



Lab 5: Recuperativo experiencias prácticas

Nivel 3

1. Objetivo

Este laboratorio tiene como objetivo reforzar los conocimientos teóricos ya aprendidos y llevarlos a un plano práctico ahora que ya cuentan con sus materiales. Esta actividad se divide en 3 partes.

1. Revisión de microcontroladores (que complementará el LAB 01)
2. Programación ATmega328P, para familiarizarse con este entorno.
3. Programación MSP430F5529, para complementar lo faltante en los Laboratorio 02 y 03.

2. Evaluación

Este laboratorio, a diferencia de los anteriores, tendrá una nota calculada de 1.0 a 7.0. No existirán bonus, al cumplir con los requisitos solicitados podrá obtener la nota máxima. Para el cálculo de nota solo se considerará el Task 2 y Task 3. El detalle se encuentra a continuación:

2.1. Task 1.

- Al cumplir con la actividad mencionada en la subsección correspondiente, obtendrá todo el puntaje correspondiente al 40 % pendiente de ese laboratorio. No habrá puntos intermedios.
- Se duplican los bonus logrados con nota máxima 7.0. Esto quiere decir, si antes obtuvo 5 décimas de bono, con el cambio actual, tendrá 10 décimas de bono. Si obtuvo 10 décimas de bono, estas se mantendrán.

2.2. Lab 2 y 3 en conjunto con Task 2 y 3

En conjunto, estos crearán una nueva nota de laboratorio (LAB 05).

Por otra parte, los bonus obtenidos en los LAB 2 y 3 serán duplicados, con tope 10 décimas. Junto con esto, cada una de las nota de los LAB 2 y 3 (luego del cambio de bonus) serán promediada con la nota del LAB 05.



A modo de ejemplo, si en el LAB 05 se obtiene un 6.5, y en el LAB 02 tengo un 6.5 y en el LAB 03 tengo un 5.9 ($5.4 + 0.5$ décimas de bonus). Sucede lo siguiente:

- En el LAB 02, las 5 décimas de bonus se convierten en 10 décimas. Quedando en 7.0. Esta nueva nota se promedia con el 6.5 del LAB 05 quedando la nueva nota del LAB 02 en 6.75
- En el LAB 03, las 5 décimas de bonus se convierten en 10 décimas, quedando en 6.4. Esta nueva nota se promedia con el 6.5 del LAB 05, quedando la nueva nota del LAB 03 en 6.45.

3. Descripción de la actividad

3.1. *Task 1*: Revisión microncontroladores

Esta actividad consistirá en realizar un blink led y revisión de los pines de sus microcontroladores. El detalle de cómo realizar se encuentra en el enunciado del Laboratorio 1.

Este Task no tendrá revisión por parte de ayudantes. Solo deberá responder el siguiente Google Form informando que lo realizó. La nota se actualizará automáticamente al responder sí en el formulario. (Solo pueden acceder con su correo @uc.cl)

<https://forms.gle/c9ECpZW2jn6XnKXf7>

3.2. *Task 2*: AVR, Contador de tiempo de botón (40 %)

3.2.1. Objetivos

En experiencias pasadas, descubrió que es posible generar señales PWM mediante el uso de módulos *Timer* en sus microcontroladores. Ahora es momento de conocer en mayor profundidad estos elementos para programar rutinas con ejecuciones precisas, mediante la combinación de *Timers* e *Interrupts*.

En esta task realizará aplicaciones concretas de este módulo, más específicamente, el monitoreo del tiempo que un botón permanece presionado.

3.2.2. Task en detalle

Esta actividad consistirá en la implementación de un contador, el cual indique por cuánto tiempo se ha mantenido presionado el botón de usuario de su MCU. La cifra en cuestión



deberá ser presentada en el *display* de 7 segmentos, teniendo en cuenta las siguientes consideraciones:

- Por un lado, si el botón ha sido presionado por menos de 1 segundo, el formato de tiempo mostrado en el *display* deberá ser de tipo:



Por consideraciones del display, el punto lo ubicaremos al **costado derecho**.

