**\*\* README document for Ubuntu 14.04 and Ubuntu 16.04 versions \*\***

**To install Quartus-II, ModelSim, UrJTAG, GHDL, GTKWave and Krypton CPLD board drivers**

-----------------------------------------------------------------------------------------------------------------------------

**\*\* Please read the instructions carefully. Type the commands without any mistake. (Don’t use COPY & PASTE)**

**A) For Ubuntu updates:**

- sudo apt-get update

**(Don’t use “IIT ftp servers” or “main server” in the Ubuntu Software centre for downloading updates. Use “Servers in India”)**

**\*\* Only for 64-bit Ubuntu,** following libraries are required on the machine. So run these (additional) commands in the terminal

-----------------------------------------------------------------------------------------------------------------------------

sudo apt-get install libxft2

sudo apt-get install libxft2:i386

sudo apt-get install libncurses5

sudo apt-get install libncurses5:i386

sudo apt-get install libxtst6

sudo apt-get install libsm6:i386

sudo apt-get install libxtst6:i386

-----------------------------------------------------------------------------------------------------------------------------

**B) Altera Quartus II:-** is a programmable logic device design software produced by Altera. Quartus II enables analysis and synthesis of HDL designs, which enables the developer to compile their designs, perform timing analysis, examine RTL diagrams, simulate a design's reaction to different stimuli, and configure the target device with the programmer. Quartus includes an implementation of VHDL and Verilog for hardware description, visual editing of logic circuits, and vector waveform simulation).

**(Source:** [**https://en.wikipedia.org/wiki/Altera\_Quartus**](https://en.wikipedia.org/wiki/Altera_Quartus)**)**

**The following steps are given for the installation of Quartus-II (Version 13.0) integrated with modelsim. The same you can follow for Quartus 16 version..**

**\* Steps to install Quartus-II (Version 13.0):**

**1) Download the TAR file <**Quartus-web-13.0.0.156-linux.tar.gz**> from the local server link <-------->**

2) Go to the location, where the file is downloaded.

3) Extract or uncompress by “right click” on the file and use “Extract Here” option or

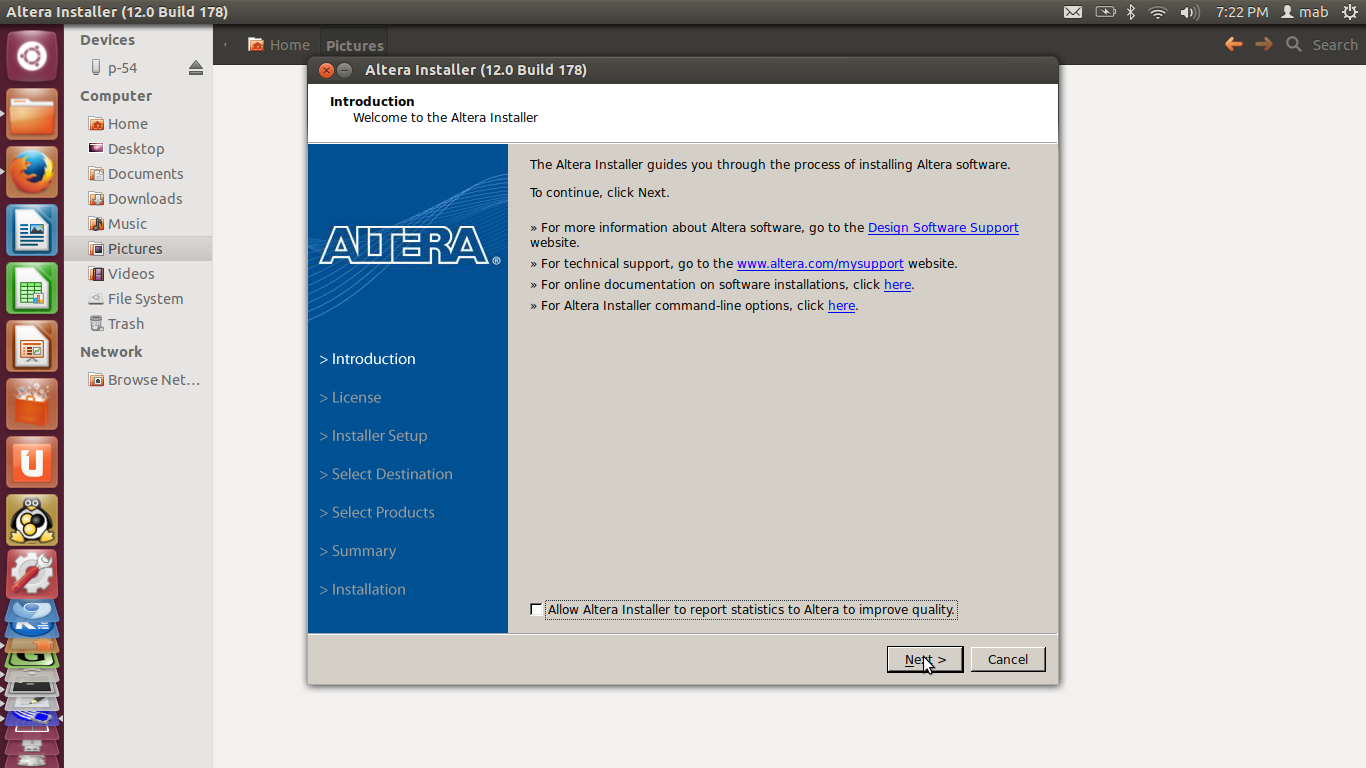
or run the command: sudo tar -zxvf Quartus-web-13.0.0.156-linux.tar.gz

