

LAB #7: PWM WAVEFORM GENERATION

INTRODUCTION

There are many devices that can be controlled by periodically pulsing one or more of their control signals. For example, stepper motors rotate some fixed amount for each pulse applied to their inputs. The intensity of a light or the speed of a D.C. motor can be controlled with pulse-width modulated (PWM) digital waveforms, with the light/motor effectively controlled by the average voltage of the signal. A PWM waveform is a periodic signal comprising pulses of variable duration (width). For a constant period, variations in the pulse width can create different desired effects.

The purpose of this lab is to generate a PWM signal waveform, with keypad-selectable pulse widths. This will be used in the next lab as a control signal to drive a D.C. motor at one of several keypad-selectable speeds.

PWM SIGNAL CHARACTERISTICS

Figure 1 illustrates three PWM signals, and the two key parameters of a PWM signal. Each waveform is characterized by two parameters:

$$T = T_1 + T_2 \quad (1)$$

and

$$\text{duty cycle} = T_1/T \quad (2)$$

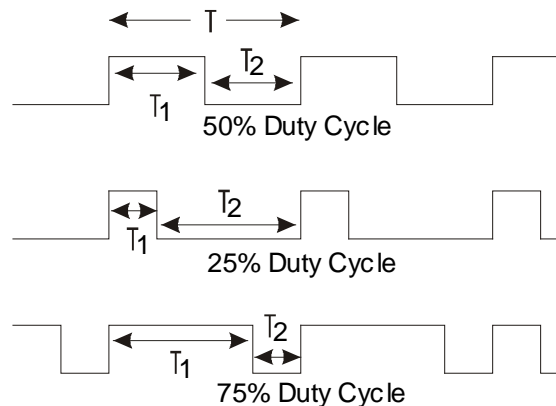


Figure 1. Pulse-width modulated (PWM waveforms)

Here, T is the *period* of the waveform, with the signal high for time T_1 and low for time T_2 . The *duty cycle* of the waveform is defined as the high time divided by the period: $T_1/(T_1 + T_2) = T_1/T$. The three waveforms in Figure 1 have duty cycles of 50%, 25%, and 75%, as indicated.

The term “pulse-width modulation” refers to the alteration (modulation) of the high time, T_1 , while maintaining a constant period, T . In the next lab, we will drive a D.C. motor with an amplified PWM signal, varying its duty cycle to control the speed of the motor. The rationale for using pulse width modulation will be motivated at that time by examining the amplifier efficiency.

PWM SIGNAL GENERATION

In this lab, PWM signals are to be generated by the programmable timer module utilized in the previous lab.

Overviews of the timer module are given in the Monday presentation slides for this lab and the previous lab, posted on the class web page, the STM32L1xx Reference Manual (chapter on general-purpose timers TIM9/10/11), plus ELEC 2220 course slides on timers, also linked to the class web page

HARDWARE AND SOFTWARE DESIGN

You are to implement a variable duty cycle PWM waveform generator with keypad-selectable duty cycles. Keypad keys A, 9, 8, ..., 0 are to select one of eleven duty cycle values for the PWM waveform: 100%, 90%, 80%, 70%, 60%, 50%, 40%, 30%, 20%, 10%, and 0% (stopped), respectively. The selected duty cycle number (10, 9, ..., 0) should be displayed on four LEDs. When waveform generation is stopped, the signal is to be in the low state. (*HINT – Consider placing counter register values for the ten non-zero duty cycles in an array, with the number corresponding to the pressed keypad button used as an index into that array.*)

A C program is to be written to produce the variable duty cycle PWM signal using programmable timer module TIM10 and output pin PA6. For this lab, the initial PWM signal period T should be 1 msec. Hence, the resulting PWM signal is said to have a “1 kHz switching frequency.”

PRE-LAB ASSIGNMENT

Prior to the laboratory, design the C program to meet the specifications described above in the section “Hardware and Software Design.” In your laboratory notebook, record the following (prior to lab).

1. Flowcharts for setting up the programmable timer in such a way that a PWM signal is produced, i.e. setting up the appropriate registers and flags.
2. Draft program (or directions to your content on H:drive)
3. A plan for testing the PWM generator

LABORATORY EXPERIMENTS

1. Verify that your hardware is operating properly by using test programs from previous labs, or some other brief program that exercises the hardware.

2. Download and run your PWM waveform generation program, displaying the generated waveform on the oscilloscope. Demonstrate to the lab instructor that you can use the keypad to select the ten duty cycles and the stop condition. Also show that the stop condition leaves the signal in the low state.
3. Measure and record the duty cycle for each of the eleven keypad buttons (0-A), and plot the measured values on a graph (vs. button number) to ensure a linear increase in duty cycle with keypad selection.
4. Repeat steps 2 and 3, changing the PWM signal frequency to 100 Hz. If time permits, also try 10 KHz.
5. Use watch variables and the logic analyzer to trace the flow of your program. For example, if the program does not appear to be responding to certain keypad entries, capture I/O port information to determine the sequence of instructions executed, the values read from the keypad, etc.

DELIVERABLES

Refer to the schedule on the course web page for this week's deliverables.

POSSIBLE INFORMATION FOR FUTURE LABORATORY REPORTS

1. Briefly describe the circuit (but not "wire by wire") and the test programs (include a circuit diagram and C program source listings).
2. Discuss your results. For each program, include a table and graphical plot of measured duty cycles, and at least one image of the oscilloscope or logic analyzer screen that shows the generated PWM signal.
3. Discuss what you had to change to generate PWM signals with different frequencies. In the next lab, be prepared to try different PWM signal frequencies to determine which produces the most linear change in motor speed, proportional to the selected duty cycle values.