University Program 3 Kit

VLSI Tutorial : LEDs & Push Buttons

Version 02.00

System Level Solutions Inc. (USA)

14702 White Cloud Ct.
Morgan Hill, CA 95037

About this tutorial

This tutorial provides information how to use Quartus II software with UP3.

How to find the information

- The Adobe Acrobat Find feature allows you to search the contents of a PDF file. Use Ctrl + F to open the Find dialog box. Use Ctrl + N to open to the Go To Page dialog box.
- Thumbnail icons, which provide miniature preview of each page, provide a link to the pages.
- Links allow you to jump to related information.

How to contact SLS

For the most up-to-date information about SLS products, go to the SLS worldwide website at http://www.slscorp.com.

TABLE 1. Contact Information

Information Type	E-mail
Product literature services, SLS literature services, Non-technical customer services, Technical support.	support@slscorp.com

CONTENTS

Introduction

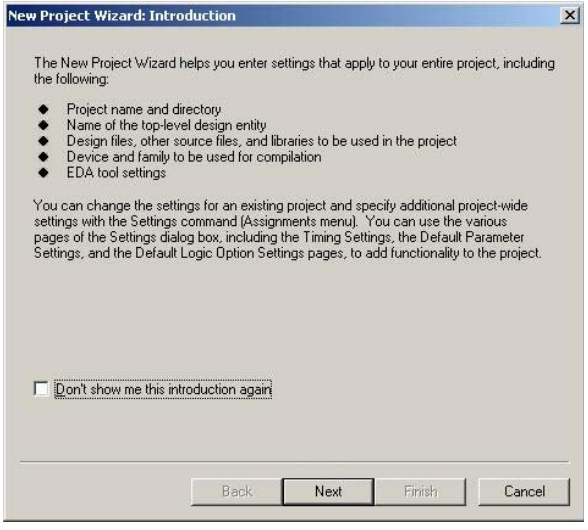
This tutorial will guide through the basic features of the Quartus II software. It explains how to design, compile, simulate and program your logic designs in the Quartus II software. The four tutorial modules - Design Entry, Compilation, Simulation and Programming - combine to show you all the steps necessary to create and process the sample design project Dip_PB_Led using Push Button, Dip Switch and LEDs on the UP3 (Universtiy Program 3 Kit).

Design Entry

The Quartus II software offers a New Project wizard to help you create a new project. You can subsequently edit these project settings using menu commands and dialog boxes. To create a new project using the New Project Wizard, follow these steps:

1. Choose New Project Wizard (File menu). The New Project Wizard appears. Opening the New Project Wizard for the first time may display the Introduction page; click Next to proceed to the first page of the wizard.

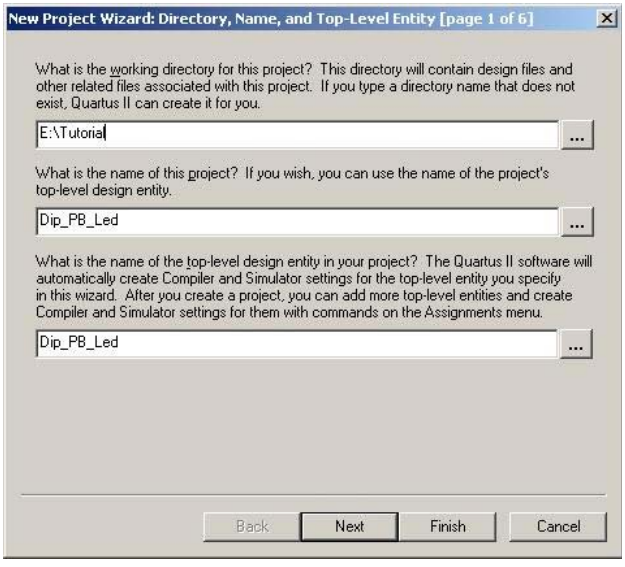
FIGURE 1. Introduction



If you don't want the introduction part to be seen in future then click in the highlighted box and select next option.

