LAB 2 – SEVEN SEGMENT DISPLAY

TO DESIGN A 7-SEGMENT DISPLAY USING QUARTUS II SOFTWARE.

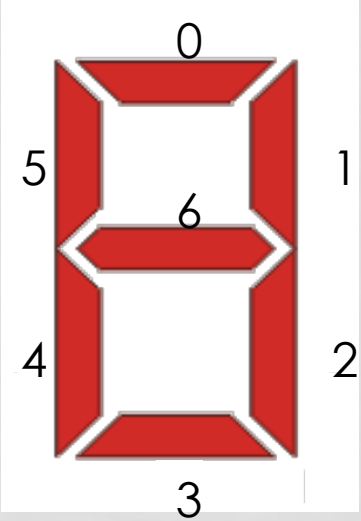
IMPLEMENTATION

Deliverables:

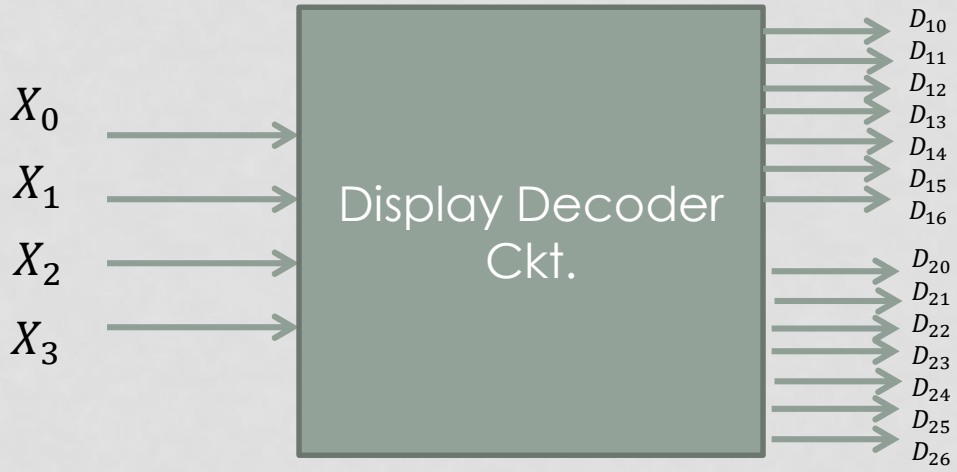
1. Implement a circuit that displays the decimal number (0 to 15) on two 7 segment displays from 4 bit binary input.

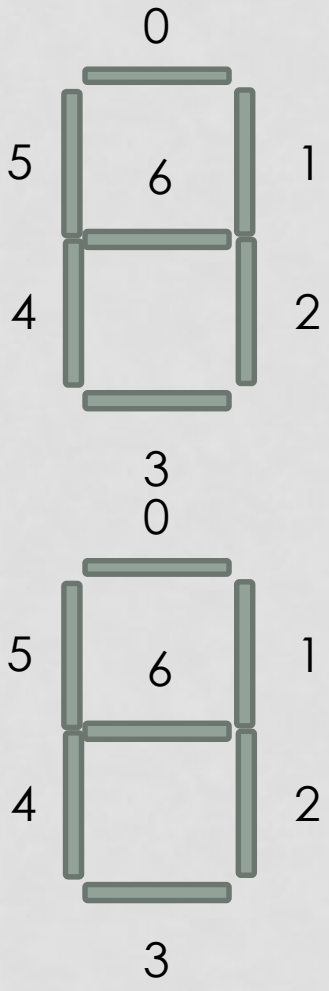
7 SEGMENT DISPLAY

- 7 segment display has 7 LEDs placed in the following manner.
- It can display the integer by simply turning on and off the desired LEDS.



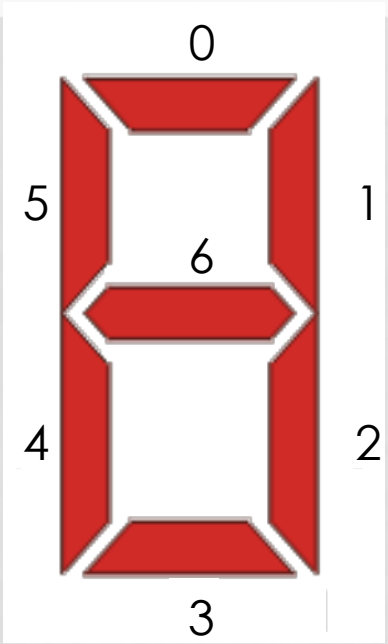
DISPLAY CKT


X_0
X_1
X_2
X_3
Display Decoder Ckt.
D_{10}
D_{11}
D_{12}
D_{13}
D_{14}
D_{15}
D_{16}
D_{20}
D_{21}
D_{22}
D_{23}
D_{24}
D_{25}
D_{26}


0
5
1
6
4
2
3
Display 1
0
5
1
6
4
2
3
Display 2

7 SEGMENT DISPLAY PATTERNS AND TRUTH TABLE

	Input $X_3X_2X_1X_0$	Display 1 $D_{10}D_{11}D_{12}D_{13}D_{14}D_{15}D_{16}$	Display 2 $D_{20}D_{21}D_{22}D_{23}D_{24}D_{25}D_{26}$
0	0 0 0 0	1 1 1 1 1 1 0	1 1 1 1 1 1 0
1	0 0 0 1	1 1 1 1 1 1 0	
2	0 0 1 0	1 1 1 1 1 1 0	
3	0 0 1 1	1 1 1 1 1 1 0	
4	0 1 0 0	1 1 1 1 1 1 0	
5	0 1 0 1	1 1 1 1 1 1 0	
6	0 1 1 0	1 1 1 1 1 1 0	
7	0 1 1 1	1 1 1 1 1 1 0	
8	1 0 0 0	1 1 1 1 1 1 0	
9	1 0 0 1	1 1 1 1 1 1 0	
10	1 0 1 0	0 1 1 0 0 0 0	
11	1 0 1 1	0 1 1 0 0 0 0	
12	1 1 0 0	0 1 1 0 0 0 0	
13	1 1 0 1	0 1 1 0 0 0 0	
14	1 1 1 0	0 1 1 0 0 0 0	
15	1 1 1 1	0 1 1 0 0 0 0	



K-MAP

- We have 4 inputs ($X_3X_2X_1X_0$) and 14 outputs (7 for each display).
- Use mux or gates (your choice) to design the Boolean function for each output.

K-Map for D_{10}

X_1X_0 \ X_3X_2	00	01	11	10
00	1	1	0	1
01	1	1	0	1
11	1	1	0	0
10	1	1	0	0

$$D_{10} = \overline{X_3} + \overline{X_2}\,\overline{X_1}$$

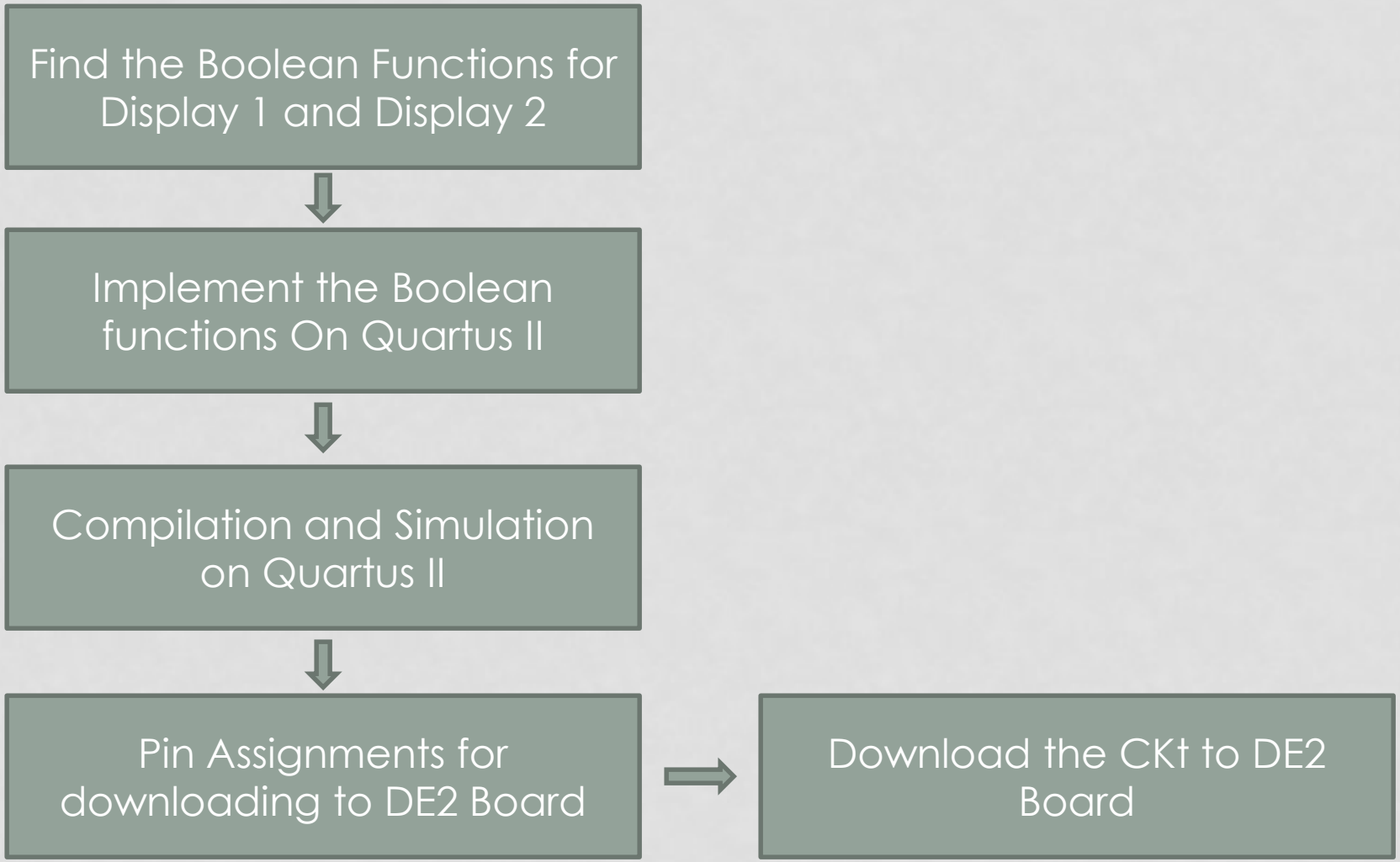
- Continue for D_{11} to D_{26}.

