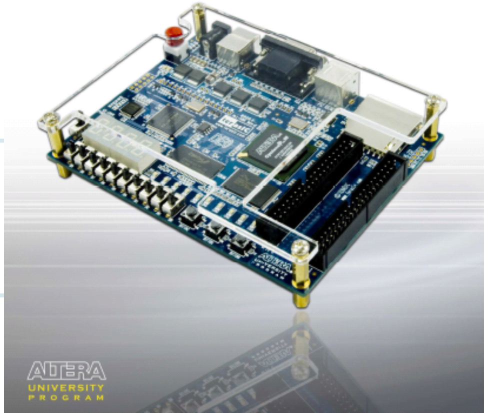
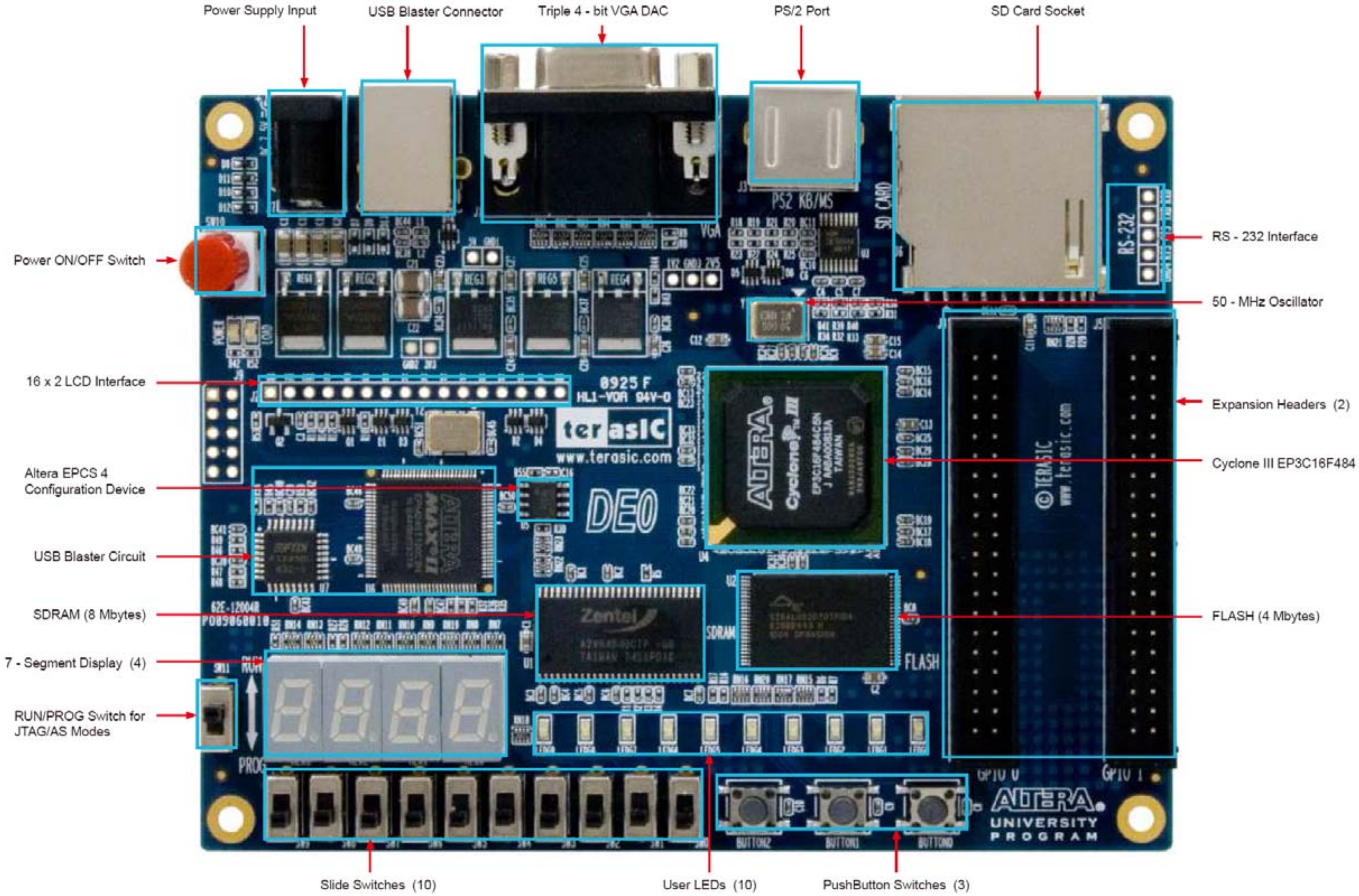
Class 5
DE0 FPGA Development Board
and
Quartus II 9.1 FPGA Design Software

Terasic DE0 Field Programmable Gate Array (FPGA) Development Board

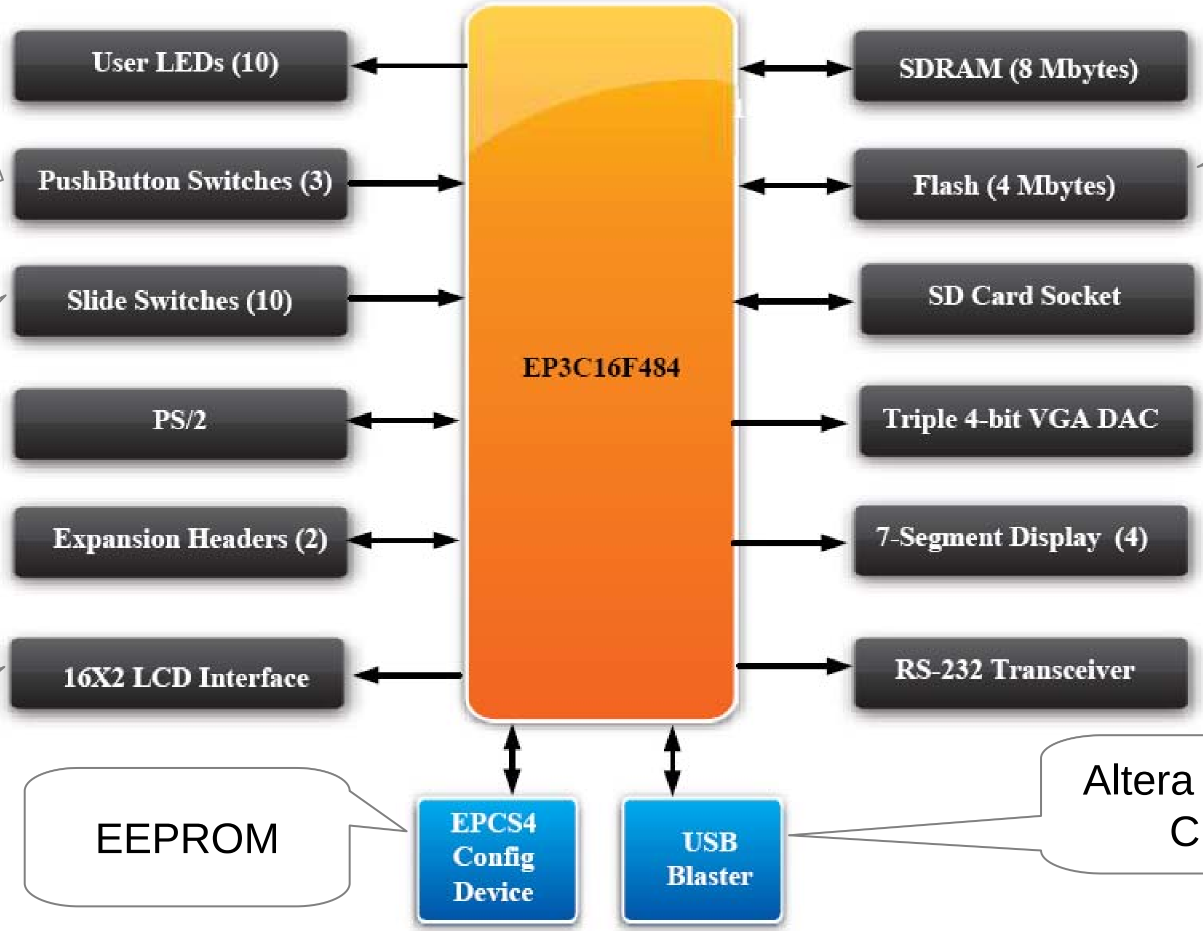


Layout and Components of DE0



Block Diagram of the DE0 Board

Cyclone III 3C16 FPGA:15,408 Les, 4 PLLs, 346 I/O pins



EP3C16F484

User LEDs (10)

PushButton Switches (3)

Slide Switches (10)

PS/2

Expansion Headers (2)

16X2 LCD Interface

SDRAM (8 Mbytes)

Flash (4 Mbytes)

SD Card Socket

Triple 4-bit VGA DAC

7-Segment Display (4)

