Simple Pulse Width Modulation Using Cyclone 10LP Evaluation Kit

This design example demonstrates a simple pulse width modulation (PWM) using the Cyclone 10LP Evaluation Kit. The brightness of an LED is used to demonstrate this feature. Instead of varying an analog voltage to the LED, the duty cycle of a pulse width modulated signal can be used to adjust the brightness.

Intended audiences:

- Students
- Engineers

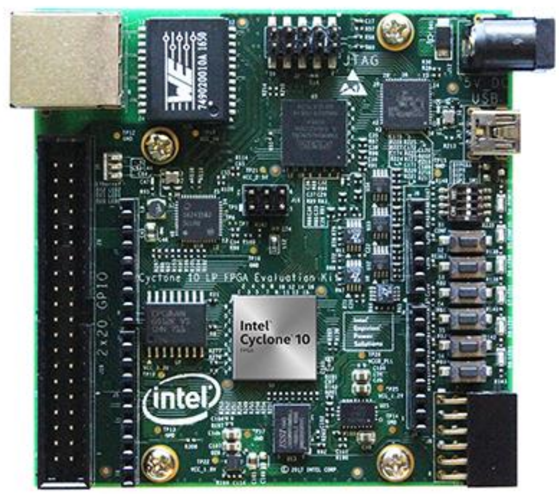
Table of Contents

About the Design

This design example accepts input from the user keys and modifies the brightness of the onboard LEDs.

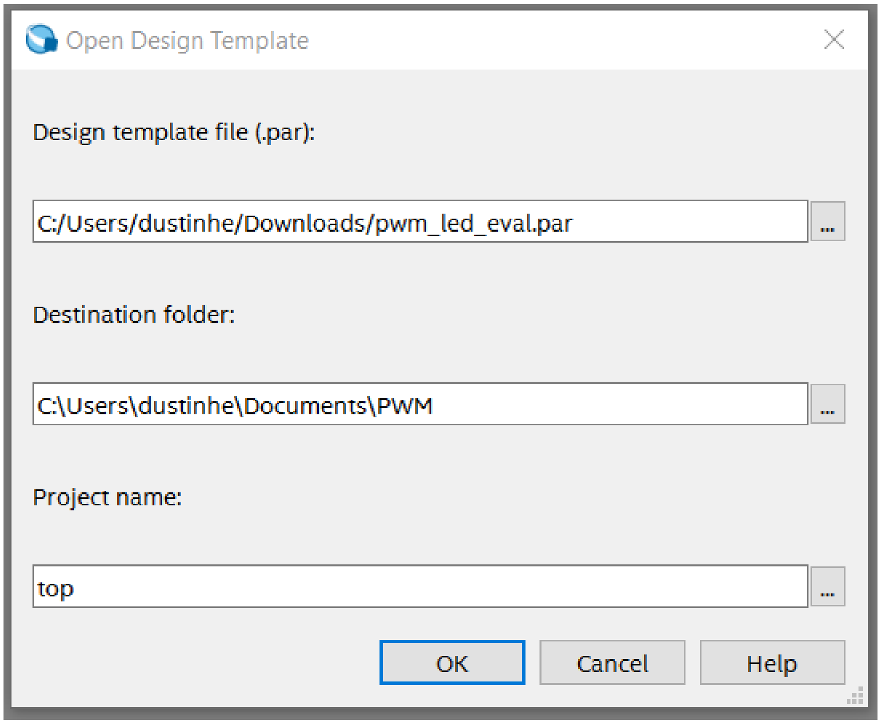
Two keys, Key2 and Key3, are used for the design. Key2 corresponds to incrementing the value of the duty cycle. Key3 corresponds to decrementing the duty cycle. The outputs are 4 LEDs. The LEDs display the change in brightness from the PWM signal.

Initially all the LEDs are off indicating a duty cycle of 0. The user can start by pressing the up or down key. Pressing the up key will cause the duty cycle to increment and the LEDs will dimly glow. The user can continue to push or hold the up key to see the LEDs brightly light up. The user may try a variety of combinations of the up and down Keys and verify the design.

