**3\_4\_PWM**



**Programación De Sistemas Embebidos**

Mecatrónica 8°A

**Maestro**: Moran Garabito Carlos

**Alumno:**

* Eduardo Robles Vázquez

**3\_4\_PWM**

**OBJETIVOS**: El alumno deberá realizar la programación de la tarjeta CY8CKIT-059 PSoC para controlar un servomotor en base a las siguientes condiciones:

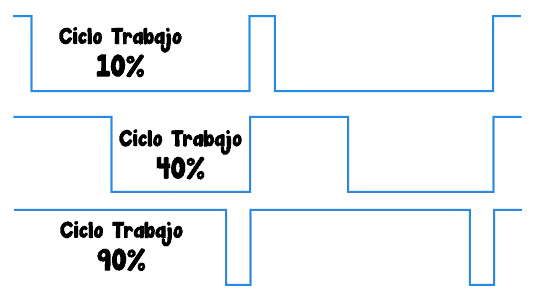
1. Al presionar el botón 1, aumentará en 10 grados la posición del servomotor.
2. Al presionar el botón 2, disminuirá en 10 grados la posición del servomotor.
3. Al presionar el botón 3, se efectuará el movimiento en servo motor con los grados marcados con anterioridad.
4. Mostrar en la LCD los grados de posición del servomotor.

**MARCO TEÓRICO:**

**PWM**

PWM son siglas en inglés que significan Pulse Width Modulation y que lo podemos traducir a español como Modulación de ancho de pulso. La modulación de ancho de pulso está formada por una señal de onda cuadrada que no siempre tiene la misma relación entre el tiempo que está en alto y el tiempo que está en bajo.

En la siguiente imagen vemos una señal que varía entre 5 voltios y 0 voltios. A lo largo del tiempo la señal varía entre dos valores de tensión. Durante un tiempo determinado la señal se encuentra en el nivel alto (en este caso 5v) y durante otro periodo de tiempo se encuentra en el segundo valor de tensión (en este caso 0v).



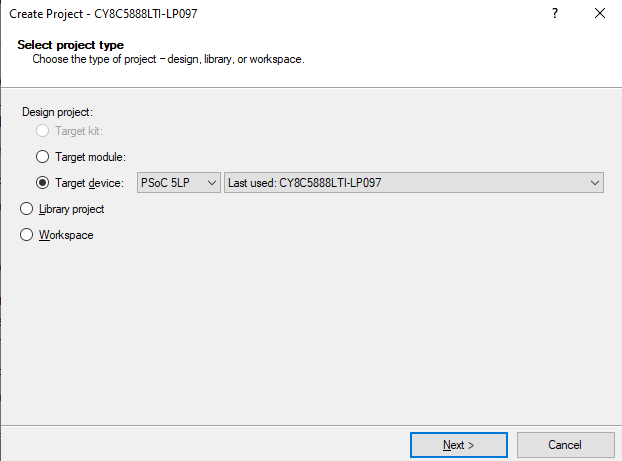
El tiempo que la señal se encuentra en el nivel alto (5 v) lo denominamos como tiempo on (Ton) mientras que el tiempo que está en nivel bajo lo denominamos tiempo off (Toff). La suma del tiempo on y el tiempo off es el periodo de la señal (T). Y como en toda señal periódica, el inverso de del periodo (1 / T) es la frecuencia de la señal.

