

# Lecture 1

## Design of Embedded Hardware and Firmware

Setting up your system

*TSM\_EmbHardw  
November, 2015*

Prof. Dr. Theo Kluter  
Bern University of Applied Sciences

# Get your files from moodle

The screenshot shows a web browser window with the following details:

- Title Bar:** Course: T-EmbHardw - Design of Embedded Hardware and Firmware - Iceweasel
- Address Bar:** https://moodle.msengineering.ch/course/view.php?id=403
- Page Content:**
  - Section Header:** Topic 8
  - Text:** Week 8 (Updated 16-11-2015)
  - Text:** This week we will start with setting up the hardware/firmware for the system as demonstrated in the first week of the course. For this purpose 2 zip-files are attached including a small summary.
  - Text:** After the system is working for all students we are going to perform a Problem-Based-Learning session to try to find a collection of learning questions required to solve the problem at hand.
  - Attachments:**
    - Slides of this week's lecture (please use the handout version in case you want to have a printed copy)
    - Handout of today's slides for printing and to make notes
    - Hardware description file of the example system
    - Software required for the example system

- ▶ Login to moodle and go to topic 8



# Get your files from moodle

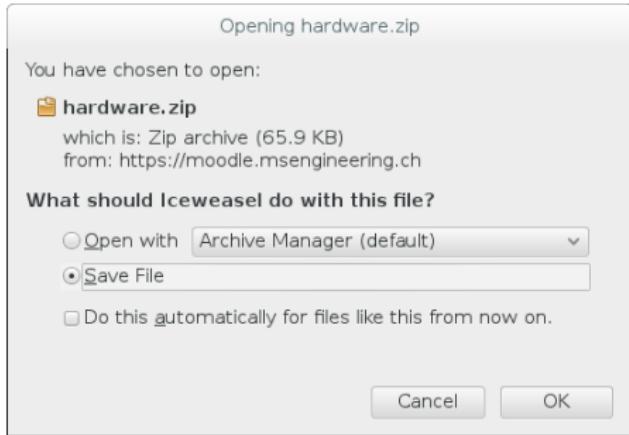
The screenshot shows a web browser window with the following details:

- Title Bar:** Course: T-EmbHardw - Design of Embedded Hardware and Firmware - Iceweasel
- Address Bar:** https://moodle.msengineering.ch/course/view.php?id=403
- Page Content:**
  - Section Header:** Topic 8
  - Text:** Week 8 (Updated 16-11-2015)
  - Text:** This week we will start with setting up the hardware/firmware for the system as demonstrated in the first week of the course. For this purpose 2 zip-files are attached including a small summary.
  - Text:** After the system is working for all students we are going to perform a Problem-Based-Learning session to try to find a collection of learning questions required to solve the problem at hand.
  - Attachments:**
    - Slides of this week's lecture (please use the handout version in case you want to have a printed copy)
    - Handout of today's slides for printing and to make notes
    - Hardware description file of the example system
    - Software required for the example system

- ▶ Login to moodle and go to topic 8
- ▶ Click on  
Hardware description file of the example system



# Get your files from moodle



- ▶ Login to moodle and go to topic 8
- ▶ Click on  
Hardware description file of the example system
- ▶ Make sure that you save the file

# Get your files from moodle

Course: T-EmbHardw - Design of Embedded Hardware and Firmware - Iceweasel

Course: T-EmbHar...

<https://moodle.msengineering.ch/course/view.php?id=403>  Search

Most Visited ▾ Getting Started

## Topic 8

Week 8 (Updated 16-11-2015)

This week we will start with setting up the hardware/firmware for the system as demonstrated in the first week of the course. For this purpose 2 zip-files are attached including a small summary.

After the system is working for all students we are going to perform a Problem-Based-Learning session to try to find a collection of learning questions required to solve the problem at hand.

Slides of this week's lecture (please use the handout version in case you want to have a printed copy)

Handout of today's slides for printing and to make notes

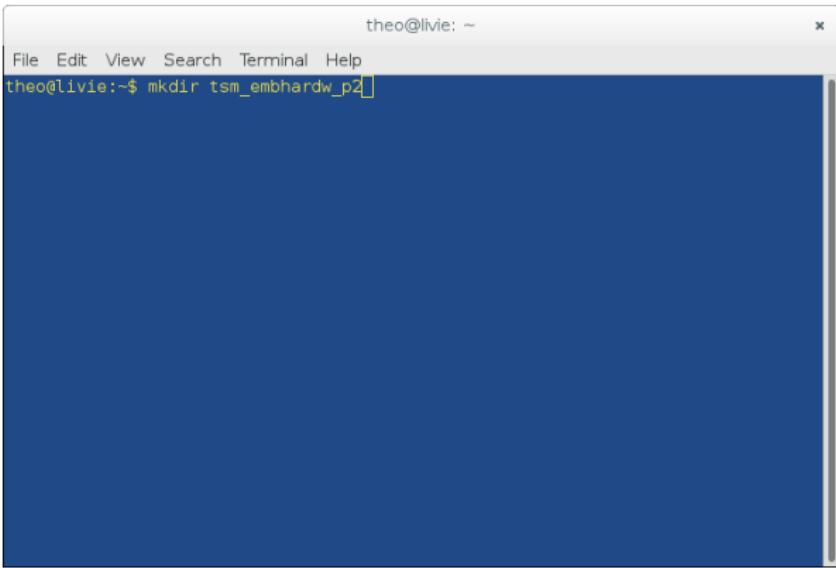
Hardware description file of the example system

Software required for the example system

- ▶ Login to moodle and go to topic 8
- ▶ Click on  
Hardware description file of the example system
- ▶ Make sure that you save the file
- ▶ do the same for  
Software required for the example system



# Setting up the directory



theo@livie: ~

File Edit View Search Terminal Help

```
theo@livie:~$ mkdir tsm_embardw_p2
```



Bern University  
of Applied Sciences

Setting up

Usage

- ▶ Open a terminal

# Setting up the directory

```
theo@livie: ~/tsm_embhardw_p2
File Edit View Search Terminal Help
theo@livie:~$ mkdir tsm_embhardw_p2
theo@livie:~$ cd tsm_embhardw_p2/
theo@livie:~/tsm_embhardw_p2$ mkdir software
```



Bern University  
of Applied Sciences

Setting up

Usage

- ▶ Open a terminal
- ▶ And perform all steps shown above

# Setting up the directory

```
theo@livie: ~/tsm_embhardw_p2
File Edit View Search Terminal Help
theo@livie:~$ mkdir tsm_embhardw_p2
theo@livie:~$ cd tsm_embhardw_p2/
theo@livie:~/tsm_embhardw_p2$ mkdir software
theo@livie:~/tsm_embhardw_p2$ unzip ~/Downloads/hardware.zip
```



- ▶ Open a terminal
- ▶ And perform all steps shown above

# Setting up the directory

```
theo@livie: ~/tsm_embhardw_p2
File Edit View Search Terminal Help
inflating: vhdl_modules/vga_controller/delay_line_behavior.vhdl
inflating: vhdl_modules/vga_controller/delay_line_entity.vhdl
inflating: vhdl_modules/vga_controller/vga_dma_entity.vhdl
inflating: vhdl_modules/vga_controller/vga_dma_behavior.vhdl
inflating: vhdl_modules/vga_controller/vga_dma_ctrl_behavior.vhdl
inflating: vhdl_modules/vga_controller/vga_entity.vhdl
inflating: vhdl_modules/vga_controller/vga_behavior.vhdl
inflating: vhdl_modules/vga_controller/vga_dma_ctrl_entity.vhdl
creating: vhdl_modules/camera_controller/
inflating: vhdl_modules/camera_controller/synchroflop_entity.vhdl
inflating: vhdl_modules/camera_controller/cam_dma_ctrl_entity.vhdl
inflating: vhdl_modules/camera_controller/cam_avalon_slave_entity.vhdl
inflating: vhdl_modules/camera_controller/cam_dma_behaviour.vhdl
inflating: vhdl_modules/camera_controller/cam_avalon_slave_behavior.vhdl
inflating: vhdl_modules/camera_controller/cam_dma_ctrl_behaviour.vhdl
inflating: vhdl_modules/camera_controller/synchroflop_behavior.vhdl
inflating: vhdl_modules/camera_controller/cam_dma_entity.vhdl
inflating: vhdl_modules/camera_controller/frame_interpreter_entity.vhdl
inflating: vhdl_modules/camera_controller/pixel_interface_entity.vhdl
inflating: vhdl_modules/camera_controller/pixel_interface_behavior.vhdl
inflating: vhdl_modules/camera_controller/frame_interpreter_behavior.vhdl
theo@livie:~/tsm_embhardw_p2$ ls
quartus_project  software  vhdl_modules
theo@livie:~/tsm_embhardw_p2$
```



- ▶ Open a terminal
- ▶ And perform all steps shown above

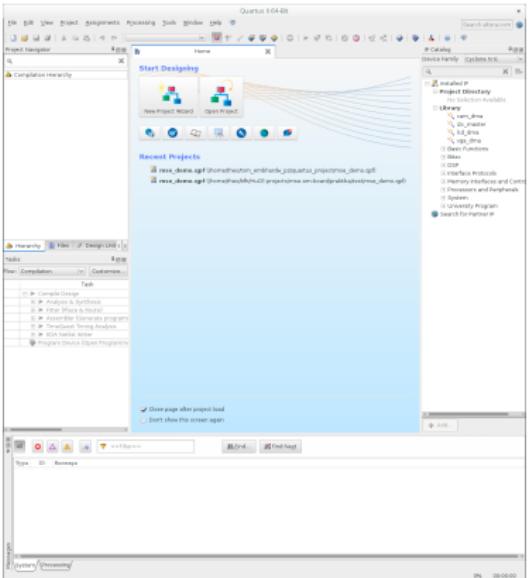
# Setting up the directory



```
theo@livie: ~/tsm_embhardw_p2/quartus_project
File Edit View Search Terminal Help
inflating: vhdl_modules/vga_controller/delay_line_entity.vhdl
inflating: vhdl_modules/vga_controller/vga_dma_entity.vhdl
inflating: vhdl_modules/vga_controller/vga_dma_behavior.vhdl
inflating: vhdl_modules/vga_controller/vga_dma_ctrl_behavior.vhdl
inflating: vhdl_modules/vga_controller/vga_entity.vhdl
inflating: vhdl_modules/vga_controller/vga_behavior.vhdl
inflating: vhdl_modules/vga_controller/vga_dma_ctrl_entity.vhdl
creating: vhdl_modules/camera_controller/
inflating: vhdl_modules/camera_controller/synchroflop_entity.vhdl
inflating: vhdl_modules/camera_controller/cam_dma_ctrl_entity.vhdl
inflating: vhdl_modules/camera_controller/cam_avalon_slave_entity.vhdl
inflating: vhdl_modules/camera_controller/cam_dma_behaviour.vhdl
inflating: vhdl_modules/camera_controller/cam_avalon_slave_behavior.vhdl
inflating: vhdl_modules/camera_controller/cam_dma_ctrl_behaviour.vhdl
inflating: vhdl_modules/camera_controller/synchroflop_behavior.vhdl
inflating: vhdl_modules/camera_controller/cam_dma_entity.vhdl
inflating: vhdl_modules/camera_controller/frame_interpreter_entity.vhdl
inflating: vhdl_modules/camera_controller/pixel_interface_entity.vhdl
inflating: vhdl_modules/camera_controller/pixel_interface_behavior.vhdl
inflating: vhdl_modules/camera_controller/frame_interpreter_behavior.vhdl
theo@livie:~/tsm_embhardw_p2$ ls
quartus_project software vhdl_modules
theo@livie:~/tsm_embhardw_p2$ cd quartus_project/
theo@livie:~/tsm_embhardw_p2/quartus_project$
```

