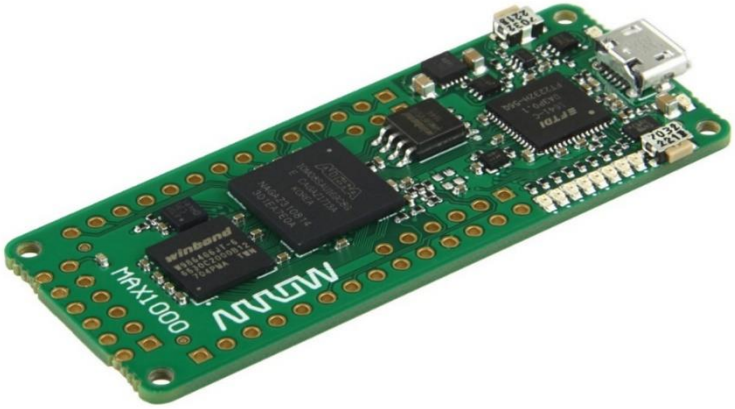
MAX1000

RTL Simulation Lab



Software and hardware requirements to complete all exercises

Software Requirements: Quartus® Prime Lite or Standard Edition version 18.0 or 18.1

ModelSim – Intel FPGA Starter Edition 10.5b

Hardware Requirements: ARROW MAX1000 Board

Revision 1.0

1. Introduction

This tutorial provides comprehensive information to help you understand how to simulate your FPGA design in the ModelSim – Intel FPGA Editor simulator. Design simulation verifies your design before programming.

Lab Notes: Many of the names that the lab asks you to choose for files, components, and other objects in this exercise must be spelled exactly as directed. This nomenclature is necessary because the pre-written software application includes variables that use the names of the hardware peripherals. Naming the components differently can cause errors. There are also other similar dependencies within the project that require you to enter the correct names.

2. Getting Started

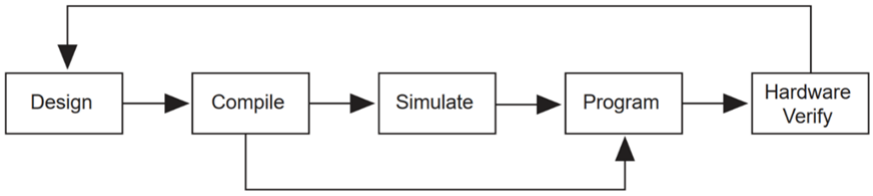
The first objective is to ensure that you have all the necessary hardware items and software installed so that the lab can be completed successfully. Below is a list of items required to complete this lab:

- MAX1000 Board **(**10M08SAU169C8G)
- USB Cable
- Lab files: RTL_Simulation_lab_template: Template files are required to complete the lab. Includes: rtl_simulation_lab.vhd, rtl_simulation_lab_tb.vhd
- Quartus Prime 18.0 Lite was used for this lab. Previous/newer versions should work (If no Quartus Prime is installed, refer to MAX1000 User Guide for instructions)
- ModelSim – Intel FPGA Starter Edition 10.5b was used for this lab. Previous/newer versions should work (If no Quartus Prime is installed, refer to MAX1000 User Guide for instructions)
- Installed Arrow USB Drivers (If not, refer to MAX1000 User Guide for instructions)
- Personal computer or laptop running 64-bit Linux / Windows 7 or later with at least an Intel i3 core (or equivalent), 4GB RAM and 12 GB of free hard disk space
- A desire to learn!

3. Design Flow

The Quartus Prime design software provides a complete, multiplatform design environment that easily adapts to your specific design needs. It is a comprehensive environment for system-on-a-programmable-chip (SOPC) design. The Quartus Prime software includes solutions for all phases of FPGA and CPLD design.

The standard FPGA design flow starts with design entry using schematics or hardware description language (HDL), such as Verilog HDL or VHDL. In this step, you create the digital circuit that is implemented inside the FPGA. The flow then proceeds through compilation, simulation, programming, and verification in the FPGA hardware.



The above diagram shows the typical design flow for the system design.

The ModelSim supports HDL design simulation at register transfer (RTL) and gate levels. You can use the Quartus Prime NativeLink feature to integrate your ModelSim simulator within the Quartus design flow and streamline simulation processing steps.

Because MAX10 devices do not support the gate level simulation, this tutorial only demonstrates the functional, RTL simulation and does not cover the gate level simulation.

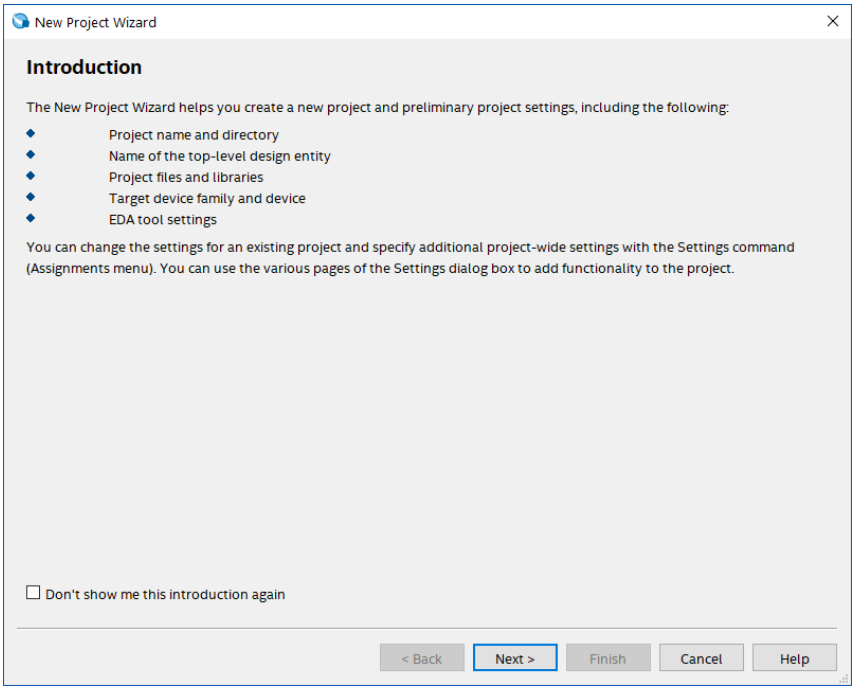
4. Project with MAX1000

4.1 Quartus Prime project

4.1.1 Create a new Quartus Prime project

4.1.1.1 If not already open, from the Start menu or the Desktop, open the Quartus Prime 18.0 Lite software.

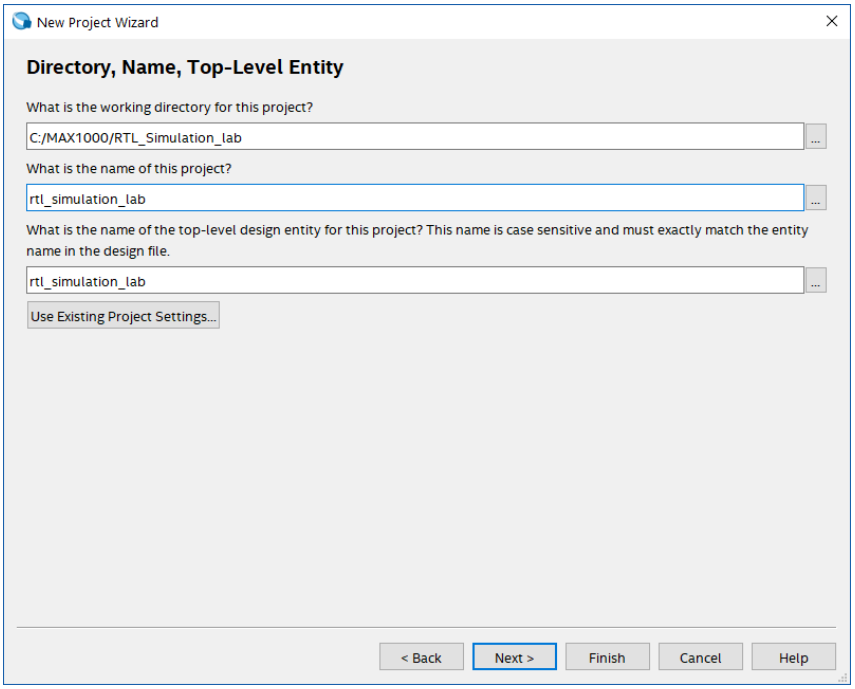
4.1.1.2 Create a new project using the New Project Wizard: **File → New Project Wizard**.



4.1.1.3 Click **Next**.

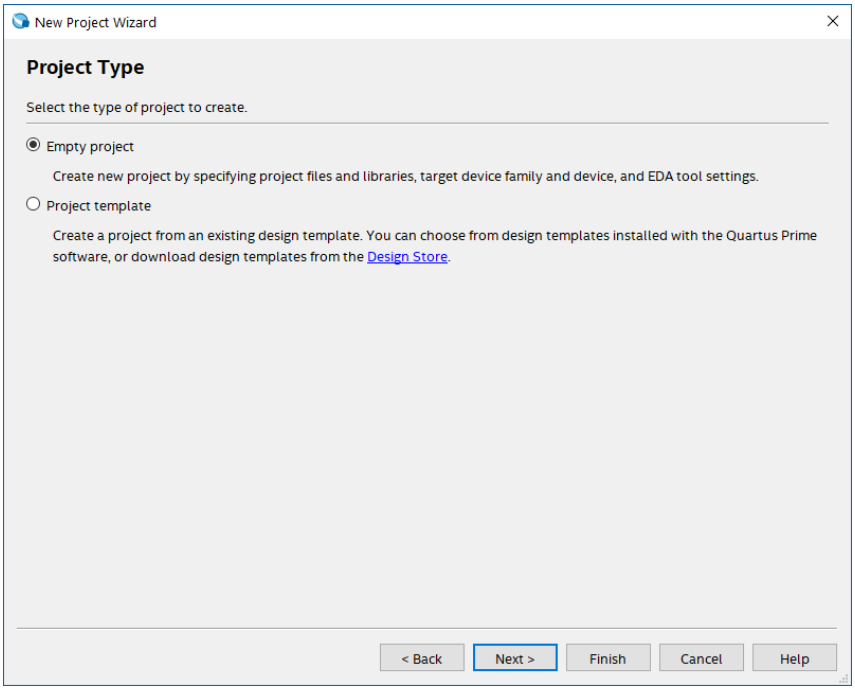
4.1.1.4 Configure the New Project Wizard directory, name and top-level entity information:

- Enter a directory in which you will store your Quartus project files for this design, for example, **C:/MAX1000/RTL_Simulation_lab**
- Specify the name of the project: **rtl_simulation_lab**
- Specify the name of the top-level entity: **rtl_simulation_lab**



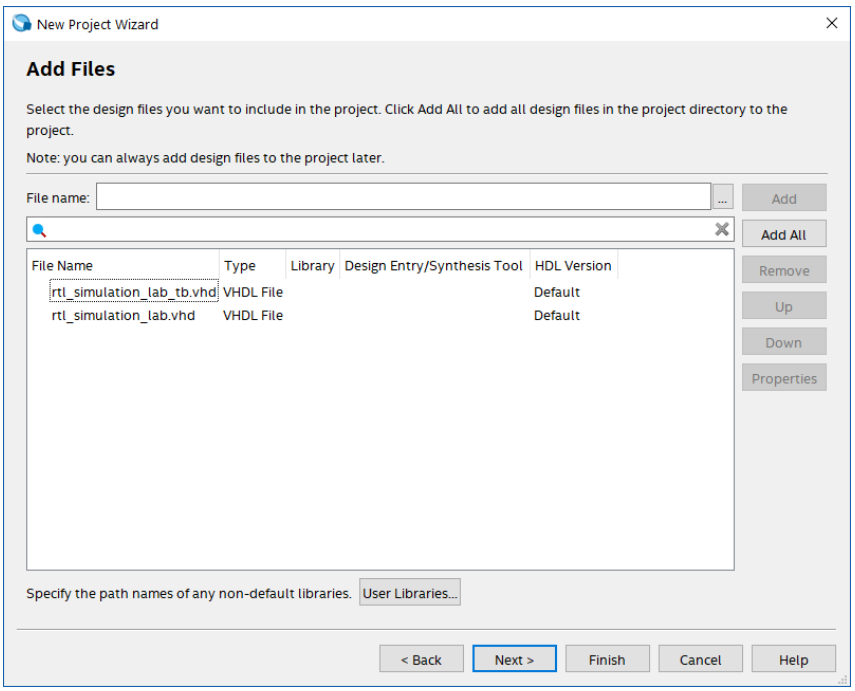
4.1.1.5 Click **Next**.

4.1.1.6 On the Project Type page, select **"Empty project"** and click **Next**.



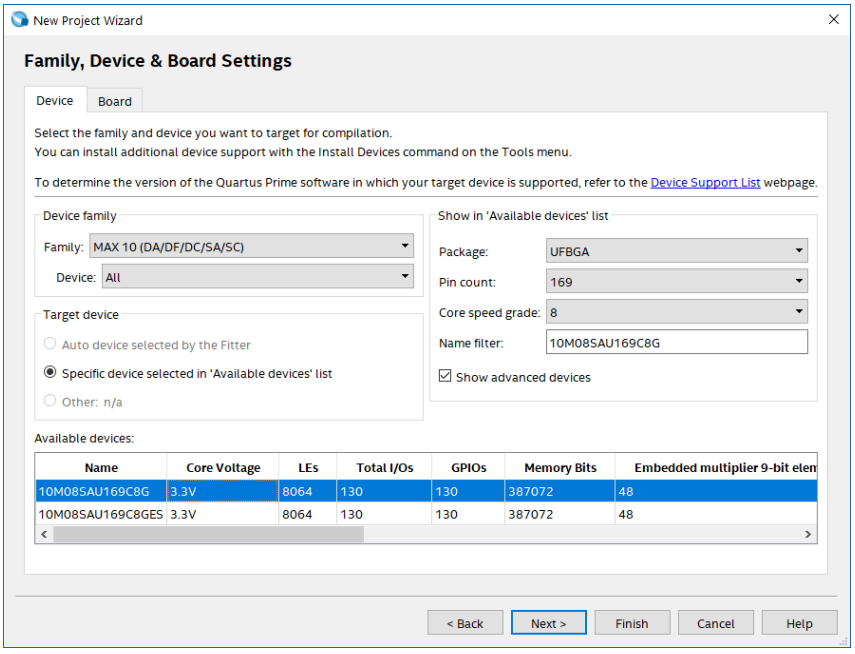
4.1.1.7 On the Add Files page, add source files to the project by clicking on the ... button and browse into the lab files folder where you will locate the provided design files and add:

- **rtl_simulation_lab_tb.vhd**
- **rtl_simulation_lab.vhd**



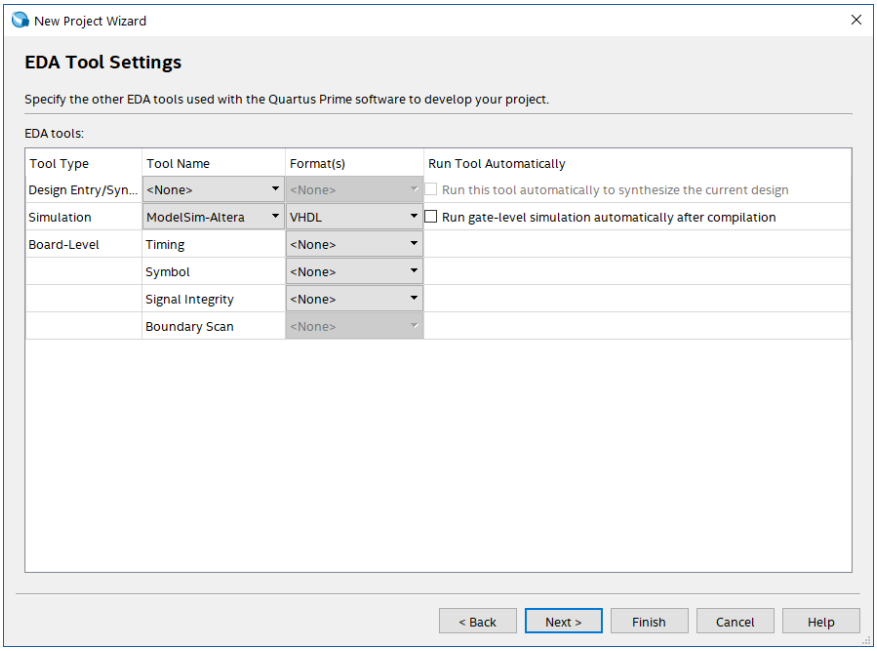
4.1.1.8 Click **Next**.

4.1.1.9 Specify Family and Device Settings. Use pull-down menus to select MAX10 family or enter the part number in the Name Filter text box. The part number is **10M08SAU169C8G**.



4.1.1.10 Click **Next**.

4.1.1.11 On the EDA Tool Settings select **ModelSim-Altera** from the pull-down menu for the Simulation tool and select VHDL as format.



4.1.1.12 Click **Finish**.

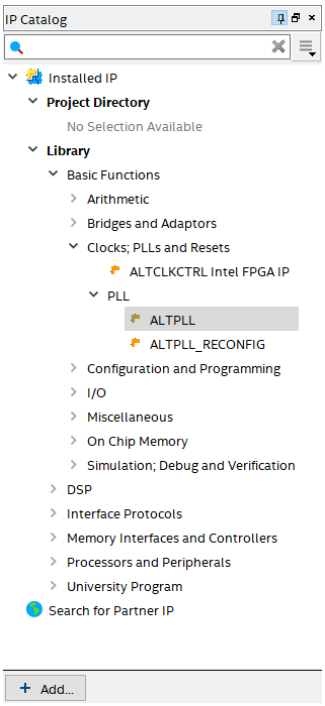
4.2 Design entry

Overview: In this module you will setup the environment for simulation and add missing component to your design.

4.2.1 Add PLL to the Quartus project

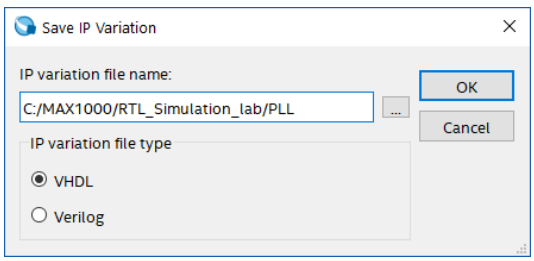
4.2.1.1 From the IP Catalog panel on the left side, expand the menus for the **Basic Functions → Clocks; PLLs and Resets → PLL** and double click on **ALTPLL**.

If the IP catalog is not visible, then right click on the toolbar and select IP catalog.



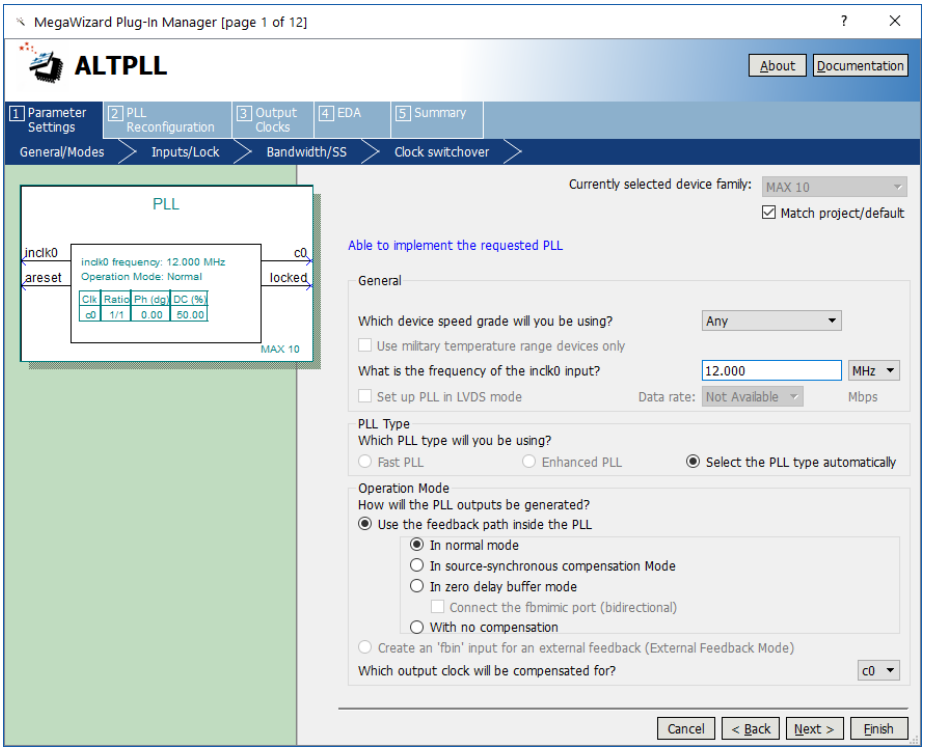
4.2.1.2 On the Save IP Variation window, enter the following information.

