ; exponent input - by value
0000 67 :
0000 68 : OWN STORAGE:
0000 69 :
0000 70 :
0000 71 :
0000 72 : PSECT DECLARATIONS:
0000 73 :
00000000 0000 74 .PSECT _OTSS$CODE PIC, USR, CON, REL, LCL, SHR, -
0000 75 EXE, RD, NOWRT, LONG
0000 76
```

```

0000 78      .SBTTL OTSS$POWCDCD_R3 - D COMPLEX*16 ** D COMPLEX*16
0000 79      :++
0000 80      : FUNCTIONAL DESCRIPTION:
0000 81      :
0000 82      : OTSS$POWCDCD_R3 evaluates the result of taking a complex base
0000 83      : to a complex power. The ANS FORTRAN X3.9-1978 standard defines
0000 84      : complex exponentiation as:
0000 85      :
0000 86      :  $x ** y = \text{CEXP}(y * \text{CLOG}(x))$ 
0000 87      :
0000 88      : where x and y are type D COMPLEX*16.
0000 89      :
0000 90      : The arguments of OTSS$POWCDCD_R3 are CALL BY VALUE.
0000 91      :
0000 92      : CALLING SEQUENCE:
0000 93      :
0000 94      : power.wdc.v = OTSS$POWCDCD_R3 (base.rdc.v, exponent.rdc.v)
0000 95      :
0000 96      : INPUT PARAMETERS:
0000 97      :
0000 98      : Both base and exponent are D COMPLEX*16 numbers, each consisting
0000 99      : of a D REAL*8 real part and a D REAL*8 imaginary part. Both are
0000 100     : CALL BY VALUE.
0000 101     :
0000 102     : IMPLICIT INPUTS:
0000 103     :
0000 104     : NONE
0000 105     :
0000 106     : OUTPUT PARAMETERS:
0000 107     :
0000 108     : NONE
0000 109     :
0000 110     : IMPLICIT OUTPUTS:
0000 111     :
0000 112     : NONE
0000 113     :
0000 114     : FUNCTION VALUE:
0000 115     :
0000 116     : The D COMPLEX*16 (REAL*8, REAL*8) result of taking the
0000 117     : COMPLEX base to the COMPLEX exponent power is returned
0000 118     : in registers R0-R3. This is a violation of the VAX
0000 119     : calling standard, but is excused for compiled code
0000 120     : support routines.
0000 121     :
0000 122     : SIDE EFFECTS:
0000 123     :
0000 124     : Modifies registers R0-R3.
0000 125     :
0000 126     : Possible error signals are:
0000 127     :
0000 128     : MTH$_INVARGMAT if base is (0.,0.).
0000 129     : MTH$_FLOOVEMAT if floating overflow occurs.
0000 130     : MTH$_SINCOSSIG if absolute value of the imaginary part of
0000 131     : (exponent * LOG(base)) > PI*2**30.
0000 132     : SSS$_ROPRAND  if reserved floating operand is fetched.
0000 133     :--
  
```