- ▶ Open a terminal
- ▶ And perform all steps shown above

# Setting up quartus



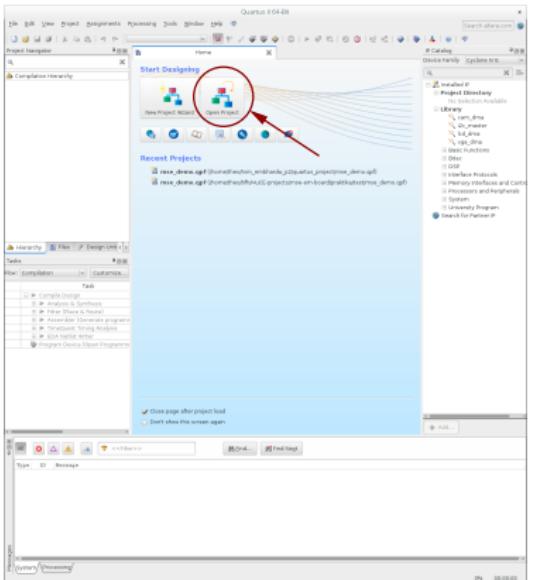
# Setting up quartus

Design of Embedded  
Hardware and  
Firmware

Prof. Dr. Theo Kluter



Bern University  
of Applied Sciences

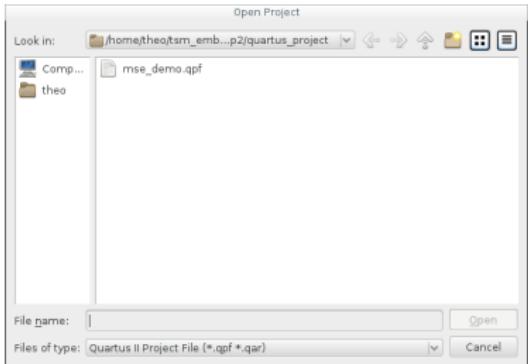


Setting up

Usage

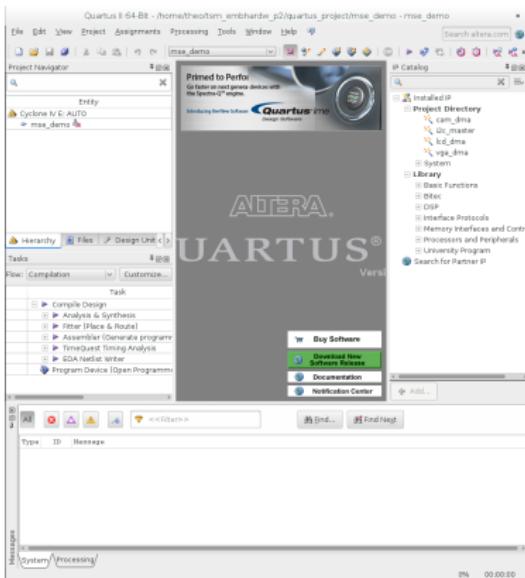
- ▶ Start quartus
- ▶ Open existing project

# Setting up quartus



- ▶ Start quartus
- ▶ Open existing project
- ▶ Select `mse_demo.qpf` and press Open

# Setting up quartus

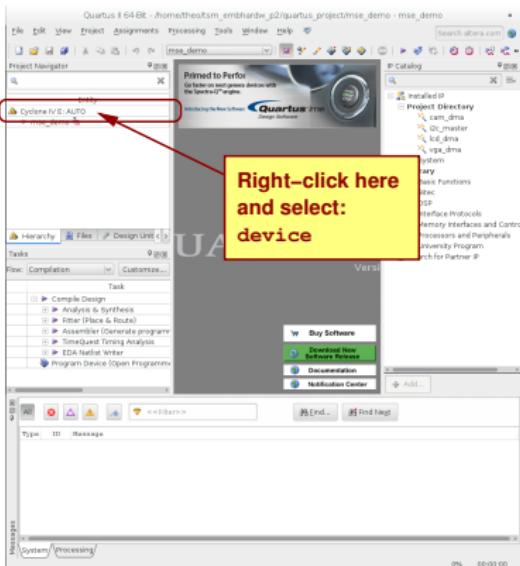


## Setting up

### Usage

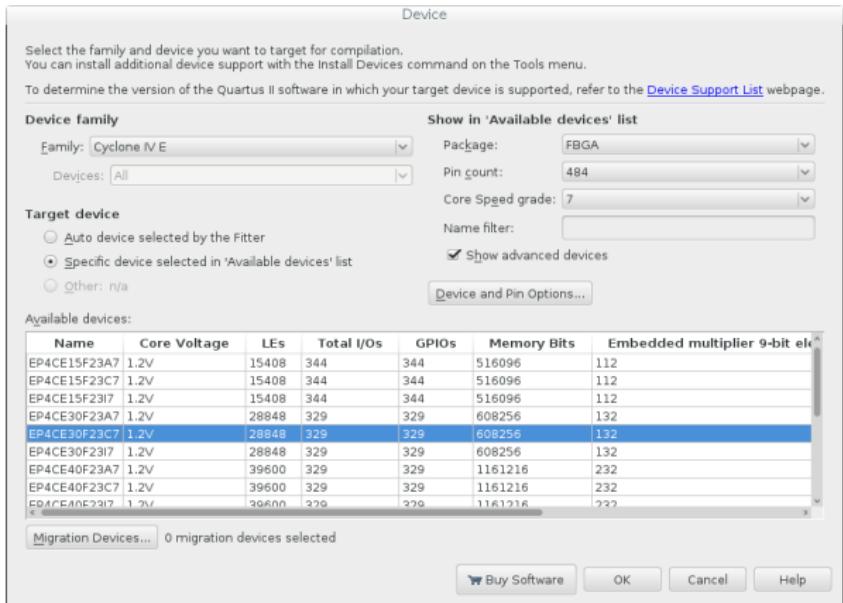
- ▶ Start quartus
- ▶ Open existing project
- ▶ Select `mse_demo.qpf` and press Open

# Assigning device



- We are going to assign the device

# Assigning device



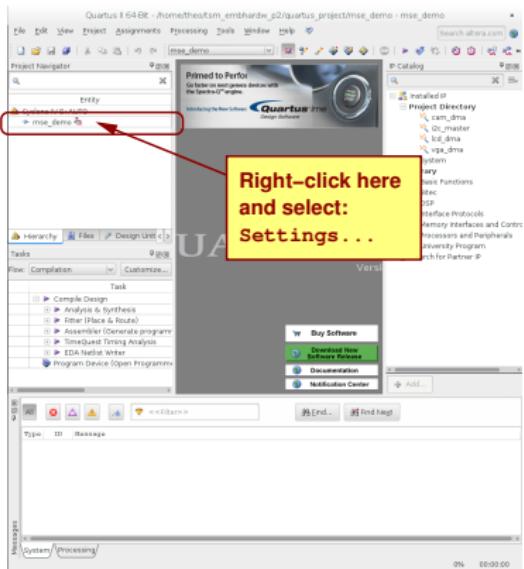
Bern University  
of Applied Sciences

Setting up

Usage

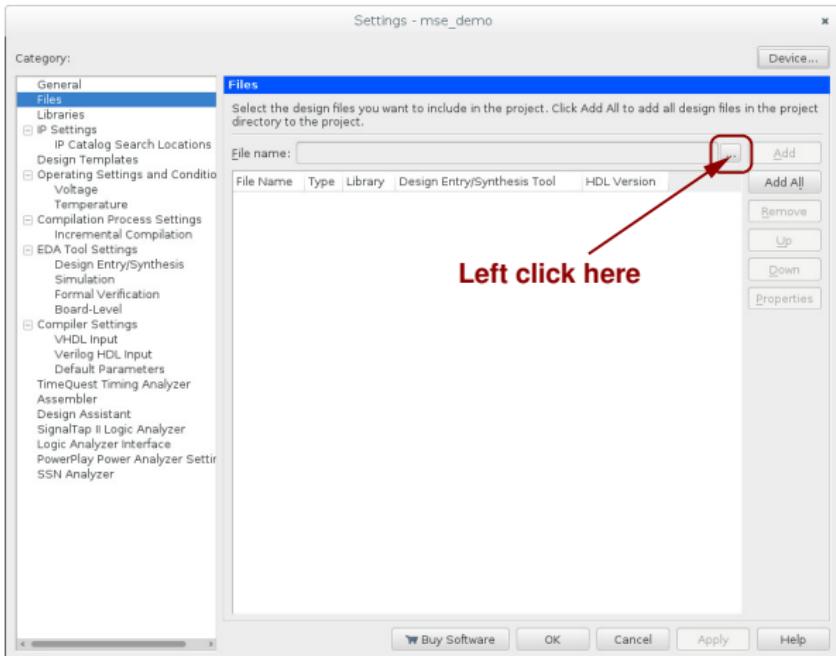
- We are going to assign the device
- Select the device EP4CE30F23C7 and click OK

# Adding qsys file



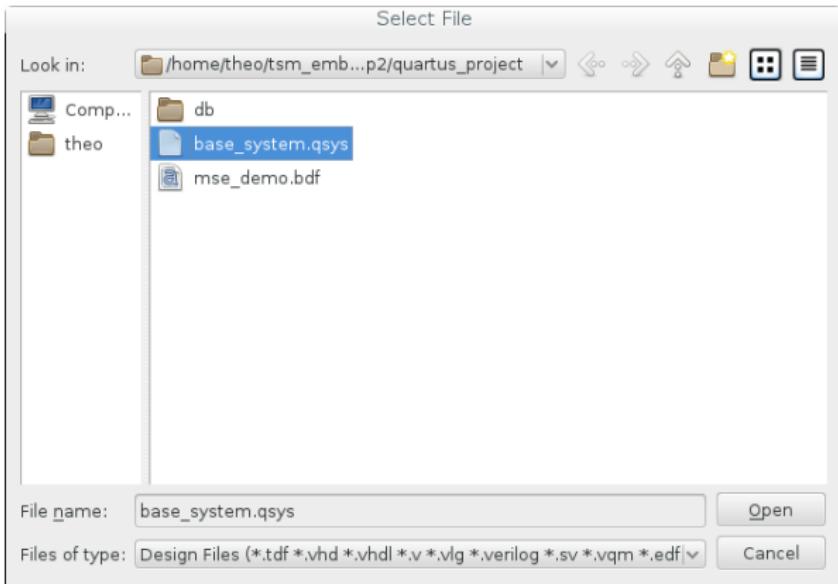
- We are going to add the qsys file

# Adding qsys file



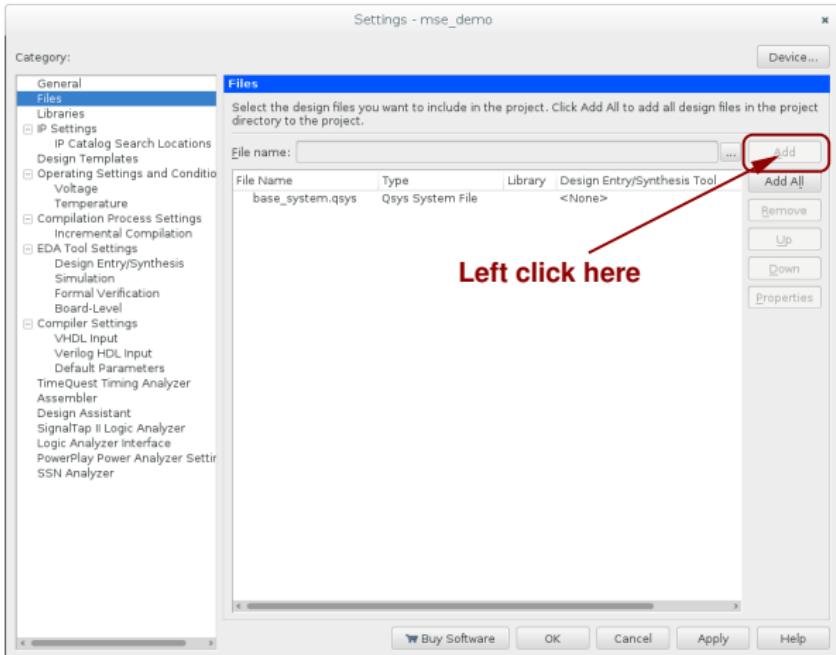
- We are going to add the qsys file
- This window will open; press the indicated button