RS-232 Transceiver

EPCS4 Config Device

USB Blaster

Normally high, generate active-low pulse when pressed

Logic 0/1 when in the Down/Up position

16 x 2 Character LCD

EEPROM

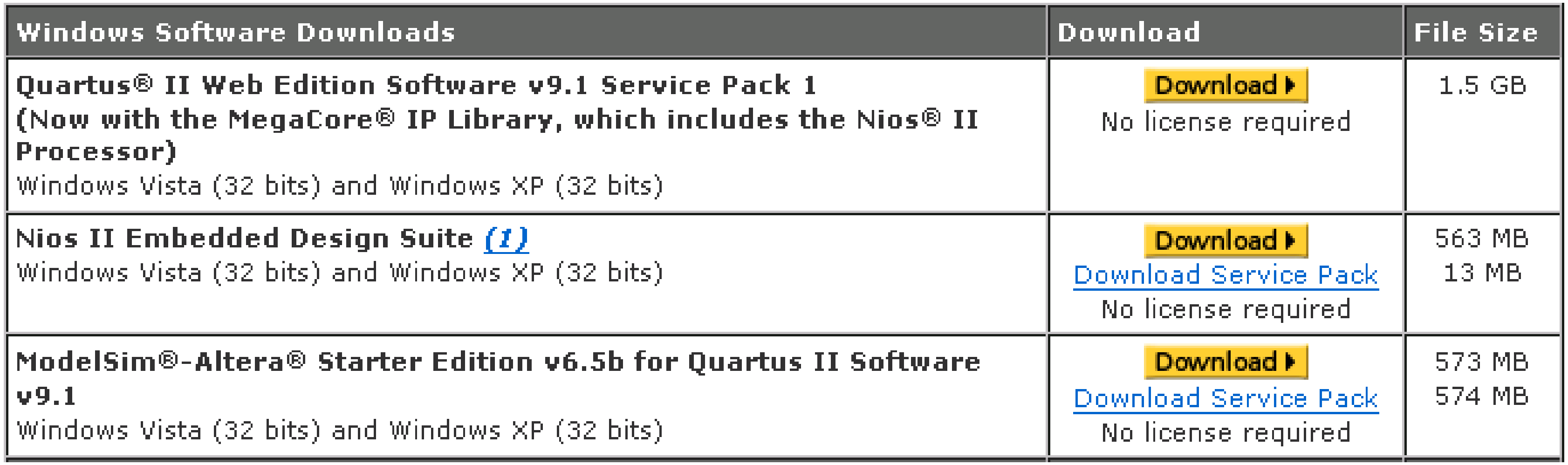
NOR flash

SPI and SD 1-bit modes

Altera EPM240 CPLD

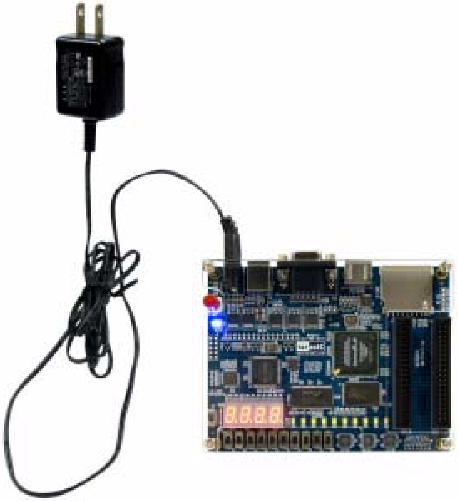
DE0 Installation

- Step 1: Install the Altera Design Software on the host computer.
  - Download the software: http://www.altera.com/download
    - Quartus II: the primary FPGA development tool
    - Nios II: soft-core embedded processor
    - ModelSim-Altera: Simulation tool

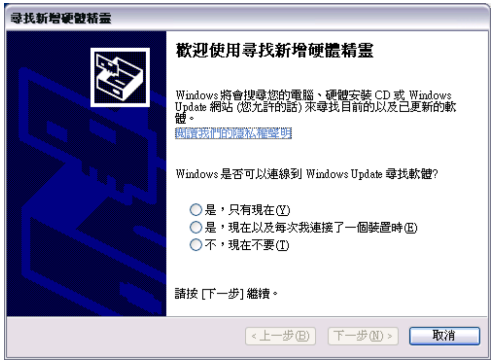


Windows Software Downloads	Download	File Size
Quartus® II Web Edition Software v9.1 Service Pack 1 (Now with the MegaCore® IP Library, which includes the Nios® II Processor) Windows Vista (32 bits) and Windows XP (32 bits)	Download ▸ No license required	1.5 GB
Nios II Embedded Design Suite *(1)* Windows Vista (32 bits) and Windows XP (32 bits)	Download ▸ Download Service Pack No license required	563 MB 13 MB
ModelSim®-Altera® Starter Edition v6.5b for Quartus II Software v9.1 Windows Vista (32 bits) and Windows XP (32 bits)	Download ▸ Download Service Pack No license required	573 MB 574 MB

DE0 Installation (Cont.)



- Step 2: Install the USB Blaster
  - Plug in the power cable.
  - Use the USB cable to connect the USB connector on the DE0 board to a USB port on a computer.
    - 1. Recognize the new hardware connected.
    - 2. Specify the path for the USB Blaster driver.
    - 3. Select appropriate driver. (**C:\altera\91\quartus\drivers\usb-blaster**)
    - 4. Install USB Blast driver. (**C:\altera\91\quartus\drivers\usb-blaster\x32**)



尋找新增硬體精靈

歡迎使用尋找新增硬體精靈

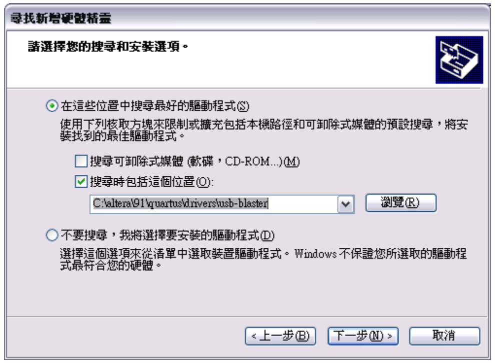
Windows 將會搜尋您的電腦、硬體安裝 CD 或 Windows Update 網站 (您允許的話) 來尋找目前的以及已更新的軟體。

閱讀我們的隱私權聲明

Windows 是否可以連線到 Windows Update 尋找軟體?

○是，只有現在(Y)
○是，現在以及每次我連接了一個裝置時(E)
○不，現在不要(T)

請按 [下一步] 繼續。



尋找新增硬體精靈

請選擇您的搜尋和安裝選項。

●在這些位置中搜尋最好的驅動程式(S)

使用下列核取方塊來限制或擴充包括本機路徑和可卸除式媒體的預設搜尋，將安裝找到的最佳驅動程式。

□搜尋可卸除式媒體 (軟碟，CD-ROM...)(M)

☑搜尋時包括這個位置(O):

C:\altera\91\quartus\drivers\usb-blaster

○不要搜尋，我將選擇要安裝的驅動程式(D)

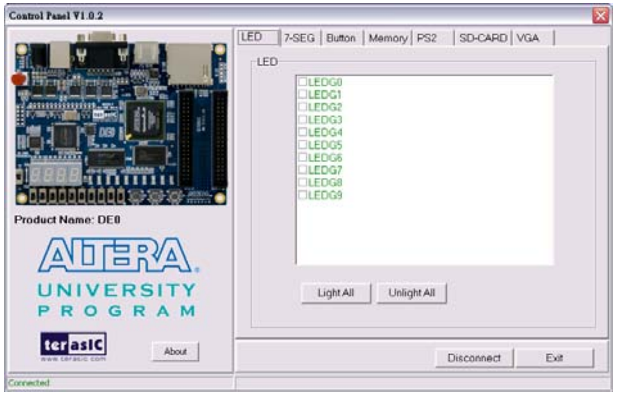
選擇這個選項來從清單中選取裝置驅動程式。Windows 不保證您所選取的驅動程式最符合您的硬體。

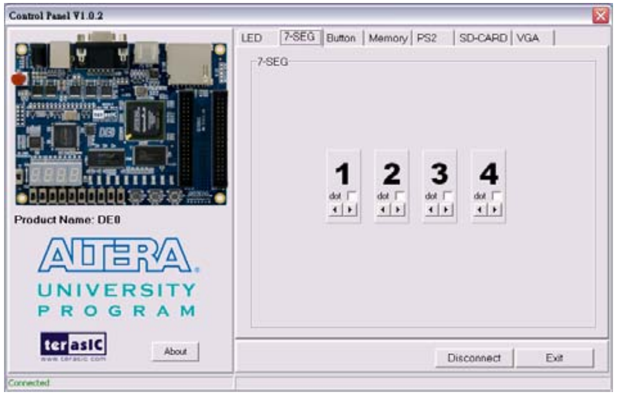
Power-Up the DE0 Board

- The DE0 board comes with a preloaded configuration bit stream to demonstrate some feature of the board.
  - All user LEDs are flashing.
  - All 7-segment displays are cycling through 0 to F.
  - The VGA monitor displays the image as shown in the right-hand side:

DE0 Control Panel

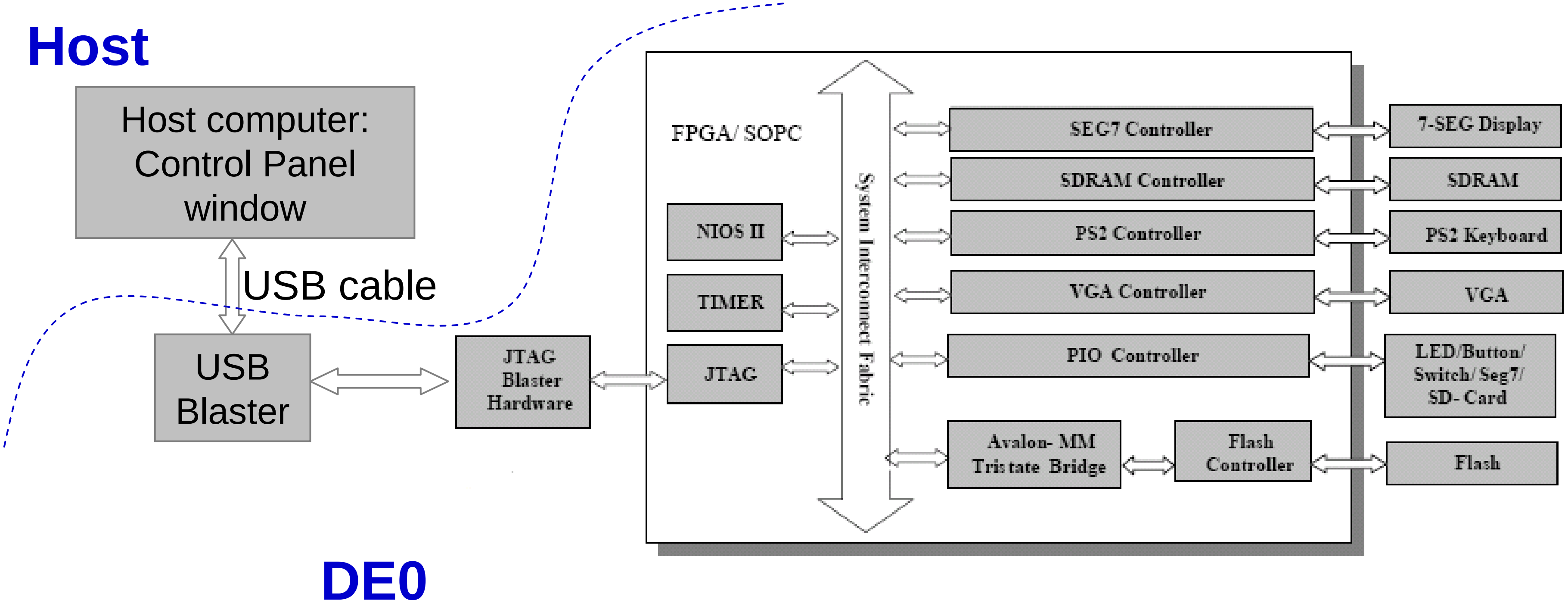
- The DE0 board comes with a Control Panel facility. (Start the executable *DE0_ControlPanel.exe*)
  - Allow users to access various components on the board from a host computer.
  - Connect the host computer with the DE0 board through a USB connection.
  - Verify the functionality of components on the board.