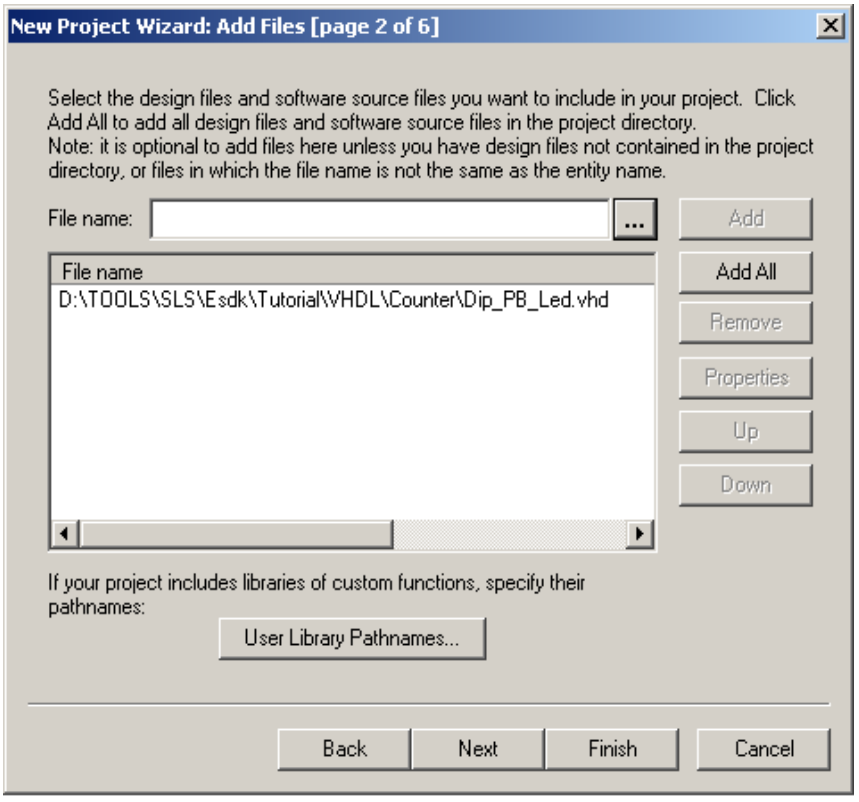
2. Type the directory name in the working directory box, or select the directory with Browse (...).
3. Type a name for the project in the project name box. For this example, type Dip_PB_Led.
4. Type Dip_PB_Led as the name of the top-level design entity of the project in the top-level design entity box.

FIGURE 2. Top Level Entry



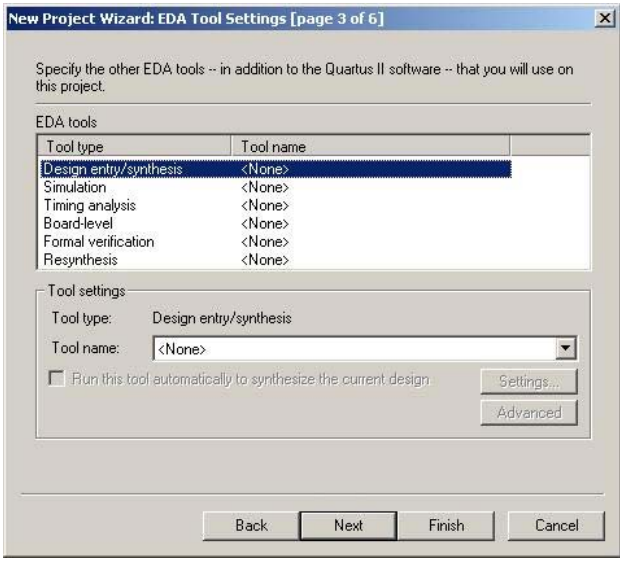
5. Click Next. The Add Files page of the New Project Wizard appears. Use Browse (...) to select the files, and then click Add to add them to the project. For this project add the Dip_PB_Led.vhd file from Tutorial\VHDL\Counter.

FIGURE 3. Add Files



6. Click Next. The EDA Tool Settings page of the New Project wizard appears. This page allows you to specify options for other EDA tools for use with this project. Since this project does not use other EDA tools, make sure that None is specified in the Tool name column for each tool type.

