



## Lab 3: PWM y ADC

### 1. Objetivo

Este laboratorio busca introducir a los estudiantes al manejo de dos nuevas operaciones de entrada/salida. La primera de estas corresponde a la generación de PWM o *Pulse Width Modulation*, la cual es una funcionalidad de *output* que permite generar salidas pseudo-analógicas en el rango de 0 a  $V_{CC}$  Volts. De esta forma, permite llevar a cabo una gran variedad de aplicaciones, como el control de velocidad motores, el brillo de LEDs o de *displays* LCD. La segunda de estas operaciones corresponde al Conversor Análogo Digital o ADC. Esta es una funcionalidad de *inputs* que permite digitalizar una señal de entrada analógica de voltaje en un pin.

A modo de concretar el objetivo antes señalado, deberán comprender como llevar a cabo estos modos de operación a partir de la documentación presente en los correspondientes *datasheets*. Una vez comprendida a cabalidad la operación de estas funcionalidades, deberán hacer uso del ADC para medir los niveles de temperatura y luminosidad sensados por un NTC y un fotorresistor. Ya digitalizados estos datos, deberán representar las mediciones obtenidas mediante señales PWM que controlen el brillo de un LED y el color de un LED RGB. Esta experiencia estará separada en **2 tareas distintas, una para cada microcontrolador**. La elección del microcontrolador a utilizar en cada una de ellas es libre (evidentemente, si usa un MCU en la primera actividad no puede usarlo en la segunda).

### 2. Descripción de la actividad

#### **Task 1: Termómetro Digital**

Para la siguiente actividad deberá hacer uso del termistor NTC 10k y el LED RGB que se le han entregado en su kit de SEP. La actividad consiste en la lectura y digitalización del voltaje medido en el termistor ante variaciones de temperatura, y su posterior conversión a una escala de temperaturas representada por un gradiente de color de rojo (máximo) a azul (mínimo) en el LED RGB.

Para concretar esta tarea deberá realizar un divisor resistivo, el cual le permita percibir variaciones de voltaje en el termistor y utilizar la funcionalidad ADC del microcontrolador escogido para medir y convertir dicho valor. Una vez realizado esto, deberá alimentar los canales Rojo y Azul de su LED RGB mediante salidas PWM con ciclos de trabajo proporcionales a la temperatura medida.

Para estandarizar la medición, deberán hacer que su programa funcione dentro del rango de 0 a 40° C, para conocer la resistencia que tiene el NTC a las temperaturas dentro de este rango. Rangos superiores o inferiores deberán ser ignorados.

**Importante:** La escala de colores generada en el RGB debe ir desde Azul (mínimo, 0° C) a Rojo (máximo, 40°) en un continuo, es decir, pasando por los distintos tonos intermedios (violeta, morado, magenta u otra tonalidad que usted sea capaz de distinguir y nombrar).



Figura 1: Escala continua para Termómetro RGB

## **Task 2: Sensor de iluminación**

La siguiente actividad corresponde a la utilización del fotorresistor entregado y un LED a su elección (puede ser rojo, verde, amarillo o el RGB). Para esto deberá controlar la intensidad del LED mediante PWM, variando la intensidad desde el umbral más bajo hasta el más alto. Esta variación deberá ser proporcional a la cantidad de luz que recibe el fotorresistor.

Para lo anterior deberán realizar su propia escala, para que pueda ser perceptible visualmente en las condiciones que se tendrán en el laboratorio. Una sugerencia es que se utilice la luz ambiente como un punto intermedio, el punto más bajo que sea al tapar el fotorresistor con la mano y que el brillo máximo sea al iluminar directamente el fotorresistor (puede ser una linterna o *flash* del celular).

**Importante:** Dada la sensibilidad a las variaciones de luz de este sensor, se recomienda trabajar durante el día en esta actividad, pues será revisada de 10:00 a 11:20 hrs en un laboratorio iluminado).