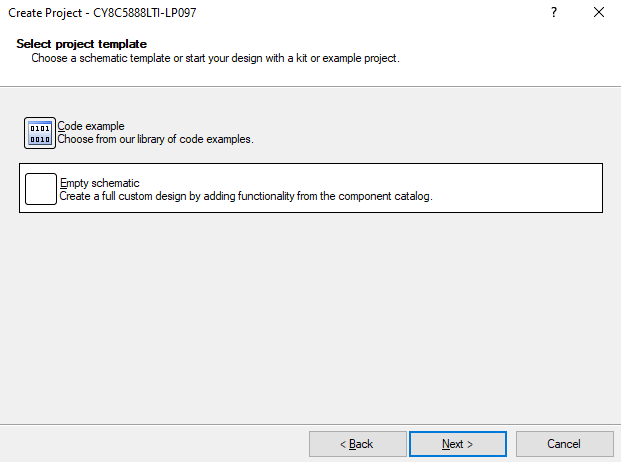
**MATERIALES:**

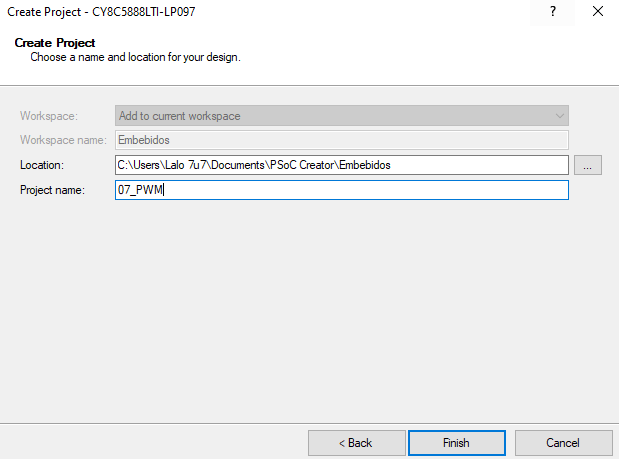
1. Computadora con software PSoC Creator.
2. Tarjeta CY8CKIT-059 PSoC
3. Potenciómetro
4. Resistencias
5. LCD
6. Servomotor
7. Botones

**PROCEDIMIENTO:**

1. En el software PSoC Creator crear un nuevo proyecto con las características necesarias para programar la tarjeta CY8CKIT-059 PSoC.





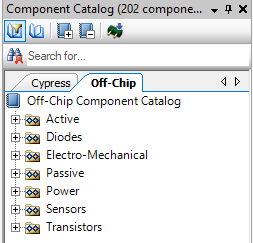


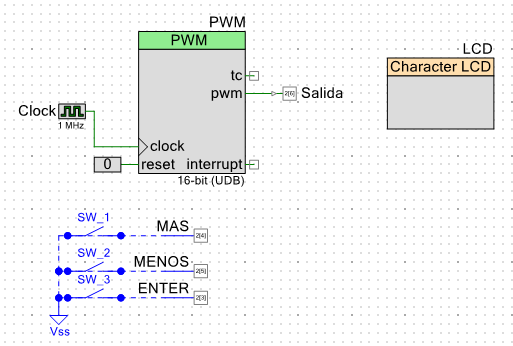
1. Insertar los componentes necesarios para su posterior uso. Una vez hecho esto se puedo construir el proyecto sin problema alguno.

* Primero ingresamos a TopDesign a través del menú que se encuentra en la izquierda.

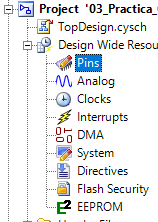
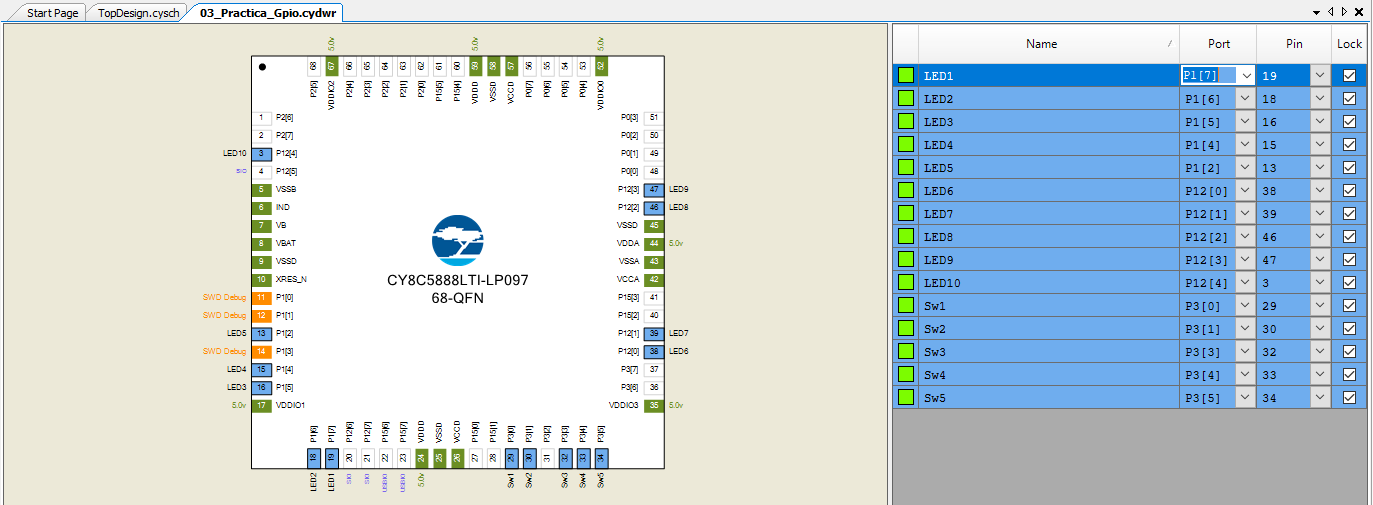


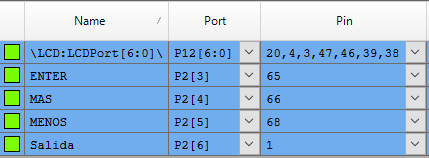
* En el menú que se encuentra a la derecha podremos escoger los componentes que necesitemos.



