EEDG/CE 6370
Design and Analysis of Reconfigurable Systems
Homework 4
Physical Design and Timing

1. Laboratory Objectives

- Understand how the physical design back-end of the FPGA tool flow.
- Learn how to control the placement through Chip planner.
- Understand the impact of the placement on the timing of the circuit.
- Learn how to use a batch script to configure the FPGA directly without requiring Quartus.

2. Summary

In this lab you will learn to use Quartus's Timing analyzer and Chip planner and learn how to modify the default FPGA placement

3. Pre-lab

- Review the lecture slides that covers physical design

4. Tool Requirements

- Quartus Prime including Timing analyzer and Chip planner (built into Quartus)

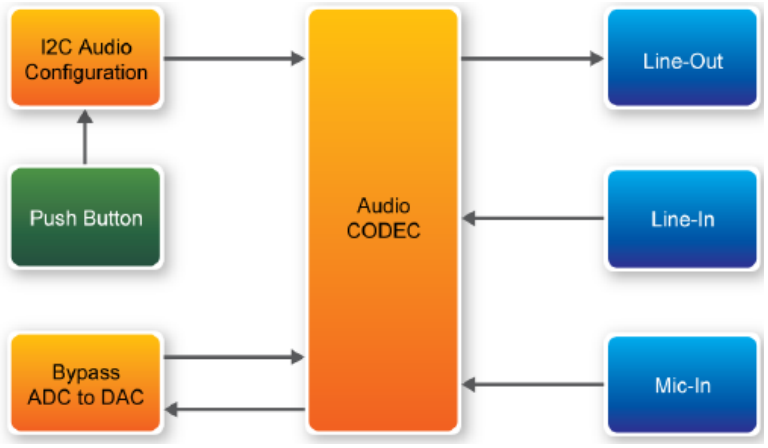
5. Preliminaries

- Most of the delay in modern FPGAs is due to the interconnect network of the FPGA. Therefore we will not know the actual delay until the synthesized circuit has been placed and routed onto the FPGA.
- To understand the placement and routing FPGA vendors provide tools that allow user to visualize and edit the design. In the case of Intel, this tool is called Chip planner

6. System Overview -Karaoke Machine

The given design is based on Terasic's demonstration programs.

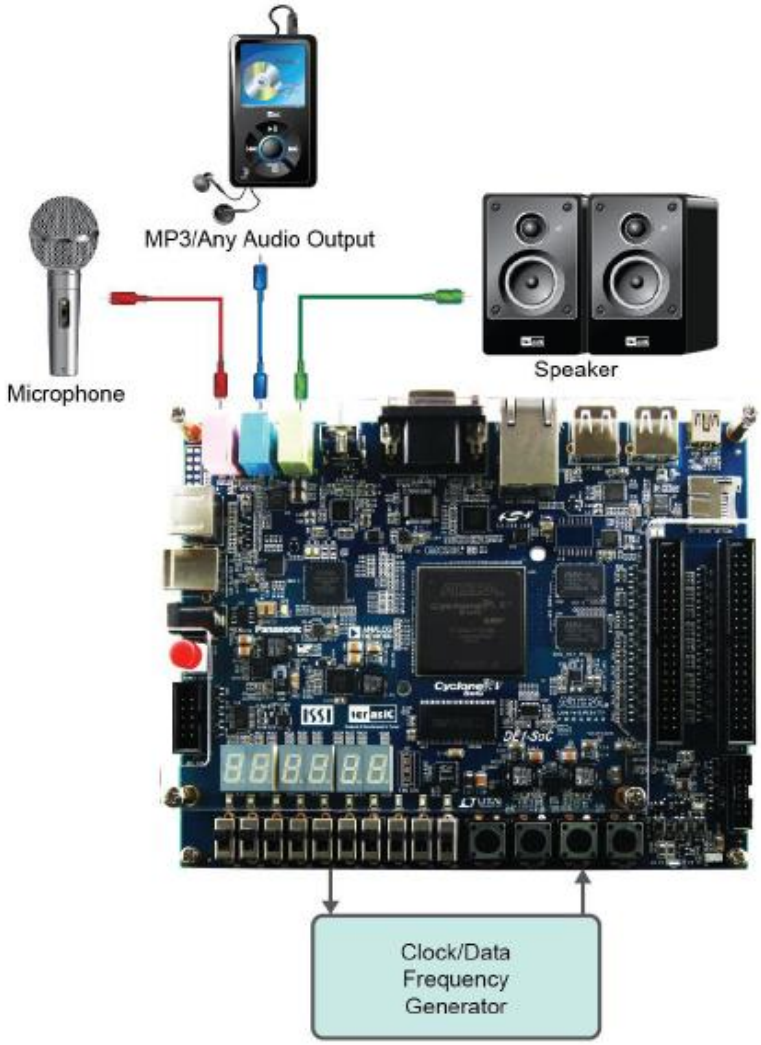
This demonstration uses the microphone-in, line-in, and line-out ports on DE1-SoC to create a Karaoke machine. The WM8731 CODEC is configured in master mode. The audio CODEC generates AD/DA serial bit clock (BCK) and the left/right channel clock (LRCK) automatically. The
I2C interface is used to configure the audio CODEC, as shown in figure below. The sample rate and gain of the CODEC are set in a similar manner, and the data input from the line-in port is then mixed with the microphone-in port. The result is sent out to the line-out port. The sample rate is set to 48 kHz in this demonstration. The gain of the audio CODEC is reconfigured via I2C bus by pressing the pushbutton KEY0, cycling within ten predefined gain values (volume levels) provided by the device.