4) Then go to the uncompressed folder “Quartus-web-13.0.0.156-linux” by using

cd Quartus-web-13.0.0.156-linux

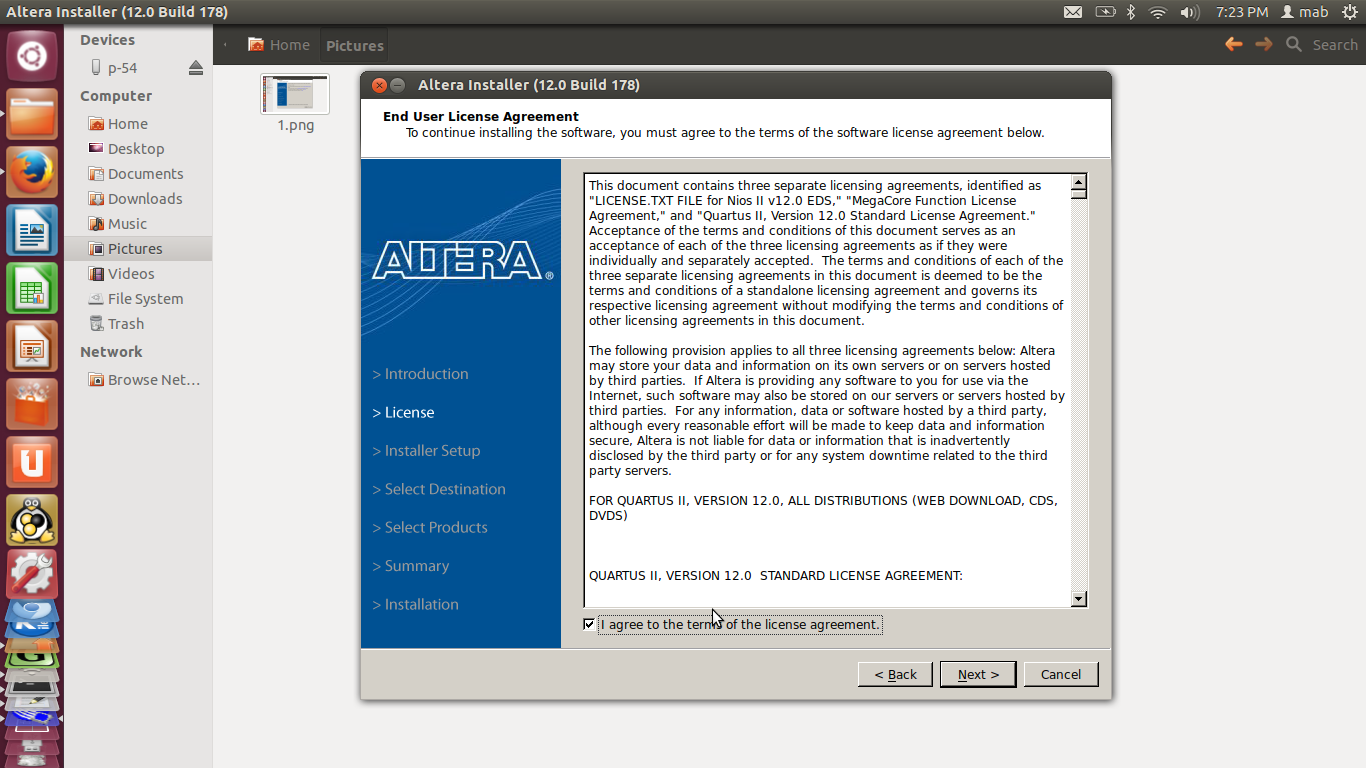
5) Then run the command: **./setup.sh**

**(The images used here for your reference are of Quartus-12.0 version. So Graphical Interface might not be similar, but installation procedure will be more or less similar).**

