International Journal of Advanced Research in Engineering and Technology (IJARET)
Volume 12, Issue 3, March 2021, pp.97-102, Article ID: IJARET_12_03_011
Available online at http://www.iaeme.com/IJARET/issues.asp?JType=IJARET&VType=12&IType=3
ISSN Print: 0976-6480 and ISSN Online: 0976-6499
DOI: 10.34218/IJARET.12.3.2021.011

IMPLEMENTATION OF SIMPLE PWM/PPM GENERATOR FOR MICROCONTROLLER USING VERILOG

Shankar A
Associate Professor, Electronics and Communication Engineering
Geethanjali College of Engineering and Technology, Hyderabad, India

Lokeshwari A
UG Student, Geethanjali College of Engineering and Technology, Hyderabad, India

Manohar B
UG Student, Geethanjali College of Engineering and Technology, Hyderabad, India

Pranavi S
UG Student, Geethanjali College of Engineering and Technology, Hyderabad, India

ABSTRACT

A Pulse Width Modulation (PWM) Signal is a method for generating an analog signal using a digital source. Now-a-days microcontrollers support PWM outputs. A PWM signal consists of two main components a duty cycle and a frequency. The PWM Generator block generates pulses for carrier-based pulse width modulation (PWM) converters using two-level topology. Most of the microcontrollers will have built in timers which helps in generation of PWM signal with various widths. PWM generator helps in controlling the brightness in smart lighting systems by controlling voltage to LED driver connected with LED bulbs. Also helps in controlling the speed of motors by varying voltage supply to it. It is also used as modulation scheme to encode message into pulsing signal for transmission. Pulse-position modulation (PPM) is a form of signal modulation in which M message bits are encoded by transmitting a single pulse in one of possible required time shifts. This is repeated every T-seconds, such that the transmitted bit rate is. bits per second. Used in non-coherent detection where a receiver does not need any Phase lock loop for tracking the phase of the carrier, Used in radio frequency (RF) communication. Also used in contactless smart card, high frequency, RFID (radio frequency ID) tags.

This project demonstrates how a simple and fast a pulse width modulator (PWM) generator and a pulse position modulator (PPM) can be implemented using Verilog programming. It is simulated using ModelSim, a multi-language (hardware description language) simulation environment from Mentor Graphics and tested on FPGA

development board. In the hardware simulation the module for PWM generator can be realized using development board. The board is built around Xilinx Spartan-3E FPGA and Atmel AT90USB2 USB controller. It provides complete, ready-to-use hardware suitable for hosting circuits, ranging from basic logic devices to complex controllers. The FPGA development board allows USB programming through the computer USB port. The program, when run, automatically detects the development board connected to the system, and allows you to program ROM memory in the FPGA board to permanently store the code, or to temporarily program the FPGA with the code. When programmed temporarily, FPGA runs the code as long as it is supplied with power. Once power disconnected, FPGA reverts back to being a blank IC, waiting to be programmed again.

Key words: Pulse Width Modulation (PWM), Pulse Position Modulation (PPM), Verilog HDL, ModelSim, FPGA.

Cite this Article: Shankar A, Lokeshwari A, Manohar B and Pranavi S, Implementation of Simple PWM/PPM Generator for Microcontroller Using Verilog, *International Journal of Advanced Research in Engineering and Technology (IJARET)*, 12(3), 2021, pp. 97-102.
http://www.iaeme.com/IJARET/issues.asp?JType=IJARET&VType=12&IType=3

1. INTRODUCTION

In Power Electronics. Pulse Width Modulation (PWM) is the core for control and has proven effective in driving modern semiconductor power devices. Majority of power electronic circuits are controlled by PWM signals of various forms. (1) PWM is effective and commonly used as control technique to generate analog signals from a digital device like a microcontroller.

The PWM/PPM Generator converts analog input signals to Pulse Width Modulated (PWM) or Pulse Position Modulated (PPM) output signals. Pulse Width Modulation (PWM) Signal is a method for generating an analog signal using a digital source. (1) Now-a-days microcontrollers support PWM outputs. Most of the microcontrollers will have built in timers which helps in generation of PWM signal with various widths. It is commonly used to control average power delivered to a load, motor speed control, generating analog voltage levels and for generating analog waveforms.

In PPM, an analog input signal is sampled a pulse whose position is proportional to the input signal amplitude is generated at each sample point. (2) Both PWM and PPM signals are of constant height (amplitude), and the pulses in PPM signals are of constant width. Although PWM and PPM are more complex forms of message processing than PAM, they still are not considered true digital signals.

