

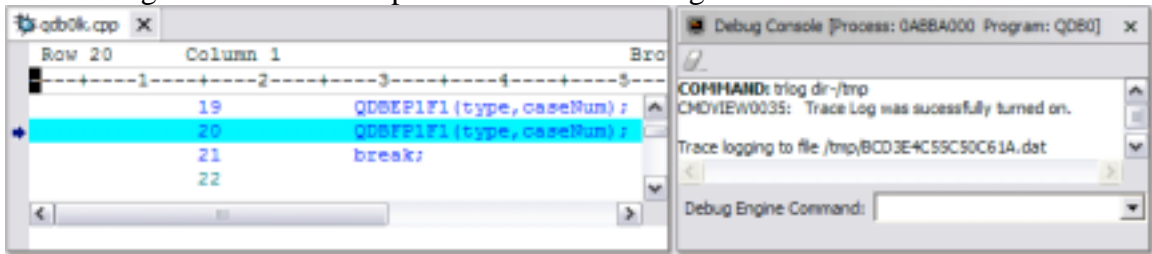
# How do I use trace log through the z/TPF debugger?

**What is trace log?** The trace log facility allows you to trace application function and macro calls to a file or real-time tape (RTx). For more information on trace log see “Trace log facility” in Program Development Support Reference.

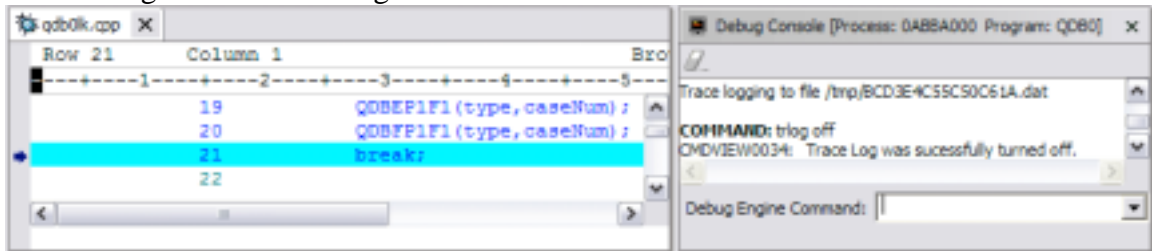
**Why use trace log through the z/TPF debugger?** The “trlog” command allows the user to selectively turn the trace log on or off for the debugged ECB without having change, rebuild, and reload the application to call the trace log function or macro.

## How do I use trace log through the z/TPF debugger?

1. Run to the location you want to start trace log.
2. Enter “trlog” with the desired parameters in the Debug Console.



3. Run to the location you want to stop trace log.
4. Enter “trlog off” in the Debug Console.



5. Post process and view the trace.

**NOTE:** See page 2 for the trace log created here.

**How do I get more information about the “trlog” command?** For more information on the “trlog” command, enter “trlog help” in the Debug Console. For more information on the Debug Console commands, enter “help”.

# How do I use trace log through the z/TPF debugger?

```

1 *****
2 *      Formated ECB Trace Log generated online by TPF.      *
3 *****
4 *
5 * Trace Log Identifier: BCD3E4C55C50C61A
6 *
7 *
8 * All entries included.
9 *
10 *****
11 TR GROUP  LOADMOD LOADSET      OBJECT NAME
12 PSW  IS  OBJ DSP FUNCTION CALL OR MACRO                TIMESTAMP
13
14 Entry # 1
15
16 IBM_DEFT  QDBF  LOADSET-BASE  OBJECT-qdbfp1
17      1      1DE call QDBFP1F1(                          BCD3E539 44C6E89A
18          enum QDBGDRV_T type=00000003,
19          int caseNum=00000001)
20      from QDB0  OBJECT-qdb0lk                OFFSET-C4
21      1      100 call QDBFP1F2(                          BCD3E539 44C8099A
22          enum QDBGDRV_T type=00000003,
23          int caseNum=00000001)
24      from QDBF  OBJECT-qdbfp1                OFFSET-260
25      1      44 call QDBFP1F3(                          BCD3E539 44C87FDA
26          enum QDBGDRV_T type=00000003,
27          int caseNum=00000001)
28      from QDBF  OBJECT-qdbfp1                OFFSET-1BA
29
30 IBM_DEFT  QDBF  LOADSET-BASE  OBJECT-qdbfal
31      1      4C call QDBF1F1(                          BCD3E539 44C9285A
32      from QDBF  OBJECT-qdbfp1                OFFSET-C2
33
34 IBM_DEFT  QDBF  LOADSET-BASE  OBJECT-qdbfc1
35      1      292 call QDBFC1F1(QDBGDRV_T type=00000003,   BCD3E539 44C9E65A
36          int caseNum=00000001)
37      from QDBF  OBJECT-qdbfal                OFFSET-A54
38
39 IBM_DEFT  CISO  LOADSET-BASE  OBJECT-memcmp
40      1      1E call memcmp(const * s1=000000000AB0006C,  BCD3E539 44CB2A1A
41          const * s2=000000038073177A,
42          size_t len=0000000000000004)
43      from QDBF  OBJECT-qdbfc1                OFFSET-2C0
44      1      1F4 return(int FFFFFFFF79) errno=00000000   BCD3E539 44CBC39A
45      from memcmp

```

NOTE: Lines 46-106 removed due to space limitations.

```

107 IBM_DEFT  QDBF  LOADSET-BASE  OBJECT-qdbfc1
108      1      4C2 return(void) errno=00000000             BCD3E539 44D6D69A
109      from QDBFC1F1
110
111 IBM_DEFT  QDBF  LOADSET-BASE  OBJECT-qdbfal
112      1      262 return errno=00000000                   BCD3E539 44D7415A
113      from QDBF1F1
114
115 IBM_DEFT  QDBF  LOADSET-BASE  OBJECT-qdbfp1
116      1      9E return(void) errno=00000000             BCD3E539 44D7A0DA
117      from QDBFP1F3
118      1      136 return(void) errno=00000000           BCD3E539 44D7FFDA
119      from QDBFP1F2
120      1      22C return(void) errno=00000000           BCD3E539 44D8959A
121      from QDBFP1F1
122
123 *****
124 *      End of Formated ECB Trace Log generated online by TPF.      *
125 *****

```