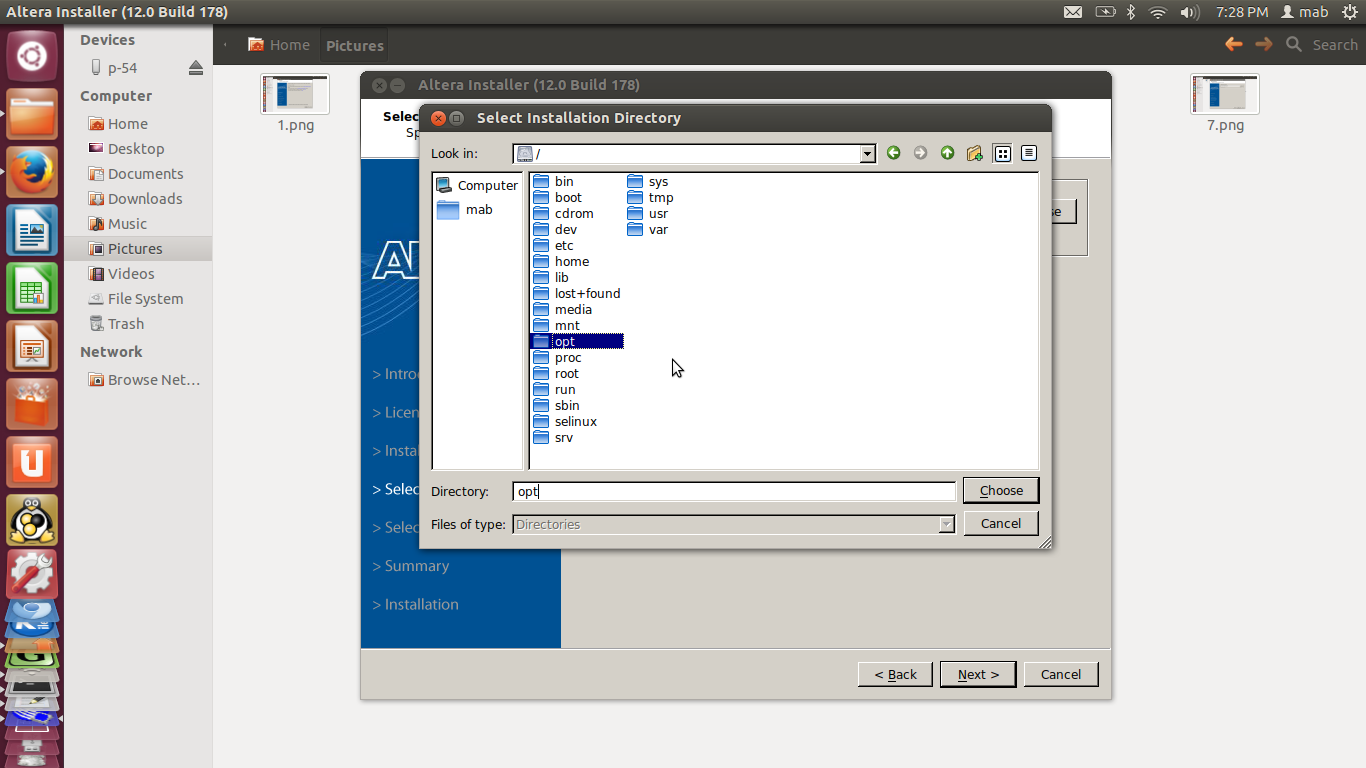


Click ‘Next’