DE0 Control Panel (Cont.)

- The control codes that perform the control functions
  - Is implemented in the FPGA board, and
  - Communicates with the Control Panel window on the host computer.

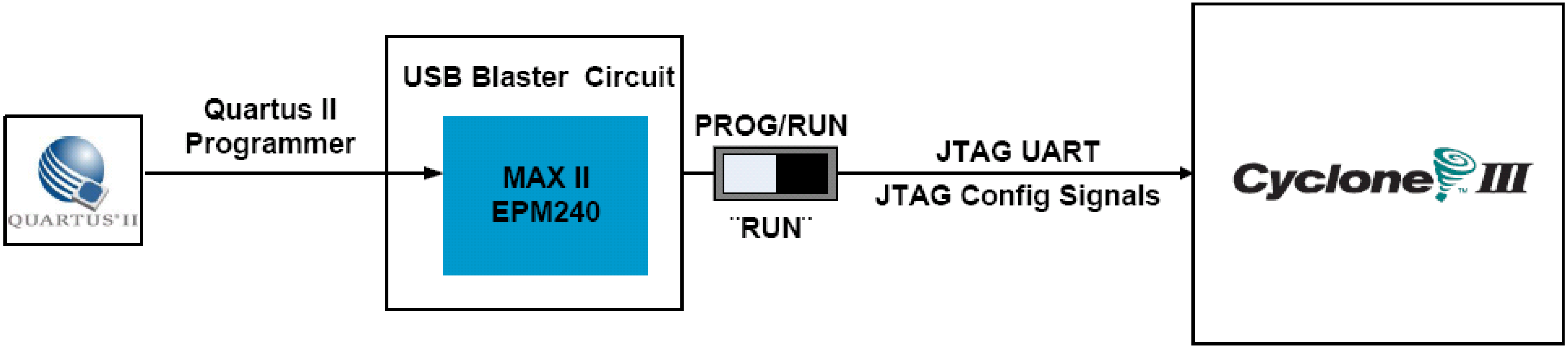


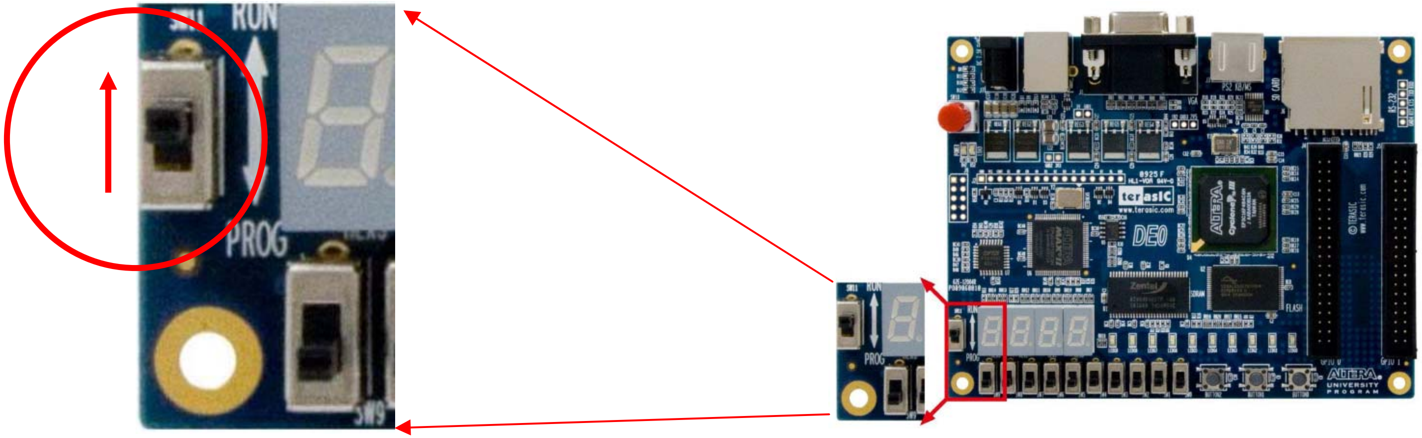
Configuring the Cyclone III FPGA

- The DE0 board contains a serial EEPROM chip (i.e., the EPCS4 device) that stores configuration data for the Cyclone III FPGA.
  - The configuration data is automatically loaded from the EEPROM chip into the FPGA once the power is applied to the board.
  - With Quartus II, it is possible to reprogram the FPGA and to change the non-volatile data in the EEPROM chip.
    - JTAG (Joint Test Action Group) programming: Download the configuration to FPGA directly, but the configuration is lost when the power is off.
    - AS (Active Serial) programming: Download the configuration into the EEPROM chip, and the configuration is retained when the power is off. When the power is on, data is loaded from the EEPROM.

Configuring the FPGA in JTAG Mode

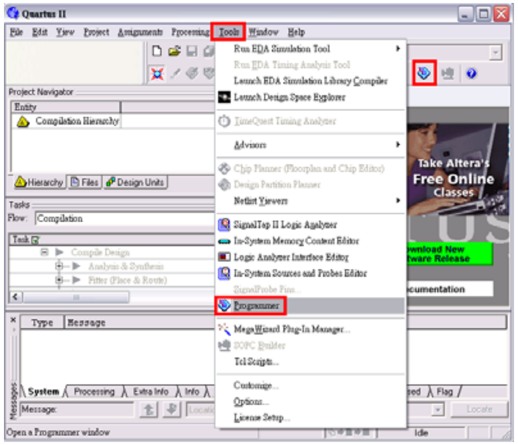
- Download the .sof (SRAM Object File) file by the programmer of Quartus II

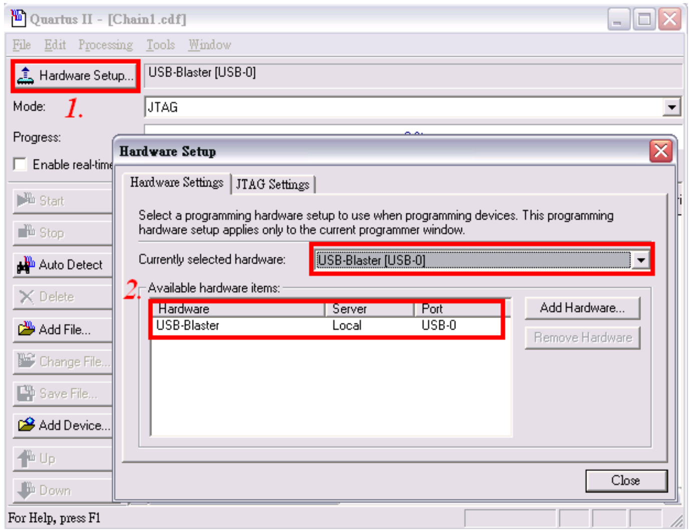




Configuring the FPGA in JTAG Mode (Cont.)

- The steps to program SRAM Object File (.sof) into the FPGA device on the DE0 board are as follows:
  - Step 1: Power on DE0 board with SW11 to **RUN** mode and connect it to the host. Step 2: Open Quartus II, and choose **Tools → Programmer**
  - Step 3: Click “**Hardware Setup**” and then Select “**USB Blaster**”
  - Step 4: Click “**Add File**” to select the .sof file in JTAG mode, and then click “**Start**” to program it.