# Adding qsys file



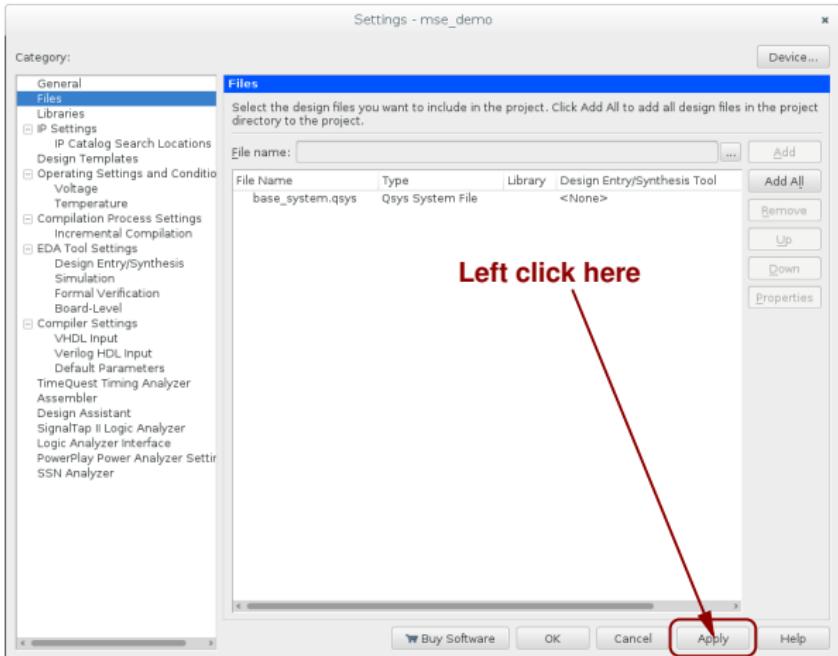
- ▶ We are going to add the qsys file
- ▶ This window will open; press the indicated button
- ▶ Select `base_system.qsys` and press Open

# Adding qsys file



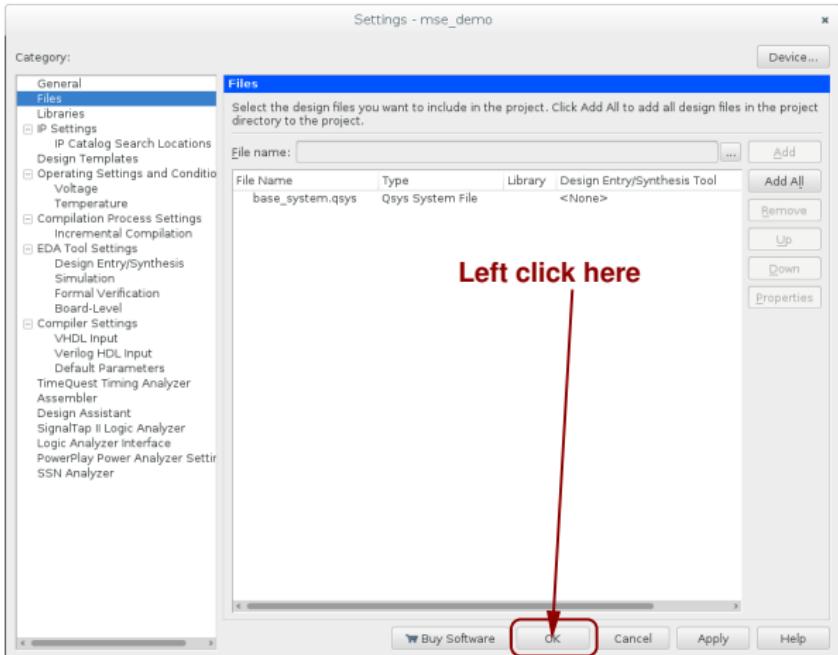
- We are going to add the qsys file
- This window will open; press the indicated button
- Select `base_system.qsys` and press Open
- Perform the indicated sequence of clicks

# Adding qsys file



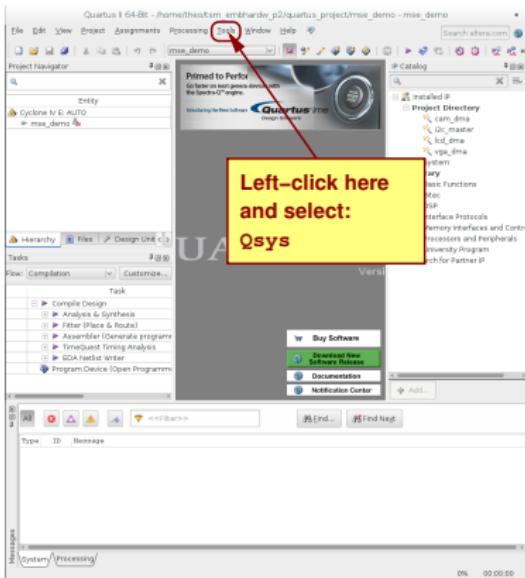
- We are going to add the qsys file
- This window will open; press the indicated button
- Select `base_system.qsys` and press Open
- Perform the indicated sequence of clicks

# Adding qsys file



- We are going to add the qsys file
- This window will open; press the indicated button
- Select `base_system.qsys` and press Open
- Perform the indicated sequence of clicks

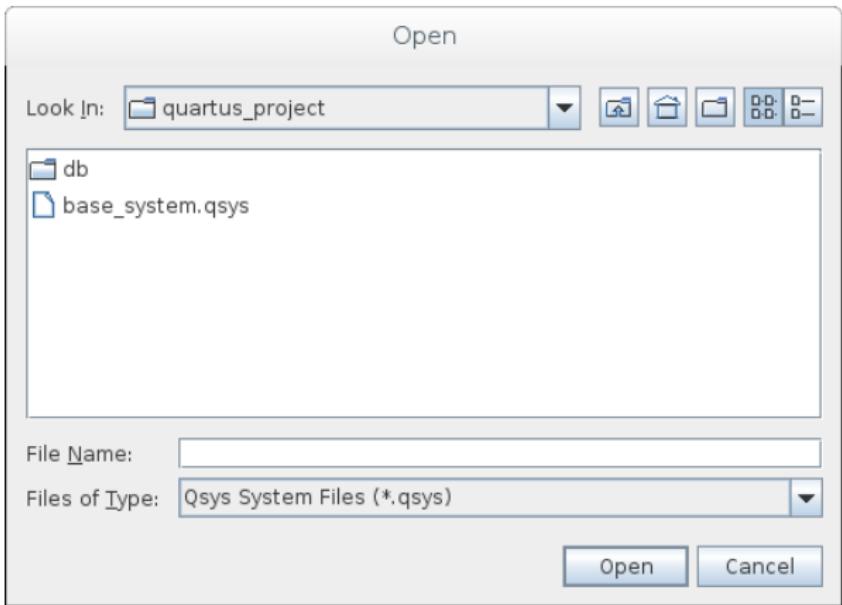
# Generating your system



► Start Qsys

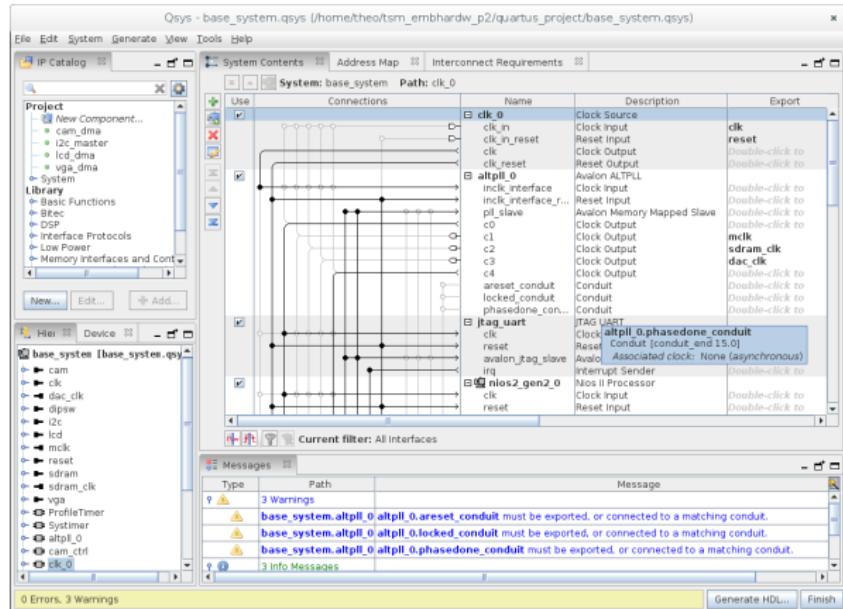


# Generating your system



- ▶ Start Qsys
- ▶ This window will open; Select `base_system.qsys` and press Open

# Generating your system

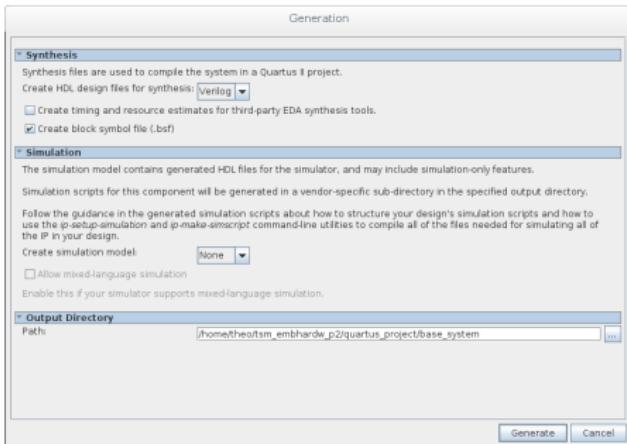


- ▶ Start Qsys
- ▶ This window will open; Select `base_system.qsys` and press Open
- ▶ Click on the button Generate HDL...