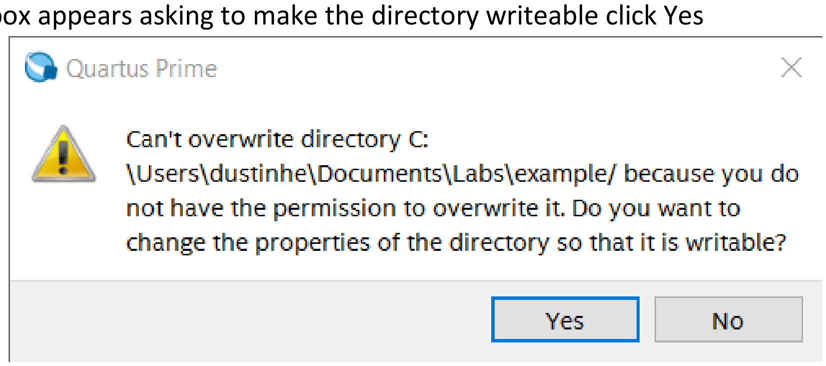


Project Setup and Programing

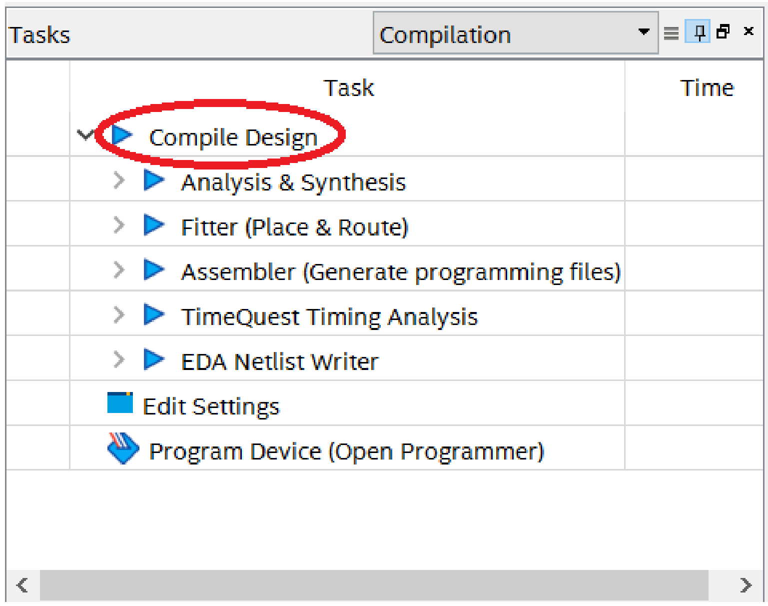
1) Download the pwm_led_eval.par from the design store.
2) Double click the .par file
3) The Quartus GUI will open and a dialog window will request a new project location
4) Enter the directory you wish to use for storing the project



