

# Xilinx & JTAG tools

- [Location](#)
- [JTAG with the Digilent USB driver](#)
- [JTAG USB "Dongle" and xmd](#)
  - [Driver installation](#)
  - [Driver installation on RHEL6](#)
  - [Multiple xmd sessions](#)
  - [Cleaning up stuck JTAG sessions](#)

## Location

- The Xilinx tools are located at `/u1/reg/package/xilinx`.
- There is also an installation at `/afs/slac/g/reseng/xilinx`.
  - This one is read-only, unless arrangements are made with the electronics group.

## JTAG with the Digilent USB driver

### JTAG USB "Dongle" and xmd

#### Driver installation

- See [Configuration Hardware User Guides](#) and the [USB Cable Installation Guide](#)
- Xilinx installations (at least some) have the driver installation script installed at `ISE_DS/common/bin/lin64/install_script/install_drivers`
  - It must be run as root
  - Need write access to the directory tree
- The package can possibly (unverified) be installed and run from a stand-alone directory, i.e., it doesn't need the full Xilinx installation
- Need to have `fxload` installed
  - ~~Yum install fxload should do it, but need internet access~~
  - Need an RPM to install `fxload` properly on RHEL5:
    - [http://rpm.pbone.net/index.php3/stat/4/idpl/15644090/dir/redhat\\_el\\_5/com/fxload-2008\\_10\\_13-2.el5.i386.rpm.html](http://rpm.pbone.net/index.php3/stat/4/idpl/15644090/dir/redhat_el_5/com/fxload-2008_10_13-2.el5.i386.rpm.html)
    - [http://rpm.pbone.net/index.php3/stat/4/idpl/15644148/dir/redhat\\_el\\_5/com/fxload-2008\\_10\\_13-2.el5.x86\\_64.rpm.html](http://rpm.pbone.net/index.php3/stat/4/idpl/15644148/dir/redhat_el_5/com/fxload-2008_10_13-2.el5.x86_64.rpm.html)
  - Need an RPM for RHEL6:
    - No 32 bit version...
    - [http://rpm.pbone.net/index.php3/stat/4/idpl/15160548/dir/redhat\\_el\\_6/com/fxload-2008\\_10\\_13-2.el6.x86\\_64.rpm.html](http://rpm.pbone.net/index.php3/stat/4/idpl/15160548/dir/redhat_el_6/com/fxload-2008_10_13-2.el6.x86_64.rpm.html)
    - See below for installation on a RHEL 6 machine
  - Can get it from sourceforge, but compilation is required
  - Easier to copy 3 files from an existing installation, e.g. `rdusr108`
    - `/sbin/fxload`
    - `/usr/share/man/man8/fxload.8.gz`
    - `/usr/share/usb/a3load.hex`
  - Set ownership to `root:root` and `read/execute` privilege as needed
- `cd` to installation directory and do `./install_drivers`
  - There is a bug that can cause it to fail when the linux version is not 2.4.\*: Modify the script to replace "2.4" in the line `'TP_VERSION_2_4'` to `4=`uname -r | grep -c "2.4" ` with "2\4"`

#### Driver installation on RHEL6

- See Xilinx [Answer Record](#) Platform Cable USB/USB-II - Libusb Driver support available on Linux
- See [FPGArelated.com](#) posts, especially the one by Jan Pech
  - Note formatting issues in this post. Get rid of all 3D instances; fix wrapped lines to be single lines; be sure to introduce spaces where they were deleted by the line wrap
  - The resulting `xusbdfwu.rules` should look like:

```
# version 0003
ATTR{idVendor}=="03fd", ATTR{idProduct}=="0008", MODE="666"
SUBSYSTEM=="usb", ACTION=="add", ATTR{idVendor}=="03fd", ATTR{idProduct}=="0007", RUN+="/sbin
/fxload -v -t fx2 -I /usr/share/xusbdfwu.hex -D $tempnode"
SUBSYSTEM=="usb", ACTION=="add", ATTR{idVendor}=="03fd", ATTR{idProduct}=="0009", RUN+="/sbin
/fxload -v -t fx2 -I /usr/share/xusb_xup.hex -D $tempnode"
SUBSYSTEM=="usb", ACTION=="add", ATTR{idVendor}=="03fd", ATTR{idProduct}=="000d", RUN+="/sbin
/fxload -v -t fx2 -I /usr/share/xusb_emb.hex -D $tempnode"
SUBSYSTEM=="usb", ACTION=="add", ATTR{idVendor}=="03fd", ATTR{idProduct}=="000f", RUN+="/sbin
/fxload -v -t fx2 -I /usr/share/xusb_xlp.hex -D $tempnode"
SUBSYSTEM=="usb", ACTION=="add", ATTR{idVendor}=="03fd", ATTR{idProduct}=="0013", RUN+="/sbin
/fxload -v -t fx2 -I /usr/share/xusb_xp2.hex -D $tempnode"
SUBSYSTEM=="usb", ACTION=="add", ATTR{idVendor}=="03fd", ATTR{idProduct}=="0015", RUN+="/sbin
/fxload -v -t fx2 -I /usr/share/xusb_xse.hex -D $tempnode"
```

- Make sure dongles are disconnected from USB and applications attempting to interact with them are not running
- Become **root** (`sudo -s`)
- Need to have `libusb`, `libusb1`, `fxload` installed (see above)

```
yum install libusb libusb1 fxload
```

