EDA Technology

Chapter 4

FPGA Hardware Implementation

4.1 Code Editing Input and System Compilation

4.1.1 Design File Edit and Input

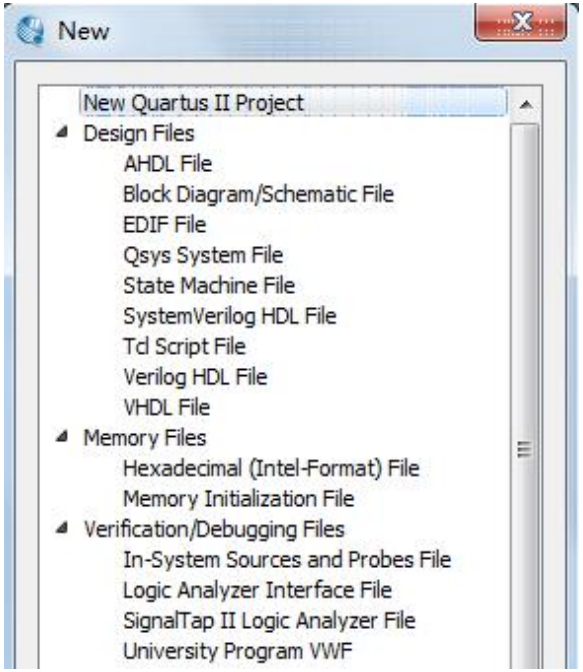


Figure: Select edit file type

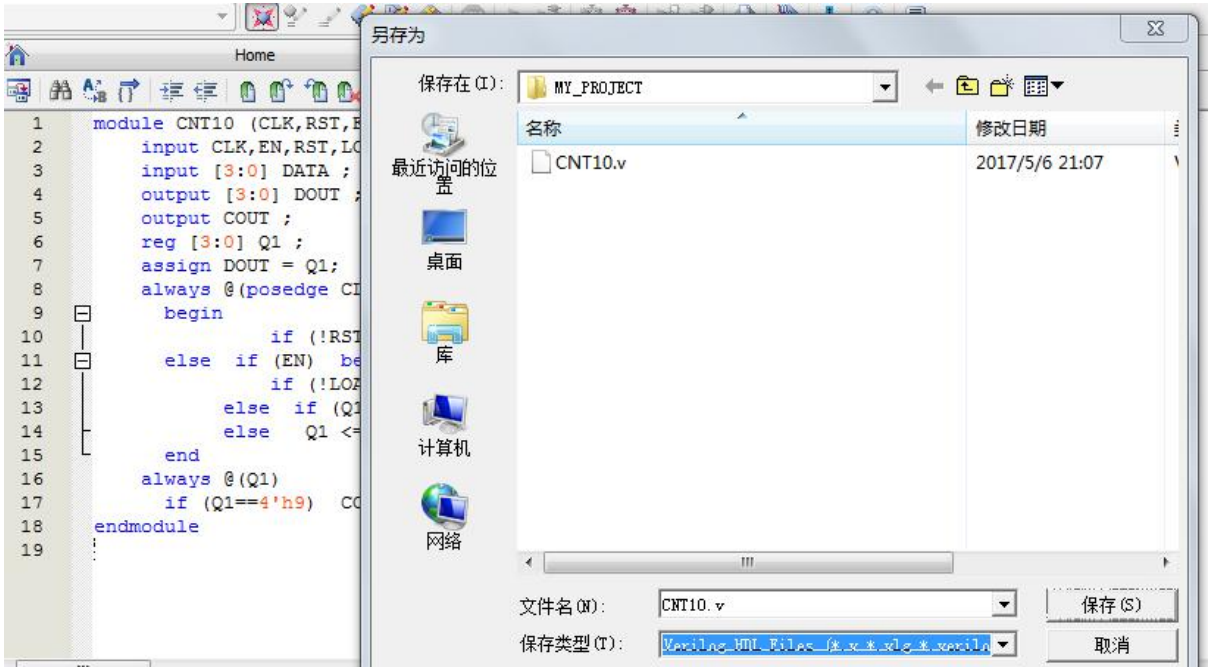
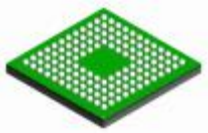


Figure: Edit input source program and save it

Any EDA design is a project, and you must firstly create a folder for this project that holds all the design files associated with the project. Different design projects are best placed in different folders, and all files for the same project are in the same folder.