- IP variation file name: **<project_directory>/PLL**
- IP variation file type: **VHDL**



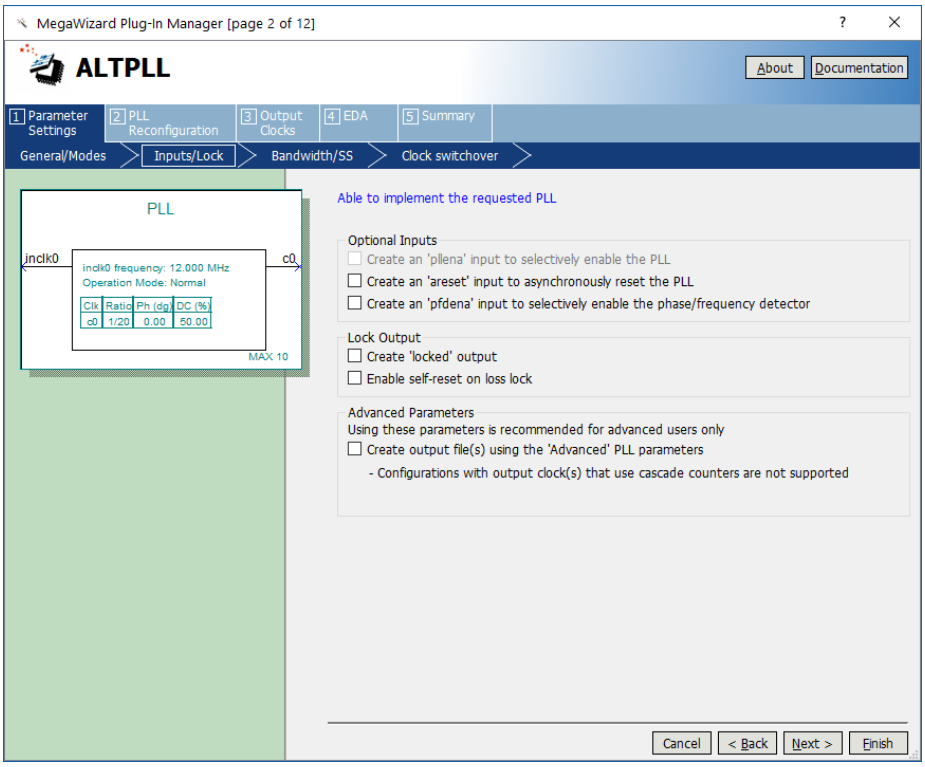
4.2.1.3 Click **OK**.

4.2.1.4 Under General/Modes tab (page 1 of 12) of PLL MegaWizard change the frequency of clock input to **12 MHz.** This source is provided by the internal oscillator in the MAX10 FPGA.



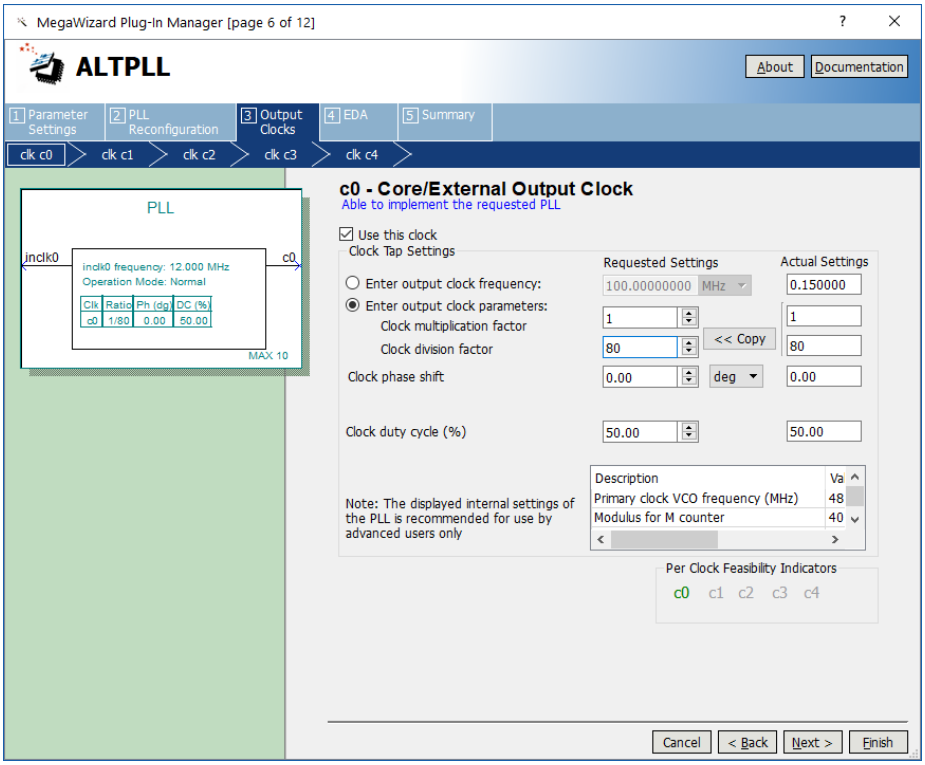
4.2.1.5 Click **Next**.

4.2.1.6 Under Input/Lock tab (page 2 of 12) uncheck ‘areset’ input and locked output option.



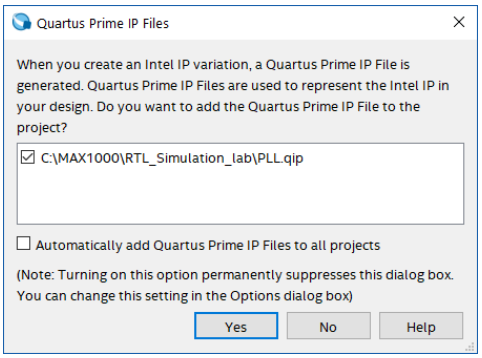
4.2.1.7 Click **Next** until you reach the **Output Clocks** tab (page 6 of 12).

4.2.1.8 Under the clk c0 tab (page 6 of 12) select “Enter output clock parameters” and set Clock division factor to **80**. Leave the rest as default.

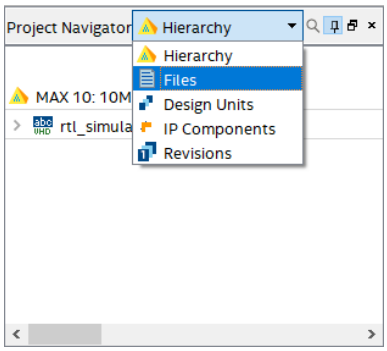


4.2.1.9 Click **Finish.** This will take you to the Summary tab (page 12 of 12). Click **Finish** again to close ALTPLL MegaWizard Manager.

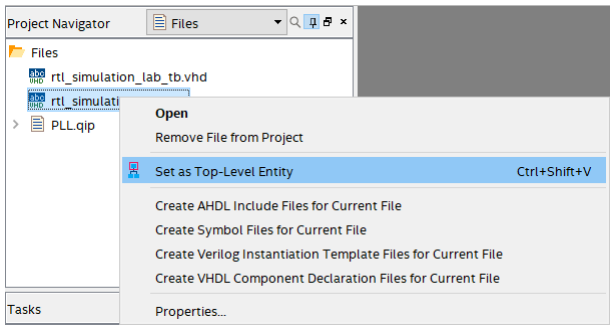
4.2.1.10 In the pop-up Quartus Prime IP Files accept all defaults and click **Yes**.



4.2.1.11 In the Project Navigator select **Files**.



4.2.1.12 Right click on rtl_simulation_lab.vhd and select **Set as Top-Level Entity** to be sure that this file will be the top-level entity.



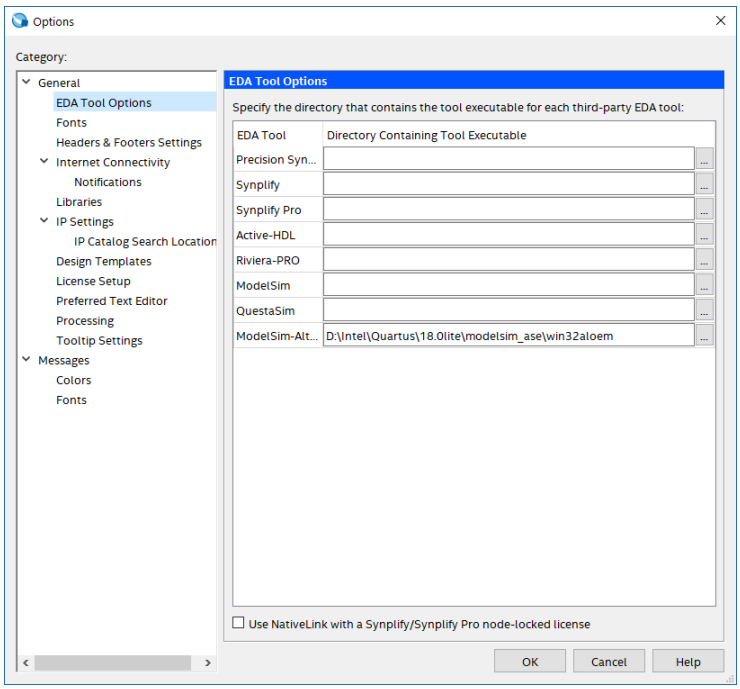
4.2.1.13 Open **rtl_simulation_lab.vhd** file.
This VHDL code is ready to simulate and the PLL module is already added. It describes a simple 8 bits up-counter circuit that operates on the PLL output clock.

4.2.1.14 Open **rtl_simulation_lab_tb.vhd** file.
This testbench is used for testing the design. Because our design only needs a clock, it only generates 12MHz input clock signal for the top-level entity.

