

## **Quartus II Basic Training**

## **Programmable Logic Families**

#### Structured ASIC

- HardCopy<sup>®</sup> II, HardCopy Stratix
- High & Medium Density FPGAs
  - Stratix II, Stratix, APEX<sup>™</sup> II, APEX
     20K, & FLEX 10K<sup>®</sup>
- Low-Cost FPGAs
  - Cyclone II & Cyclone
- FPGAs with Clock Data Recovery
  - Stratix II GX
- CPLDs
  - MAX II, MAX 7000 & MAX 3000
- Embedded Processor Solutions
  - Nios II
- Configuration Devices
  - Serial (EPCS) & Enhanced (EPC)





**Cyclone** *II* 





## MAX 7000A & MAX 3000A Family Overview

Parameter		MAX 3000A					M	AX 70	<b>A00</b>	
	EPM3032A	EPM3064A	EPM3128A	EPM3256A	EPM3512A	EPM7032AE	EPM7064AE	EPM7128AE	EPM7256AE	EPM7512AE
Useable Gates	600	1,250	2,500	5,000	10,000	600	1,250	2,500	5,000	10,000
Macrocells	32	64	128	256	512	32	64	128	256	512
Maximum User I/O Pins	34	66	96	158	208	36	68	100	164	212
t <sub>PD</sub> (ns)	4.5	4.5	5.0	7.5	7.5	4.5	4.5	5.0	5.5	7.5
f <sub>CNT</sub> (MHz)	227	222	192	127	116	227	222	192	172	116
t <sub>su</sub> (ns)	2.9	2.8	3.3	5.2	5.6	2.9	2.8	3.3	3.9	5.6
t <sub>CO1</sub> (ns)	3.0	3.1	3.4	4.8	4.7	3.0	3.1	3.4	3.5	4.7



# Complete Voltage Portfolio5.0 V3.3 V2.5 V

#### **MAX 7000S**

- Performance Leader
- Feature Leader
- Wide Range of Package Offerings
- Industrial-Grade Offerings

#### **MAX 7000AE**

- High Performance
- Feature Leader
- Wide Range of Package Offerings

#### MAX 7000B

- High Performance
- Feature Leader
- Wide Range of Package Offerings

#### MAX 3000A

- Price Leader
- Feature & Package Subset of MAX 7000AE





### **MAX Device Block Diagram**





### MAX II: The Lowest-Cost CPLD Ever

#### New Logic Architecture

- 1/2 the Cost
- 1/10 the Power Consumption
- 2X the Performance
- 4X the Density
- Non-Volatile, Instant-On
- Supports 3.3-, 2.5- & 1.8-V Supply Voltages



#### Breakthrough Technology to Expand the Market



## **Flexible Supply Voltage**

#### On-Chip Voltage Regulator

- Accepts 3.3-, 2.5- & 1.8-V Supply Inputs
- Internally Converted to 1.8-V Core Voltage



#### Convenience of 3.3 V with the Power & Performance of 1.8 V



## **MAX II Device Family**

Device	Logic Elements (LEs)	Typical Macro- cells	User I/O Pins	Speed Grades	Fastest t <sub>pd1</sub> (ns)	User Flash Memory (bits)
EPM240	240	192	80	3, 4, 5	4.7	8,192
EPM570	570	440	160	3, 4, 5	5.5	8,192
EPM1270	1,270	980	212	3, 4, 5	6.3	8,192
EPM2210	2,210	1,700	272	3, 4, 5	7.1	8,192



Ē



Device	100-Pin TQFP <sup>1</sup> 0.5-mm Pitch 16 x 16 mm	144-Pin TQFP 0.5-mm Pitch 22 x 22 mm	256-Pin FBGA <sup>2</sup> 1.0-mm Pitch 17 x 17 mm	324-Pin FBGA 1.0-mm Pitch 19 x 19 mm
EPM240	80			
EPM570	76	116	160	
EPM1270		116	212	
EPM2210			204	272



## **New Small Packages**



Packages minimize PCB area and optimize ease-of-use

- Partial arrays allow for 2 layer PCB break out



### **MAX II Architecture**



## **MAX II Logic Element (LE)**





## **User Flash Memory**

#### Feature

- Flash Memory Storage Bank
- 8,192 Bits Per Device
- Interface to SPI, I<sup>2</sup>C, Parallel, or Proprietary Buses
- Applications
  - Store Revision & Serial Number Data
  - Store Boot-Up & Configuration Data





