

MUNI
FI

PV200

Introduction to hardware description languages

Week 02: Combination logic - Schematic, basic gates

Ing. Jiří Čulen, 5.10.2021

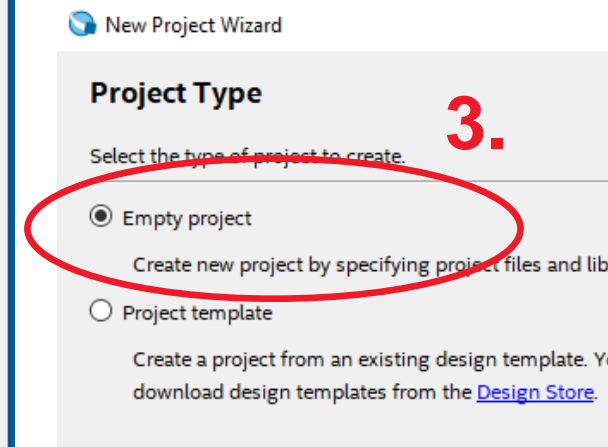
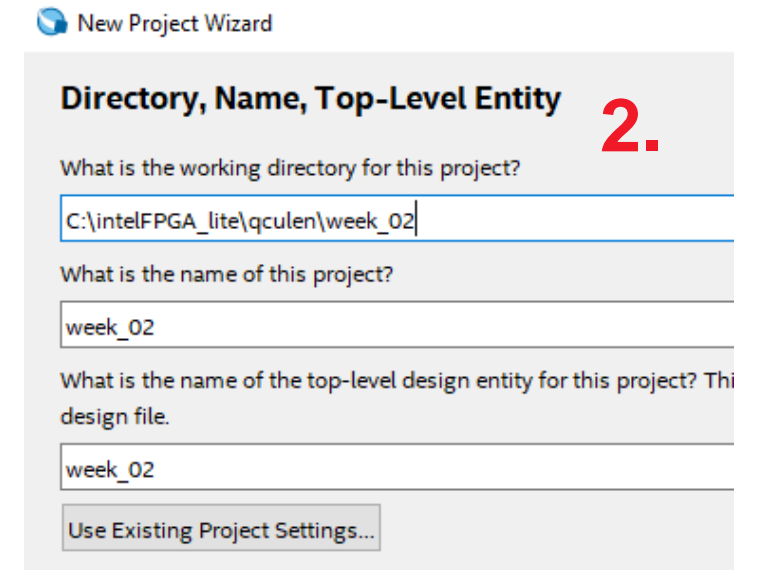
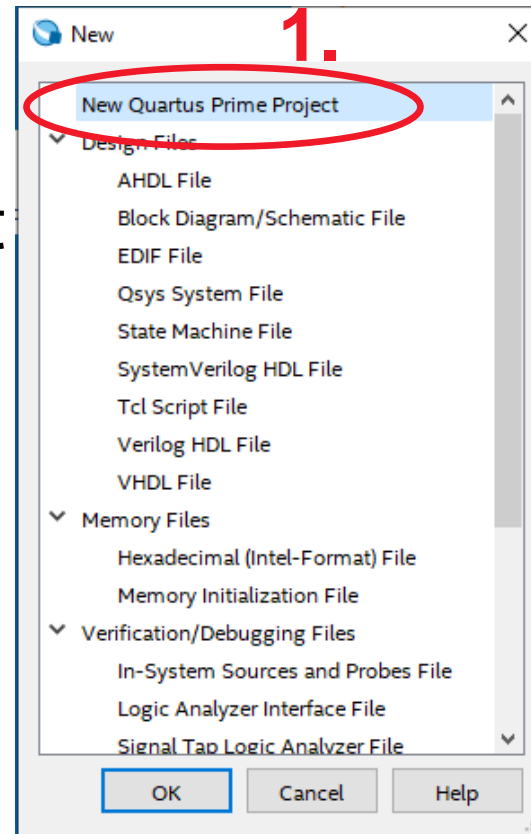


Agenda

1. New empty project in the Quartus
2. Schematic editor
3. I/O settings
4. Programming the device
5. Half adder
6. Full adder
7. Adding of two bits number

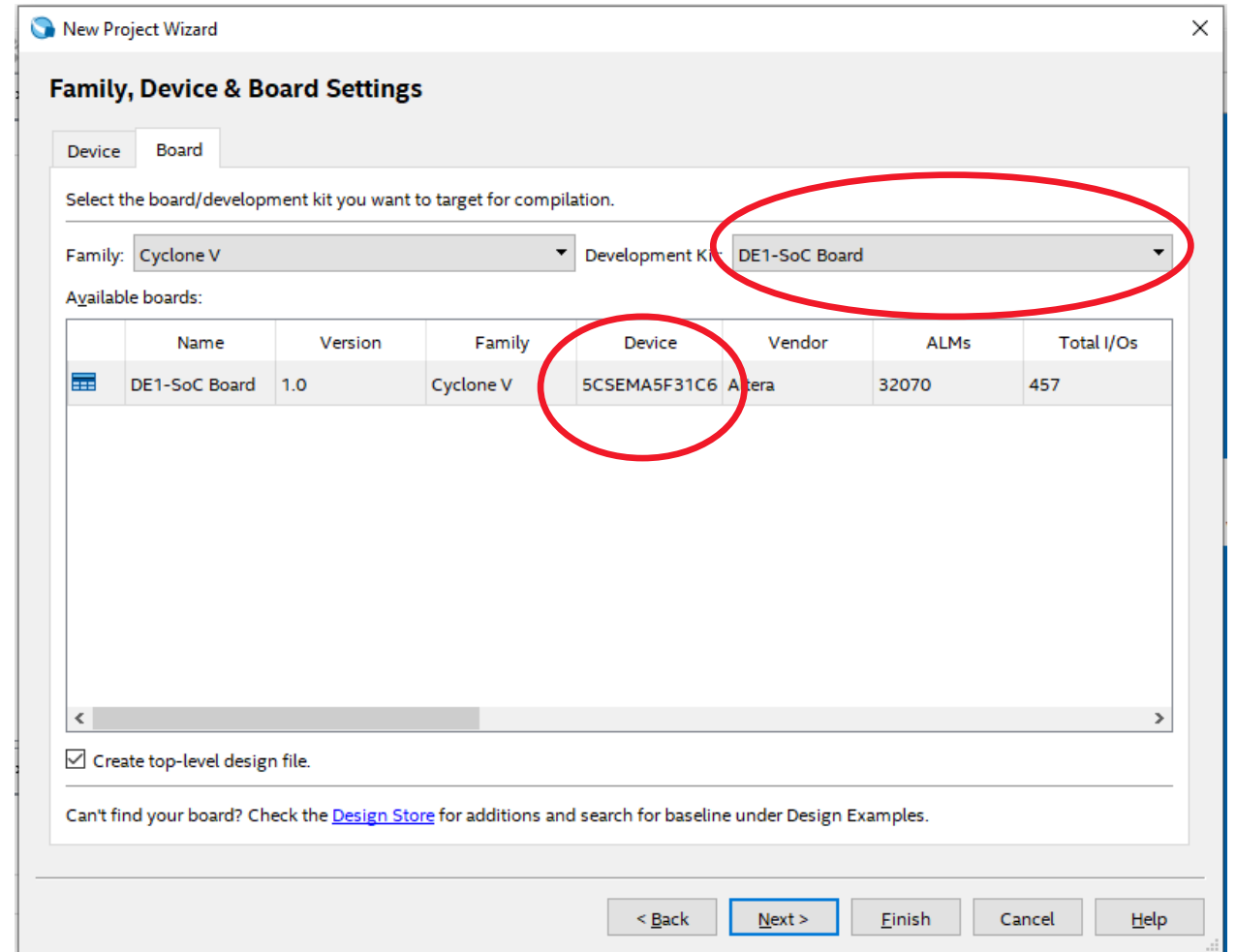
New empty project in the Quartus

- File -> New ->New Quartus Prime Project
- Select working directory and name of the project
- Define name of the project
- Project type – Empty project
- Skip add files



