

.....

[illegible]


```
1 0001 0 MODULE STR$COMPARE ( ! Compare 2 strings with blank fill
2 0002 0
3 0003 0 IDENT = '1-006' ! File: STR$COMPARE.B32 Edit: RKR1006
4 0004 0
5 0005 0 ) =
6 0006 1 BEGIN
7 0007 1
8 0008 1
9 0009 1 *****
10 0010 1 *
11 0011 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
12 0012 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
13 0013 1 * ALL RIGHTS RESERVED.
14 0014 1 *
15 0015 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
16 0016 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
17 0017 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
18 0018 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
19 0019 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
20 0020 1 * TRANSFERRED.
21 0021 1 *
22 0022 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
23 0023 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
24 0024 1 * CORPORATION.
25 0025 1 *
26 0026 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
27 0027 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
28 0028 1 *
29 0029 1 *
30 0030 1 *****
31 0031 1
32 0032 1
33 0033 1 ++
34 0034 1 FACILITY: String support library
35 0035 1
36 0036 1 ABSTRACT: This module takes 2 input strings of any supported class
37 0037 1 and dtype and returns a +1 if string1 > string2,
38 0038 1 a 0 if string1 = string2, or a -1 if string1 < string2.
39 0039 1
40 0040 1 ENVIRONMENT: User mode, AST level or not or mixed
41 0041 1
42 0042 1 AUTHOR: R. Will, CREATION DATE: 26-Mar-79
43 0043 1
44 0044 1 MODIFIED BY:
45 0045 1
46 0046 1 R. Will, 26-Mar-79 : VERSION 01
47 0047 1 1-001 - Original
48 0048 1 1-002 - String cleanup, change name to STR$, remove = length.
49 0049 1 RW 5-Nov-79
50 0050 1 1-003 - Use CH$COMPARE instead of CH$EQL. CH$EQL returns 1 if
51 0051 1 strings match, 0 otherwise which is not what this routine is
52 0052 1 supposed to do! SBL 1-Oct-1980
53 0053 1 1-004 - Extend routine to recognize additional classes of descriptors
54 0054 1 by using $STR$GET_LEN_ADDR to extract length and address
55 0055 1 of first byte of data from descriptors. Remove string
56 0056 1 interlocking code. RKR 13-APR-81
57 0057 1 1-005 - Speed up code. RKR 7-OCT-1981.
```