```

0000 0000 135 .ENTRY OTSSPOWCDCD_R3, ^M<> ; disable integer ovflo
      0002 136
      0002 137 MTH$FLAG_JACKET ; establish math error handler
6D 00000000'GF 9E 0002 MOVAB G^MTH$$JACKET_HND, (FP) ; set handler address to jacket
      0009 ; handler
      0009
      0009 138
      0009 139 :+
      0009 140 :+
      0009 141 :-
      5E 10 C2 0009 142 Get complex logarithm of base
      04 AC DF 000C 143 SUBL2 #16, SP ; return complex on stack
      04 AE 9F 000F 144 PUSHAL base(AP) ; address of base
00000000'GF 02 FB 0012 145 PUSHAB 4(SP) ; address of result
      0019 146 CALLS #2, G^MTH$CDLOG ; (SP) gets LOG(base)
      0019 147 :+
      0019 148 CLOG(base) is at (SP). Multiply by exponent.
      0019 149 Do multiplication out of line.
      0019 150 :-
      7E 1C AC 7D 0019 151 MOVQ exp+8(AP), -(SP) ; Put exponent on stack
      7E 14 AC 7D 001D 152 MOVQ exp(AP), -(SP) ; CLOG(base) is already there!
00000000'GF 08 FB 0021 153 CALLS #8, G^OTSS$MULCD_R3 ; R0-R3 gets CLOG(base) * exp
      0028 154
      0028 155 :+
      0028 156 :-
      0028 157 :-
      0028 158 Now compute CEXP(product)
      7E 52 7D 0028 159 MOVQ R2, -(SP) ; put product on stack
      7E 50 7D 002B 160 MOVQ R0, -(SP)
      5E 10 C2 002E 161 SUBL2 #16, SP ; Make room for result
      10 AE 9F 0031 162 PUSHAB 16(SP) ; Address of product
      04 AE 9F 0034 163 PUSHAB 4(SP) ; Address of result
00000000'GF 02 FB 0037 164 CALLS #2, G^MTH$CDEXP ; Result is at (SP)
      50 8E 7D 003E 165 MOVQ (SP)+, R0 ; Pop result into R0-R3
      52 8E 7D 0041 166 MOVQ (SP)+, R2
      04 0044 167
      04 0044 168 RET ; all done, exit
      0045 169 .END
    
```

OTSSPOWCDCD
Symbol table

- D COMPLEX*16 ** D COMPLEX*16 routine F 9

16-SEP-1984 01:55:29
6-SEP-1984 11:27:47

VAX/VMS Macro V04-00
[MTHRTL.SRC]OTSSPOWCDC.MAR;1

Page 5
(4)

```

BASE          = 00000004
EXP           = 00000014
MTH$$JACKET_HND ***** X 01
MTH$CDEXP    ***** X 00
MTH$CDLOG    ***** X 00
OTSSMULCD_R3 ***** X 00
OTSSPOWCDCD_R3 00000000 RG 01

```

```

+-----+
! Psect synopsis !
+-----+

```

PSECT name	Allocation	PSECT No.	Attributes	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE
ABS	00000000 (0.)	00 (0.)	NOPIC	USR								
_OTSSCODE	00000045 (69.)	01 (1.)	PIC	USR	CON	REL	LCL	SHR	EXE	RD	NOWRT	NOVEC

```

+-----+
! Performance indicators !
+-----+

```

Phase	Page faults	CPU Time	Elapsed Time
Initialization	30	00:00:00.10	00:00:00.78
Command processing	112	00:00:00.73	00:00:03.32
Pass 1	79	00:00:00.59	00:00:02.95
Symbol table sort	0	00:00:00.00	00:00:00.01
Pass 2	44	00:00:00.48	00:00:02.02
Symbol table output	2	00:00:00.01	00:00:00.01
Psect synopsis output	3	00:00:00.03	00:00:00.05
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	272	00:00:01.96	00:00:09.15

The working set limit was 750 pages.
2559 bytes (5 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 7 non-local and 0 local symbols.
229 source lines were read in Pass 1, producing 11 object records in Pass 2.
1 page of virtual memory was used to define 1 macro.

```

+-----+
! Macro library statistics !
+-----+

```

Macro library name	Macros defined
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS\$:OTSSPOWCDC/OBJ=OBJ\$:OTSSPOWCDC MSRC\$:MTHJACKET/UPDATE=(ENH\$:MTHJACKET)+MS

0264 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

The image displays a grid of 100 small panels, each containing technical diagrams and data. The panels are arranged in a 10x10 grid. Many panels have labels such as 'OTSMULCD LIS', 'OTSPOWGC LIS', 'OTSDIUC LIS', 'OTSPOWDD LIS', 'OTSPOWCC LIS', 'OTSPOWCJ LIS', 'MHTAN LIS', 'MTHVECTOR LIS', 'OTSDIUCG LIS', 'OTSPOWCJ LIS', 'OTSPOWDLJ LIS', 'MHTANH LIS', 'OTSMULCG LIS', 'OTSPOWCGJ LIS', 'OTSPOWDJ LIS', 'OTSDIUCD LIS', and 'OTSPOWDC LIS'. The diagrams include various charts, tables, and flowcharts, representing technical specifications or data for different components or systems.