	Input $X_3X_2X_1X_0$	Display 1 $D_{10}D_{11}D_{12}D_{13}D_{14}D_{15}D_{16}$	Display 2 $D_{20}D_{21}D_{22}D_{23}D_{24}D_{25}D_{26}$
0	0 0 0 0	1 1 1 1 1 1 0	1 1 1 1 1 1 0
1	0 0 0 1	1 1 1 1 1 1 0	
2	0 0 1 0	1 1 1 1 1 1 0	
3	0 0 1 1	1 1 1 1 1 1 0	
4	0 1 0 0	1 1 1 1 1 1 0	
5	0 1 0 1	1 1 1 1 1 1 0	
6	0 1 1 0	1 1 1 1 1 1 0	
7	0 1 1 1	1 1 1 1 1 1 0	
8	1 0 0 0	1 1 1 1 1 1 0	
9	1 0 0 1	1 1 1 1 1 1 0	
10	1 0 1 0	0 1 1 0 0 0 0	
11	1 0 1 1	0 1 1 0 0 0 0	
12	1 1 0 0	0 1 1 0 0 0 0	
13	1 1 0 1	0 1 1 0 0 0 0	
14	1 1 1 0	0 1 1 0 0 0 0	
15	1 1 1 1	0 1 1 0 0 0 0	

	Input $X_3X_2X_1X_0$	Display 1 $D_{10}D_{11}D_{12}D_{13}D_{14}D_{15}D_{16}$	Display 2 $D_{20}D_{21}D_{22}D_{23}D_{24}D_{25}D_{26}$
0	0 0 0 0	1 1 1 1 1 1 0	1 1 1 1 1 1 0
1	0 0 0 1	1 1 1 1 1 1 0	
2	0 0 1 0	1 1 1 1 1 1 0	
3	0 0 1 1	1 1 1 1 1 1 0	
4	0 1 0 0	1 1 1 1 1 1 0	
5	0 1 0 1	1 1 1 1 1 1 0	
6	0 1 1 0	1 1 1 1 1 1 0	
7	0 1 1 1	1 1 1 1 1 1 0	
8	1 0 0 0	1 1 1 1 1 1 0	
9	1 0 0 1	1 1 1 1 1 1 0	
10	1 0 1 0	0 1 1 0 0 0 0	
11	1 0 1 1	0 1 1 0 0 0 0	
12	1 1 0 0	0 1 1 0 0 0 0	
13	1 1 0 1	0 1 1 0 0 0 0	
14	1 1 1 0	0 1 1 0 0 0 0	
15	1 1 1 1	0 1 1 0 0 0 0	

- D_{10} has the same output as D_{13}, D_{14} *and* D_{15}.
- D_{11} *and* D_{12} are always '1'.
- D_{16} is always '0'.
- Complete the truth table for Display 2 and drive Boolean logic for display 2.

ASSIGNMENT FLOW CHART



QUARTUS II

- Use Quartus II software to implement the display circuit.
- Quartus II is a programmable logic development systems.
- Allows to
  - create and process the digital circuit.
  - Compilation, simulation and
  - Programming to FPGA device.
- You can use Lab Computer or download Quartus II student version from (https://www.altera.com).

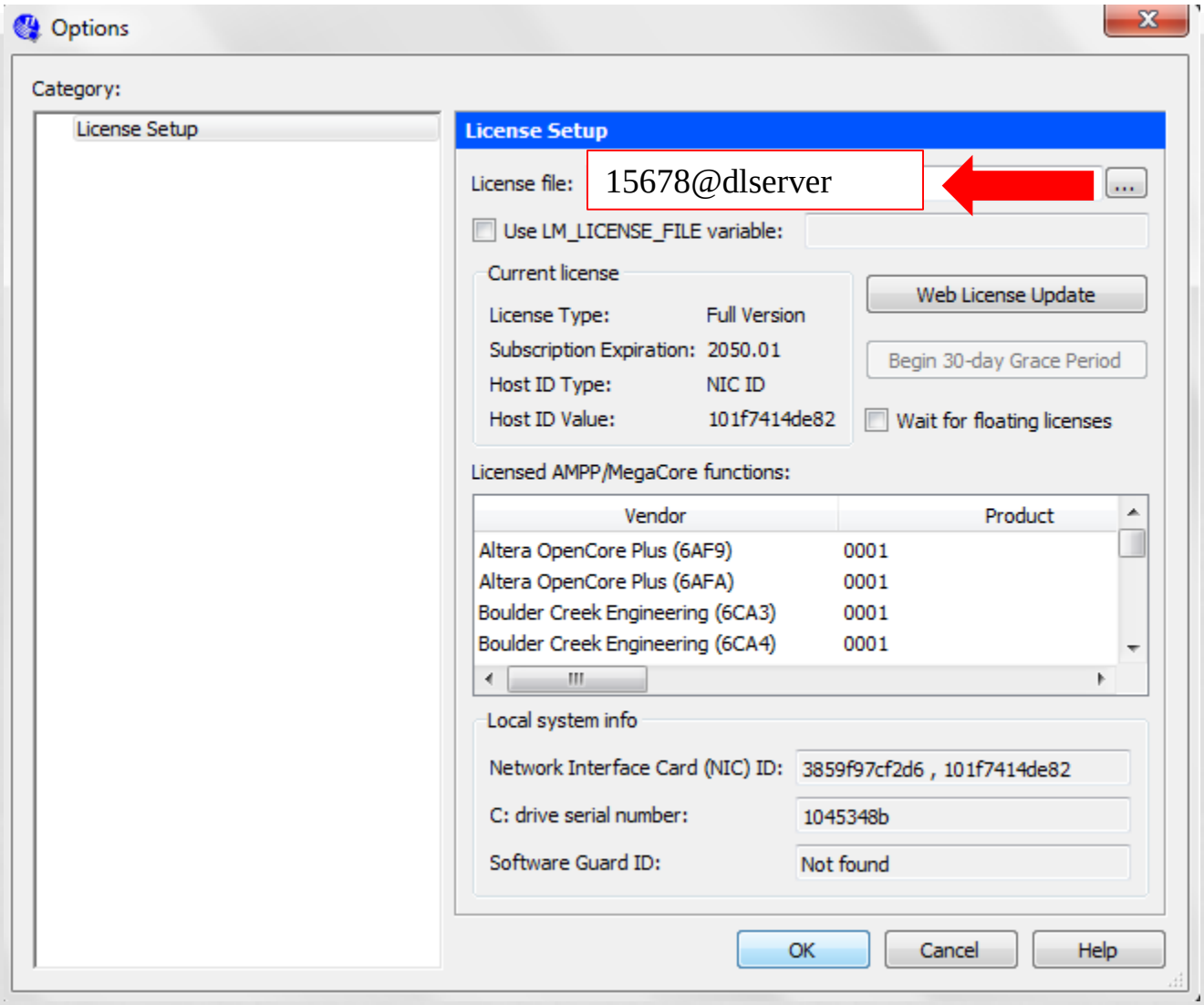
ACCESSING LAB COMPUTER

- 10 user accounts for 315 Lab.
- Make sure you are signing in DigitalLab Domain.
- Username is DIGITALLAB\ECE315GroupX (X = {1,…,10}).
- Password is Qwerty123$$
- Once you sign in you will be asked to change password.
- Lab computers do not have internet.
- You need to have USB to save your work once you are done.

SETTING UP QUARTUS LICENSE

- First create a new folder in a desktop to save your project.
- Do not use the Quartus II default project directory as your project directory!!!!!!!!!!!!!
- Start the Quartus II software and make sure you are on License version.
- If you are not on License version, your project wont compile properly.
- You can check the license from
- Tools > License Steup.