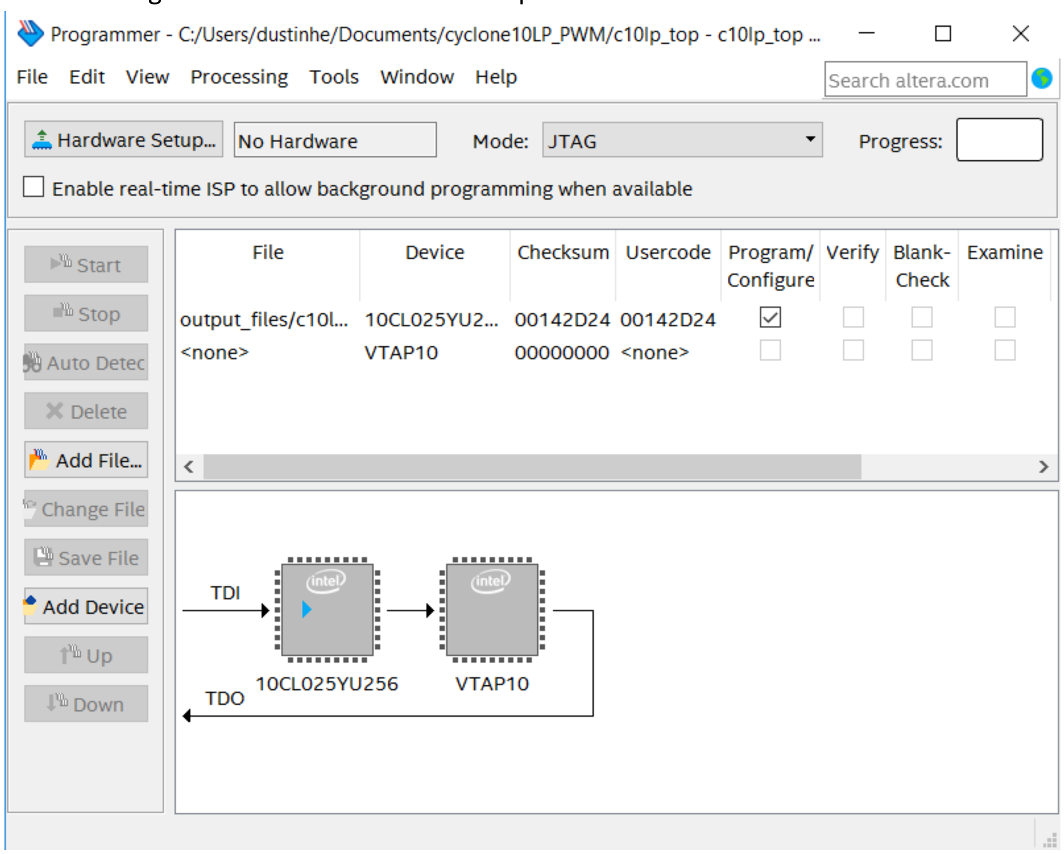
5) Click Ok
    a) If a dialog box appears asking to make the directory writeable click Yes



6) Compile the design by double clicking the Compile Design button

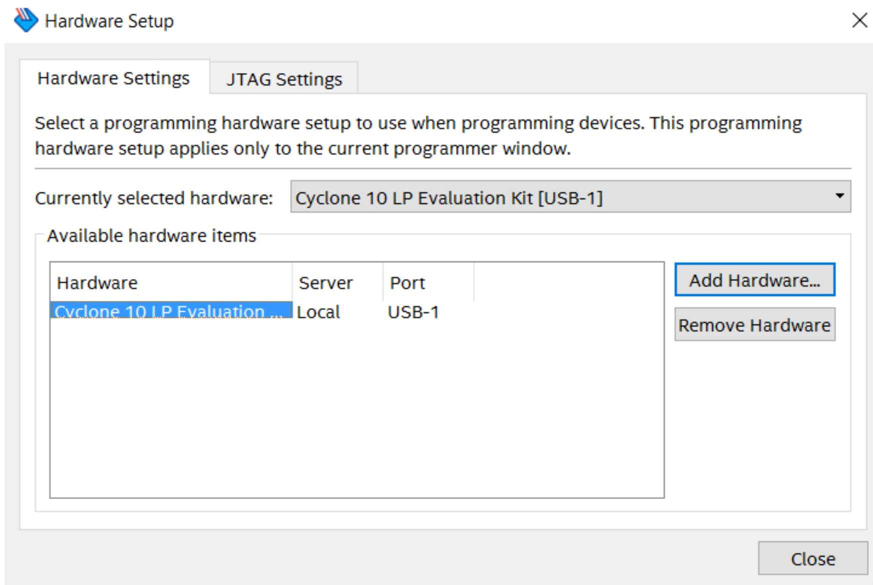


7) Connect and power on the development kit via the USB mini port
8) Program the development kit
   a) Go to Tools>Programmer. A new window will open

