

Xilinx

Contents

1	Introduction	1
2	Xilinx installations	1
3	Setup ISE 11.5 on Ubuntu 10.04 (lucid)	2
4	Setup ISE DS 12.2 on Ubuntu 10.04 (lucid)	3
5	Install JTAG support	5
5.1	Configure udev	5
5.2	Install usb driver	6
6	FPGA Editort	7
7	Desktop shortcut icons	7
7.1	Xilinx 11	7
7.2	Xilinx 12	8
8	Scan chain and load firmware	9

1 Introduction

XILINX FPGA are embeded into :

- DIF (1)
- LDA (3: startup, main and hdmi), using version 11 for IP gigabit ethernet (but **not DS**)
- DCC, using version 12

Tools access the boards using JTAG (Joint Test Action Group stand for IEEE 1149.1 also called « Standard Test Access Port and Boundary-Scan Architecture »).

- IMPACT and ISE for XILINX's JTAG in and out streams
- OPENOCD for JTAG output streams: Linux Free Open Source Software (not used)
- QUARTUS for ALTERA (not XILINX) embed into DHCAL.
- FPGA EDITOR for add by hand pinoches in bitstreams. (XILINX: available in ISE)

File *svn/calice/online-sw/trunk/doc/altera-linux.txt* describe QUARTUS installation on polcaldaq. It wasn't tested. It was intended de be used by Guillaume for ALTERA FPGA.

2 Xilinx installations

Create a compte on Xilinx web site to download "SFD" software, wich is the fisrt minor relase (other are only updates and do not provide installer). File *svn/calice/online-sw/trunk/doc/xilinx-linux.txt* also describes the ISE installations. Installers binaries packages are saved at *calice@polcaldaq: /Download/pkg/*.

- Make sure you have at least 30 GB available where you want to install ISE (in the end, only 10 GB will be used but the installation process is... "under-optimized")
-
- Download *Xilinx_11.1_ISE_SFD.tar* from Xilinx,
- Untar it in */tmp*: `$ tar -C /tmp -xf Xilinx_11.1_ISE_DS_SFD.tar`,
- run `$ /tmp/Xilinx_11.1_ISE_SFD/xsetup ...`
 - leave the "cable drivers" checkbox unchecked.
 - I choose */mnt/data1/ubuntu/usr/xilinx11* as the base.
 - When asking for licence, close the dialog box.
 - When asking for Update, close the dialog box too (If not you have to jump the following point).
- You should remove installer now: `$ rm -fr /tmp/Xilinx_11.1_ISE_DS_SFD`.
- Download *Xilinx_11.5_ISE_DS_lin[64].tar* from Xilinx,
- Untar it in */tmp*,
- run `$ /tmp/Xilinx_11.5_ISE_DS_lin[64]/xsetup` and choose the same base.
- You should remove installer now.
- Installation is finished, you can test:

```
$ /mnt/data1/ubuntu/usr/xilinx11/ISE/bin/lin[64]/ise
$ /mnt/data1/ubuntu/usr/xilinx11/ISE/bin/lin[64]/impact
```

Note that ISE complain for a licence.

3 Setup ISE 11.5 on Ubuntu 10.04 (lucid)

- Load variables for licence */mnt/data1/ubuntu/etc/env-xilinx.sh*

```
XLX_TARGET=/mnt/data1/ubuntu/usr/xilinx11/
#XLX_LD_PRELOAD=/mnt/data1/ubuntu/usr/lib/libusb-xilinx-driver.so

# Licences
XILINXD_LICENSE_FILE='5296@ccflex04.in2p3.fr,5296@ccflex05.in2p3.fr,5296@ccflex06.in2p3.fr'
## XILINXD_LICENSE_FILE='1700@127.0.0.1,1800@127.0.0.1,1900@127.0.0.1'
export XILINXD_LICENSE_FILE

# X11 issues with fpga_editor
#LD_LIBRARY_PATH="$LD_LIBRARY_PATH:$XLX_TARGET/ISE/X11R6/lib/lin64"
#export LD_LIBRARY_PATH
```