Project directory: DE1_SOC_i2sound includes:

1. Source code to be compiled
2. SDC file
3. QSF file
4. Batch file to configure the FPGA (.bat) and the already generated .sof file (the batch file program the FPGA with the sof file directly without requiring to open Quartus).

1. Connect a microphone to the microphone-in port (pink color)
2. Connect the audio output of a music player, such as a MP3 player or computer, to the line-in port (blue color)
3. Connect a headset/speaker to the line-out port (green color)
4. Load the bitstream into the FPGA by executing the batch file ‘DE1_SOC_i2sound’ in the directory DE1_SOC_i2sound\demo_batch
5. You should be able to hear a mixture of microphone sound and the sound from the music player.
6. Press KEY0 to adjust the volume; it cycles between volume level 0 to 9



Setup for Karaoke Machine

7. Programming the DE1-SoC FPGA in batch mode

Quartus Prime is an IDE that calls different programs when clicking on any of the buttons. You can see all of these programs at:

C:\intelFPGA_lite\22.1std\quartus\bin64

- Open the De1_SoC_i2sound.bat file with any text editor and see how the "quartus_pgm.exe" binary is used to program the FPGA.
- Run the batch file with the FPGA board connected to your computer and check that the programming was done correctly. Store this script in order to program the FPGA file directly without the need to use Quartus once you have the .sof file.
- If you get any errors, you might:

1. Check that the QUARTUS_BIN path is set in the environment variables in windows.
2. Run the script in admin mode.
3. Make sure that you install the USB blaster to program the FPGA during the installation.

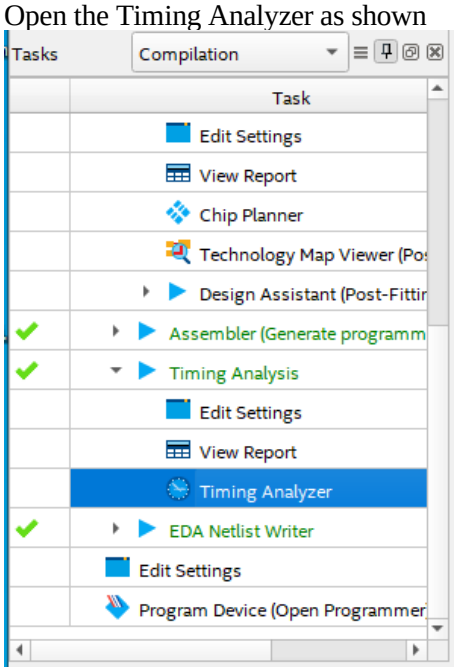
8. Synthesizing the Desing using Quartus Prime

