

LAB 5: Introduction to Quartus || software
Design
ITI1100 B -Digital Systems
Winter 2017
University of Ottawa

LAB session 5

Group **16** :

Yibo Wang, 8070847

Kang Jiang, 7985594

Experiment Date: 16/03/2017

Submission Date: 30/03/2017

Objectives:

Provide insight into the characteristics of several important latches and flip-flops.
Build latches and flip-flops from basic gates.
Explain concepts of latching and edge-Triggering.
Test latches and flip-flops to understand their operation

Equipment and Components

QUARTUS || 13.0 Service-Pick 1
Alter DE2-115 circuit board

PART 1

Circuit diagrams

Simple logic circuit

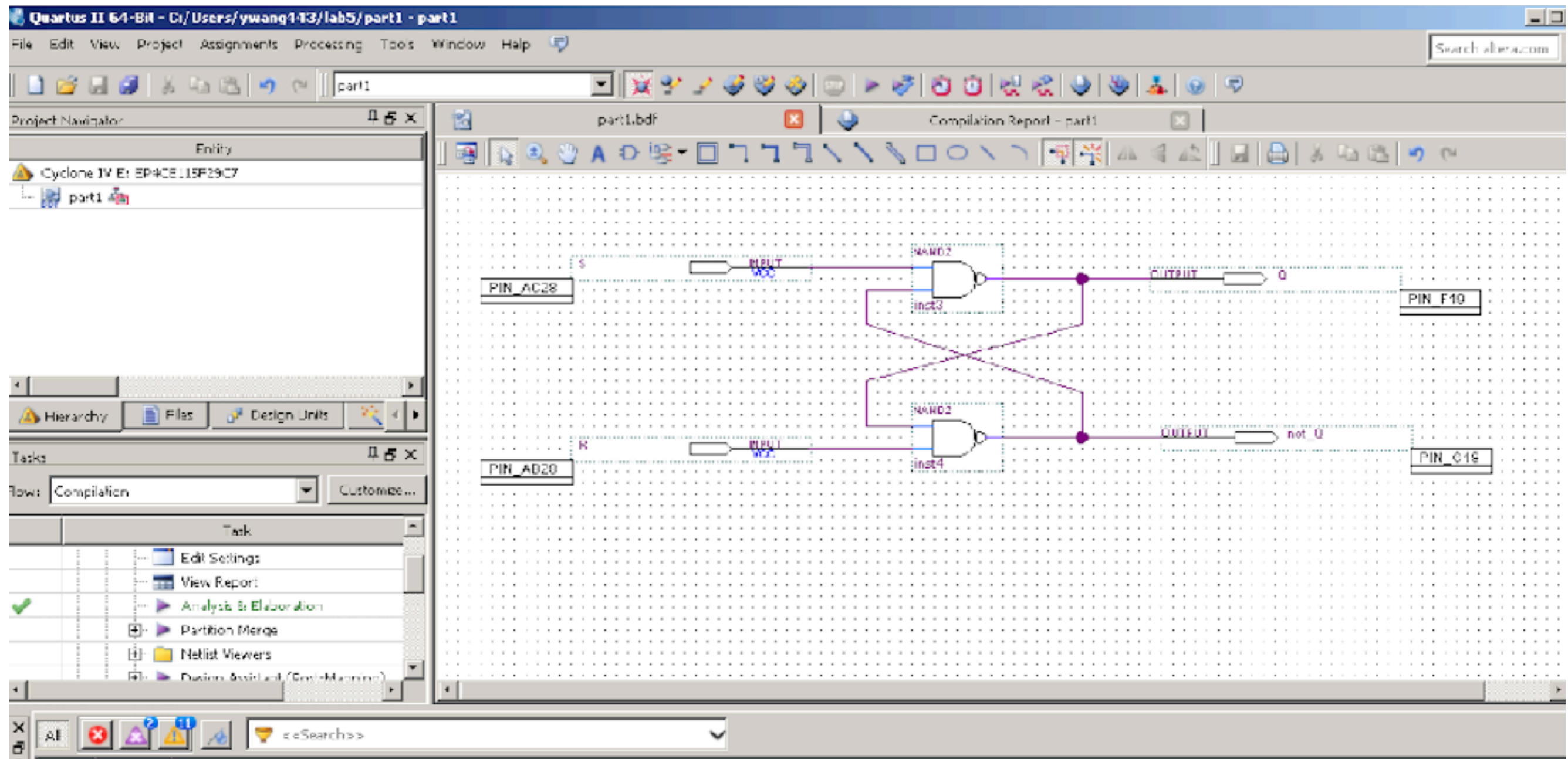


Figure 1: (Screen-shot of a simple circuit with AND, XOR, OR gates)