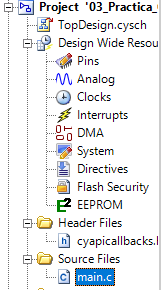


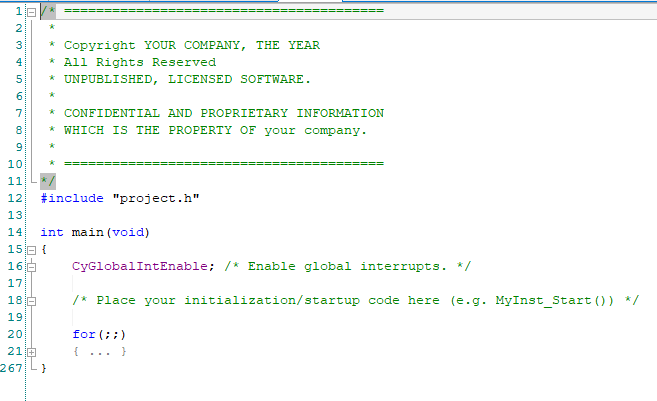
1. Definir los pines que usaremos para cada componente. Este lo haremos ingresando al menú dando clic en Pins.





1. Procedemos a realizar la programación ingresando al Main.c



* Cuando ingresamos al main se nos muestra la siguiente ventana donde escribiremos nuestro programa dentro del For.
* A continuación, se muestra un fragmento del código utilizado:

#include "project.h"

int main(void)

{

PWM\_Start();

LCD\_Start();

LCD\_ClearDisplay();

CyGlobalIntEnable;

float x=1500;

float y=111;

float z=(((x-500)\*180)/2000)+.3;

PWM\_WriteCompare(x);

for(;;)

{

LCD\_ClearDisplay();

LCD\_Position(0,5);

LCD\_PrintString("Grados");

LCD\_Position(1,7);

LCD\_PrintNumber(z);

CyDelay(500);

if(MAS\_Read()){

x = x + y;

z=(((x-500)\*180)/2000)+.3;

CyDelay(500);

}

if(MENOS\_Read()){

x = x - y;

z=(((x-500)\*180)/2000)+.3;

CyDelay(500);

}

LCD\_ClearDisplay();

LCD\_Position(0,5);

LCD\_PrintString("Grados");

LCD\_Position(1,7);

LCD\_PrintNumber(z);

CyDelay(500);

if(ENTER\_Read()){

if(x<500){

x=1500;

z=(((x-500)\*180)/2000)+.3;

}

if(x>2500){

x=1500;

z=(((x-500)\*180)/2000)+.3;

}

PWM\_WriteCompare(x);

LCD\_ClearDisplay();

LCD\_Position(0,5);

LCD\_PrintString("Grados");

LCD\_Position(1,0);

LCD\_PrintString("-------OK-------");

CyDelay(1000);

}

LCD\_ClearDisplay();

LCD\_Position(0,5);

LCD\_PrintString("Grados");

LCD\_Position(1,7);

LCD\_PrintNumber(z);

CyDelay(500);

//PWM\_WriteCompare(2500);