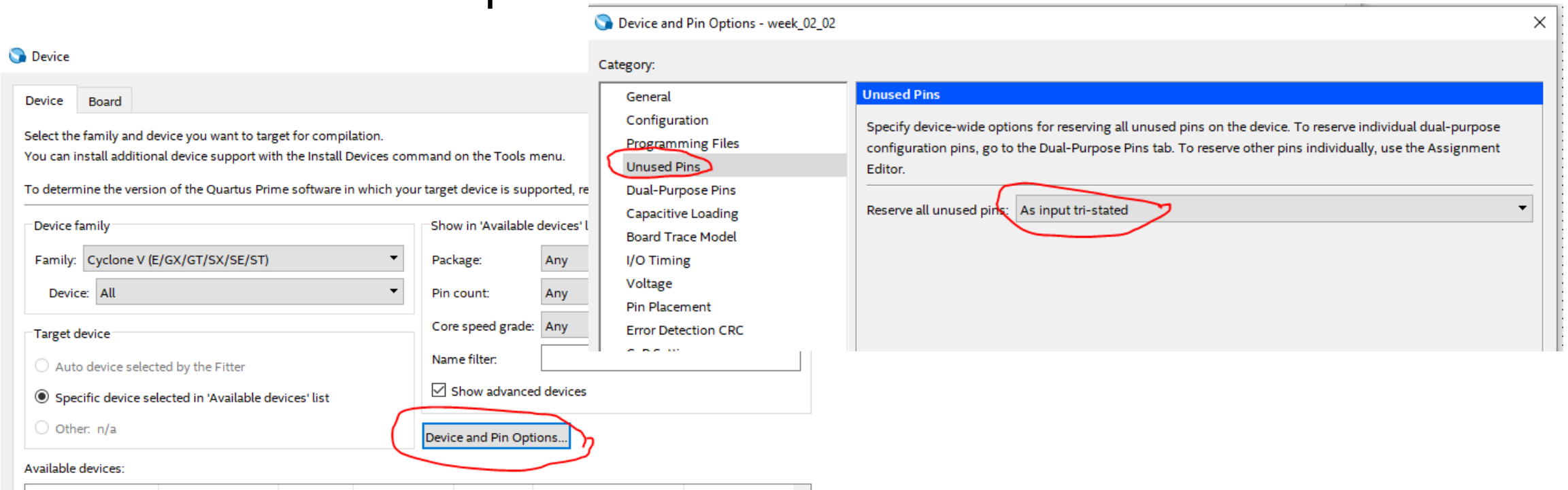
New empty project in the Quartus

- Select device from the Board DE1-SoC or manually 5CSEMA5F31C6
- Skip EDA Tool Settings
- Finish



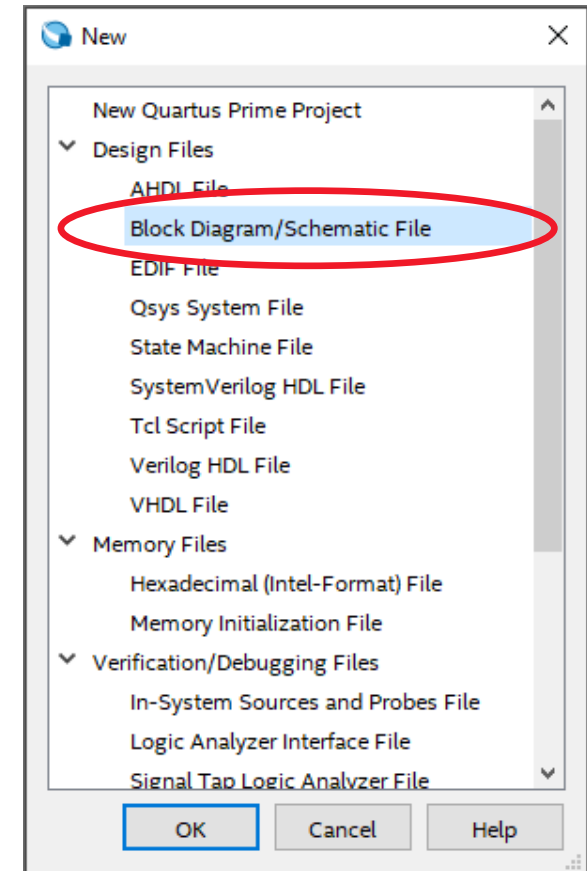
New empty project in the Quartus

- Assignment -> Device -> Device and Pin Options
- Unused Pins -> As input tri-stated



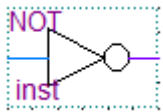
Schematic editor

– File -> New -> Block Diagram/Schematic File

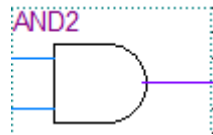


Schematic editor

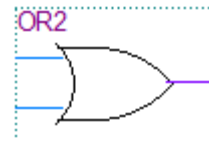
– We will need four basic gates



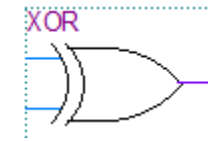
| A | Q |
|---|---|
| 0 | 1 |
| 1 | 0 |



| A | B | Q |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |



| A | B | Q |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |



| A | B | Q |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Schematic editor

- We will use keys from the board, which are in negative logic. We need to invert it for positive logic.