4.2.2 Set simulation environment

4.2.2.1 Open the **Options** window from **Tools → Options**.

4.2.2.2 Under 'General' select **EDA Tool Options** category.



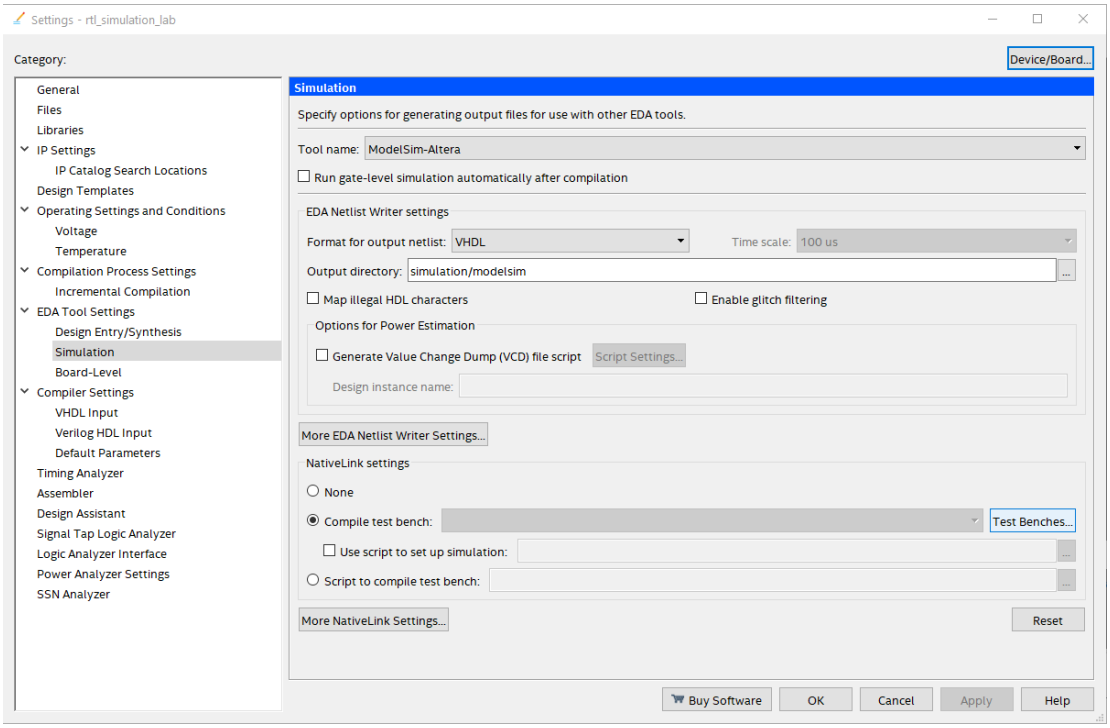
4.2.2.3 Specify the path to the directory with the executable of ModelSim-Altera by clicking on ... This folder should be **<quartus_installation_directory>/modelsim_ase/win32aloem**.

4.2.2.4 Click **OK**.

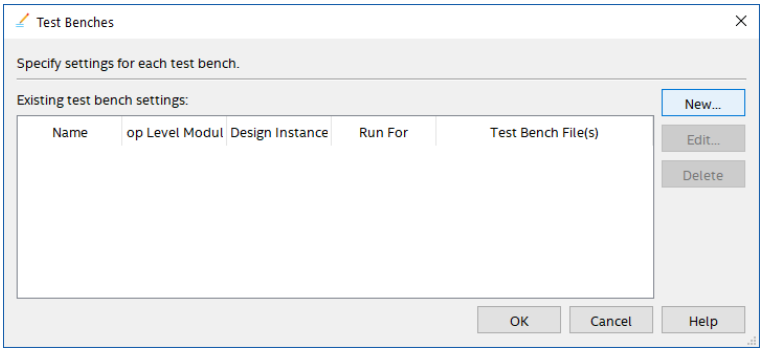
4.2.2.5 Open **Settings** window from **Assignments → Settings**.

4.2.2.6 Under EDA Tool Settings select **Simulation** category.

4.2.2.7 Under NativeLink settings choose **Compile test bench** and click **Test Benches…**

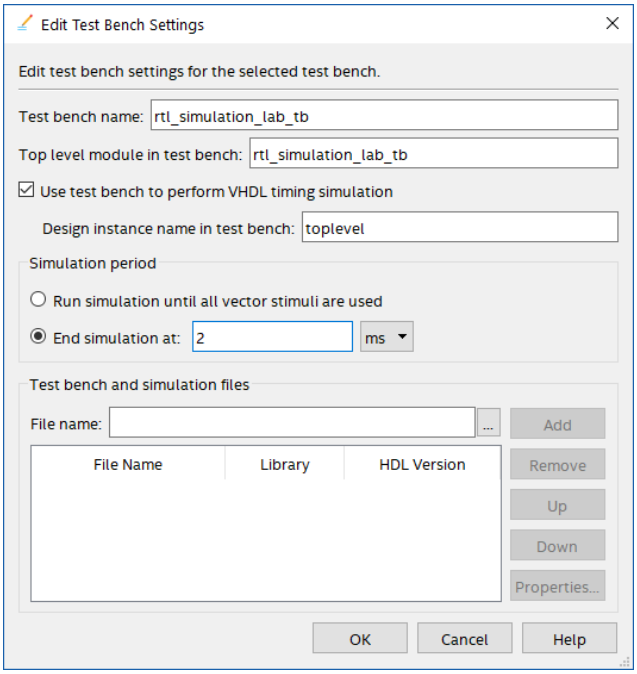


4.2.2.8 In the Test Benches window click **New…**



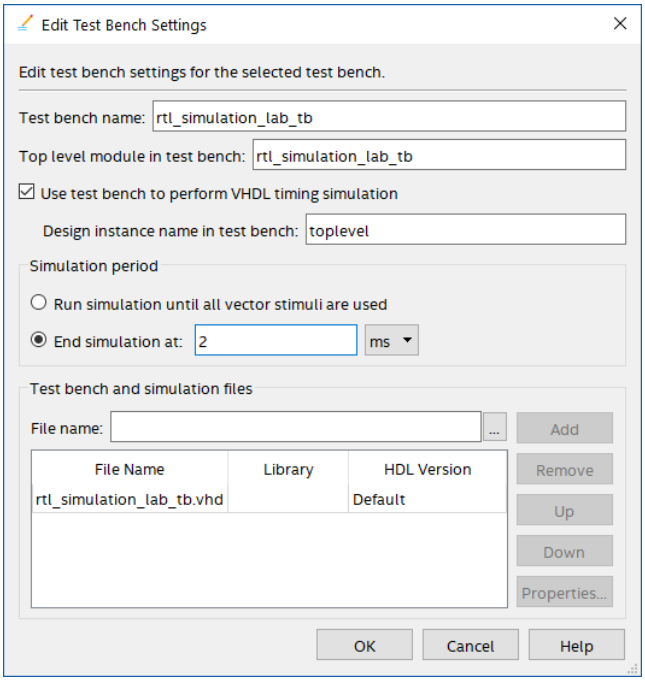
4.2.2.9 In the New Test Bench Settings window add the following:

- Test bench name: **rtl_simulation_lab_tb**
- Top level module in test bench: **rtl_simulation_lab_tb**
- Check ‘Use test bench to perform VHDL timing simulation’
- Design instance name in test bench: **toplevel**
- Choose **End simulation at** and set it to **2 ms**



4.2.2.10 Under 'Test bench simulation files' click on ... button to browse into the lab files folder and add **rtl_simulation_lab_tb.vhd**.

4.2.2.11 Click **Add**.



4.2.2.12 Click **OK**.

4.2.2.13 In the Test Benches window click **OK**.

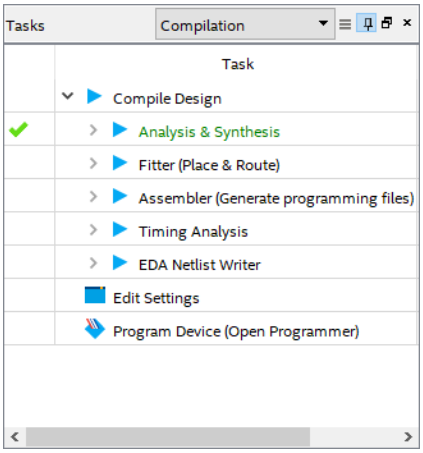
4.2.2.14 In the Settings window click **Apply** and **OK**.

4.3 Simulation

4.3.1 Analysis and Synthesis

4.3.1.1 Run Analysis and Synthesis by clicking on  button on the toolbars, or **Processing → Start → Analysis and Synthesis**.

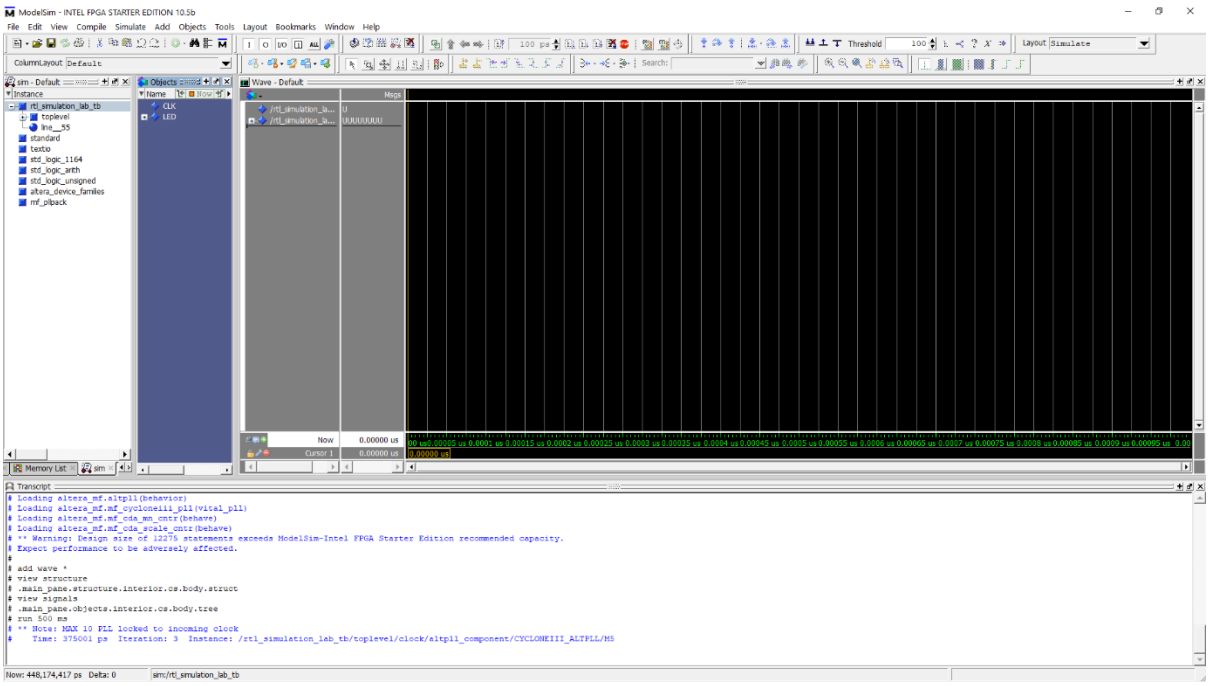
There should be no errors. If there are errors, they should be fixed before continuing. If there are no errors the compilation task windows should look like this:



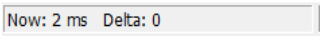
4.3.2 Run simulator

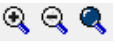
4.3.2.1 Start simulation by opening **Tools → Run Simulation Tool → RTL Simulation**.

