CSCE 611
Quartus and DE2 Board Tutorials

Instructor: Jason D. Bakos

UNIVERSITY OF SOUTH CAROLINA

Setup Your Environment

- Do this once:
  - Open ~/.bashrc
  - Add a line:

```
source /usr/local/3rdparty/cad_setup_files/altera.bash
```

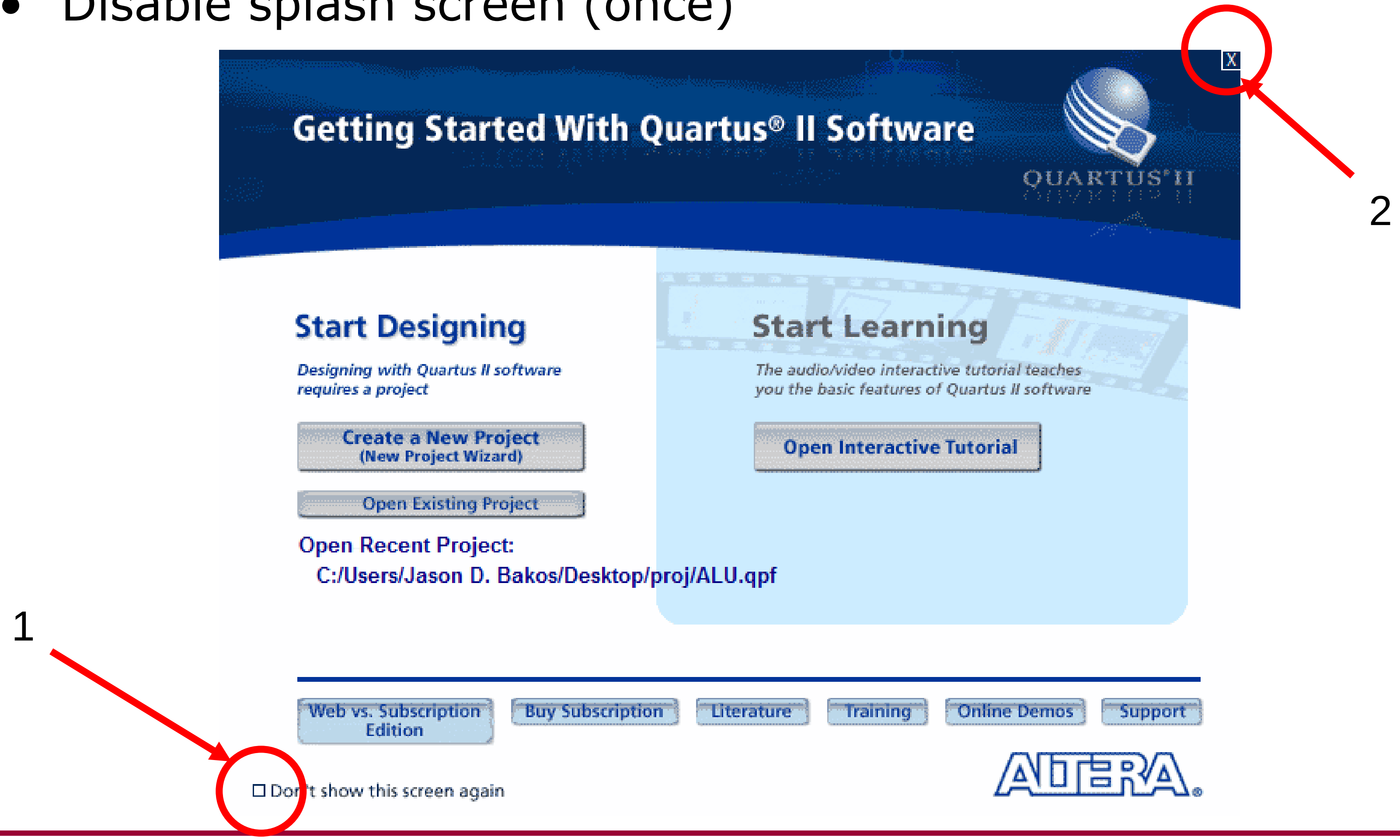
  - Log out, log back in

- Launch Quartus:

```
quartus&
```

Quartus

- Disable splash screen (once)