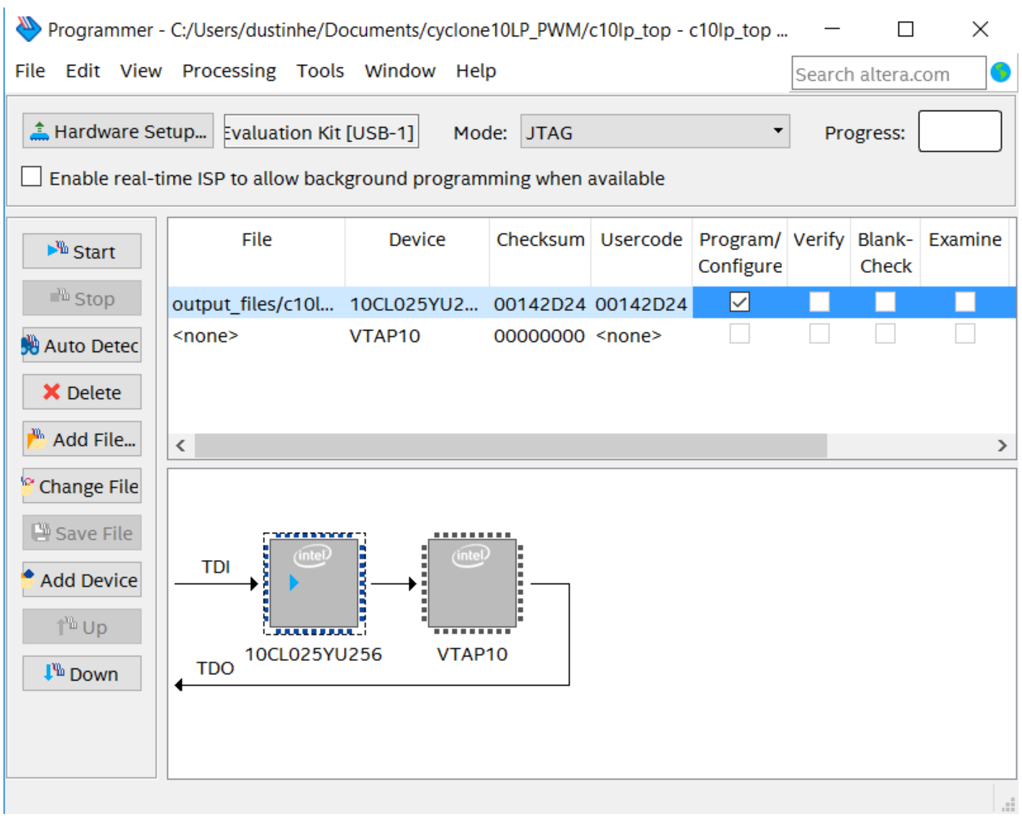


   b) Click on Hardware Setup. A hardware Setup dialog box will open
   c) Select Cyclone 10 LP Evaluation Kit under currently selected hardware. Click on close.
      i) NOTE: If the kit does not appear in the hardware check:

(1) The board is connected and turned on
(2) The USB Blaster Drivers are properly installed (Link)

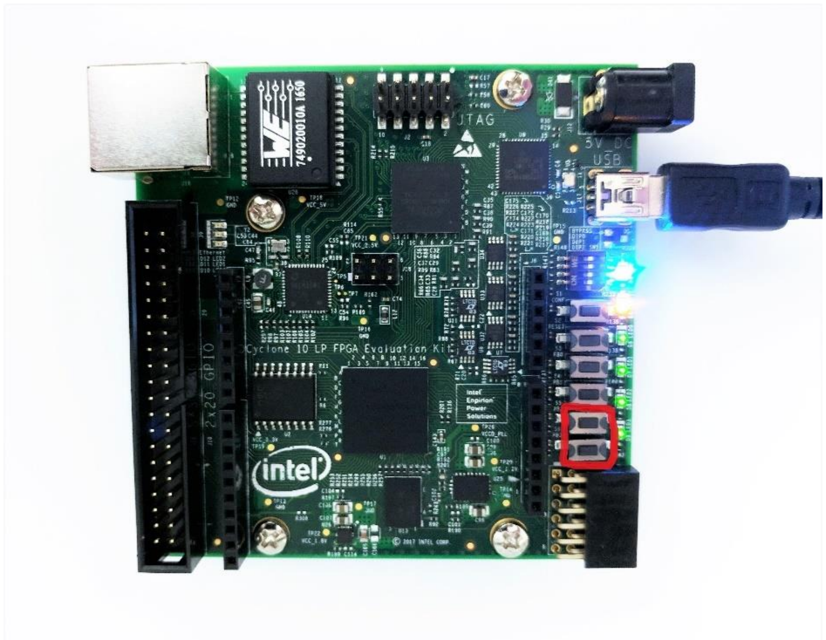


d) Click auto detect
e) Click on the gray box labeled 10CL025YU256 to select it
f) Click Change File
g) Select the .sof file in the output_files directory
h) Select the checkbox Program/Configure. The programing window should look like the figure below



i) Click on start

9) When the programing tool is completed the user may use the keys



PWM Applications

This section of the documentation briefly discusses some of the potential applications of PWM. PWM can be an extremely useful tool if applied to the appropriate application.