- Now ISE doesn't complain for the licence:

```
$ . /mnt/data1/ubuntu/etc/env-xilinx.sh
$ /mnt/data1/ubuntu/usr/xilinx11/ISE/bin/lin[64]/ise
```

- In the main *settings[64/32].sh* file, replace the `pushd/popd` by a `_here='/bin/pwd'`, `cd $d`, `cd "$_here"`, and source becomes `''`:

```

<<<
foreach i ( $newLst )
...
        pushd "$d" > /dev/null
        source "$sfn"
        if ( $rc == 0 ) then
            popd > /dev/null
===
_here='/bin/pwd'
for i in $newLst
...
        cd "$d"
        . "$sfn"
        if [ $rc = 0 ]; then
            cd "$_here"
>>>

```

- Create a script wrapper `/mnt/data1/ubuntu/usr/bin/xlx`:

```

#!/bin/sh

. /mnt/data1/ubuntu/etc/env-xilinx.sh
. /mnt/data1/ubuntu/usr/xilinx11/settings64.sh
exec env LD_PRELOAD="$XLX_LD_PRELOAD" "$@"

```

- Finished:

```

$ /mnt/data1/ubuntu/usr/bin/xlx impcat
$ /mnt/data1/ubuntu/usr/bin/xlx ise

```

4 Setup ISE DS 12.2 on Ubuntu 10.04 (lucid)

For version 12, you will find working configuration in `svn/calice/online-sw/trunk/doc/xilinx-ds12-lucid64.tgz`. This page describe the installation:

Validated on 3 different 64b machines.

- Download `Xilinx_ISE_DS_12.2_M.63c.1.1.tar` from Xilinx.
- Untar it in `/tmp`
- Run `./xsetup` until completion. Let's call `DSBASE` the location on disk where you installed ISE : You should have a single `ISE_DS` directory under `DSBASE`
- In all subdirectories under `DSBASE/IDE_DS/` (ie. `ISE`, `EDK`, `PlanAhead`, `common`, ...), edit the `settings64.sh` file:
- Comment-out the line: `XIL_SCRIPT_LOC="$1"`
- Comment-out the line: `unset XIL_SCRIPT_LOC`
- Edit `@DSBASE@/ISE_DS/settings64.sh`:
- Remove the whole block from `if [ $# != 0 ]; then` until `unset XIL_SCRIPT_LOC_TMP_UNI ; fi`
- Right before the line `'. $d/settings64.sh "$d"`, add the line: `XIL_SCRIPT_LOC="$d"`
- Add the following script as `DSBASE/xilinx12.sh`:

```

#!/bin/sh

unset LANG
unset LANGUAGE

_here='dirname "$0"'
here='cd "$_here" && /bin/pwd'

. "$here"/ISE_DS/settings64.sh "$here"/ISE_DS
unset here

if [ -n "$XLX_LD_PRELOAD" ] ; then
    LD_PRELOAD="$LD_PRELOAD $XLX_LD_PRELOAD"
    export LD_PRELOAD
fi

exec "$@"

```

- `chmod 755 this @DSBASE@/xilinx12.sh` script
- Create a new directory: `@DSBASE@/ISE_DS/ZZZ_IN2P3`
- Add the following `settings64.sh` file in that newly created `DSBASE/ISE_DS/ZZZ_IN2P3/` directory:

```

XILINXD_LICENSE_FILE='5296@ccflex04.in2p3.fr,5296@ccflex05.in2p3.fr,5296@ccflex06.in2p3.fr'

LM_LICENSE_FILE="$XILINXD_LICENSE_FILE"
export XILINXD_LICENSE_FILE LM_LICENSE_FILE

XILINX_IN2P3=${XIL_SCRIPT_LOC}
export XILINX_IN2P3

if [ -n "$PATH" ]
then
    PATH=${XILINX_IN2P3}/bin:${PATH}
else
    PATH=${XILINX_IN2P3}/bin
fi
export PATH

# USB Cable
XIL_IMPACT_USE_LIBUSB=1
export XIL_IMPACT_USE_LIBUSB

XLX_LD_PRELOAD=${XILINX_IN2P3}/../usb-driver/libusb-driver.so
export XLX_LD_PRELOAD

