Subject: Re: Problem building from commandline on Linux
Posted by dolik.rce on Sat, 24 Nov 2012 13:33:33 GMT
View Forum Message <> Reply to Message

steffen wrote on Sat, 24 November 2012 13:23My 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.
Hi Steffen,

Fisrt of all: I see you are using theide to compile, but I'd rather recommend you the umk utility, it is better supported. It is not really possible to make fine-tuning similar to "output mode" dialog on a command line. However, there are at least two options that come on my mind right now:

1) Create a new build method (or copy&modify your GCC.bm) and add '-g' switch to RELEASE_OPTIONS. That will add debugging symbols in the executable. You can also fine tune some other things here, e.g. add DEBUG_MINIMAL to RELEASE_FLAGS (applies to all packages). Building then should be the same, just supply the edited build method on the commandline.

2) Use the makefile generated for release and modify it afterwards. It can by simply done in script using e.g. sed:sed '/Macro_StandAlone =/s/$/ -g -DflagDEBUG_FULL/' /tmp/Makefile

Best regards,
Honza