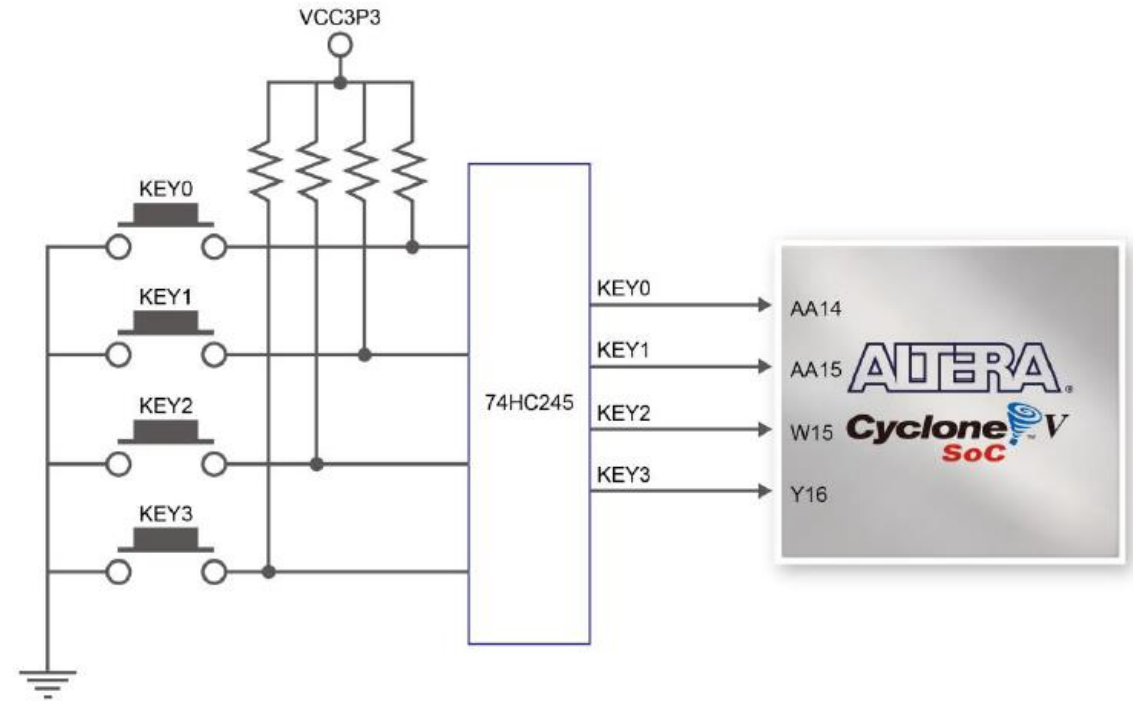
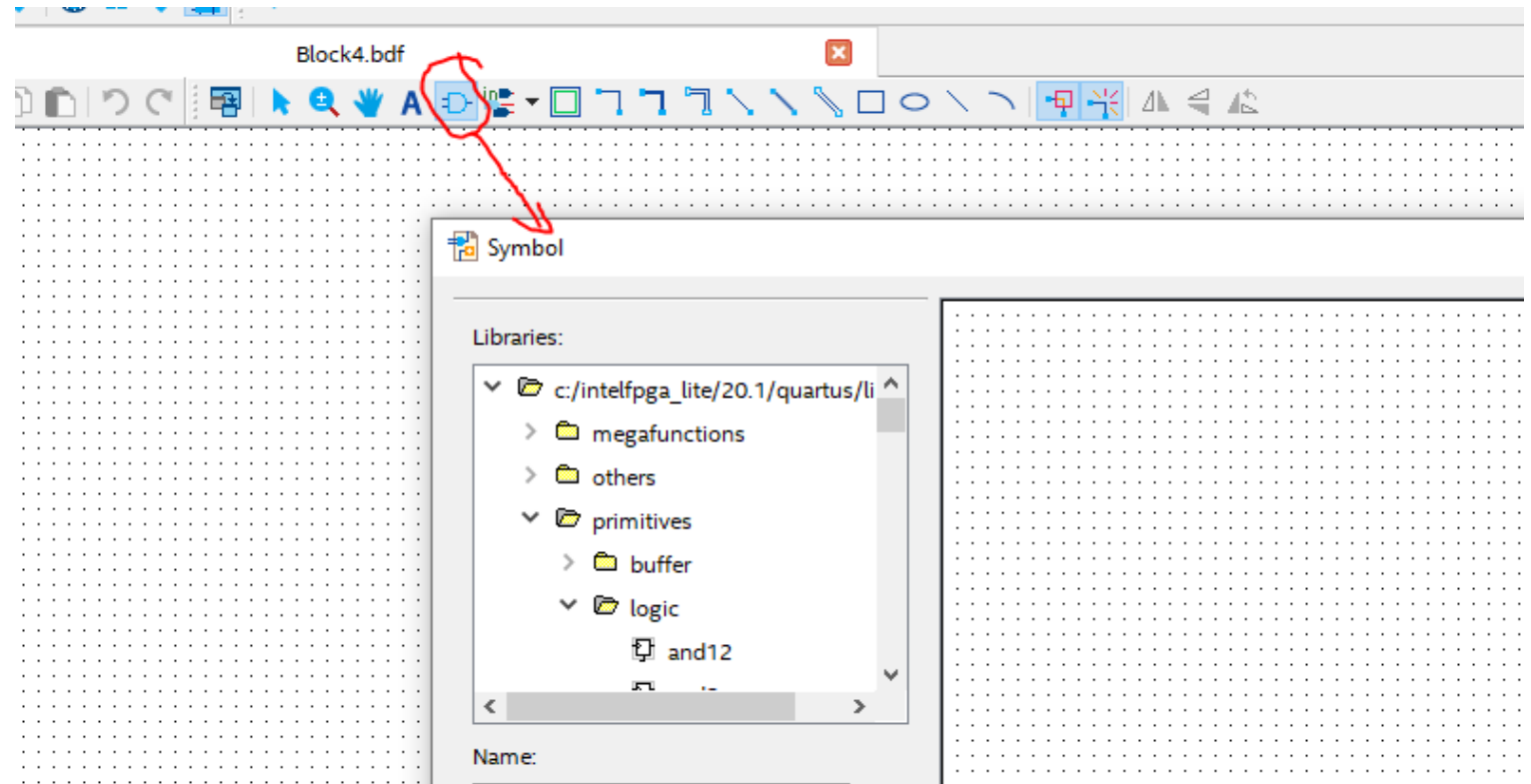


Figure 3-14 Connections between the push-buttons and the Cyclone V SoC FPGA

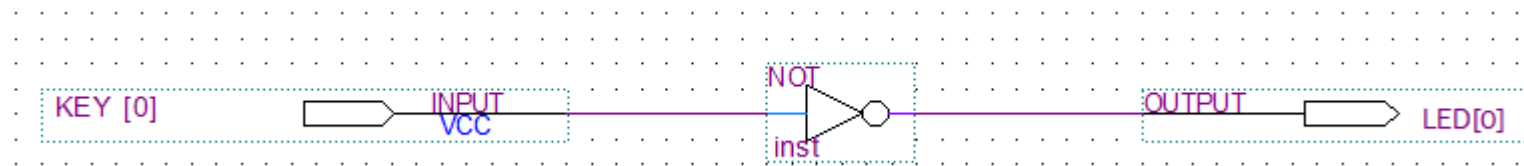
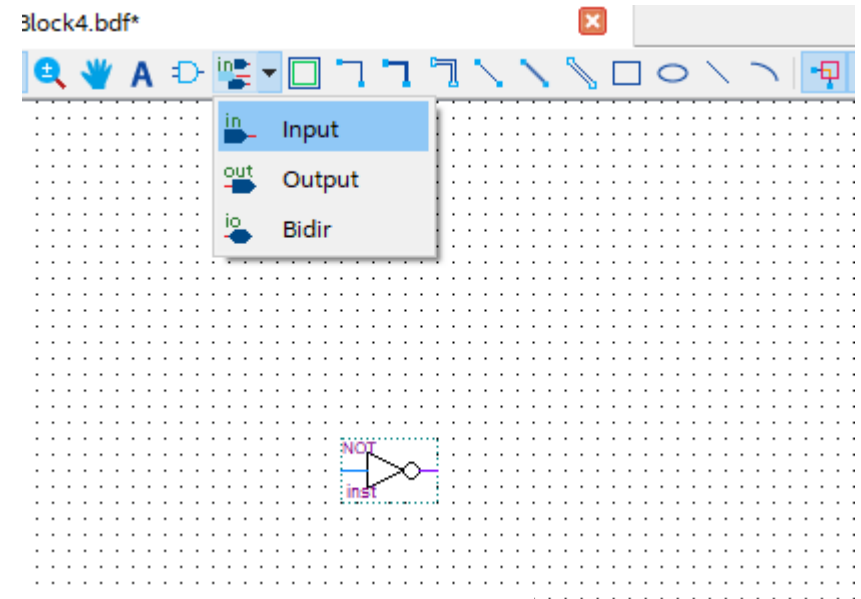
Schematic editor

–Place a symbol not



Schematic editor

- Place one input port and one output port and connect it to the NOT gate.
- The name of the input port will KEY[0], of the output LED[0].
- Save the schematic file.