```

- Now create the following `bin` directory: `DSBASE/ISE_DS/ZZZ_IN2P3/bin`
- Add the following `xterm` script in that `DSBASE/ISE_DS/ZZZ_IN2P3/bin` directory:

```

#!/bin/sh

if [ x"$*" = x"-e sh --noprofile --norc" ] ; then
    exec gnome-terminal --sm-client-disable --disable-factory
    --profile=xilinx-12 --title="XPS Terminal"
elif [ x"$1" = x-e ] ; then
    shift

```

```

    exec gnome-terminal --sm-client-disable --disable-factory
--profile=xilinx-12 --title="XPS run $1" -- "$@"
fi

```

```

exec /usr/bin/xterm "$@"

```

- Add the following *make* script in that *DSBASE/ISE_DS/ZZZ_IN2P3/bin* directory:

```

#!/bin/sh

_here='dirname "$0"'
_here='cd "$_here" && /bin/pwd'

# Make sure make/impact/etc. from here are used first, if any
PATH="$_here:$PATH"
export PATH

# Some xilinx tools refuse to work with the usb driver wrapper lib (64b)
unset LD_PRELOAD
exec /usr/bin/make "$@"

```

- symlink as gmake:

```

ln -s make @DSBASE@/ISE_DS/ZZZ_IN2P3/bin/gmake

```

- Add the following *impact* script in that *DSBASE/ISE_DS/ZZZ_IN2P3/bin* directory:

```

#!/bin/sh

# Make sure the LD_PRELOAD is set correctly
_here='dirname "$0"'
_here='cd "$_here" && /bin/pwd'

LD_PRELOAD="$XLX_LD_PRELOAD"
export LD\_PRELOAD

set -x

exec "$_here"/../../ISE/bin/lin64/impact "$@"

```

- Fix permissions:

```

chmod a+x DSBASE/ISE\_DS/ZZZ\_IN2P3/bin/*

```

From here, you should be able to run ISE, XPS, etc, but the JTAG won't work + you may see warning related to LD_PRELOAD not correctly defined and/or *libusb-driver.so* unavailable (-> see next section).

Note: due to limitations in FPGA editor, when launching ISE on the local screen of the machine, make sure the DISPLAY environment variable is set to `localhost:0`.

5 Install JTAG support

Trying to initialize the bundary chain using impact says: `Module windrvr6 is not loaded. Please reinstall the`
This topic help on resolv that.

5.1 Configure udev

- Plug the xilinx cable, \$ `dmesg` (or # `udevadm monitor`) talk little and the led is unlighted:

```
usb 2-2: new full speed USB device using ohci_hcd and address 3
usb 2-2: not running at top speed; connect to a high speed hub
usb 2-2: configuration #1 chosen from 1 choice
```

- Running \$ `lsusb` should show the device:

```
Bus 002 Device 017: ID 03fd:0007 Xilinx, Inc.
```

- Unplug the cable
- Create `/etc/udev/rules.d/xusbdfwu.rules`

```
$ cd DSBASE/common/bin/lin64/install_script/install_drivers/linux_drivers/pcusb

# Replace '=' by '-ge' when testing "ps -e | grep -c udevd"
$ sed -e 's:\(if \[ $TP_UDEV_ENABLED\) = \(("1" \]\):\1 -ge \2:' -i setup_pcusb
$ sudo ./setup_pcusb
```

- restart udev: # `service udev restart`

```
$ tail /var/log/syslog
udevd[5826]: SYSFS{=} will be removed in a future udev version... in /etc/udev/rules.d/xusbdfwu.rules:2
udevd[5826]: BUS= will be removed in a future udev version... in /etc/udev/rules.d/xusbdfwu.rules:3
```

