Remote debugging

Purpose

Learn how to set up GDB Server, and use it to debug remotely.

Steps

Recompile Crossgcc2 with C++ Enabled

1. `export PATH=$PATH:$HOME/WORK/crossgcc2/bin`
2. `mkdir build_gcc2`
3. `cd build_gcc2`
4. `../gcc-7.3.0-RC-20180122/configure --prefix=$HOME/WORK/crossgcc2 --target=arm-linux-gnueabihf --enable-languages=c,c++ --with-sysroot=$HOME/WORK/sysroot --with-arch=armv6 --with-fpu=vfp --with-float=hard --disable-libmudflap --enable-libgomp --disable-libssp --enable-libquadmath --enable-libquadmath-support --disable-libsanitizer --enable-lto --enable-threads=posix --enable-target-optspace --with-linker-hash-style=gnu --disable-nls--disable-multilib --enable-long-long`
5. `make -j 7`
6. `make install -j 7`

Build GDB Server

1. `export CC=arm-linux-gnueabihf-gcc`
2. `export CXX=arm-linux-gnueabihf-g++`
3. `export CFLAGS=-static`
4. `export CXXFLAGS=-static`
5. `./configure --host=arm-linux-gnueabihf --target=arm-linux-gnueabihf`
6. `make -j 7`

Compile C Code

1. `arm-linux-gnueabihf-gcc -static test.c -o test_static.exe`
2. `arm-linux-gnueabihf-gcc test.c -o test.exe`
3. `arm-linux-gnueabihf-gcc -static -g test.c -o test_static_g.exe`
4. `arm-linux-gnueabihf-gcc -g test.c -o test_g.exe`

Recomopile GDB

1. `sudo dnf install expat-devel`
2. `mkdir build_gdb`
3. `../gdb-8.1/configure --prefix=$HOME/WORK/crossgcc2 --target=arm-linux-gnueabihf --enable-tui=yes`
4. `make -j 7`
5. `make install`

Put `gdbserver` and the compiled programs on Raspberry Pi.

Setup Connection between PC and Raspberry Pi

- On Raspberry Pi:
  - `ifconfig eth0 192.168.1.5 broadcast 192.168.1.255 netmask 255.255.255.0`
- On PC:
  - `sudo ifconfig enp62s0u1 192.168.1.3 broadcast 192.168.1.255 netmask 255.255.255.0`

Debug

- On Raspberry Pi:
  - `./gdbserver 192.168.1.3:1234 ./test_static_g.exe`
- On PC:
  1. `arm-linux-gnueabihf-gdb`
  2. `file test_static_g.exe`
  3. `target remote 192.168.1.5:1234`

Results

- test.exe: can't be executed by gdb
- test_g.exe: can't be executed by gdb
- test_static.exe: can be executed by gdb with assembly information
  test_static_g.exe: can be executed by gdb with original c code information
- The `-static` option packs dynamic libraries into the binary file, so that it can be executed without target machine's library support.
- The `-g` option packs the original c code into the binary file.