I/O settings

- Assignment of keys from the DE1-SoC manual

Table 3-7 Pin Assignment of Push-buttons

| <i>Signal Name</i> | <i>FPGA Pin No.</i> | <i>Description</i> | <i>I/O Standard</i> |
|--------------------|---------------------|-----------------------|---------------------|
| KEY[0] | PIN_AA14 | Push-button[0] | 3.3V |
| KEY[1] | PIN_AA15 | Push-button[1] | 3.3V |
| KEY[2] | PIN_W15 | Push-button[2] | 3.3V |
| KEY[3] | PIN_Y16 | Push-button[3] | 3.3V |

I/O settings

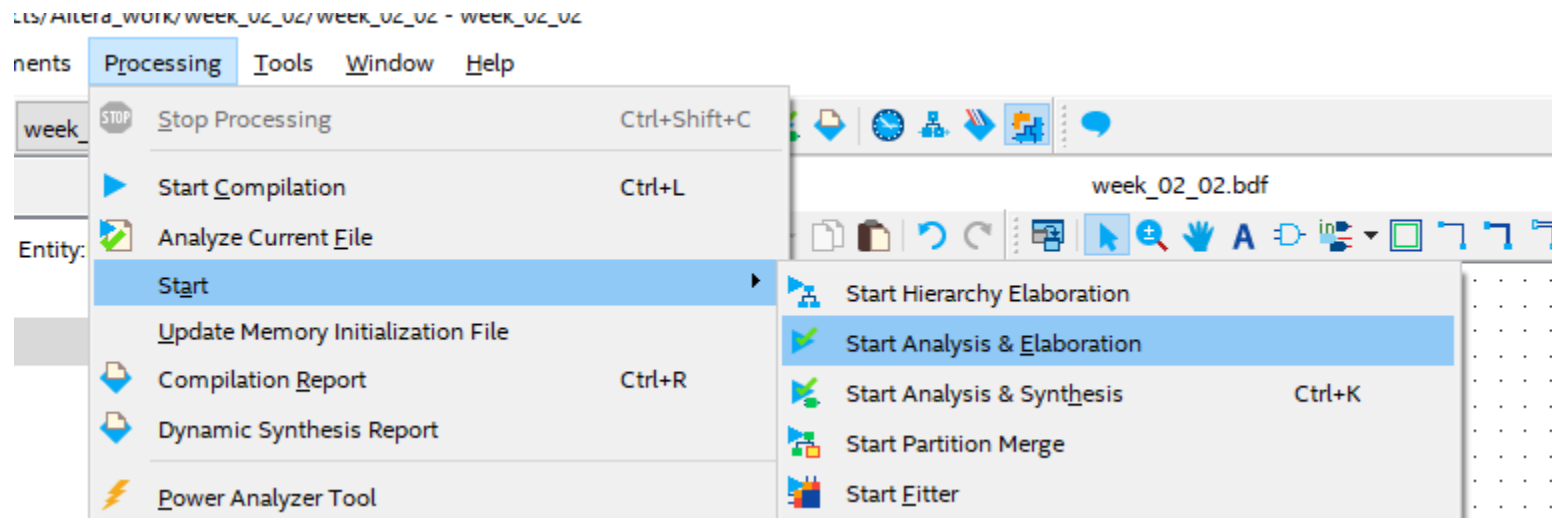
- Assignment of LEDs from the DE1-SoC manual

Table 3-8 Pin Assignment of LEDs

| <i>Signal Name</i> | <i>FPGA Pin No.</i> | <i>Description</i> | <i>I/O Standard</i> |
|--------------------|---------------------|--------------------|---------------------|
| LEDR[0] | PIN_V16 | LED [0] | 3.3V |
| LEDR[1] | PIN_W16 | LED [1] | 3.3V |
| LEDR[2] | PIN_V17 | LED [2] | 3.3V |
| LEDR[3] | PIN_V18 | LED [3] | 3.3V |
| LEDR[4] | PIN_W17 | LED [4] | 3.3V |
| LEDR[5] | PIN_W19 | LED [5] | 3.3V |
| LEDR[6] | PIN_Y19 | LED [6] | 3.3V |
| LEDR[7] | PIN_W20 | LED [7] | 3.3V |
| LEDR[8] | PIN_W21 | LED [8] | 3.3V |
| LEDR[9] | PIN_Y21 | LED [9] | 3.3V |

I/O settings

– Processing -> Start -> Start Analysis & Elaboration



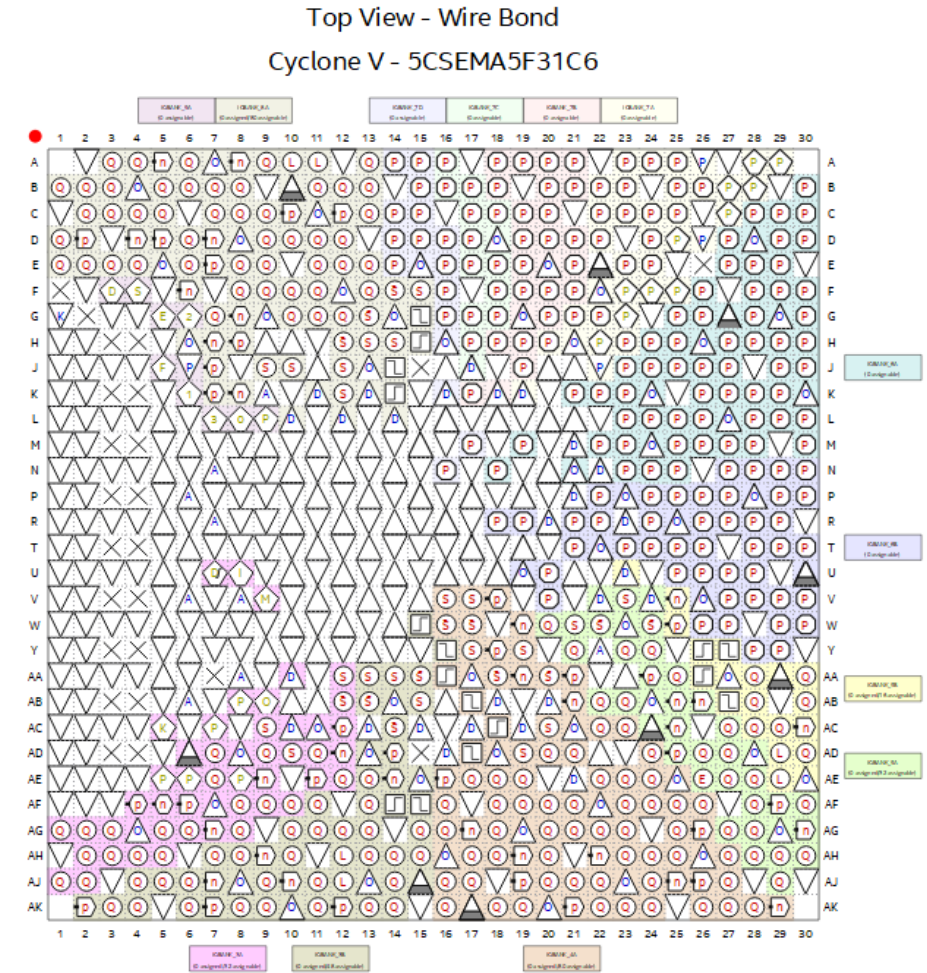
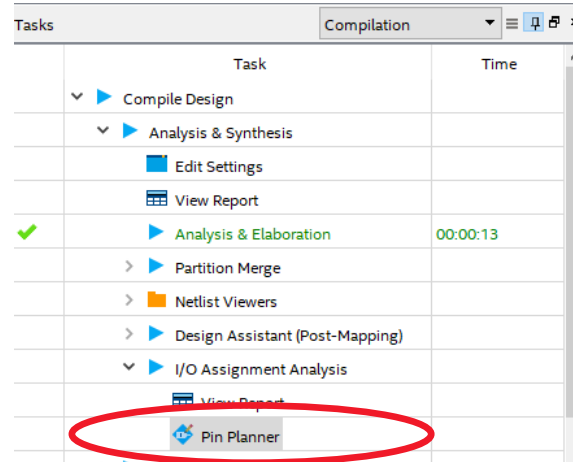
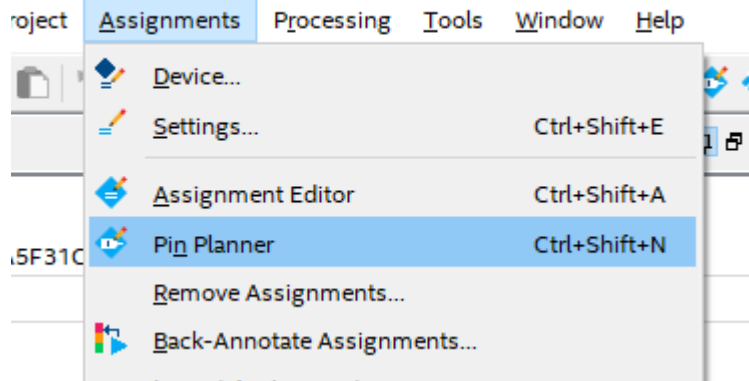
The screenshot shows the 'Tasks' window in a software application. The window is titled 'Compilation' and contains a table with two columns: 'Task' and 'Time'. The table lists several tasks, including 'Compile Design', 'Analysis & Synthesis', 'Edit Settings', 'View Report', 'Analysis & Elaboration', 'Partition Merge', 'Netlist Viewers', 'Design Assistant (Post-Mapping)', and 'I/O Assignment Analysis'. The 'Analysis & Elaboration' task is highlighted with a red oval and has a green checkmark in the 'Time' column, indicating it is completed. The time for this task is '00:00:13'.