Most of the microcontrollers have special pins assigned for PWM as in Arduino-UNO it has 6 PWM pins on it. Similarly, PIC Microcontrollers also have PWM pins but unfortunately 8051 Microcontroller doesn't have this luxury means there's no special PWM pins available in 8051 Microcontroller. But PWM is necessary so we are manually generating the PWM pulse using Verilog. CCP Modules are available with a number of PIC Microcontrollers.

This project demonstrates how a simple and fast pulse width modulator (PWM)/pulse position modulator (PPM) generator can be implemented using Verilog programming. (5) It is simulated using ModelSim, a multi-language (hardware description language) simulation environment from Mentor Graphics and tested on Basys 2 FPGA development board from Digilent. (6)

2. PULSE WIDTH MODULATOR (PWM) / PULSE POSITION MODULATOR (PPM) GENERATOR

The PWM/PPM Generator converts analog input signal to Pulse Width Modulation (PWM) or Pulse Position Modulation (PPM) output signals. In PWM, an analog input signal is sampled, and a pulse whose width (duration) is proportional to the input signal amplitude is generated at each sample point. In PPM, an analog input signal is sampled and a pulse whose position is proportional to the input signal amplitude is generated at each sample point. Both PWM and PPM signals are of constant height (amplitude), and the pulses in PPM signals are of constant width. Although PWM and PPM are more complex forms of message processing than PAM, they still are not considered true digital signals.

Using the PWM/PPM Generator, students can gain an understanding of how PWM and PPM signals are generated. The noise resistance characteristics of PWM/PPM signals can be studied also.

3. PRINCIPLE OF GENERATOR

The Pulse Width Modulation (PWM) and Pulse Position Modulation (PPM) Generators works on two principles one is a counter and a comparator were modulated signal and sawtooth signal form the input to the comparator for analog signal input and digital equivalent of analog input for digital signals. (1) One input of the comparator is fed by the input message or modulating signal and the other input as the analog or digital input which operates at carrier frequency.

The counter produces the carrier frequencies to the comparator and plays a major role in Pulse Width Modulation and Pulse Position modulation during the generation process. In this the message Bits are encoded by transmitting a single pulse in one possible required time shifts into the Microcontroller.

4. PRINCIPLE OF FPGA

Principle for PWM generator can be realized using Basys 2 development board. The board is built around Xilinx Spartan-3E FPGA (9) and Atmel AT90USB2 USB controller. It provides complete, ready-to-use hardware suitable for hosting circuits, ranging from basic logic devices to complex controllers. (4) This board is used, along with the following two software, to allow for effortless programming and debugging of the FPGA board:

- Xilinx ISE Webpack 14.7
- Digilent Adept v2.0 or higher.

Digilent Adept v2.0 or higher program allows USB programming through computer USB port. The program, when runs, automatically detects the development board connected to the system, and allows you to program ROM memory in the FPGA board to permanently store the code, or to temporarily program the FPGA with the code. When programmed temporarily, (5) FPGA runs the code as long as it is supplied with power. Once power disconnected, FPGA reverts back to being a blank IC, waiting to be programmed again. Adept software requires a bitmap file with. bit extension. This is usually created using synthesizer software like ISE from Xilinx or Simplify from Synopsys for logical synthesis. (6) This is where Xilinx ISE webpack is used, is a free software that can be used by anyone. Include the Verilog file into a new project along with a user constraints file (ucf extension).

5. WORKING OF GENERATOR FOR MICROCONTROLLER

The working of the PWM/PPM Generator is simple. It uses one counter and one comparator. The microcontroller unit provides 8-bit input into PWM module. Counter used in the PWM module is 8-bit. It increments it's value on the positive edge of the clock (positive edge

triggered). Comparator used in PWM generator is also 8-bit. Input given to PWM module is compared to the current value of the counter using the comparator. If current value of the counter is greater than the value given to the module as input, PWM output is pulled low. However, if current value of the counter is less than the value given as input to the module, output of PWM generator is pulled high.

Let us take a test case to understand the operation. Suppose, input to PWM module is 128 (10000000_b). Counter is initialized using Reset button, so that output is 0(00000000_b). During the first clock cycle, value of counter and input value to PWM module are compared.

The module finds that, value of counter is less than the value provided. This prompts the module to pull output of PWM generator high. This same paradigm keeps output of the generator pulled high until value of the counter reaches 128 (10000000_b).

In the next clock cycle, after the counter has reached 128, counter increments its value to 129 (10000001_b). Comparator finds that, current counter value is greater than the value provided to PWM module, therefore output must be pulled low.