Setting up

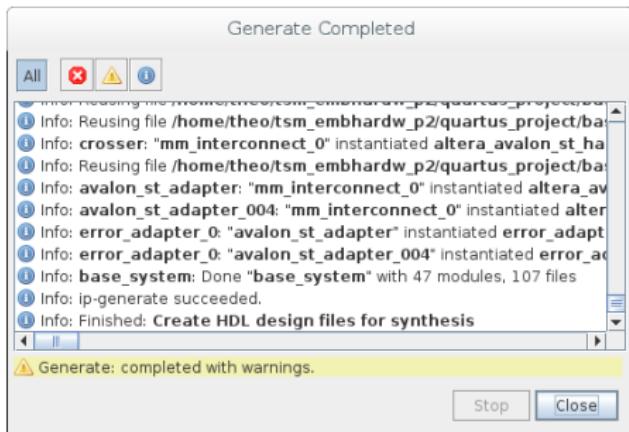
Usage

# Generating your system



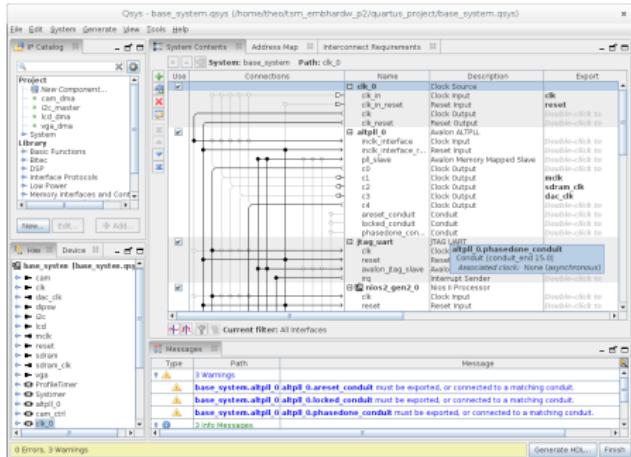
- ▶ Start Qsys
- ▶ This window will open; Select `base_system.qsys` and press Open
- ▶ Click on the button Generate HDL...
- ▶ This window will open; Click on the button Generate

# Generating your system



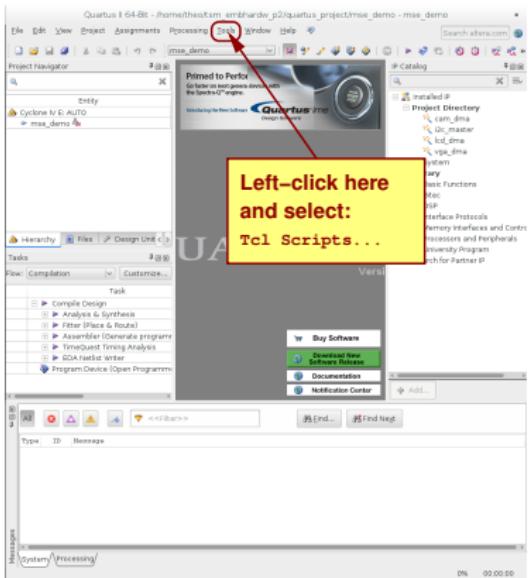
- ▶ Start Qsys
- ▶ This window will open; Select `base_system.qsys` and press Open
- ▶ Click on the button Generate HDL...
- ▶ This window will open; Click on the button Generate
- ▶ This window will open; Click on the button Close when the generation has finished (this may take a while)

# Generating your system



- ▶ Start Qsys
  - ▶ This window will open; Select `base_system.qsys` and press Open
  - ▶ Click on the button Generate HDL...
  - ▶ This window will open; Click on the button Generate
  - ▶ This window will open; Click on the button Close when the generation has finished (this may take a while)
  - ▶ And click on the button Finish

# Assigning the pins

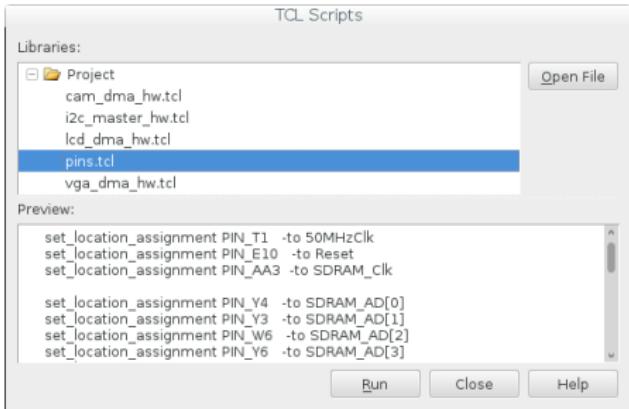


Setting up

Usage

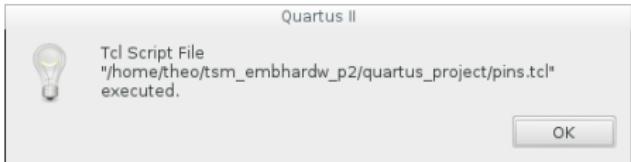
- ▶ Start TCL-scripting

# Assigning the pins



- ▶ Start TCL-scripting
- ▶ This window will open; Select `pins.tcl` and press Run

# Assigning the pins



- ▶ Start TCL-scripting
- ▶ This window will open; Select `pins.tcl` and press Run
- ▶ This window will open; press OK

# Synthesis and P&R

Design of Embedded  
Hardware and  
Firmware

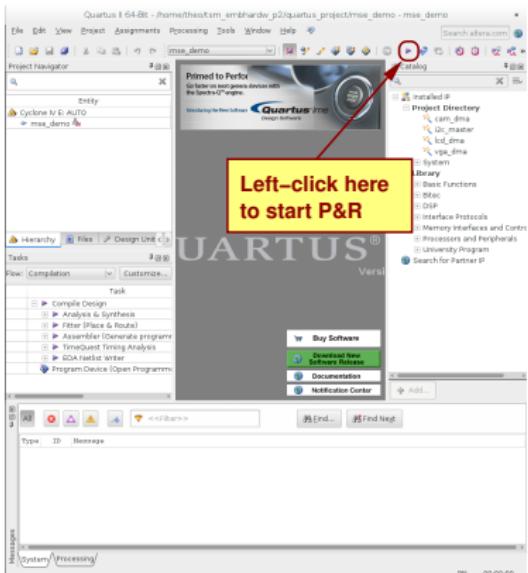
Prof. Dr. Theo Kluter



Bern University  
of Applied Sciences

Setting up

Usage



- ▶ Start synthesis and place-and route.
- ▶ After a while Quartus is ready and the window should look like this.

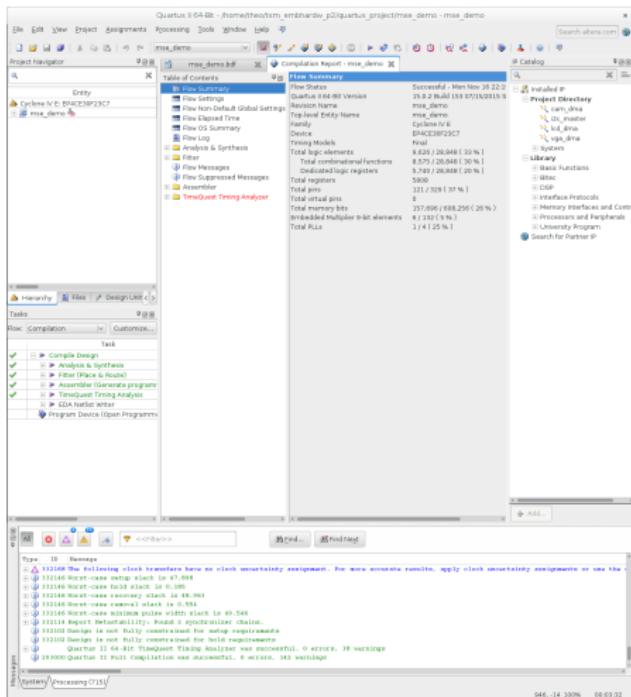
# Synthesis and P&R

Design of Embedded  
Hardware and  
Firmware

Prof. Dr. Theo Kluter



Bern University  
of Applied Sciences



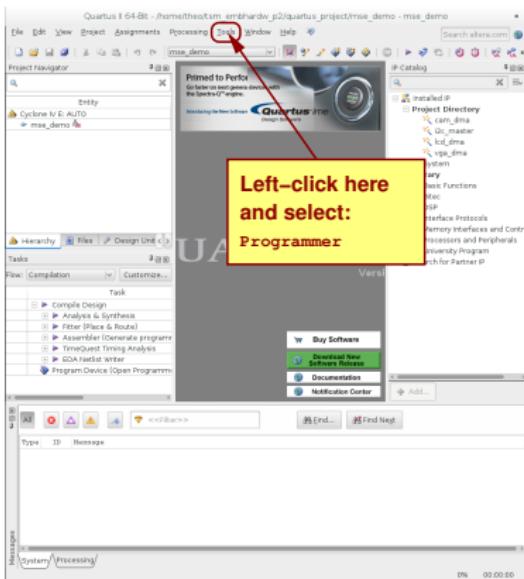
- ▶ Start synthesis and place-and route.
- ▶ After a while Quartus is ready and the window should look like this.

# Downloading



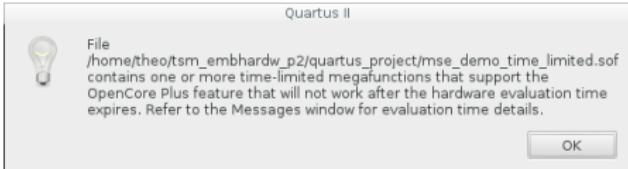
## Setting up

### Usage



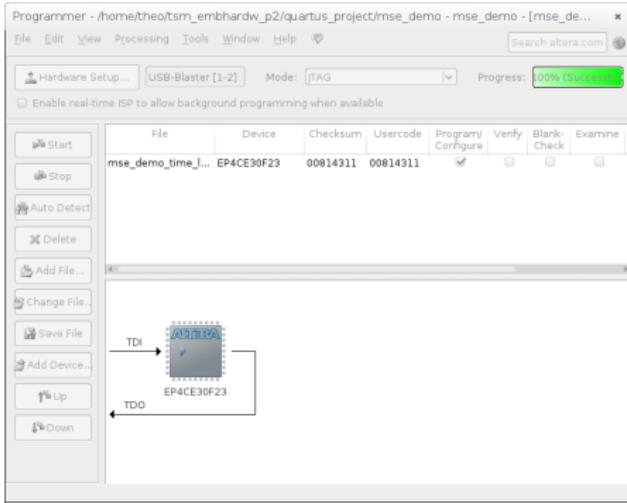
- ▶ Start the download tool (make sure your board is connected to the laptop!).

# Downloading

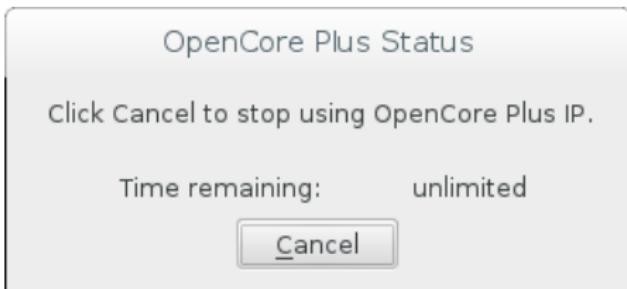


- ▶ Start the download tool (make sure your board is connected to the laptop!).
- ▶ A window similar like this one will pop-up. Click on OK.

# Downloading



- ▶ Start the download tool (make sure your board is connected to the laptop!).
- ▶ A window similar like this one will pop-up. Click on OK.
- ▶ The programmer window pops up. Make sure the hardware is set up and press the Start button.



- ▶ Start the download tool (make sure your board is connected to the laptop!).
- ▶ A window similar like this one will pop-up. Click on OK.
- ▶ The programmer window pops up. Make sure the hardware is set up and press the Start button.
- ▶ After the programming completed, this window will pop-up.  
**IMPORTANT:** Do not close this window!



Bern University  
of Applied Sciences

Setting up

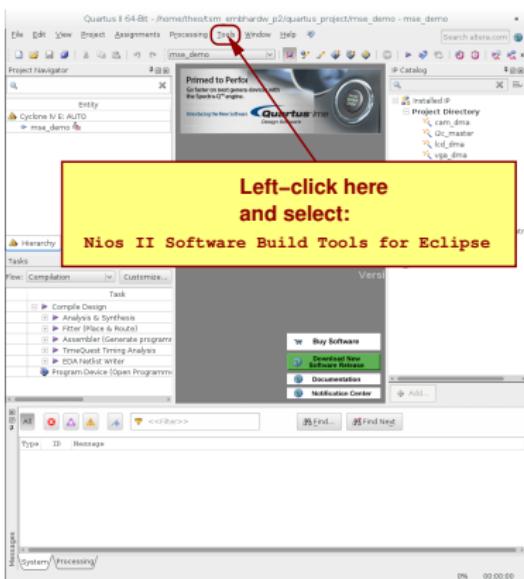
Usage

# Setting-up software

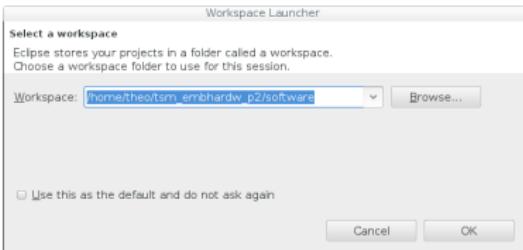


## Setting up

### Usage



- ▶ Start eclipse.

