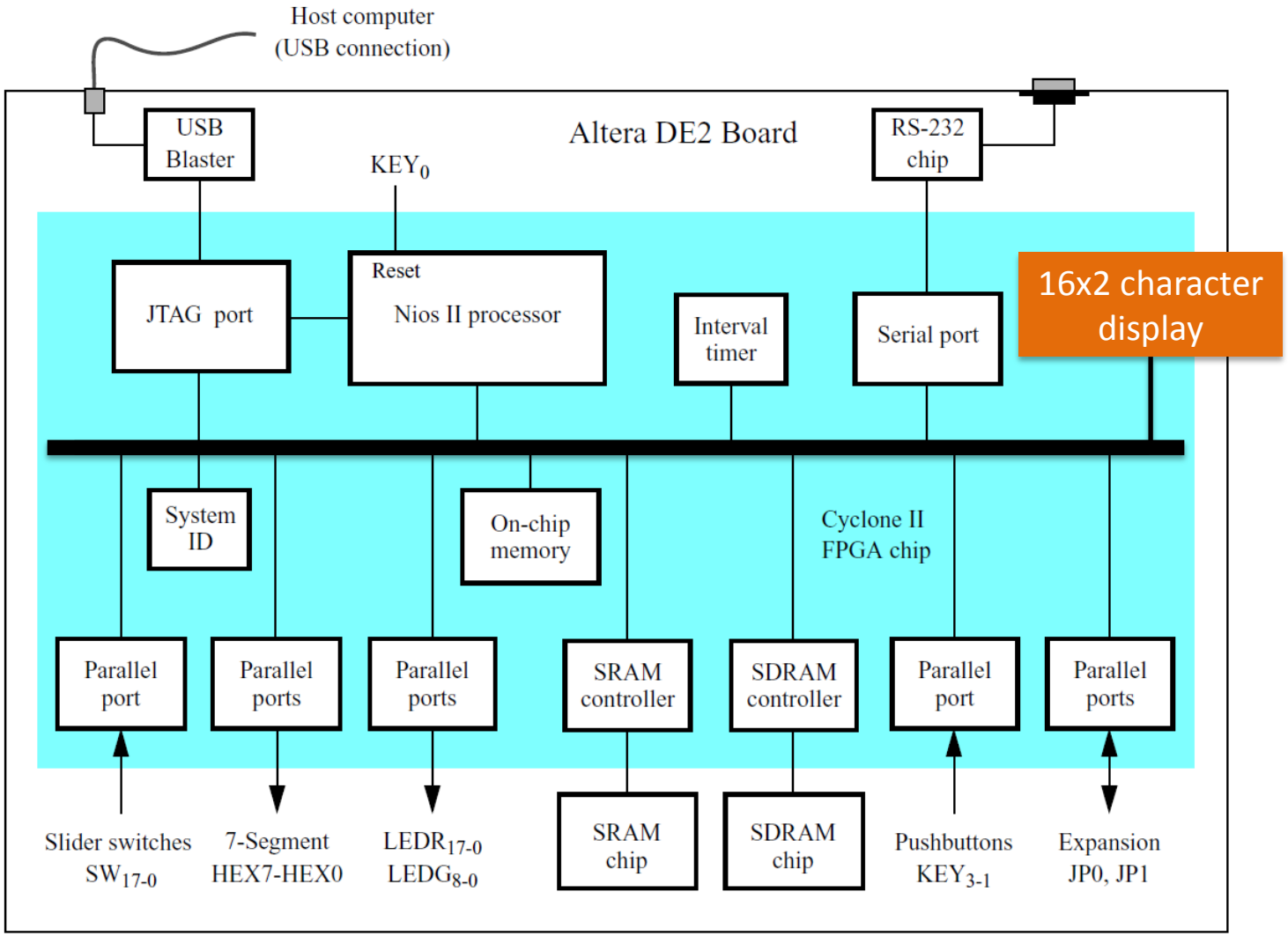
SISTEMI EMBEDDED
AA 2011/2012

Manipulating an existing
Nios II processor

Guided example (1a)