- Por otro lado, si el botón ha sido presionado por más de 1 segundo, el formato de tiempo mostrado en el *display* deberá ser de tipo:



En esta parte, el punto se ubicará al costado derecho de la unidad, perdiendo resolución de milisegundos.

En otras palabras, **su contador deberá tener una precisión de 1ms**, y el tiempo que el botón es presionado debe estar en el rango de 0 a 9.99 segundos. En caso de que el tiempo que el botón es presionado por un tiempo fuera de ese rango, considere señalarlo mediante algún mensaje de error; para esto debe escribir **Err** en el *display*.

Cabe señalar que para el tiempo a desplegar en el display, puede mostrar cómo aumenta mientras el botón se mantiene presionado o mostrar el valor solamente al soltar el botón. Sin embargo, es importante que este número se mantenga visible para poder observarlo una vez suelto el botón. Para resetearlo, puede ser al volver a presionar el botón, o tener un botón de reset de este valor.



3.3. *Task 3* MSP430: Tu color favorito :) (60 %)

Para la siguiente actividad deberá hacer uso del potenciómetro, el LED RGB y display de 7 seg en su kit de SEP. La actividad consiste en la lectura y digitalización del voltaje medido en la línea central del potenciómetro al variar la resistencia del mismo, y su posterior conversión a un color de su elección en el led RGB.

Para concretar esta actividad, deberá conectar el potenciómetro tal como se muestra en la figura 1, para así poder medir el voltaje en la línea central, utilizando el ADC incorporado en el microcontrolador.

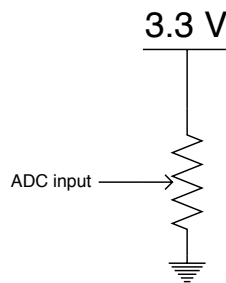


Figura 1: Conexión potenciómetro

Una vez obtenido dicho valor, debe convertirlo a una escala que permita modificar el ciclo de trabajo de una PWM, para poder controlar la intensidad de los canales Rojo, Verde y Azul del led RGB, permitiendo obtener una gama de colores distintos al mezclar visualmente dichos canales.

Para poder configurar cada canal por separado, solamente cuenta con un potenciómetro, los dos botones del MSP, el display de 7 segmentos y los dos leds de la placa. El proceso para cambiar a cada canal es el siguiente:

- Con el botón izquierdo debe seleccionar qué canal seleccionar. A modo de visualizar el canal seleccionado usará los leds de la placa tal como se indica en la siguiente tabla:

Canal	LED 1	LED 2
Rojo	ON	OFF
Verde	OFF	ON
Azul	ON	ON



- Para seleccionar el valor de duty cycle, girar la perilla del potenciómetro. Para que el usuario pueda conocer que valor está eligiendo, en el display de 7 seg deberá mostrarse en una escala de 0 a 255 el valor seleccionado, en donde 0 es apagado y 255 es el brillo máximo.

[**Importante**] Recuerde que el 0-255 es una escala que necesita de **8 bits** para mostrarse. Por otra parte el ADC del MSP430 es de **12 bits** y los timers del MSP430 son de **16 bits**. Debe cuidar qué tipo de variable está utilizando.

- Una vez seleccionado el valor a mostrar, debe presionar el **botón derecho** que funcionará al estilo de “OK”. Es importante notar que el cambio en el duty cycle debe ocurrir cuando se presiona este botón y no mientras se está moviendo la perilla del potenciómetro.

Por ejemplo, si es que desea que el LED RGB solo muestre el **color rojo** a intensidad máxima, y actualmente se encuentra configurando el **canal verde** y en un **color aleatorio**, lo que deberá hacer es lo siguiente:

1. Llevar el **canal verde** al valor mínimo girando la perilla y luego presionar **OK**.
2. Presionar el botón (cambiando así a la configuración del **canal azul**).
3. Llevar el **canal azul** al valor mínimo girando la perilla y luego presionar **OK**.
4. Presionar el botón (cambiando así a la configuración del **canal rojo**).
5. Llevar al máximo la perilla, luego presionar **OK**. Lo que mostrará únicamente el **color rojo**.