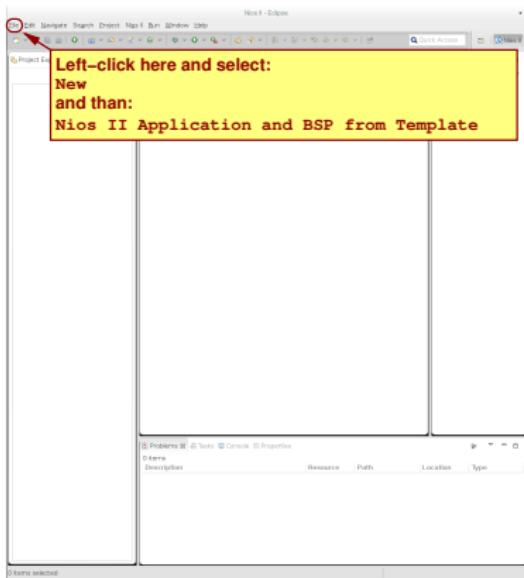


Setting up

Usage

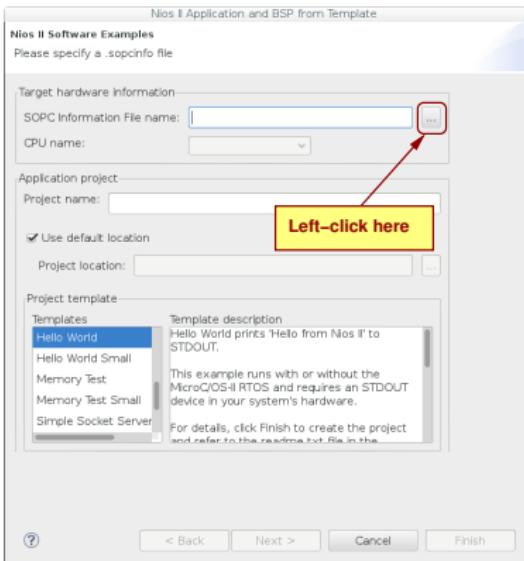
- ▶ Start eclipse.
- ▶ A window like this one will pop-up. Select as workspace the directory `~/tsm_embhardw_p2/software`.

# Setting-up software



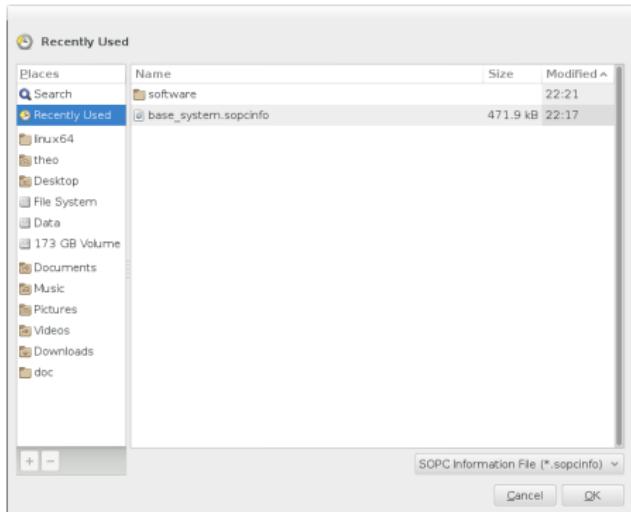
- ▶ Start eclipse.
- ▶ A window like this one will pop-up. Select as workspace the directory `~/tsm_embhardw_p2/software`.
- ▶ Create a new project.

# Setting-up software



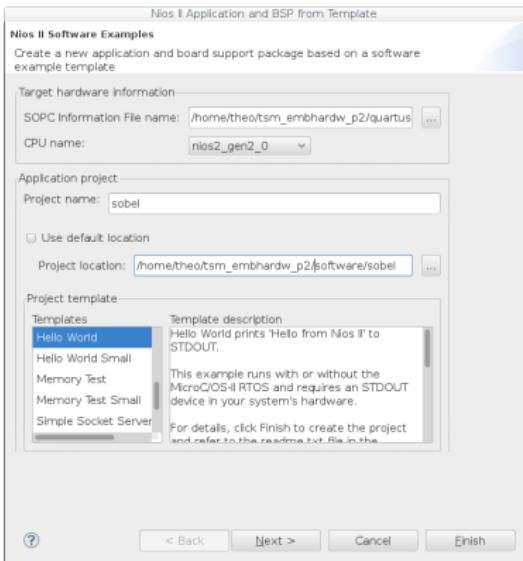
- A window like this one pops up, click on . . .

# Setting-up software



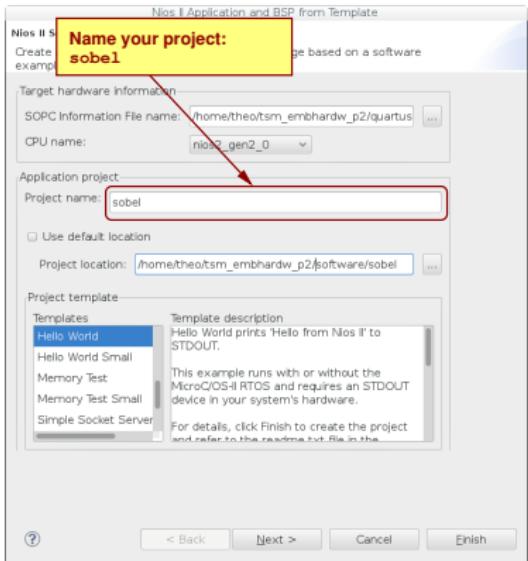
- ▶ A window like this one pops up, click on . . .
- ▶ This window pops up, go to the directory  
`~/tsm_embhw_p2/`, select `base_system.sopcinfo`,  
and click on OK.

# Setting-up software



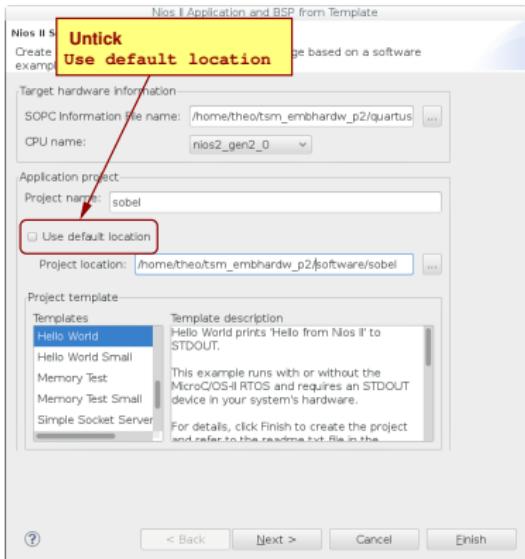
- ▶ A window like this one pops up, click on . . .
- ▶ This window pops up, go to the directory  
~/tsm\_embhardw\_p2/, select `base_system.sopcinfo`,  
and click on OK.
- ▶ This will return to this window.

# Configuring project



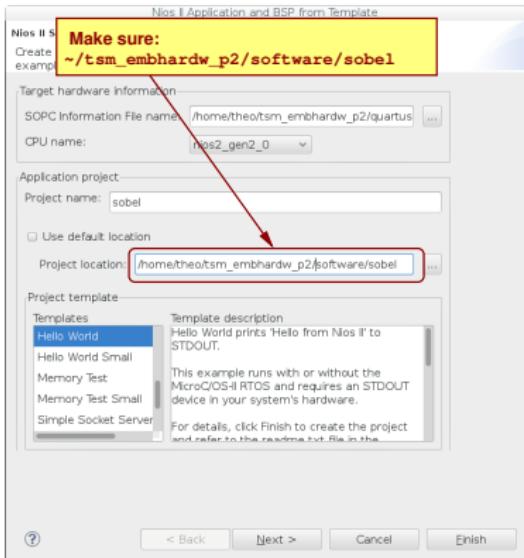
- Name your project sobel

# Configuring project



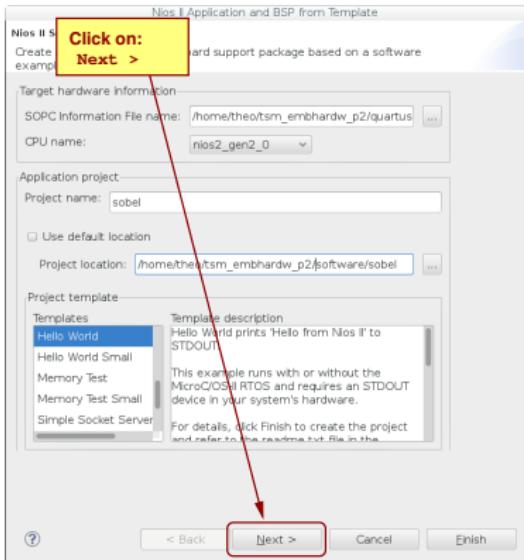
- Name your project sobel
- Untick Use default location

# Configuring project



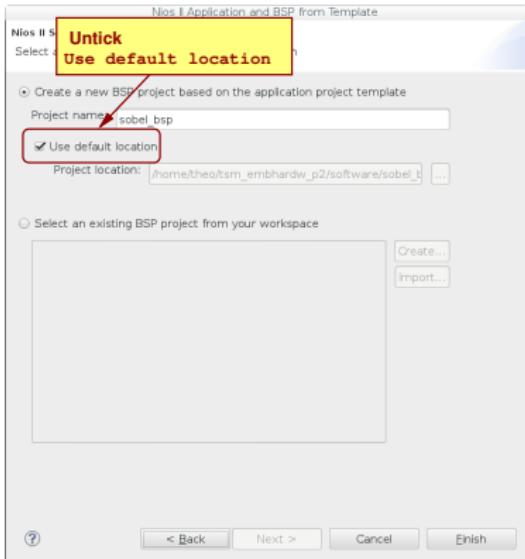
- Name your project sobel
- Untick Use default location
- Make sure that your Project location is  
~/tsm\_embhardw\_p2/software/sobel

# Configuring project



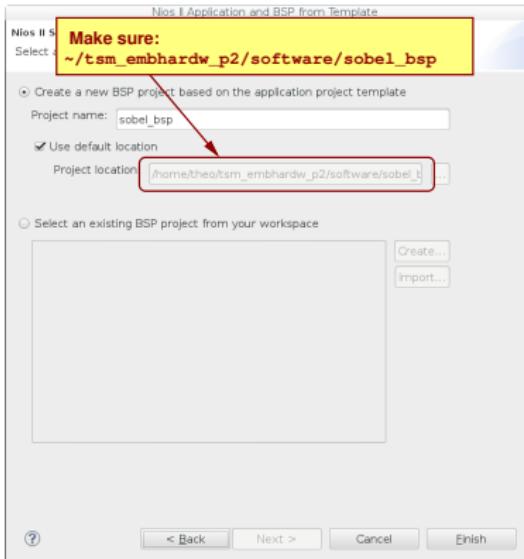
- Name your project sobel
- Untick Use default location
- Make sure that your Project location is  
~/tsm\_embhardw\_p2/software/sobel
- Click on Next