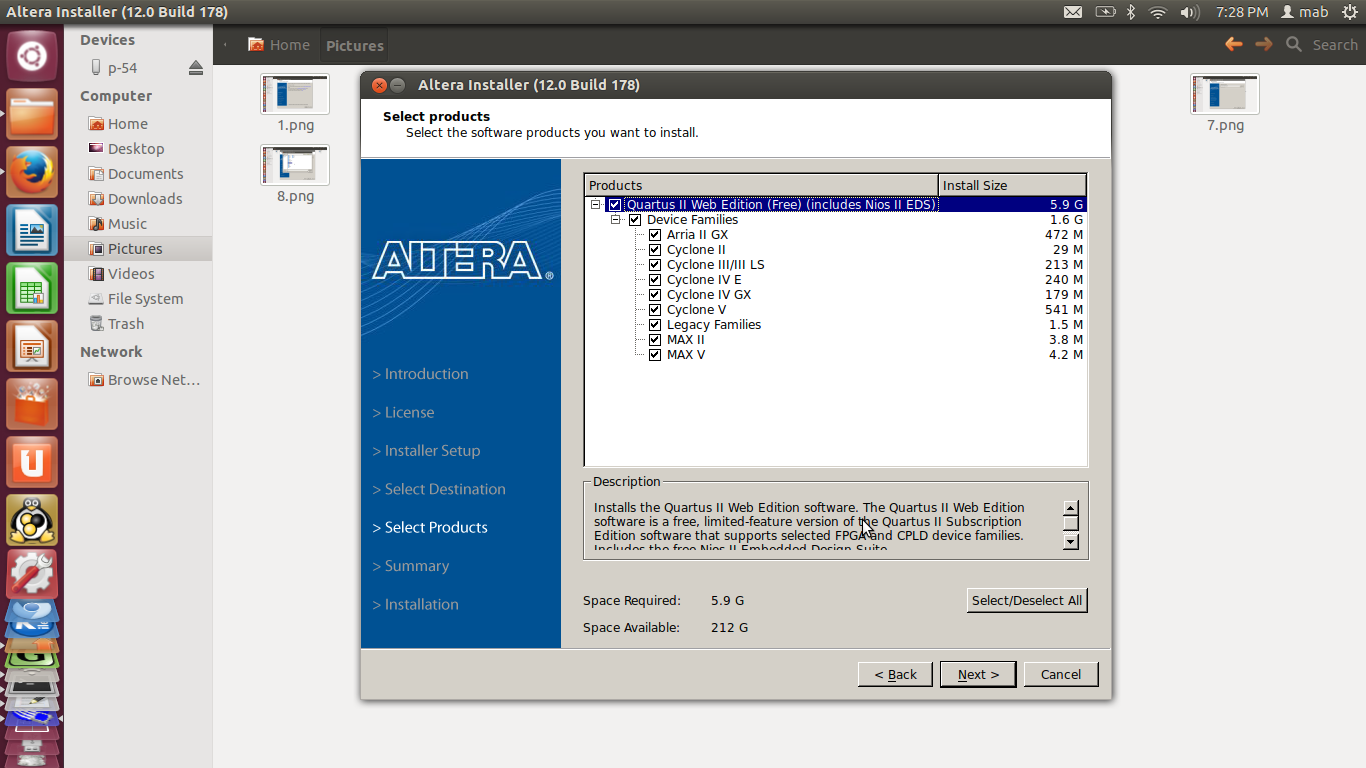
6) Tick  I Agree and Click ‘Next’



