Download binary .elf file to traget board

-  Helpdesk robot
- **Forum name:** #Board Bring-Up

Hello,
I am new with TRACE32 and would like to get your help . I installed TRACE32 version 2014.10 and I'm targetting microprocessor ARM7 LPC2294. It looks like the installation has been successfully done and the TRACE32 SW recognised the debugger. Please see the Pic1 attached. Howeer, I have a problem to download a binary .elf to my target board. On the website there are many pdf files, but I can't find a concrete example of how to download a binary file and start a debug session. I search in C:\T32\demo\arm\hardware but, still can't find. Could you please give me concrete example for ARM7 LPC2294 how to update my target with binary .elf file and start debug session? Thank you!

Comments (4)

Khaled Jmal
3 years ago
Hello Genady, first stop the program execution using the Break command in the TRACE32 command line, using the Break button or the F8 keyboard key. If the application should run out of RAM, then you can then download your ELF file using the command Data.LOAD.ELF < file.elf > In case you want to write your application to the flash memory, then you should use the script lpc22xx.cmm located in the TRAC32 installation directory under demo/arm/flash, example: CD.DO C:\T32\demo\arm\flash\lpc22xx.cmm A good entry point document is www.lauterbach.com/pdf/debugger_tutorial.pdf

Anonymous
3 years ago
Hi Khaled, thank you for the response! I followed your instruction and successfully updated the target with .elf file using .cmm script located in C:\T32\demo\arm\flash\lpc22xx.cmm. However, my problem now during the debug session only assemblly code is visable but, no C source code (no symbols of the C code). I tried to updated the path to the source code using Symbol -> Source Search Path and added path to the source files. But, that didn't help. Could you please give more insights how to add source code symbols (functions and variables of the C code) to debugger? Thank You!

Khaled Jmal
3 years ago

Hello Genady, first you need to compile your code with enabled debug info. This can be done for instance with gcc using the -g switch. The ELF file will then include the debug symbols as variables, functions and source code lines. Each source code line has generally one start and one end address. This is how the debugger can display the source lines corresponding to the program addresses in the List window. The source files are not included in the elf file by referenced, so the elf file includes the compile path of the source files. In case the source files cannot be found under this path (e.g. you are debugging on a different PC than the one that was used for compiling), then the debugger displays hatches in the List window. In this case, you need to inform the debugger about the location of the source files. This can be simply done by doing a right mouse click on the hatches and selecting "resolve path". You can find additional information about this topic in the following video: https://support.lauterbach.com/kb/articles/displaying-the-source-code In case you only see assembly code without hatches at all, then you need to check your compiler settings.

Anonymous

3 years ago

Hi Khaled, your explanation and the video suggested by you was very helpful. Thank you!