ECE3411 – Fall 2015

Lab 5a.

PWM: Pulse Width Modulation

Marten van Dijk, Syed Kamran Haider
Department of Electrical & Computer Engineering
University of Connecticut
Email: {vandijk, syed.haider}@engr.uconn.edu

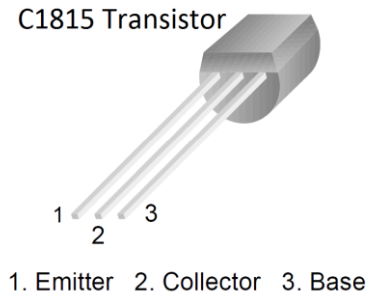
UCONN

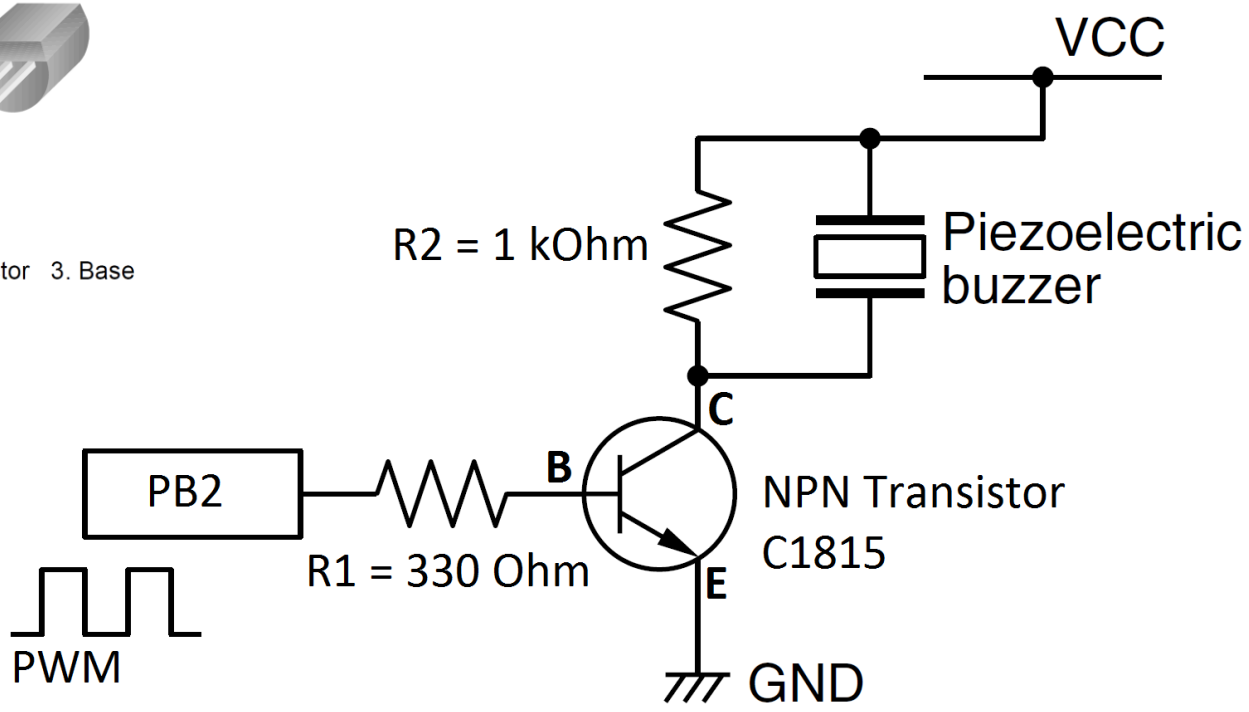
Using Timer1 for PWM

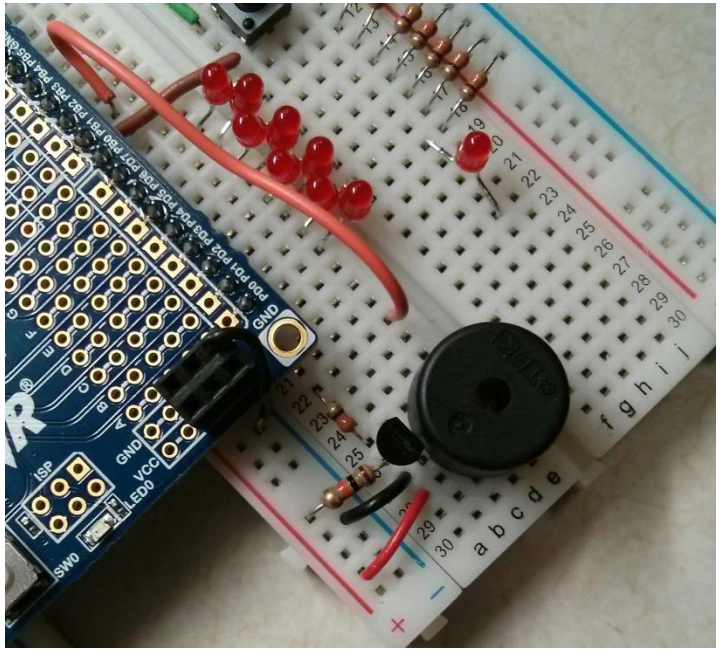
- We'll be using Timer1 in Fast PWM Mode with OCR1A as TOP
  - I.e. (WGM13, WGM12, WGM11, WGM10) = (1,1,1,1)
- PWM signal is output at pin OC1B which is PB2 on ATmega328P
- OCR1A controls the frequency of the resulting PWM signal
  - Every compare match A starts a new cycle of PWM waveform, i.e. PWM pin is 'set'.
- OCR1B controls the duty cycle of the resulting PWM signal
  - At every compare match B, the PWM signal is 'cleared' (non-inverting mode).
- Compare Match A ISR can be used to update OCR1A and OCR1B registers
  - OCR1A changes frequency and OCR1B changes duty cycle.
- If PWM signal is needed at additional pin(s):
  - Compare Match A ISR can be used to 'set' that pin(s).
  - Compare Match B ISR can be used to 'clear' that pin(s).

Connecting the Buzzer

- In this lab, we'll be using a buzzer that'll be driven by a PWM signal.
- Connect the buzzer according to the following schematic.







Task1: Low Frequency PWM Signal

Use Timer1 to generate a PWM signal on ***PB2 and PORTD*** such that:

- The PWM signal has a frequency of 1Hz
- While Switch 1 is pressed, the duty cycle of the PWM signal is gradually increased (say in steps of 5%) up to 100%
- While Switch 2 is pressed, the duty cycle of the PWM signal is gradually decreased (say in steps of 5%) down to 0
- Print the current duty cycle on LCD.