7) Choose Installation Directory of your choice e.g.“/home” and Click “Next”.



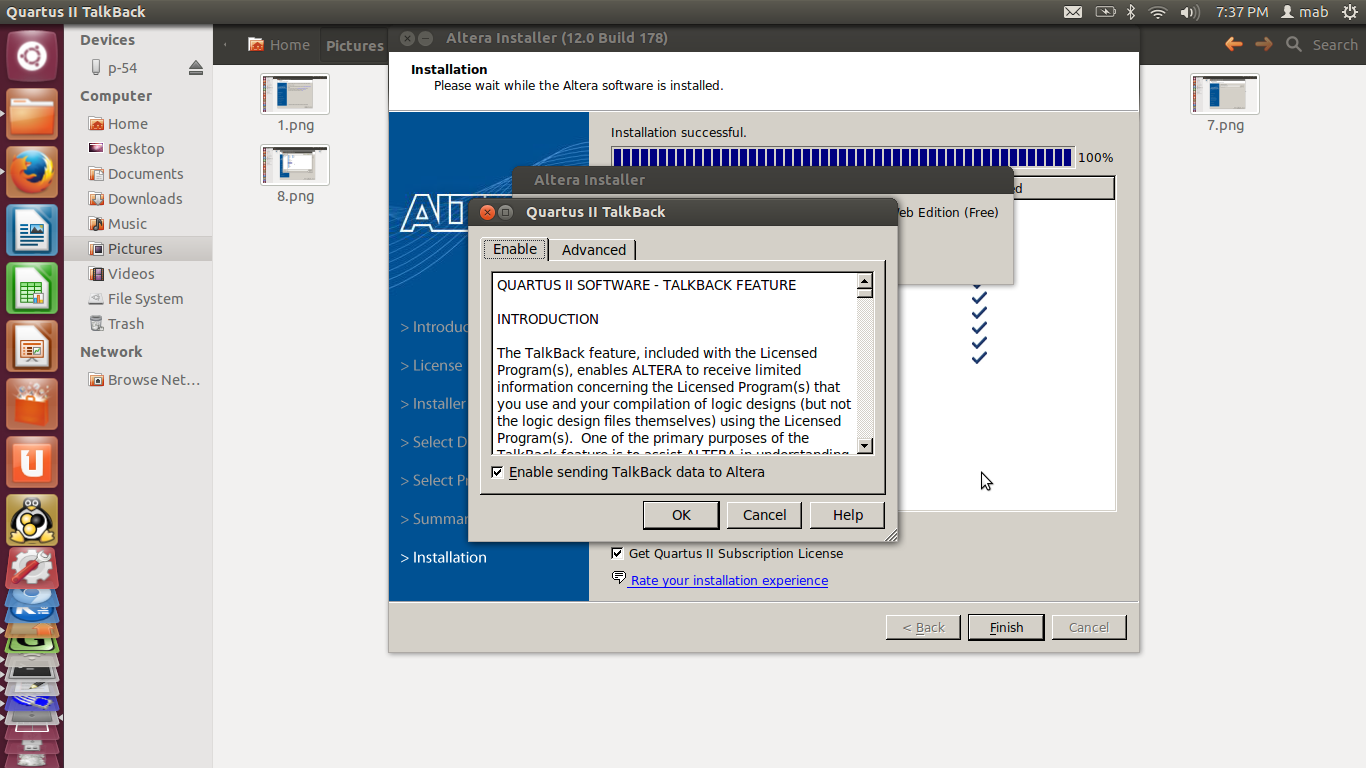
8) Select the software products (Tick  all products) and modelsim-altera starter edition (free) and Click “Next”.