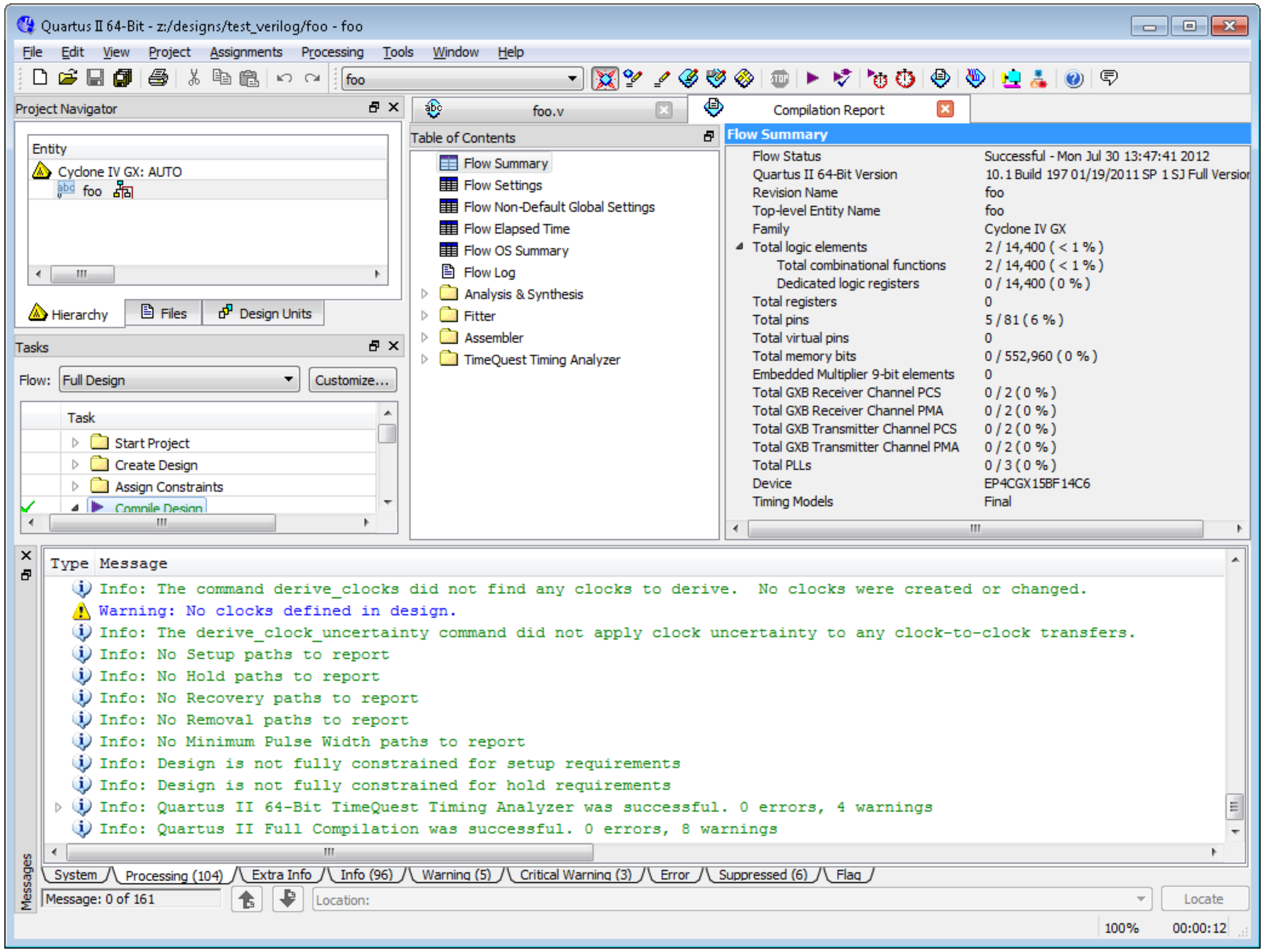


Getting Started With Quartus® II Software
QUARTUS® II
Start Designing
Designing with Quartus II software requires a project
Create a New Project (New Project Wizard)
Open Existing Project
Start Learning
The audio/video interactive tutorial teaches you the basic features of Quartus II software
Open Interactive Tutorial
Web vs. Subscription Edition
Buy Subscription
Literature
Training
Online Demos
Support
ALTERA
X
1
2

Open Recent Project:
C:/Users/Jason D. Bakos/Desktop/proj/ALU.qpf

☐ Don't show this screen again

Quartus



Flow Summary

Flow Status	Successful - Mon Jul 30 13:47:41 2012
Quartus II 64-Bit Version	10.1 Build 197 01/19/2011 SP 1 SJ Full Versior
Revision Name	foo
Top-level Entity Name	foo
Family	Cyclone IV GX
Total logic elements	2 / 14,400 (< 1 %)
Total combinational functions	2 / 14,400 (< 1 %)
Dedicated logic registers	0 / 14,400 (0 %)
Total registers	0
Total pins	5 / 81 (6 %)
Total virtual pins	0
Total memory bits	0 / 552,960 (0 %)
Embedded Multiplier 9-bit elements	0
Total GXB Receiver Channel PCS	0 / 2 (0 %)
Total GXB Receiver Channel PMA	0 / 2 (0 %)
Total GXB Transmitter Channel PCS	0 / 2 (0 %)
Total GXB Transmitter Channel PMA	0 / 2 (0 %)
Total PLLs	0 / 3 (0 %)
Device	EP4CGX15BF14C6
Timing Models	Final

```
Type Message
     Info: The command derive_clocks did not find any clocks to derive.  No clocks were created or changed.
     Warning: No clocks defined in design.
     Info: The derive_clock_uncertainty command did not apply clock uncertainty to any clock-to-clock transfers.
     Info: No Setup paths to report
     Info: No Hold paths to report
     Info: No Recovery paths to report
     Info: No Removal paths to report
     Info: No Minimum Pulse Width paths to report
     Info: Design is not fully constrained for setup requirements
     Info: Design is not fully constrained for hold requirements
     Info: Quartus II 64-Bit TimeQuest Timing Analyzer was successful. 0 errors, 4 warnings
     Info: Quartus II Full Compilation was successful. 0 errors, 8 warnings
```

Editor Settings

- Editor settings
  - Go to Tools | Options | Text Editor | Autocomplete Text
  - Turn off!

Quartus

- Create a new project:
  - File | New | New Quartus II Project
  - On first screen, turn on “Don’t show me this introduction again” and click Next
  - Working directory:
    - /acct/<your username>/quartus_work
  - Project name:
    - “my_611_project” (or anything else)
  - Click FINISH