- Open Quartus Prime (right click in icon)

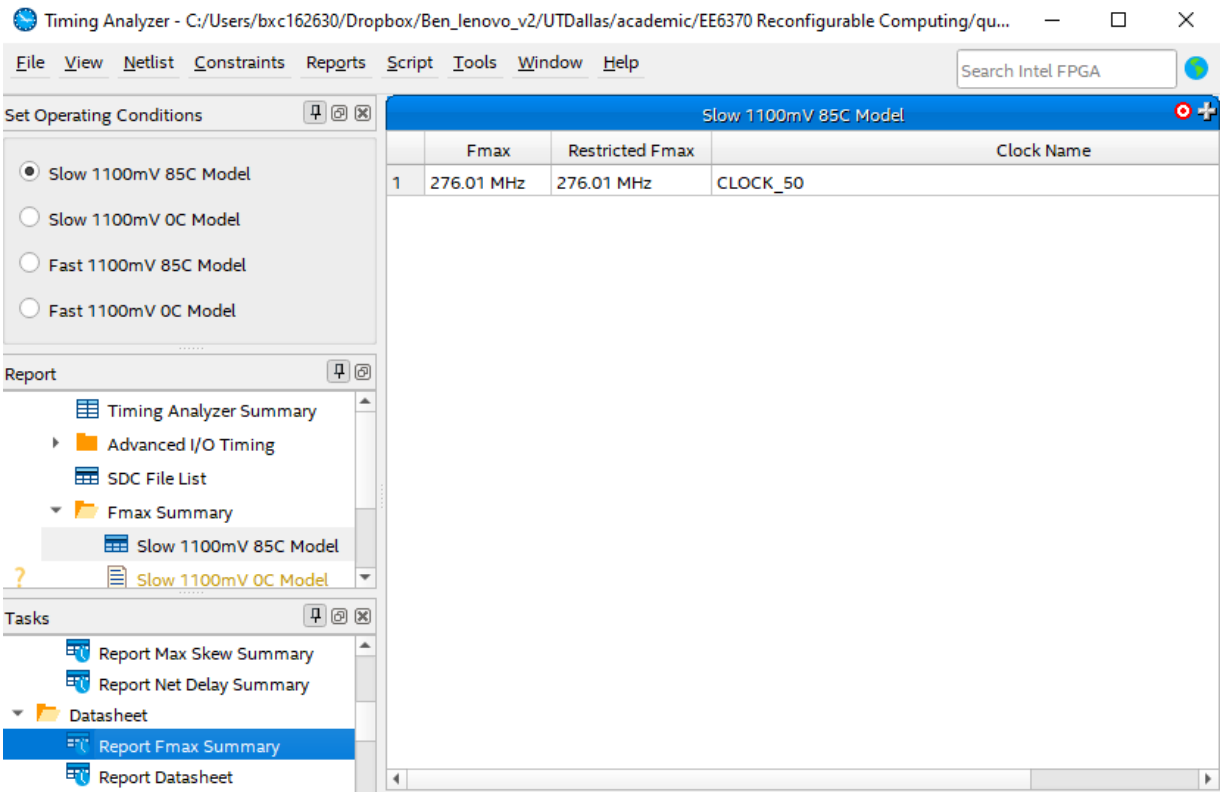
In case of Windows run the tools always as administrator

- Create a new project.
  File ➔ New Project Wizard → DE1SoC_i2sound
- Click next → empty project →
- Click next (do not add any files)
- Select Cyclone V 5CSEMA5F31C6 device.
- Click next and Finish.
- Add the given Verilog files to the project:
  1. DE1_SoC_i2sound.v
  2. i2c.v
  3. keytr.v
  - Add the given sdc file to the project (DE1_SoC_i2sound.sdc)
  - Overwrite the .qsf file generated with the .qsf given. This basically include the pin assignments for all the peripherals that are used by this project (Leave the path to the Verilog files !)
  - Compile the entire project and check that there are no errors