PART 2

Circuit diagrams

Simple logic circuit

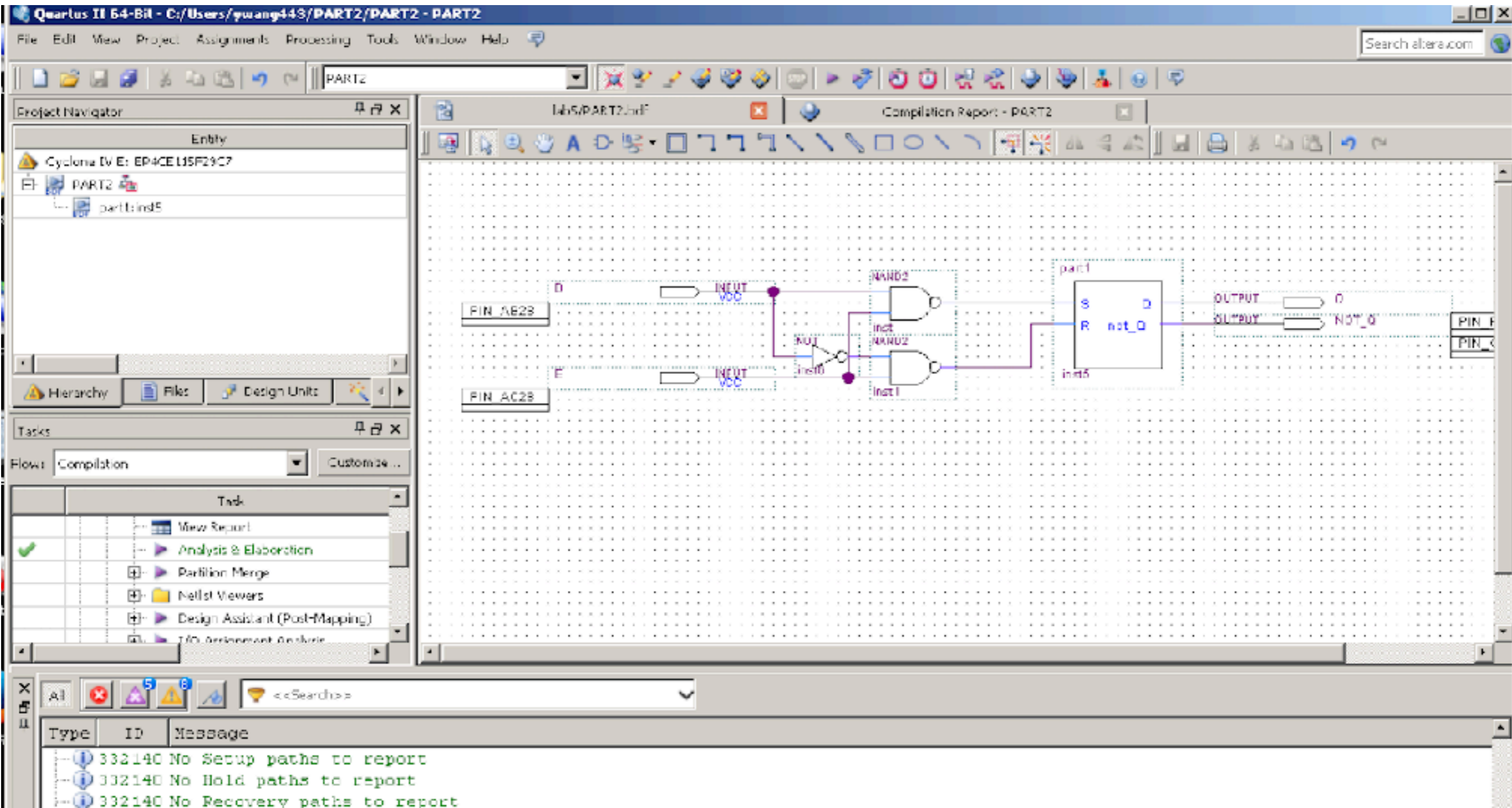


Figure 1: (Screen-shot of a simple circuit with – Parallel Adder)