STR\$COMPARE
1-006

6 2
16-Sep-1984 01:33:01
14-Sep-1984 12:40:02

VAX-11 Bliss-32 V4.0-742
[LIBRTL.SRC]STR\$COMPARE.B32;1

Page (1) 2

```

: 58
: 59
: 60
: 61

0058 1 | 1-006 - Use SWITCHES ZIP to avoid some cross-jumping that would
0059 1 | normally occur in this module. RKR 18-NOV-1981.
0060 1 | --
0061 1 | <BLF/PAGE>
```



```

63 0062 1 |
64 0063 1 | SWITCHES:
65 0064 1 |
66 0065 1 |
67 0066 1 | SWITCHES ADDRESSING MODE
68 0067 1 | (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
69 0068 1 |
70 0069 1 | SWITCHES ZIP;
71 0070 1 |
72 0071 1 | LINKAGES
73 0072 1 |
74 0073 1 | REQUIRE 'RTLIN:STRLNK';          ! Linkage to LIB$ANALYZE_DESC_R3
75 0258 1 |
76 0259 1 |
77 0260 1 | TABLE OF CONTENTS:
78 0261 1 |
79 0262 1 |
80 0263 1 | FORWARD ROUTINE
81 0264 1 | STR$COMPARE; ! compare 2 strings, same contents with blank fill
82 0265 1 |
83 0266 1 |
84 0267 1 | INCLUDE FILES:
85 0268 1 |
86 0269 1 |
87 0270 1 | REQUIRE 'RTLIN:RTLPSECT';      ! Declare PSECTS code
88 0365 1 | REQUIRE 'RTLIN:STRMACROS';    ! use string macros to write code
89 1281 1 | LIBRARY 'RTLSTARLE';          ! STARLET library for macros and symbols
90 1282 1 |
91 1283 1 |
92 1284 1 | MACROS: NONE
93 1285 1 |
94 1286 1 |
95 1287 1 |
96 1288 1 | EQUATED SYMBOLS: NONE
97 1289 1 |
98 1290 1 |
99 1291 1 |
100 1292 1 | PSECT DECLARATIONS
101 1293 1 |
102 1294 1 |
103 1295 1 | DECLARE_PSECTS (STR);
104 1296 1 |
105 1297 1 |
106 1298 1 | OWN STORAGE: NONE
107 1299 1 |
108 1300 1 |
109 1301 1 |
110 1302 1 | EXTERNAL REFERENCES
111 1303 1 |
112 1304 1 | NONE
```

```
114 1305 1 GLOBAL ROUTINE STR$COMPARE ( ! compare with blank fill
115 1306 1
116 1307 1     STRING1, ! pointer to 1st string descriptor
117 1308 1     STRING2 ! pointer to 2nd string descriptor
118 1309 1
119 1310 1 ) : =
120 1311 1
121 1312 1 ++
122 1313 1 FUNCTIONAL DESCRIPTION:
123 1314 1
124 1315 1     This routine takes two source strings of any supported
125 1316 1     dtype and class, compares their contents with blank fill
126 1317 1     for the shorter string and returns
127 1318 1     a -1 if string1 < string2, 0 if both are the same with
128 1319 1     blank fill or a 1 if string1 > string2.
129 1320 1
130 1321 1 FORMAL PARAMETERS:
131 1322 1
132 1323 1     STRING1.rt.dx     pointer to 1st string descriptor
133 1324 1     STRING2.rt.dx     pointer to 2nd string descriptor
134 1325 1
135 1326 1 IMPLICIT INPUTS:
136 1327 1
137 1328 1     NONE
138 1329 1
139 1330 1 IMPLICIT OUTPUTS:
140 1331 1
141 1332 1     NONE
142 1333 1
143 1334 1 ROUTINE VALUE:
144 1335 1
145 1336 1     MATCH.wl.v     -1 if string1 < string2
146 1337 1                 0 if both are the same with blank fill
147 1338 1                 1 if string1 > string2
148 1339 1
149 1340 1 SIDE EFFECTS:
150 1341 1
151 1342 1     May signal STR$_ILLSTRCLA on bad string class
152 1343 1
153 1344 1 --
154 1345 1
155 1346 2 BEGIN
156 1347 2
157 1348 2 MAP STRING1 : REF $STR$DESCRIPTOR;
158 1349 2 MAP STRING2 : REF $STR$DESCRIPTOR;
159 1350 2
160 1351 3 IF (.STRING1 [DSC$B_CLASS] LEQU DSC$K_CLASS_D AND
161 1352 3     .STRING2 [DSC$B_CLASS] LEQU DSC$K_CLASS_D )
162 1353 2 THEN
163 1354 3     BEGIN
164 1355 4     RETURN ( CH$COMPARE ( .STRING1 [DSC$W_LENGTH],
165 1356 4         .STRING1 [DSC$A_POINTER],
166 1357 4         .STRING2 [DSC$W_LENGTH],
167 1358 4         .STRING2 [DSC$A_POINTER],
168 1359 3         STR$_FILL_CHAR ) ) ;
169 1360 3
170 1361 3     END
```



```
171      ELSE      ! do it the hard way
172
173      BEGIN
174      LOCAL
175      S1_LENGTH,  ! Length of first string
176      S1_ADDR,    ! Address of 1st data byte of first string
177      S2_LENGTH,  ! Length of second string
178      S2_ADDR;    ! Address of 1st data byte of second string
179
180      !+
181      !- Compute the lengths and address of first bytes involved
182
183      $STR$GET_LEN_ADDR (STRING1, S1_LENGTH, S1_ADDR) ;
184
185      $STR$GET_LEN_ADDR (STRING2, S2_LENGTH, S2_ADDR) ;
186
187      !+
188      !- Return the value from CH$COMPARE
189
190      RETURN (CH$COMPARE (.S1_LENGTH, .S1_ADDR,      ! contents = ?
191                          .S2_LENGTH, .S2_ADDR,
192                          STR$K_FILL_CHAR));
193
194      END ;
195
196      END;
197
```