4.1 Code Editing Input and System Compilation

4.1.2 Creating a Project



Fine-pitch Ball Grid Array

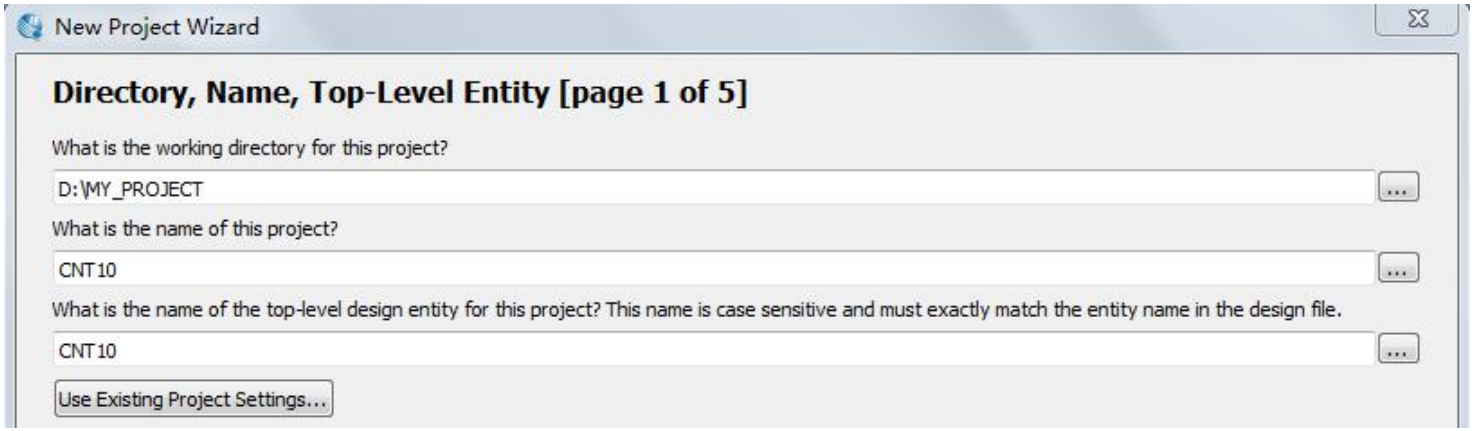


Figure: Creating project CNT10 using New Project Wizard

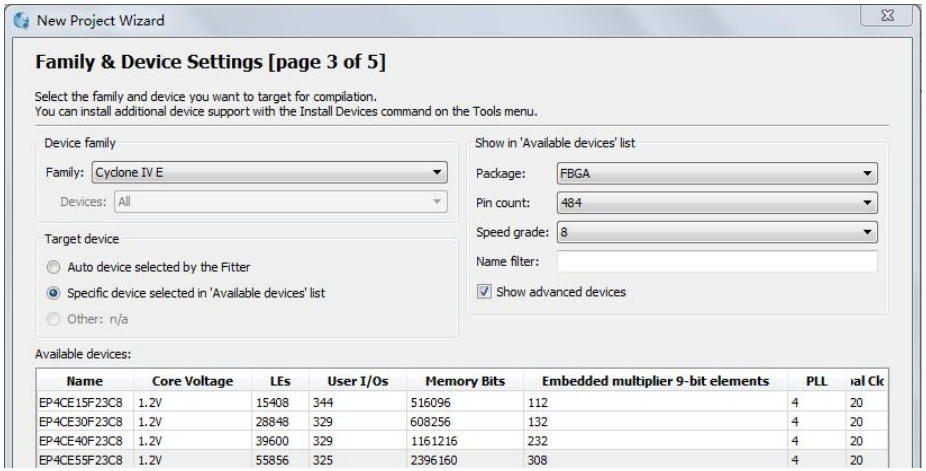


Figure: Select the target device EP4CE55F23C8

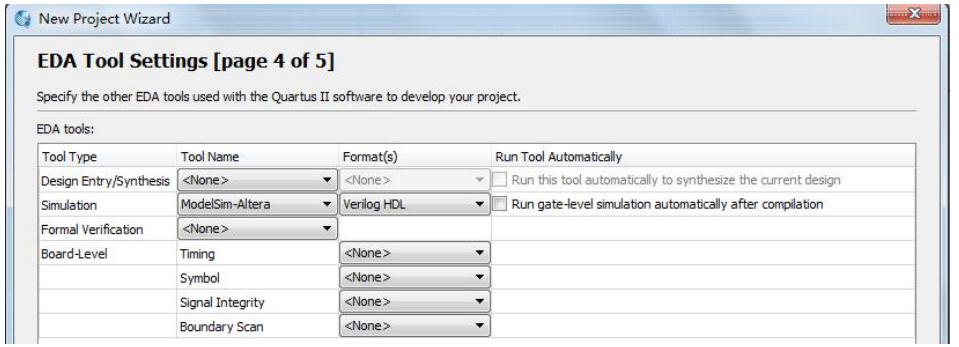


Figure: Design and verification tool selection

4.1 Code Editing Input and System Compilation

4.1.3 Constraint Item Setting

Assignments →Settings → More Settings

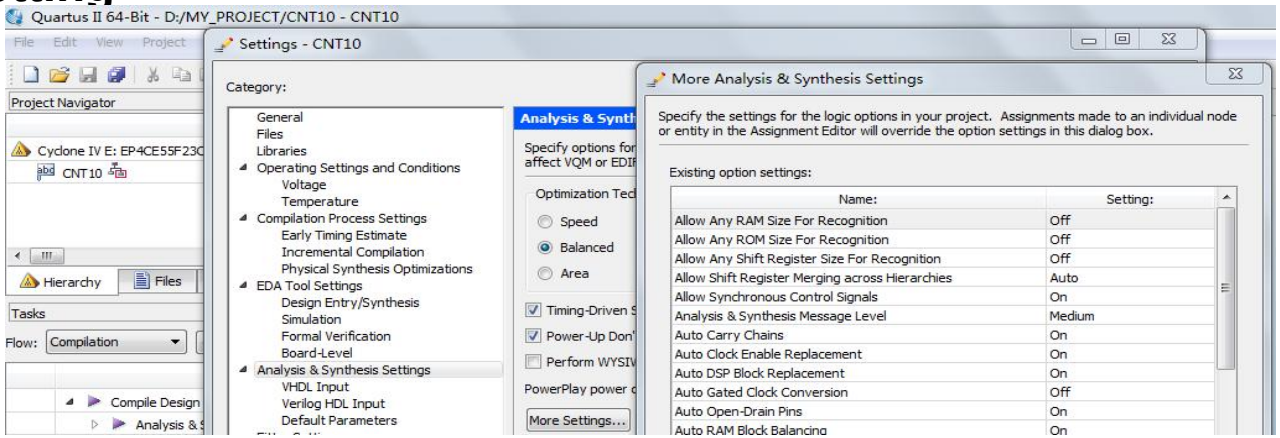


Figure: Select compilation synthesis work mode

Assignments →Device

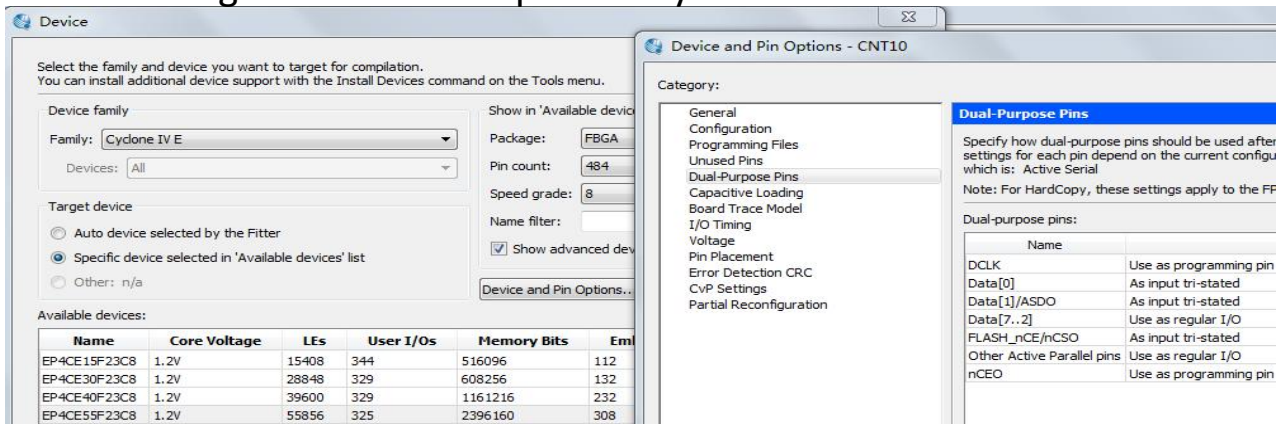


Figure: Select the target device and operation mode

(1) Select compilation constraints. (2) choose the way to configure the device. (3) choose configuration device and programming mode. (4) select the target device pin port state.

4.1 Code Editing Input and System Compilation

4.1.4 Synthesis and Compilation

Processing -> Start Compilation

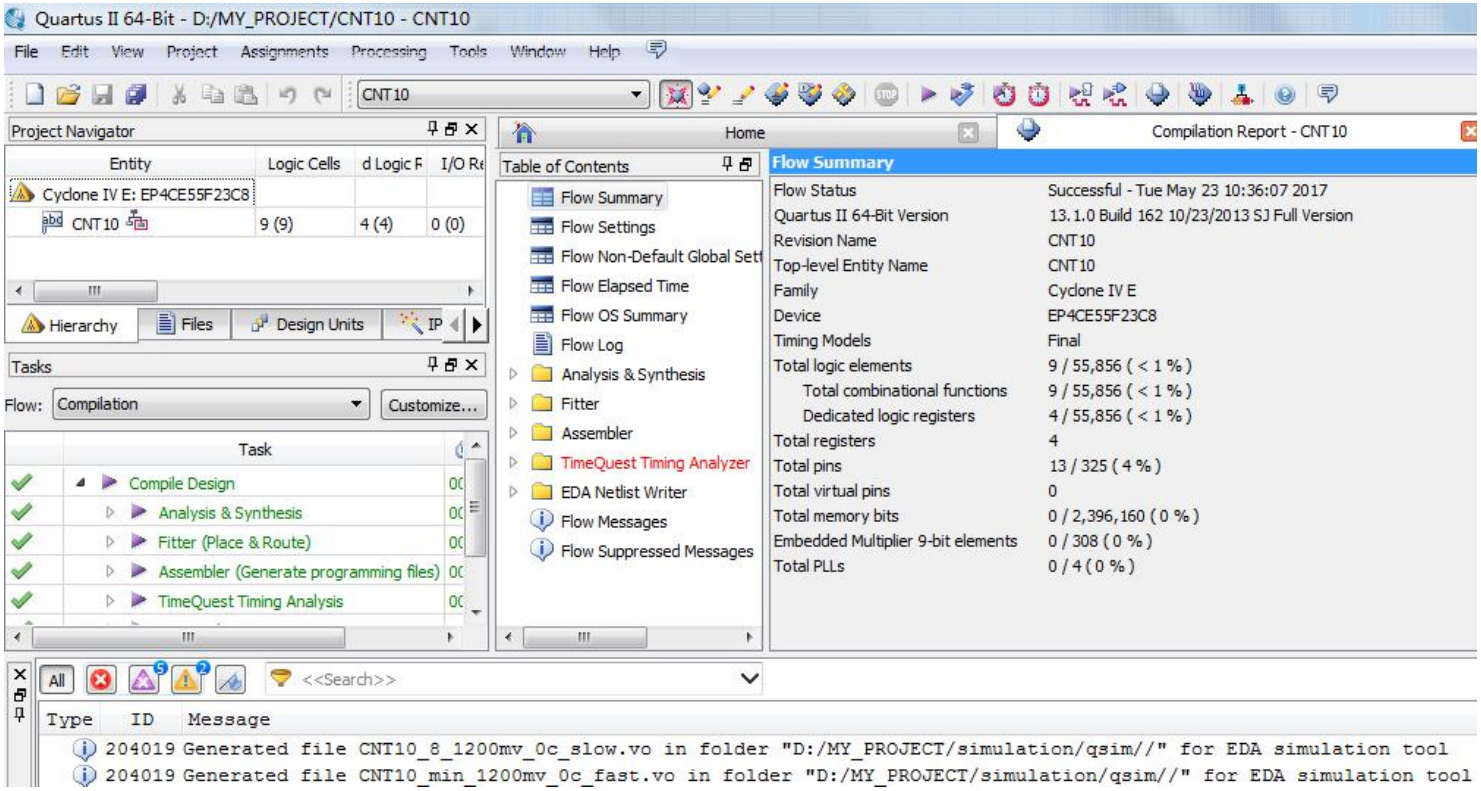


Figure: Report information after the entire compilation is error-free

Compilation includes Quartus II's multiple processing operations on the design input, including input file troubleshooting, netlist file extraction, logic synthesis, adaptation, and assembly files (simulation files and programming profile) generation, and timing analysis of the project.