9) Now, the window will show “Summary of the installation”, Click “Next”.

10) Next window will show the Installation progress.

11) At the end of the installation, you will get a pop-up window (Untick “sending feedback”) and Click “OK”.



12) Untick “Get Quartus-II Subscription License”. Also, choose the option “create shortcut on Desktop” and Click “Finish”.

**You are done with the Quartus-II Installation.**

Now, to invoke Quartus-II GUI for your assignments, select

Option-1: Double click directly on the desktop icon or

Option-2: - Goto the <Quartus\_Installation\_PATH>/altera/13.0/quartus/bin

- ./quartus (Press enter)

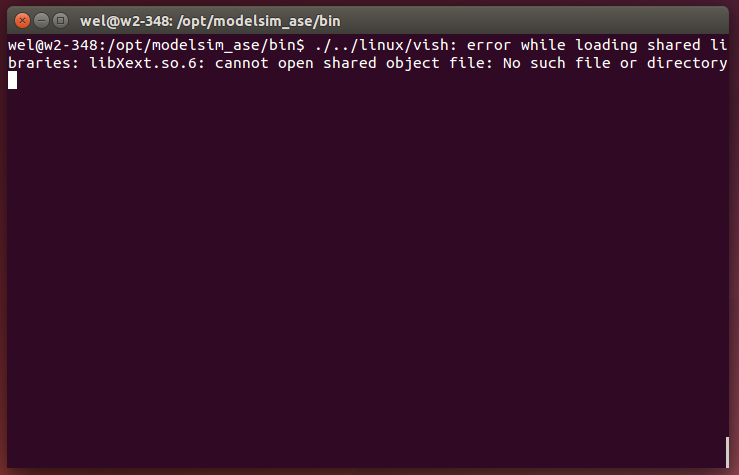
**----------------------------------------------------------------------------------------------------------------------------**

**For Testing Modelsim:**

After installation goto, cd <Quartus\_installtion\_PATH>/altera/13.0/modelsim\_ase/bin then run the command **./vsim**. ModelSim Graphical environment will be available to test your simulation results.

If graphical interface doesn’t work and shows the following error, install following dependencies:

* sudo apt-get install libxft2:i386
* sudo apt-get install libxrender1:i386 libxtst6:i386 libxi6:i386



After these installations, if you still find errors in invoking modelsim through Quartus-II, try out the following steps to make **ModelSim ALTERA STARTER EDITION vsim** work on **Ubuntu-14.04** or in the newer versions (16.04) of Ubuntu.

**(Source: Matthew Swabey** [**http://mattaw.blogspot.in/2014/05/making-modelsim-altera-starter-edition.html**](http://mattaw.blogspot.in/2014/05/making-modelsim-altera-starter-edition.html) **)**

**Solutions for Ubunutu 14.04 and 16.04 (Try all these instructions in the given sequence)**

The free version of ModelSim Altera Edition is 32-bit only while the normal Linux PC will be 64-bit. On Linux this requires us to install the 32-bit versions of the libraries that it depends on and it is fully supported on linux like **Ubuntu 14.04 or 16.04**. Run the following commands one-by-one **(Type the commands. Don’t use COPY and PASTE. This may create some errors).**

