Linux

Here are the installation commands for a few Linux distributions.

Packages

- Ubuntu 18.04 or newer / Debian stretch or newer

NOTE `gdb-multiarch` is the GDB command you'll use to debug your ARM Cortex-M programs

---

```
sudo apt install gdb-multiarch openocd qemu-system-arm
```

- Ubuntu 14.04 and 16.04

NOTE `arm-none-eabi-gdb` is the GDB command you'll use to debug your ARM Cortex-M programs

---

```
sudo apt install gdb-arm-none-eabi openocd qemu-system-arm
```

- Fedora 27 or newer

```
sudo dnf install gdb openocd qemu-system-arm
```

- Arch Linux

NOTE `arm-none-eabi-gdb` is the GDB command you'll use to debug ARM Cortex-M programs

---

```
sudo pacman -S arm-none-eabi-gdb qemu-arch-extra openocd
```

udev rules

This rule lets you use OpenOCD with the Discovery board without root privilege.

Create the file `/etc/udev/rules.d/70-st-link.rules` with the contents shown below.

```
# STM32F3DISCOVERY rev A/B - ST-LINK/V2
ATTRS{idVendor}=="0483", ATTRS{idProduct}=="3748", TAG+="uaccess"

# STM32F3DISCOVERY rev C+ - ST-LINK/V2-1
ATTRS{idVendor}=="0483", ATTRS{idProduct}=="374b", TAG+="uaccess"
```

Then reload all the udev rules with:

```
sudo udevadm control --reload-rules
```

If you had the board plugged to your laptop, unplug it and then plug it again.

You can check the permissions by running this command:

```
lsusb
```

Which should show something like

```
(..)
Bus 001 Device 018: ID 0483:374b STMicroelectronics ST-LINK/V2.1
(..)
```

Take note of the bus and device numbers. Use those numbers to create a path like `/dev/bus/usb/<bus>/<device>`. Then use this path like so:

```
ls -l /dev/bus/usb/001/018
```

```
crw-------+ 1 root root 189, 17 Sep 13 12:34 /dev/bus/usb/001/018
```

```
getfacl /dev/bus/usb/001/018 | grep user
```

```
user::rw-
user:you:rw-
```

The `+` appended to permissions indicates the existence of an extended permission. The `getfacl` command tells the user `you` can make use of this device.

Now, go to the next section.