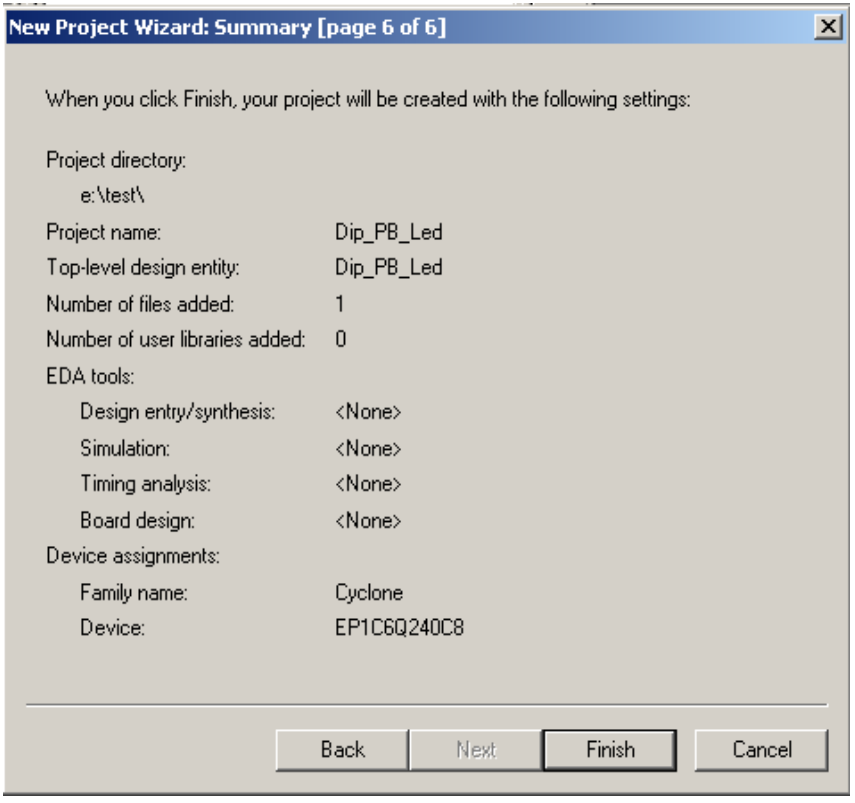
FIGURE 4. EDA Tool Setting



7. Click Next. Select the target family. Since we have Cyclone Device on the UP3, select the Cyclone family. Select the YES button, as we want to target the specific device, EP1C6Q240C8 (on UP3) in the Cyclone family.

Clicking on Next will display the Summary for the whole project.

FIGURE 5. Summary



Compilation

The Quartus II software allows you to compile an entire design, or to compile any constituent part of a design. The "compilation focus," which is the design entity you want to compile, can be selected from any portion of a project's hierarchy. You can also specify Compiler settings by following the steps in the Compiler Settings Wizard (Assignments menu).

1. Choose the Start Compilation from the Processing menu or click on the Start Compilation icon.
2. As the design compiles, the Status window automatically displays, as a percentage, the total compilation progress and the time spent in each module of the compilation. The results of the compilation are displayed in the Compilation Report window.
3. You should correct any errors in your design and recompile it until it is error-free and you receive a message indicating that full compilation was successful. The message provides the following information about the compilation:

- The final status of the compilation.
- The Compiler settings and top-level entity names.
- The current target device.
- The timing requirements, if any.
- The total number of logic cells, pins, memory, and PLLs used in the device.

The compilation result will be displayed in a small window. It shows the percentage of compilation. After compilation the compilation summary is displayed in another window.