# Configuring project



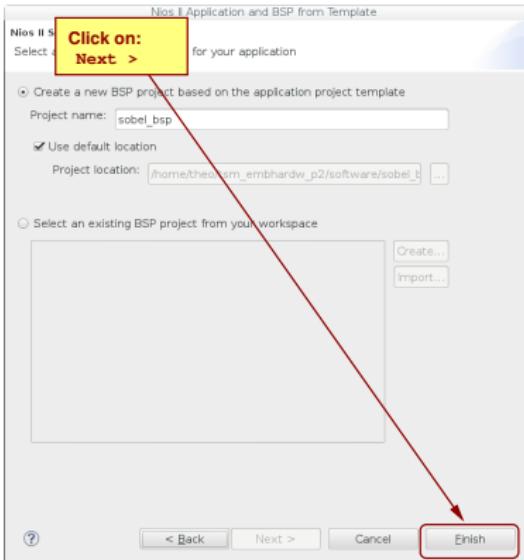
► Untick Use default location

# Configuring project



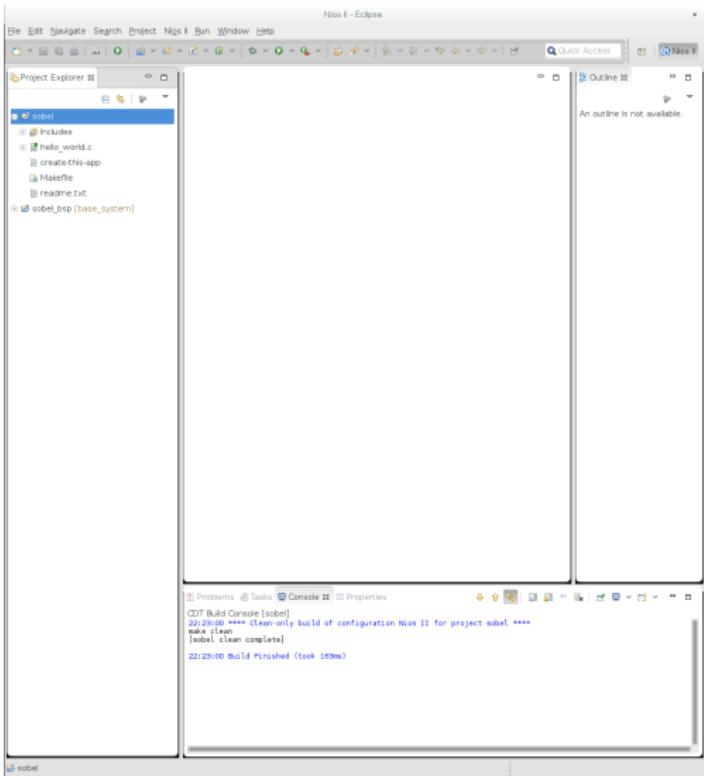
- ▶ Untick Use default location
- ▶ Make sure that your Project location is  
`~/tsm_embhardw_p2/software/sobel_bsp`

# Configuring project



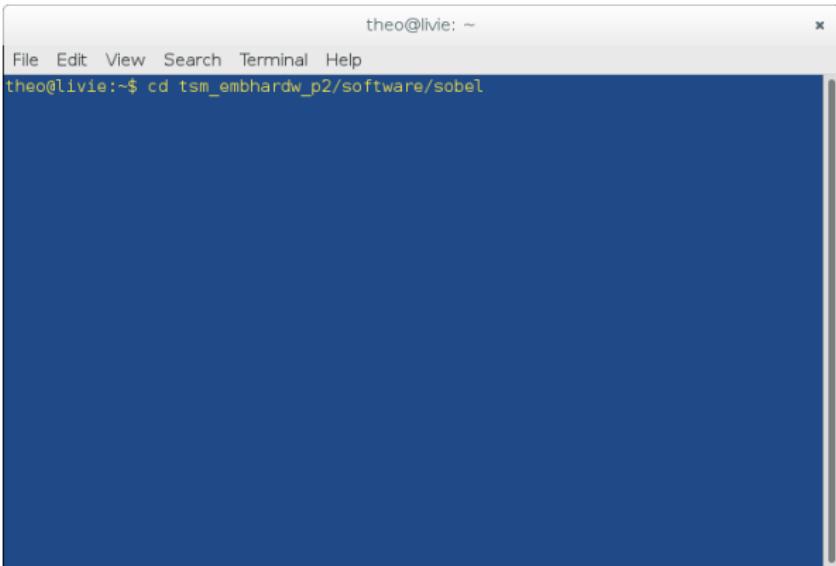
- Untick Use default location
- Make sure that your Project location is  
~/tsm\_embhardw\_p2/software/sobel\_bsp
- Click on Finish

# Configuring project



- Your eclipse window should look after a small moment like this one (in case you click on the + next to sobel)

# Replacing project



A screenshot of a terminal window titled "theo@livie: ~". The window has a standard OS X-style title bar with icons for close, minimize, and zoom. The menu bar at the top includes "File", "Edit", "View", "Search", "Terminal", and "Help". Below the menu bar, the prompt "theo@livie:~\$ cd tsm\_embhardw\_p2/software/sobel" is visible. The main body of the terminal is a large blue rectangular area, likely a placeholder for the terminal output.



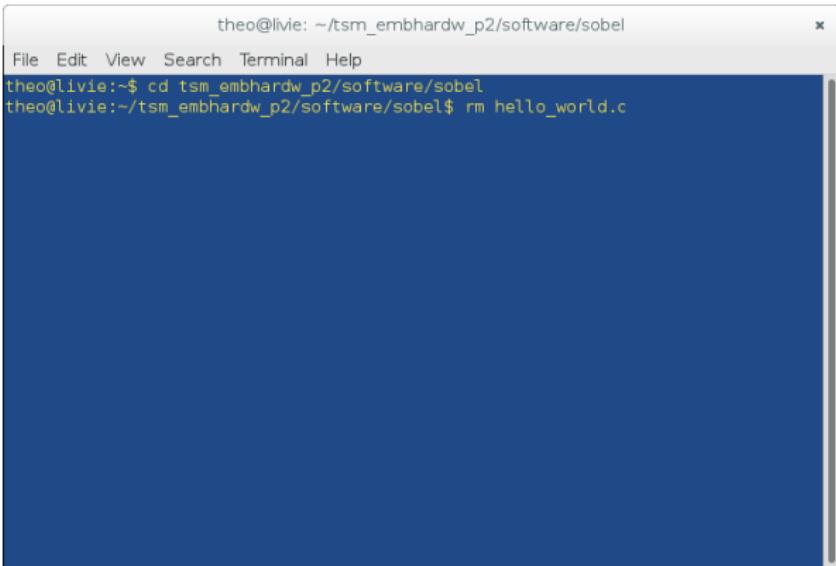
Bern University  
of Applied Sciences

Setting up

Usage

- ▶ Open a terminal and follow the above shown steps

# Replacing project



A screenshot of a terminal window titled "theo@livie: ~/tsm\_embhardw\_p2/software/sobel". The window has a standard OS X-style title bar with icons for close, minimize, and zoom. The main area of the terminal shows the following command history:

```
theo@livie:~/tsm_embhardw_p2/software/sobel
File Edit View Search Terminal Help
theo@livie:~$ cd tsm_embhardw_p2/software/sobel
theo@livie:~/tsm_embhardw_p2/software/sobel$ rm hello_world.c
```



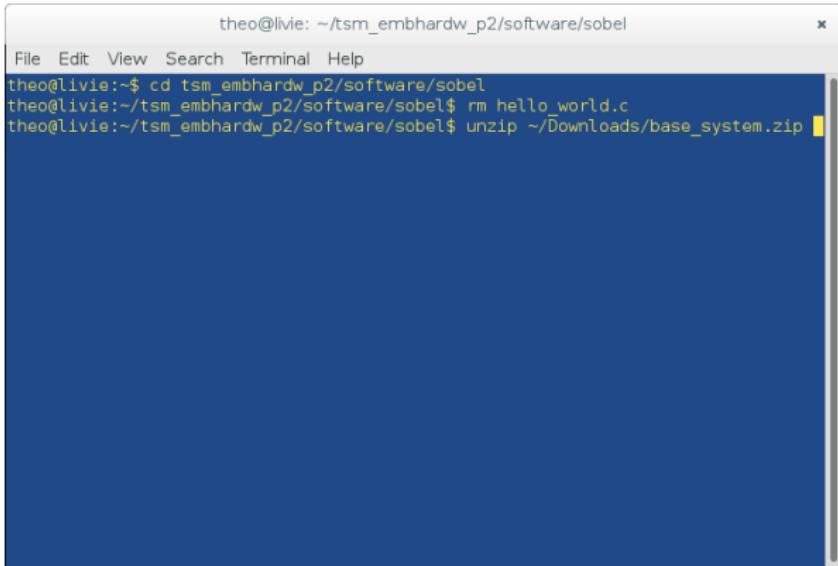
Bern University  
of Applied Sciences

Setting up

Usage

- ▶ Open a terminal and follow the above shown steps

# Replacing project



A terminal window titled "theo@livie: ~/tsm\_embhardw\_p2/software/sobel". The window shows the following command history:

```
theo@livie:~/tsm_embhardw_p2/software/sobel
File Edit View Search Terminal Help
theo@livie:~$ cd tsm_embhardw_p2/software/sobel
theo@livie:~/tsm_embhardw_p2/software/sobel$ rm hello_world.c
theo@livie:~/tsm_embhardw_p2/software/sobel$ unzip ~/Downloads/base_system.zip
```



Bern University  
of Applied Sciences

Setting up

Usage

- ▶ Open a terminal and follow the above shown steps

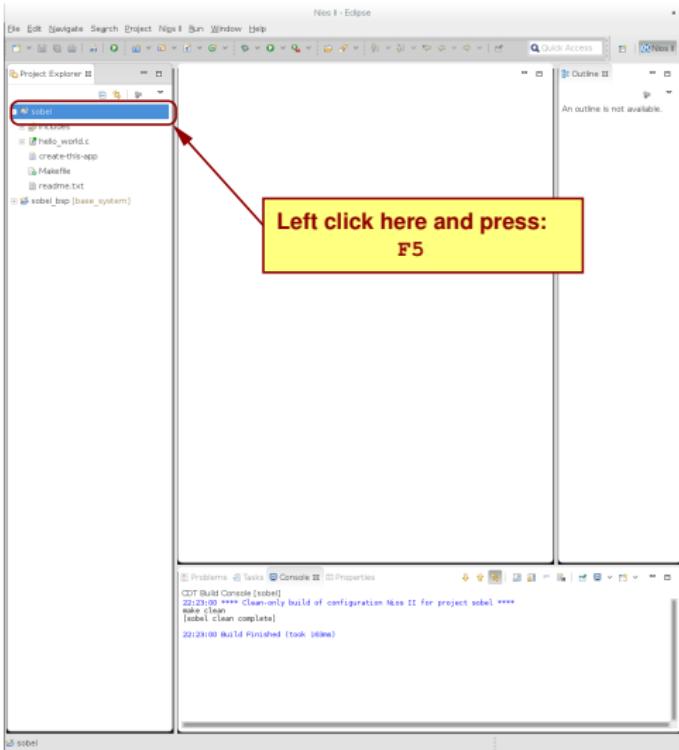
# Replacing project

```
theo@livie: ~/tsm_embhardw_p2/software/sobel
File Edit View Search Terminal Help
theo@livie:~$ cd tsm_embhardw_p2/software/sobel
theo@livie:~/tsm_embhardw_p2/software/sobel$ rm hello_world.c
theo@livie:~/tsm_embhardw_p2/software/sobel$ unzip ~/Downloads/base_system.zip
Archive: /home/theo/Downloads/base_system.zip
  inflating: camera.c
  inflating: dipswitch.c
  inflating: grayscale.c
  inflating: i2c.c
  inflating: lcd_simple.c
  inflating: main.c
  inflating: sobel.c
  inflating: vga.c
  inflating: camera.h
  inflating: dipswitch.h
  inflating: grayscale.h
  inflating: i2c.h
  inflating: lcd_simple.h
  inflating: mt9d112.h
  inflating: sobel.h
  inflating: vga.h
theo@livie:~/tsm_embhardw_p2/software/sobel$ █
```