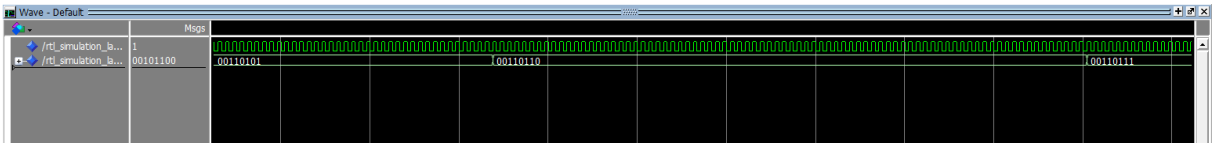
The ModelSim – Intel FPGA STARTER EDITION will open and start to load and simulate the project. At this point should be no errors in the message window. If there are, please refer again to the previous steps to resolve it.



4.3.2.2 Click on the **Wave** window and click on the  icon on the top button toolbar to refresh the Wave window or wait until it will finish with the simulation. When it finished, it will automatically refresh the Wave window and the time counter will stop on the left bottom corner.

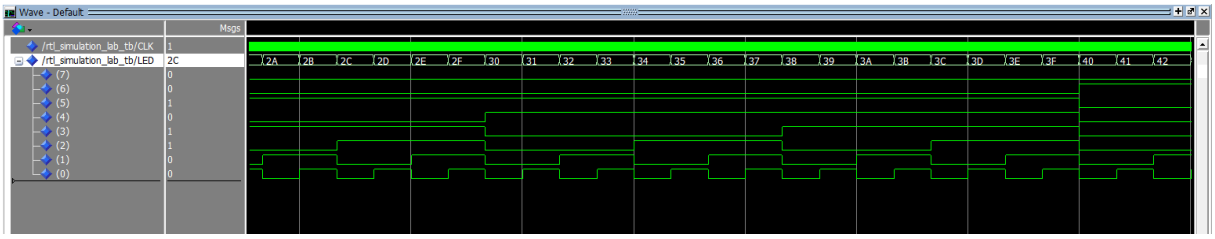


4.3.2.3 Use the Zoom  toolbar or press 'Ctrl' on your keyboard and scroll into the waveform to search for the specific points of the waveform.

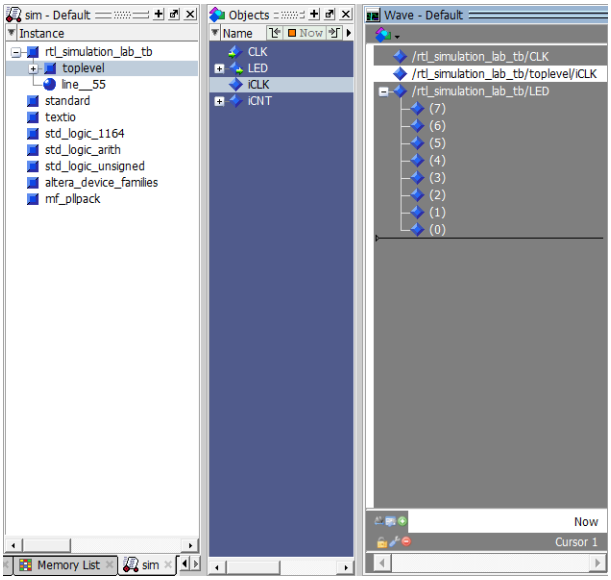


4.3.2.4 Right click on **/rtl_simulation_lab_tb/LED** and select **Radix → Hexadecimal** to change the value display.

4.3.2.5 Expand the **/rtl_simulation_lab_tb/LED** by clicking on  icon next to it.

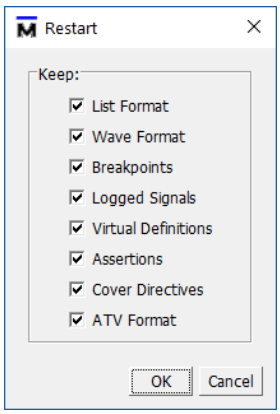


4.3.2.6 Add PLL output clock by clicking **toplevel** in the Sim window on the left side of display. Drag and drop **iCLK** to the Wave window.

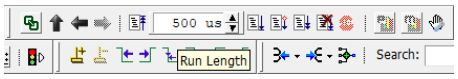


4.3.2.7 On the menu go to **Simulate →Restart…**

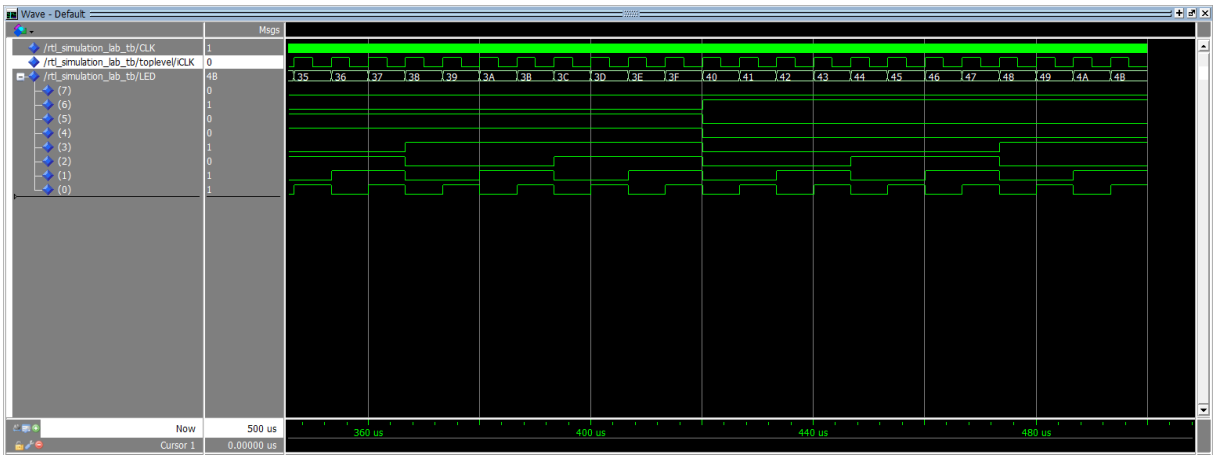
4.3.2.8 In the Restart popup window check everything and click **OK**.



4.3.2.9 Set 500 µs for the run length in the button toolbar and click on  button to run simulation.



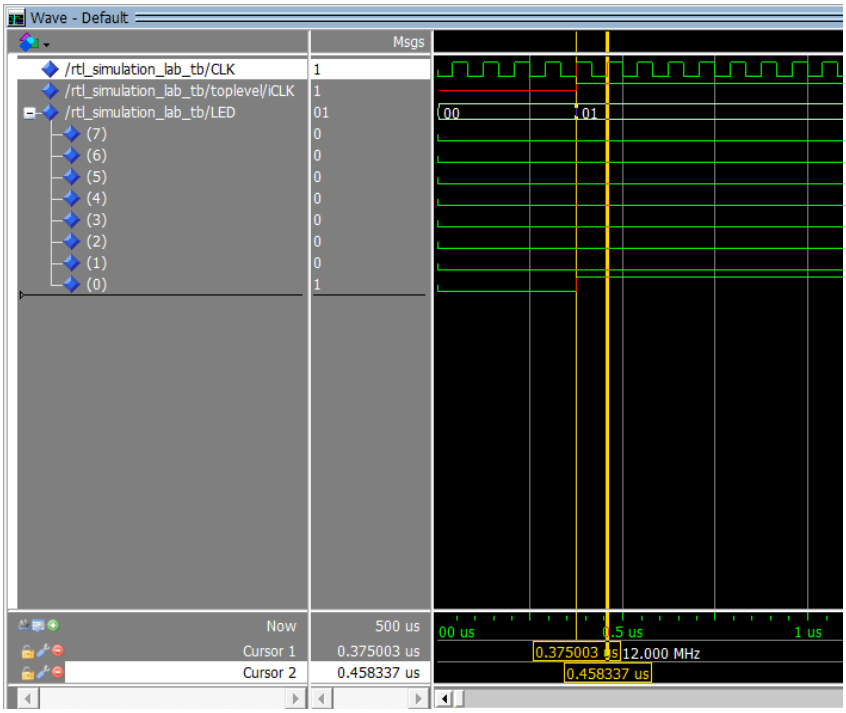
4.3.2.10 When the simulation finished, it should show the following waveform:



4.3.2.11 Add a second cursor by clicking on  button in the toolbar.

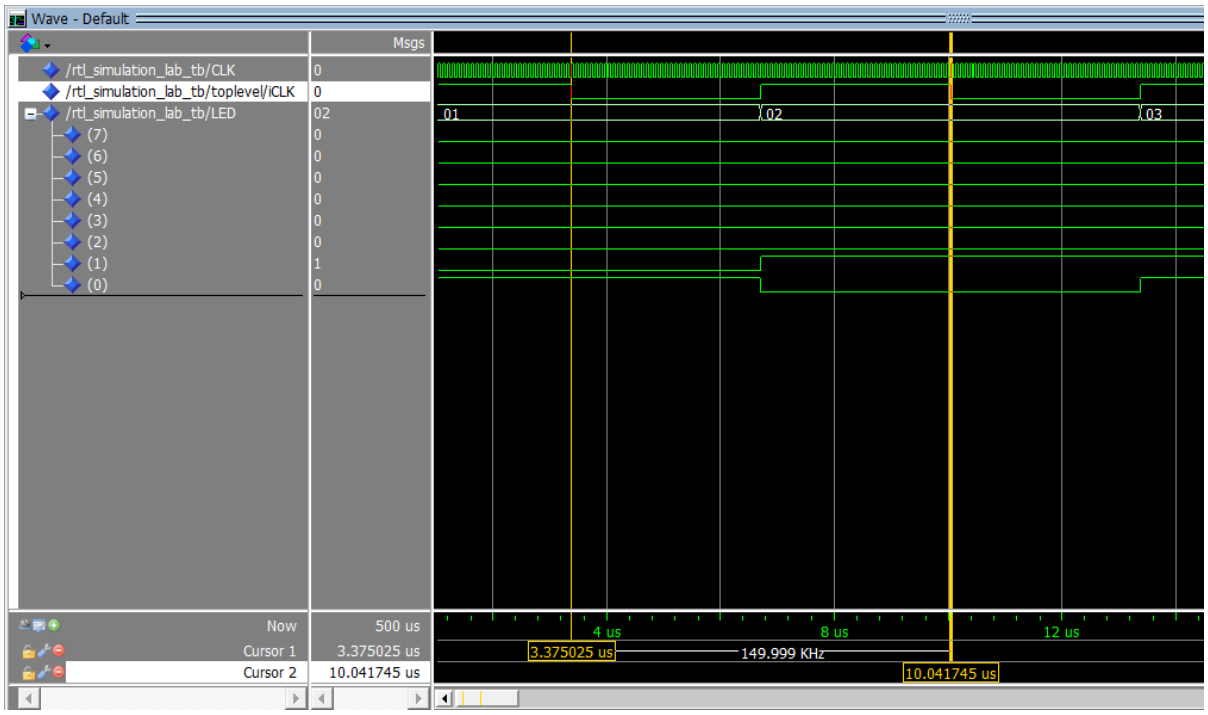
4.3.2.12 Zoom in into the waveform, and measure 1 CLK period by the cursors. You can easily adjust the cursors to the edge by these buttons on the toolbar:

4.3.2.13 Verify that 1 CLK period is 12 MHz.



4.3.2.14 Repeat the 4.3.2.12 point with 1 iCLK period.

4.3.2.15 Verify that 1 iCLK period is 150 kHz.



4.3.2.16 **Close** ModelSim.