* sudo dpkg --add-architecture i386
* sudo apt-get update **(Don’t use “IIT ftp servers” in the Ubuntu software centre for downloading updates. Use “Servers in India”)**
* sudo apt-get install build-essential
* sudo apt-get install gcc-multilib g++-multilib lib32z1 lib32stdc++6 lib32gcc1 expat:i386 fontconfig:i386 libfreetype6:i386 libexpat1:i386 libc6:i386 libgtk-3-0:i386 libcanberra0:i386 libpng12-0:i386 libice6:i386 libsm6:i386 libncurses5:i386 zlib1g:i386 libx11-6:i386 libxau6:i386 libxdmcp6:i386 libxext6:i386 libxft2:i386 libxrender1:i386 libxt6:i386 libxtst6:i386

Then solution to solve the following error appears while running vsim:

\*\* Fatal: Read failure in vlm process (0,0)

Segmentation fault (core dumped)

you need to build a new version of freetype, a font setting library and modify ModelSim to use it. First download the source code of freetype 2.4.12.  
<http://download.savannah.gnu.org/releases/freetype/freetype-2.4.12.tar.bz2> or **download it from the local server link**  <------->

Now install the build dependencies needed for libfreetype6, extract the source (using tar) and configure and build libfreetype:

* sudo apt-get build-dep -a i386 libfreetype6
* tar -xjvf freetype-2.4.12.tar.bz2 **(path to “freetype-2.4.12.tar.bz2”)**
* cd freetype-2.4.12 **(go to the extracted “freetype-2.4.12” folder)**
* ./configure --build=i686-pc-linux-gnu "CFLAGS=-m32""CXXFLAGS=-m32""LDFLAGS=-m32"
* make -j8

The finished libraries are now available inside the "objs/.libs" directory. As they are necessary to run ModelSim, we need to copy them into the install directory and then modify ModelSim's vsim script to use the new libraries instead of the system wide versions.

Change directory to the directory where ModelSim is installed,

cd /<Quartus\_installtion\_PATH>/altera/13.0//modelsim\_ase

Then, - sudo mkdir lib32

* sudo cp {freetype-2.4.12 PATH}/objs/.libs/libfreetype.so\* ./lib32

Now, edit the vsim launch script to ensure the new freetype libraries are used:

* sudo gedit /<Quartus\_installtion\_PATH>/altera/13.0/modelsim\_ase/bin/vsim

Search for the following line: dir=’dirname $arg0’

and underneath add the following new line: export LD\_LIBRARY\_PATH=${dir}/lib32

Finally, Open quartus, go to --> Tools --> Options --> General --> EDA Tools options --> ModelSim-Altera and set the path /<Quartus\_installtion\_PATH>/altera/13.0/modelsim\_ase/bin. Then press OK.

**Additional settings for Ubuntu 16.04 only**

If you are getting the error (Could not find ../linux\_rh60) for ModelSim on Ubuntu 16.04, use the following corrections.

- cd /<Quartus\_installtion\_PATH>/altera/13.0/modelsim\_ase/bin

- sudo gedit vsim

**(Keep a original copy of a vsim file as a backup. If you make any mistake in the editing, then it will be difficult for you to find out the errors)**

- change vco="linux\_rh60" to vco="linux"

- change mode=${MTI\_VCO\_MODE:-""} to mode=${MTI\_VCO\_MODE:-"32"}

**Testing of Modelsim through Quartus-II (Download the “Modelsim Test files” from the local server)**

* Open the .qpf project file in the Quatus-II environment (This will include all the related files)
* Open the VDHL files of this project
* Compile the project files
* Then, run **RTL Simulation** in the Tools menu (This will invoke the modelsim window).

If there’s any error please check all the installation steps again.