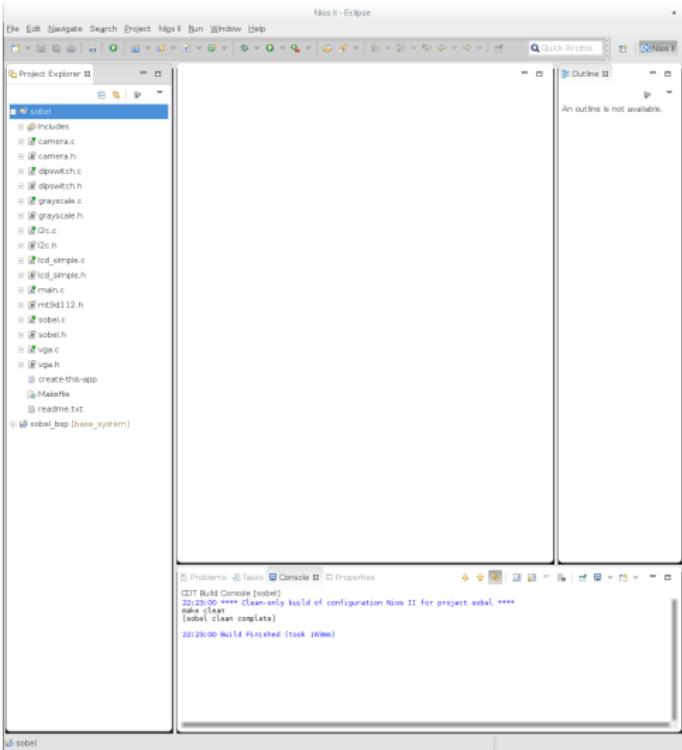
- ▶ Open a terminal and follow the above shown steps
- ▶ Close the terminal and go back to eclipse

# Replacing project



- ▶ Click on `sobel` and press the button F5

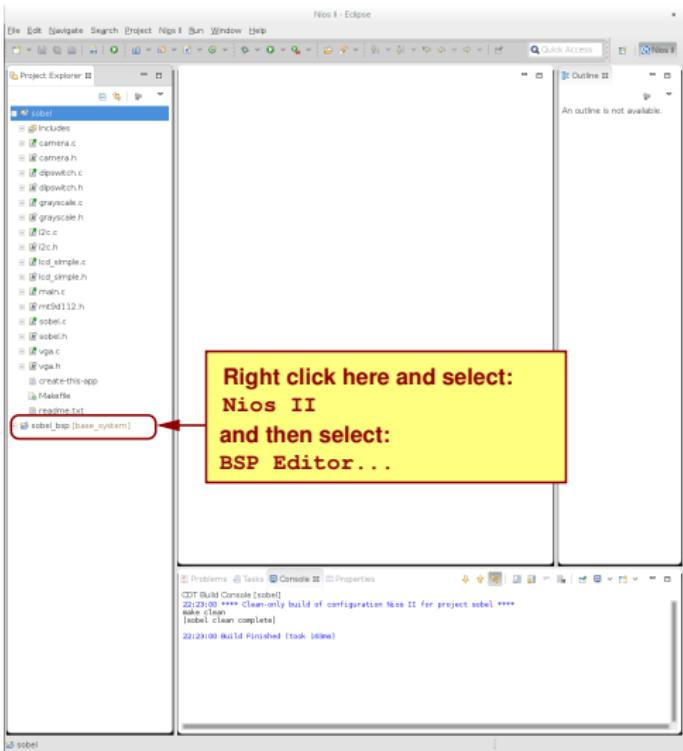
# Replacing project



- ▶ Click on `sobel` and press the button F5
- ▶ The project should now look like this



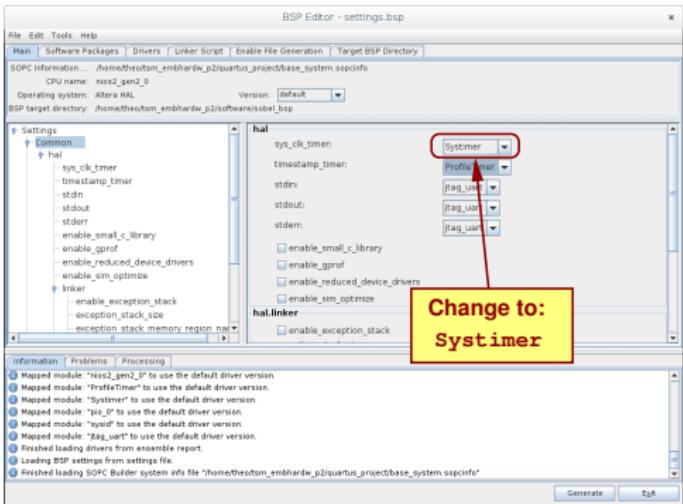
# Configuring project



- ▶ Open the BSP Editor

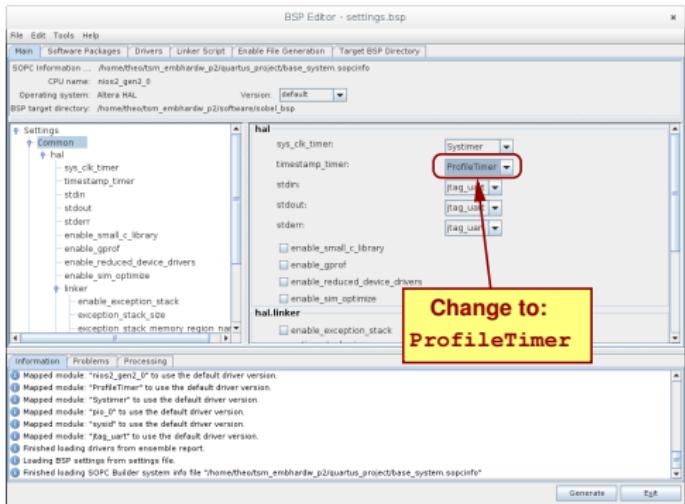


# Configuring project



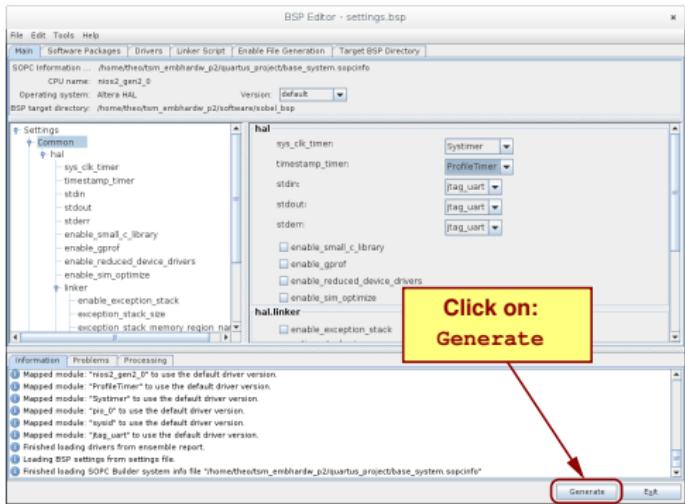
- ▶ Open the BSP Editor
- ▶ The BSP-Editor pops up, change `sys_clk_timer`: to Systimer

# Configuring project



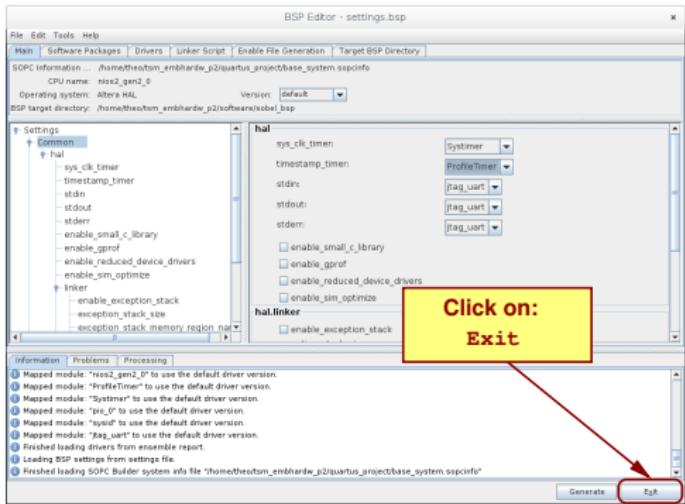
- ▶ Open the BSP Editor
- ▶ The BSP-Editor pops up, change `sys_clk_timer`: to Systimer
- ▶ and `timestamp_timer`: to ProfileTimer

# Configuring project



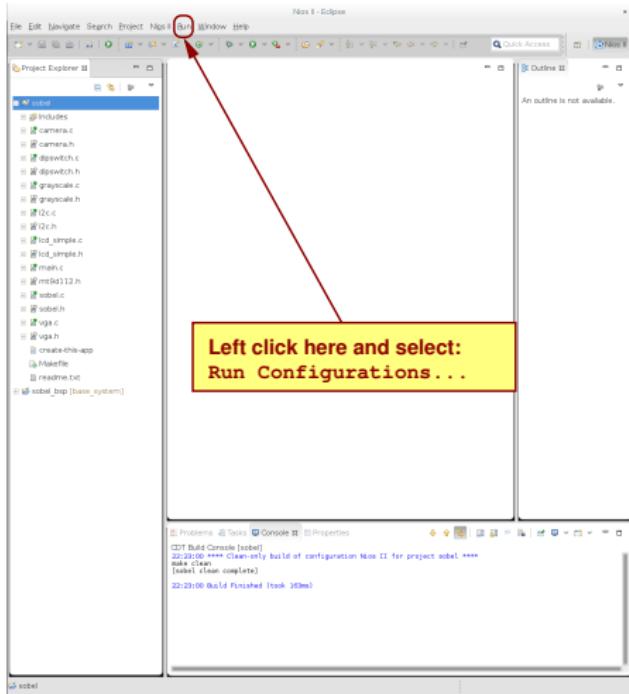
- ▶ Open the BSP Editor
- ▶ The BSP-Editor pops up, change `sys_clk_timer`: to Systimer
- ▶ and `timestamp_timer`: to ProfileTimer
- ▶ Click Generate

# Configuring project



- Open the BSP Editor
- The BSP-Editor pops up, change `sys_clk_timer`: to Systimer
- and `timestamp_timer`: to ProfileTimer
- Click Generate
- Click Exit