### **MAX & MAX II Comparison**

Parameter	MAX	MAX II
Process Technology	0.3-um EEPROM	0.18-um Flash
Logic Architecture	Product Term	Look-Up Table (LUT)
Density Range	32 to 512 Macrocells	128 to 2210 Macrocells
		(240 to 2,210 LEs)
Routing Architecture	Global	Row & Column
<b>On-Chip Flash Memory</b>	None	8 Kbits
Maximum User I/O Pins	212	272
Supply Voltage	5.0 V, 3.3 V, 2.5 V	3.3 V/2.5 V, 1.8 V
I/O Voltages	5.0 V, 3.3 V, 2.5 V, 1.8 V	3.3 V, 2.5 V, 1.8 V, 1.5 V
Global Clock Networks	2 per Device	4 per Device
Output Enables (OEs)	6 to 10 per Device	1 per I/O Pin
Schmitt Triggers	None	1 per I/O Pin



### What is Nios II?

Altera's Second Generation Soft-Core 32 Bit RISC Microprocessor

- Nios II Plus All Peripherals Written In HDL
- Can Be Targeted For All Altera FPGAs
- Synthesis Using Quartus II Integrated Synthesis



## **Nios II Processor Architecture**

#### Classic Pipelined RISC Machine

- 32 General Purpose Registers
- 3 Instruction Formats
- 32-Bit Instructions
- 32-Bit Data Path
- Flat Register File
- Separate Instruction and Data Cache (configurable sizes)
- Tightly-Coupled Memory Options
- Branch Prediction
- 32 Prioritized Interrupts
- On-Chip Hardware (Multiply, Shift, Rotate)
- Custom Instructions
- JTAG-Based Hardware Debug Unit





#### System On A Programmable Chip (SOPC) Power



#### CPU is a Critical Control Function Required for System-Level Integration



## Licensing

#### Nios II Delivered As Encrypted Megacore

- Licensed Via Feature Line In Existing Quartus II License File
- Consistent With General Altera Megacore Delivery Mechanism
- Enables Detection Of Nios II In Customer Designs (Talkback)

#### No Nios II Feature Line (OpenCore Plus Mode)

- System Runs If Tethered To Host PC
- System Times Out If Disconnected from PC After ~ 1 hr

#### Nios II Feature Line (Active Subscriber)

- Subscription and New Dev Kit Customers Obtain Licenses From <u>www.altera.com</u>
- Nios II CPU RTL Remains Encrypted

#### Nios II Source License

- Available Upon Request On Case-By-Case Basis
- Included With Purchase Of Nios II ASIC License





### **Quartus II Basic Training**

Quartus II Development System Feature Overview

## **Software & Development Tools**



- All Stratix, Cyclone & Hardcopy Devices
- APEX II, APEX 20K/E/C, Excalibur, & Mercury Devices
- FLEX 10K/A/E, ACEX 1K, FLEX 6000 Devices
- MAX II, MAX 7000S/AE/B, MAX 3000A Devices

#### Quartus II Web Edition

- Free Version
- Not All Features & Devices Included
  - See <u>www.altera.com</u> for Feature Comparison







**ARTUS®II** 

## **Quartus II Development System**

#### Fully-Integrated Design Tool

- Multiple Design Entry Methods
- Logic Synthesis
- Place & Route
- Simulation
- Timing & Power Analysis
- Device Programming



## **Typical PLD Design Flow**



## **Typical PLD Design Flow**



**Timing Analysis** 

- Verify Performance Specifications Were Met
- Static Timing Analysis

#### **Gate Level Simulation**

- Timing Simulation
- Verify Design Will Work in Target Technology



#### **PC Board Simulation & Test**

- Simulate Board Design
- Program & Test Device on Board
- Use SignalTap II for Debugging



## **Design Entry Methods**



- 3rd-Party EDA Tools
  - EDIF
  - HDL
  - VQM

#### Mixing & Matching Design Files Allowed





#### Quartus II Basic Training Quartus II Quick Start LAB1

## **Objectives**

- Create a project using the New Project Wizard
- Name the project
- Add design files
- Pick a device



## Step 1 (Setup Project for QII5\_1)

Under File, Select **New Project Wizard**.... A new window appears. If an Introduction screen appears, click Next.



