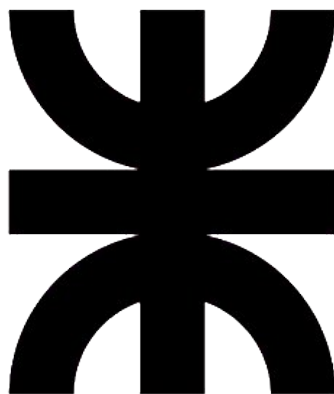


UNIVERSIDAD TECNOLÓGICA NACIONAL



FACULTAD REGIONAL PARANÁ

Carrera: Ingeniería electrónica

Cátedra: Técnicas Digitales II

**Trabajo Práctico N° 10: Control de intensidad de LED con
PWM**

PROFESORES:

Caballero, Raúl Manuel
Maggiolo, Gustavo Daniel
Britos, Rubén Adrián

INTEGRANTES:

Battaglia, Carlo Ignacio
Escobar, Gabriel Hernán

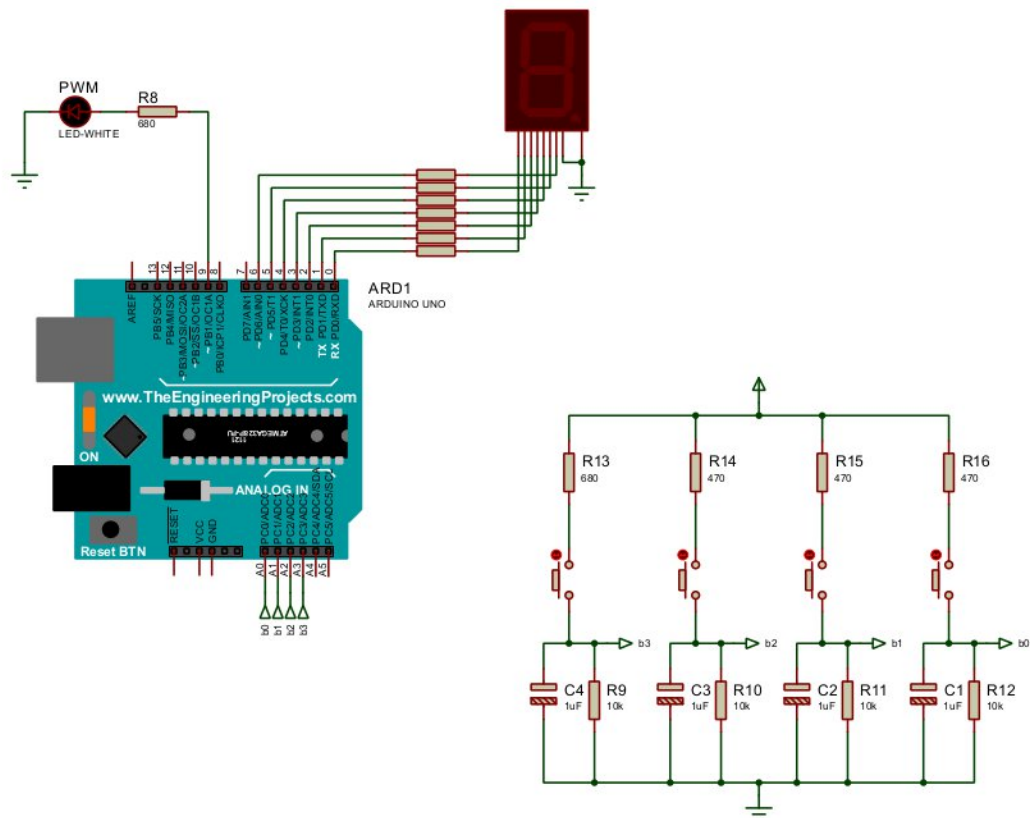
Fecha de entrega: 28/09
Año Lectivo: 2022

Actividad

Se desea que el alumno sea capaz de desarrollar e implementar un sistema completo en la placa Arduino, trabajando con el entorno Microchip Studio 7.0, en ensamblador. Debiendo ser realizado con todos los puntos citados debajo. Se deberá entregar un informe, donde describa el funcionamiento general del sistema implementado; como así también: el código, los esquemas del circuito y el programa en ensamblador.

- 1. El circuito “base” será el utilizado en el trabajo práctico Nro 8.**
- 2. El circuito deberá tener, además:**
 - a. Un LED, de alto brillo, conectado convenientemente a una salida PWM de la placa Arduino.**
- 3. El funcionamiento general del circuito es:**
 - a. El sistema al encendido deberá estar todo apagado.**
 - b. De acuerdo con el valor que se ingrese con los 4 pulsadores, deberá cambiar la intensidad lumínica del LED.**
 - c. Deberá tener 16 valores posibles de intensidad lumínica. Con todos 0 el LED apagado y con todos 1 el LED a máxima intensidad.**
 - d. En el display se deberá indicar que valor de intensidad se está generando (valor de 0h a Fh).**

El circuito implementado es el siguiente:



Para implementar el sistema requerido, elegimos el *Timer/Counter1* de 16 bits que posee el *ATmega328p* para hacer uso de su función *Fast PWM*.

Configuramos este modo de manera que el ciclo de trabajo de la señal PWM esté dada por el valor almacenado en el registro *OCR1A* y el valor máximo del contador está dado por el registro *ICR1*. Esto nos permite elegir la resolución requerida en el enunciado (16 valores) fijando el máximo en 15 a través de *ICR1*.

Además, para cumplir con la condición de obtener una señal completamente nula al ser el ciclo de trabajo igual a 0, configuramos el modo *Fast PWM* para hacer uso de la señal en modo invertido, tal como se ve en la siguiente tabla:

COM1A1/COM1B1	COM1A0/COM1B0	Description
0	0	Normal port operation, OC1A/OC1B disconnected.
0	1	WGM13:0 = 14 or 15: Toggle OC1A on compare match, OC1B disconnected (normal port operation). For all other WGM1 settings, normal port operation, OC1A/OC1B disconnected.
1	0	Clear OC1A/OC1B on compare match, set OC1A/OC1B at BOTTOM (non-inverting mode)
1	1	Set OC1A/OC1B on compare match, clear OC1A/OC1B at BOTTOM (inverting mode)

Esto es debido a que en el caso particular en que *OCR1A* es seteado al mismo valor que *BOTTOM* (0x0000) la salida no será una señal nula como habría de esperarse, sino que será un pico angosto con un ancho de un ciclo de timer cada vez que se alcance *TOP+1*.

Al tomar esta decisión de diseño, debimos tener en cuenta que los valores tomados directamente de los pulsadores leyendo el registro *PINC* estarán invertidos a fines prácticos, por lo que han de ser tratados mediante una operación *XOR* con un valor de 0x0F antes de poder ser cargados en el registro *OCR1A*.