Using Quartus II, UrJTAG and Krypton Board

Wadhwani Electronics Lab

Department of Electrical Engineering
Indian Institute of Technology Bombay

Jan 2018

Before reading this document...

Make sure that

- you have installed Quartus II and ModelSim on your Laptop/PC.
- you have successfully compiled your design.
- you have successfully simulated it.
- you have installed UrJTAG as per the instruction handout uploaded on Moodle
- you have downloaded Test files for Krypton board from Moodle.

Lab session

In this session, you will

1. be introduced to Krypton board
2. generate .svf file.
3. load .svf file into the CPLD
4. test your Krypton with the test files provided, in order to confirm its working
5. verify your full adder design by loading your fulladder.svf file and testing using the switches and LEDs

Introduction to Krypton

You will be briefly introduced with Krypton in this presentation. For more information please refer to the Lab manual and user manual of Krypton.

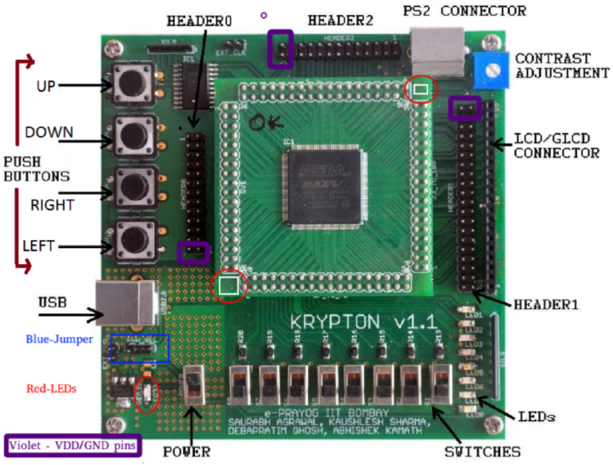


Figure: Caption

Krypton features

Main features of the board:

1. CPLD used:5M1270ZT144C5N, Altera's MAX V family with 980 macro-cells/1270 logic elements
2. USB programmable and powered, with provision for external DC supply
3. Preconfigured on-board I/Os- 8 switches, 8 LEDs and 4 push-buttons
4. On-board clock of 1Hz and 50MHz, and provision for external clock source connection
5. Connectors provided on-board to interface standard peripherals directly (LCD/GLCD/PS2)
6. Large number of on-board I/Os (86) provided for various applications

Testing the Krypton board and UrJTAG

- In this test we are going to test:
  - On-board I/Os - 8 switches, 8 LEDs and 4 push-buttons.
  - On-board clock of 1Hz and 50MHz.
- To test the above peripherals we need to load the Serial Vector File(.svf) file to the Krypton Board.
- To load the .svf file on the board, we will use UrJTAG.
- Now connect your Krypton board to your Laptop/PC.

Testing the Krypton board: loading .svf File

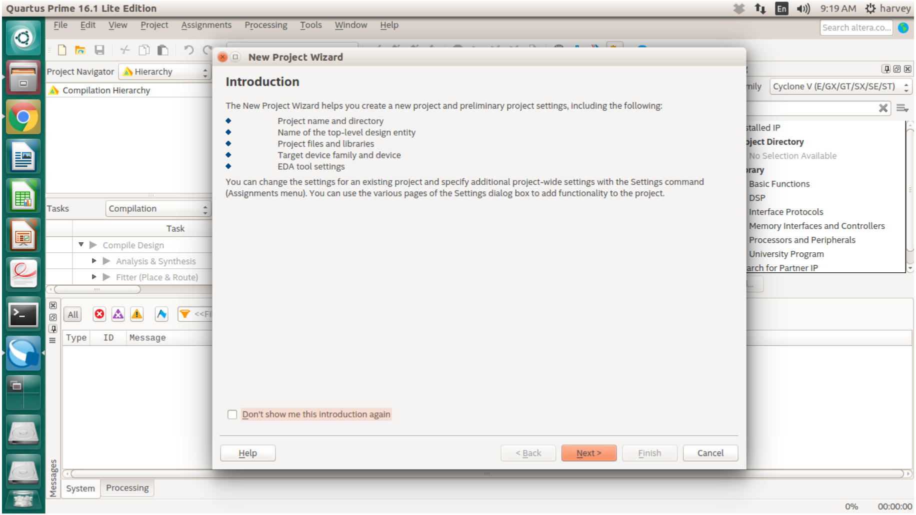
- Now open the terminal and type:
  jtag
- Now you will be presented with a command console. Type:
  cable ft2232 vid=0x0403 pid=0x6010
  An indication should now appear that the driver is connected.
- Now type:
  detect
  which displays the detected CPLD device (Details like IRChain length, Manufacturer, Device ID, Stepping etc.)
- Load the svf file into the CPLD device by typing
  svf On_board_peripheral.svf progress
  Make sure that you specify full path of .svf file.