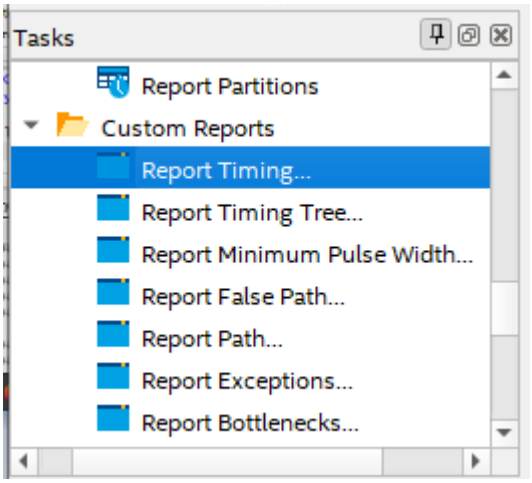
  - Open the Timing Analyzer as shown

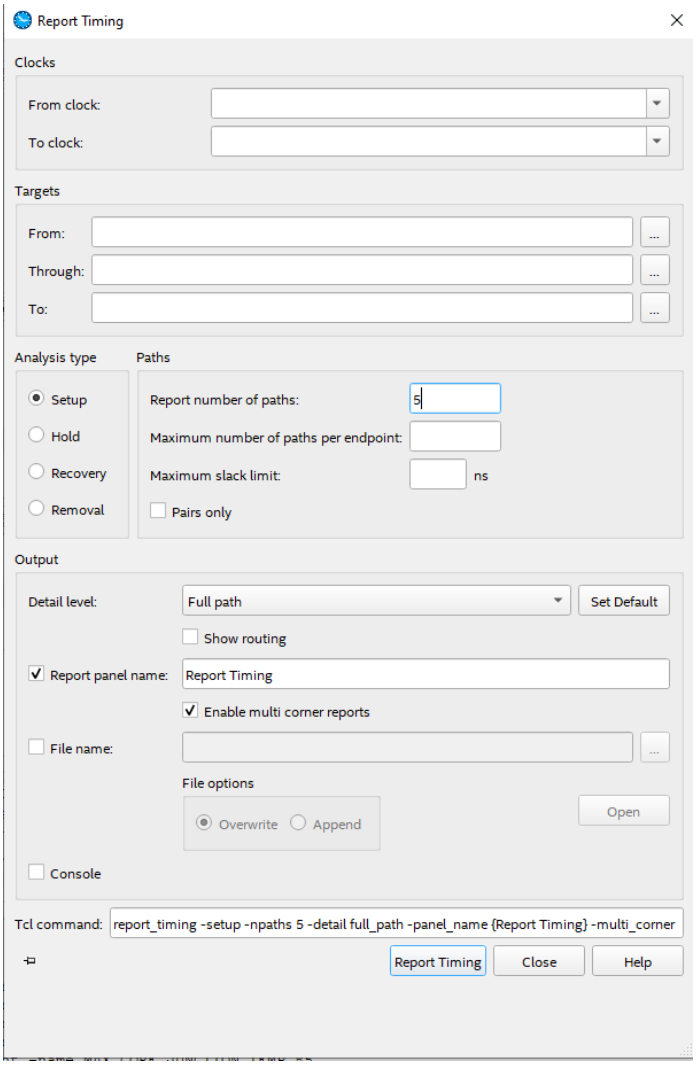


  - Select the “report Fmax Summary” to show the maximum frequency at which the circuit can be clocked. The frequency should be hire than the clock requirements specified in the .sdc file