4.3.3 Modification of the design

4.3.3.1 Open **rtl_simulation_lab.vhd** file in Quartus.

4.3.3.2 Change line 42 to:

```
signal iCNT : std_logic_vector(15 downto 0) := (others => '0');
```

4.3.3.3 Change line 66 (LED <= iCNT;) to:

```
LED <= iLED;
```

4.3.3.4 Add these codes after line 42:

```
signal iDIR : std_logic := '1';
signal iLED : std_logic_vector(7 downto 0) := x"01";
```

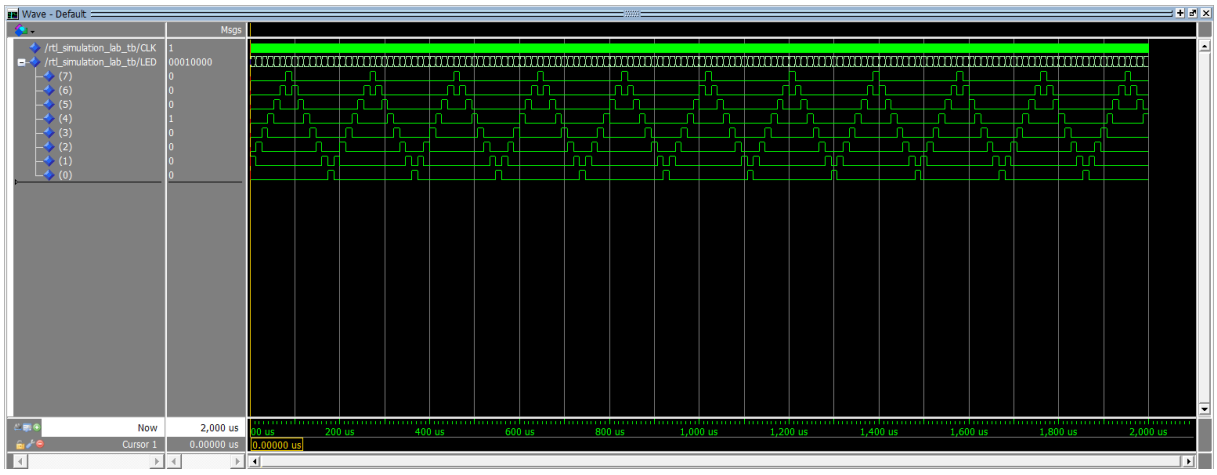
4.3.3.5 Add these codes after line 65 (after the end line of 'counter : process'):

```
leds : process(iCNT(0))
begin
      if(iCNT(0) = '1' and iCNT(0)'event) then
            if(iDIR = '1') then
                  iLED <= iLED(6 downto 0) & '0';
                  if(iLED = x"40") then
                        iDIR <= '0';
                  end if;
            else
                  iLED <= '0' & iLED(7 downto 1);
                  if(iLED = x"02") then
                        iDIR <= '1';
                  end if;
            end if;
      end if;
end process;
```

Notes: Because the simulation would take a very long time, we will use iCNT(0) for the process sensitivity in simulation to check this code functionality and will change it to iCNT(15) for the board configuration.

4.3.3.6 Save your design by clicking on button or **File → Save**.

4.3.3.7 Repeat the Analysis and Synthesis and simulation process from 4.3.1.1 point. After the simulation the waveform should look as follow:



4.3.3.8 When you finish with the simulation, **close** ModelSim.

4.3.3.9 Open **rtl_simulation_lab.vhd** file in Quartus.

4.3.3.10 Change every iCNT(0) to **iCNT(15)** from line 68 to line 70 (the first 3 line of 'leds : process').
This modification only changes the clock source, no other effect on the function.

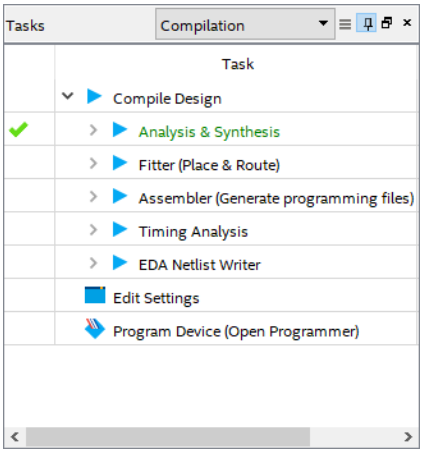
4.3.3.11 Save your design by clicking on button or **File → Save**.

4.4 Compile design

4.4.1 Analysis and Synthesis

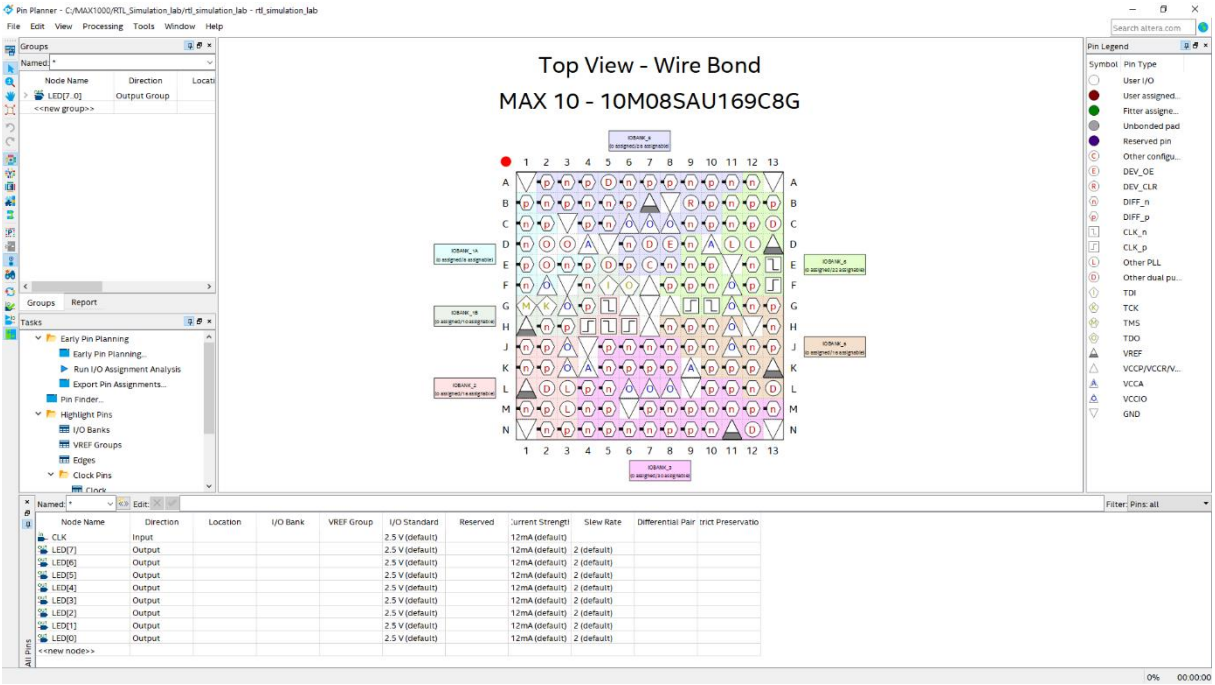
4.4.1.1 Run Analysis and Synthesis by clicking on button on the toolbars, or **Processing → Start → Analysis and Synthesis**.

There should be no errors. If there are errors, they should be fixed before continuing. If there are no errors the compilation task windows should look like this:



4.4.2 Pin Assignments

4.4.2.1 Open **Pin Planner** by clicking on button on the toolbars, or **Assignments → Pin Planner**.



4.4.2.2 In the bottom table, type **PIN_H6** in Location column of the clk_clk.

Node Name	Direction	Location	I/O Bank	VREF Group	I/O Standard	Reserved	Current Strengtl	Slew Rate	Differential Pair	trict Preservatio
CLK	Input	PIN_H6	2	B2_N0	2.5 V (default)		12mA (default)			
LED[7]	Output				2.5 V (default)		12mA (default)	2 (default)		
LED[6]	Output				2.5 V (default)		12mA (default)	2 (default)		
LED[5]	Output				2.5 V (default)		12mA (default)	2 (default)		
LED[4]	Output				2.5 V (default)		12mA (default)	2 (default)		
LED[3]	Output				2.5 V (default)		12mA (default)	2 (default)		
LED[2]	Output				2.5 V (default)		12mA (default)	2 (default)		
LED[1]	Output				2.5 V (default)		12mA (default)	2 (default)		
LED[0]	Output				2.5 V (default)		12mA (default)	2 (default)		
<<new node>>										

4.4.2.3 Repeat the previous step with the following assignments:

Node Name	Pin Location
LED[7]	PIN_D8
LED[6]	PIN_C10
LED[5]	PIN_C9
LED[4]	PIN_B10
LED[3]	PIN_A10
LED[2]	PIN_A11
LED[1]	PIN_A9
LED[0]	PIN_A8

Node Name	Direction	Location	I/O Bank	VREF Group	I/O Standard	Reserved	Current Strengtl	Slew Rate	Differential Pair	trict Preservatio
CLK	Input	PIN_H6	2	B2_N0	2.5 V (default)		12mA (default)			
LED[7]	Output	PIN_D8	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
LED[6]	Output	PIN_C10	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
LED[5]	Output	PIN_C9	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
LED[4]	Output	PIN_B10	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
LED[3]	Output	PIN_A10	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
LED[2]	Output	PIN_A11	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
LED[1]	Output	PIN_A9	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
LED[0]	Output	PIN_A8	8	B8_N0	2.5 V (default)		12mA (default)	2 (default)		
<<new node>>										

4.4.2.4 Double click in the I/O Standard column for any pins to open a drop-down list and change the 2.5V (default) to the **3.3-V LVTTL** for each pin.