Testing the Krypton board: On_board_peripheral

- Once the file is loaded, initially all switches are to be kept off. For each test, **exactly one of the switches** is to be turned on. The tests are as follows:
- **Switch s1:** The eight LED's will be lit up to display an 8-bit count which is updated every 0.04194s.
- **Switch s2:** The eight LED's will be lit up to display an 8 bit count which is updated every second.
- **Switch s3:** All LEDs are lit up. There are four push buttons. You can check each of them by pressing them one at a time and checking the observations which should be as follows:
  - If push-button up is pressed, LED 1 and 5 will turn off
  - If push-button down is pressed, LED 2 and LED 6 will turn off and so on.
- Switches s4,s5,s6,s7,s8 (one at a time): The corresponding LED (i.e. LED-4 for switch s4 etc.) will turn on.

Using Quartus II - New Project

In the introductory page, click Next.



New Project Wizard

Introduction

The New Project Wizard helps you create a new project and preliminary project settings, including the following:

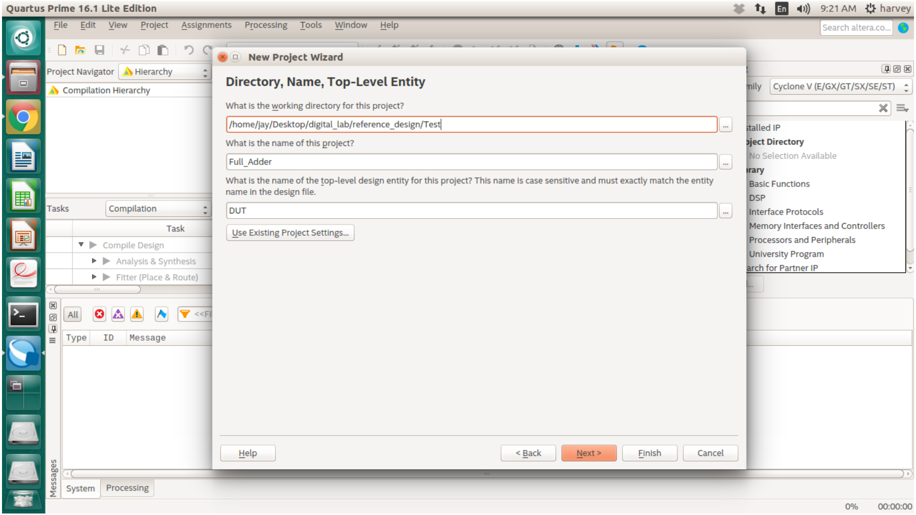
- Project name and directory
- Name of the top-level design entity
- Project files and libraries
- Target device family and device
- EDA tool settings

You can change the settings for an existing project and specify additional project-wide settings with the Settings command (Assignments menu). You can use the various pages of the Settings dialog box to add functionality to the project.

□ Don't show me this introduction again

Using Quartus II- Project Directory and Top-level Module

In this page, specify a working directory for your project. It is a good practice to open a new folder for every new project.



Directory, Name, Top-Level Entity

What is the working directory for this project?

/home/jay/Desktop/digital_lab/reference_design/Test

What is the name of this project?