First, PWM is commonly used to control and modify power delivery of DC devices. Dimming LEDs is just one small example of this application. DC motor speed can also be controlled using PWM.

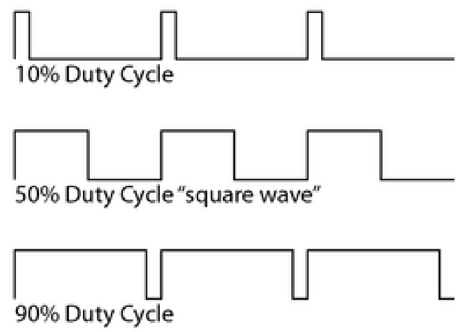
Second, many external devices use PWM as a control method. Many low-cost servomotors and electronic speed controllers interface via PWM.

Last, more complex PWM algorithms can be used for voltage regulation and inversion. Switching regulators like boost and buck converters use PWM to control voltages in analog circuits. Additionally, DC to AC converters use PWM algorithms like sinusoidal pulse with modulation(SPWM) and space vector pulse with modulation(SVPWM).

Theory of Operation

PWM:

PWM simply refers to switching a signal on and off with varying duty cycles to produce a desired result. In our example project, we are using the duty cycle to manipulate the power delivered to our LEDs. The duty cycle refers to the percentage of time the signal is turned on. The average power delivered by the PWM can be calculated by the equation $I * V * (Duty\ Cycle\ \%) = P$. If we assume that our LED operates at 1.8V and 10mA we get the following average power ratings.

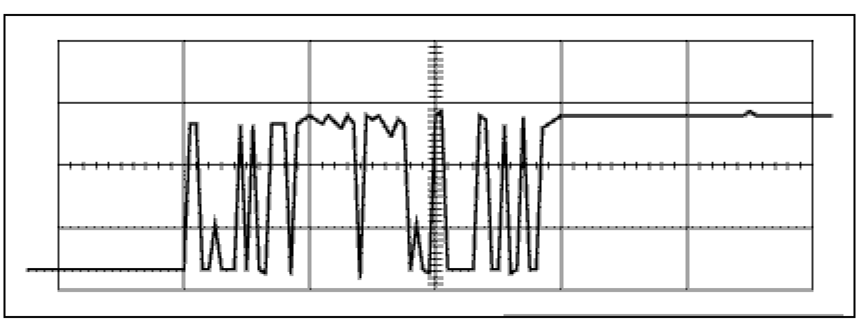


Duty Cycle	Calculated Average Power	Brightness
0%	0	Off
10%	1.8mW	Very Dim
50%	9mW	Dim
90%	16.2mW	Bright
100%	18mW	Brightest

If we look at average power vs the brightness in the table, we can notice that the amount of power delivered to the LED directly affects how bright it is.

Switch Debounce:

Switches are notoriously known for being extremely noisy signals. This phenomenon is often referred to as switch bounce. The noise in the signal comes from the mechanical contacts within the switch bouncing on each other. Removing the noise from this signal is known as switch debouncing. While there are many different methods for switch debounce the method in this project utilizes a state machine. The state machine Ignores the signal from the switch for a brief amount of time after the initial press. This filters the noise from the signal received by the other modules in the project.



Acknowledgement

This design is based on Simple PWM Tutorial for Be Micro MAX 10 FPGA Evaluation Kit produced by Arrow Electronics.

Document Revision History

Documentation and design based off Simple PWM (Pulse Width Modulation) Using MAX 10 FPGA Evaluation Kit.

Date	Version	Changes
November 2014	Original MAX 10 Document	Initial Release for MAX
August 16, 2017	1.0	Updated to Quartus 17.0, Ported Design to Cyclone 10LP, Updated Documentation Template.