4.1 Code Editing Input and System Compilation

4.1.5 Application of RTL Viewer

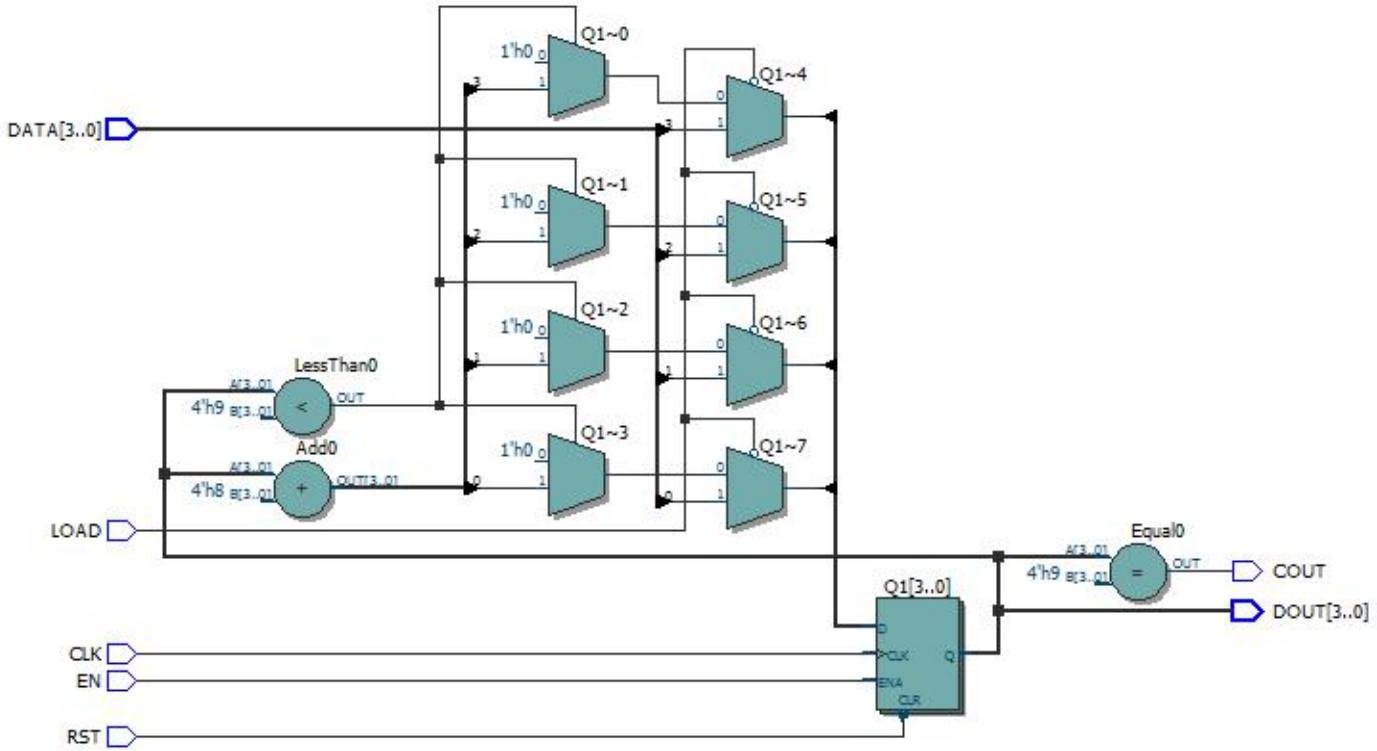


Figure: RTL diagram of the CNT10 project

Select the Tools → Netlist Viewers

4.2 Timing Simulation

If using Modelsim:

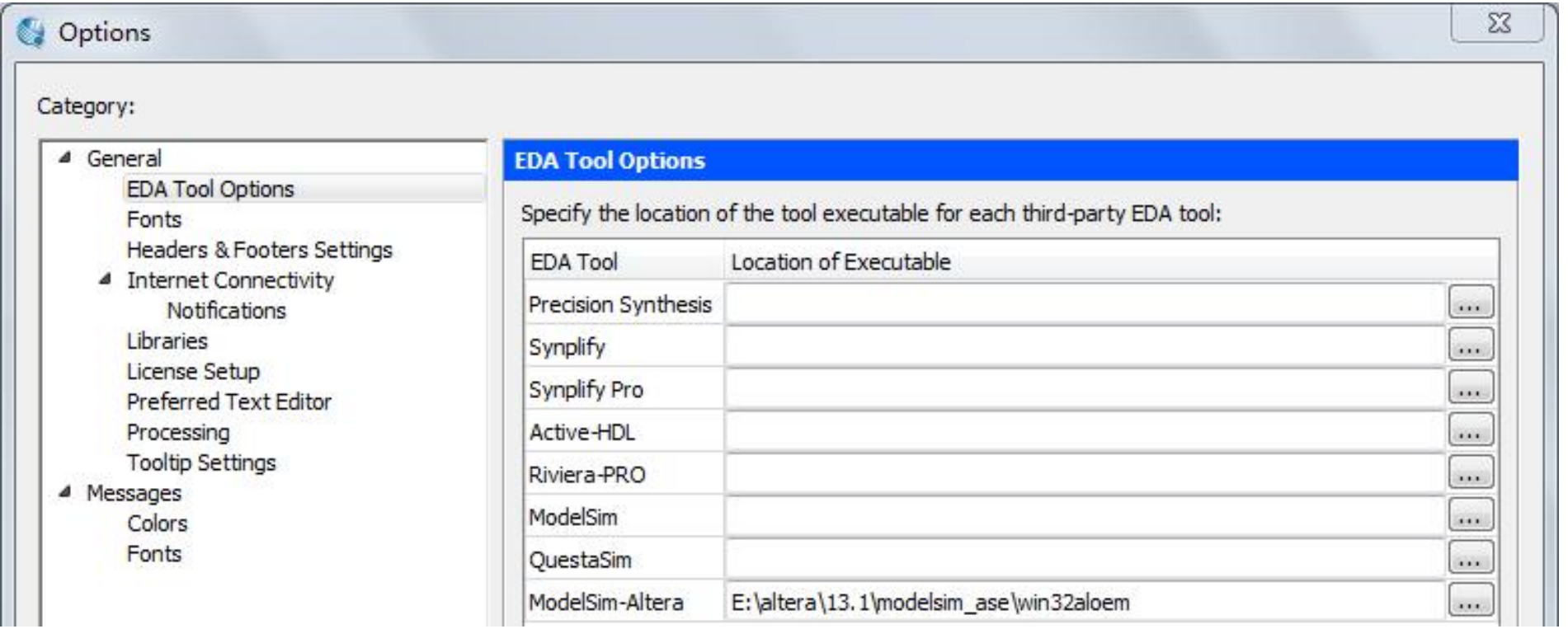


Figure: Viewing the Quartus Simulation Tool Path to Modelsim Simulation Software

Make sure the simulation tool in Quartus II points to the path where Modelsim is located