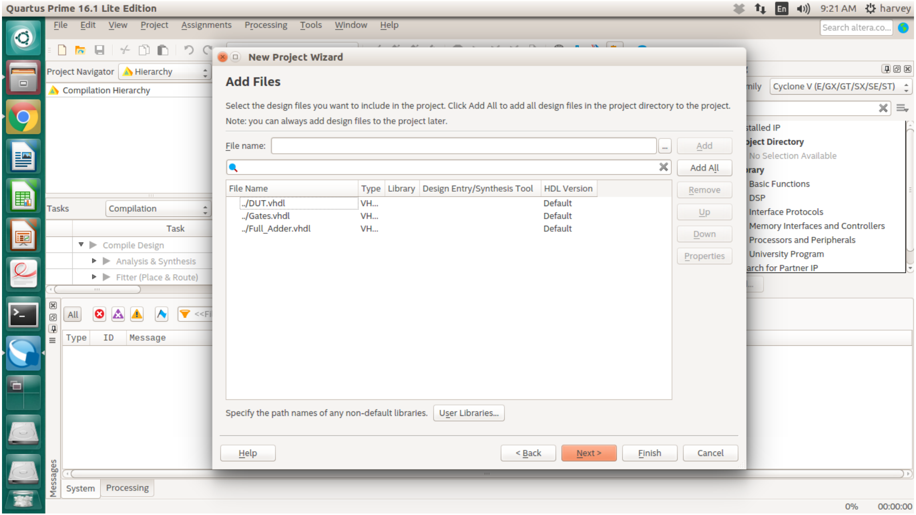
Full_Adder

What is the name of the top-level design entity for this project? This name is case sensitive and must exactly match the entity name in the design file.

DUT

Using Quartus II- Adding Files to Project

Next page may be skipped, In this page add all relevant files to your project.



New Project Wizard

Add Files

Select the design files you want to include in the project. Click Add All to add all design files in the project directory to the project.

Note: you can always add design files to the project later.

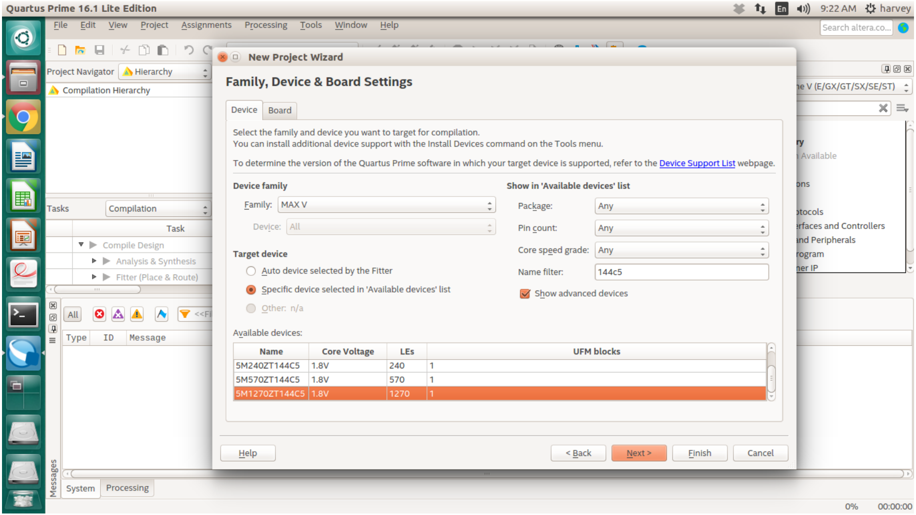
File name:

File Name	Type	Library	Design Entry/Synthesis Tool	HDL Version
../DUT.vhdl	VH...			Default
../Gates.vhdl	VH...			Default
../Full_Adder.vhdl	VH...			Default

Specify the path names of any non-default libraries.

Using Quartus II- Device Selection

In this page, select the target CPLD. Select Max V from Device family. Then type 144c5 in Name fitter and select last one.



New Project Wizard

Family, Device & Board Settings

Device | Board

Select the family and device you want to target for compilation.
You can install additional device support with the Install Devices command on the Tools menu.

To determine the version of the Quartus Prime software in which your target device is supported, refer to the Device Support List webpage.