Quartus II
File Edit View Project Assignments Processing Tools Window Help
Run EDA Simulation Tool
Run EDA Timing Analysis Tool
Launch EDA Simulation Library Compiler
Launch Design Space Explorer
TimeQuest Timing Analyzer
Advisors
Chip Planner (Floorplan and Chip Editor)
Design Partition Planner
Netlist Viewers
SignalTap II Logic Analyzer
In-System Memory Content Editor
Logic Analyzer Interface Editor
In-System Sources and Probes Editor
SignalProbe Pins...
Programmer
MegaWizard Plug-In Manager...
SOPC Builder
Tcl Scripts...
Customize...
Options...
License Setup...
Project Navigator
Entity
Compilation Hierarchy
Hierarchy
Files
Design Units
Tasks
Flow: Compilation
Task
Compile Design
Analysis & Synthesis
Fitter (Place & Route)
Take Altera's Free Online Classes
Download New Software Release
Documentation
Type
Message
System
Processing
Extra Info
Info
Flag
Message:
Locate
Open a Programmer window
Idle


Quartus II - [Chain1.cdf]

Hardware Setup... USB-Blaster [USB-0]

Mode: JTAG

Hardware Setup

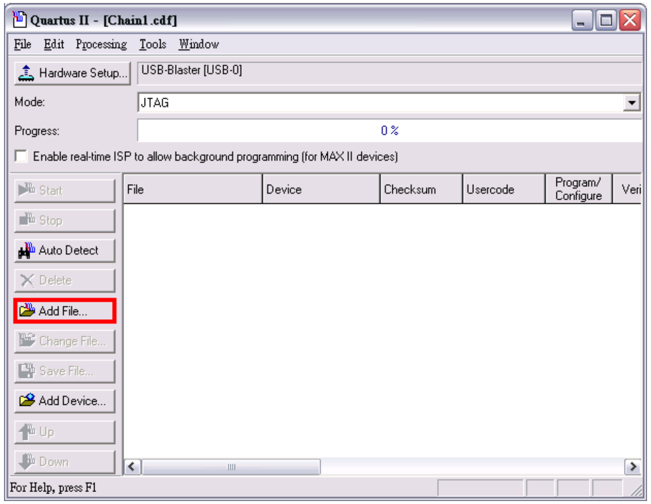
Select a programming hardware setup to use when programming devices. This programming hardware setup applies only to the current programmer window.

Currently selected hardware: USB-Blaster [USB-0]

Available hardware items:

Hardware	Server	Port
USB-Blaster	Local	USB-0

For Help, press F1


Quartus II - [Chain1.cdf]

Hardware Setup... USB-Blaster [USB-0]

Mode: JTAG

Progress: 0 %

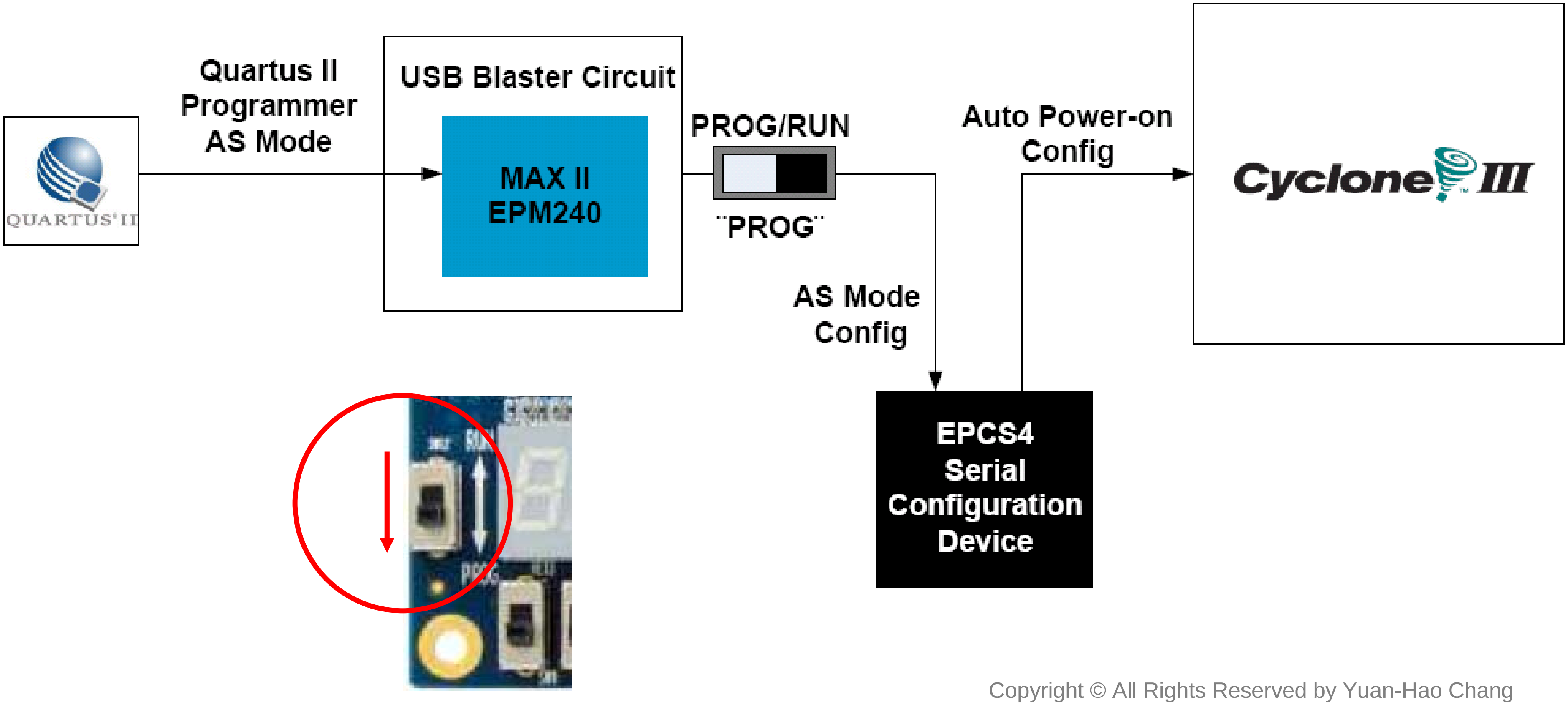
□ Enable real-time ISP to allow background programming (for MAX II devices)

File	Device	Checksum	Usercode	Program/ Configure	Veri

For Help, press F1

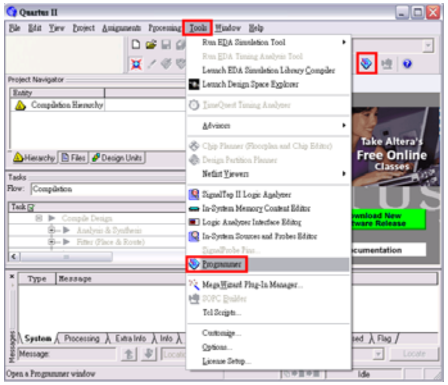
Configuring the EPCS in AS Mode

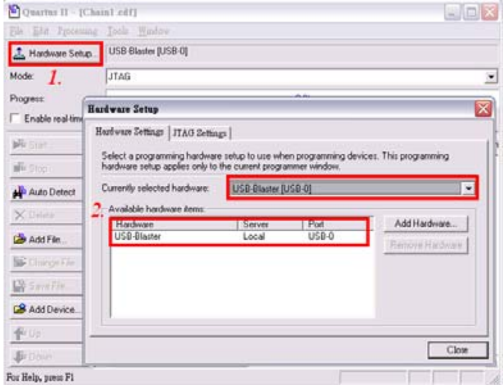
- Download the .pof (Programmer Object File) file by the programmer of Quartus II



Configuring the EPCS4 in AS Mode (Cont.)

- The steps to program Programmer Object File (.pof) into the EPCS4 device on the DE0 board are as follows:
  - Step 1: Power on DE0 board with SW11 to **PROG** mode and connect it to the host.
  - Step 2: Open Quartus II, and choose **Tools → Programmer**
  - Step 3: Click "**Hardware Setup**" and then Select "**USB Blaster**"
  - Step 4: Click "**Add File**" to select the .pof file in Active Serial Programming mode, and then click "**Start**" to program it. (Remember to select "Add Device →"EPCS4")


Quartus II
File Edit View Project Assignments Processing Tools Window Help
Run EDA Simulation Tool
Run EDA Timing Analysis Tool
Launch EDA Simulation Library Compiler
Launch Design Space Explorer
TimeQuest Timing Analyzer
Advisors
Chip Planner (Floorplan and Chip Editor)
Design Partition Planner
Netlist Viewers
SignalTap II Logic Analyzer
In-System Memory Content Editor
Logic Analyzer Interface Editor
In-System Sources and Probes Editor
SignalProbe Pins...
Programmer
MegaWizard Plug-In Manager...
SOPC Builder
Tcl Scripts...
Customize...
Options...
License Setup...
Project Navigator
Entity
Compilation Hierarchy
Hierarchy
Files
Design Units
Tasks
Flow: Compilation
Task
Compile Design
Analysis & Synthesis
Fitter (Place & Route)
Type
Message
System
Processing
Extra Info
Info
Flag
Locate
Take Altera's Free Online Classes
Download New Software Release
Documentation
Open a Programmer window
Idle



USB-Blaster [USB-0]

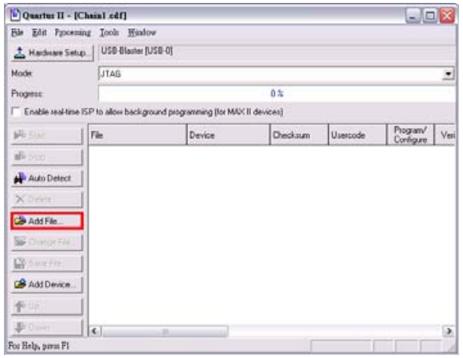
Mode: JTAG

Select a programming hardware setup to use when programming devices. This programming hardware setup applies only to the current programmer window.