4.2 Timing Simulation

(1) **Open the Simulation Waveform Editor**

CNT10.vwf

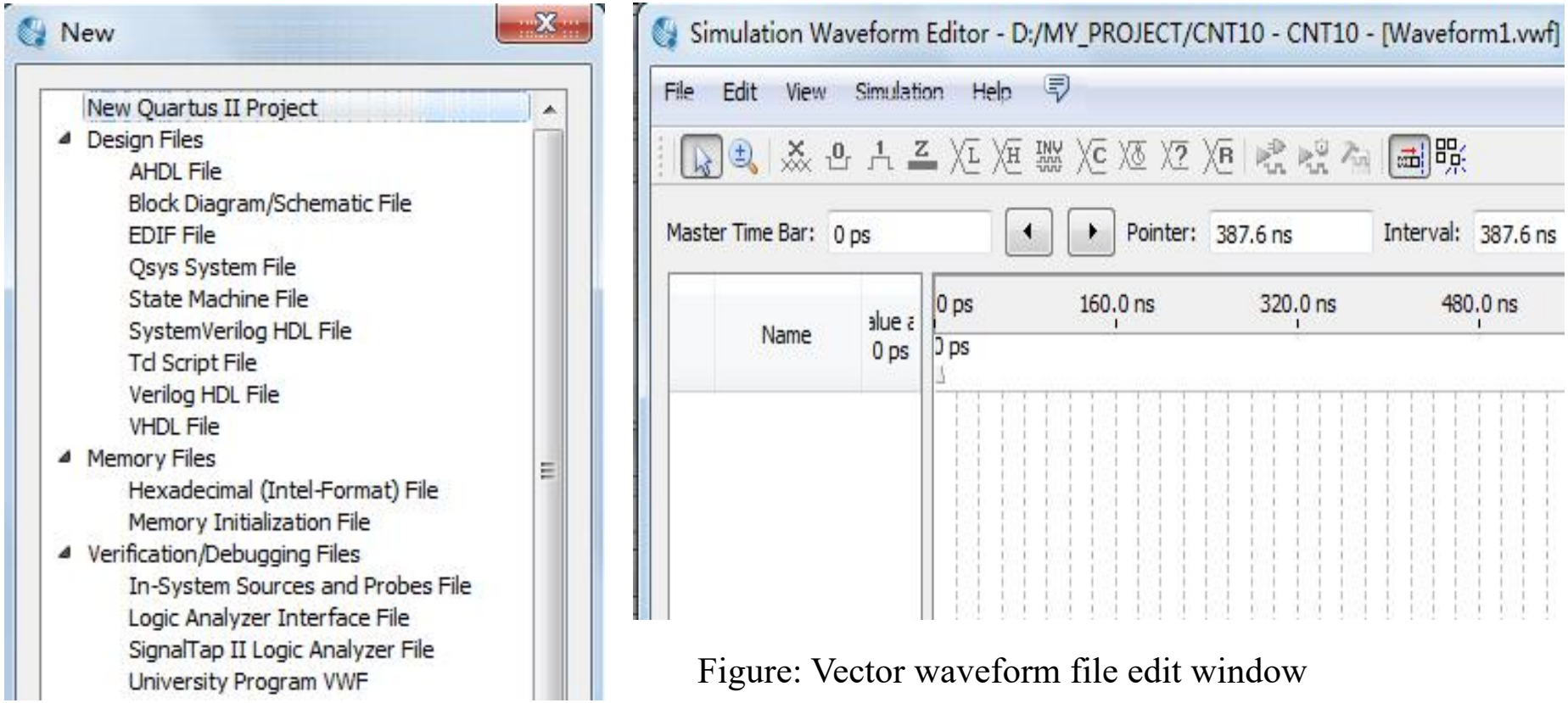


Figure: Vector waveform file edit window

File->New->Vector Waveform File-> Zoom ->Fit in Window

4.2 Timing Simulation

(2) **Set the simulation time**

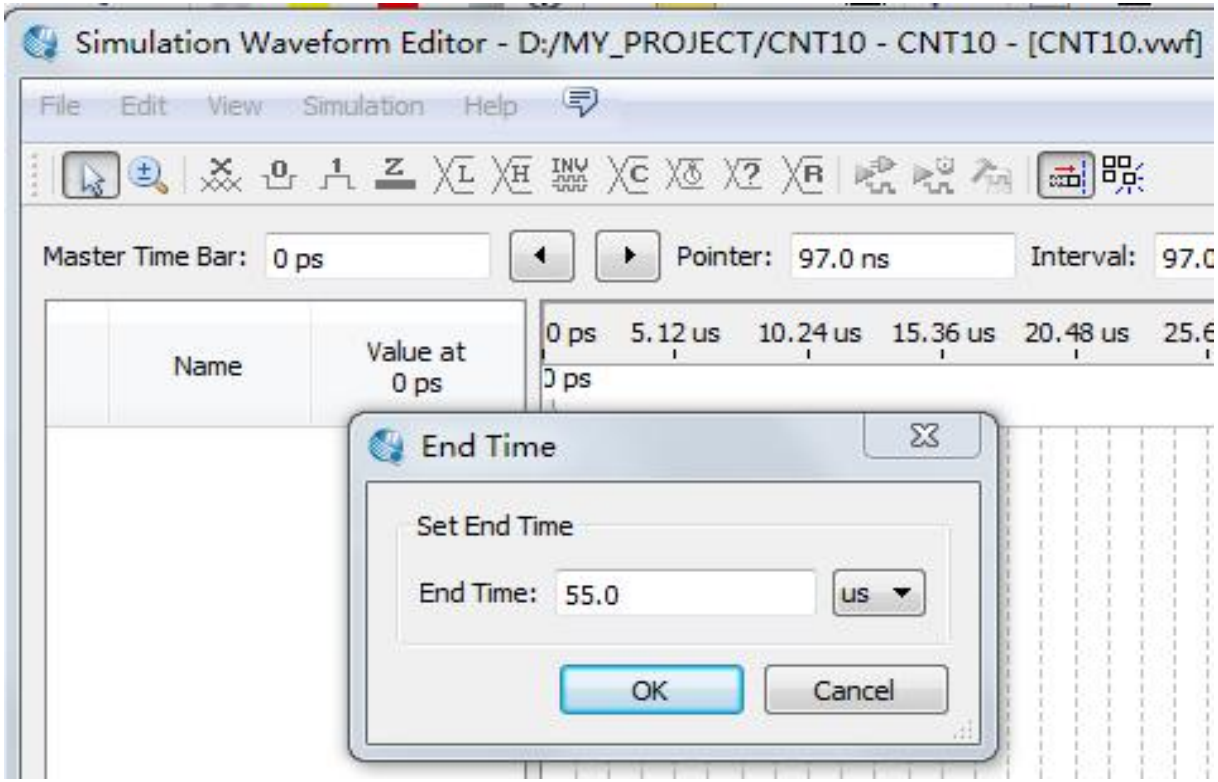


Figure: Set the simulation time

Edit-> Set End Time (usually 10~100 us)

4.2 Timing Simulation

Use Node Finder to select the signal nodes into the waveform

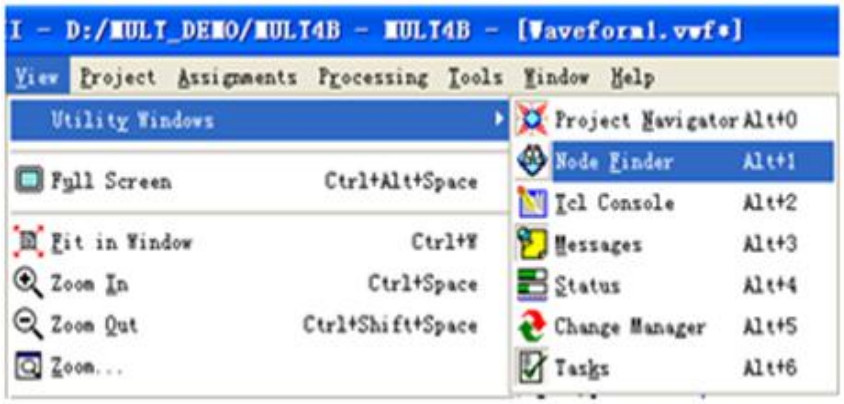


Figure: Open Signal Node Query Window

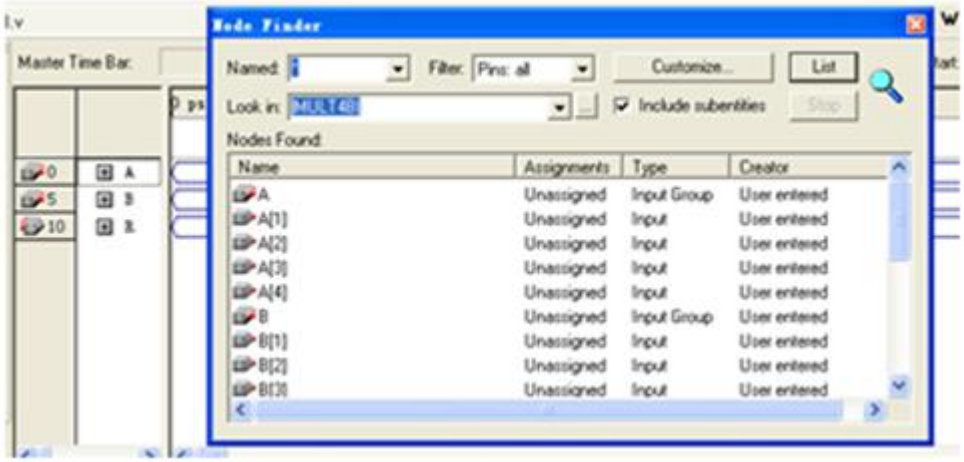


Figure: Drag the signal node into the waveform editor

If the node name is not displayed, compile once again, and the signal nodes will be shown.