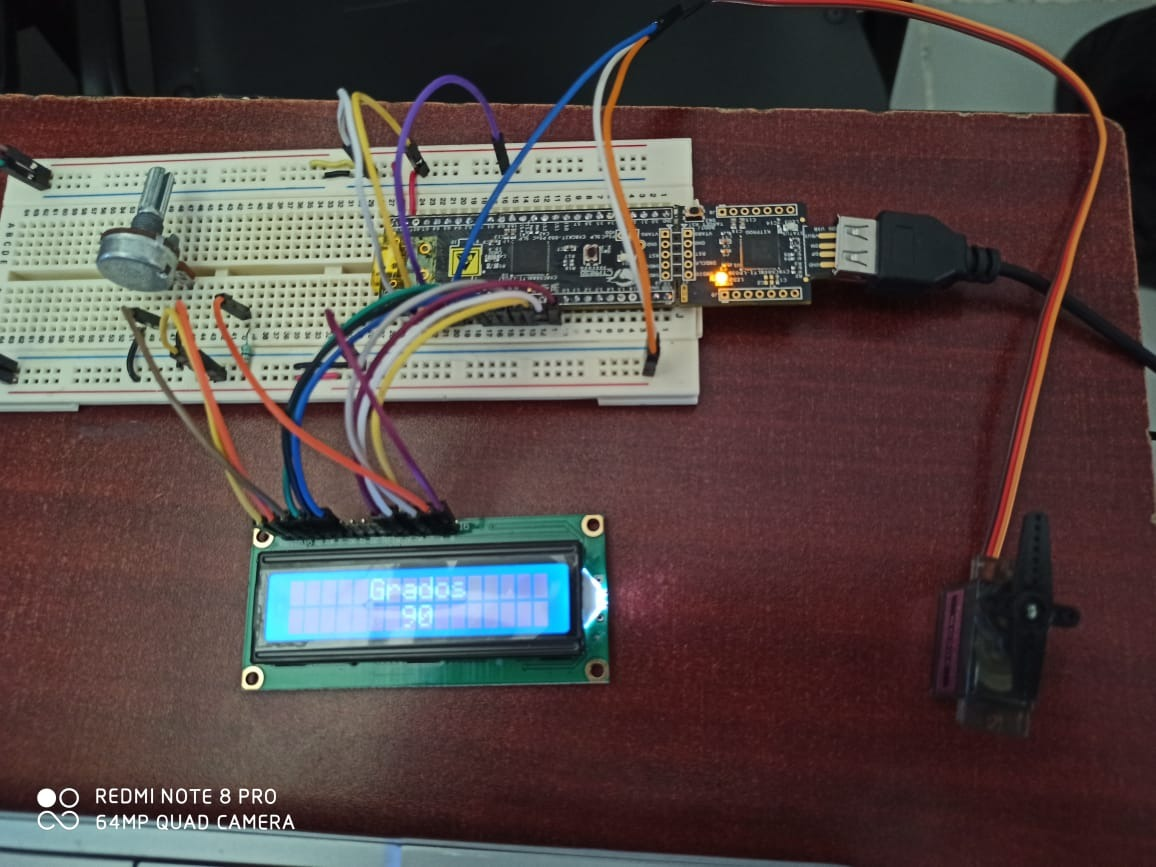
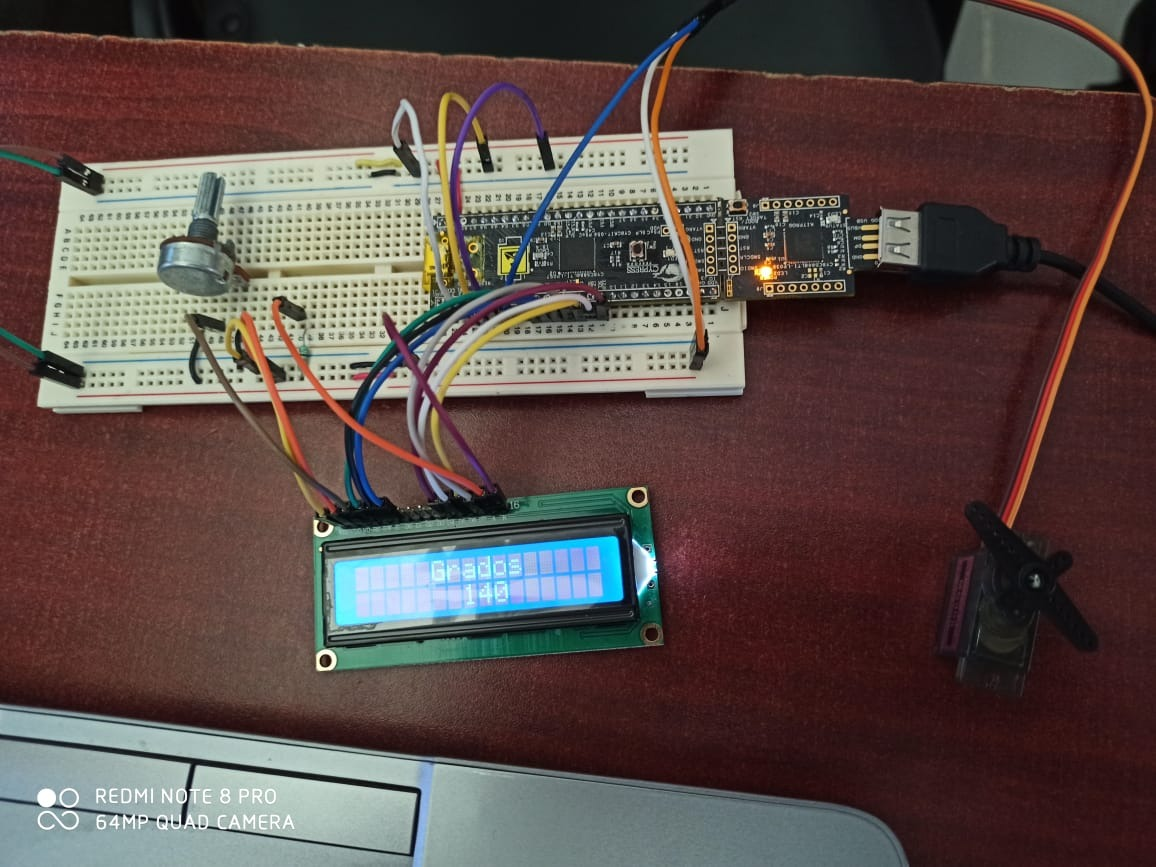
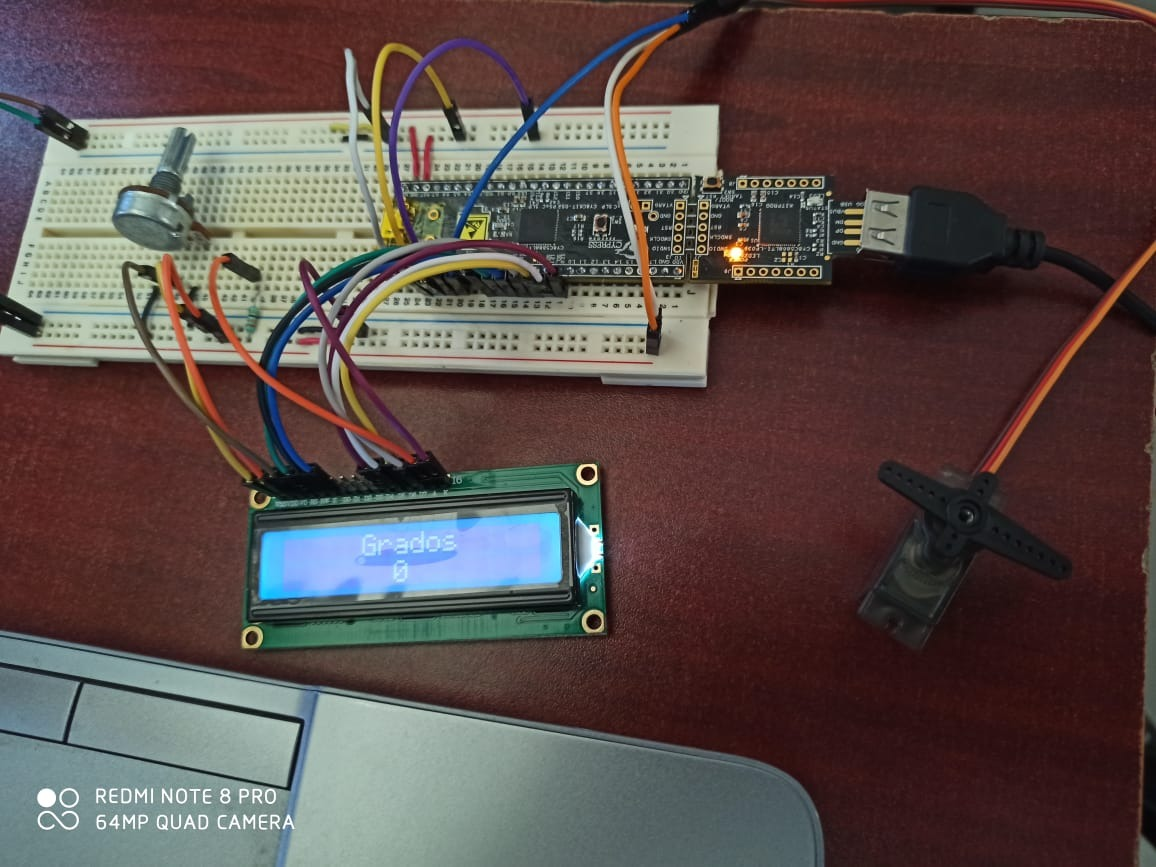
}

return 0;

}

1. Programamos la tarjeta para proceder a ver los resultados.

**Resultados:**



**CONCLUSIÓN:**

Esta practica fue interesante y honestamente fue al comienzo muy difícil para mí, puesto que no comprendía por completo lo que era un PWM, pero una vez que asimile mejor la información, esto permitió que la practica se facilitara. Es importante seguir desarrollando practicas de este estilo para poder aterrizar los conocimientos teóricos lo más posible.