SETTING UP QUARTUS LICENSE

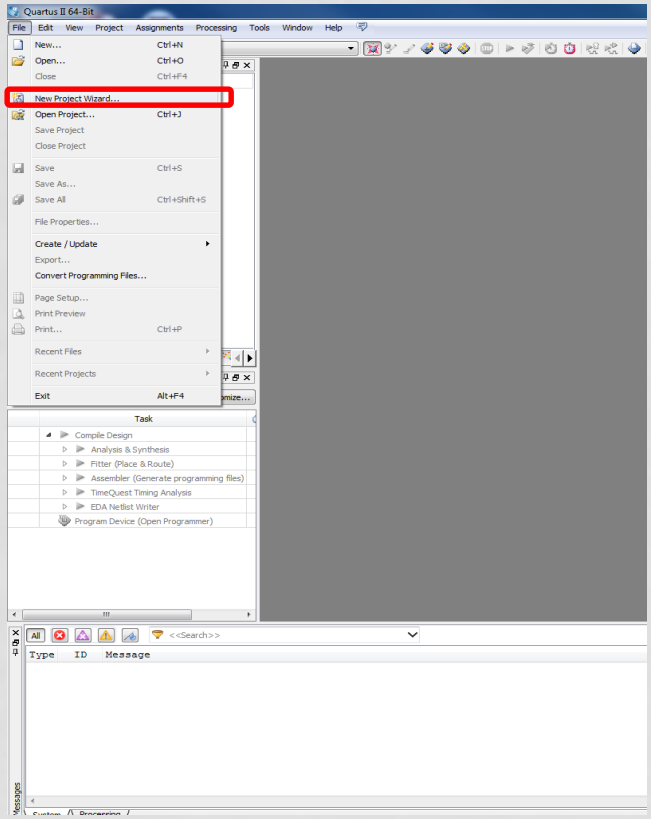


CREATING QUARTUS II PROJECT

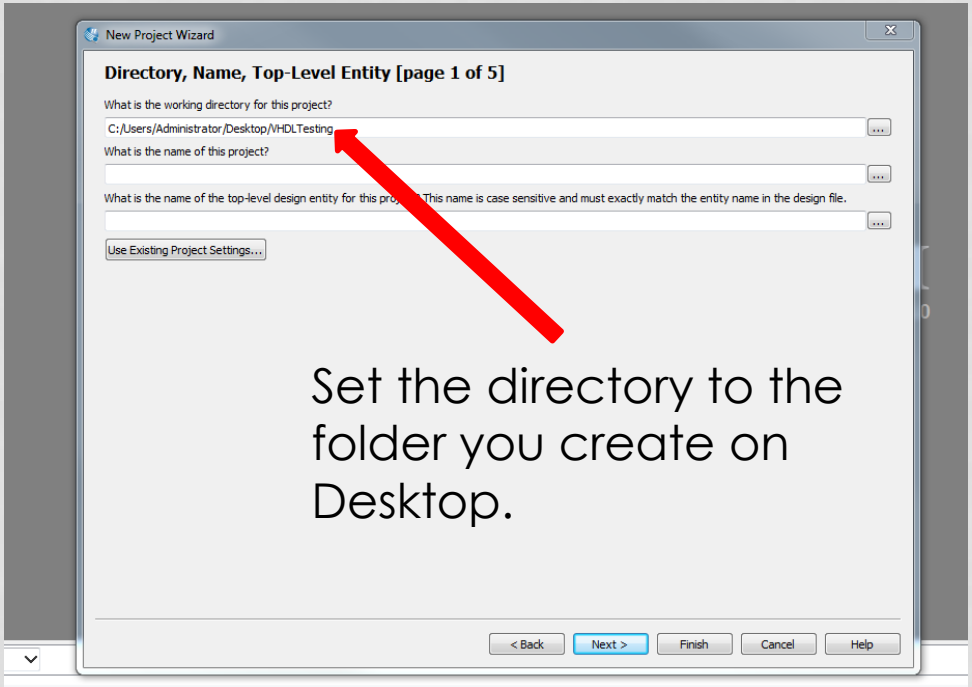
- Once you are on License version, you can start implementing your circuit.
- Please follow the Quartus II tutorial pdf on BB for circuit implementation and Simulation for details.
- You should start Quartus II after you get the Boolean expressions for all the outputs.

CREATING CKT ON QUARTUS

1. File -> New Project Wizard



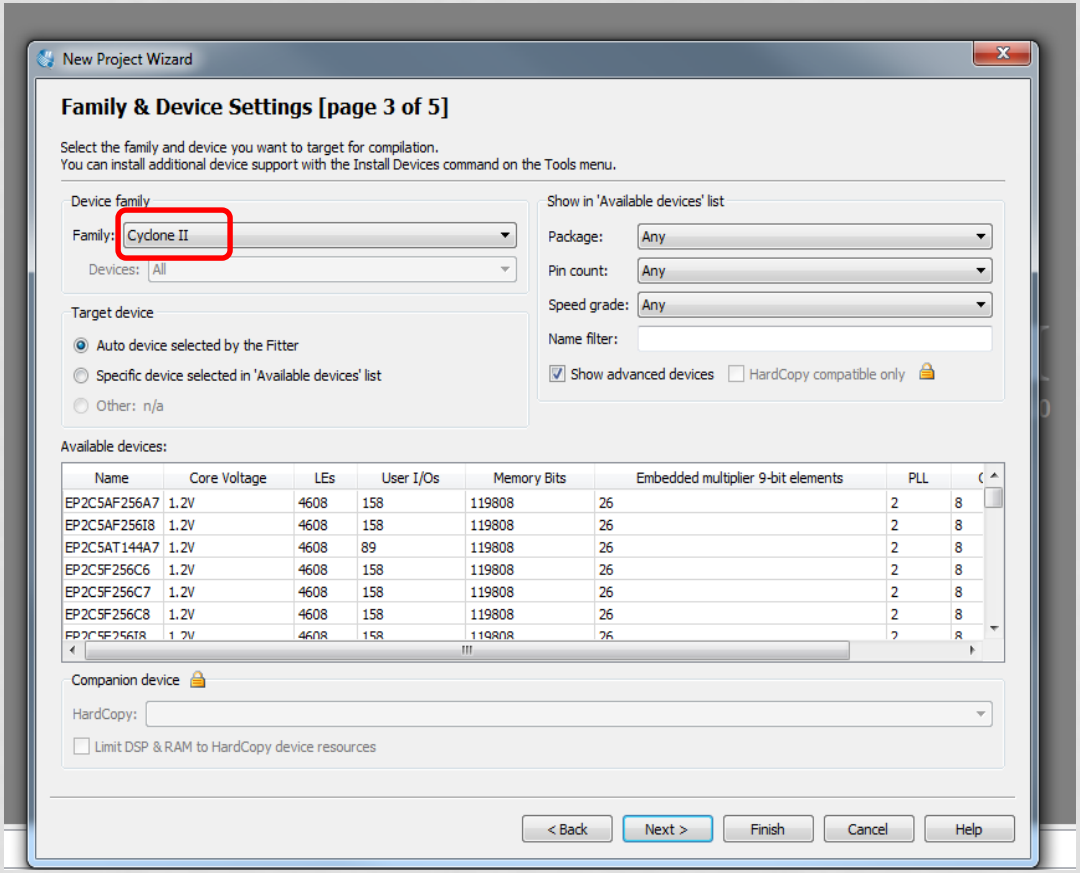
2. Select the project directory.



Set the directory to the folder you create on Desktop.

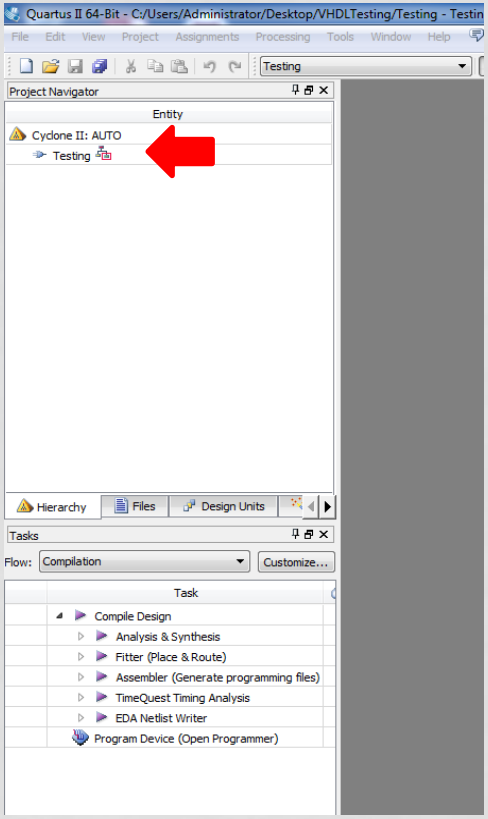
CREATING CKT ON QUARTUS

3. From 2. Click Next till you see the below window. Select the device **EP2C35F672C6.**

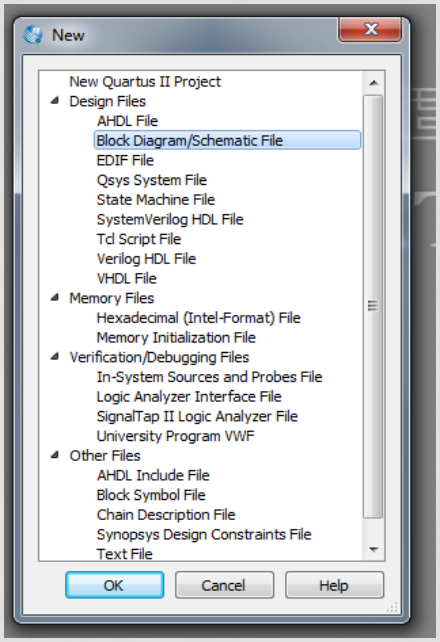


CREATING CKT ON QUARTUS

Once the project has be created, you will see it in Project Navigator window.