4.2 Timing Simulation

Join signal nodes needed for simulation

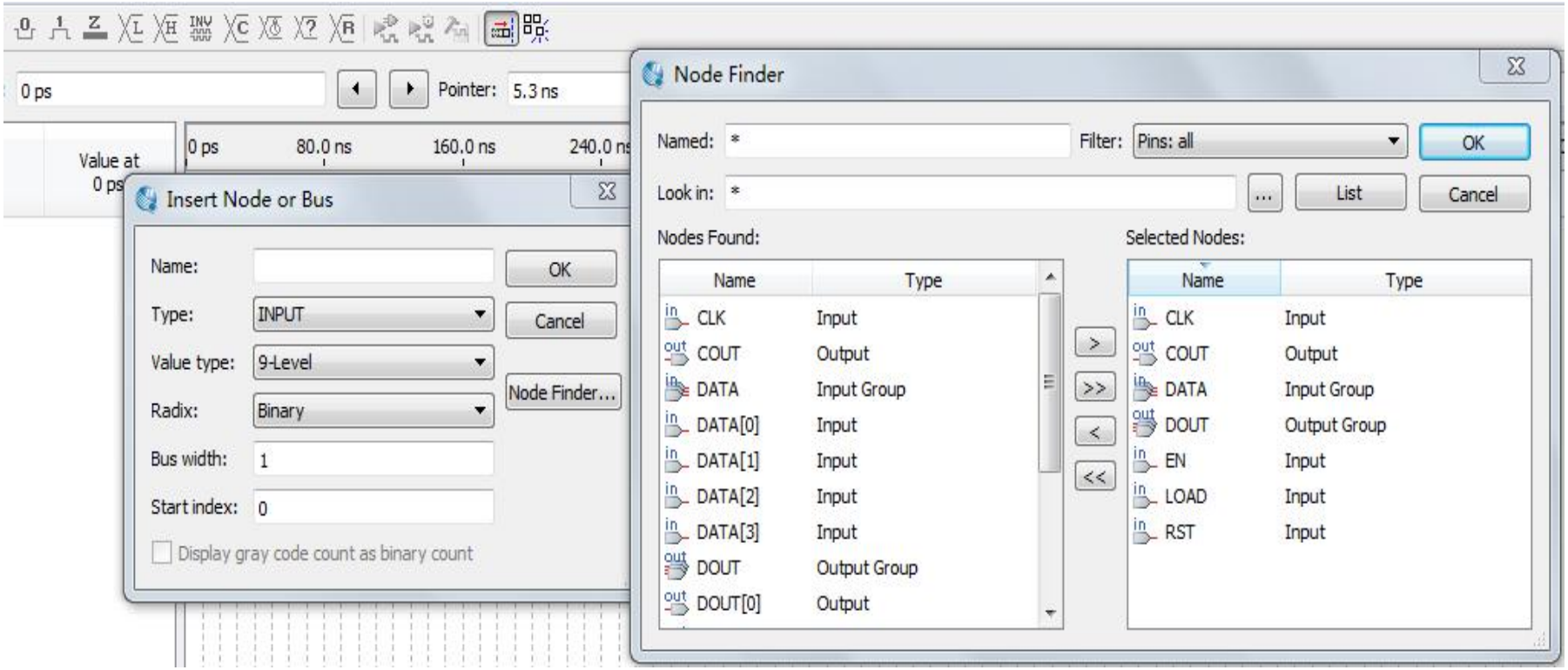


Figure: Join signal nodes needed for simulation

Radix -> Hexadecimal

4.2 Timing Simulation

Setting the bus data format

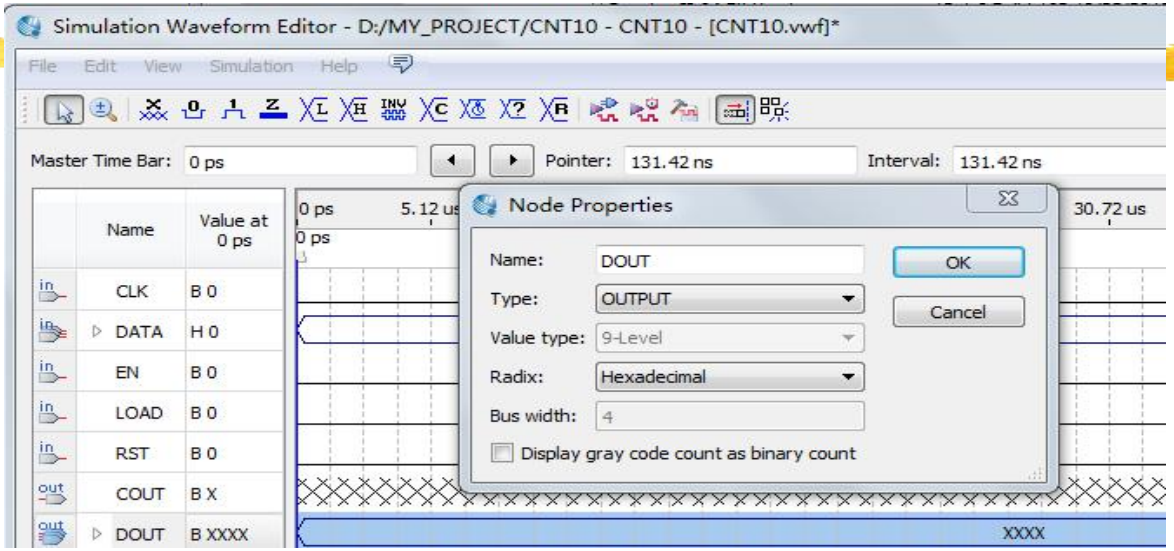


Figure: Setting the bus data format

Setting the clock parameters

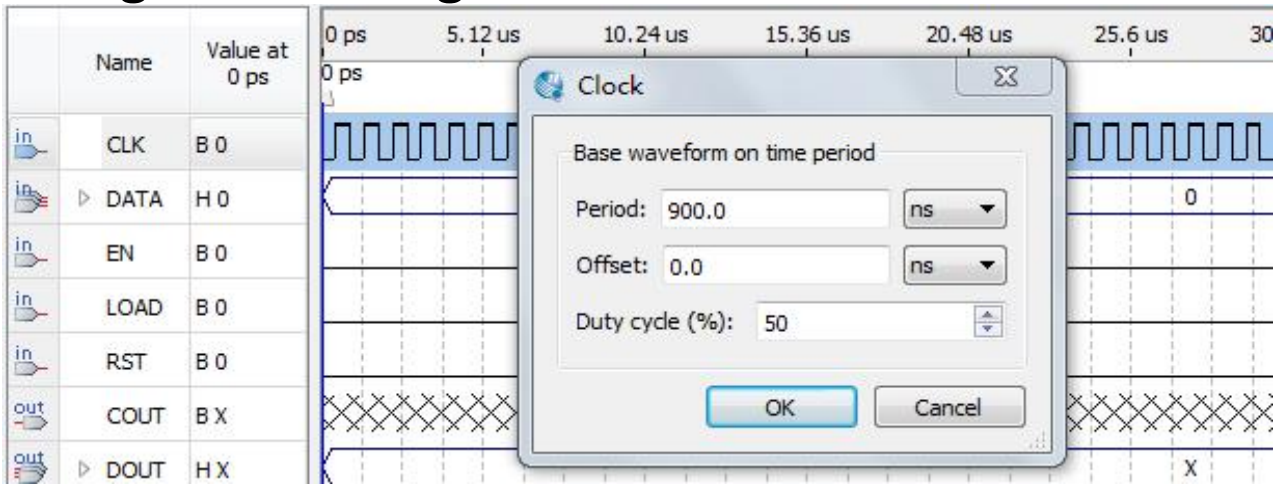


Figure: Setting clock parameters

4.2 Timing Simulation

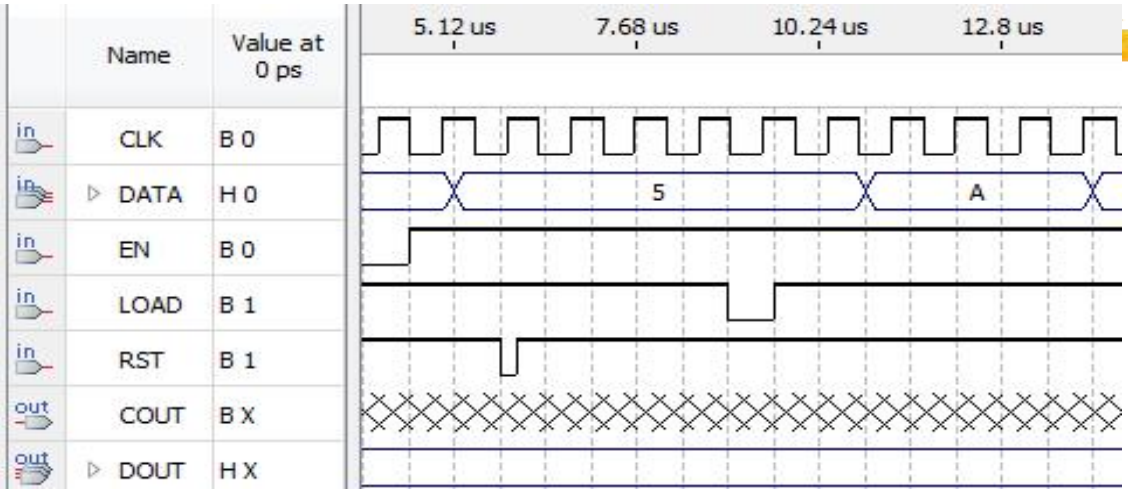


Figure: Editing the excitation waveform

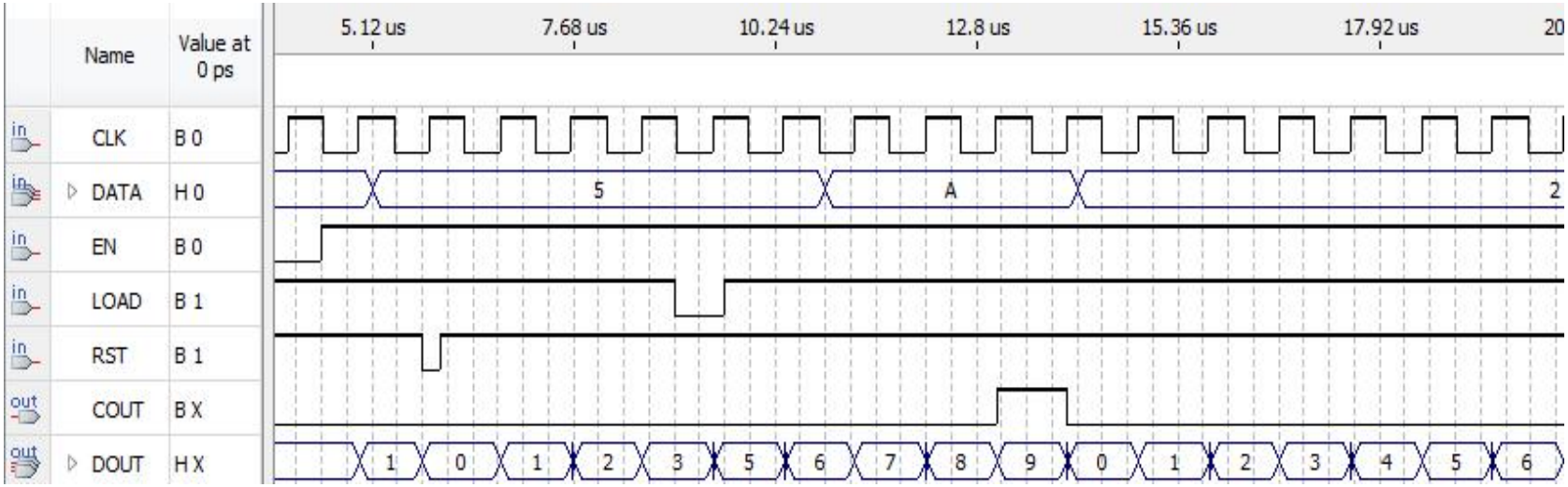


Figure: Waveform file for simulation output

4.3 Hardware Testing

4.3.1 Pin Assignment

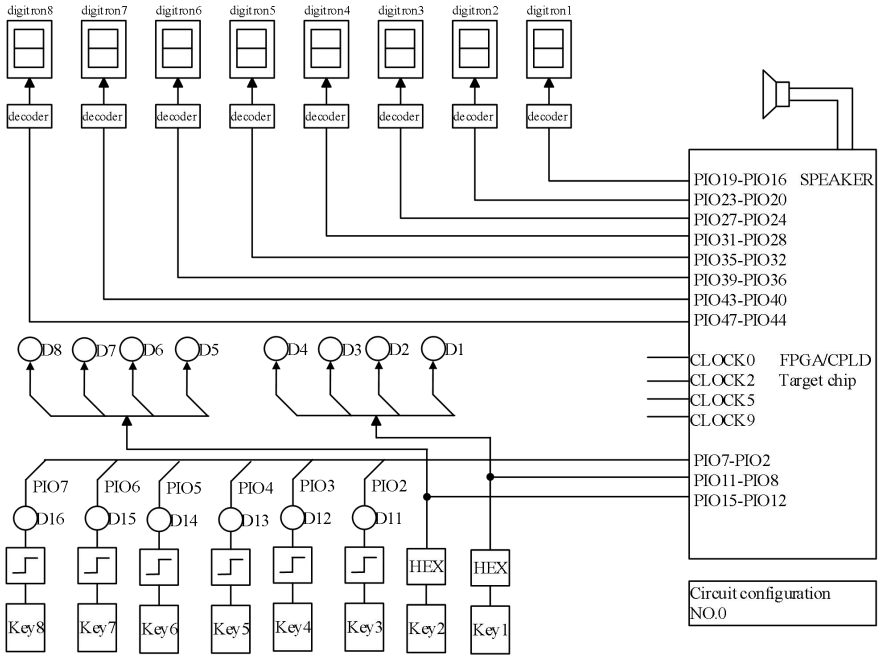


Figure: Experimental Circuit for FPGA in Mode 0

Table 4-1 Pin Locking Based on EP4CE55F23C8 (Available from the List of Appendix A.4)

Counter signal name	CLK	EN	LOAD	RST	DATA(3)	DATA(2)	DATA(1)
Mode 0 circuit control	Key 8	Key 7	Key 6	Key 5	Key1:D4	key1:D3	key1:D2
Mode 0 circuit signal	PIO7	PIO6	PIO5	PIO4	PIO11	PIO10	PIO9
Corresponding FPGA pins	AB6	Y7	AA6	AB3	AB5	AA3	W2
Counter signal name	DATA(0)		COUT	DOUT(3)	DOUT(2)	DOUT(1)	DOUT(0)
Mode 0 circuit control	Key 1:D1		Digital 2:a segment	Digital 1	Digital 1	Digital 1	Digital 1
Mode 0 circuit signal	PIO8		PIO20	PIO19	PIO18	PIO17	PIO16
Corresponding FPGA pins	U2		AA1	V2	W1	R2	U1

Experimental Circuit for FPGA

4.3 Hardware Testing