- Write permission is needed to run the installation script, so

```
cp -r $XILINX/bin/lin64 /tmp/xilinx
```

on the machine on which the drivers are to be installed

- If a 32 bit machine, use `cp -r $XILINX/bin/lin /tmp/xilinx`
- Modify `/tmp/xilinx/lin64/setup_pcusb`:
  - Force it to use `udev` by changing the lines:

```
TP_USE_UDEV="0"
TP_UDEV_ENABLED=`ps -e | grep -c udevd`
```

to:

```
TP_USE_UDEV="1"
TP_UDEV_ENABLED="1"
```

- Make the script executable (`chmod +x setup_pcusb`)
  - Execute it (`./setup_pcusb`)
- Apparently errors are ignorable. Output looks something like:

```
--File /usr/share/xusbdfwu.hex does not exist.
--Error getting file version for /usr/share/xusbdfwu.hex.
--Updating xusbdfwu.hex file.
--File /usr/share/xusb_xlp.hex does not exist.
--Error getting file version for /usr/share/xusb_xlp.hex.
--Updating xusb_xlp.hex file.
--File /usr/share/xusb_emb.hex does not exist.
--Error getting file version for /usr/share/xusb_emb.hex.
--Updating xusb_emb.hex file.
--File /usr/share/xusb_xpr.hex does not exist.
--Error getting file version for /usr/share/xusb_xpr.hex.
--Updating xusb_xpr.hex file.
--File /usr/share/xusb_xup.hex does not exist.
--Error getting file version for /usr/share/xusb_xup.hex.
--Updating xusb_xup.hex file.
--File /usr/share/xusb_xp2.hex does not exist.
--Error getting file version for /usr/share/xusb_xp2.hex.
--Updating xusb_xp2.hex file.
--File /usr/share/xusb_xse.hex does not exist.
--Error getting file version for /usr/share/xusb_xse.hex.
--Updating xusb_xse.hex file.
--File /etc/udev/rules.d/xusbdfwu.rules exists.
--File /etc/udev/rules.d/xusbdfwu.rules version = 0003
--File xusbdfwu.rules exists.
--File xusbdfwu.rules version = 0003
--File xusbdfwu.rules is already updated.
```

- Ensure the hex files were installed in `/usr/share`

```
$ ls -l /usr/share/*.hex
-rw-r--r-- 1 root root 21708 Feb  9 13:30 /usr/share/xusb_emb.hex
-rw-r--r-- 1 root root 21708 Feb  9 13:30 /usr/share/xusb_xlp.hex
-rw-r--r-- 1 root root 22956 Feb  9 13:30 /usr/share/xusb_xp2.hex
-rw-r--r-- 1 root root 20740 Feb  9 13:30 /usr/share/xusb_xpr.hex
-rw-r--r-- 1 root root 22956 Feb  9 13:30 /usr/share/xusb_xse.hex
-rw-r--r-- 1 root root 21666 Feb  9 13:30 /usr/share/xusb_xup.hex
-rw-r--r-- 1 root root 21666 Feb  9 13:30 /usr/share/xusbdfwu.hex
```

If the permissions or ownership aren't as above, change them

- Ensure the rules were installed in `/etc/udev/rules.d`

```
$ ls -l /etc/udev/rules.d/xusbdfwu.rules
-rw-r--r-- 1 root root  987 Feb  9 15:04 xusbdfwu.rules
```

If the permissions or ownership isn't as above, change them

- Plug a dongle into a USB port
- Use `lsusb` to see whether a line like `Bus 002 Device 013: ID 03fd:0008 Xilinx, Inc.` appears. You should find a file called `/proc/bus/usb/<Bus>/<Device>`.
- Use `dmesg` to look for errors. When it's working properly, you'll see something like:

```
usb 2-1: new high speed USB device using ehci_hcd and address 13
usb 2-1: New USB device found, idVendor=03fd, idProduct=0008
usb 2-1: New USB device strings: Mfr=1, Product=2, SerialNumber=0
usb 2-1: Product: XILINX
usb 2-1: Manufacturer: XILINX
usb 2-1: configuration #2 chosen from 1 choice
```

- If it's not working properly, try running the `fxload` command found in `/etc/udev/rules.d/xusbdfwu.rules` by hand. Use the one that has the same `idVendor` and `idProduct` given in the `dmesg` output. Use `/proc/bus/usb/<Bus>/<Device>` for `$tempnode`.
- The drivers are loaded and working when the dongle's light is either amber or green.

## Multiple xmd sessions

- Multiple `xmd` sessions **can** nominally run on one computer. See [this](#) and [this](#).
  - Add `-cable type xilinx_platformusb port usb2[#]` to your `xmd.ini` or `.xmddrc` file, where `#` is the USB2 port number (from 1 to some large number) you wish to use.
  - `/sbin/lsusb` lists USB devices
  - To get a list of which cables are which can be seen by `xmd`, you can run the `xrcableesn xmd` command. This will list the ESN (Electronic Serial Number) for each cable and which `usb2#` port it is on. If you already know the ESN for the cable you want to connect to, you can replace the above "port usb2#" argument with "esn #####".

## Cleaning up stuck JTAG sessions

- Also see [here](#).
- If an `xmd` process was aborted, it may have left the cable resources in an unclear state, preventing reconnecting to that cable. To clean these up, issue the `xclean xmd` command.



### Warning

Doing this affects *all* JTAG dongles connected to the machine it is executed from!