Changing input to PWM module will consequently change the threshold value, where transition from high to low state occurs. Thus, PWM can be achieved by varying the input provided to PWM

module. When the analog input value is greater than the counter value it is sent to the d flip flop which stores the greater than value. PPM clock (2) is another input given to the d flip flop. After some delay they output of the d flip flop is given to the and gate and when the input is less than or equal to counter output it is given as the second input to the and gate so that during the negative fall of PWM output from high to low then ppm generates the pulse.

6. BLOCK DIAGRAM

Block diagram of the PWM and PPM generator is shown in Fig. Working principle of the generator is simple. It uses one counter and one comparator. The microcontroller unit provides 8-bit input into PWM module. Counter used in the PWM module is 8-bit. It increments it's value on the positive edge of the clock (positive edge triggered). Comparator used in the PWM generator is also 8-bit.

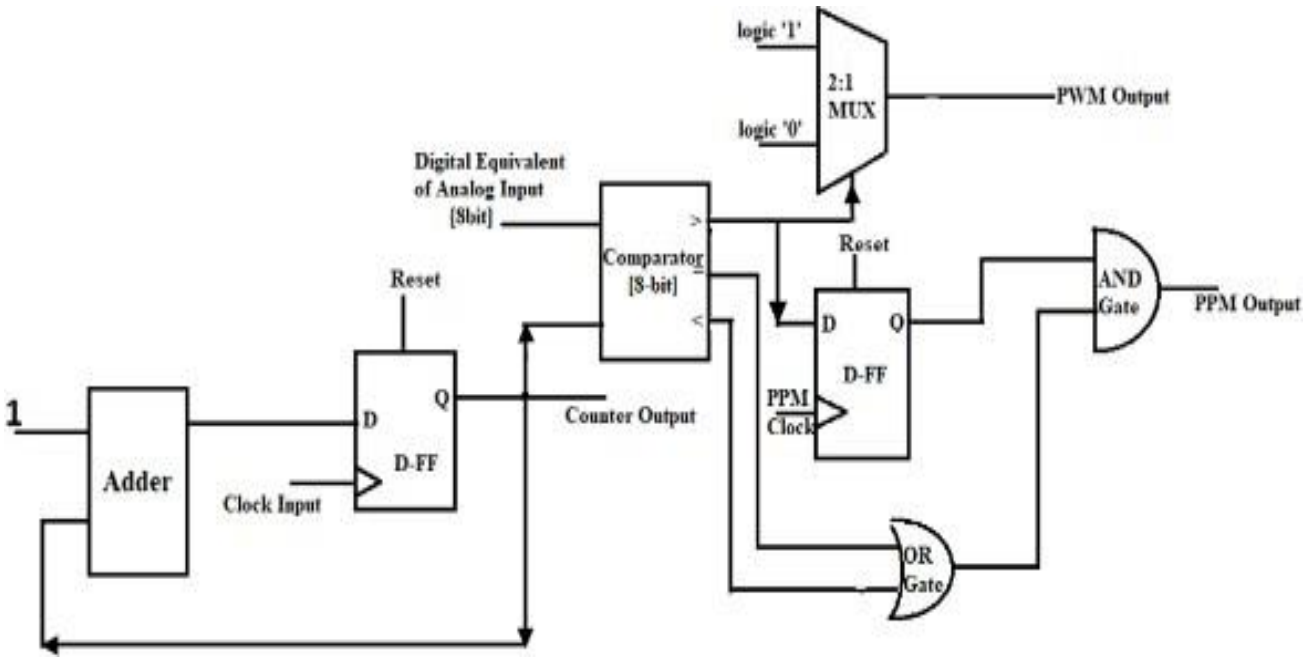

Figure 1 Block diagram of PWM/PPM generator.

In the PWM generator is also 8-bit. (4) Input given to PWM module is compared to the current value of the counter using the comparator. If current value of the counter is greater than the value given to the module as input, PWM output is pulled low. However, if current value of the counter is less than the value given as input to the module, output of PWM generator is pulled high. When the analog input value is greater than the counter value it is sent to the d flip flop which stores the greater than value. Clock 2 is another input given to the d flip flop. After some delay they output of the d flip flop is given to the and gate and when the I/p is less than or equal to counter output it is given as the second input to the and gate so that during the negative fall of PWM output from high to low then PPM generates the pulse.