#### New Project Wizard: Introduction The New Project Wizard helps you create a new project and preliminary project settings, including the following:

- Project name and directory
- Name of the top-level design entity
- Project files and libraries
- Target device family and device
- EDA tool settings

You can change the settings for an existing project and specify additional project-wide settings with the Settings command (Assignments menu). You can use the various pages of the Settings dialog box to add functionality to the project.

Don't show me this in Don't show me this Don't show me this Don't show me this Don't show me this Don't sho	ntroduction again		



## Step 2 (Setup Project for QII5\_1)

#### Page 1 of the wizard should be completed with the following

working directory for this project	<lab_install_directory> \Dsp_7_segment\</lab_install_directory>
name of project	Dsp_7_segment
top-level design entity	Dsp_7_segment

Ne	w Project Wizard: Directory, Name, Top-Level Entity [page 1 of 5]		×
	What is the working directory for this project?		
	D:\Dsp_7_segment		
	What is the name of this project?		
	Dsp_7_segment		
	What is the name of the top-level design entity for this project? This name is case ser and must exactly match the entity name in the design file.	isitive	
	Dsp_7_segment		
	<u>U</u> se Existing Project Settings		
_			
	< Back Next > Finish	収	211

## Copy "state\_machine.v" and past in Dsp\_7\_segment

#### Click Next to advance to the Project Wizard: Add Files [page 2 of 5].



## Step 3 (Setup Project for QII5\_1)

Using the browse button, select **state\_machine.v Add** to the project. Click **Next**.

ew Project Wizard: Add Files [page 2 of 5] Select the design files you want to include in the proje in the project directory to the project. Note: you can al later. <u>F</u> ile name:	ct. Click Add All to a ways add design file	add all design files es to the project
File name state_machine.v	Type Verilog H	Add All <u>R</u> emove <u>Properties</u> <u>Up</u> <u>D</u> own
Specify the path names of any non-default libraries.	User Libraries Next > Fi	



## Step 4 (Setup Project for QII5\_1)

On page 3, select Stratix as the Family. Also, in the Filters section, set Package to FBFA, Pin count to 780, and Speed grade to 5. Select the EP1S25F780C5 device from the Available devices: window. Click Next.