Device family

Family: MAX V

Device: All

Target device

○ Auto device selected by the Fitter

● Specific device selected in 'Available devices' list

○ Other: n/a

Show in 'Available devices' list

Package: Any

Pin count: Any

Core speed grade: Any

Name filter: 144c5

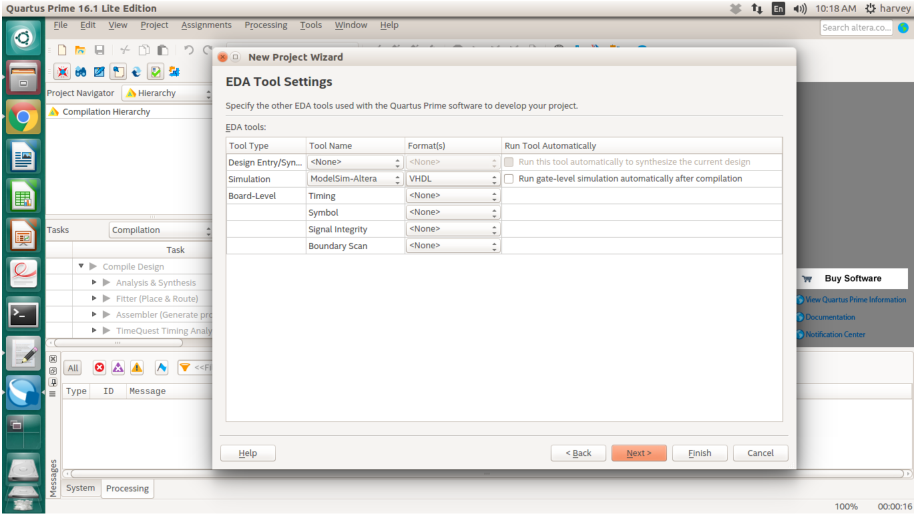
☑ Show advanced devices

Available devices:

Name	Core Voltage	LEs	UFM blocks
5M240ZT144C5	1.8V	240	1
5M570ZT144C5	1.8V	570	1
5M1270ZT144C5	1.8V	1270	1

Using Quartus II- Simulation tool and HDL Selection

In this page, select the target simulation tool as Modelsim-Altera and language as VHDL.



New Project Wizard

EDA Tool Settings

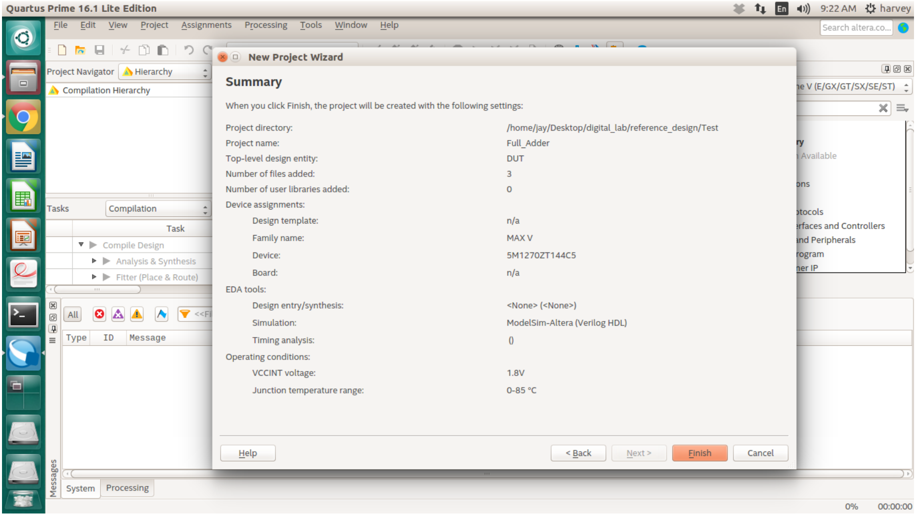
Specify the other EDA tools used with the Quartus Prime software to develop your project.

EDA tools:

Tool Type	Tool Name	Format(s)	Run Tool Automatically
Design Entry/Syn...	<None>	<None>	□ Run this tool automatically to synthesize the current design
Simulation	ModelSim-Altera	VHDL	□ Run gate-level simulation automatically after compilation
Board-Level	Timing	<None>	
	Symbol	<None>	
	Signal Integrity	<None>	
	Boundary Scan	<None>	

Using Quartus II- Summary

This page shows you a project summary- the project name, top level module, selected device etc. If there are mistakes, you can go back and change them.