Currently selected hardware: USB-Blaster [USB-0]

Available hardware items:

Hardware	Server	Port
USB-Blaster	Local	USB-0



USB-Blaster [USB-0]

Mode: JTAG

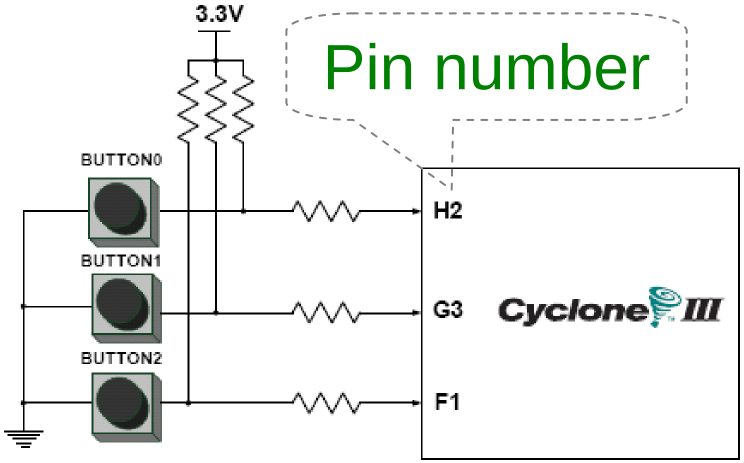
Progress: 0 %

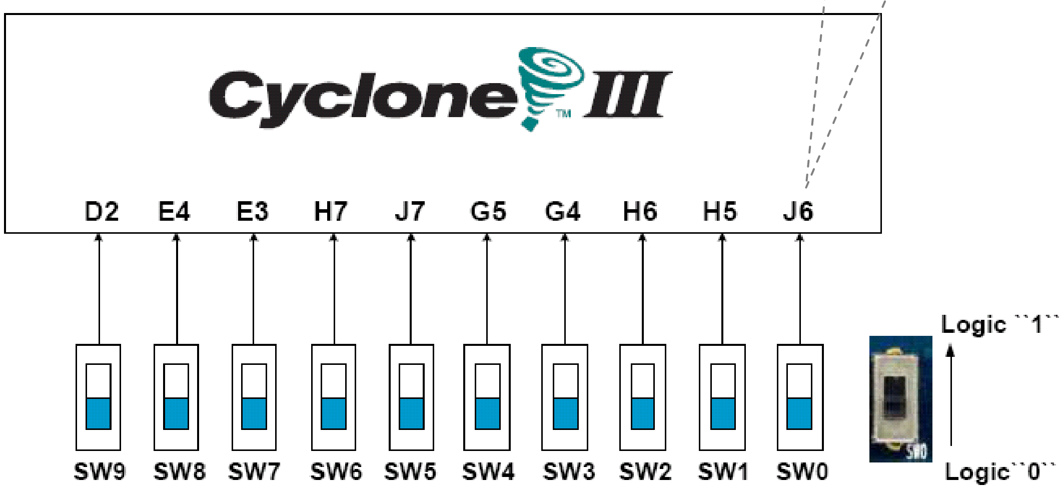
☐ Enable real-time ISP to allow background programming (for MAX II devices)

Setting for generating .pof file: Assignments→Settings→Device→Device and Pin Options→Configuration→ "Active Serial", EPCS4", and check "Use configuration device" & "Generate compressed bitstreams".

Pushbutton and Slide Switches

Pin number





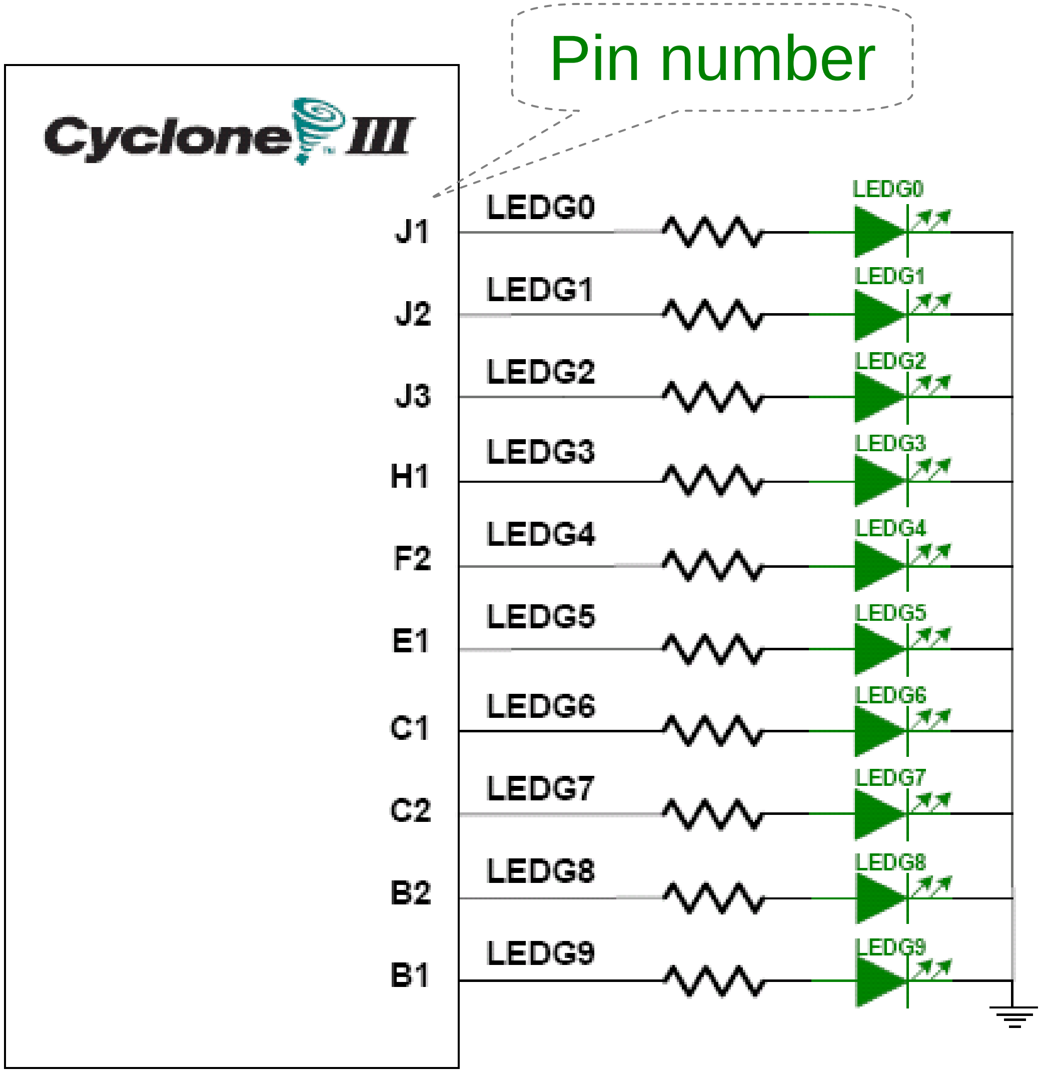
3 Pushbutton switches:
Not pressed → Logic High
Pressed → Logic Low

Signal Name	FPGA Pin No.
BUTTON [0]	PIN_ H2
BUTTON [1]	PIN_ G3
BUTTON [2]	PIN_ F1

10 Slide switches (Sliders):
Up → Logic High
Down → Logic Low

SW[0]	PIN_J6
SW[1]	PIN_H5
SW[2]	PIN_H6
SW[3]	PIN_G4
SW[4]	PIN_G5
SW[5]	PIN_J7
SW[6]	PIN_H7
SW[7]	PIN_E3
SW[8]	PIN_E4
SW[9]	PIN_D2

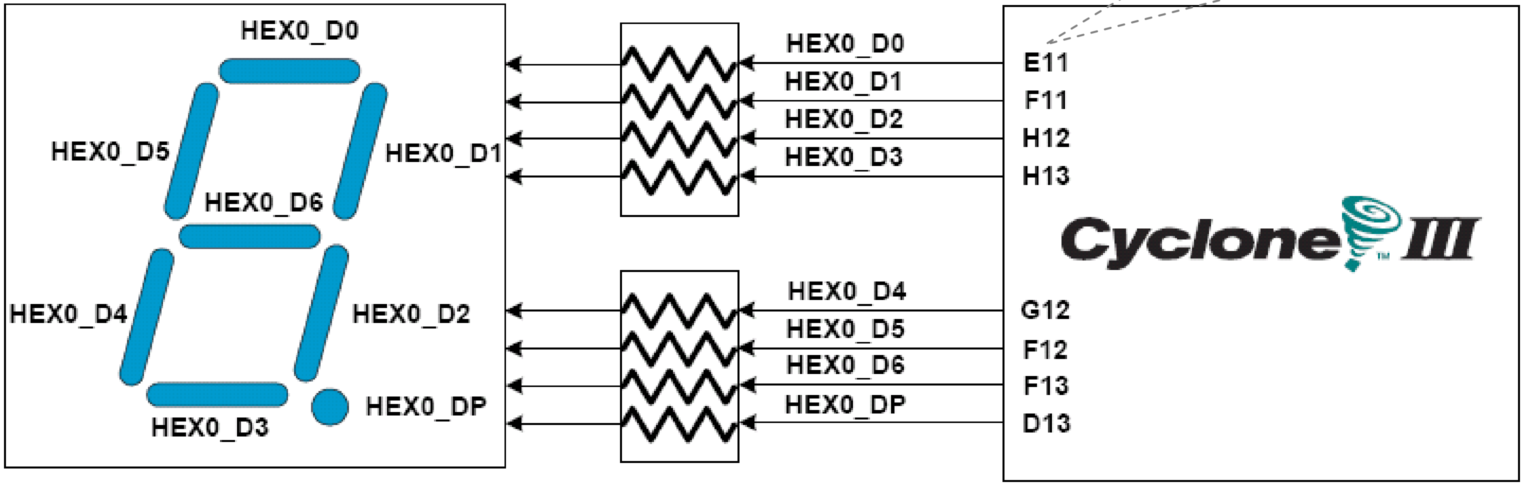
LEDs



10 LEDs
Opuput high → LED on
Output low → LED off

Signal Name	FPGA Pin No.
LEDG[0]	PIN_J1
LEDG[1]	PIN_J2
LEDG[2]	PIN_J3
LEDG[3]	PIN_H1
LEDG[4]	PIN_F2
LEDG[5]	PIN_E1
LEDG[6]	PIN_C1
LEDG[7]	PIN_C2
LEDG[8]	PIN_B2
LEDG[9]	PIN_B1