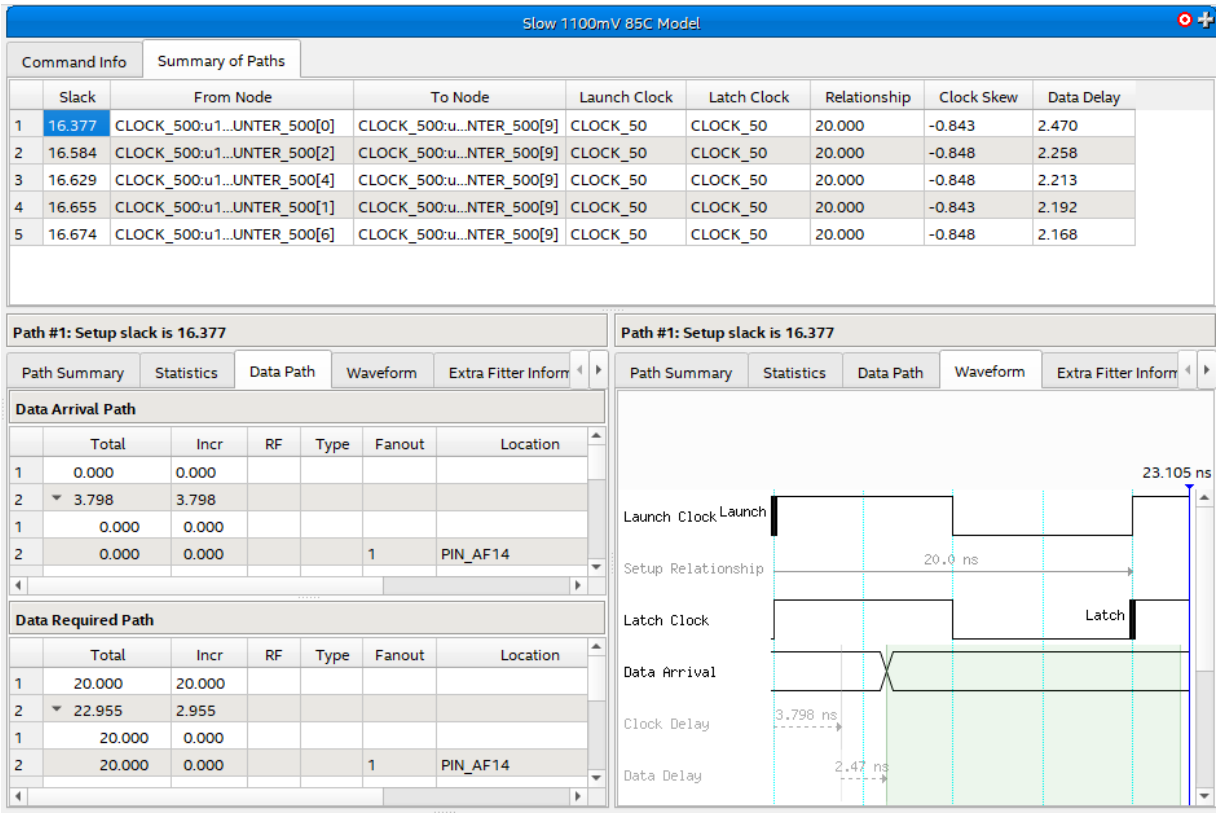


- Highlight the 5 longest critical paths. Report Timing → Change Report number of paths to 5 → Report timing