FIGURE 6. Compilation Result

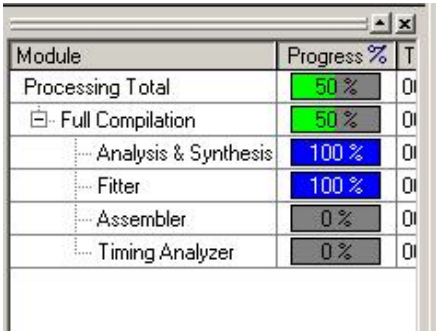
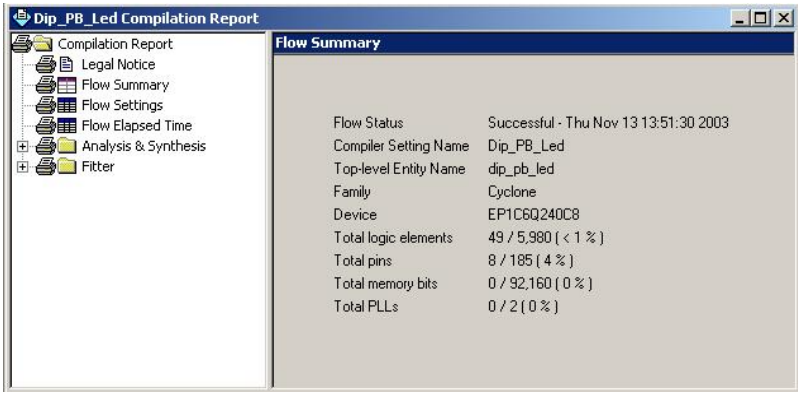


FIGURE 7. Compilation Summary



Pin Assignment & Other Options

Choose Settings (from Assignments menu). In the Category list, select Device under Compiler Settings. Current device settings will get displayed. It should be EP1C6Q240C8.

Click on Device & Pin Options. In this window, under General tab select Generate Compressed bitstreams option. Under Configuration tab select the Configuration Scheme as Active Serial and the Use Configuration Device as EPCS1.

In same window under Unused Pin tab select Reserve all unused pins - As input, tri-stated. Click OK.

Now to assign pins click on Assign Pins... Under the Available pins & existing assignments list, select the pin number for the pin to which you want to assign, change, or delete a node name assignment. Then under Assignment, type a node name.

For example, in Dip_PB_Led project, we want to assign clk signal to pin number 29. Select the pin number 29. Then under Assignment, type a Pin name as ‘clk’ (pin/signal/node name should be exactly same as mentioned in the source .vhdl or .v file). And click Add.

The Table below shows signals and pins where it should be assigned for this project.

TABLE 2. Pin Assignment

Signal	FPGA Pin No.
clk	29
reset_b	23
PBSwitch	48
DipSwitch	58
Led_inv[0]	56
Led_inv[1]	55

TABLE 2. Pin Assignment

Signal	FPGA Pin No.
Led_inv[2]	54
Led_inv[3]	53

User can directly jump to ‘Assign Pins’(section 3) before ‘Compilation’ (section 2).

Simulation

Simulation allows you to test a design thoroughly to ensure that it responds correctly in every possible situation before you program or configure a device. You must supply input vectors as the stimuli for the Quartus II Simulator. The Simulation tutorial module guides you through the steps necessary to create a Vector Waveform File (.vwf), specify Simulator settings, run a timing simulation, and analyze the simulation results.

Before Simulation, it is necessary to compile the current design.

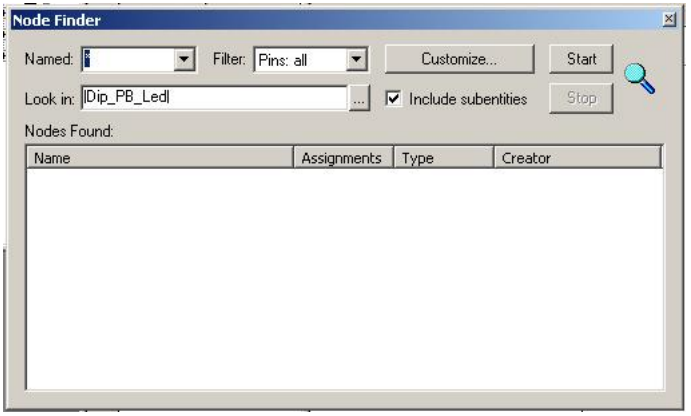
1. To create a new Vector Waveform File (.vwf), follow these steps:

- Choose New (File menu). The New dialog box appears.
- To select VWF as the file type, click the Other Files tab and select Vector Waveform File.
- Click OK. The Waveform Editor opens, displaying an empty waveform file.
- To change the end time for the file, choose End Time (Edit menu).
- In the Time box, type 100 and select ms in the list.
- Click OK.
- Save the file as Dip_PB_Led.vwf.