Select the family and device you want to target for compilation.         Eamily:       Stratix         Target device <ul> <li>Auto device selected by the Fitter from the 'Available devices' list</li> <li>Specific device selected in 'Available devices' list</li> </ul> Available devices: <ul> <li>Filters</li> <li>Package:</li> <li>Any</li> <li>Package:</li> <li>Any</li> <li>Pin gount:</li> <li>Any</li> <li>Speed grade:</li> <li< th=""><th>New Project Wizard: Family &amp; Devic</th><th>e Settings [page 3 of 5</th><th>5]</th><th></th><th>×</th></li<></ul>	New Project Wizard: Family & Devic	e Settings [page 3 of 5	5]		×
Target device         Auto device selected by the Fitter from the 'Available devices' list         Specific device selected in 'Available devices' list         Available devices:         EP1S25B672C7         EP1S25F672C6         EP1S25F672C7         EP1S25F672C7         EP1S25F672C7         EP1S25F672C7         EP1S25F672C8         EP1S25F780C6         EP1S25F780C6         EP1S25F780C6         EP1S25F780C6         EP1S25F780C6         EP1S25F780C6         EP1S25F700C7         EP1S25F700C6         EP1S25F700C7         EP1S25F1020C6         EP1S25F1020C6         EP1S25F1020C6         EP1S25F1020C6         EP1S25F1020C7         EP1S25F1020C6         EP1S25F1020C6         EP1S25F1020C7         EP1S25F1020C6         EP1S25F1020C7         EP1S25F1020C7         EP1S25F1020C7         EP1S25F1020C7         EP1S25F1020C7         EP1S25F1020C7         EP1S25F1020C7         EP1S25F1020C6         EP1S25F1020C7         EP1S25F1020C7         EP1S25F1020C7         EP1S25F1020C	Eamily: Stratix	want to target for cor	mpilation.		-
Available devices:          EP1S258672C7         EP1S25F672C6         EP1S25F672C7         EP1S25F672C7         EP1S25F672C8         EP1S25F672C8         EP1S25F780C5         EP1S25F780C6         EP1S25F780C6         EP1S25F780C6         EP1S25F780C6         EP1S25F780C6         EP1S25F780C7         EP1S25F700C6         EP1S25F1020C5         EP1S25F1020C6         EP1S25F1020C6         EP1S25F1020C6         EP1S25F1020C7         EP1S25F1020C7         EP1S25F1020C7         EP1S25F1020C7         EP1S25F1020C7         EP1S25F1020C7	Target device C <u>A</u> uto device selected by th C <u>S</u> pecific device selected in	ne Fitter from the 'Avai n 'Available devices' li	ilable devices' list ist	:	
EP1S25F672C6         EP1S25F672C7         EP1S25F672C8         EP1S25F672C8         EP1S25F780C6         EP1S25F78016         EP1S25F78016         EP1S25F78016         EP1S25F78016         EP1S25F78016         EP1S25F78007         EP1S25F102005         EP1S25F102006         EP1S25F102006         EP1S25F102007         FP1S25F102007         EP1S25F102007         EP1S25F102007         EP1S25F102007         EP1S25F102006         EP1S25F102007         EP1S25F102007         EP1S25F102007         EP1S25F102007	Available <u>d</u> evices: EP1S25B672C7		- Filters		]
EP1320F072C0         EP1320F072C0         EP1325F780C5         EP1325F780C6         EP1325F780C7         EP1325F1020C5         EP1325F1020C6         EP1325F1020C6         EP1325F1020C7	EP1S25F672C6 EP1S25F672C7 EP1S25F672I7 EP1S25F672I7		<u>P</u> ackage: Pin <u>c</u> ount:	Any  Any	
EP1S25F780C7 EP1S25F1020C5 EP1S25F1020C6 EP1S25F1020I6 EP1S25F1020I6 EP1S25F1020C7 EP1S25F1020C7 EP1S25F22C6 HABDCOPY EPSA PB0T0	EP1525F780C5 EP1525F780C6 EP1525F780C6 EP1525F78016		Speed grade: Core voltage:	Any   1.5V	
EP1S25F102016 EP1S25F1020C7 EP1S25F572C6_HABDCOPY_EPGA_PB0T0	EP1S25F780C7 EP1S25F1020C5 EP1S25F1020C6		Show Advar	nced Devices	]
EPIS25F672C7_HARDCOPY_FPGA_PROTO	EP1S25F102016 EP1S25F1020C7 EP1S25F672C6_HARDC0PY_I EP1S25F672C7_HARDC0PY_I	FPGA_PROTO	HardCopy II: Limit DSP & device reso	RAM to HardCopy II	
< Back Next > Finish 取消		< Back	Next >	Finish 】 取消	_



## Step 5 (Setup Project for QII5\_1)

On page 4, you can specify any third party EDA tools you may be using along with Quartus II. Since these exercises will be done entirely within Quartus II, click **Next.** 

New Project Wizard: EDA Tool Se	ttings [page 4 of 5]	×
Specify the other EDA tools	in addition to the Quartus II software used with the project.	
EDA design entry / synthesis tool:	Not available	
EDA simulation tool:	Not available	
EDA timing analysis tool:	Not available	
	< Back Next > Finish 取消	



## Step 6 (Setup Project for QII5\_1)

The summary screen appears as shown. Click **Finish.** The project is now created.

New Project Wizard: Summary [page	5 of 5]	×
When you click Finish, the project	ct will be created with the following settings:	
Project directory: D: 応大Training/sam_test/F	)sp. 7. segment/	
Project name:	Dsp 7 segment	
Top-level design entity:	Dsp 7 segment	
Number of files added:	1	
Number of user libraries added:	0	
Device assignments:		
Family name:	Stratix	
Device:	EP1S25F780C6	
EDA tools:		
Design entry/synthesis:	<none></none>	
Simulation:	<none></none>	
Timing analysis:	<none></none>	
		-
	< Back Next > Finish 取消	





### Quartus II Basic Training Quartus II Quick Start LAB2

## **Objectives**

- Create a counter using the MegaWizard Plug-in Manager
- Build a design using the schematic editor
- Analyze and elaborate the design to check for errors