4.3.1 Pin Assignment

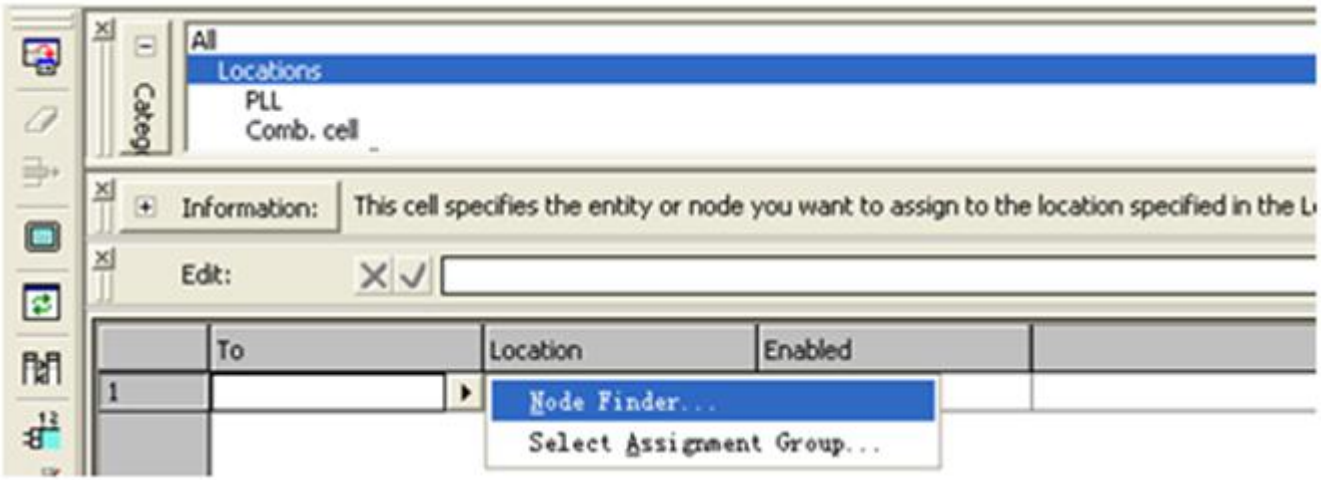


Figure: Locking the pin of the FPGA by using the Assignment Editor

Assignments->Assignment Editor->Locations or PIN planner

To: the port in the program
Location: the pin on the board, such as PIN_23

4.3 Hardware Testing

Node Name	Direction	Location	I/O Bank	VREF Group	Fitter Location
CLK	Input				PIN_J4
COUT	Output				PIN_T8
DATA[3]	Input				PIN_R8
DATA[2]	Input				PIN_R10
DATA[1]	Input				PIN_T9
DATA[0]	Input				PIN_V6
DOUT[3]	Output				PIN_R9
DOUT[2]	Output				PIN_T5
DOUT[1]	Output				PIN_R6
DOUT[0]	Output				PIN_P8
EN	Input				PIN_T7
LOAD	Input				PIN_R7
RST	Input				PIN_P4

Figure: The newly opened Pin Planner window

Node Name	Direction	Location
CLK	Input	PIN_AB6
COUT	Output	PIN_AA1
DATA[3]	Input	PIN_AB5
DATA[2]	Input	PIN_AA3
DATA[1]	Input	PIN_W2
DATA[0]	Input	PIN_U2
DOUT[3]	Output	PIN_V2
DOUT[2]	Output	PIN_W1
DOUT[1]	Output	PIN_R2
DOUT[0]	Output	PIN_U1
EN	Input	PIN_Y7
LOAD	Input	PIN_AA6
RST	Input	PIN_AB3

Figure: After pin locking is completed

After locking the PIN, you must compile the project again, Start Compilation

4.3 Hardware Testing

4.3.2 Compiled File Download

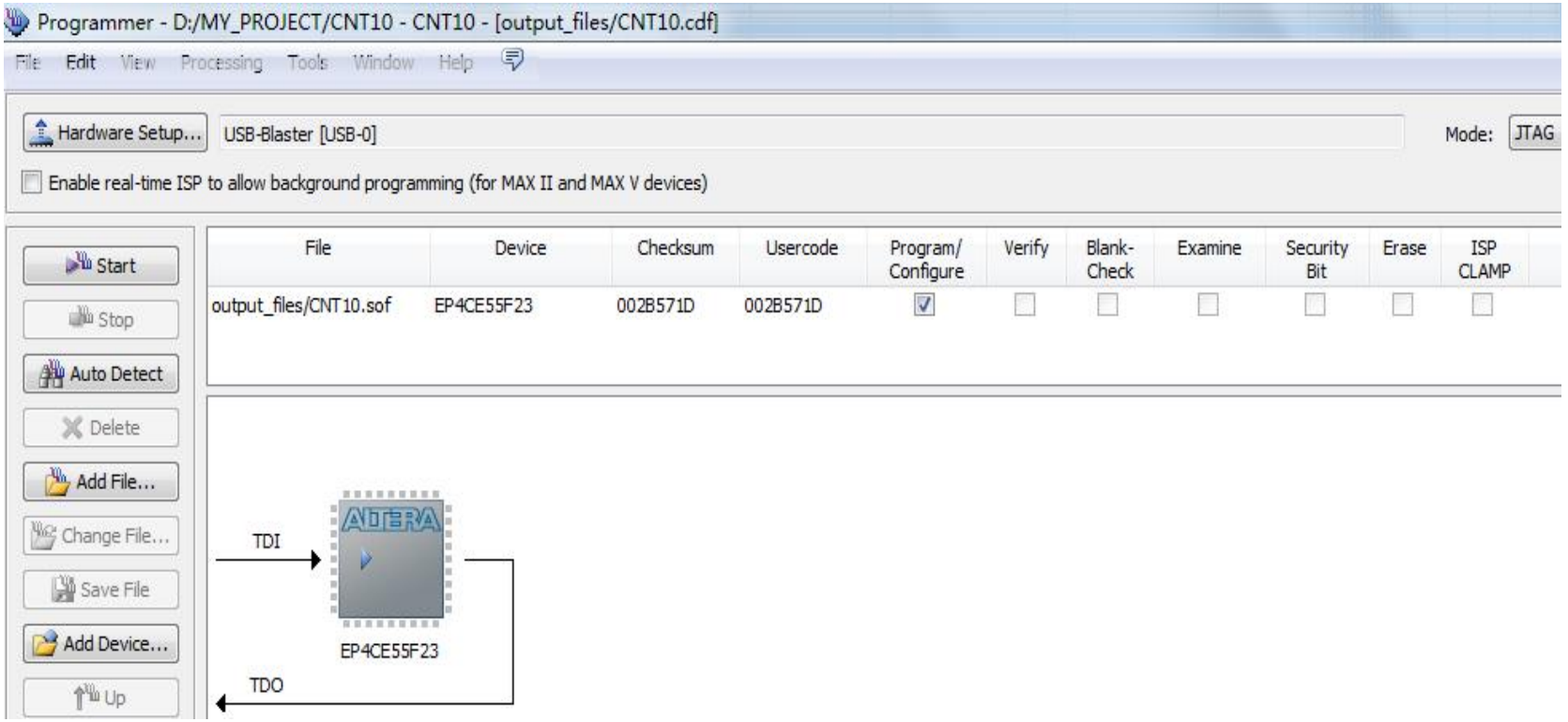


Figure: Select JTAG programming mode to configure SOF file into FPGA

Download SOF configuration file into FPGA. Install the driver and find the hardware device.