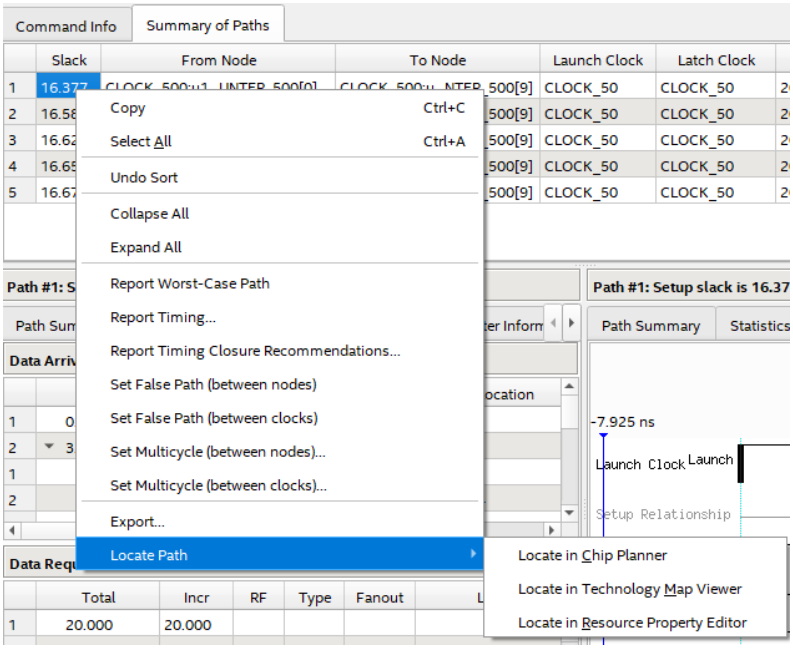




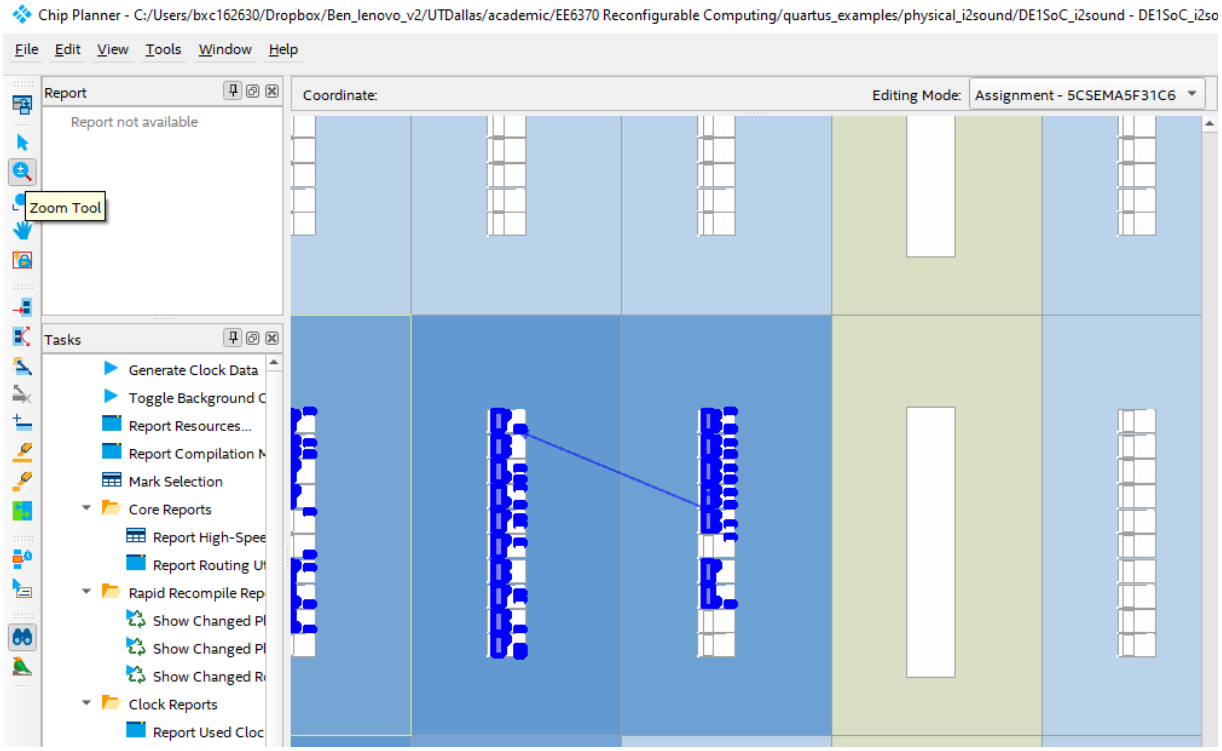
- The 5 longest paths will be listed in a new window, including the slack. The longest timing path will have the smallest slack.
  Remember: Positive slack means that the timing is met, while negative slack means that the timing is not met!