7. RESULT AND DISCUSSION

Figure 2

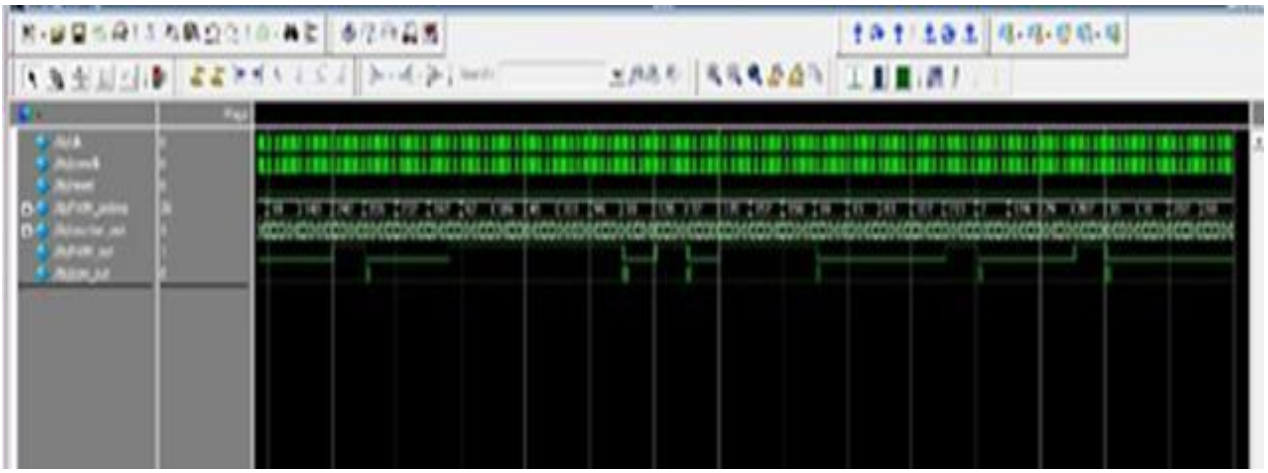

Figure 3 PWM/PPM wave-forms

- This PWM has a maximum operating frequency of 232.504MHz, which allows the most dynamic and fast operation applications to be accommodated.
- Total power dissipated by the design at an operation frequency of 50MHz is 38mW.
- Static power dissipation for 34mW (power required to drive the development board) and the power consumed by the oscillator to produce the clock is 1mW.
- Average power consumed by PWM output is 3mW, which indicates that the design is also power efficient.

8. CONCLUSION

This project demonstrates how a simple and fast pulse width modulator (PWM)/Pulse Position Modulator (PPM) generator can be implemented using Verilog programming. It is simulated using ModelSim, a multi-language (hardware description language) simulation environment from Mentor Graphics and tested on Basys 2 FPGA development board from Digilent.

REFERENCES

[1] R.M. Jalanekar & K.S. Jog, "Pulse –Width-Modulation Techniques: A Review," IETE journal of research, Vol. 46, No. 3, May-June 2000.

[2] O. Cadenas, G. Megson, "A clocking technique for FPGA pipelined designs," Journal of Systems Architecture 50, 2004, pp. 687–696.

[3] M.M. Islam, D. Allee, S. Konasani, A. Rodriguez, "A low-cost digital controller for a switching DC converter with improved voltage regulation," IEEE Power Electronics Letters 2, 2004, pp. 121–124.

[4] A. Arbit, D. Pritzker, A. Kuperman, R. Rabinovici, "A DSP-controlled PWM generator using field programmable gate array," 23rd IEEE Convention of Electrical and Electronic Engineers, vol. 1, 2004, pp. 325– 328.

[5] Brian von Herzen, "Signal processing at 250 MHz using high-performance FPGA s," IEEE Transactions on VLSI Systems 6, 1998, pp. 238–246.

[6] R. Ramos, D. Biel, E. Fossas, F. Guinjoan, "A fixed- frequency quasi sliding control algorithm: application to power inverters design by means of FPGA implementation," IEEE Transactions on Power Electronics 18, 2003, pp. 344–355.

[7] A.M. Omar, N. Rahim, S. Mekhilef, "Three-phase syn- chronous PWM for flyback converter with power-factor correction using FPGA ASIC design," IEEE Transactions on Industrial Electronics 51, 2004 pp. 96- 106.

[8] Koutroulis, Eftichios, Apostolos Dollas, and Kostas Kalaitzakis. "High frequency pulse width modulation implementation using FPGA and CPLD ICs." Journal of Systems Architecture 52.6, 2006, pp. 332-344.

[9] Nouman, Z. "Generating PWM Signals with Variable Duty From 0% to 100% Based FPGA SPARTAN3AN." Electrorevue Journal, International Society for Science and Engineering publications 4.4, 2013, pp. 75-79.

[10] Youichi Ito and Scoichi Kawauchi, "Microprocessor – Based Robust Digital control for UPS with three phase PWM Inverter," IEEE Transactions on Power Electronics, Vol.10, No.2, March 1995.