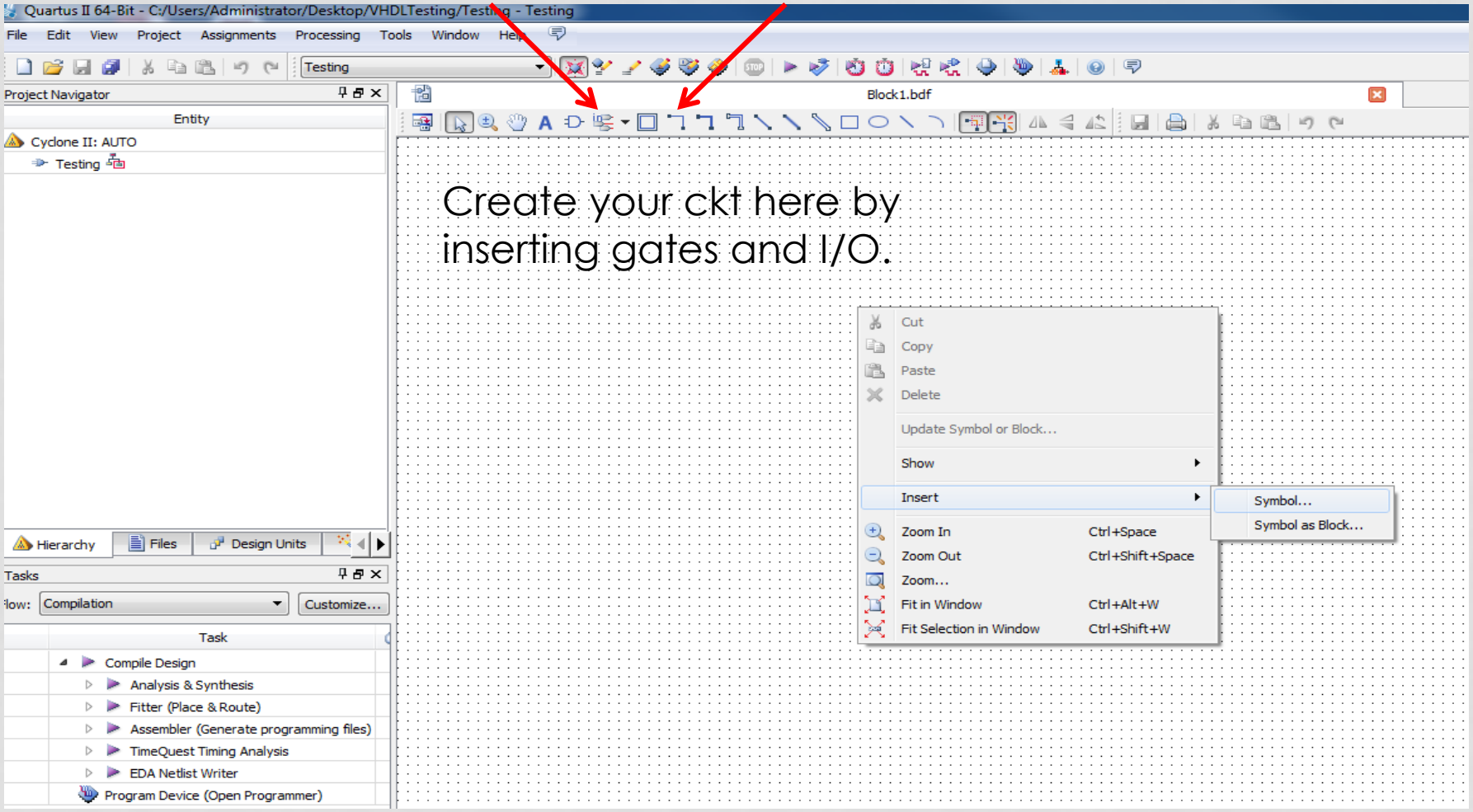


4. Click New-> Block Diagram/Schematic File.

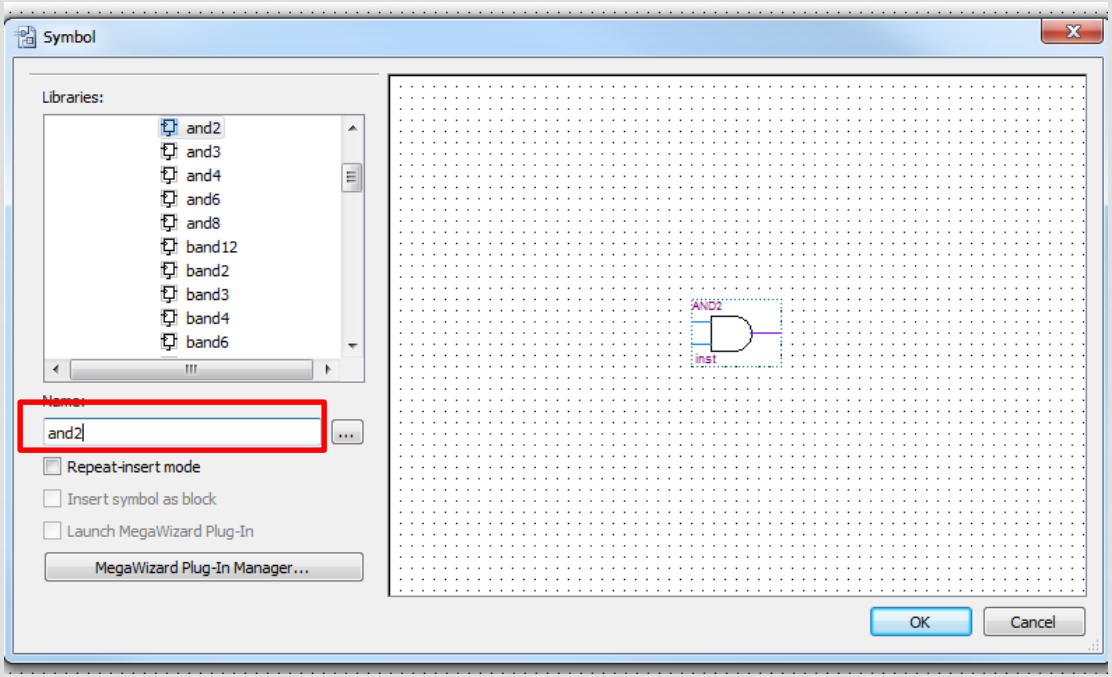


CREATING CKT ON QUARTUS

Input/Output Connector

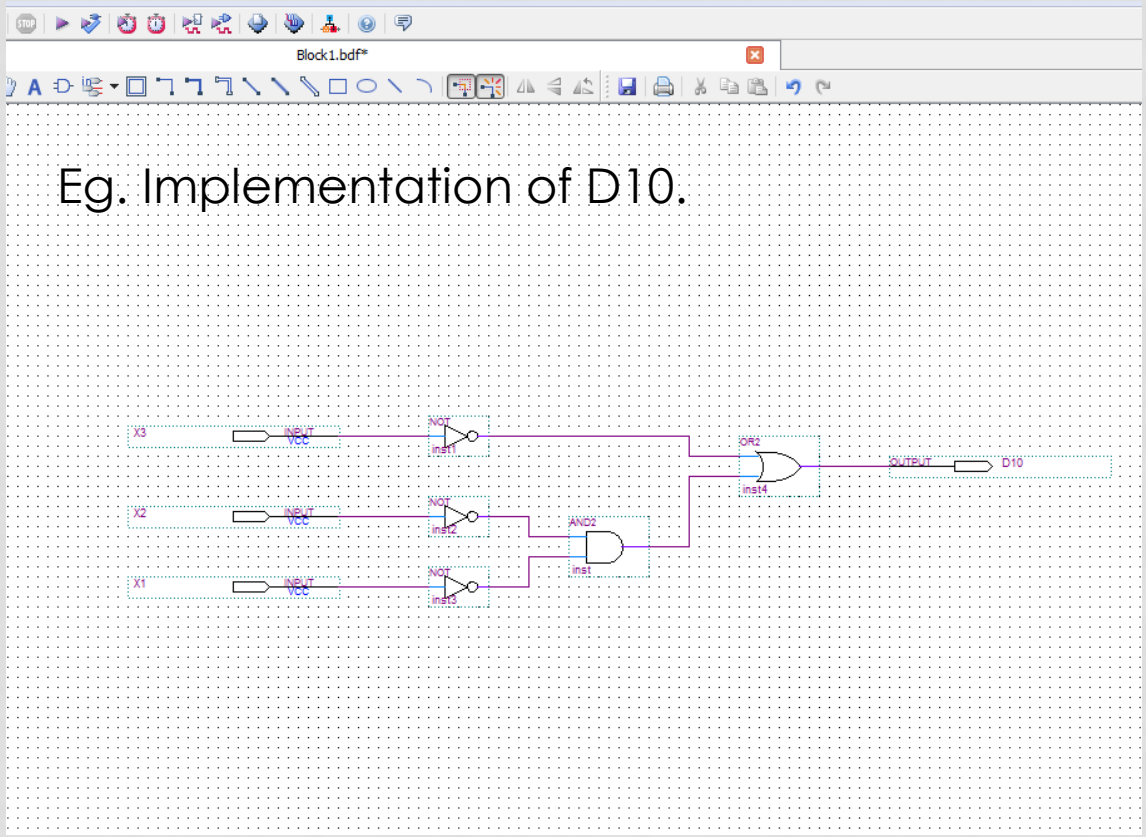


5. You can add the logic components to your ckt form Insert -> Symbols . For example, 7474 Dual D FF.



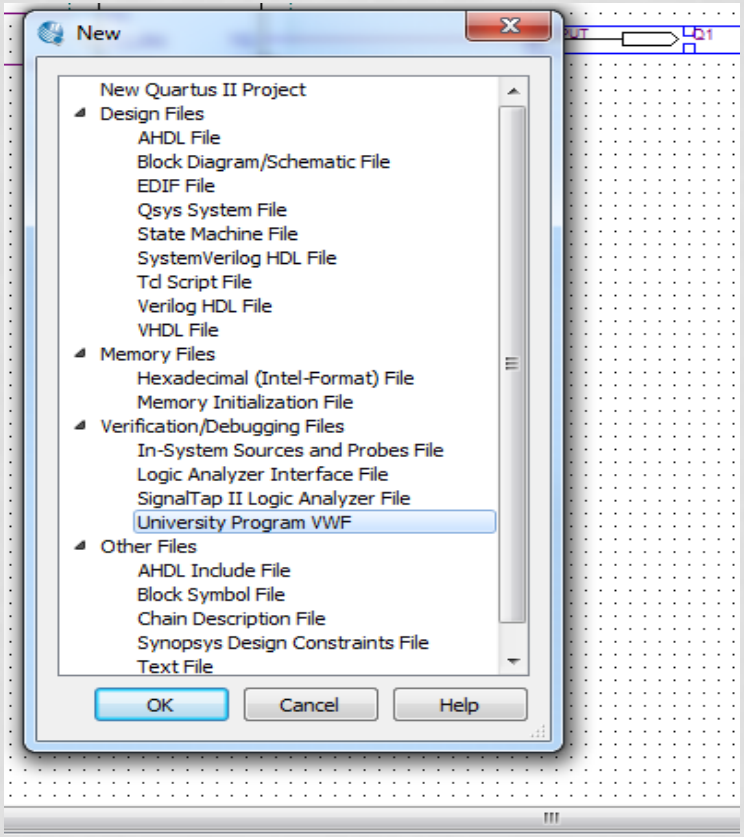