! End of STR\$COMPARE

.TITLE STR\$COMPARE
.IDENT \1-006\

.EXTRN STR\$ANALYZE_SDESC_R1

.PSECT _STR\$CODE,NOWRT, SHR, PIC,2

.ENTRY STR\$COMPARE, Save R2,R3,R4,R5,R6

MOVQ STRING1, R5

CLRL R1

CMPB 3(R5), #2

BGTRU 1\$

INCL R1

MOVL STRING2, R0

CMPB 3(R0), #2

BGTRU 1\$

MOVL #1, R4

CMPC5 (R5), @4(R5), #32, (R6), @4(R6)

BRB 6\$

BLBC R1, 2\$

MOVZWL (R5), S1_LENGTH

MOVL 4(R5), ST_ADDR

BRB 3\$

MOVL R5, R0

JSB STR\$ANALYZE_SDESC_R1

MOVL R0, R3

MOVL R1, R2

```
55      04      AC 7D 00002
02      03      A5 91 00008
19      1A 0000C
51      D6 0000E
50      08      AC D0 00010
02      03      A0 91 00014
0D      1A 00018
54      01 D0 0001A
66      20      04 65 2D 0001D
B5      04      B6      00023
3C      11 00025
09      51 E9 00027 1$:
53      65 3C 0002A
52      04      A5 D0 0002D
OF      11 00031
50      55 D0 00033 2$:
00000000G 00 16 00036
53      50 D0 0003C
52      51 D0 0003F
```

```
: 1305
: 1351
:
:
: 1352
:
: 1355
:
: 1375
:
```

STR\$COMPARE
1-006

K 2
16-Sep-1984 01:33:01
14-Sep-1984 12:40:02

VAX-11 Bliss-32 V4.0-742
[LIBRTL.SRC]STR\$COMPARE.B32;1

Page 6
(3)

STR
1-0

02	03	A6	91	00042	3\$:	CMPB	3(R6), #2	:	1377
		09	1A	00046		BGTRU	4\$	:	
50		66	3C	00048		MOVZWL	(R6), S2_LENGTH	:	
51	04	A6	D0	0004B		MOVL	4(R6), S2_ADDR	:	
		09	11	0004F		BRB	5\$	:	
50		56	D0	00051	4\$:	MOVL	R6, R0	:	
	00000000G	00	16	00054		JSB	STR\$ANALYZE_SDESC_R1	:	
54		01	D0	0005A	5\$:	MOVL	#1, R4	:	1383
62		53	2D	0005D		CMPC5	S1_LENGTH, (S1_ADDR), #32, S2_LENGTH, -	:	
		61		00062			(S2_ADDR)	:	
		03	1A	00063	6\$:	BGTRU	7\$	:	
54		01	D9	00065		SBWC	#1, R4	:	1364
50		54	D0	00068	7\$:	MOVL	R4, R0	:	1388
		04	00	0006B		RET		:	

; Routine Size: 108 bytes, Routine Base: _STR\$CODE + 0000

STR\$COMPARE
1-006

L 2
16-Sep-1984 01:33:01
14-Sep-1984 12:40:02

VAX-11 Bliss-32 V4.0-742
[LIBRTL.SRC]STR\$COMPARE.B32;1

Page 7
(4)

: 199 1389 1 END
: 200 1390 0 ELUDOM

!End of module

PSECT SUMMARY

Name	Bytes	Attributes
_STR\$CODE	108	NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	4	0	581	00:00.8

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS\$:STR\$COMPARE/OBJ=OBJ\$:STR\$COMPARE MSRC\$:STR\$COMPARE/UPDATE=(ENH\$:STR\$COMPARE)

: Size: 108 code + 0 data bytes
: Run Time: 00:05.1
: Elapsed Time: 00:17.4
: Lines/CPU Min: 16257
: Lexemes/CPU-Min: 39368
: Memory Used: 82 pages
: Compilation Complete

0214 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