Summary

When you click Finish, the project will be created with the following settings:

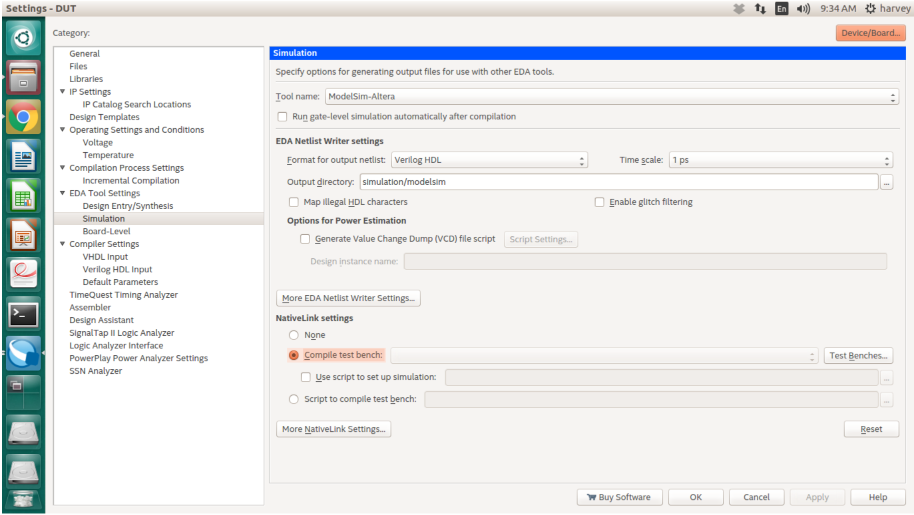
Project directory:	/home/jay/Desktop/digital_lab/reference_design/Test
Project name:	Full_Adder
Top-level design entity:	DUT
Number of files added:	3
Number of user libraries added:	0
Device assignments:	
Design template:	n/a
Family name:	MAX V
Device:	5M1270ZT144C5
Board:	n/a
EDA tools:	
Design entry/synthesis:	<None> (<None>)
Simulation:	ModelSim-Altera (Verilog HDL)
Timing analysis:	0
Operating conditions:	
VCCINT voltage:	1.8V
Junction temperature range:	0-85 °C

Using Quartus II- Analysis and Synthesis

- Once you have created project and added files start compilation.
- **Make sure that you have selected proper Top-Level entity and did the full compilation.**
- If you are getting any errors resolve them. Warning can be ignored as of now.
- Now the next step is Gate Level Simulation.

Using Quartus II- Compiling Test Bench

Add the given Test bench in Compile the Test bench Section. (i.e. Assignmenets > Settings > Simulation). Then select Test Benches and Select New.



Settings - DUT

9:34 AM harvey

Category:

- General
- Files
- Libraries
- IP Settings
  - IP Catalog Search Locations
- Design Templates
- Operating Settings and Conditions
  - Voltage
  - Temperature
- Compilation Process Settings
  - Incremental Compilation
- EDA Tool Settings
  - Design Entry/Synthesis
  - Simulation
  - Board-Level
- Compiler Settings
  - VHDL Input
  - Verilog HDL Input
  - Default Parameters
- TimeQuest Timing Analyzer
- Assembler
- Design Assistant
- SignalTap II Logic Analyzer
- Logic Analyzer Interface
- PowerPlay Power Analyzer Settings
- SSN Analyzer

Device/Board...

Simulation

Specify options for generating output files for use with other EDA tools.

Tool name: ModelSim-Altera

□ Run gate-level simulation automatically after compilation

EDA Netlist Writer settings

Format for output netlist: Verilog HDL   Time scale: 1 ps

Output directory: simulation/modelsim

□ Map illegal HDL characters   □ Enable glitch filtering

Options for Power Estimation

□ Generate Value Change Dump (VCD) file script   Script Settings...

Design instance name:

More EDA Netlist Writer Settings...

NativeLink settings

○ None

● Compile test bench: Test Benches...