- Fix udev configuration:

```
# sed -e 's:SYSFS:ATTR:g;s:BUS:SUBSYSTEM:g;s:TEMPNODE:tempnode:g' -i /etc/udev/rules.d/xusbdfwu.rules
```

- Install fxload: # `aptitude install fxload`
- Plug the xilinx cable, \$ `dmesg` says more and the led blind and now is lighted orange:

```
usb 2-2: new full speed USB device using ohci_hcd and address 4
usb 2-2: not running at top speed; connect to a high speed hub
usb 2-2: configuration #1 chosen from 1 choice
usb 2-2: USB disconnect, address 4
usb 2-2: new full speed USB device using ohci_hcd and address 5
usb 2-2: not running at top speed; connect to a high speed hub
usb 2-2: configuration #3 chosen from 1 choice
usb 2-2: USB disconnect, address 5
```

- Finally \$ `lsusb` output make sure the firmware has been uploaded to the JTAG pod: should print the :0008 as in the following output

```
Bus 001 Device 022: ID 03fd:0008 Xilinx, Inc.
```

- Note: that you may want to also fix the `udev` installation layout:

```
# mv /etc/udev/rules.d/xusbdfwu.rules /lib/udev/rules.d/99-xusbdfwu.rules
```

5.2 Install usb driver

To install the USB userland driver for the Xilinx tools:

- Download <http://git.zerfledert.de/cgi-bin/gitweb.cgi/usb-driver?a=snapshot;h=HEAD;sf=tgz>
- `# aptitude install libusb-dev` needed for compilation
- Untar, cd to the directory created and then: `make`
- Copy the resulting `libusb-driver.so` file to its destination:

```
$ cp libusb-driver.so /mnt/data1/ubuntu/usr/lib/libusb-xilinx-driver.so
```

- Uncomment the corresponding line in `/mnt/data1/ubuntu/etc/env-xilinx.sh`
- Now Error message is: `WARNING:IMPACT:923 - Can not find cable, check cable setup`
- Allow the current user to access the JTAG cable:

```
sudo adduser 'id -nu' lp
```

- Sorry but now you should reboot.

Note: Never use 2 programs that access the jtag at the same time ! (you will have to reboot the machine)

6 FPGA Editort

- Add needed libraries

```
$ /mnt/data1/ubuntu/usr/bin/xlx fpga_edito
libXm.so.3: cannot open shared object file: No such file or directory
```

```
# aptitude install libmotif3
$ /mnt/data1/ubuntu/usr/bin/xlx fpga_editor
libstdc++.so.5: cannot open shared object file: No such file or directory
```

```
$ cd tmp
$ wget 'http://fr.archive.ubuntu.com/ubuntu/pool/universe/g/gcc-3.3/libstdc++5_3.3.6-17ubuntu1_amd64.deb'
$ dpkg -x libstdc++5_3.3.6-17ubuntu1_amd64.deb .
# cp /tmp/usr/lib/libstdc++.so.5 /usr/local/lib
# ldconfig
$ rm -fr /tmp/usr/
$ /mnt/data1/ubuntu/usr/bin/xlx fpga_editor
Wind/U X-toolkit Error: wuDisplay: Can't open display
```

- Export display: create the `/etc/gdm/custom.conf` file and restart gdm:

```
[security]
# Needed for Xilinx FPGA editor ???
DisallowTCP=false
```

- Allow connection

```
$ env DISPLAY=localhost:0.0 /mnt/data1/ubuntu/usr/bin/xlx fpga_editor
Cannot register service: RPC: Unable to receive; errno = Connection refused
```

```
# aptitude install portmap
$ env DISPLAY=localhost:0.0 /mnt/data1/ubuntu/usr/bin/xlx fpga_editor
Symbol '_XtperDisplayList' causes overflow in R_X86_64_PC32 relocation
Symbol '_XtGetPerDisplayInput' causes overflow in R_X86_64_PC32 relocation
Warning!!: XKEYSYMDB environment variable is set to a wrong location
```