PART 3

Circuit diagrams

Simple logic circuit

The image shows the Quartus II software interface for a logic circuit design. The main workspace displays a circuit diagram on a grid. The circuit includes:

- Two input pins: **PIN_A02B** and **PIN_B14**.
- Two inverters: **NOT1** and **NOT2**.
- Three NAND gates: **inst1**, **inst2**, and **inst3**.
- A D flip-flop component labeled **part1** with inputs **S** (Set) and **R** (Reset), and output **not_Q**.
- Two output pins: **PIN_F14** and **PIN_G15**.

The circuit logic is as follows: **PIN_A02B** is inverted by **NOT1** and then combined with **PIN_B14** in NAND gate **inst1**. **PIN_B14** is also inverted by **NOT2** and combined with the output of **inst1** in NAND gate **inst2**. The output of **inst2** is combined with **PIN_A02B** in NAND gate **inst3**. The output of **inst3** is connected to the **S** input of the flip-flop. The output of **inst1** is connected to the **R** input of the flip-flop. The flip-flop's output **not_Q** is connected to **PIN_F14** and **PIN_G15**.

On the left side, the Project Navigator shows the project structure for **PART3**. The Tasks pane shows the compilation process, with the following messages:

| Type | ID | Message |
|-------------|--------|-----------------------------|
| Information | 832140 | No Setup paths to report |
| Information | 832140 | No Hold paths to report |
| Information | 832140 | No Recovery paths to report |

PART 4

Circuit diagrams

Simple logic circuit

The screenshot displays the Quartus II software interface. The main workspace shows a logic circuit diagram on a grid. The circuit includes:

- Two input pins: PIN_A25 and PIN_A28.
- A 2-input OR gate with output 'out2'.
- A D flip-flop component labeled 'PAR2' with inputs 'CLK' and 'NOT_0', and output 'OUTOUT'.
- A 2-input AND gate with inputs 'OUTOUT' and 'NOT_2', and output 'PIN_F19'.
- A clock signal 'CLK' is connected to the flip-flop's clock input.
- Other components include a 'NOT' gate and a 'PIN_B16' output pin.

The bottom-left pane shows the 'Tasks' window with the following tasks listed:

