$\mathcal{A}_2$ Traps

Sven Stauber

June 6, 2009

Blah Blah

Contents

1 Traps

When a run-time error is detected, the system generates a numbered trap. The information included with the trap can be used to diagnose the problem. Especially useful is the module and procedure name and PC location where the trap occured. This allows a programmer to find the exact location in the source code.

1.1 Example

```
1  MODULE TrapDemo;
2
3  PROCEDURE Proc2();
4  VAR string : POINTER TO ARRAY OF CHAR;
5  BEGIN
6    string := NIL;
7    ASSERT(string # NIL);
8  END Proc2;
9
10 PROCEDURE Proc1;
11 VAR a, b : LONGINT;
12 BEGIN
13   a := 99; b := 11;
14   Proc2();
15 END Proc1;
```

```
16
17  PROCEDURE Demo*;
18  VAR string : ARRAY 8 OF CHAR;
19  BEGIN
20    string := "Demo!";
21    Proc1();
22  END Demo;
23
24  END TrapDemo.
```

```
1   [1] TRAP 8 PL 3 8 ASSERT failed WinAos Revision 2081 (19.02.2009)
2   CS:=00000023 DS:=0000002B ES:=0000002B SS:=0000002B PC=0ECA7F92
3   ESI=0CCB679A EDI=05FBFF46 ESP=05FBFF20 PID=000017A8
4   EAX=00000000 EBX=00000000 ECX=75443D09 EDX=00000000
5   EBP=05FBFF28 FS:=00000053 GS:=0000002B TMR=00A97689
6   FLAGS: cPaZstIdo iopl0 {1..2, 6, 9}
7   Process: 6056 run 0 20ECE9BF0:Commands.Runner NIL {0, 28}
8   TrapDemo.Proc2 pc=34 [00000022H]
9     string=00000000H (NIL)
10  State TrapDemo:
11    @Self=0ECE9950H (Modules.Module)
12  TrapDemo.Proc1 pc=65 [00000041H]
13    a=99 (00000063H)
14    b=11 (0000000BH)
15  TrapDemo.Demo pc=95 [0000005FH]
16    string="Demo!"
17  Commands.Runner.@Body pc=1042 [00000412H]
18    @Self=0ECE9BF0H (Commands.Runner)
```