La configuración para otros colores intermedios se deberá realizar de la misma forma, cambiando canal por canal, girando la perilla del potenciómetro y presionando **OK**. (por ejemplo, el **morado** se forma al llevar al máximo el **canal azul** y el **canal rojo** y llevando al mínimo el **canal verde**). Es importante que la configuración pueda realizarse en todo momento, sin requerir resetear el microcontrolador. Debe mantenerse constante el valor de la PWM en cada canal y permitir que el usuario cambie el valor de las variables la cantidad de veces que desee.

Importante: La PWM se debe mantener en todo momento. Es por eso que al cambiar de canal, el canal anterior debe mantener el valor configurado previamente, lo que permitirá ver los distintos colores. Otro tipo de soluciones no se permitirán.

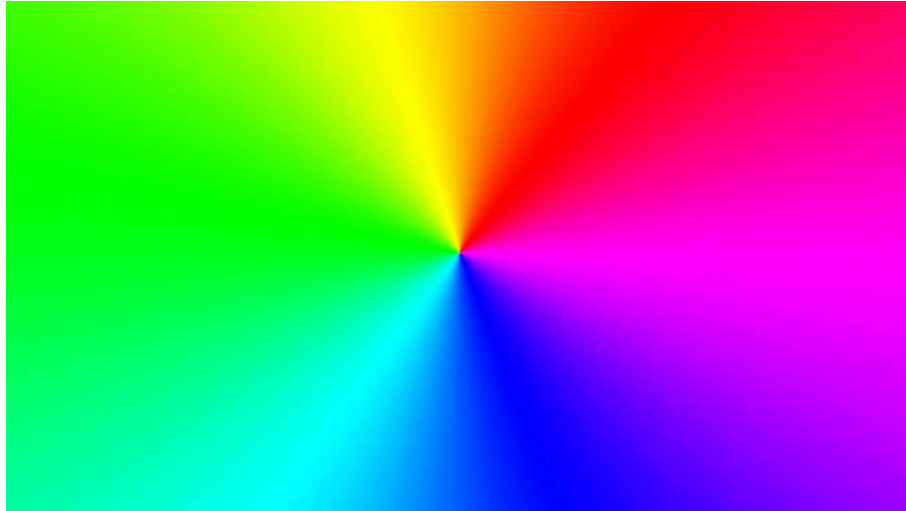


Figura 2: Escala continua para LED RGB

4. Lectura recomendada

- [ATmega328/P Complete Datasheet](#).
- [MSP430x5xx and MSP430x6xx Family User's Guide](#)
- [MSP430F552x, MSP430F551x Mixed-Signal Microcontrollers datasheet](#).
- Capitulo 28: Analog to Digital Converter del MSP .
- Capítulos 19 al 22: TC0, TC1, TC0/1 Prescalers y TC2 del [ATmega328/P Complete Datasheet](#).
- Capitulo 28: ADC12_A del [MSP430x5xx and MSP430x6xx Family User's Guide](#).
- Capitulo 17 y 18: Timer_A y Timer_B del [MSP430x5xx and MSP430x6xx Family User's Guide](#).
- [MSP430F552x, MSP430F551x Mixed-Signal Microcontrollers datasheet](#).
- Datasheet del [LED RGB](#).
- Ayudantías y códigos subidos



5. Rúbrica de Evaluación

5.1. Consideraciones generales

- Debe cumplir con el horario asignado. Tendrá un periodo de 30 min de revisión, separados en bloques.
 - Bloque 1 (10 min), deberá mostrar el funcionamiento del AVR o MSP. Aquí su ayudante estará conectado. Solo dispondrá de esta oportunidad para mostrar. Le sugerimos comenzar con el MSP.
 - Bloque 2 (10 min), en este periodo, su ayudante se desconectará de la sesión y estará revisando a otro alumno. Usted tendrá 10 min para cambiar las conexiones de un microcontrolador a otro.
 - Bloque 2 (10 min), deberá mostrar el funcionamiento del micro contrario al bloque 1. Aquí su ayudante se volverá a conectar. Solo dispondrá de esta oportunidad para mostrar.
- 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.
- No se reciben trabajos después del módulo de presentación. Trabajos no entregados son calificados con nota mínima.
- Solo dispone del tiempo asignado para ser revisado, atrasarse y no cumplir hará que no sea posible revisarlo de forma correcta.
- **IMPORTANTE:** se prohibirá el uso de funciones de Arduino IDE y Energia.nu para la programación de las tarjetas de desarrollo, esto por la simplicidad que involucra, por lo que deberán mostrar que entienden qué están realizando al momento de programar.