7-Segment Displays



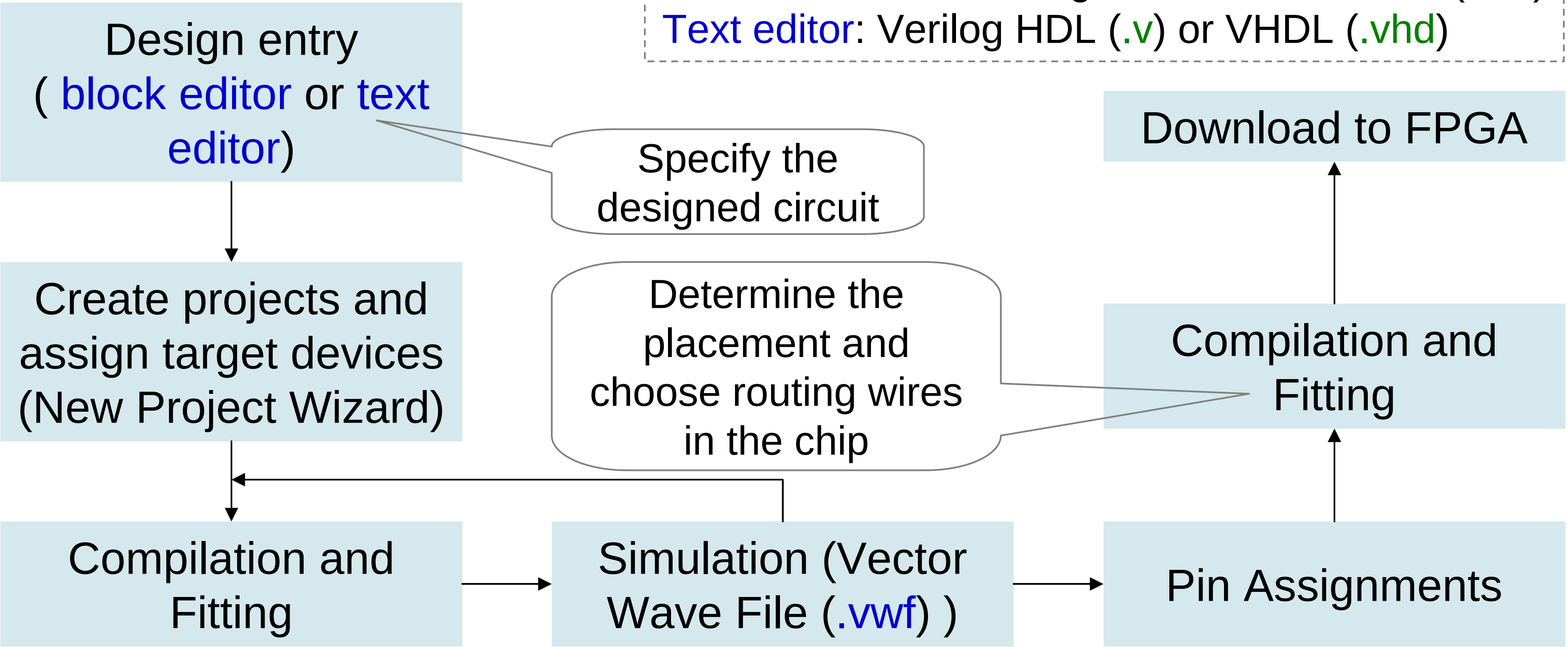
Pin number (active-low)

Signal Name	FPGA Pin No.
HEX0_D[0]	PIN_E11
HEX0_D[1]	PIN_F11
HEX0_D[2]	PIN_H12
HEX0_D[3]	PIN_H13
HEX0_D[4]	PIN_G12
HEX0_D[5]	PIN_F12
HEX0_D[6]	PIN_F13
HEX0_DP	PIN_D13
HEX1_D[0]	PIN_A13
HEX1_D[1]	PIN_B13
HEX1_D[2]	PIN_C13
HEX1_D[3]	PIN_A14
HEX1_D[4]	PIN_B14
HEX1_D[5]	PIN_E14
HEX1_D[6]	PIN_A15
HEX1_DP	PIN_B15
HEX2_D[0]	PIN_D15
HEX2_D[1]	PIN_A16
HEX2_D[2]	PIN_B16
HEX2_D[3]	PIN_E15
HEX2_D[4]	PIN_A17
HEX2_D[5]	PIN_B17
HEX2_D[6]	PIN_F14
HEX2_DP	PIN_A18
HEX3_D[0]	PIN_B18
HEX3_D[1]	PIN_F15
HEX3_D[2]	PIN_A19
HEX3_D[3]	PIN_B19
HEX3_D[4]	PIN_C19
HEX3_D[5]	PIN_D19
HEX3_D[6]	PIN_G15
HEX3_DP	PIN_G16

Quartus II 9.1 FPGA Design Software

Simplified Design Flow of Quartus II

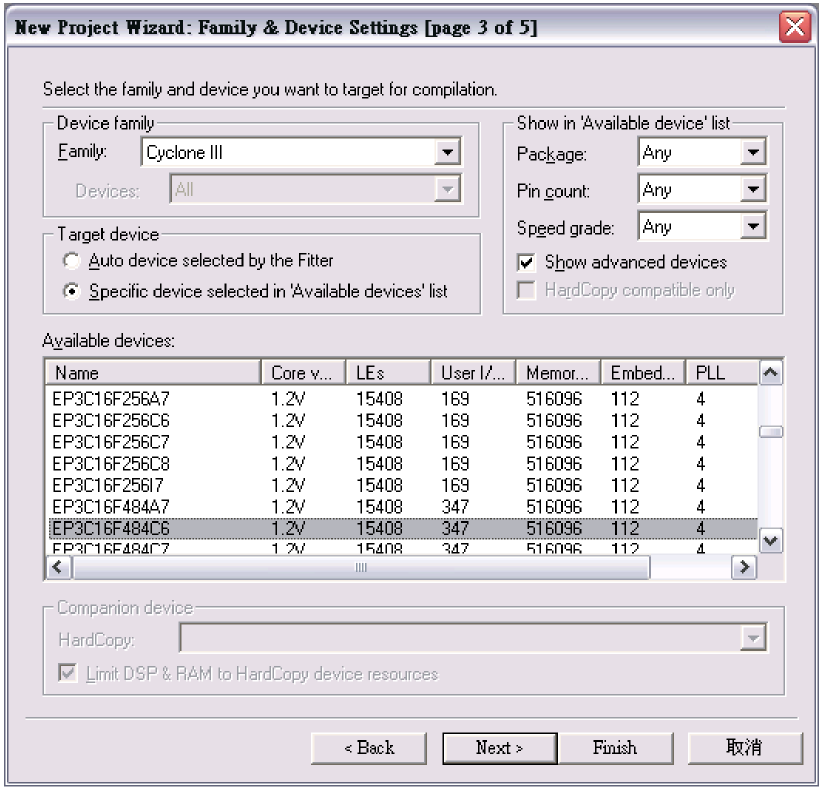
Block editor: Block Diagram/Schematic File (.bdf)
Text editor: Verilog HDL (.v) or VHDL (.vhd)


Design entry (block editor or text editor)
Specify the designed circuit
Create projects and assign target devices (New Project Wizard)
Compilation and Fitting
Simulation (Vector Wave File (.vwf))
Pin Assignments
Compilation and Fitting
Determine the placement and choose routing wires in the chip
Download to FPGA

Timing simulation: Verify functional correctness and timing issues.
Functional simulation: Verify functional correctness without considering timing issues.

Schematic Design with Quartus II

- Example: When the BUTTON0 is pressed, LEDG0 shows the ANDed result of SW0 and SW1 and LEDG1 shows the ORed result of SW0 and SW1.
- Step 1: Start a new project
  - Select File → New Project Wizard
    - Working directory: Class5
    - Project name: Class5
    - Top-level design entry: Class5
  - Family & Device Settings
    - Device family: Cyclone III
    - Available device: EP3C16F484C6
  - EDA Tool Settings
    - Leave it alone at the moment



New Project Wizard: Family & Device Settings [page 3 of 5]

Select the family and device you want to target for compilation.

Device family

Family: Cyclone III

Devices: All

Target device

○ Auto device selected by the Fitter

● Specific device selected in 'Available devices' list

Show in 'Available device' list

Package: Any

Pin count: Any

Speed grade: Any

☑ Show advanced devices

□ HardCopy compatible only

Available devices:

Name	Core v...	LEs	User I/...	Memor...	Embed...	PLL
EP3C16F256A7	1.2V	15408	169	516096	112	4
EP3C16F256C6	1.2V	15408	169	516096	112	4
EP3C16F256C7	1.2V	15408	169	516096	112	4
EP3C16F256C8	1.2V	15408	169	516096	112	4
EP3C16F256I7	1.2V	15408	169	516096	112	4
EP3C16F484A7	1.2V	15408	347	516096	112	4
EP3C16F484C6	1.2V	15408	347	516096	112	4
EP3C16F484C7	1.2V	15408	347	516096	112	4

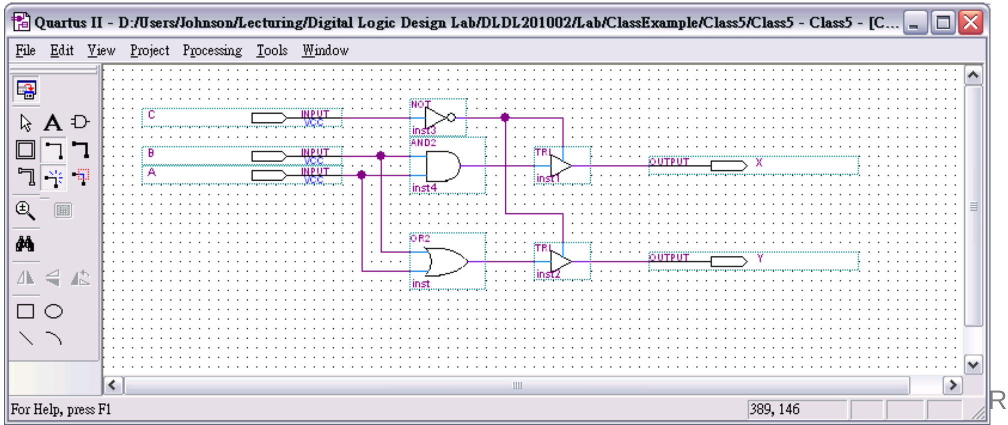
Companion device

HardCopy:

☑ Limit DSP & RAM to HardCopy device resources

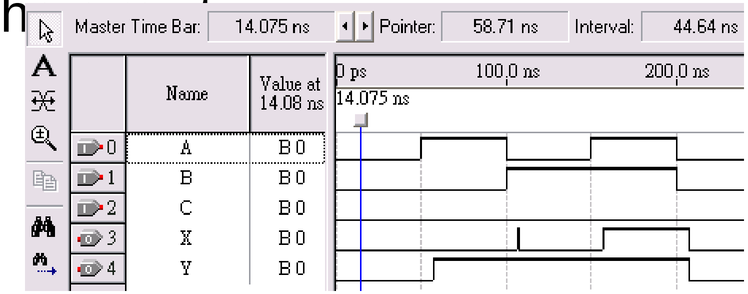
Schematic Design with Quartus II (Cont.)