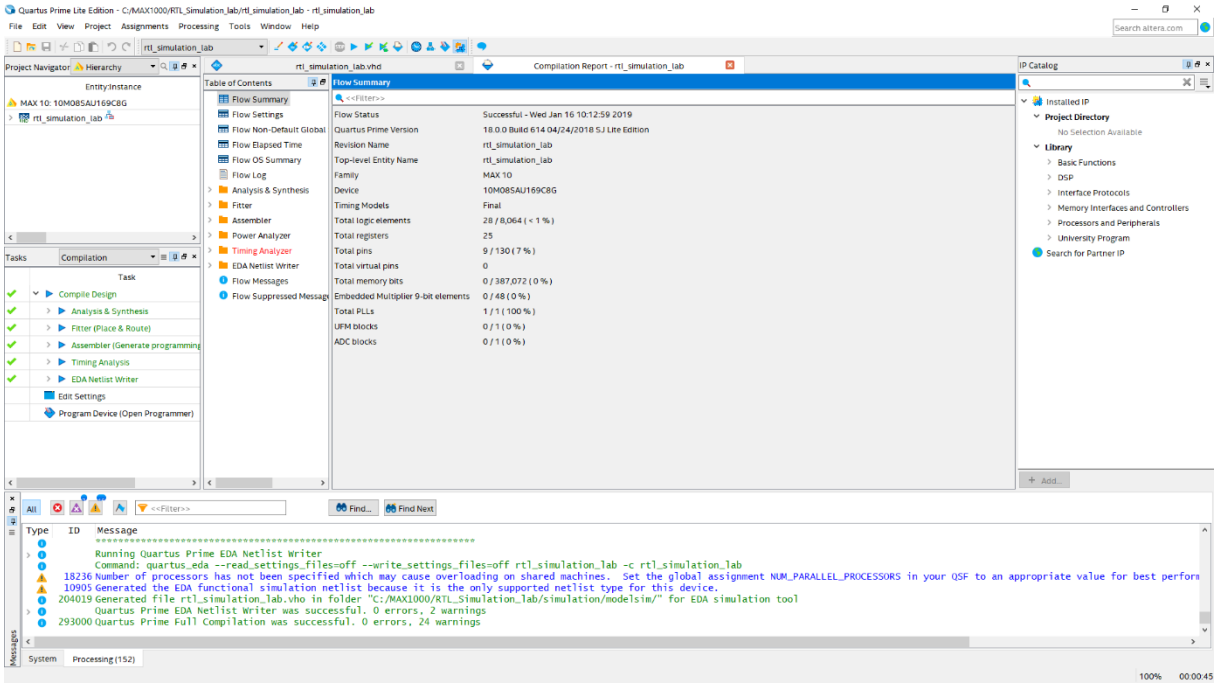
Node Name	Direction	Location	I/O Bank	VREF Group	I/O Standard	Reserved	Current Strengtl	Slew Rate	Differential Pair	trict Preservatio
CLK	Input	PIN_H6	2	B2_N0	3.3-V LVTTL		8mA (default)			
LED[7]	Output	PIN_D8	8	B8_N0	3.3-V LVTTL		8mA (default)	2 (default)		
LED[6]	Output	PIN_C10	8	B8_N0	3.3-V LVTTL		8mA (default)	2 (default)		
LED[5]	Output	PIN_C9	8	B8_N0	3.3-V LVTTL		8mA (default)	2 (default)		
LED[4]	Output	PIN_B10	8	B8_N0	3.3-V LVTTL		8mA (default)	2 (default)		
LED[3]	Output	PIN_A10	8	B8_N0	3.3-V LVTTL		8mA (default)	2 (default)		
LED[2]	Output	PIN_A11	8	B8_N0	3.3-V LVTTL		8mA (default)	2 (default)		
LED[1]	Output	PIN_A9	8	B8_N0	3.3-V LVTTL		8mA (default)	2 (default)		
LED[0]	Output	PIN_A8	8	B8_N0	3.3-V LVTTL		8mA (default)	2 (default)		
<<new node>>										

4.4.2.5 Close the Pin Planner, the settings are automatically saved.

4.4.3 Compiling the Design

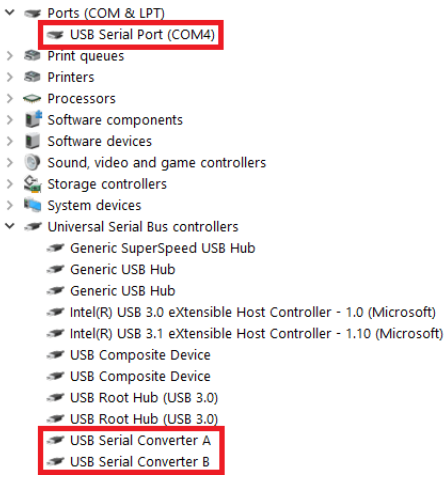
4.4.3.1 Start Compilation by clicking on  button on the toolbars, or **Processing → Start Compilation**.

There should be no errors. If there are errors, they should be fixed before re-compilating. The 100% in the lower right corner or a green checkmark next to the Compile Design in the Compilation task window indicates that the compilation was successful.

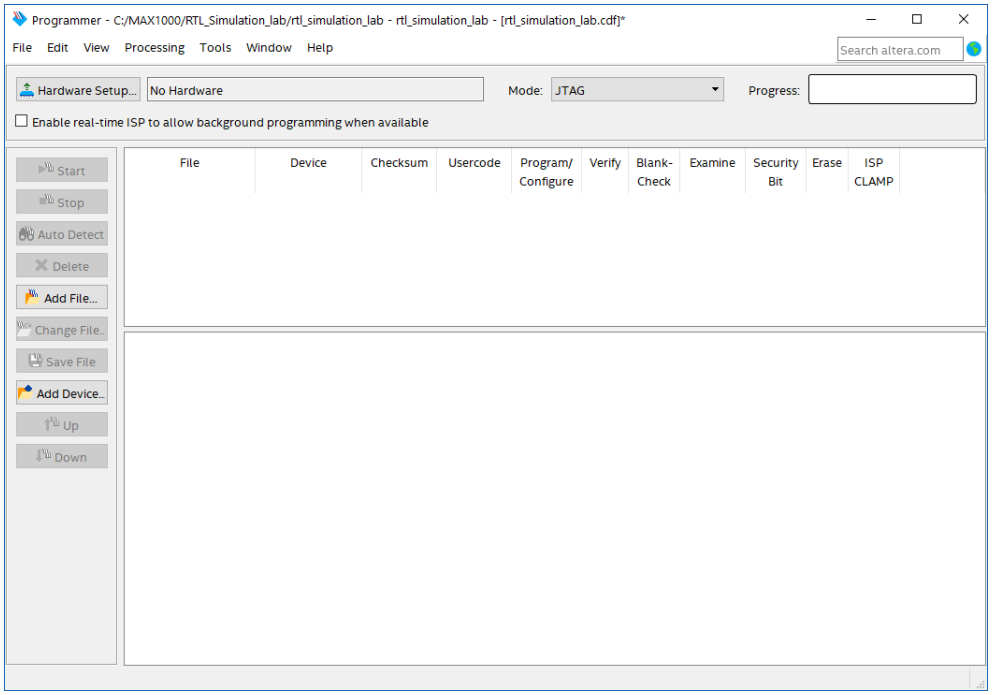


4.4.4 Configuration

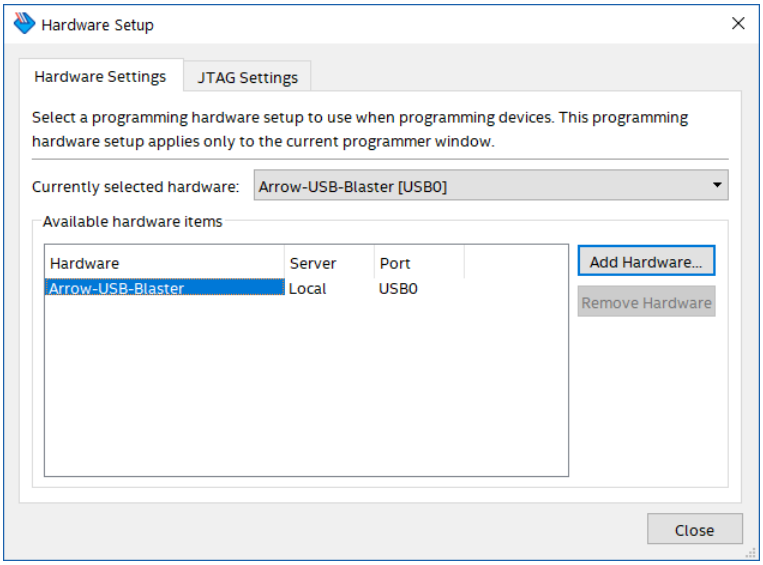
4.4.4.1 Connect your MAX1000 board to your PC using an USB cable. Since the Arrow USB Blaster should be already installed, the Window's Device Manager should display the following entries are highlighted in red (port number may differ depending on your PC):



4.4.4.2 Open the Quartus Prime Programmer from **Tools → Programmer** or double click on Program Device (Open Programmer) from the Task window.

