Rahul Vora
rhlvora@gmail.com
rvora@unm.edu

Open-Sourcing FPGA Development

Detailed in this document are the steps to developing for Xilinx FPGAs on open source operating systems. It will be assumed the host computer in question will be running Fedora Core (7), or RHEL.

Installing & Executing the Xilinx WebPack

The first step in any case is to download the ISE WebPack software from the Xilinx web site (http://www.xilinx.com). To do so will require registration, which is free. The download itself is around 1.7GiB.

32 Bit CPU Installation

Extract the newly downloaded file with the unzip command. The command is as follows for version 9.2 of the software, be sure to exchange the name with the actual name of the package.

```
$ unzip WebPack_SFD_92i.zip
```

Change directories to the newly extracted directory, and execute the shell script named "setup" as root. Example commands follow.

```
$ cd webpack
$ su
# sh setup
```

By convention, the software should be installed in "/usr/local", however, the choice is left to the individual.

64 Bit CPU Installation

Installation on 64 bit systems is greatly complicated by the fact that as of now, Xilinx refuses to support 64 bit systems by not releasing a 64 bit version of the

software or including so much as a 64 bit compatible installer. It is because of this that owners of 64 bit computers will have to creatively use the 32 bit version.

Firstly, extract the file with the following command as above, making sure to exchange the name for the true one.

```
$ unzip WebPack_SFD_92i.zip
```

Change directories to the newly extracted directory, here is where the installation first deviates from that of the 32 bit installation. In order to run the installation, a copy of the 32 bit standard C library, for some reason not shipped with standard 64 bit distributions, is required.

Somehow acquire the library named "libc-2.6.so". It is very effective to use standard internet search engines to locate rpms or binaries with said file, and extract it manually. If one is unable to do this, it is indeed possible to take it from a fellow 32 bit computer with a compatible operating system.

Once it is acquired, place it in the directory "/lib" as root, and create a symbolic link to "/lib/libc.so.6".

```
$ su
# mv libc-2.6.so /usr/lib
# ln -s /lib/libc-2.6.so /lib/libc.so.6
```

Now, return to the software directory, and edit the file "setup" such that line 28 no longer reads "echo "Un-supported 64 bit platform: $platform" and instead reads "$setuploc/bin/lin/setup". Then simply run the file "setup" as root like so.

```
$ su
# ./setup
```

Alternatively, and perhaps more easily, one can simply run the file "bin/lin/setup" as root and forgo the editing process entirely.

```
$ su
# cd bin/lin
# ./setup
```

Again, it is recommended that due to convention, it should be installed in "/usr/local".

Execution

Again, the execution of the Xilinx ISE WebPack Project Manager is a bit more complicated than one would expect. Essentially, in order to run the Project Manager, one needs to either execute it manually or automate the process by themselves.
To execute it manually, first change directories to where the WebPack was installed. The likely location being "/usr/local/xilinx", then execute the shell script "settings.sh" before executing "bin/lin/ise". Example commands are as follows.

```
$ cd /usr/local/xilinx
$ ./settings.sh
$ bin/lin/ise
```

SELinux

On many modern operating systems, SELinux is active be default, for those with strict policies, it is recommended that the user disable SELinux entirely, however, for those with targeting policies, the following procedure works well. Essentially, all libraries that will be used by the WebPack should be relabeled to type "textrel_shlib_t". An example follows.

```
$ su
# cd /usr/local/xilinx/bin/lin
# chcon -t textrel_shlib_t *.so
```

If WebPack still does not run, do "$ sealert -b" and follow the convenient instructions.