| Task | Time |
|---------------------------------|----------|
| Compile Design | |
| Analysis & Synthesis | |
| Edit Settings | |
| View Report | |
| Analysis & Elaboration | 00:00:13 |
| Partition Merge | |
| Netlist Viewers | |
| Design Assistant (Post-Mapping) | |
| I/O Assignment Analysis | |
| View Report | |

I/O settings

— Assignments -> Pin Planner

dition - C:/Projects/Altera_work/week_02_02/week_02_02 - week_02_02

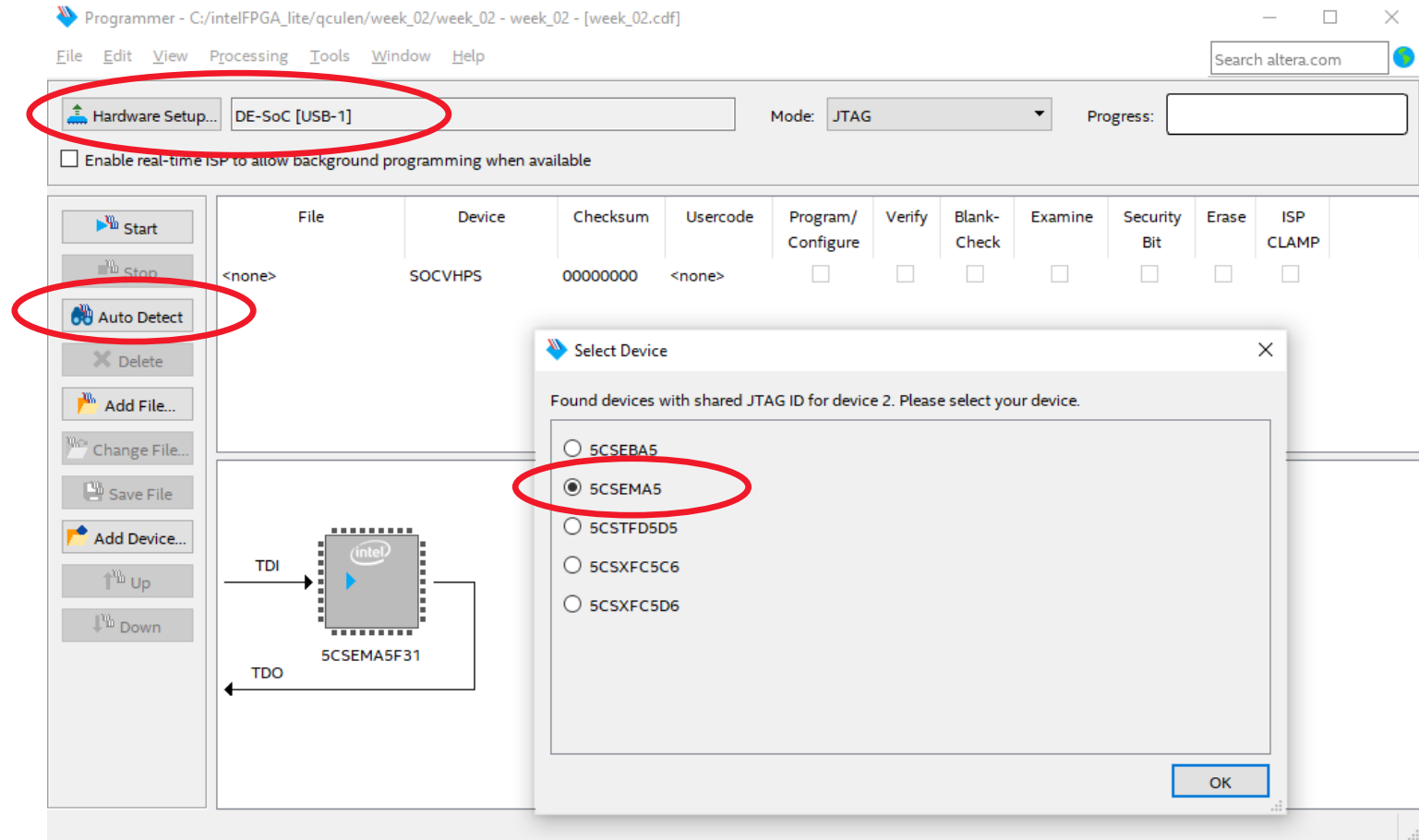


— Set the right parameters for our pins

| Node Name | Direction | Location | I/O Bank | VREF Group | I/O Standard | Reserved | Current Strength | Slew Rate | Diff |
|--------------|-----------|----------|----------|------------|--------------|----------|------------------|-----------|------|
| in KEY[0] | Input | PIN_AA14 | 3B | B3B_N0 | 3.3-V LVCMOS | | 2mA (default) | | |
| out LED[0] | Output | PIN_V16 | 4A | B4A_N0 | 3.3-V LVCMOS | | 2mA (default) | 0 | |
| <<new node>> | | | | | | | | | |

Programming the device

- Compile the project
- Open the programmer
- Check connection to the DE SoC board
- Select 5CSEMA5
- Detect your FPGA