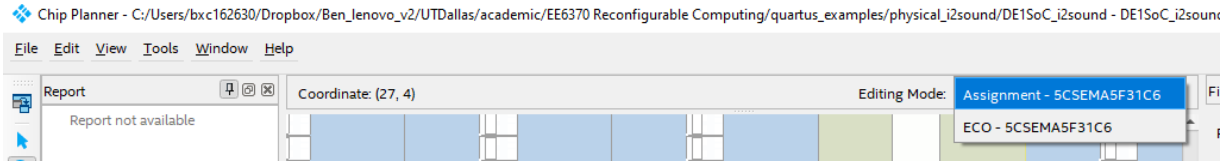
- The timing analyzer allows you to visually see where the critical path is located by clicking on the path entry→ right click →Locate Path



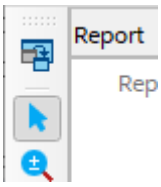
- Open the Chip planner from the Timing analyzer in order to visualize the placed and routed circuit including the critical paths. Locate Path → Locate in Chip planner.
- Zoom in and locate the critical path as shown.



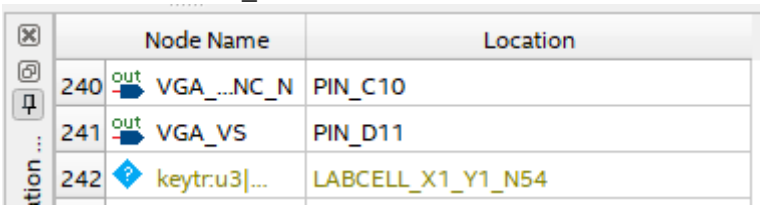
- Modify the Chip planners view to assignment mode as shown below.



- Select one of the LUT or register in the critical oath and move it to the lower left corner of the FPGA. For this select the 'arrow' option → Click on LUT or register and draw all the way to the top corner.

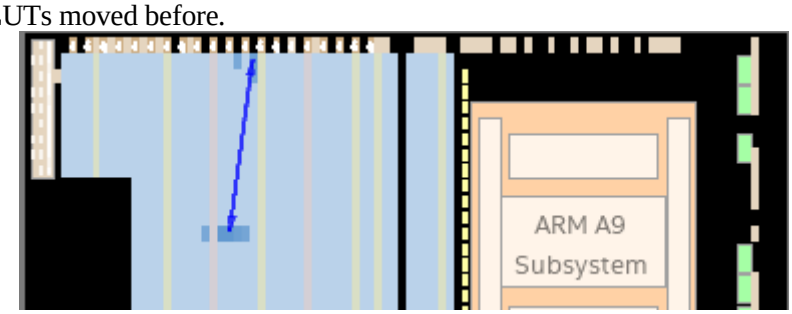


- The location pane in Chip planer should show the new location of the instanced moved. In the case of the top part of the FPGA : X1_Y1



- Move another instances of the design to the top of the FPGA: X<num>:Y80
- Close the Chip planner. You will see that the location of these new LUTs has been annotated at the end of the .qsf file

- Re-compile the entire design with the new placement.
- Check the new timing by re-opening the Timing analyzer
- Visualize the new critical path in Chip planner. and check if the critical path is now part of one of the two LUTs moved before.



Answer the question in the report sheet.

UT Dallas Honor Code

UT Dallas values academic integrity. Therefore, all students must understand the meaning and consequences of cheating, plagiarism and other academic offences under the Code of Student Conduct and Disciplinary Procedures.

Group assignments must be completed solely by the members of the group. Cross-group work is not allowed. Moreover, similar assignments have been offered before at UT Dallas and other universities. Any use of information from previous assignments is prohibited. The tutorials are to be taken individually. Failure to respect this rule constitutes dishonesty and is a direct violation of the University Honor Code.

[END]