You don't need to debounce the switches.

Hint: Use Compare Match A and B ISRs to generate PWM on PORTD in software.

Connect a pair of a LED and a current limiting resister (330 Ohm) to PB2.

Now you should be able to observe the LEDs' blinking behavior with different duty cycles.

Task2: LED Brightness Control

Use Timer1 to generate a PWM signal on ***PB2 and PORTD*** such that:

- The PWM signal has a frequency of 1000Hz
- While Switch 1 is pressed, the duty cycle of the PWM signal is gradually increased (say in steps of 1%) up to 100%
- While Switch 2 is pressed, the duty cycle of the PWM signal is gradually decreased (say in steps of 1%) down to 0

You don't need to debounce the switches.

Hint: Use Compare Match A and B ISRs to generate PWM on PORTD in software.

Connect a pair of a LED and a current limiting resister (330 Ohm) to PB2.

Now you should be able to observe the LEDs' brightness behavior with different duty cycles.

Task3: The Ambulance Siren

Using Timer1 PWM generation, implement an Ambulance Siren.

In particular, implement the following:

- Using Timer1, generate a PWM signal at PB2 with 25% duty cycle.
- Using Timer0, gradually vary the frequency of the PWM signal from 1kHz to 4kHz, and then from 4kHz back to 1kHz and so on.
  - I.e. Timer 0 overflows every ~16ms → ~120 overflows in 2 seconds.
  - Each overflow changes the OCR1A value by 100 ticks.
- Notice that the duty cycle remains 25% for each frequency.
- Connect a Buzzer to the PWM signal as shown in earlier slides.

Now you should hear an ambulance siren.

Play with the frequency ranges and frequency update rates to transform the ambulance siren into a *Cop Car Siren* ☺