COMPILATION

5. One finish connecting all the components, compile your ckt. Processing -> Start Compilation.

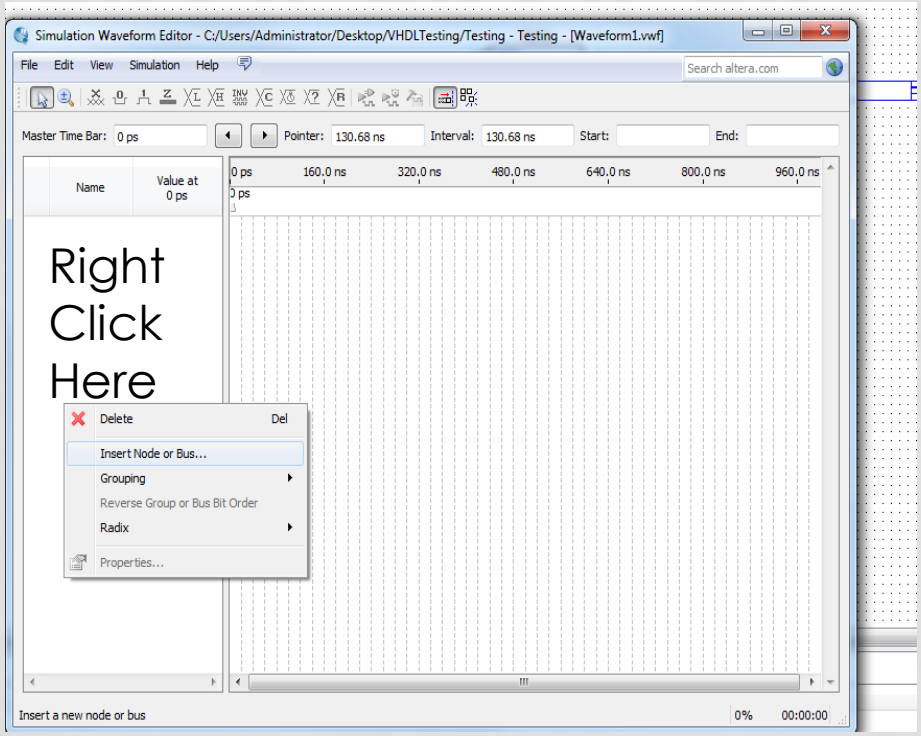


SIMULATION

When compilation is successful, create a waveform vector file for simulation.

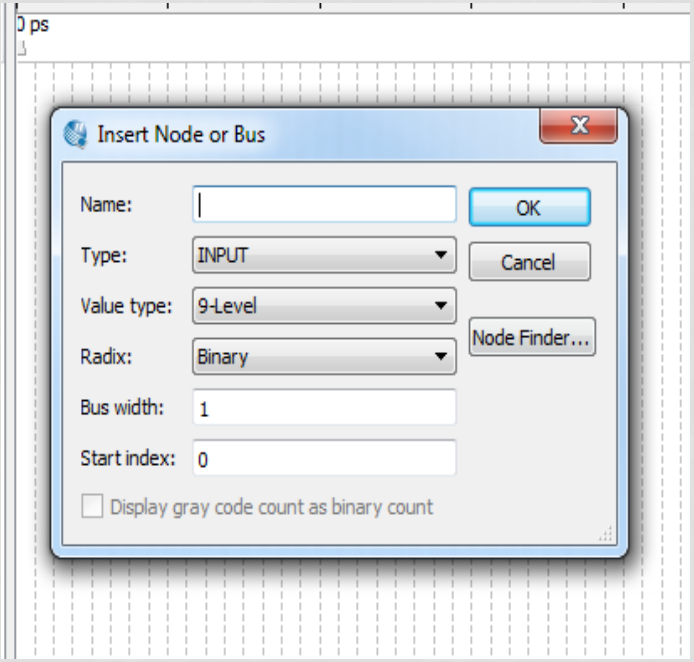


Add input and output pin the file. From Right Click, select Insert Node or Bus.

