Subject: Problem building from commandline on Linux
Posted by steffen on Sat, 24 Nov 2012 12:23:48 GMT

Hi,

I'm trying to build my package from the command line, but I'm having trouble getting the right build flags.

In TheIde I have set the following build flags: GUI,MT,GCC,Optimal.
I have changed the "Output Mode" for my main package Debug=Full, to include the debug symbols for stack trace parsing.
This works well when compiling from inside TheIde.

When I try to compile it from the command line it seems to ignore my settings entirely:

theide UppDTS StandAlone GCC -brs +GUI,MT
/home/vcas/Projects/UppDTS/StandAlone/StandAlone
Using the -d flag for debug is not possible, because that changes the optimizations flags, enables all asserts and so on. All other attempts result in a program without debug symbols.

I have tried the -M flag and looked at the generated makefile and I cant get the same result as when I export a makefile from TheIde.
Macro_StandAlone = $(Macro) -DflagMAIN <-- Wrong
vs
Macro_StandAlone = $(Macro) -DflagDEBUG_FULL -DflagMAIN <-- Correct
Is TheIde "Output Mode" settings stored somewhere outside my package dir? Perphaps ".upp/theide/cfg", if so, can I point the command line to it somehow?

My goal is to get an optimal release build, with debug symbols.
Then I strip the debug symbols from the executable file and keep them in a separate file.
Now if my program segfaults in the field it will simply generate a small list of return addresses by calling backtrace, and I can use GDB with the program and symbol file to see where it crashed.
The program is running on a small embedded system, so a full debug release is not an option, neither is full core dumps.

Everything works fine, except the command line build.
If anyone could tell me what I'm doing wrong, I would really appreciate it.

Regards,
Steffen