Programming the device

- FPGA 5CSEMA5 is detected

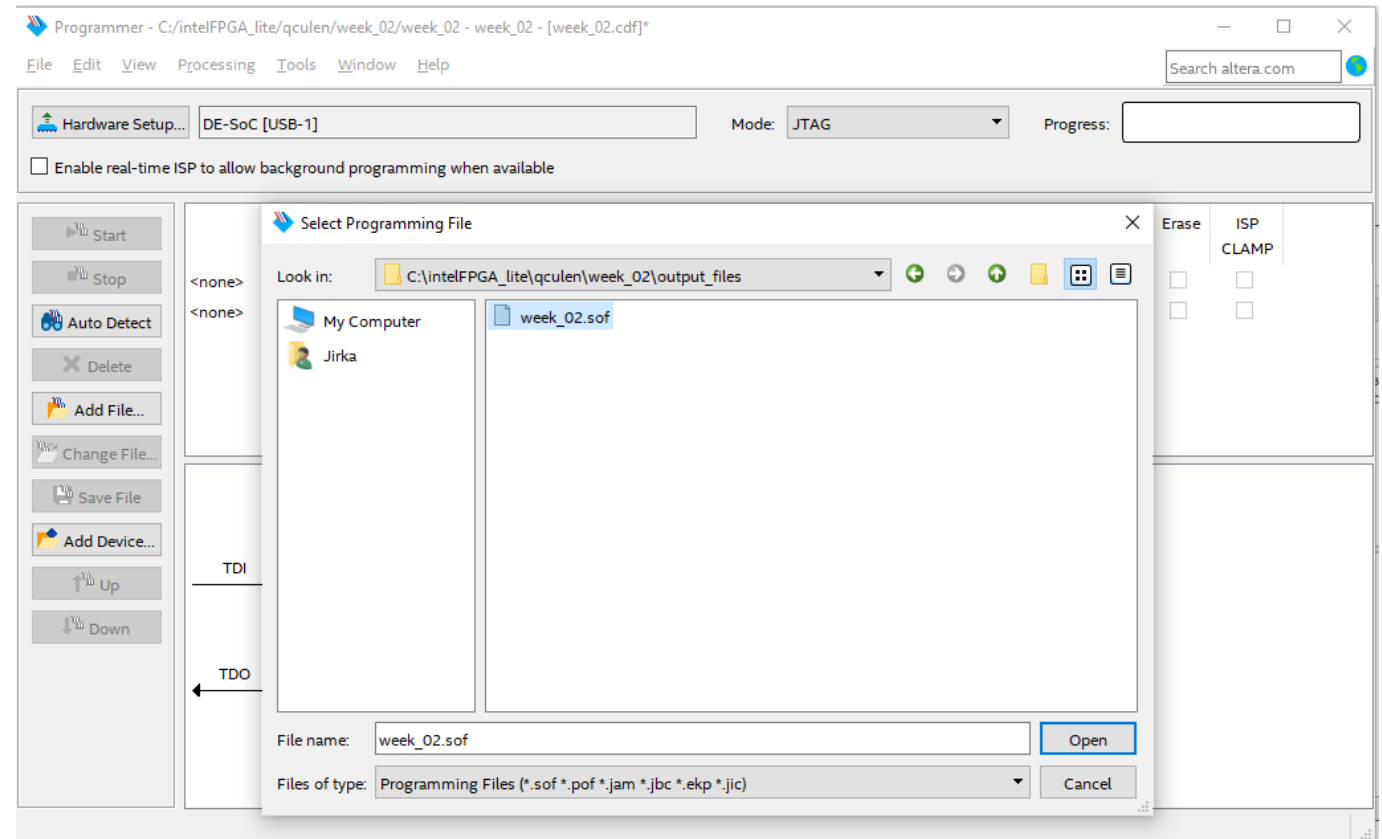
The screenshot shows the Intel Programmer software interface. The title bar indicates the file path: C:/intelFPGA_lite/qcuden/week_02/week_02 - week_02 - [week_02.cdf]*. The hardware setup is DE-SoC [USB-1] and the mode is JTAG. A table lists detected devices, with 5CSEMA5 circled in red. Below the table is a schematic diagram showing the connection between the SOC VHPS and the 5CSEMA5 device.

| File | Device | Checksum | Usercode | Program/Configure | Verify | Blank-Check | Examine | Security Bit | Erase | ISP CLAMP |
|--------|---------|----------|----------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| <none> | SOCVHPS | 00000000 | <none> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| <none> | 5CSEMA5 | 00000000 | <none> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

The schematic diagram shows two Intel devices: SOC VHPS and 5CSEMA5. A TDI (Test Data In) signal is connected to the SOC VHPS, and a TDO (Test Data Out) signal is connected to the 5CSEMA5. The devices are connected via a JTAG chain.

Programming the device

- Add file to the 5CSEMA5 device from the project



Programming the device

— Delete device 5CSEMA5

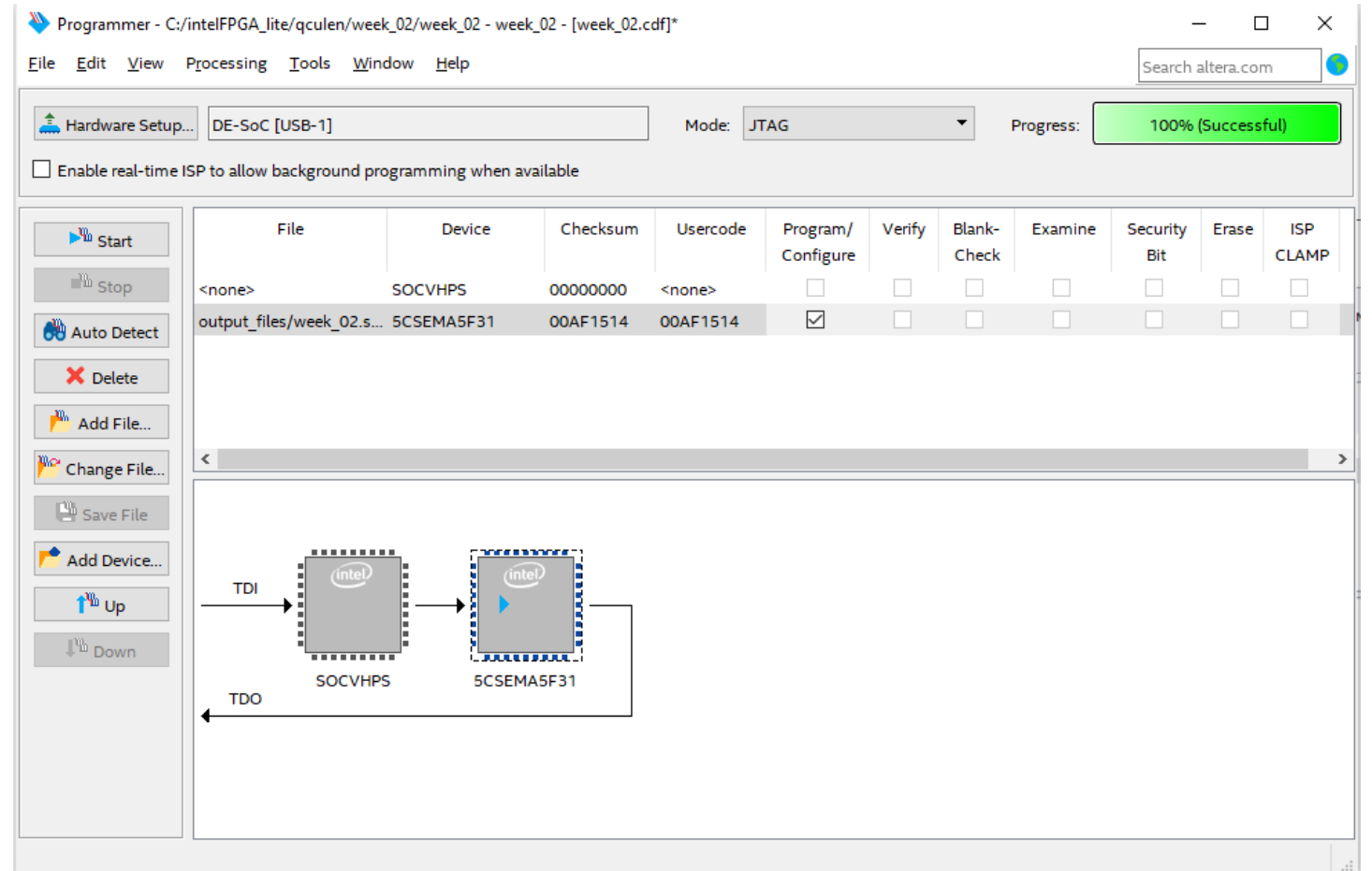
The screenshot shows the Intel Quartus Programmer interface. The window title is "Programmer - C:/intelFPGA_lite/qcuden/week_02/week_02 - week_02 - [week_02.cdf]*". The hardware setup is "DE-SoC [USB-1]" in "JTAG" mode. A table lists the devices to be programmed:

| File | Device | Checksum | Usercode | Program/Configure | Verify | Blank-Check | Examine | Security Bit | Erase | ISP CLAMP |
|---------------------------|------------|----------|----------|-------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| <none> | SOCVHPS | 00000000 | <none> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| <none> | 5CSEMA5 | 00000000 | <none> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| output_files/week_02.s... | 5CSEMA5F31 | 00AF1514 | 00AF1514 | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

Below the table is a schematic diagram showing three Intel FPGAs: SOCVHPS, 5CSEMA5, and 5CSEMA5F31. TDI (Test Data In) is connected to the SOCVHPS, and TDO (Test Data Out) is connected to the 5CSEMA5F31. The 5CSEMA5 device is highlighted with a dashed blue border, indicating it is the target of the current operation.