- Step 2: Design entry using the graphic editor
  - Select File → New → Block Diagram/Schematic File (.bdf)
  - Save as "Class5.bsf" (check "Add file to current project")
  - Select "primitives" of "Symbol Tool" to add
    - Three input pins A, B, and C, two output pins X and Y
    - One AND gate, one OR gate, two tri-state buffers, and one NOT gate.
  - Select "Orthogonal Node Tool" to connect the nodes.
  - Select "Start Compilation" to compile the circuit



Schematic Design with Quartus II (Cont.)

- Step 3: Simulation with Vector Waveform File (.vwf)
  - Select File → New → Vector Waveform File (.vwf)
  - Save as "Class5.vwf" (check "Add file to current project")
  - Select "Edit → Insert → Insert Node or Bus → Node Finder" (Filter: Pins: all) to add input/output pins into the simulation.
  - Select "Edit → End Time" and select "Edit → Grid Size" to configure the simulation period to 500ns and count period.
    - A: count value, binary, start value 0, count every 50ns, multiplied by 1.
    - B: count value, binary, start value 1, count every 50ns, start time 20ns, multiplied by 2.
    - C: forcing high or forcing low.
  - Select "Start Simulation" to simulate th
  - Functional simulation
    - Select "Assignments → Settings → Simulator Settings" to set "Simulation mode" as Functional.
    - Select "Processing → Generate Functional Simulation Netlist"
    - Select "Start Simulation" to simulate the circuit.

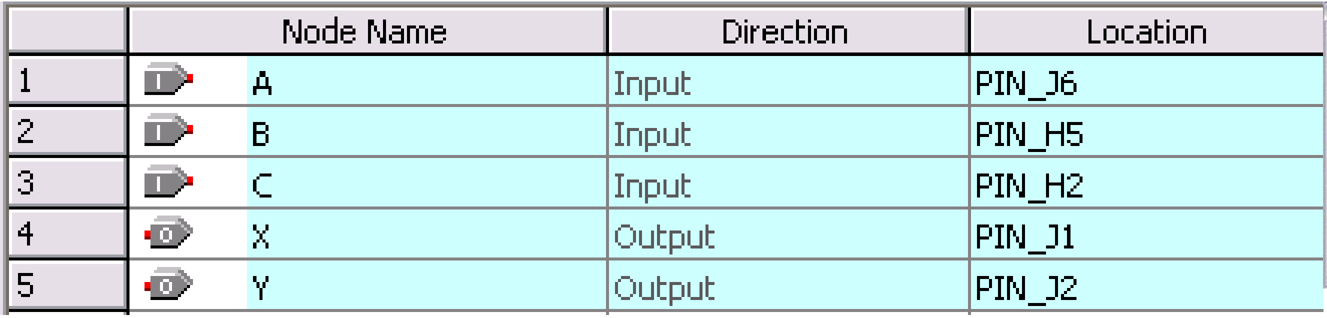


Master Time Bar: 14.075 ns Pointer: 58.71 ns Interval: 44.64 ns

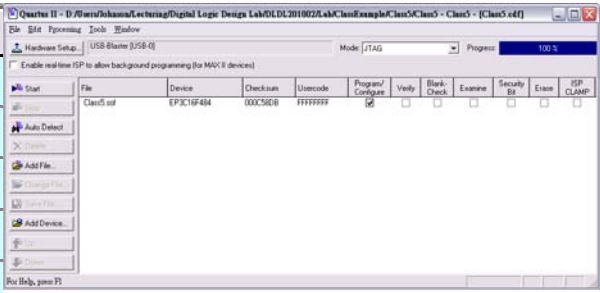
	Name	Value at 14.08 ns
0	A	B 0
1	B	B 0
2	C	B 0
3	X	B 0
4	Y	B 0

Schematic Design with Quartus II (Cont.)

- Step 3: Simulation with Vector Waveform File (.vwf)
  - Select “Assignments → Device” to configure the board settings.
    - Set Family as Cyclone III and Device as EP316F484C6
    - Select “Device and Pin Options”
      - Select and set “Unused Pins” as “As input tri-stated” and
      - Select “Configuration” to set configuration scheme as “Active Serial” and configuration device as “EPCS4”
  - Select “Assignments → Pins” to activate the “Pin Planner”.
  - Select “Start Compilation” to compile the circuit with circuit assignment.
  - Select “Tools → Programmer” to download the .soft file to the FPGA board for testing.



	Node Name	Direction	Location
1	A	Input	PIN_J6
2	B	Input	PIN_H5
3	C	Input	PIN_H2
4	X	Output	PIN_J1
5	Y	Output	PIN_J2



Quartus II - D:/Users/Johnson/Lecturing/Digital Logic Design Lab/DLDL201002/Lab/ClassExample/Class5/Class5 - Class5 - [Class5.cdf]

USB-Blaster [USB-0]   Mode: JTAG   Progress: 100 %

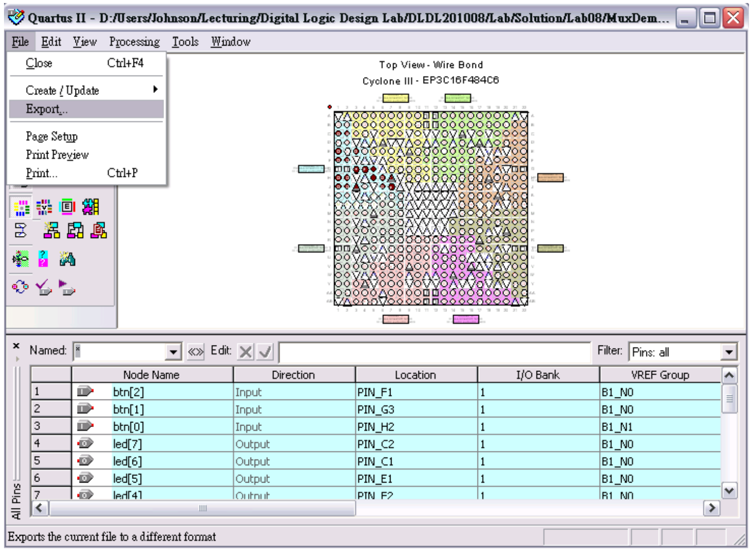
□ Enable real-time ISP to allow background programming (for MAX II devices)

File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank-Check	Examine	Security Bit	Erase	ISP CLAMP
Class5.sof	EP3C16F484	000C58DB	FFFFFFFF	☑	□	□	□	□	□	□

Pin Assignment Backup and Export

- Backup and export pin assignments
  - You can backup the ProjectName.pin file under your project directory.
  - You can export the pin assignment by exporting it as an .csv file.
    - 1. Under “Pin Planner”, Select File → Export
    - 2. Use EXCEL or any text editor to open this file.

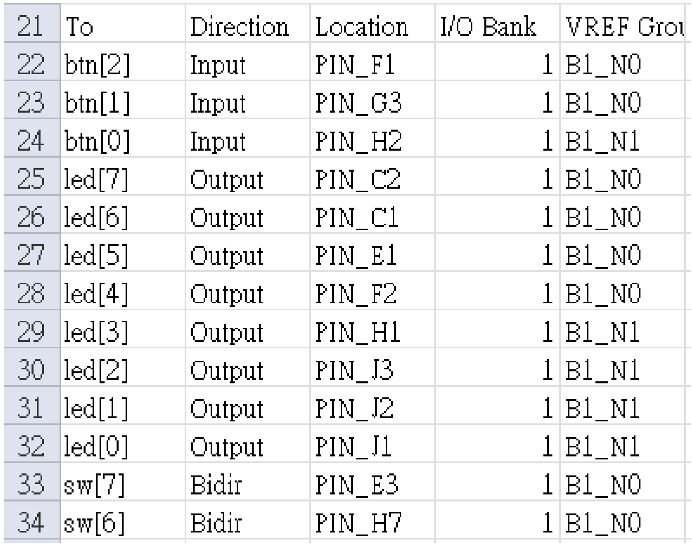
1



		Node Name	Direction	Location	I/O Bank	VREF Group
1		btn[2]	Input	PIN_F1	1	B1_N0
2		btn[1]	Input	PIN_G3	1	B1_N0
3		btn[0]	Input	PIN_H2	1	B1_N1
4		led[7]	Output	PIN_C2	1	B1_N0
5		led[6]	Output	PIN_C1	1	B1_N0
6		led[5]	Output	PIN_E1	1	B1_N0
7		led[4]	Output	PIN_F2	1	B1_N0