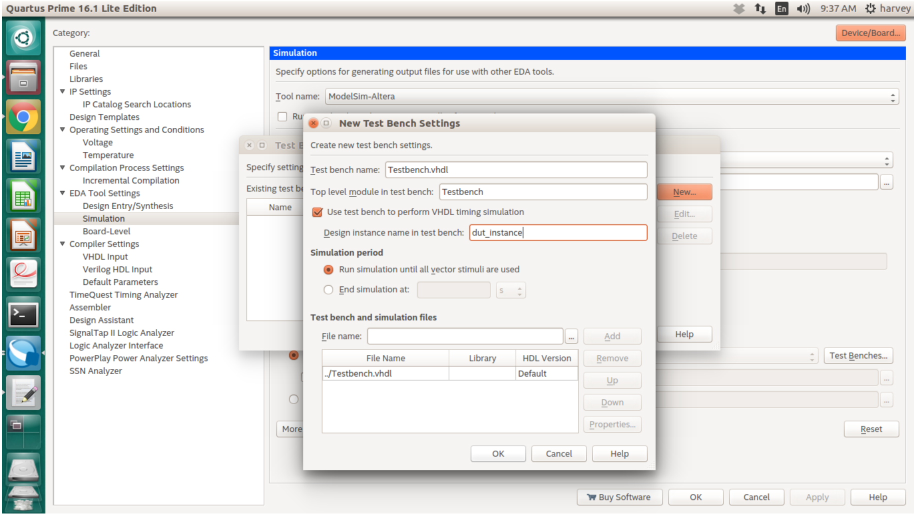
□ Use script to set up simulation:

○ Script to compile test bench:

More NativeLink Settings...   Reset

Using Quartus II- Compiling Test Bench

Add the Test bench file and specify Top level module in the test bench file. Tick the Use tench bench to perform timing simulation and select the instance of design file mentioned in test file.



Quartus Prime 16.1 Lite Edition

Simulation

Specify options for generating output files for use with other EDA tools.

Tool name: ModelSim-Altera

New Test Bench Settings

Create new test bench settings.

Test bench name: Testbench.vhdl

Top level module in test bench: Testbench

☑ Use test bench to perform VHDL timing simulation

Design instance name in test bench: dut_instance

Simulation period

● Run simulation until all vector stimuli are used

○ End simulation at:

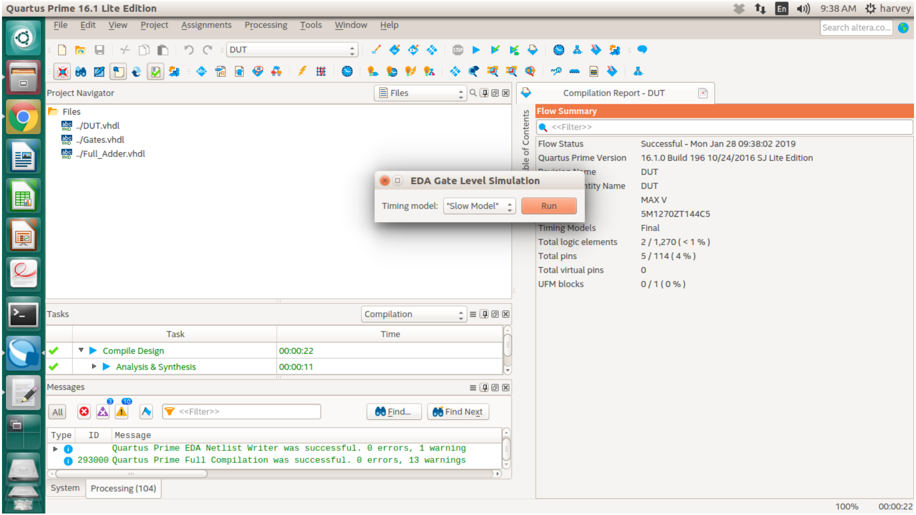
Test bench and simulation files

File name:

File Name	Library	HDL Version
../Testbench.vhdl		Default

Using Quartus II- Gate Level Simulation

Once you are done with setting up the test bench file run gate level simulation. (Tools > Run Simulation Tool > Gate Level Simulation)