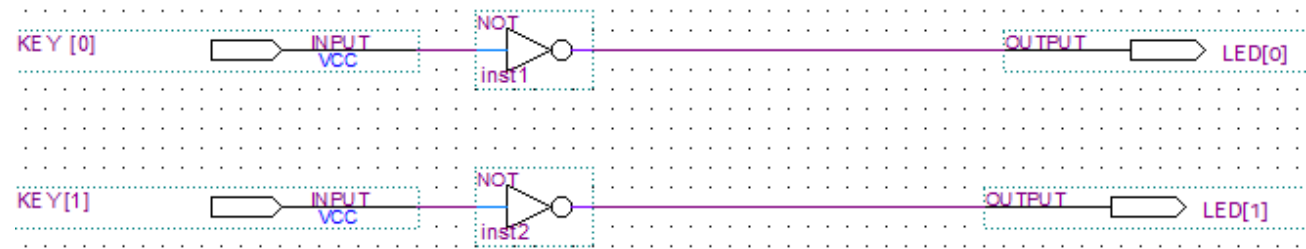
Programming the device

- Select 5CSEMA5F31
- Start the programming
- Test your application



Half adder

- Add the next pair of KEY and LED.
- For loading new ports to the I/O planner, You need repeat start of the Analysis & Elaboration
- Compile it and test it.

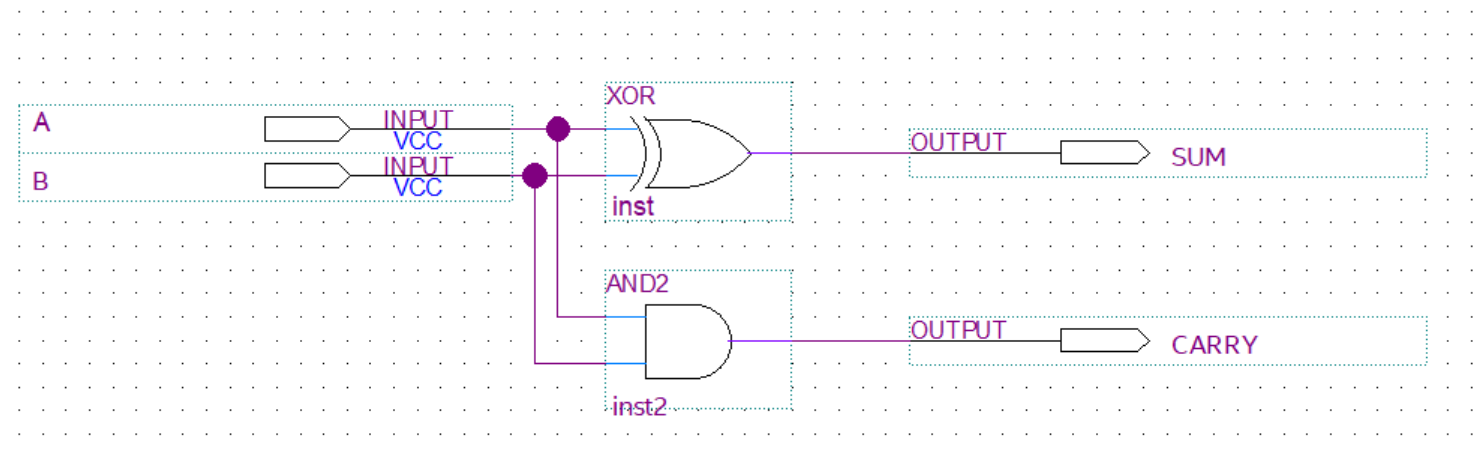


| Node Name | Direction | Location | I/O Bank | VREF Group | Fitter Location | I/O Standard | Reserved | Current Strength | Slew Rate |
|------------|-----------|----------|----------|------------|-----------------|--------------|----------|------------------|-----------|
| in KEY[1] | Input | PIN_AA15 | 3B | B3B_N0 | PIN_AA15 | 3.3-V LVCMOS | | 2mA (default) | |
| in KEY[0] | Input | PIN_AA14 | 3B | B3B_N0 | PIN_AA14 | 3.3-V LVCMOS | | 2mA (default) | |
| out LED[1] | Output | PIN_W16 | 4A | B4A_N0 | PIN_W16 | 3.3-V LVCMOS | | 2mA (default) | 0 |
| out LED[0] | Output | PIN_V16 | 4A | B4A_N0 | PIN_V16 | 3.3-V LVCMOS | | 2mA (default) | 0 |

Half adder

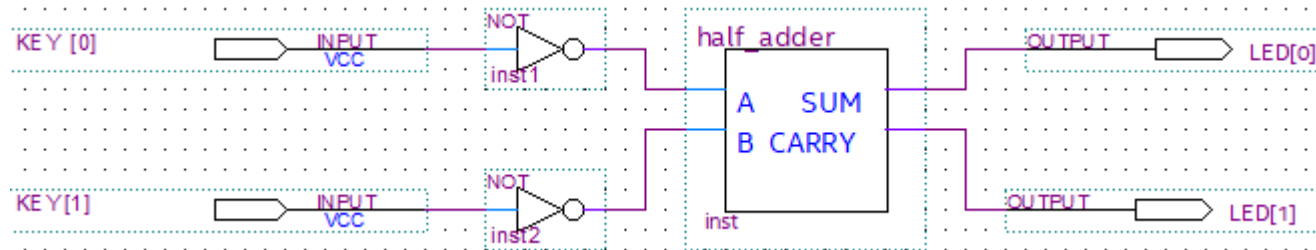
- Create a new schematic file “half_adder”
- Use File -> Create/Update -> Create Symbol Files for Current File

| A | B | SUM | CARRY |
|---|---|-----|-------|
| 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 |



Half adder

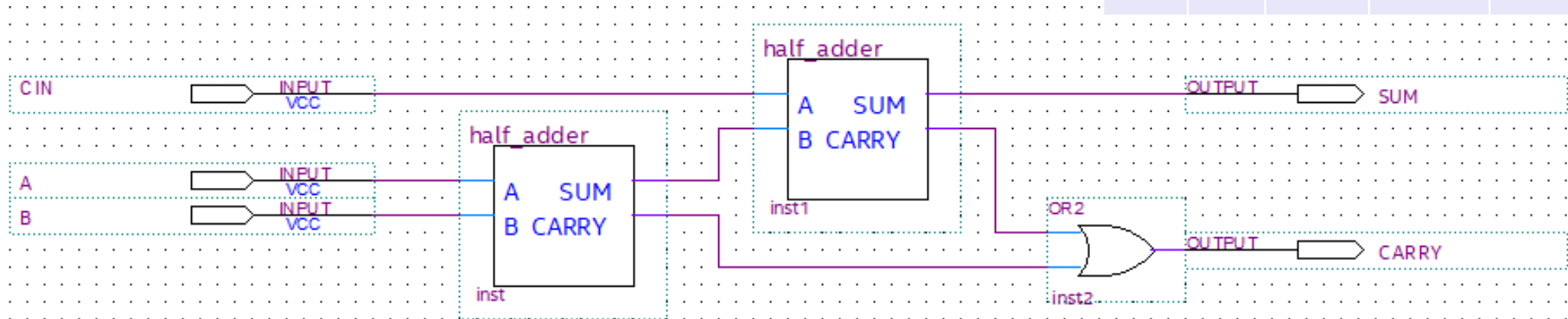
- Put the half adder to the top schematic
- Compile it, program it, and test it...