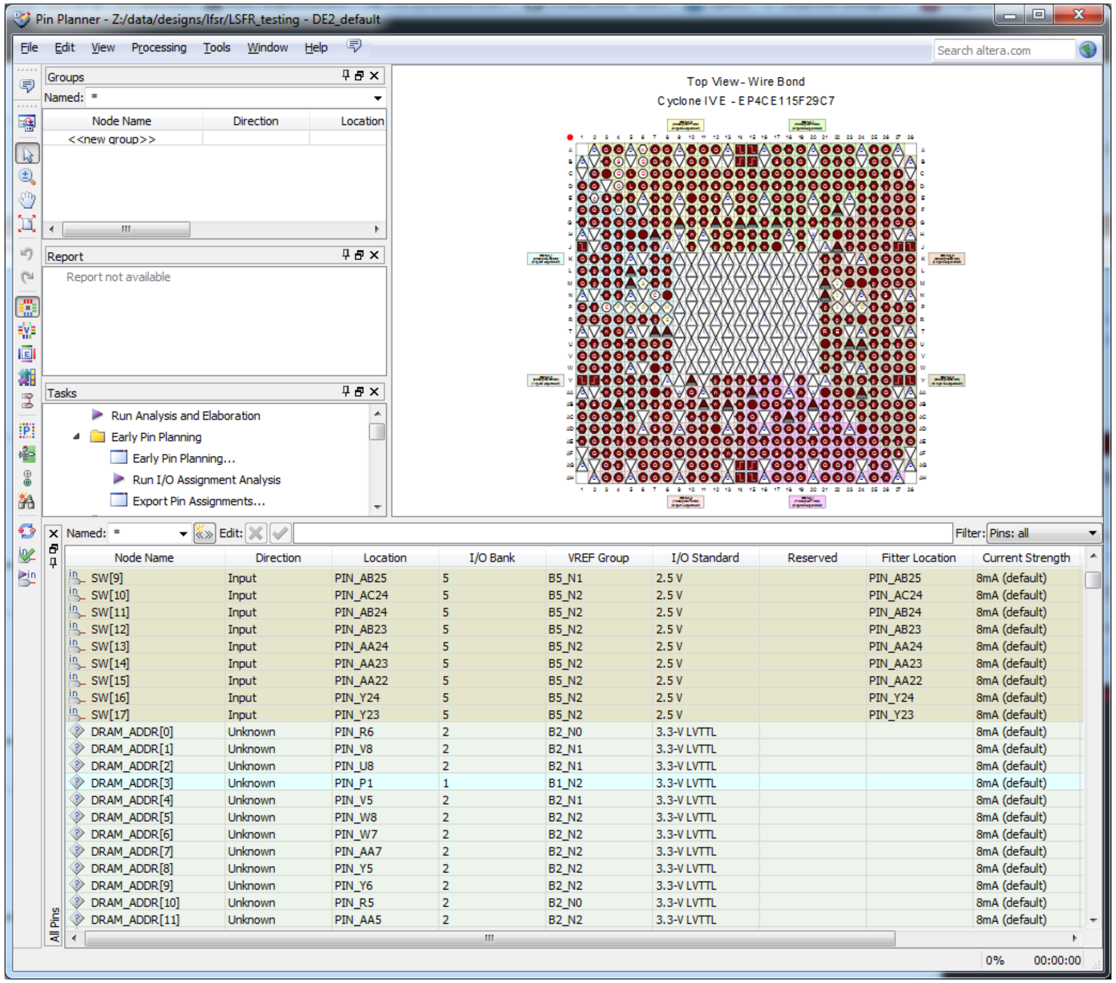
Interfacing to Board Components

- Set up FPGA:
  - Go to Assignments | Device
  - For device, choose
    - **Family**: Cyclone IV E (DE2-115) / Cyclone II (DE2)
    - **Package:** FBGA
    - **Pin count:** 780/672
    - **Speed grade:** 7/6
    - **Device:** EP4CE115F29C7 (DE2-115) / EP2C35F672C6 (DE2)

Pin Mappings

- It’s easy to connect signals in your top-level design to devices on the FPGA board
- Pin mapping file:
  `/usr/local/3rdparty/csce611/CPU_support_files/DE2_115_pin_assignments.qsf (DE2-115)`
  `/usr/local/3rdparty/csce611/CPU_support_files/DE2_pin_assignments.csv (DE2)`
- To use, select Assignments | Import Assignments
- Select file and click OK
- To verify: Assignments | Pin Planner
- Shows top-level I/Os and names of FPGA interface pins
- To connect to pins, use these names as I/O in top-level design

Pin Planner


Pin Planner - Z:/data/designs/lfsr/LSFR_testing - DE2_default
File Edit View Processing Tools Window Help
Search altera.com
Groups
Named: *
Report
Tasks
Top View - Wire Bond
Cyclone IV E - EP4CE115F29C7
Named: *
Edit:
Filter: Pins: all
All Pins
0%
00:00:00

Node Name	Direction	Location
<<new group>>		

Report not available

- Run Analysis and Elaboration
- Early Pin Planning
  - Early Pin Planning...
  - Run I/O Assignment Analysis
  - Export Pin Assignments...

