PWM Generator

February 2015 **Reference Design RD1178**

Introduction

Pulse Width Modulation (PWM) of a signal involves the modulation of its duty cycle, to convey either information over a communication channel or control the amount of power sent to a load. PWM is employed in a variety of applications, ranging from measurements and communications to power control and conversion, mainly because of its low power, noise-free and low cost characteristics. This document provides a brief description of PWM Generator and its implementation.

The design is implemented in VHDL. The Lattice iCEcube2™ Place and Route tool integrated with the Synopsys Synplify Pro® synthesis tool is used for the implementation of the design. The design can be targeted to other iCE40™ FPGA product family devices.

Features

The following features are supported:

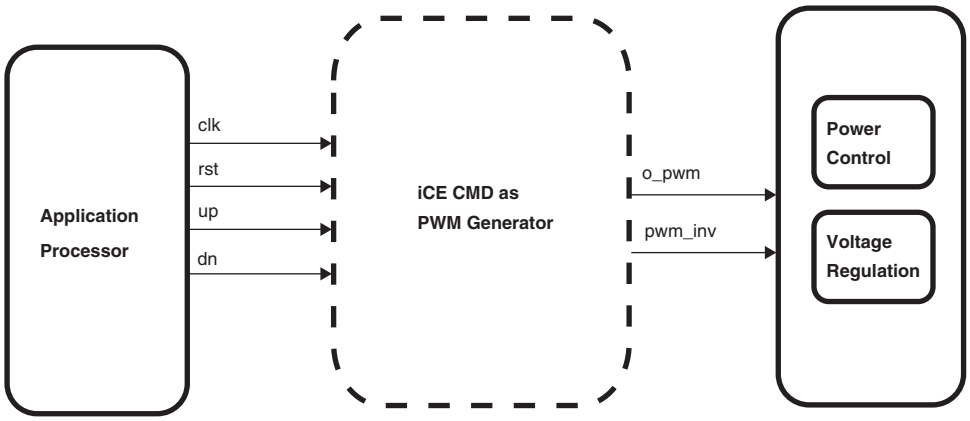
- User configurable PWM Cycle width
- Pulse Width Control using up/down signals
- Both direct and inverted PWM outputs
- VHDL RTL, testbench for simulation

The following feature is NOT supported:

- Configurable resolution

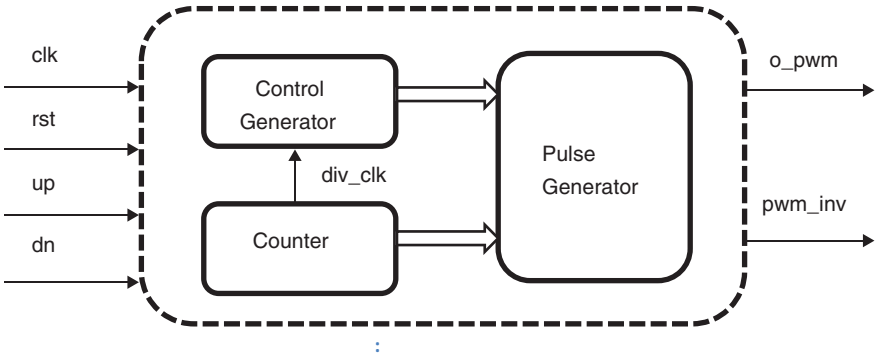
System Block Diagram

Figure 1. System Block Diagram



Functional Description

Figure 2. Functional Block Diagram



The following are brief descriptions of the internal blocks.

Control Word Generator

Generates a Control Word based on 'up' and 'dn' signals. Control word increases when 'up' is held high and decreases when 'dn' is held high. The Resolution (pwm_size) parameter controls the control word range. The supported pwm_size for the given design is 8 and 16.

Counter

This is a simple counter, cycles through counts 0 to maximum range, which is controlled by the generic parameter pwm_size.

Pulse Generator

This module compares the Counter and the Control Word Generator. Generates a High on PWM output until Counter output reaches the Control word and Low afterwards.

Signal Description

Table 1. Signal Description

Signal	Width	Type	Description
clk	1	Input	System Clock
rst	1	Input	Asynchronous Active High Reset
up	1	Input	Increase in Duty cycle when kept high .up has higher priority than dn.
dn	1	Input	Decrease in duty cycle when kept high
o_pwm	1	Output	PWM signal
pwm_inv	1	Output	Inverted PWM signal

Operation Sequence

Initialization Condition

When 'rst' is High, 'pwm' will be Low and 'pwm_inv' will be High.

Example #1

Keep 'up' High. The duty cycle of 'pwm' will keep increasing (shown in Figure 3), and the duty cycle of 'pwm_inv' will decrease.