4.3 Hardware Testing

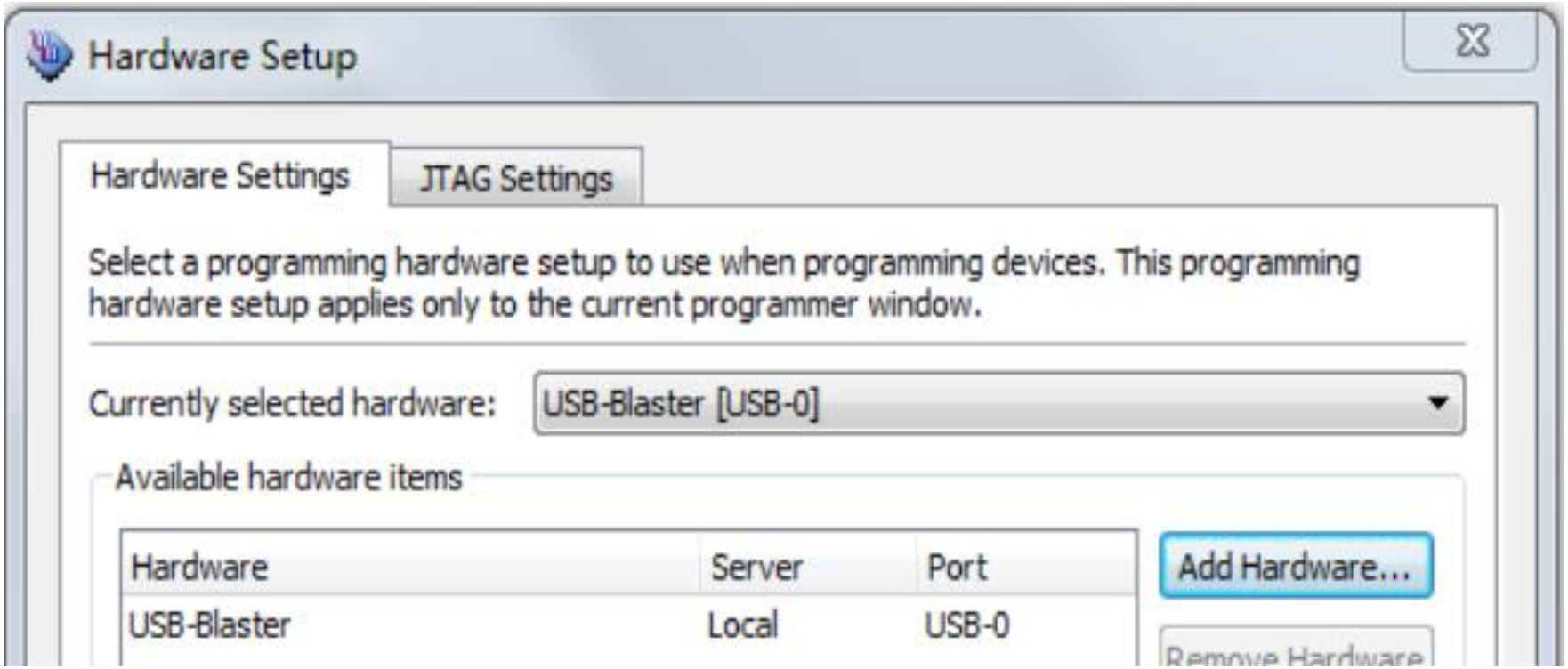


Figure: Add a programming download method

4.3 Hardware Testing

4.3.4 USB-Blaster Driver Installation

- One end of the USB Blaster programmer into the USB port of the PC. A USB driver dialog box will pop up.
- Select the user to search for the driver according to the dialog guidance.
- It is assumed that the Quartus II software is installed on the E drive and the driver's path is E:\altera\quartus\drivers\usb-blaster.

  D:\altera\quartus90\drivers\use-blaster

4.4 Circuit Schematic Design Flow

4.4.1 Half-adder Design

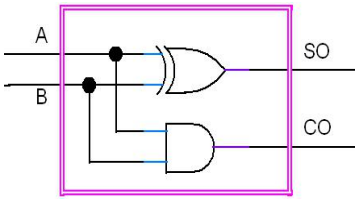


Figure: Circuit structure of a half-adder

A	B	SO	CO
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Figure: Truth table of half adder

