

hifrog - Bug #7279

crash in VTT (2)

07/05/2018 19:38 - Sepideh Asadi

Status:	Resolved	Start date:	07/05/2018
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:		Spent time:	0.00 hour
Description File is:../cprover/regression/funfrog/VTT/P2P_Joints_TG3_e.c File is:../cprover/regression/funfrog/VTT/VTT1.c File: /cprover/regression/funfrog/VTT/P2P_Joints_TG3_e_BUG.c File /cprover/regression/funfrog/VTT/VTT.c File /cprover/regression/funfrog/VTT/P2P_Joints_TG3_e.c Claim number # 1 is SAFE Caught exception: identifier tag-#anon#ST[S32'a1' S32'a2' S32'v' S32't1' S32't2' S32't3' S32'delta' S32'initPos' S32'initVel'] not found time 108.58			

History

#1 - 09/05/2018 13:36 - Karine Even Mendoza

Sometimes we can skip this bug when using --no-error-trace option

#2 - 21/06/2018 16:24 - Martin Blich

The problem is in the end when we are reporting results about the assertion.
The exception is thrown in CProver method from_expr() in this piece of code:

```
res.status() << "Claim number # " << claim_number << " is " << (safe ? "SAFE" : "UNSAFE") << res.eom;
```

```
res.status()
    <<" File: " << assertion->source_location.get_file()
    <<" \n Function: " << assertion->source_location.get_function()
    <<" \n Line: " << assertion->source_location.get_line()
    << "\n " << ((assertion->is_assert()) ? "Guard: " : "Code") <<" ( "
    << from_expr(assertion->guard) <<" ) \n";
```

From my point of view this is inability of CProver to convert its inner representation back to source code representation.

#3 - 22/06/2018 10:29 - Martin Blich

- Status changed from New to Resolved

Fixed by letting the from_expr function know about the current symbol table. See git commit 1603c58a.