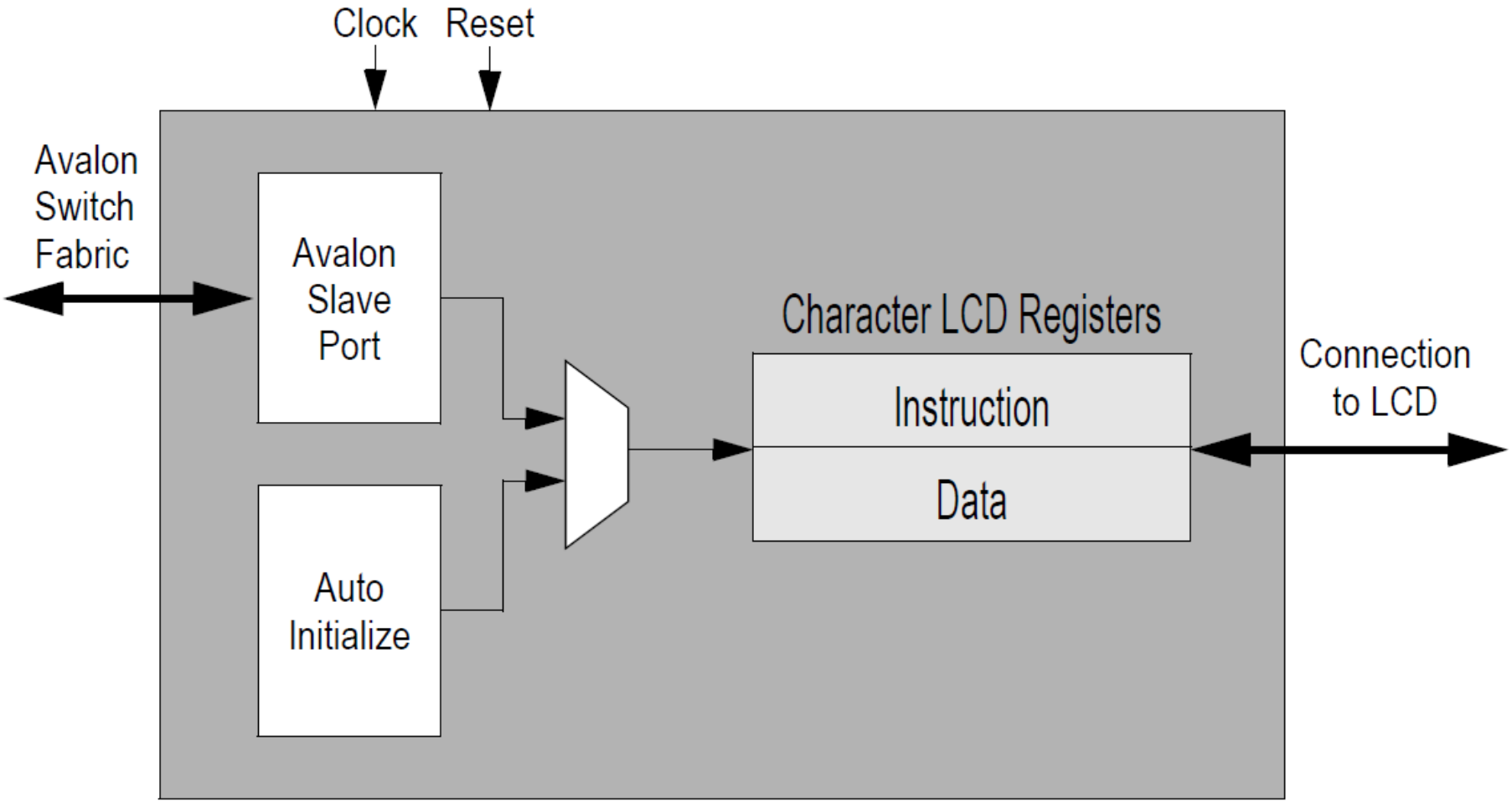
- Expanding the DE2 Basic Computer adding the **16x2 Character Display** peripheral



Guided example (1b)

- Expanding the DE2 Basic Computer adding the **16x2 Character Display** peripheral

Architecture of the 16x2 character display peripheral



Guided example (1c)

- Expanding the DE2 Basic Computer adding the **16x2 Character Display** peripheral

Port declaration of the 16x2 character LCD module

```
module character_lcd_0 (
        // Inputs
        clk,
        reset,
        address,
        chipselect,
        read,
        write,
        writedata,

        // Bidirectionals
        LCD_DATA,

        // Outputs
        LCD_ON,
        LCD_BLON,
        LCD_EN,
        LCD_RS,
        LCD_RW,
        readdata,
        waitrequest
);
```