Exports the current file to a different format

2

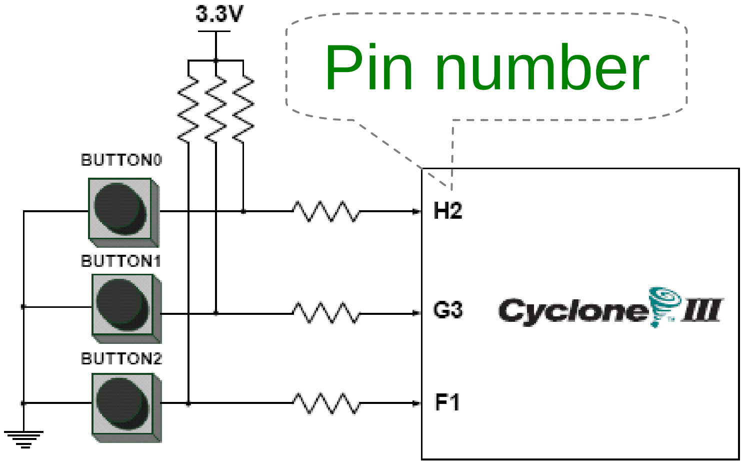


21	To	Direction	Location	I/O Bank	VREF Grou
22	btn[2]	Input	PIN_F1	1	B1_N0
23	btn[1]	Input	PIN_G3	1	B1_N0
24	btn[0]	Input	PIN_H2	1	B1_N1
25	led[7]	Output	PIN_C2	1	B1_N0
26	led[6]	Output	PIN_C1	1	B1_N0
27	led[5]	Output	PIN_E1	1	B1_N0
28	led[4]	Output	PIN_F2	1	B1_N0
29	led[3]	Output	PIN_H1	1	B1_N1
30	led[2]	Output	PIN_J3	1	B1_N1
31	led[1]	Output	PIN_J2	1	B1_N1
32	led[0]	Output	PIN_J1	1	B1_N1
33	sw[7]	Bidir	PIN_E3	1	B1_N0
34	sw[6]	Bidir	PIN_H7	1	B1_N0

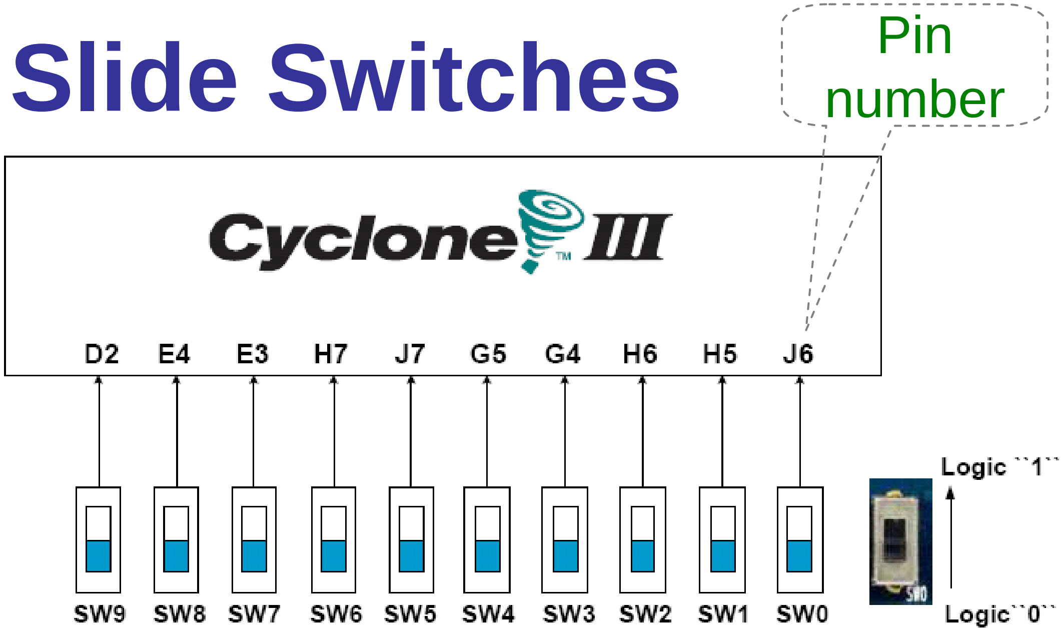
Lab 5

- Part 1 - Simulation
  - Use the block editor (Block Diagram/Schematic File: .bdf) to design a NAND gate with one output pin F and two input pin A and B. Then use Vector Waveform File (.vwf) to simulate the results.
    - A: count value, binary, start value 1, simulation period=4us, advanced by 1 every 100ns
    - B: count value, binary, start value 0, simulation period=4us, start time: 20ns, advanced by 1 every 200ns
  - Map A to SW0, B to SW1, and F to LED0 of DE0, and program it.
- Part 2 - Transferring a Design to a Target FPGA
  - Use three slides (SW1-SW0) as the binary input value.
    - The corresponding LED (LEDG0-3) is on when it is selected by the binary input. Other LEDs are off. E.g., 10 (SW1-SW0) lights LEDG2.

Pushbutton and Slide Switches



Pin number



Pin number

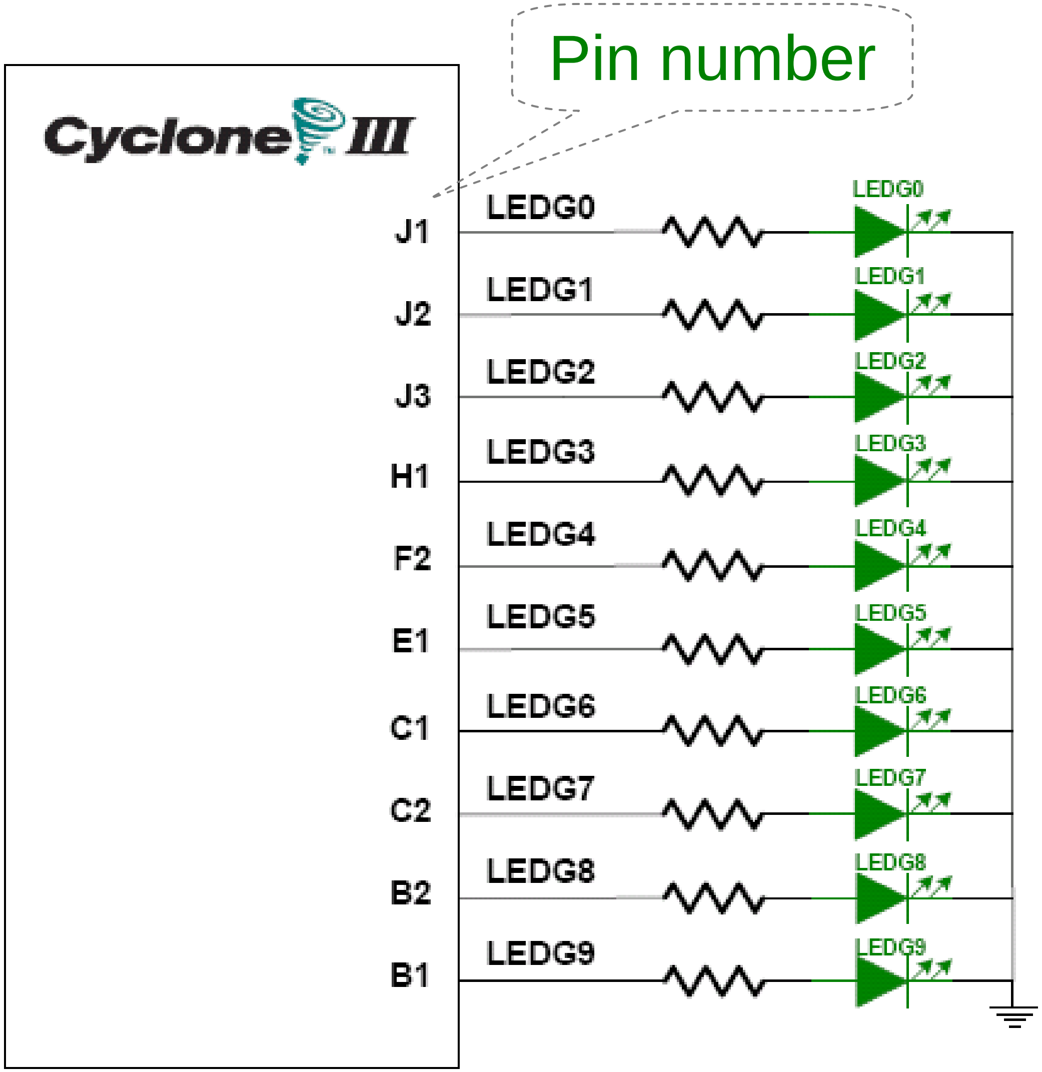
3 Pushbutton switches:
Not pressed → Logic High
Pressed → Logic Low

10 Slide switches (Sliders):
Up → Logic High
Down → Logic

Signal Name	FPGA Pin No.
BUTTON [0]	PIN_ H2
BUTTON [1]	PIN_ G3
BUTTON [2]	PIN_ F1

SW[0]	PIN_J6	SW[5]	PIN_J7
SW[1]	PIN_H5	SW[6]	PIN_H7
SW[2]	PIN_H6	SW[7]	PIN_E3
SW[3]	PIN_G4	SW[8]	PIN_E4
SW[4]	PIN_G5	SW[9]	PIN_D2

LEDs



10 LEDs
Opuput high → LED on
Output low → LED off

Signal Name	FPGA Pin No.
LEDG[0]	PIN_J1
LEDG[1]	PIN_J2
LEDG[2]	PIN_J3
LEDG[3]	PIN_H1
LEDG[4]	PIN_F2
LEDG[5]	PIN_E1
LEDG[6]	PIN_C1
LEDG[7]	PIN_C2
LEDG[8]	PIN_B2
LEDG[9]	PIN_B1