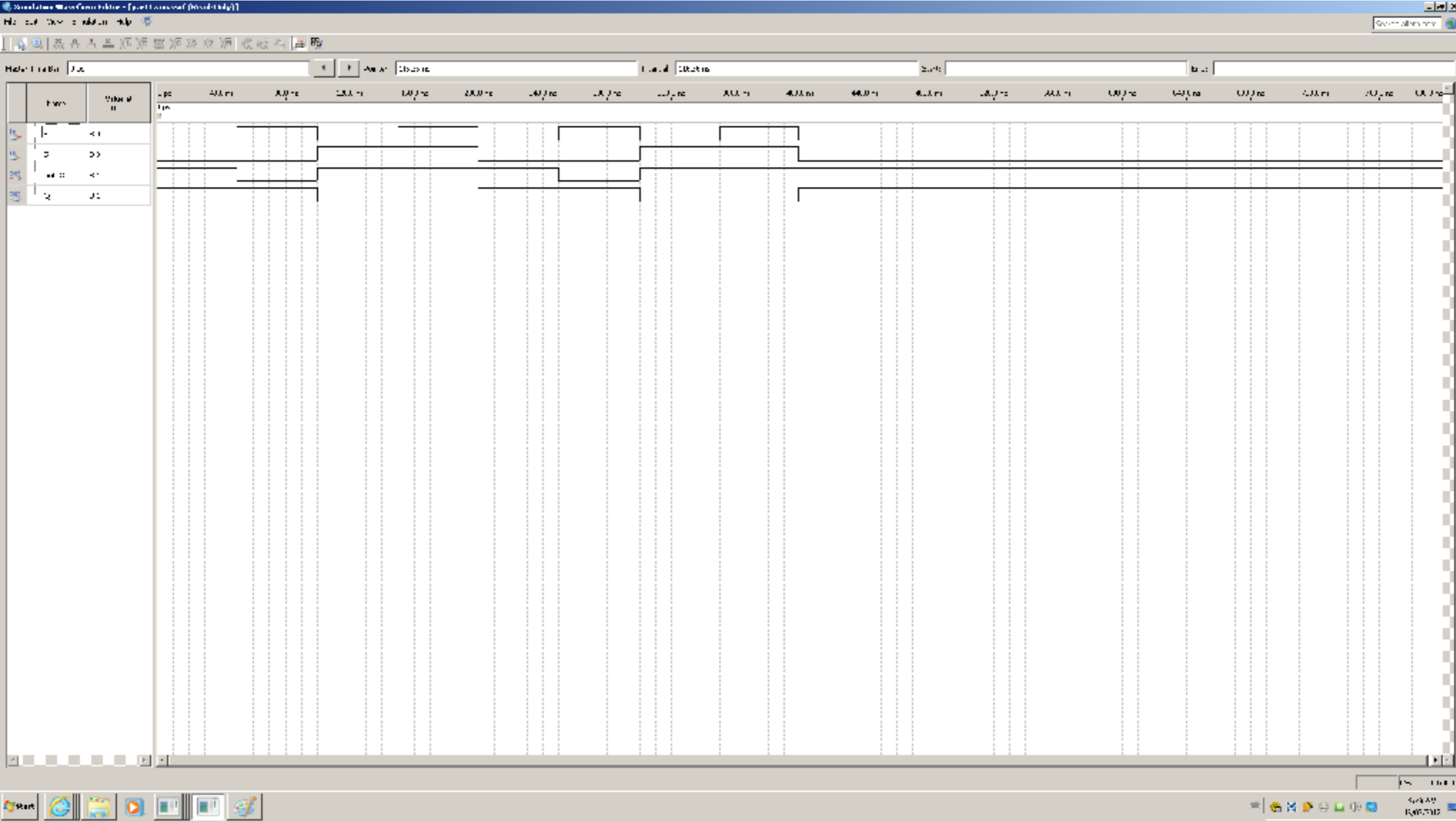
- Compile Design
- Analysis & Synthesis
 - Edit Settings
 - New Report
- Analysis & Elaboration
- Partition Merge
- Netlist Viewers
- Design Assistant (Post-Mapping)
- I/O Assignment Analysis

The bottom-right pane shows the 'Messages' window with the following error messages:

- 332140 No Setup paths to report
- 332140 No Hold paths to report
- 332148 Timing requirements not met
- 332145 Worst-case recovery slack is -1.019
- 332145 Worst-case removal slack is 0.374

Experiment Data and Data Processing

Simple logic circuit



Finger 2:(Simulation Output waveform from the simple logic circuit) **part1**

Experiment Data and Data Processing

Simple logic circuit

Finger 2:(Simulation Output waveform from the simple logic circuit) **part2**

Table 1(PART1): Experimental data observed from the Altera DE2-115 board for simple circuit designed

| Input given from the slide switches | | Observed Output from the board | |
|-------------------------------------|---|--------------------------------|-------|
| R | S | Q | Not Q |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |

Table 2(PART1): Comparison of Theoretical and Experimental results for the simple circuit

| Input given from the slide switches | | Expected Results | | Actual Results | |
|-------------------------------------|---|------------------|-------|----------------|-------|
| R | S | Q | Not Q | Q | Not Q |
| 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 1 | 0 |

Table 3(PART1)

| Input given from the slide switches | | Observed Output from the board | |
|-------------------------------------|---|--------------------------------|--------------------|
| R | S | Q | Not Q |
| 0 | 0 | 1(LED didn't glow) | 1(LED didn't glow) |
| 0 | 1 | 1(LED didn't glow) | 0(LED did glow) |
| 1 | 0 | 0(LED did glow) | 1(LED didn't glow) |
| 1 | 1 | 1(LED didn't glow) | 0(LED did glow) |

Discussions & Conclusion

We used Boolean logic to express complicated logic circuits into simplest form. We also used AND, XOR, OR gates to express full adder, then used full adders to express parallel adder. We also use adder function table to do the sum in both binary and hexadecimal, and showed the carry output separately.

In our experiment, we created two projects, designed their diagrams, compiled them, assigned pin names. Then, we simulated them and compared the experimental results with our theoretical results. And find the expected results and actual results and identical.