2. To add the input and output nodes follow these steps:

- To find the node names you want to add to the file, choose Utility Windows > Node Finder (View menu). The Node Finder appears.
- In the Node Finder, select Pins: all in the Filter list.

FIGURE 8. Node Finder

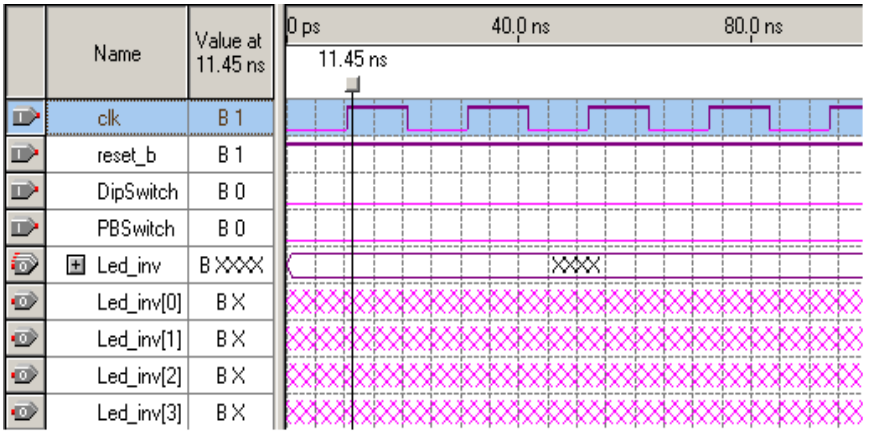


- To find the nodes you want to add to the VWF, click Start.
- In the Nodes Found list, select the reset, clk, DipSwitch, PBSwitch and Led_inv pins and drag them into the Name column of the VWF.

3. To edit the clk Input Waveform

- To select the entire clk input node waveform, click the Selection and Smart Drawing Tool on the "handle" of the clk node. The entire node, including the waveform, is highlighted.
- i.e. As shown below, select clk signal from the name window of the Dip_PB_Led.vwf file and it will get highlighted. To assign clk period, right click the clk signal and choose Value > Clock.
- The Clock dialog box appears. Type 20.8 (Clock frequency is of 48 MHz on UP3) in the Period box, and select ns in the list. In the Duty Cycle list, type or select 50. Click OK. The waveform displays a repeating waveform with a period of 20.8 ns and a duty cycle of 50% on the clk input signal.

FIGURE 9. Clock Edit



4. To edit other signals.

- Select reset_b signal. Right-click the selected interval and choose Value > Forcing High. Now select time 0 ns on the reset_b input waveform and drag the pointer to time 40 ns. Right-click the selected interval and choose Value > Forcing Low, this is to provide active low reset to the program.
- Similarly Force Low the DipSwitch signal, as this is an active high reset for counter.
- PBSwitch is the signal, on every negative edge of which the internal de-bounce counter is enabled, after the de-bounce count if the PBSwitch is still pressed the actual led count will be decremented. As the PBSwitch is manually pressed the de-bounce count value is very high in the program. But only for simulation purpose one should reduce the de-bounce counter value to speed-up the waveform simulation.
- Note: Any change in the HDL code requires recompilation of project.
- Now edit the waveform for this signal and provide proper negative pulses by forcing the signal Low and then High several times. On every negative pulse the led counter will be decremented, the negative pulse period should be more than de-bounce period.

FIGURE 10. Reset_b Edit.