Files
- ../DUT.vhdl
- ../Gates.vhdl
- ../Full_Adder.vhdl

EDA Gate Level Simulation

Timing model: "Slow Model"   Run

Flow Summary

Flow Status	Successful - Mon Jan 28 09:38:02 2019
Quartus Prime Version	16.1.0 Build 196 10/24/2016 SJ Lite Edition
Revision Name	DUT
...tity Name	DUT
	MAX V
	5M1270ZT144C5
Timing Models	Final
Total logic elements	2 / 1,270 (< 1 %)
Total pins	5 / 114 (4 %)
Total virtual pins	0
UFM blocks	0 / 1 (0 %)

Task	Time
✓ Compile Design	00:00:22
✓ Analysis & Synthesis	00:00:11

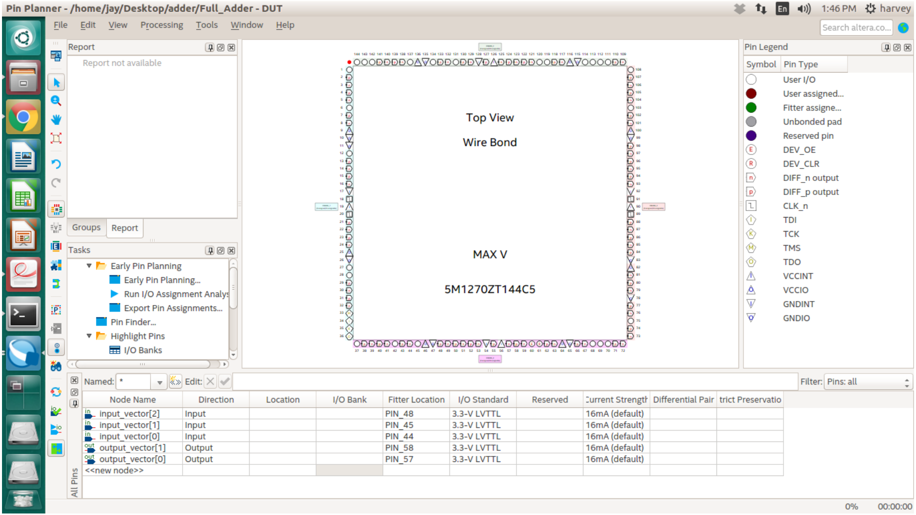
```
Type  ID     Message
           Quartus Prime EDA Netlist Writer was successful. 0 errors, 1 warning
      293000 Quartus Prime Full Compilation was successful. 0 errors, 13 warnings
```

Using Quartus II- Generating SVF file

- After successfully completing Gate level simulation, now we will generate svf file which will be loaded on the board.
- **Gate Level Simulation successfully Completed??** If no then change the settle time in the Testbench.vhdl from 1 ns to 20 ns. (Line no. 110)
- Now open pin planner from Assignments > Pin Planner or Ctrl + Shift + N.
- Please note that signals which are mentioned as input and output in the Top level entity will come in the Pin Planner.

Using Quartus II- Pin Planning

If you have followed all the steps correctly then you will be able to see all 5 signals of DUT come to list.

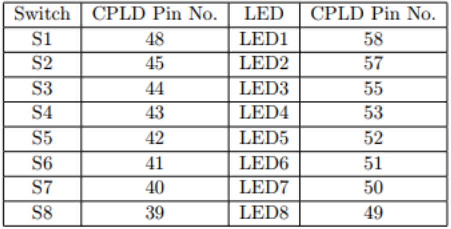


Symbol	Pin Type
○	User I/O
●	User assigned...
●	Fitter assigne...
●	Unbonded pad
●	Reserved pin
Ⓔ	DEV_OE
Ⓡ	DEV_CLR
n	DIFF_n output
p	DIFF_p output
⌊	CLK_n
⬠	TDI
⬠	TCK
⬠	TMS
⬠	TDO
△	VCCINT
△	VCCIO
▽	GNDINT
▽	GNDIO

Node Name	Direction	Location	I/O Bank	Fitter Location	I/O Standard	Reserved	Current Strengt	Differential Pair	trict Preservatio
input_vector[2]	Input			PIN_48	3.3-V LVTTL		16mA (default)		
input_vector[1]	Input			PIN_45	3.3-V LVTTL		16mA (default)		
input_vector[0]	Input			PIN_44	3.3-V LVTTL		16mA (default)		
output_vector[1]	Output			PIN_58	3.3-V LVTTL		16mA (default)		
output_vector[0]	Output			PIN_57	3.3-V LVTTL		16mA (default)		
<<new node>>									

Using Quartus II- Pin Planning

- Now we need to assign Input/Output pins to the signals. We have 3 inputs which we will assign to 3 switches and 2 outputs will be assigned to LEDs.
- For assigning pin, write pin number in Location against the respective Input/Output.
- Once pin assignments is done, close the window and compile it again (use Ctrl L).