Full adder

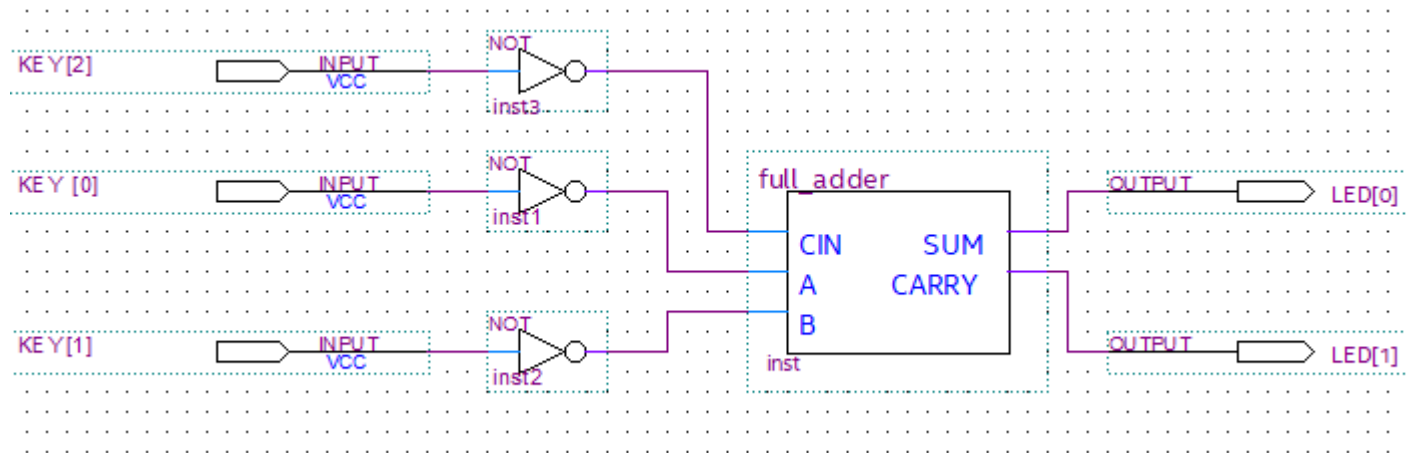
- We can build full adder by using of half adders or by many others way...
- Create the new schematic file “full_adder”, and create symbol files

| A | B | CIN | SUM | CARRY |
|---|---|-----|-----|-------|
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |



Full adder

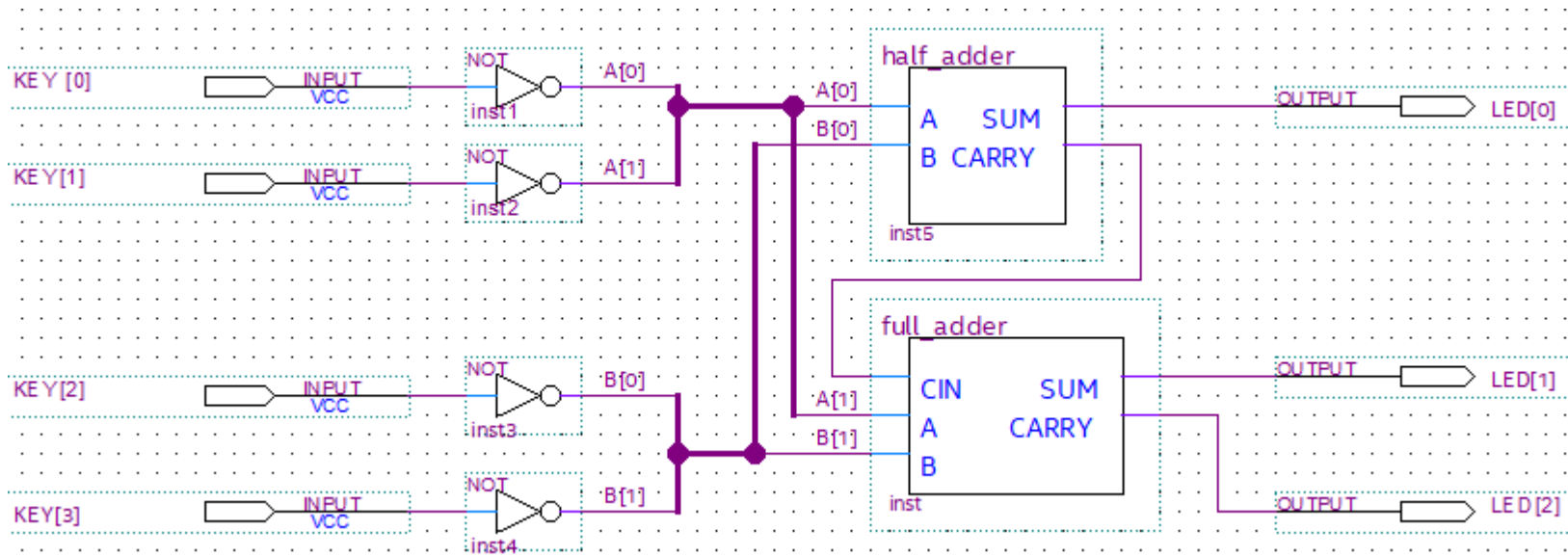
- Put the half adder to the top schematic
- Compile it, program it, and test it...



| Node Name | Direction | Location | I/O Bank | VREF Group | Fitter Location | I/O Standard | Reserved | Current Strength | Slew Rate |
|------------|-----------|----------|----------|------------|-----------------|--------------|----------|------------------|-----------|
| in KEY[2] | Input | PIN_W15 | 3B | B3B_NO | PIN_W15 | 3.3-V LVCMOS | | 2mA (default) | |
| in KEY[1] | Input | PIN_AA15 | 3B | B3B_NO | PIN_AA15 | 3.3-V LVCMOS | | 2mA (default) | |
| in KEY[0] | Input | PIN_AA14 | 3B | B3B_NO | PIN_AA14 | 3.3-V LVCMOS | | 2mA (default) | |
| out LED[1] | Output | PIN_W16 | 4A | B4A_NO | PIN_W16 | 3.3-V LVCMOS | | 2mA (default) | 0 |
| out LED[0] | Output | PIN_V16 | 4A | B4A_NO | PIN_V16 | 3.3-V LVCMOS | | 2mA (default) | 0 |

Adding of two bits number

– Modify the top schema and test it



| Node Name | Direction | Location | I/O Bank | VREF Group | Fitter Location | I/O Standard | Reserved | Current Strength | Slew Rate |
|------------|-----------|----------|----------|------------|-----------------|--------------|----------|------------------|-----------|
| in KEY[3] | Input | PIN_Y16 | 3B | B3B_NO | PIN_Y16 | 3.3-V LVCMOS | | 2mA (default) | |
| in KEY[2] | Input | PIN_W15 | 3B | B3B_NO | PIN_W15 | 3.3-V LVCMOS | | 2mA (default) | |
| in KEY[1] | Input | PIN_AA15 | 3B | B3B_NO | PIN_AA15 | 3.3-V LVCMOS | | 2mA (default) | |
| in KEY[0] | Input | PIN_AA14 | 3B | B3B_NO | PIN_AA14 | 3.3-V LVCMOS | | 2mA (default) | |
| out LED[2] | Output | PIN_V17 | 4A | B4A_NO | PIN_V17 | 3.3-V LVCMOS | | 2mA (default) | 0 |
| out LED[1] | Output | PIN_W16 | 4A | B4A_NO | PIN_W16 | 3.3-V LVCMOS | | 2mA (default) | 0 |
| out LED[0] | Output | PIN_V16 | 4A | B4A_NO | PIN_V16 | 3.3-V LVCMOS | | 2mA (default) | 0 |

MUNI
FI

Thank you for attention

Ing. Jiří Čulen

jiri.culen@mail.muni.cz