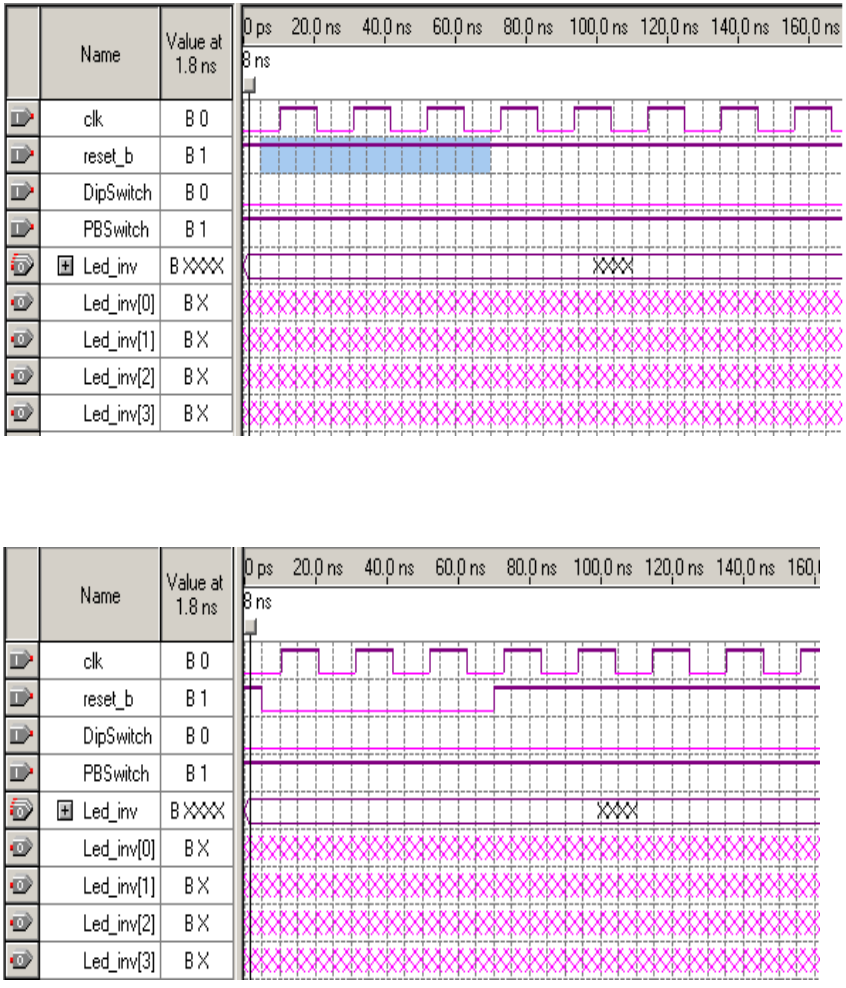
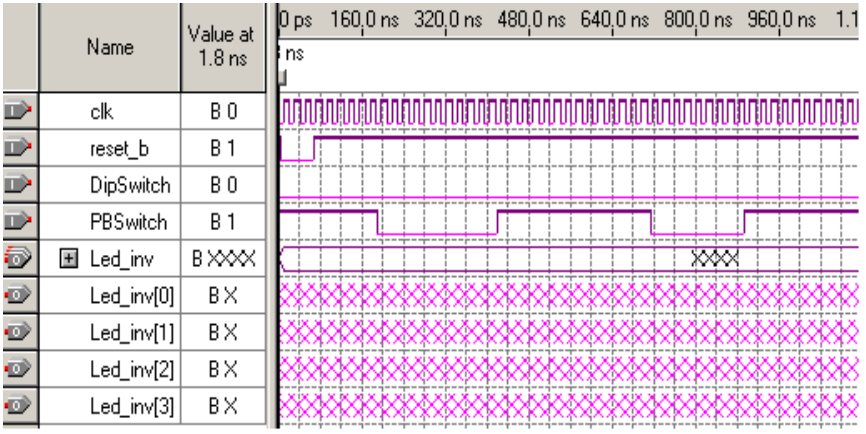
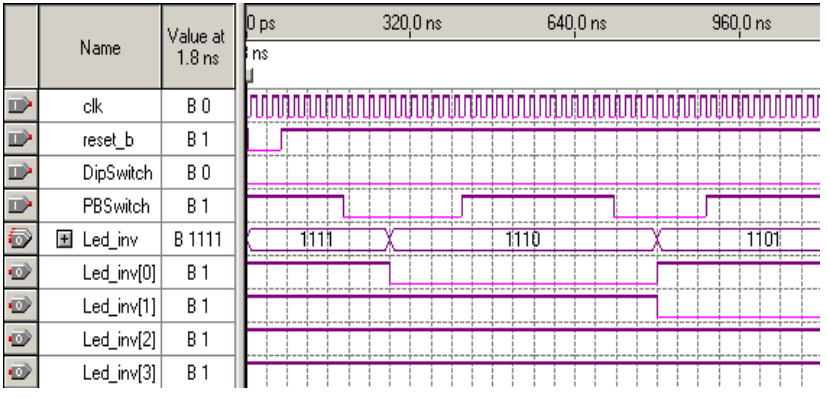


