

Lab.2: Cross development environment (debugger)

Purpose

Learn how to build a cross debugger from its source code.

Steps

1. Download source codes from:

1. <https://ftp.gnu.org/gnu/gdb/gdb-8.1.tar.gz>

2. Build Cross Debugger:

1. `tar -zxvf gdb-8.1.tar.gz`
2. `cd gdb-8.1.tar.gz`
3. `sudo dnf install ncurses-devel`
4. `export PATH=$PATH:$HOME/WORK/crossgcc2/bin`
5. `sudo dnf install ncurses-devel`
6. `mkdir build_gdb`
7. `../gdb-8.1/configure --prefix=$HOME/WORK/crossgcc2 --target=arm-linux-gnueabi --enable-tui=yes`
8. `make -j 7`
9. `sudo dnf install texinfo`
10. `make install`

3. Test:

1. `$HOME/WORK/crossgcc2/bin/arm-linux-gnueabi-gdb test`

GDB Stub

A method to use GDB remotely.

- Target: C code links itself with subroutines that implement the GDB "remote serial protocol".
- Host: Run `target remote` to establish the connection to target's C code. And the host can control the target via the protocol.
- Functionality is more complete than the GDB Server method.

GDB Server

A method to use GDB remotely.

- Target: Running `gdbserver`. C code doesn't link itself with subroutines that implement the GDB "remote serial protocol". Target machine doesn't need the symbol table of the C code.
- Host: Needs a copy of the C code, including the all the libraries it uses. Run `target remote` to establish the connection to target's C code.