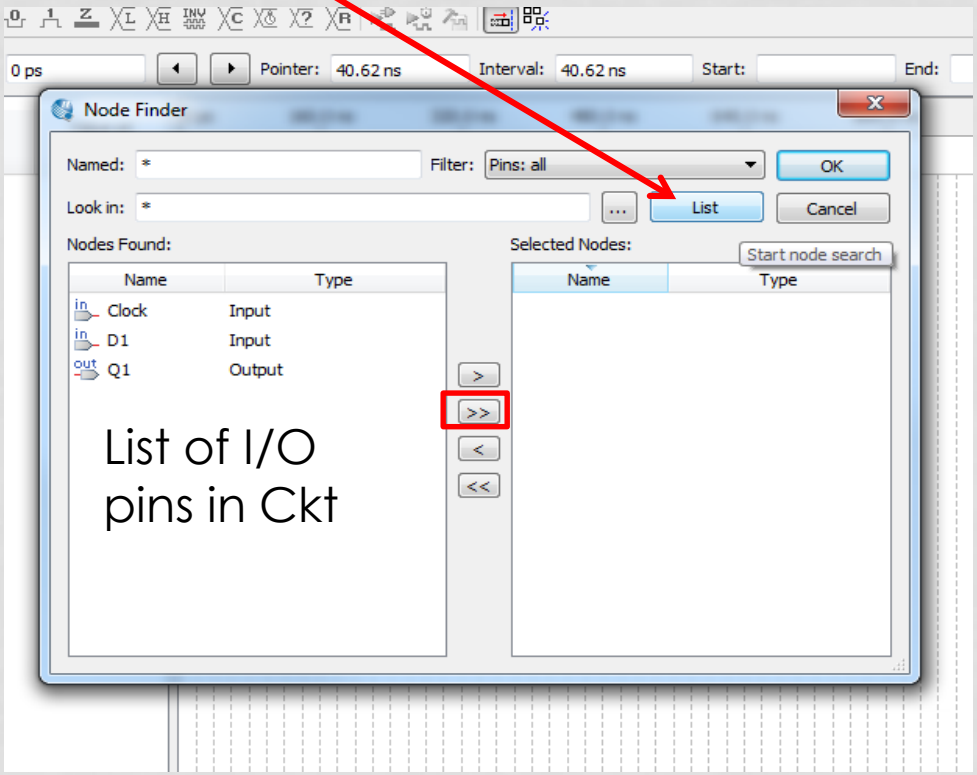


SIMULATION

Click on Node Finder

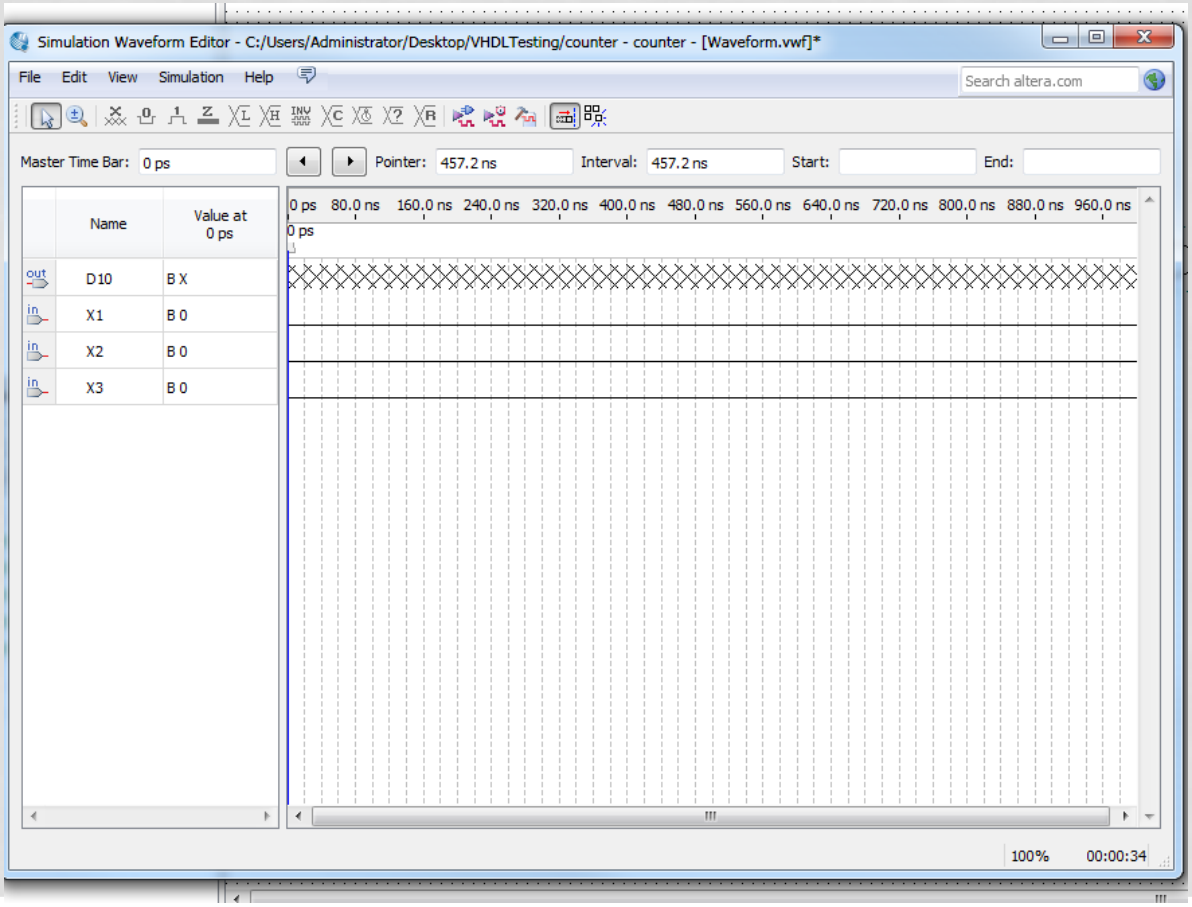


Click List. Add the I/O pin to the file.



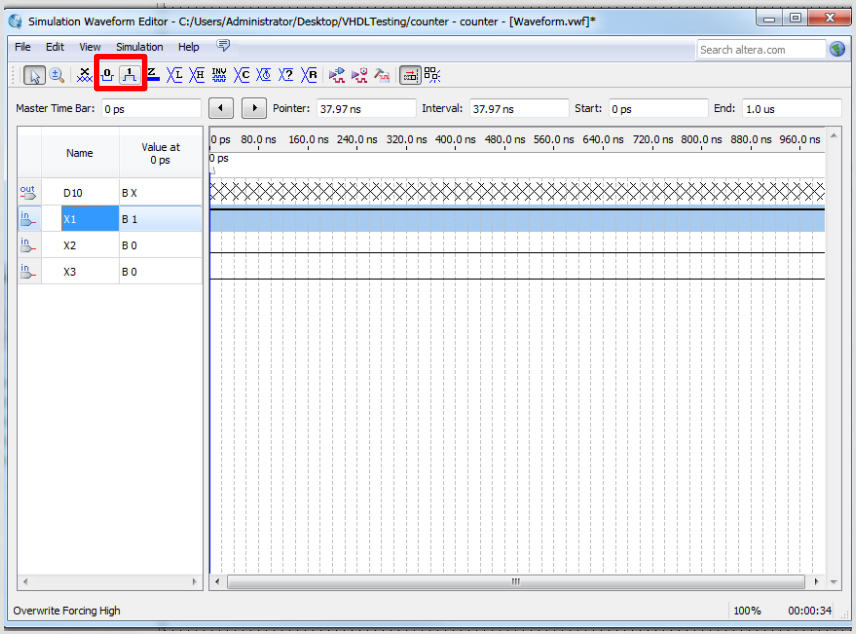
SIMULATION

Add your pins and Ok. Now you can see your pins in the vwf file.
Under Simulation -> Option and Choose the Quartus II simulator.

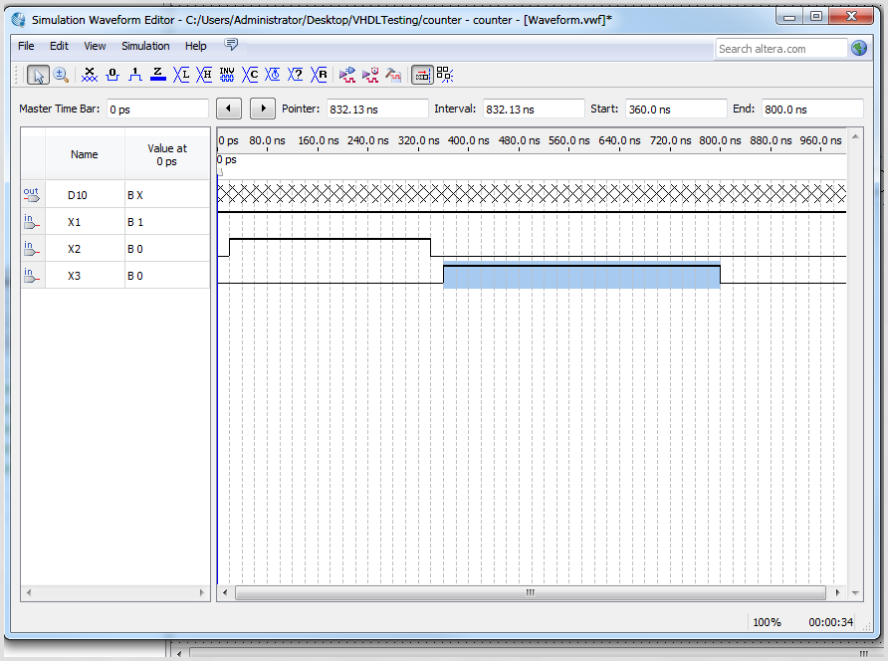


SIMULATION

Select on the Input pin and Assign the value “1” or ‘0’.