Node Name	Direction	Location	I/O Bank	VREF Group	I/O Standard	Reserved	Fitter Location	Current Strength
SW[9]	Input	PIN_AB25	5	B5_N1	2.5 V		PIN_AB25	8mA (default)
SW[10]	Input	PIN_AC24	5	B5_N2	2.5 V		PIN_AC24	8mA (default)
SW[11]	Input	PIN_AB24	5	B5_N2	2.5 V		PIN_AB24	8mA (default)
SW[12]	Input	PIN_AB23	5	B5_N2	2.5 V		PIN_AB23	8mA (default)
SW[13]	Input	PIN_AA24	5	B5_N2	2.5 V		PIN_AA24	8mA (default)
SW[14]	Input	PIN_AA23	5	B5_N2	2.5 V		PIN_AA23	8mA (default)
SW[15]	Input	PIN_AA22	5	B5_N2	2.5 V		PIN_AA22	8mA (default)
SW[16]	Input	PIN_Y24	5	B5_N2	2.5 V		PIN_Y24	8mA (default)
SW[17]	Input	PIN_Y23	5	B5_N2	2.5 V		PIN_Y23	8mA (default)
DRAM_ADDR[0]	Unknown	PIN_R6	2	B2_N0	3.3-V LVTTL			8mA (default)
DRAM_ADDR[1]	Unknown	PIN_V8	2	B2_N1	3.3-V LVTTL			8mA (default)
DRAM_ADDR[2]	Unknown	PIN_U8	2	B2_N1	3.3-V LVTTL			8mA (default)
DRAM_ADDR[3]	Unknown	PIN_P1	1	B1_N2	3.3-V LVTTL			8mA (default)
DRAM_ADDR[4]	Unknown	PIN_V5	2	B2_N1	3.3-V LVTTL			8mA (default)
DRAM_ADDR[5]	Unknown	PIN_W8	2	B2_N2	3.3-V LVTTL			8mA (default)
DRAM_ADDR[6]	Unknown	PIN_W7	2	B2_N2	3.3-V LVTTL			8mA (default)
DRAM_ADDR[7]	Unknown	PIN_AA7	2	B2_N2	3.3-V LVTTL			8mA (default)
DRAM_ADDR[8]	Unknown	PIN_Y5	2	B2_N2	3.3-V LVTTL			8mA (default)
DRAM_ADDR[9]	Unknown	PIN_Y6	2	B2_N2	3.3-V LVTTL			8mA (default)
DRAM_ADDR[10]	Unknown	PIN_R5	2	B2_N0	3.3-V LVTTL			8mA (default)
DRAM_ADDR[11]	Unknown	PIN_AA5	2	B2_N2	3.3-V LVTTL			8mA (default)

Constraints File

- File | New | Synopsys Design Constraints File

```
create_clock -name CLOCK_50 -period 20 [get_ports CLOCK_50]
```