### **Step 1** Create schematic file

Select File  $\Rightarrow$  New and select Block Diagram/Schematic File. Click OK. Select File  $\Rightarrow$  Save As and save the file as

<lab\_install\_directory> \Dsp\_7\_segment\ Dsp\_7\_segment.bdf

New	×
Device Design Files   Software Files   Other Files	
AHDL File	- 11
Block Diagram/Schematic File	
EDIF File	
Verilog HDL File	
VHDL File	
,	
	<u>a</u>
	,61



### Step 2 Build an 23 bits counter using the MegaWizard Plug-in Manager

Choose Tools ⇒ MegaWizard Plug-In Manager. In the window that appears, select Create a new custom megafunction variation. Click on Next.
 On page 2a of the MegaWizard expand the arithmetic folder and select LPM\_COUNTER.

#### 3.Choose Verilog HDL output For the name of the output file, type timer\_1s. Click on Next







- 1. Set the output bus to **27** bits. For the remaining settings in this window, use the defaults that appear .. Select **next**
- 2. .Turn on "Modulus , with a count modulus of "and key in 79999999
- 3. Select finish

MegaWizard Plug-In Manager - LPM_COUNTER [page 3 of 6]	MasaWin Phys-In Manager - LPM COIINTER Inage 4 of 61	X
How wide should the 'q' output bus be? 27 • bits what should the counter direction be? • Up only • Down only • Create an 'updown' input port to allow me to do both 11 counts up; 0 counts down)	MegaWL At Plug-In Manager - LPM_COUNTER [page 4 of 6]         Which type of counter do you want?         Plain binary         Plain binary         Clock modulus 79999999         g[26.0]         Do you want any optional additional ports?         Clock Enable       Carry-in         Count Enable       Carry-out	X
Documentation         Cancel         < Back         Next >         Finish	Resource Estimate     Documentation     Cancel     < Back     Next >	<u> </u>







In the **Graphic Editor**, **double-click** in the screen so that the **Symbol** Window appears. Inside the symbol window, **click** on **+** to expand the symbols defined in the **Project** folder. **Double-click** on **timer\_1s. Click** the **left mouse button** to put down the symbol inside the schematic file.. *The symbol for "timer\_1s" now appears in the schematic.* 







- 1. From the **File** menu, **open** the file **state\_machine.v**
- From the File menu, go the Create/Update menu option and select Create Symbol Files for Current File. Click Yes to save changes to Dsp\_7\_segment.bdf.
- 3. Once Quartus II is finished creating the symbol, click OK. Close the state\_machine.v file
- 4. In the **Graphic Editor**, **double-click** in the screen so that the **Symbol** Window appears again. **Double-click** on **state\_machine** in the **Project** folder. **Click OK...** *The symbol for* **state\_machine** *now appears in the schematic.*



## **Step 6 Add Pins to the Design**

Input	Output					
sys_clk	7_out[60]					
reset	Dig1					

For each of the pins listed in left Table , you must insert a pin and change its name



- 1. To place pins in the schematic file, go to  $Edit \Rightarrow Insert \Rightarrow Symbol OR$ double-click in any empty location of the Graphic Editor.
- 2. Browse to **libraries** ⇒ **primitives** ⇒ **pin** folder. **Double-click** on **input** or **output** *Hint: To insert multiple pins select* **Repeat Insert Mode**.
- 3. To rename the pins double-click on the pin name after it has been inserted.
- 4. Type the name in the Pin name(s) field and Click OK



#### **Step 7** Connect the Pins and Blocks in the Schematic

- 1. In the left hand tool bar click on button to draw a wire and button to draw a bus. <u>Another</u> way to draw wires and busses is to place the cursor next to the port of any symbol. When you do this, the wire or bus tool will automatically appear.
- 2. Connect all of the pins and blocks as shown in the figure below





### **Step 8 Save and check the schematic**

- 1. Click on the **Save** button in the toolbar **I** to save the schematic.
- From the Project menu, select Add/Remove Files in Project.
   Click on the browse button to make sure the Dsp\_7\_segment.bdf, timer\_1s and state\_machine are added to the project.
- 3. From the **Processing** menu, select **Start**  $\Rightarrow$  **Start Analysis & Elaboration**.