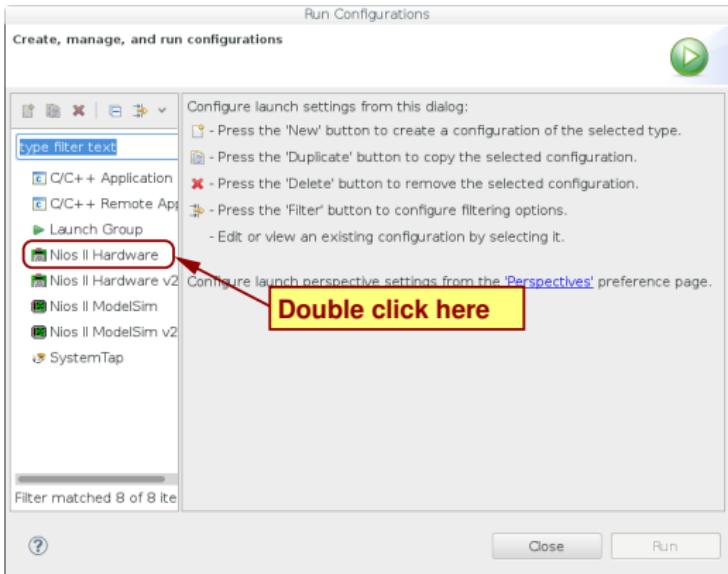
# Configuring project



- ▶ Open the Run configuration Editor

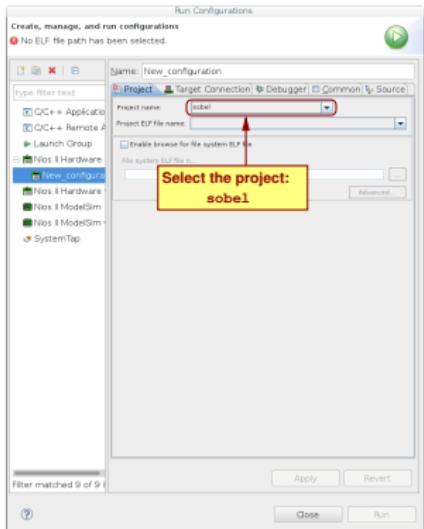


# Configuring project



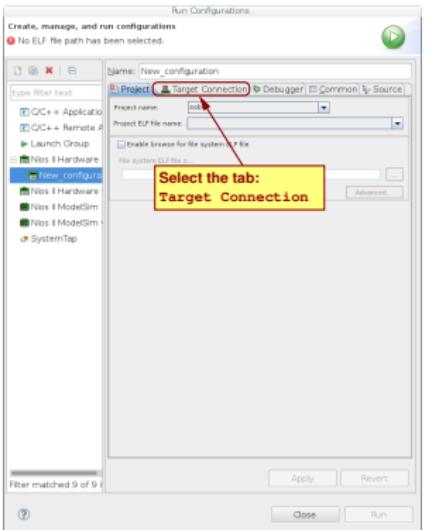
- ▶ Open the Run configuration Editor
- ▶ The RC-Editor pops up, double click on  
Nios II Hardware

# Configuring project



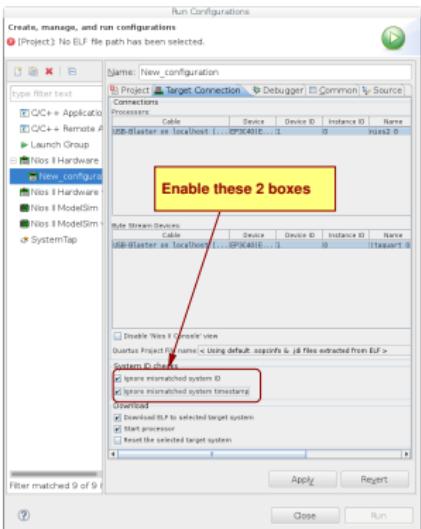
- ▶ Open the Run configuration Editor
- ▶ The RC-Editor pops up, double click on Nios II Hardware
- ▶ Select the project sobel

# Configuring project



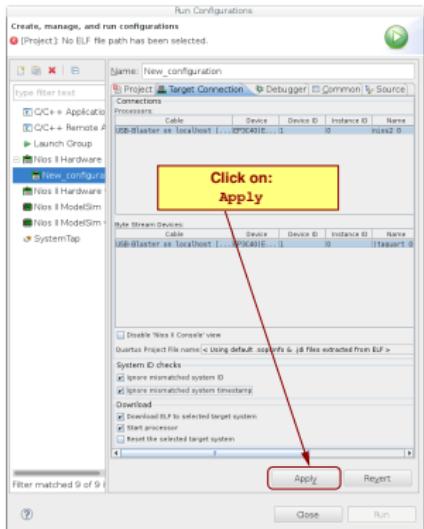
- ▶ Open the Run configuration Editor
- ▶ The RC-Editor pops up, double click on Nios II Hardware
- ▶ Select the project sobel
- ▶ Goto the tab Target Connection

# Configuring project



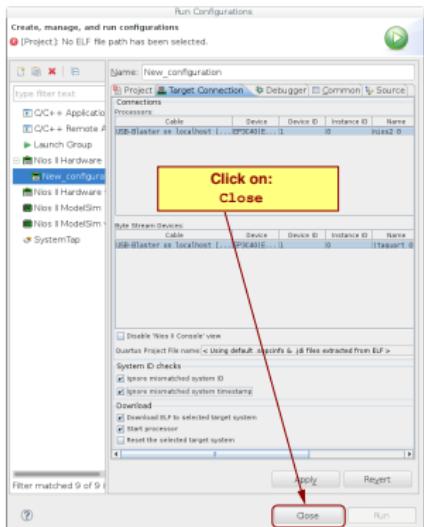
- ▶ Open the Run configuration Editor
- ▶ The RC-Editor pops up, double click on Nios II Hardware
- ▶ Select the project sobel
- ▶ Goto the tab Target Connection
- ▶ Tick the two shown boxes

# Configuring project



- ▶ Open the Run configuration Editor
- ▶ The RC-Editor pops up, double click on Nios II Hardware
- ▶ Select the project sobel
- ▶ Goto the tab Target Connection
- ▶ Tick the two shown boxes
- ▶ Click on Apply and than Close

# Configuring project



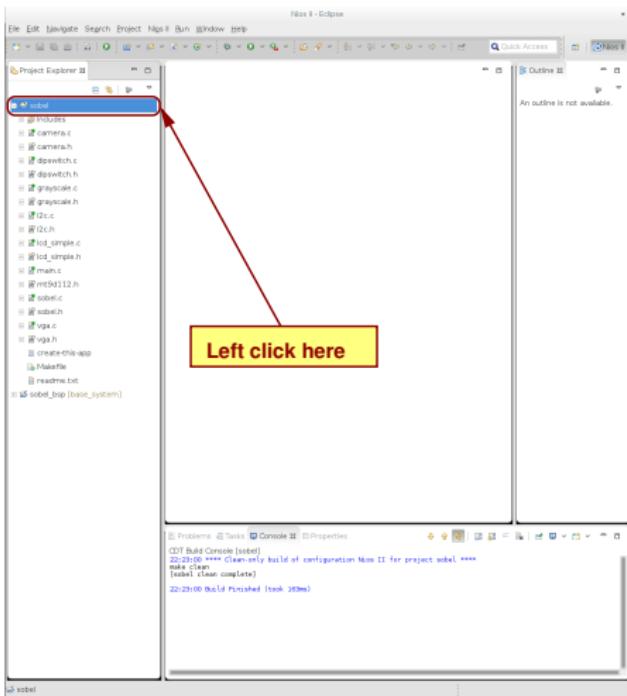
- ▶ Open the Run configuration Editor
- ▶ The RC-Editor pops up, double click on Nios II Hardware
- ▶ Select the project sobel
- ▶ Goto the tab Target Connection
- ▶ Tick the two shown boxes
- ▶ Click on Apply and than Close

# Running



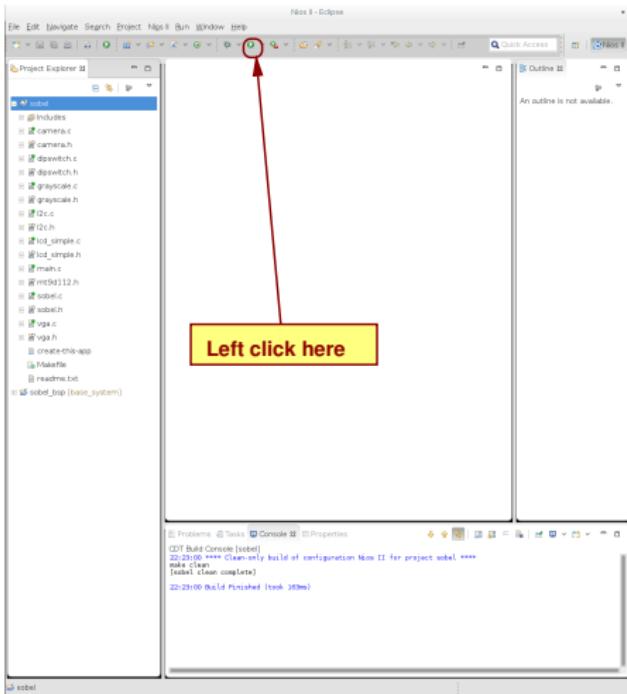
Setting up

Usage



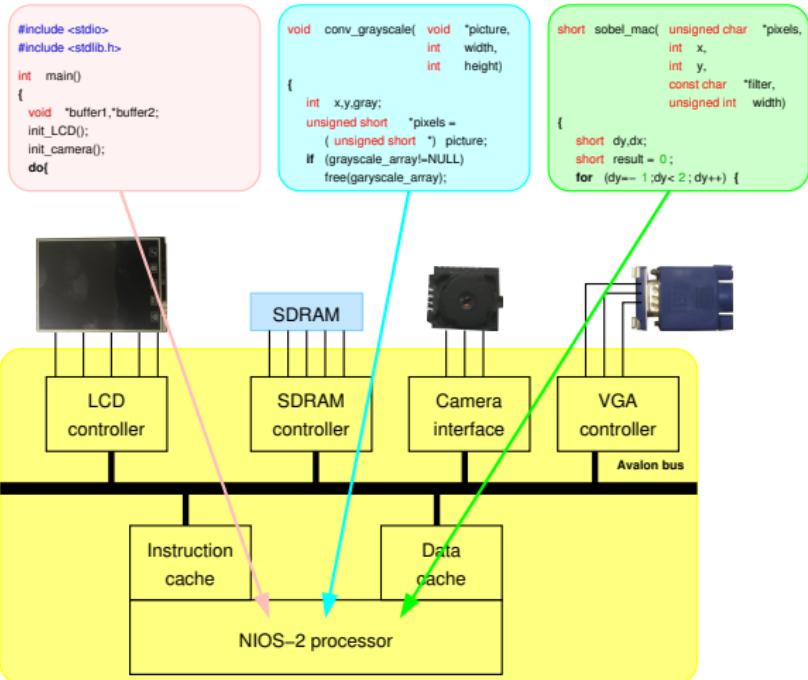
- Left click on sobel

# Running



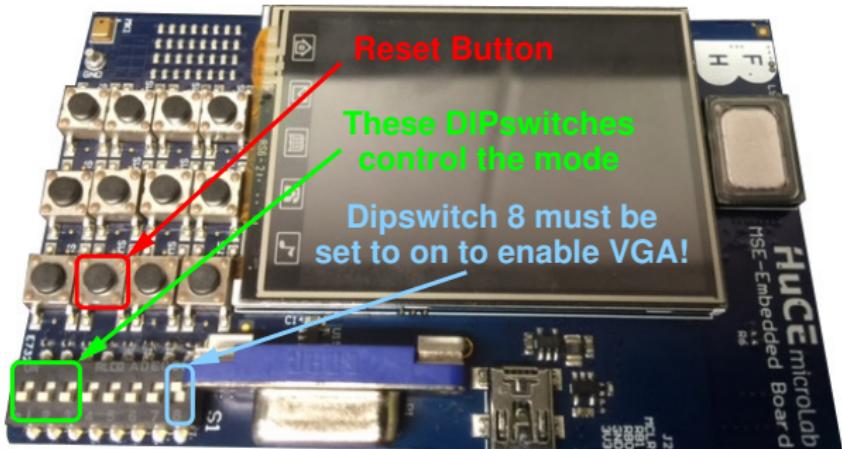
- Left click on sobel
- Press the Run button

# Running



- ▶ Left click on sobel
- ▶ Press the Run button
- ▶ Wenn all went well your system is running

# Usage



SW1	SW2	SW3	Function
off	off	off	Direct camera -> LCD
on	off	off	Grayscale
off	on	off	Sobel-X
on	on	off	Sobel-Y
X	X	on	Sobel-complete