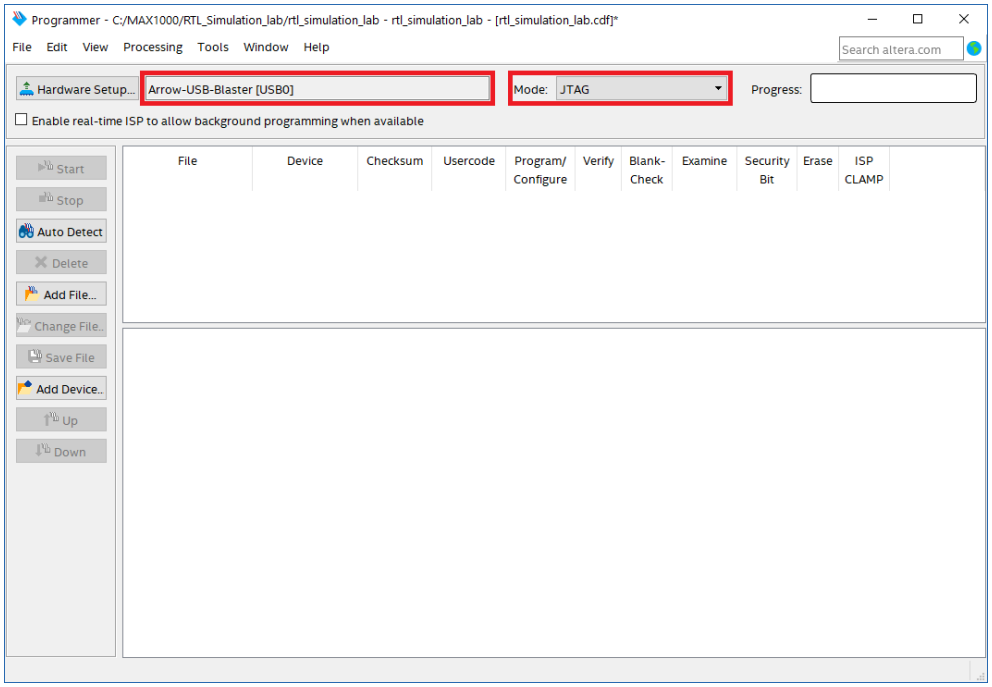


4.4.4.3 Click **Hardware Setup…** and double click **Arrow-USB-Blaster** entry in the Hardware Setup tab. The Currently selected hardware should now show Arrow-USB-Blaster [USB0] (depending on your PC, the USB port number may variant).



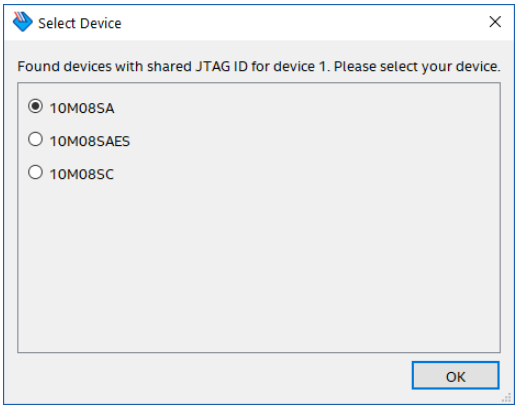
4.4.4.4 Click **Close**.

4.4.4.5 Make sure the hardware setup is Arrow-USB-Blaster [USB0] and the mode is JTAG. Click **Auto Detect**.



4.4.4.6 If the configuration has been added by default, you can skip the following steps and continue with the 4.4.4.11 point.

4.4.4.7 Select **10M08SA** device and click **OK** in the pop-up window.

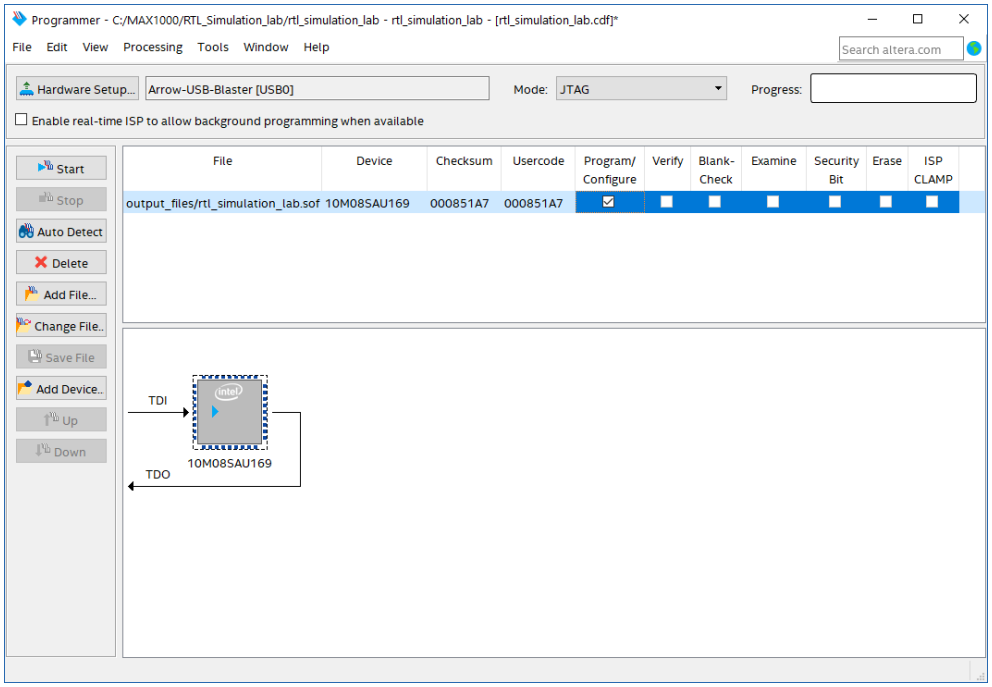


4.4.4.8 Double click <none> or click **Change File…** to choose the programming file.

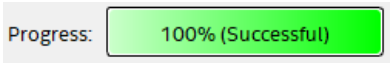
4.4.4.9 Navigate to **<project_directory>/output_files/** and select the **simple_nios_lab.sof** file.

4.4.4.10 Click **Open**.

4.4.4.11 Make sure the Programmer shows the correct file and the correct part in the JTAG chain and check the Program/Configure checkbox.



4.4.4.12 Click **Start** to program the board. When the configuration is complete, the Progress bar should show 100% (Successful).



4.4.5 Testing the design

4.4.5.1 Verify that the LEDs on MAX1000 board toggles in the order shown by the simulation.

CONGRATULATIONS! YOU HAVE SUCCESSFULLY COMPLETED THE RTL SIMULATION LAB!

5 Revision History

Version	Change Log	Date of Change
V1.0	Initial Version	16/01/2019

6 Legal Disclaimer

ARROW ELECTRONICS

EVALUATION BOARD LICENSE AGREEMENT

By using this evaluation board or kit (together with all related software, firmware, components, and documentation provided by Arrow, "Evaluation Board"), You ("You") are agreeing to be bound by the terms and conditions of this Evaluation Board License Agreement ("Agreement"). Do not use the Evaluation Board until You have read and agreed to this Agreement. Your use of the Evaluation Board constitutes Your acceptance of this Agreement.

PURPOSE

The purpose of this evaluation board is solely intended for evaluation purposes. Any use of the Board beyond these purposes is on your own risk. Furthermore, according the applicable law, the offering Arrow entity explicitly does not warrant, guarantee or provide any remedies to you with regard to the board.

LICENSE

Arrow grants You a non-exclusive, limited right to use the enclosed Evaluation Board offering limited features only for Your evaluation and testing purposes in a research and development setting. Usage in a live environment is prohibited. The Evaluation Board shall not be, in any case, directly or indirectly assembled as a part in any production of Yours as it is solely developed to serve evaluation purposes and has no direct function and is not a finished product.

EVALUATION BOARD STATUS

The Evaluation Board offers limited features allowing You only to evaluate and test purposes. The Evaluation Board is not intended for consumer or household use. You are not authorized to use the Evaluation Board in any production system, and it may not be offered for sale or lease, or sold, leased or otherwise distributed for commercial purposes.

OWNERSHIP AND COPYRIGHT

Title to the Evaluation Board remains with Arrow and/or its licensors. This Agreement does not involve any transfer of intellectual property rights ("IPR) for evaluation board. You may not remove any copyright or other proprietary rights notices without prior written authorization from Arrow or it licensors.

RESTRICTIONS AND WARNINGS

Before You handle or use the Evaluation Board, You shall comply with all such warnings and other instructions and employ reasonable safety precautions in using the Evaluation Board. Failure to do so may result in death, personal injury, or property damage.

You shall not use the Evaluation Board in any safety critical or functional safety testing, including but not limited to testing of life supporting, military or nuclear applications. Arrow expressly disclaims any responsibility for such usage which shall be made at Your sole risk.

WARRANTY

Arrow warrants that it has the right to provide the evaluation board to you. This warranty is provided by Arrow in lieu of all other warranties, written or oral, statutory, express or implied, including any warranty as to merchantability, non-infringement, fitness for any particular purpose, or uninterrupted or error-free operation, all of which are expressly disclaimed. The evaluation board is provided "as is" without any other rights or warranties, directly or indirectly.

You warrant to Arrow that the evaluation board is used only by electronics experts who understand the dangers of handling and using such items, you assume all responsibility and liability for any improper or unsafe handling or use of the evaluation board by you, your employees, affiliates, contractors, and designees.

LIMITATION OF LIABILITIES

In no event shall Arrow be liable to you, whether in contract, tort (including negligence), strict liability, or any other legal theory, for any direct, indirect, special, consequential, incidental, punitive, or exemplary damages with respect to any matters relating to this agreement. In no event shall arrow's liability arising out of this agreement in the aggregate exceed the amount paid by you under this agreement for the purchase of the evaluation board.

IDENTIFICATION

You shall, at Your expense, defend Arrow and its Affiliates and Licensors against a claim or action brought by a third party for infringement or misappropriation of any patent, copyright, trade secret or other intellectual property right of a third party to the extent resulting from (1) Your combination of the Evaluation Board with any other component, system, software, or firmware, (2) Your modification of the Evaluation Board, or (3) Your use of the Evaluation Board in a manner not permitted under this Agreement. You shall indemnify Arrow and its Affiliates and Licensors against and pay any resulting costs and damages finally awarded against Arrow and its Affiliates and Licensors or agreed to in any settlement, provided that You have sole control of the defense and settlement of the claim or action, and Arrow cooperates in the defense and furnishes all related evidence under its control at Your expense. Arrow will be entitled to participate in the defense of such claim or action and to employ counsel at its own expense.

RECYCLING

The Evaluation Board is not to be disposed as an urban waste. At the end of its life cycle, differentiated waste collection must be followed, as stated in the directive 2002/96/EC. In all the countries belonging to the European Union (EU Dir. 2002/96/EC) and those following differentiated recycling, the Evaluation Board is subject to differentiated recycling at the end of its life cycle, therefore: It is forbidden to dispose the Evaluation Board as an undifferentiated waste or with other domestic wastes. Consult the local authorities for more information on the proper disposal channels. An incorrect Evaluation Board disposal may cause damage to the environment and is punishable by the law.