- Save as SDC1.sdc

DE2 Test Design: Binary Counter

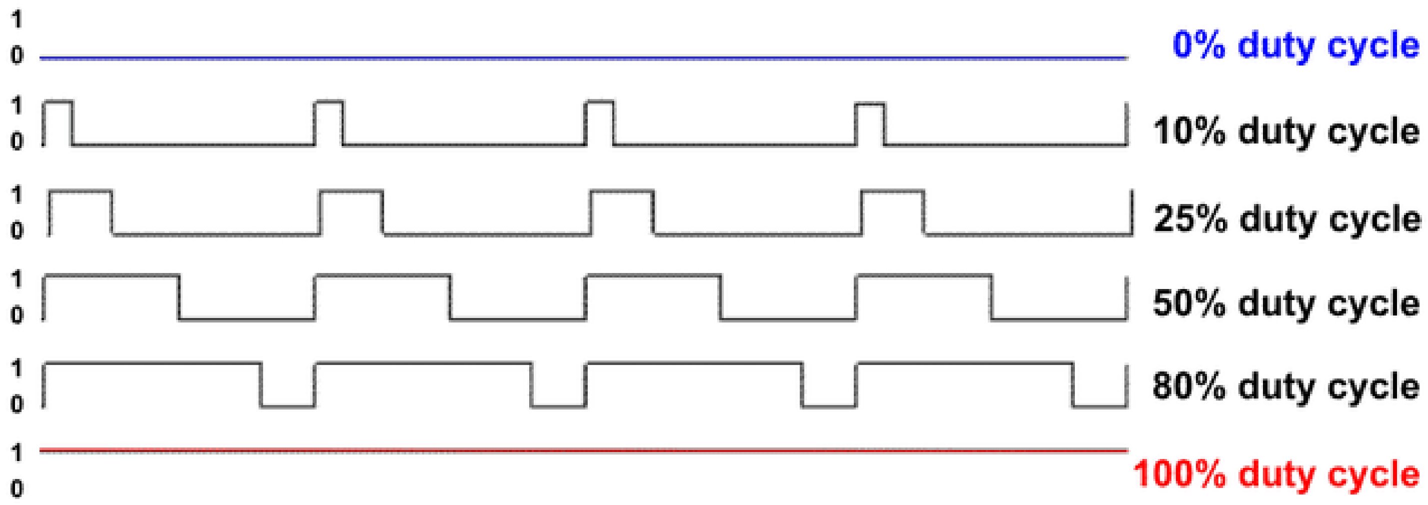
```
module test_de2 (input CLOCK_50,output reg [17:0] LEDR);
reg [23:0] clk_div;
initial begin
  LEDR <= 18'b0;
  clk_div <= 24'b0;
end
always @(posedge CLOCK_50) begin
 // divide 50 MHz clock by 2^24 (16 million)
  clk_div <= clk_div+24'b1;
end
always @(posedge clk_div[23]) begin
  LEDR <= LEDR + 18'b1;
end
endmodule
```

DE2 Test Design: Shifter

```
module test_de2 (input CLOCK_50,output reg [17:0] LEDR);
reg [23:0] clk_div;
reg left;
initial begin
  LEDR <= 18'b1;
  clk_div <= 24'b0;
  left <= 1'b1;
end
always @(posedge CLOCK_50) begin
 // divide 50 MHz clock by 2^24 (16 million)
  clk_div <= clk_div+24'b1;
end
always @(posedge clk_div[23]) begin
  if (left) LEDR <= LEDR << 1; else LEDR <= LEDR >> 1;
  if ((left && LEDR[16]) || (!left && LEDR[1])) left <= !left;
end
endmodule
```

Pulse Width Modulation

- Way to implement an analog value with a digital pin
- Regular periodic signal whose duty cycle determines average voltage over time



- Easy to implement: use a counter to divide the clock, set output high when counter < desired width

DE Test Design: PWM Modulator

```
module pwm (input CLOCK_50,output reg [17:0] LEDR);
reg [25:0] clk_div;
reg [10:0] pwm_width;
reg [10:0] pwm_cnt;
reg increase;
wire pwm_out;
always @(posedge CLOCK_50) begin
  pwm_cnt = pwm_cnt+8'b1;
end
assign pwm_out = (pwm_cnt <= pwm_width) ? 1'b1 : 1'b0;
initial begin
  LEDR <= 0;
  clk_div <= 0;
end
always @(*) LEDR = {18{pwm_out}};
always @(posedge CLOCK_50) begin
  clk_div <= clk_div+26'b1;
end
always @(posedge clk_div[13]) begin
  if (increase) pwm_width = pwm_width+1'b1; else pwm_width = pwm_width-1'b1;
  if (pwm_width==11'd2047) increase=1'b0;
  if (pwm_width==11'd0) increase=1'b1;
end
endmodule
```