Guided example (2)

- Expanding the DE2 Basic Computer adding the **16x2 Character Display** peripheral
  - Create a new project in Quartus II
  - Launch SoPC Builder
    - Open system_nios.sopc from DE2_Basic_Computer folder and save it in the new project folder
    - Add the 16x2 character display and configure it
    - Modify the System_ID = 2 (System ID Peripheral)
    - Generate the expanded Nios II processor
  - Back to Quartus II
    - Import pin assignment from de2.qsf
    - Open the DE2_Basic_Computer.v, save it as a <*top_level_entity*>.v and add the file to the project

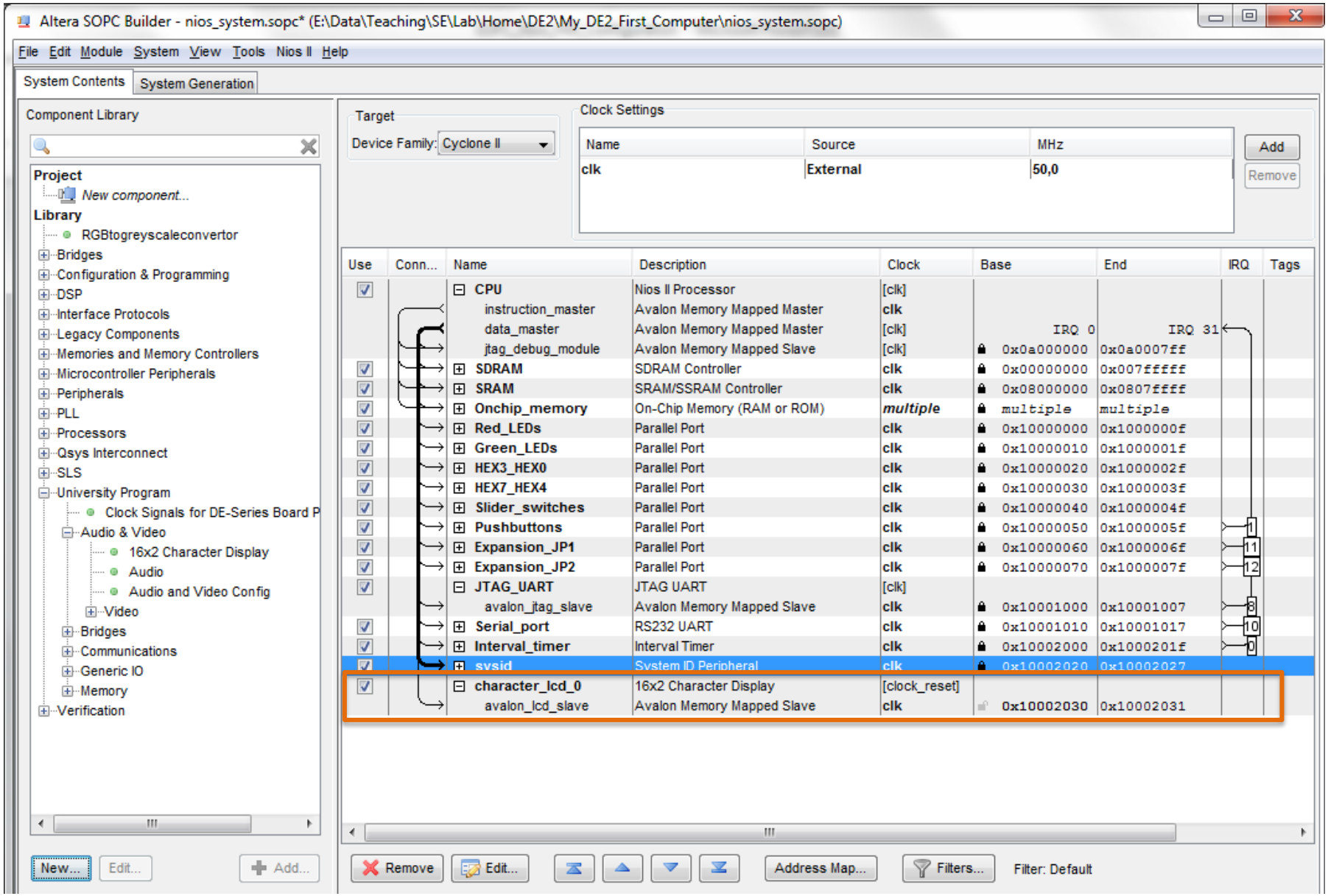
Guided example (3)