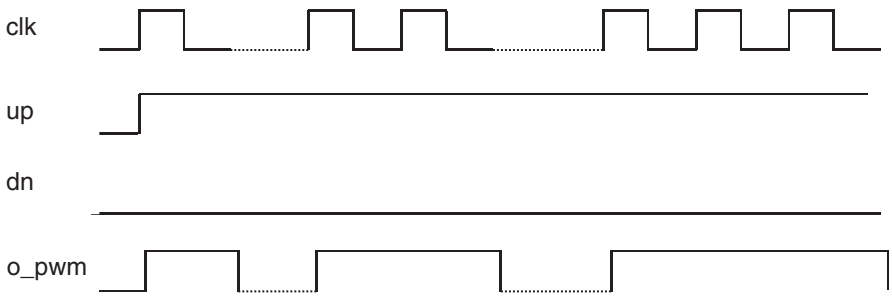
(If both 'up' and 'dn' are high, 'up' will have higher priority than 'dn')

Example #2

Keep 'dn' High and 'up' low. The duty cycle of 'pwm' will keep decreasing (shown in Figure 4) and the duty cycle of 'pwm_inv' will increase.

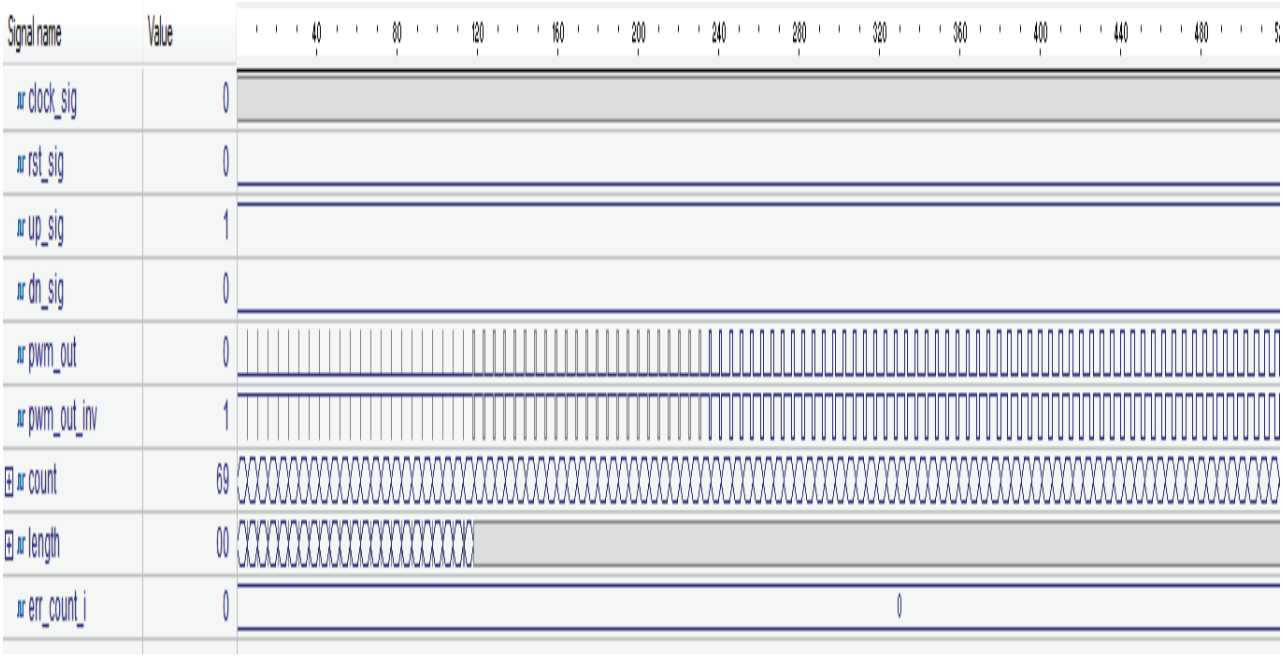
Timing Diagram

Figure 3. Timing diagram when up=1



Simulation Waveforms

Figure 4. Simulation Waveforms



Implementation

This design is implemented in VHDL. When using this design in a different device, density, speed or grade, performance and utilization may vary.

Table 2. Performance and Resource Utilization

Device Family	Language	Synthesis Tool	Utilization (LUTs)	fMAX (MHz)	I/Os	Architecture Resources
iCE40[1]	VHDL	LSE	41	>50	6	(6/160) PLBs
		Syn	45	>50	6	(9/160) PLBs

1. Performance and utilization characteristics are generated using iCE40LP1K-CM121 with iCEcube2 2014.12 design software.

References

- DS1040, iCE40 LP/HX Family Data Sheet

Technical Support Assistance

e-mail: techsupport@latticesemi.com

Internet: www.latticesemi.com

Revision History

Date	Version	Change Summary
February 2015	1.1	Updated Implementation section. Updated Table 2, Performance and Resource Utilization. — Added LSE support.
		Updated References section.
		Updated Technical Support Assistance information.
April 2013	01.0	Initial release.