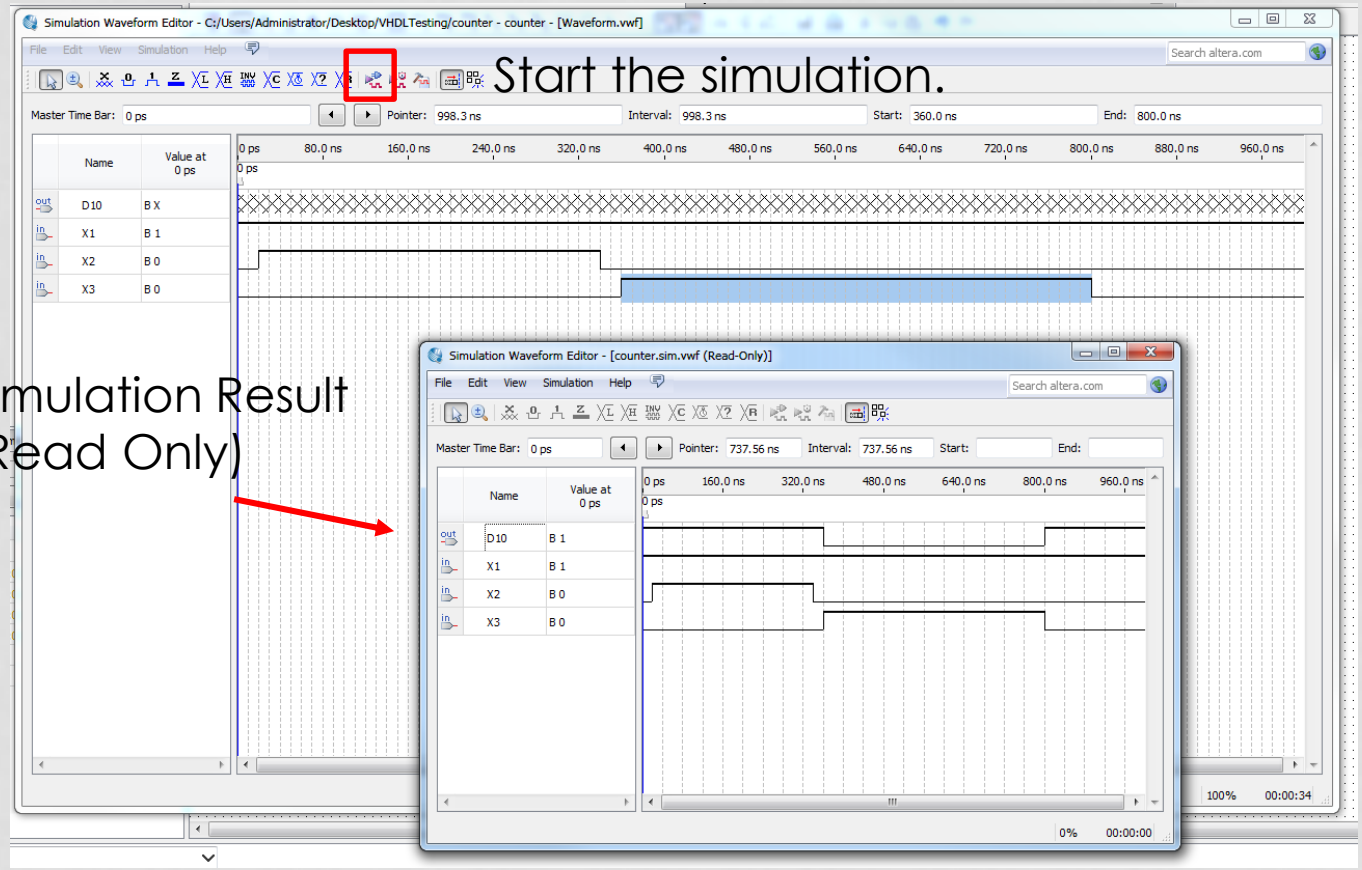


To assign a specific value to a specific period, highlight the interval and assign the value.



SIMULATION RESULT

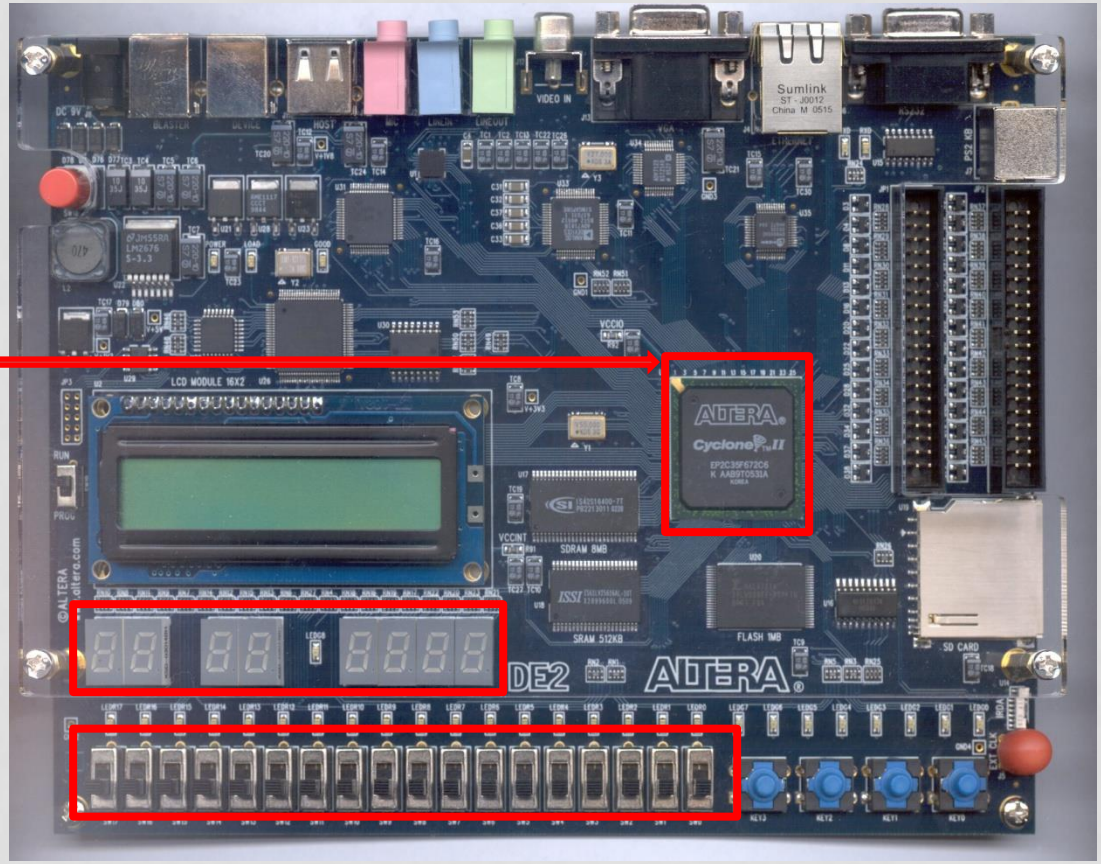
Once you finish assigning the desired input value, you can start simulation. Simulation -> Run Functional Simulation or from the button show below.



DOWNLOADING TO DE2 BOARD

- Use Altera Cyclone DE2 Device to test your ckt.

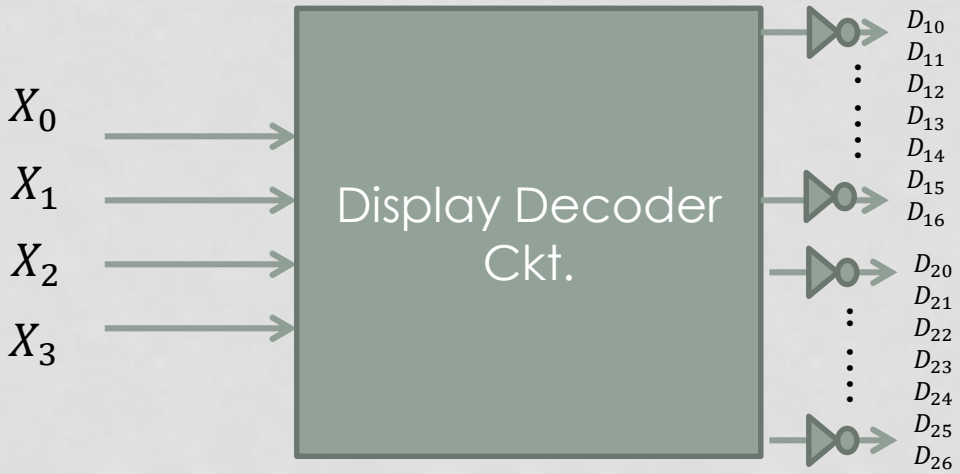
- FPGA (Filed Programmable Gate Array)
- Huge array of gates which can be programmed.



- DE2 has 8 seven-segment displays. Use any two for your ckt Display 1 and Display 2.
- 18 toggle switches.
- Use any 4 toggle switches for $X_3X_2X_1X_0$.