FIGURE 11. Input waveform Vector



5. Open the simulator setting wizard... from Assignments menu. Type the setting name Dip_PB_Led. Select the functional simulation (user can run the timing simulation once the functional simulation is verified). Now provide the waveform vector file that you have just created. The Next step is to provide the simulation end time, 100 us. In the last step select YES to report the simulation coverage.
6. To run the waveform simulation, click on the Start Simulation in Processing Menu.

FIGURE 12. Output Simulation waveform



Hardware Programming

There are two ways to program the FPGA on the UP3. (Refer to UP3 reference manual for more detail)

1. JTAG programming mode
2. Active Serial mode

For JTAG mode. 1.

- Choose Programmer (Tools menu). A new Chain Description file (.cdf) opens in the Programmer window automatically listing the Dip_PB_Led.sof file as the current programming file. Select Mode as JTAG in .cdf file (if not selected).
- Connect the Byte Blaster Cable to JTAG connector header (JP12) on UP3.
- Select the Program / Configure option in the .cdf file.
- Click Start Programming from Processing Menu.
- As soon as the programming starts, percentage will get displayed in the Progress bar. After it reaches 100%, Configuration done led will glow on the UP3.

For Active Serial Mode.

- Select Mode as Active Serial Programming.
- Select the Dip_PB_Led.pof file from current project directory.
- Connect the Byte Blaster Cable to Active Serial header (JP11) on UP3.
- Repeat steps 3 and 4 as given in JTAG mode.

For further information refer to Quartus II software > Help>Tutorial.

Appendix: List of Experiments

Before trying doing anything with the board read the UP3 reference manual and especially the "Remember while using UP3 with 8051 IP core" topic.

UP3 Board Understanding.

Take the UP3 board and read the UP3 board reference manual. Go through component description one by one and accordingly understand the board.

HDL Programs

Counter Programs using LEDs. Push Buttons, DIP switches

- Write an HDL program for a 4-bit counter with load value, reset and hold features. Use DIPswitches or Push Buttons as inputs. Run the above program at different frequency. Like 14.318 MHz, 48 MHz, 33 MHz, 66 MHz, 100 MHz. Display the counter value on LEDs.

Serial Port Interfacing programs

- Write an HDL program to get the serial data stream for a Serial port and convert data from serial to parallel. The serial data can be transmitted from a PC's hyper terminal.
- Write an HDL program to generate Serial stream data (start bit, data byte, stop bit) and receive the data on PC's hyper terminal.

Parallel port interfacing programs

- Write program that accepts data on the serial port and display the lower nibble of the received byte on the LED. For this you should have program that send data out on the PC's parallel port, which is connected to the UP3's parallel port.

SNAP ON Board for Santa Cruz Connectors.

User can himself prepare snap on boards that can be mounted on Santa Cruz Headers on UP3 board.

- Create a Snap On board that has 7-segment displays, LEDs, switches and Analog to Digital Converter. Write a program to 2-digit BCD counter and display the results on 7-segment. Write a program to drive a 12V Relay.
- Create a Snap board such that it can control floppy Drive Stepper motor or similar kind of motor. Write a program to control the direction, speed of this motor.

Apart from the program mentioned over here one can think of number of programs and test them.