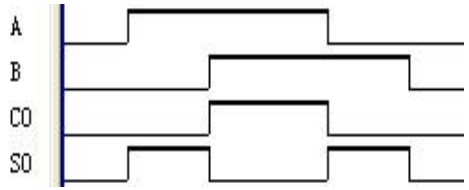


Figure :Simulation function

$$SO=A \oplus B;\quad CO=A \cdot B$$

4.4 Circuit Schematic Design Flow

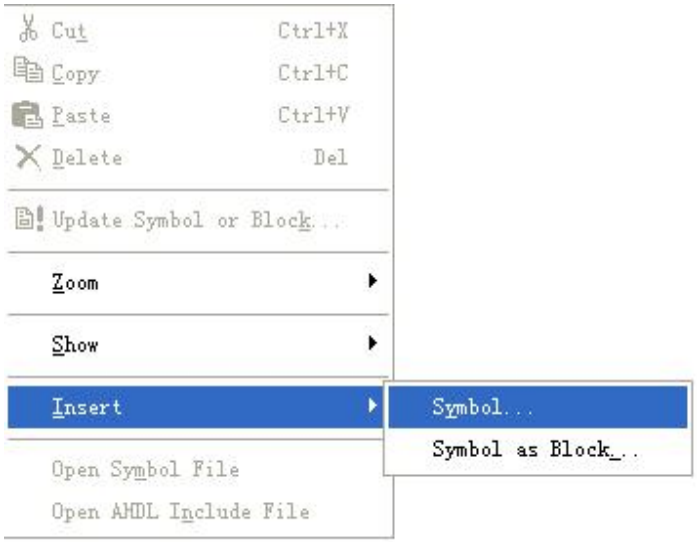


Figure: Select Open Component Entry Window

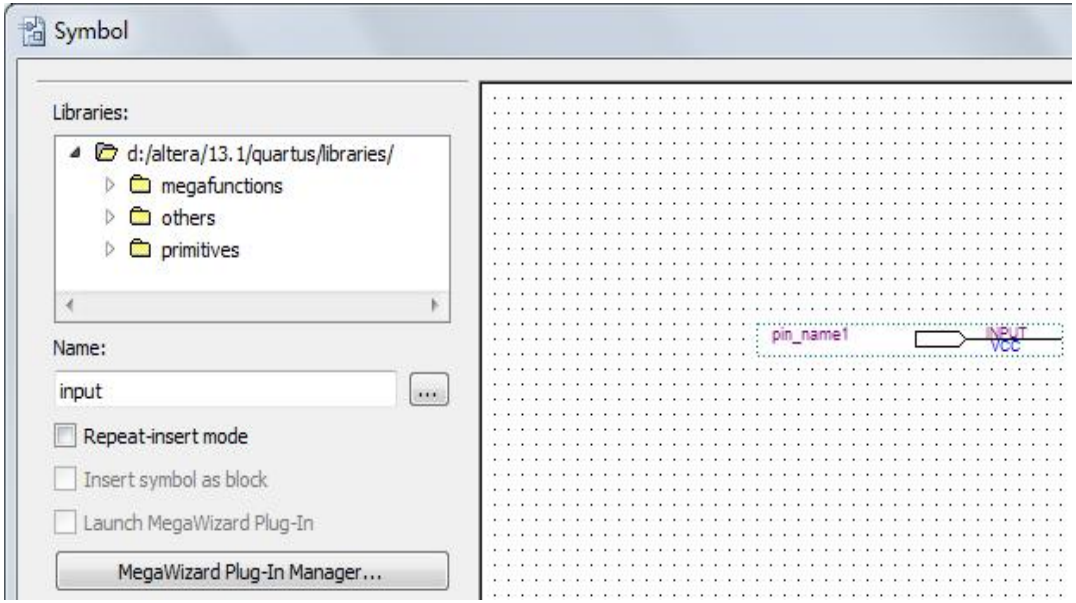


Figure: Input Pins in the Component Input Dialog Box

4.4 Circuit Schematic Design Flow

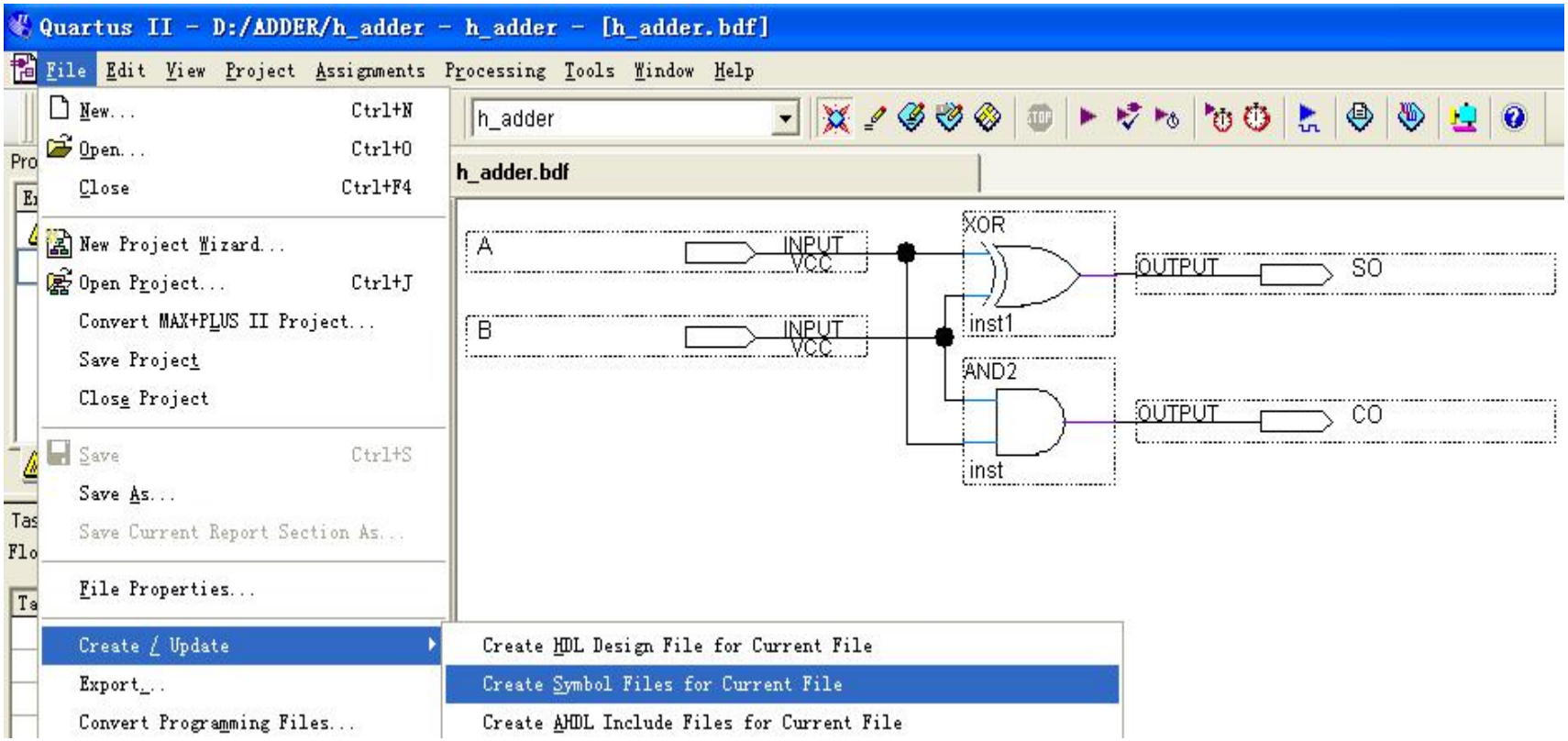


Figure: Complete design and pack the half adder as a component for calling in higher level design

4.4 Circuit Schematic Design Flow

4.4.2 Top-level Design of Full-adder

Figure: Full adder f_adder.bdf project settings

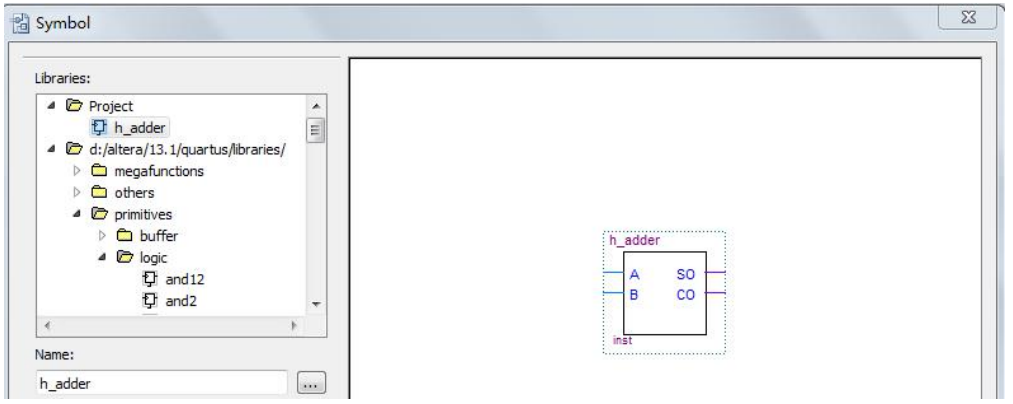


Figure: Add a half adder to the f_adder project

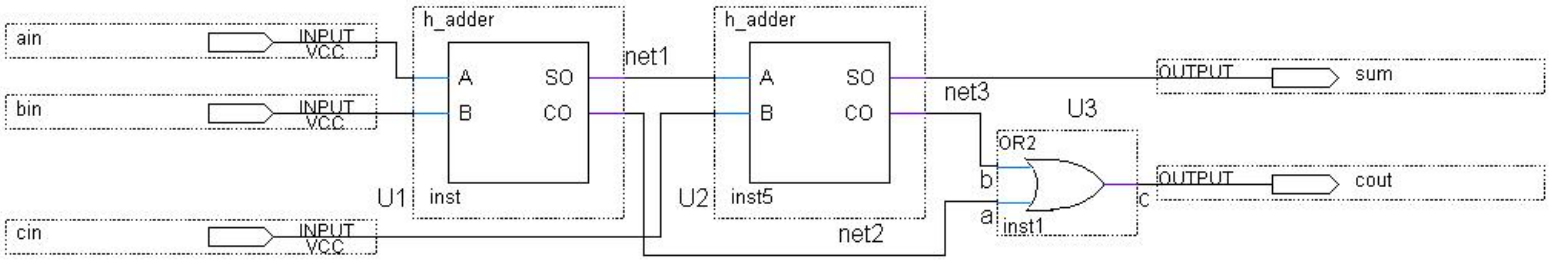


Figure: Full adder f_adder circuit diagram

4.4 Circuit Schematic Design Flow

4.4.3 Timing Simulation and Hardware Testing of Full Adders

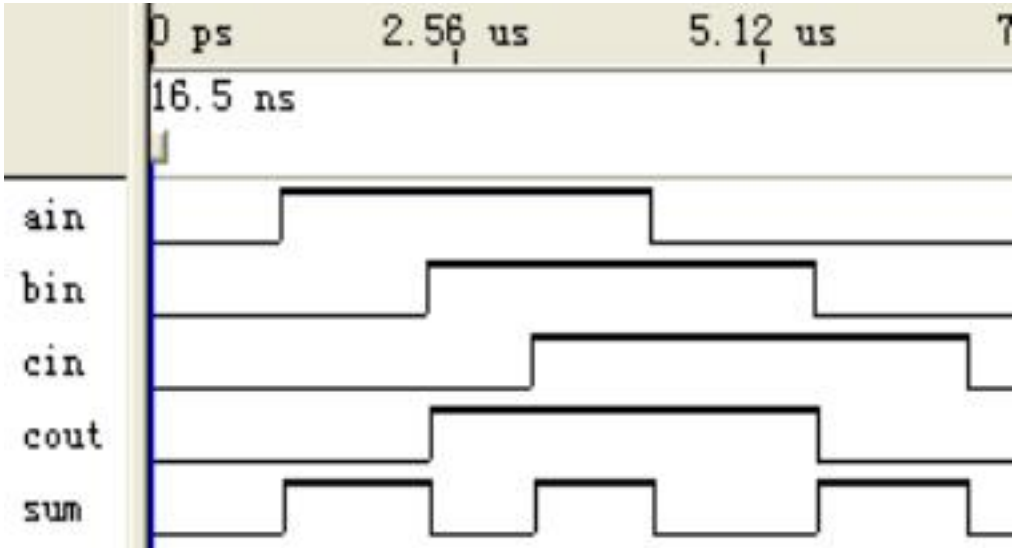


Figure: Simulation waveform of the full adder

4.5 Pin Assignment Using Attributes

Direct control of pin locking in Verilog code

[Example]

```
module CNT10 (CLK,RST,EN,LOAD,COUT,DOUT,DATA);
  input [3:0] DATA   /* synthesis chip_pin="AB5,AA3,W2,U2" */;
  output [3:0] DOUT  /* synthesis chip_pin="V2,W1,R2,U1" */;
  output COUT        /* synthesis chip_pin="AA1" */;
  input  CLK         /* synthesis chip_pin = "AB6" */;
  input   EN         /* synthesis chip_pin = "Y7" */;
  input  RST         /* synthesis chip_pin = "AB3" */;
  input LOAD         /* synthesis chip_pin = "AA6" */;
     ...
```

- (1) It must correspond to the determined target device, and the attribute statements in this book apply only to Quartus II;
- (2) It can only be defined in the top-level design file.