7 Desktop shortcut icons

7.1 Xilinx 11

- */Desktop/impact 11.desktop:*

```
[Desktop Entry]
Version=1.0
Type=Application
Terminal=false
Exec=/mnt/data1/ubuntu/usr/bin/xlx impact
Name=impact 11
Icon=/usr/share/pixmaps/apple-green.png
```

- */Desktop/ISE 11.desktop:*

```
[Desktop Entry]
Version=1.0
Type=Application
Terminal=false
Icon[en_US]=gnome-network-properties
Exec=/mnt/data1/ubuntu/usr/bin/xlx ise
Name[en_US]=ISE 11
Name=ISE 11
Icon=/usr/share/pixmaps/apple-red.png
```

7.2 Xilinx 12

- */Desktop/Xilinx-XTerm-12.desktop:*

```
#!/usr/bin/env xdg-open
```

```
[Desktop Entry]
Version=1.0
Type=Application
Terminal=false
Icon[en_US]=gnome-network-properties
Name[en_US]=Terminal Xilinx 12
Exec=@DSBASE@/xilinx12.sh gnome-terminal --profile=xilinx-12
Name=XTerm-xilinx-12
Icon=gnome-network-properties
```

- */Desktop/Impact-12.desktop:*

```
#!/usr/bin/env xdg-open
```

```
[Desktop Entry]
Version=1.0
Type=Application
Terminal=false
Icon[en_US]=gnome-network-properties
Name[en_US]=Xilinx Impact 12
Exec=@DSBASE@/xilinx12.sh impact
Name=Impact-12
Icon=gnome-network-properties
```

- */Desktop/XPS-12.desktop:*

```
#!/usr/bin/env xdg-open
```

```
[Desktop Entry]
Version=1.0
Type=Application
Terminal=false
Icon[en_US]=gnome-network-properties
Exec=@DSBASE@/xilinx12.sh xps
Name=XPS-12
Icon=gnome-network-properties
Name[en_US]=Xilinx XPS 12
```

- */Desktop/ISE-12.desktop:*

```
#!/usr/bin/env xdg-open
```

```
[Desktop Entry]
Version=1.0
Type=Application
Terminal=false
Icon[en_US]=gnome-network-properties
Name[en_US]=Xilinx ISE 12
Exec=env DISPLAY=localhost:0 @DSBASE@/xilinx12.sh ise
Name=ISE-12
Icon=gnome-network-properties
```

- */Desktop/SDK-12.desktop:*

```
#!/usr/bin/env xdg-open
```

```
[Desktop Entry]
Version=1.0
Type=Application
Terminal=false
Icon[en_US]=gnome-network-properties
Exec=@DSBASE@/xilinx12.sh xsdk
Name=SDK-12
Icon=gnome-network-properties
Name[en_US]=Xilinx SDK 12
```

8 Scan chain and load firmware

There is 2 ways to programs the FPGA:

- bitstream (*.bit*) for test
- flash memory (*.mcs*) that preserve firmware at next reboot

Please follow these steps:

- use LD_PRELOAD environment variable to shortcut the library calls which is done by using */mnt/data1/ubuntu/usr/bin/ldx*)
- double click on “boundary scan” and right click in the window and select “initialize chain” that will scan the FPGA chain plug to JTAG, for instance on the LDA:
 - first flash memory: click “bypass”
 - second flash memory: click “bypass”

- bitstream RAM memory: choose “bitstream file”
- click “assign new configuration file” and “program”

Bitstreams are:

- DCC: *svn/calice/hardware/trunk/DCC/production/Projet_dcc_prod_hdl/ise/dcc_prod/top_dcc.bit*
- LDA main: *svn/calice/hardware/hardware/trunk/UK_firmware/LDA/firmware/ise/main/md2.bit*
- LDA hdmi: *svn/calice/hardware/trunk/UK_firmware/LDA/firmware/ise/hdmi/hdmi_module_top_level.bit*
- DIF ECAL: */mnt/data1/karmic64/home/calice/DIF2/DEV1/DIF2_DEV1_0.bit*