- Expanding the DE2 Basic Computer adding the **16x2 Character Display** peripheral
  - Modify the module name
  - Add the LCD ports and connect them to the corresponding Nios II processor signals (to and from the 16x2 character display peripheral)
  - Generate the sdram_pll
    - **DRAM_CLK** must lead the **system_clock** to compensate for clock skew due to DE2 PCB connections
  - Add constraint for **CLOCK_50** as clock
  - Compile the design

Guided example (4)

- Expanding the DE2 Basic Computer adding the **16x2 Character Display** peripheral
  - Move to the Nios II SBT – Eclipse
    - Create a new Nios II Application and BSP from Template; **use the new system_nios.sopcinfo**
    - Write a program to test the LCD display
    - Program the FPGA w/ the new soc file
    - Run the LCD test application!

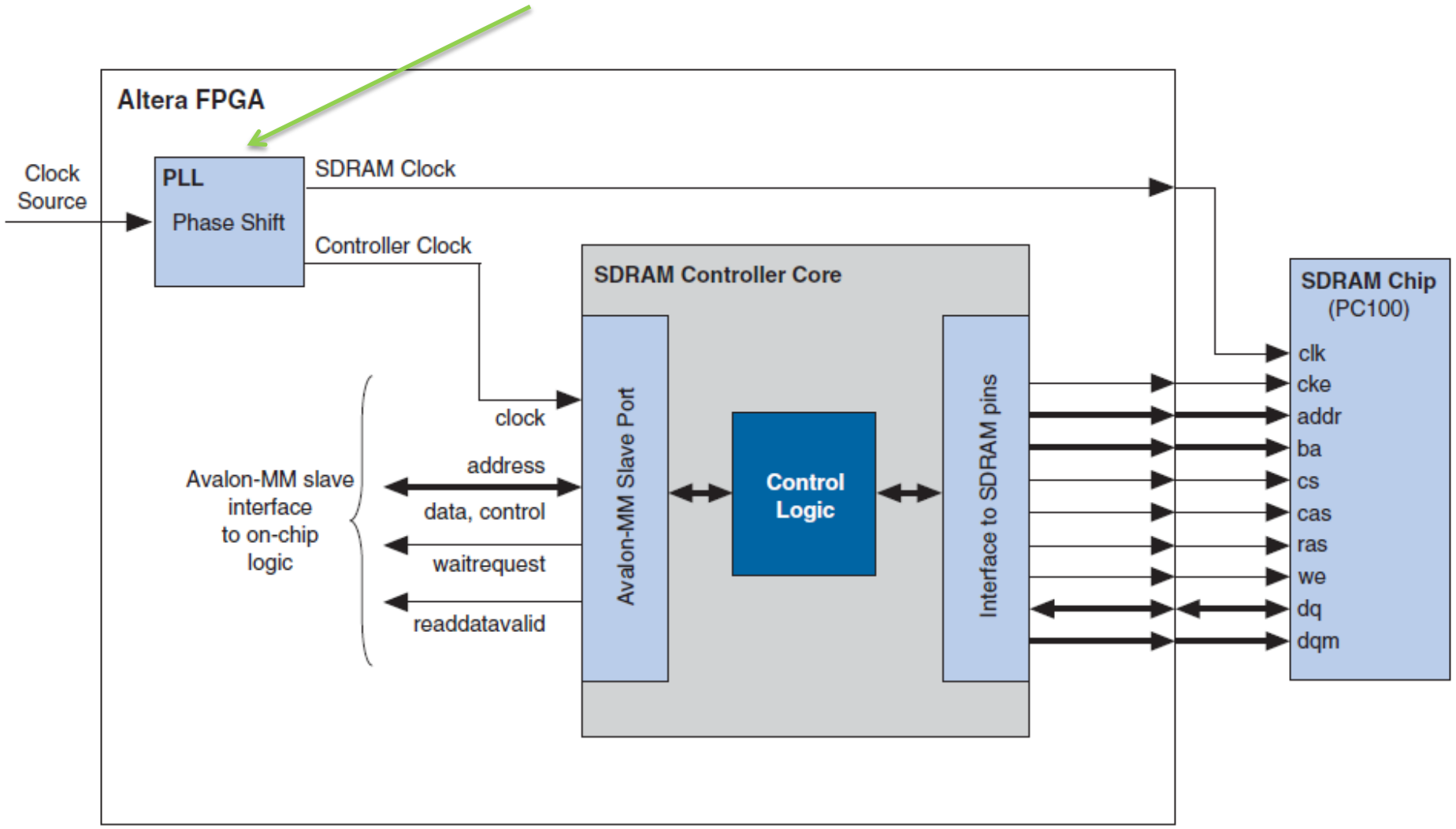
SoPC Builder: new nios_system.sopc



SDRAM controller

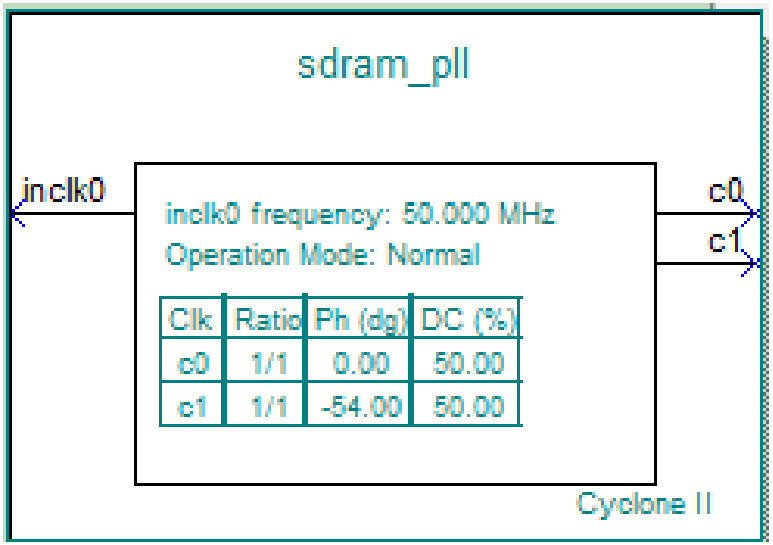
A PLL can be used to compensate for the clock skew introduced by PCB connections

DE2 board *SDRAM Clock* must lead *Controller Clock* by 3 ns (Phase shift)



SDRAM Clock

- Require instantiating and configuring a PLL
  - Can be done using the MegaWizard Plug-in Manager [I/O Library]
  - C0 and c1 have the same frequency as inclok0, i.e. 50 MHz but are shifted eachother by 3 ns