Switch	CPLD Pin No.	LED	CPLD Pin No.
S1	48	LED1	58
S2	45	LED2	57
S3	44	LED3	55
S4	43	LED4	53
S5	42	LED5	52
S6	41	LED6	51
S7	40	LED7	50
S8	39	LED8	49

Table 2: Pin mapping for on-board Switches and LEDs

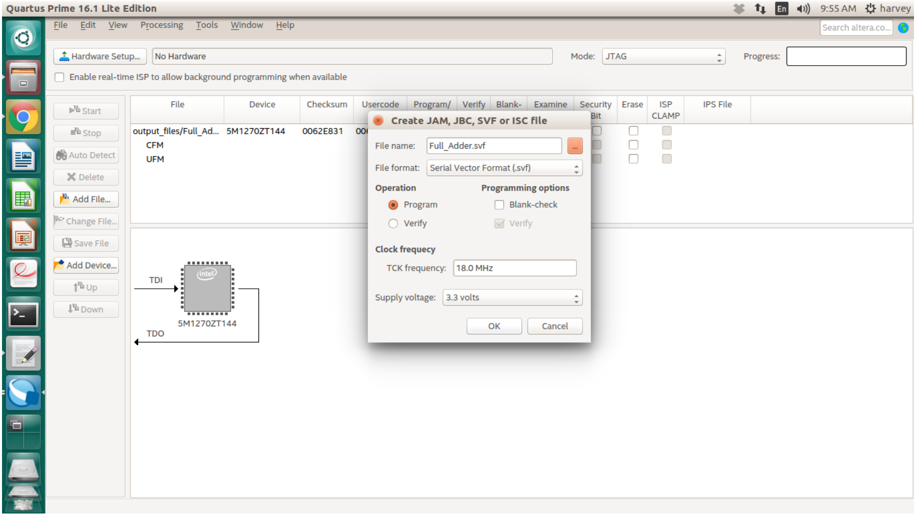
You can refer Krypton User manual for other pin configuration.

Using Quartus II- Generation of SVF file

- For generating svf (Serial Vector File) go to Tools > Programmer.
- You should be able to see your device with DUT.pof file.
- If this window is **coming blank then go to Add File** from left side panel. Then go to output_files and select .pof file.
- Once you are ready with .pof file then go to File > Create JAM,JBC,SVf or ISC file.
- Select File format as Serial Vector Format(.svf) and give File Name as Full_Adder.svf and select OK to generate .svf file.
- Generally .svf is generated under output_files folder.

Using Quartus II- Generation of SVF file

- Once .svf is generated load that file to Krypton board using UrJAG.
- Check for all 8 combinations using switches and see the correponding outputs on LEDs.



No Hardware

Mode: JTAG

Progress:

☐ Enable real-time ISP to allow background programming when available

File	Device	Checksum	Usercode	Program/	Verify	Blank-	Examine	Security Bit	Erase	ISP CLAMP	IPS File
output_files/Full_Ad...	5M1270ZT144	0062E831	00					☐	☐	☐	
CFM								☐	☐	☐	
UFM								☐	☐	☐	

Create JAM, JBC, SVF or ISC File

File name: Full_Adder.svf

File format: Serial Vector Format (.svf)

Operation
- ● Program
- ○ Verify

Programming options
- ☐ Blank-check
- ☑ Verify

Clock frequecy

TCK frequency: 18.0 MHz

Supply voltage: 3.3 volts

Four Bit Adder Experiment

- Now you should be able to check your four bit adder design on the board.
- Assign inputs to switches (4 bits 2 inputs) and outputs to LEDs (4 LEDs).
- Check for the correct functionality of adder by giving at least 10 different inputs and observing the status of LEDs.