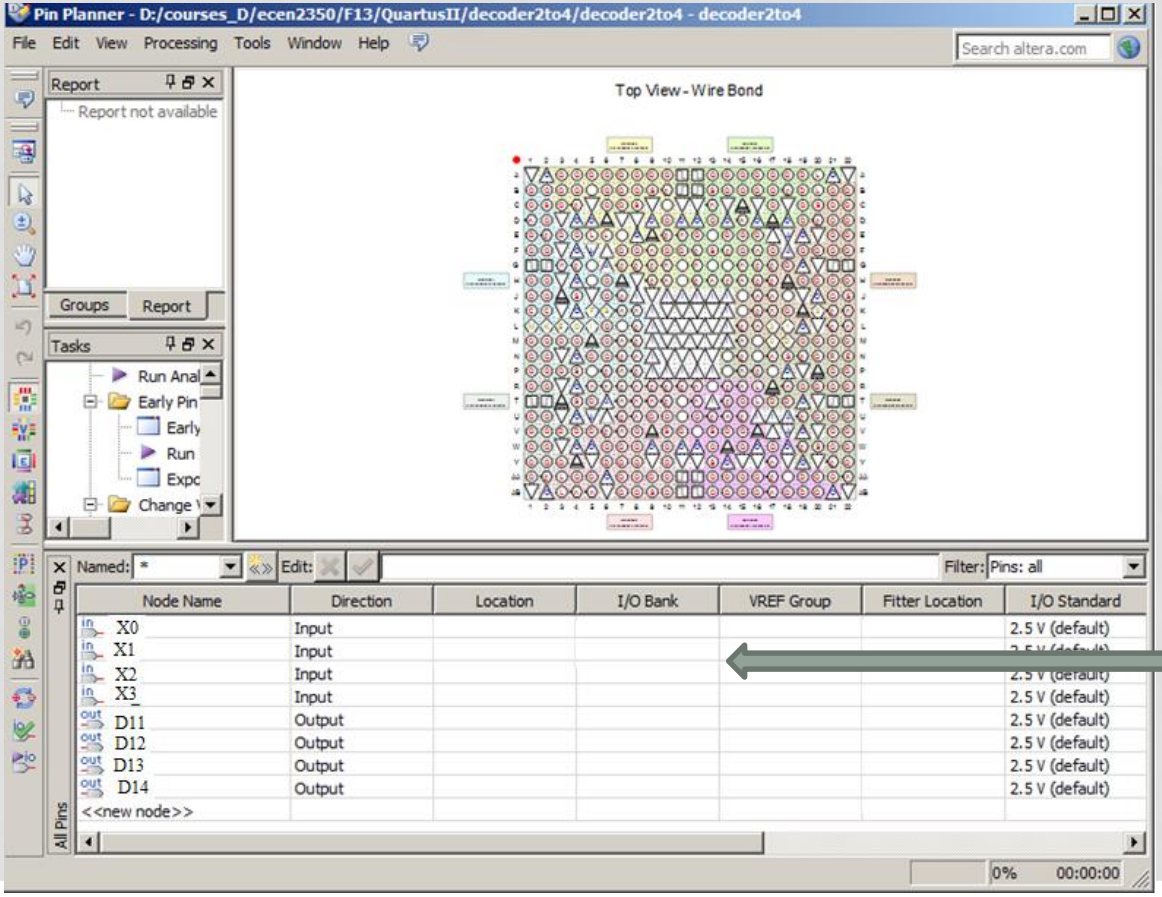
7 SEGMENT ON DE 2

- The displays are designed to light up when applying a low level voltage.
- Applying a Vcc to display will turn off the LEDs.
- Before downloading to DE2 board
  - Not all your outputs.



PIN ASSIGNMENT ON DE2

- To assign your ckt inputs and outputs on DE 2 open
  - Assignments -> Pin Planner. (The pin planner window will pop up.)



You will see your ckt inputs and outputs here.

DE 2 TOGGLE SWITCH PIN TABLE

Signal Name	FPGA Pin No.
SW[0]	PIN_N25
SW[1]	PIN_N26
SW[2]	PIN_P25
SW[3]	PIN_AE14
SW[4]	PIN_AF14
SW[5]	PIN_AD13
SW[6]	PIN_AC13
SW[7]	PIN_C13
SW[8]	PIN_B13
SW[9]	PIN_A13
SW[10]	PIN_N1
SW[11]	PIN_P1
SW[12]	PIN_P2
SW[13]	PIN_T7
SW[14]	PIN_U3
SW[15]	PIN_U4
SW[16]	PIN_V1
SW[17]	PIN_V2

7 SEGMENT PIN TABLE

HEX0[0]	PIN_AF10
HEX0[1]	PIN_AB12
HEX0[2]	PIN_AC12
HEX0[3]	PIN_AD11
HEX0[4]	PIN_AE11
HEX0[5]	PIN_V14
HEX0[6]	PIN_V13
HEX1[0]	PIN_V20
HEX1[1]	PIN_V21
HEX1[2]	PIN_W21
HEX1[3]	PIN_Y22
HEX1[4]	PIN_AA24
HEX1[5]	PIN_AA23
HEX1[6]	PIN_AB24
HEX2[0]	PIN_AB23
HEX2[1]	PIN_V22
HEX2[2]	PIN_AC25
HEX2[3]	PIN_AC26
HEX2[4]	PIN_AB26
HEX2[5]	PIN_AB25
HEX2[6]	PIN_Y24

HEX3[0]	PIN_Y23
HEX3[1]	PIN_AA25
HEX3[2]	PIN_AA26
HEX3[3]	PIN_Y26
HEX3[4]	PIN_Y25
HEX3[5]	PIN_U22
HEX3[6]	PIN_W24

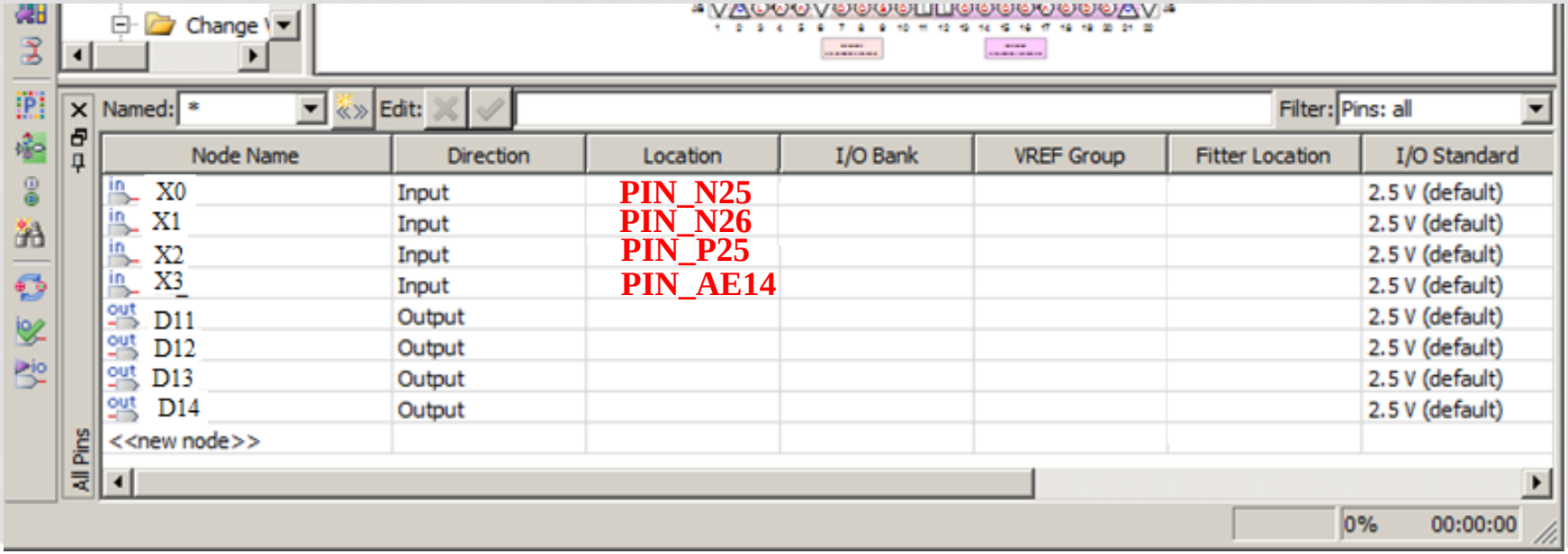
HEX4[0]	PIN_U9
HEX4[1]	PIN_U1
HEX4[2]	PIN_U2
HEX4[3]	PIN_T4
HEX4[4]	PIN_R7
HEX4[5]	PIN_R6
HEX4[6]	PIN_T3
HEX5[0]	PIN_T2
HEX5[1]	PIN_P6
HEX5[2]	PIN_P7
HEX5[3]	PIN_T9
HEX5[4]	PIN_R5
HEX5[5]	PIN_R4
HEX5[6]	PIN_R3
HEX6[0]	PIN_R2
HEX6[1]	PIN_P4
HEX6[2]	PIN_P3
HEX6[3]	PIN_M2
HEX6[4]	PIN_M3
HEX6[5]	PIN_M5
HEX6[6]	PIN_M4

HEX7[0]	PIN_L3
HEX7[1]	PIN_L2
HEX7[2]	PIN_L9
HEX7[3]	PIN_L6
HEX7[4]	PIN_L7
HEX7[5]	PIN_P9
HEX7[6]	PIN_N9

PIN ASSIGNMENT EXAMPLE

- For example, use SW3 to SW0 as $X_3 - X_0$.
- Assign the output pins and
- Compile your ckt.

Signal Name	FPGA Pin No.
SW[0]	PIN_N25
SW[1]	PIN_N26
SW[2]	PIN_P25
SW[3]	PIN_AE14



Node Name	Direction	Location	I/O Bank	VREF Group	Fitter Location	I/O Standard
X0	Input	PIN_N25				2.5 V (default)
X1	Input	PIN_N26				2.5 V (default)
X2	Input	PIN_P25				2.5 V (default)
X3	Input	PIN_AE14				2.5 V (default)
D11	Output					2.5 V (default)
D12	Output					2.5 V (default)
D13	Output					2.5 V (default)
D14	Output					2.5 V (default)
<<new node>>						

DOWNLOADING TO DE2 BOARD

1. Connect the DE2 Board to the Computer using USB Blaster port.
2. make sure that the RUN/PROG switch located to the left of the LCD display is in RUN position. As a matter of fact, that switch must always be in RUN, so never change it.
3. Once the device is turned on, you should observe
   - a default pattern on the board's LEDs,
   - a welcome message on the LCD,
   - "POWER" and "GOOD" LEDs (blue, above the LCD) on.

DOWNLOADING TO DE2 BOARD

4. Select TOOLS -> Programmer.
5. Make sure the hardware Setup is USB-Baster.
(if you see anything else, click on Hardware Setup, and choose USB-Blaster)
6. The file to download to the board should be appear automatically.
If you don't see any file,
Click Add Files and
Choose the file from directory
YourProjectfolder/Output_files/XXXXX.sof
Select .sof file.
7. Once you see the file, Click Start and that will Download the file to the Board.