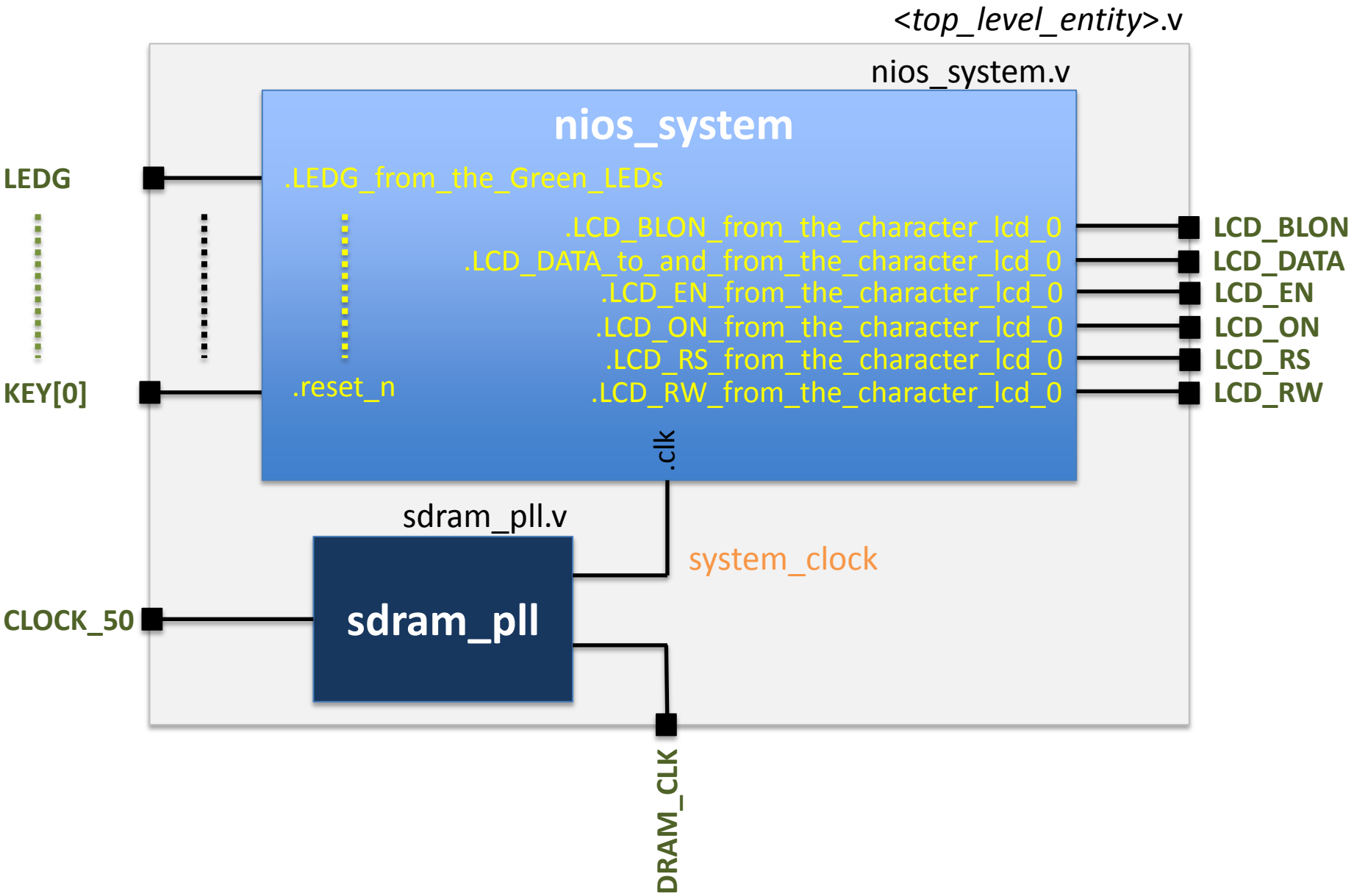
Assembling the SoPC system (1)

The ports of the nios_system and sdram_pll have to be connected eachother or to the external signals of the FPGA

External signals of the FPGA connected to the 16x2 character display

Signal Name	FPGA Pin No.	Description
LCD_DATA[0]	PIN_J1	LCD Data[0]
LCD_DATA[1]	PIN_J2	LCD Data[1]
LCD_DATA[2]	PIN_H1	LCD Data[2]
LCD_DATA[3]	PIN_H2	LCD Data[3]
LCD_DATA[4]	PIN_J4	LCD Data[4]
LCD_DATA[5]	PIN_J3	LCD Data[5]
LCD_DATA[6]	PIN_H4	LCD Data[6]
LCD_DATA[7]	PIN_H3	LCD Data[7]
LCD_RW	PIN_K4	LCD Read/Write Select, 0 = Write, 1 = Read
LCD_EN	PIN_K3	LCD Enable
LCD_RS	PIN_K1	LCD Command/Data Select, 0 = Command, 1 = Data
LCD_ON	PIN_L4	LCD Power ON/OFF
LCD_BLON	PIN_K2	LCD Back Light ON/OFF

Assembling the SoPC system (2)



Character LCD API

- Header file: altera_up_character_lcd.h
- Device type: alt_up_character_lcd_dev
- Function prototypes:
  - alt_up_character_lcd_dev* alt_up_character_lcd_open_dev(const char* name);
  - void alt_up_character_lcd_init(alt_up_character_lcd_dev *lcd);
  - int alt_up_character_lcd_set_cursor_pos (alt_up_character_lcd_dev *lcd, unsigned x_pos, unsigned y_pos);
  - void alt_up_character_lcd_string(alt_up_character_lcd_dev *lcd, const char *ptr);
  - ...

Test the new Nios II system

- Write a simple program that wtites a string on the 16x2 character display

References

- Altera, “Using the SDRAM Memory on Altera’s DE2 Board,” *tut_DE2_sdram_verilog.pdf*
- with Verilog Design
- Altera, “16x2 Character Display for Altera DE2-Series Boards,” *Character_LCD.pdf*