**----------------------------------------------------------------------------------------------------------------------------**

**Other simulation and graphical output waveform Viewer tools for HDL (GTKWAVE and GHDL)**

**C) To install GTKWAVE (waveform viewer):**

* sudo apt-get install gtkwave

**----------------------------------------------------------------------------------------------------------------------------**

**D) To install GHDL simulation software (files are available in the separate folder):**

- sudo apt install gnat-4.9-base\_4.9.2-1\_amd64.deb

- sudo apt install libgnat-4.9\_4.9.2-1\_amd64.deb

- sudo apt install ghdl\_0.33-1jessie1\_amd64.deb

- sudo apt-get install zlib1g-dev

**Testing of GHDL (“GHDL Test files” folder is available in the GHDL folder on the local server)**

**(Use the following commands in the working directory)**

* To compile all the VHDL files in a directory: ghdl -a \*.vhd \*.vhdl
* After compilation, use the following command to create an executable: ghdl -m top\_entity\_name
* Finally, to run the simulation and generate the waveforms,

ghdl -r top\_entity\_name --stop-time=XXXns --wave=waveform.ghw

* Then, to view waveforms: gtkwave waveform.ghw

**----------------------------------------------------------------------------------------------------------------------------**

**E) To install USB drivers for the Krypton CPLD board:**

**1) Go to the local server link to download <51usbblaster.zip> and** and unzip it.

**2) sudo cp –r /<PATH>/51-usbblaster.rules /etc/udev/rules.d** (copy the file to the given location)

(Write the correct location of the "51-usbblaster.rules" file, while using the command, e.g., if the location of the file is /home/user/Downloads/, then the command will be "cp -r /home/user/Downloads/51-usbblaster.rules /etc/udev/rules.d")

**Reload drivers:**

* sudo udevadm control --reload

**-------------------------------------------------------------------------------------------------------------------------------------------------------**

**F) To install UrJTAG (Software tool to download bit file from PC to CPLD board)**

1) sudo apt-get remove urjtag (to remove other versions of Urjtag, if any)

2) Download from your local server <urjtag\_0.10+r2007-1\_i386.deb>

3) Install the package by using,

sudo apt install /<PATH of Urjtag file>/urjtag\_0.10+r2007-1\_i386.deb

4) Then follow the steps to add the "device information file" (Download the file “altera.tar.gz” from the local server <------->

- sudo cp -r /<PATH>/altera.tar.gz /usr/share/urjtag (Mention correct location of "altera.tar.gz" file)

- cd /usr/share/urjtag

- sudo tar -zxvf altera.tar.gz

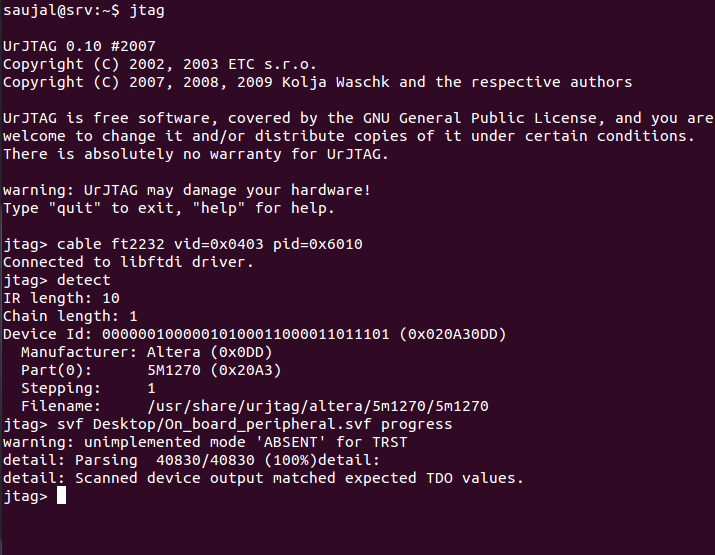
- sudo chmod -R 755 altera

- jtag

**To check the connection with Krypton CPLD and to download “bit file” to the board, use following commands**

1. jtag (**Connect the board with the PC/ laptop)**
2. cable ft2232 vid=0x0403 pid=0x6010
3. detect
4. svf (location of the svf file/filename.svf) progress **(to download the bit file to CPLD board)**

**See the results in the following snapshot after running each command**



**Prepared By: Mahesh Bhaganagare (mab@ee.iitb.ac.in)**