5.2. Criterios de aprobación

5.2.1. Task 1: Revisión microcontroladores

Solo debe responder el formulario indicado, nada más.

<https://forms.gle/c9ECpZW2jn6XnKXf7>



5.2.2. Task 2: ATmega328P

1. Funcionamiento de los requerimientos. El alumno realiza una presentación de su trabajo y se responsabiliza de exponer que su trabajo satisfaga todos los requerimientos mínimos solicitados en la *descripción de la actividad*, los cuales incluyen en este laboratorio:
 - Debe detectarse el botón mediante interrupciones de pin.
 - El tiempo comienza cuando se presiona el botón y termina cuando se suelta el botón.
 - El contador de tiempo debe mostrarse en el display de 7 segmentos utilizando la estructura mencionada.
 - Debe usar interrupciones de Timer para contar el tiempo y para mostrar los dígitos del display de 7 segmentos.
 - Actualiza el display de 7 segmentos a una frecuencia suficiente para dar la sensación de que se encuentran los 3 números activos al mismo tiempo (**HINT: una frecuencia de aproximadamente 120 Hz debería ser suficiente.**)
 - Mantiene el último tiempo registrado de forma visible al soltarse el botón.
2. Preguntas. Se responde satisfactoriamente preguntas aleatorias al momento de la presentación final, las cuales abarcan los siguientes temas:
 - Funcionamiento del sistema de interrupciones de Timers del microcontrolador especificado. Teoría de operación, registros utilizados, etc
 - Beneficios de utilizar Timers e interrupciones de Timers en el contexto de sistemas embebidos. ¿Por qué son importantes ?
 - Qué representa cada línea de código y en qué se traducen en el funcionamiento del programa.

5.2.3. Task 3: MSP430

1. Funcionamiento de los requerimientos. El alumno realiza una presentación de su trabajo y se responsabiliza de exponer que su trabajo satisfaga todos los requerimientos mínimos solicitados en la *descripción de la actividad*, los cuales incluyen en este laboratorio:
 - Programa compilado y ejecutándose



- Utilización de canal rojo, verde y azul del LED RGB.
- Variación visible de color al girar el potenciómetro y luego seleccionar el valor.
- Utilización de ADC para medir el voltaje en el potenciómetro.
- Mantiene fijo el valor de los dos canales que no están siendo utilizados.
- Utilización de *Timers* dedicados a la generación de PWM (No se permite usar *dekays* para simular una PWM).
- Deberá justificar la configuración de ADC, ya sea con o sin interrupciones.
- No es necesario utilizar interrupciones de pin para los botones, pero sugerimos realizarlo. Hará más fácil la programación.
- **IMPORTANTE:** Los pines del MSP430 no tienen debouncing por hardware como si los tiene el AVR, por lo que deberá tener cuidado a la hora de programar, ya que se podrían producir glitches que afecten el comportamiento esperado. Para solucionar esto, les recomendamos revisar el enunciado del LAB 02 Task 2 “Debouncing” y la Guía AVR del profesor Marcelo Guarini, que se encuentra en CANVAS.

2. Preguntas. Se responde satisfactoriamente 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 y únicamente será posible si es que cumple con el mínimo informado en la rúbrica. Fallar en este requisito se traduce en que no se considerarán los puntos obtenidos en la parte **Funcionamiento de requerimientos**.

5.3. Bonus

Al cumplir con todos los puntos indicados en la **Rúbrica de Evaluación** obtendrá la nota máxima equivalente a 7.0. En este laboratorio no hay bonus.