Analysis and elaboration checks that all the design files are present and connections have been made correctly.

4. Click **OK** when analysis and elaboration is completed





### Quartus II Basic Training Quartus II Quick Start LAB3

## **Objectives**

#### Pin assignment

- Perform full compilation Build a design using the schematic editor
- How to Download programming file





- 1. Choose **Assignments**  $\Rightarrow$  Assignment editor.
- 2. From the View menu, select Show All Know Pin Names.
- 3. Please click **Pin** in **Category**

	То	Location	I/O Bank	I/O Standard	General Function	Spi
1	🗇 7_out			LVTTL		
2	💿 7_out[0]			LVTTL		
3	7_out[1]			LVTTL		
4	7_out[2]			LVTTL		
5	💿 7_out[3]			LVTTL		
6	🐼 7_out[4]			LVTTL		
7	@7_out[5]			LVTTL		
8	@7_out[6]			LVTTL		
9	iiiPreset			LVTTL		
10	iii ∎sys_clk			LVTTL		
11	< <new>&gt;</new>	< <new>&gt;</new>				





- 1. Pls install DSP Development Kit Stratix editon CD
- 2. Open ds\_stratix\_dsp\_bd.pdf from C:\megacore\stratix\_dsp\_kit-v1.1.0\Doc
- Check clk , pushbotton and seven segment display pin location from ds\_stratix\_dsp\_bd.pdf
- 4. Key your pin number in location
- 5. Click on the **Save** button in the toolbar
- 6. From Assignments, select Device. Click Device & Pin options. Click Unused pins .Select As input tri-stated from Reserve all unused pins
- 7. From the Processing menu. select Start Compilation

То	Location
💿7_out[0]	PIN_L18
7_out[1]	PIN_D24
💿 7_out[2]	PIN_L23
💿 7_out[3]	PIN_L24
7_out[4]	PIN_L22
7_out[5]	PIN_L20
💿 7_out[6]	PIN_L19
🖻 reset	PIN_F24
iii ∎>sys_clk	PIN_K17
🐻 7_out	
	To 7_out[0] 7_out[1] 7_out[2] 7_out[3] 7_out[3] 7_out[4] 7_out[5] 7_out[6] reset sys_clk 7_out





- 1. From the **Tools** menu, select **programmer**
- 2. Click on Add File. Select Dsp\_7\_segment.sof.
- 3. Check Hardware Setup. Select your download cable on Currently selected hardware(ByteBlasterII)
- 4. Select **JTAG** from **Mode**

🛓 🔔 Hardware Setup.	USB-Blaster [USB-0]				
Mode:	JTAG				
Progress:			100 %		
Enable real-time IS	SP to allow background progra	mming (for MAX II d	evices)		
🏓 Start	File	Device	Checksum	Usercode	Program/ Configure Ve
🖬 Stop	Dsp_7_segment.sof	EP1S25F780	002BC2E7	FFFFFFF	
\mu Auto Detect					
🗙 Delete					
🍰 Add File					
👺 Change File					
🗳 Save File					
😂 Add Device					
📫 Up					

urdware Kup Hardware Settings   JTAG Settin Select a programming hardware	gs   e setup to use whe	en programming de	≥ evices. This programming
Currently selected hardware:	USB-Blaster [L	ISB-0]	<b>_</b>
Hardware USB-Blaster ByteBlasterII	Server Local Local	Port USB-0 LPT1	Add Hardware Remove Hardware
			Close





- 1. Turn on **Program/configure. Or see figure below**
- 2. Click Start
- 3. See 7-segment status

🔔 Hardware Setup	USB-Blaster [USB-0]					Мо	de: JTAG	i	•	Progre	ss:	0%
Enable real-time ISP to allow background programming (for MAX II devices)												
Mart Start	File	Device	Checksum	Usercode	Program/ Configure	Verify	Blank- Check	Examine	Security Bit	Erase	ISP CLAMP	
🖬 Stop	Dsp_7_segment.sof	EP1S25F780	00288067	FFFFFFF								
Auto Detect												
🗙 Delete												
📛 Add File												
👺 Change File												
🕒 Save File												
😂 Add Device												
📲 Up												
🔑 Down												