## 2.1. *Hint*

A modo de orientarlo en la ejecución y organización de su código, se adjunta el siguiente pseudo-código:

```
1  /* Puede que necesite librerias adicionales, como math.h */
2  #include librerias.h
3
4  ***Rutina principal***
5  int main(){
6      //Configuracion ADC y PWM (lea el datasheet);
7      //Lectura de la entrada;
8      //Conversion de voltaje a escala de Temperatura o Luz;
9      //Gestion de las salidas PWM;
10 }
11
12 ***Rutinas secundarias***
13 *Una buena practica es crear funciones separadas
14 del codigo principal, que posteriormente sean
15 llamadas en este, asi tendra un codigo mas ordenado
16 y sencillo de debuggear.*
17 int funcion (variables){
18     //codigo;
19 }
```

## 3. Lectura recomendada

- [ATmega328/P Complete Datasheet](#).
- [MSP430x5xx and MSP430x6xx Family User's Guide](#)
- [MSP430F552x, MSP430F551x Mixed-Signal Microcontrollers datasheet](#).
- Datasheet del [fotorresistor](#).
- Datasheet del [LED RGB](#).
- NTC Part number: NTCLE100E3103 Datasheet del [termistor NTC 10k](#).



## 4. Pauta de Evaluación

### 4.1. Consideraciones generales

- El laboratorio será evaluado exclusivamente con nota 1.0 (**R**eprobado), 4.0 (**S**uficiente), 5.5 (**A**probado) y 7.0 (**D**istinguido). En ningún caso habrán notas intermedias.
- No se reciben trabajos después del módulo de presentación. Trabajos no entregados son calificados con nota 1.0 y son considerados dentro del criterio de aprobación del curso.
- Cualquier consulta sobre los criterios de evaluación de cada laboratorio debe ser realizada en las **issues**, donde estará disponible para que sea revisada por todos los alumnos.

### 4.2. Criterios de Aprobación

Se requiere cumplir con todos los puntos mencionados a continuación para poder aprobar. No existen casos excepcionales.

Funcionamiento de los requerimientos El alumno realiza una presentación de su trabajo y se responsabiliza de exponer que este satisfaga todos los requerimientos mínimos solicitados en la *Descripción de la actividad*, los cuales incluyen en este laboratorio:

#### Requerimientos *Task 1*

- Programa compilado y ejecutándose.
- Utilización de canal rojo y azul del LED RGB.
- Variación visible de color al cambiar temperatura.
- Funciona en el rango solicitado.
- Utilización de *Timers* dedicados a la generación de PWM ( No se permite usar *delays* para simular una PWM).

#### Requerimientos *Task 2*

- Programa compilado y ejecutándose.
- Variación visible de la intensidad del LED.



- Variación cumple con lo solicitado, (Brillo bajo LED → casi nula presencia de luz en el sensor , Brillo alto LED → directa iluminación al sensor).
- Utiliza un microcontrolador distinto al *task* anterior.

## Preguntas

Se responde satisfactoriamente a 2 de 3 preguntas aleatorias al momento de la presentación final, las cuales abarcan los siguientes temas:

- Modo utilizado para generar la PWM.
- Modo utilizado para el ADC.
- Especificaciones ADC y *Timer* utilizado (cantidad de bits, modos, conversión, entre otros).
- Qué representa cada línea de código y en qué se traducen en el funcionamiento del programa.

Solo se dispone de una oportunidad para responder estas preguntas. Fallar en este requisito se traduce en la reprobación inmediata de la experiencia de forma inapelable.

## 4.3. Criterios de Distinción

La distinción representa un trabajo adicional que sobresale a los requerimientos mínimos para la aprobación. Agregados adicionales no constituyen por sí mismo una distinción si no representan un verdadero trabajo adicional de comprensión y/o análisis.

Los trabajos distinguidos pueden caer (no exclusivamente) en algunas de las siguientes líneas generales:

- Portabilidad
- Utilización de características más avanzadas, por ejemplo, UART.
- Funcionalidades creativas, por ejemplo, puente H controlado por PWM.
- Funcionalidades sobresalientes en la línea de *Timers*, Interrupciones, ADC *multiple channel-multiple conversion mode*.

**Las Distinciones son discutidas caso a caso por la totalidad del equipo de ayudantes al finalizar la corrección del laboratorio. Serán notificadas públicamente después del módulo de evaluación.**