



Lab 9: I^2C , Inter-Integrated Circuit

1. Objetivos

Este laboratorio busca introducirlo al uso del protocolo de comunicación conocido como I^2C (Inter-Integrated Circuit) ¹ entre distintos dispositivos.

Para esto se empleará el uso de dos componentes periféricos: un sensor de temperatura TMP102 y un conversor de señal digital a analógica MCP4725. En el desarrollo del laboratorio deberá ajustar los registros de su microcontrolador para poder iniciar la comunicación con TMP102. Además, deberá escribir un programa que genere una onda sinusoidal desde el DAC para conectarla a un LED y que este muestre un brillo oscilante.

2. Descripción actividad

2.1. *Task* 1: Comunicación con el sensor TMP102

Para esta actividad deberá utilizar el sensor TMP102. Se le han proporcionado distintos archivos que deberá modificar para poder llevar a cabo la comunicación con este dispositivo.

Como podrá observar, se tiene el archivo `i2c_temperature.c` el que contiene la estructura lista para poder realizar la comunicación con el sensor TMP y poder enviar los datos mediante UART.

- Investigue acerca del protocolo I^2C ,
 - ¿Cómo se realiza la comunicación? ¿Qué son los *masters* y *slaves* ?
 - ¿Qué valores se necesitan para poder entablar la conexión con un dispositivo ?
 - ¿Qué son las condiciones de **Start**, **Repeated Start** y **Stop** ?
 - ¿Cómo se puede leer o escribir desde un dispositivo ? ¿Cómo se le dice que tipo de información quieres leer ?
 - Investigue acerca de las distintas velocidades que los dispositivos soportan. En general es de 100 kHz a 400kHz, por lo que utilizar el más rápido es lo deseable. Actualmente hay pocos componentes que soportan modos más lentos de comunicación.
 - Lea el datasheet del TMP102 para poder comprender su funcionamiento.

¹También conocido por Two Wire Interface en algunos dispositivos



- Complete los archivos dentro de la carpeta USART con las funciones ya utilizadas en laboratorios anteriores.
- Posterior a lo anterior, deberá completar la funciones de los archivos que se encuentran dentro de la carpeta I2C.
- De realizar los pasos anteriores de forma correcta, deberá ser capaz de poder medir la temperatura.

Advertencias

El sensor de temperatura TMP102 funciona con un voltaje máximo de 3.3V. NO intente alimentarlo con valores más altos. Es necesario conectar el pin VCC de este a una alimentación de 3.3V. Las líneas SDA y SCL, están llevadas al riel de VCC con las resistencias de pull-up de los dispositivos en la línea, por lo que su valor no debería alcanzar voltajes más altos. Sin embargo, en caso de configurar estos pines como outputs en su microcontrolador podría accidentalmente suministrar 5.0V a esta línea. Es por esto que **SDA y SCL deben ser configurados como inputs mientras se este conectado a TMP102.**

2.2. *Task 2:* Establecer comunicación con el MCP4725

En esta actividad deberá establecer una comunicación con el DAC MCP4725 para poder generar una senoide en un LED, la que podrá ser revisada con osciloscopio.

- Para esto necesitará toda la preparación ya realizada en el *Task 1*.
- Revise el datasheet del DAC entregado y observe el código entregado que se encuentra en el archivo `i2c_DAC.c`.
- Este es un DAC de 12 bits, piense en cómo enviar los datos necesarios para poder entregar la información necesaria y lograr que funcione.
- Encuentre una forma básica de establecer comunicación con este dispositivo. Es útil realizar un diagrama de flujo del programa para poder programarlo de forma adecuada.
- Conecte un LED o un osciloscopio para ver la salida que se produce con este DAC.

El MCP4725 si soporta voltajes de 5.0V, sin embargo como debe conectar ambos dispositivos a las líneas de comunicación, **MCP4725 debe ser alimentado con 3.3V también.**



2.3. *Task* final a revisar

Debe lograr que ambas *Task* funcionen de forma concurrente en un mismo código, preferentemente mediante un menú UART para seleccionar la opción a desplegar. No está permitido el uso de códigos separados para cada *Task*.

3. Información Adicional

- [ATmega328/P Complete Datasheet](#).
- [MSP430x5xx and MSP430x6xx Family User's Guide](#).
- [MSP430F552x, MSP430F551x Mixed-Signal Microcontrollers datasheet](#).
- [TMP102](#) es la página del sensor de temperatura TMP102. Contiene su datasheet y más.
- [MCP4725](#) es la página del MCP4725 DAC, contiene su datasheet y más.
- [Application Note de AVR acerca de I2C](#)
- [Historia de I²C](#) Breve historia de la creación de I²C.



4. Pauta de Evaluación

4.1. Consideraciones generales

- El laboratorio será evaluado exclusivamente con nota 1.0 (**R**eprobado), 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. La hora límite para inscribir a revisión es a las 10:20 hrs, posterior a esto se asignará una posición aleatoria.
- 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

1. Microcontrolador utilizado. La elección del μC es libre.
2. 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 el enunciado del laboratorio, los cuales incluyen:
 - **Tarea 1.** Se completan las librerías especificadas o se diseña una propia que cumpla especificaciones².
 - **Tarea 2.** Funcionamiento y operación adecuados del MCP4725. Frecuencia de forma **sinusoidal**, comprobable con osciloscopio.
 - **Concurrencia.** Ambas tareas deben ejecutarse de forma concurrente en un mismo flashing de la tarjeta de desarrollo. Puede resultar ideal que diseñe un menú en USART.
3. Preguntas: Se responde satisfactoriamente preguntas aleatorias al momento de la presentación final, las cuales abarcan los siguientes temas:
 - Qué representa cada línea de código mostrada en el programa.
 - Funcionamiento detallado del protocolo I^2C .

²Una **librería** propia, no un **código** propio.



- Operación del TMP102 y MCP4725. Comprensión mínima de los comandos necesarios para poder realizar las tareas.

4. Formalidades: Para la presentación debe cumplirse con las siguientes formalidades:

- a) Presentación **breve** del funcionamiento y el código implementado.
- b) Uso de buenas prácticas. No asigne directamente los registros, utilice los *flags* respectivos. Defina macros, o use las ya existentes como `_BV()`.
- c) Código debidamente comentado y ordenado, en la línea de las especificaciones dadas en laboratorios anteriores. Separe sus métodos de inicialización o envío de caracteres de `int main()`.

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 si mismo una distinción si no representan un verdadero trabajo adicional de comprensión y/o análisis.

Asimismo, el trabajo primero debe cumplir los criterios de aprobación para ser un candidato a distinción. Es decir, trabajos que hayan dejado de funcionar por culpa de la extensión agregada al código con fin de distinguir son reprobados.

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

- Implementar la tarea en cada μC .
- Presentación creativa.
- Creación de sonidos utilizando el DAC.
- Utilización de la EEPROM para guardar los valores de la sinusoide con la que se creará la sinusoide del DAC.
- Desplegar un gráfico de temperatura utilizando la pantalla TFT.

Si usted utilizó una característica en un laboratorio anterior para lograr distinción, esta no vuelve a considerarse para una distinción en este laboratorio.