



Lab 05, Protocolos de comunicación

Laboratorio Nivel 4

1. Introducción

Este laboratorio busca introducir el uso de nuevos protocolos de comunicación y extender las funcionalidades del ya implementado. En esta ocasión, se estudiará el protocolo conocido como I^2C (Inter-Integrated Circuit) ¹, el que permitirá conectar a nuevos distintos dispositivos. Para esto, se empleará el uso de dos componentes periféricos, que se encuentran incorporados en el Booster Pack: un sensor de temperatura infrarrojo TMP006 y un sensor de luz ambiente OPT3001.

Junto con lo anterior, se introducirá del protocolo *Serial Peripheral Interface*, más conocido como SPI. Para esto se hará uso de un *display* TFT, que posee un controlador integrado y *framebuffer* ST7735, el cual hace que el módulo TFT sea fácil de manejar por medio del microcontrolador.

El objetivo de este laboratorio es lograr la implementación de un pequeño sistema de adquisición de datos, en que un dispositivo es el encargado de medir y procesar, para poder ser visualizados.

2. Actividad

La actividad a desarrollar, consiste en la implementación de un sistema de adquisición de datos. Para esto, se obtendrán datos utilizando un sensor de temperatura TMP006 y un sensor de luz ambiente OPT3001 a intervalos de tiempo regulables y configurables por un terminal seria, junto el giroscopio incluido en el Booster Pack.

Para la visualización de los datos y control del sistema, se deberá utilizar la pantalla LCD del Booster Pack, en la que se tendrán distintas pantallas de visualización que serán explicadas en las siguientes secciones.

¹También conocido por Two Wire Interface en algunos dispositivos



2.1. Menú Principal y Menús secundarios

Al momento de iniciarse el sistema, deberá mostrar un menú principal que debe poseer las siguientes opciones:

1. Configurar hora inicial.
2. Configurar tiempos de adquisición sensores I2C
3. Visualizar datos adquiridos sensores I2C
4. Visualizar orientación.
5. Resetear el sistema.

Para poder seleccionar cada una de las opciones, deberá contar con un botón que actúe de “OK” y alguna forma de desplazarse entre las opciones, lo que puede ser realizado con el ADC o con alguno de los botones. Deberá explicar su elección en el README de su laboratorio.

En la pantalla principal, se debe mostrar la hora en alguna de las esquinas de la pantalla. Cada uno de los siguientes submenús, se presenta a continuación:

2.2. Configurar hora inicial

En este menú, debe ser posible configurar una hora inicial, para que el sistema se sincronice con el mundo real. Las condiciones son las siguientes:

- La configuración de la hora debe realizarse enviando un mensaje vía UART a través del terminal serial en el formato HH:MM:SS. Una vez recibida esta instrucción, se debe dar inicio a que la hora continúe sin problemas.
- Para avanzar en la hora, se debe utilizar el modulo RTC del microcontrolador. Puede utilizar el código entregado en clases para basarse.
- Debe ser posible volver al menú principal de alguna forma.



2.3. Configuración tiempos de adquisición sensores I2C

En este menú, se debe mostrar y cambiar la configuración actual de los sensores I2C, mediante un mensaje estructurado enviado por UART. Las condiciones que se debe cumplir en este menú son las siguientes:

- Visualizar la configuración actual de los sensores que se encuentra guardada en memoria. Por defecto se comienza en una medición cada 5 seg para cada sensor.
- Debe ser posible cambiar la configuración de cada sensor por un mensaje UART.
- Será posible definir los tiempos de medición en segundos de cada uno de los sensores, estos estarán en el rango de 1 a 99 seg y se enviarán de la siguiente forma:

SET OPT XX
SET TMP XX

Siendo OPT para el sensor de luz y TMP para el sensor de temperatura. XX indica el valor en segundos, si es menor a 10, agregará un cero a la izq. Esta instrucción debe ser robusta, por lo que si en vez de escribir un dígito **se escribe letras**, el microcontrolador debe reconocer que la instrucción está incorrecta y enviar un NK. En caso positivo, debe informar con un OK por consola o en pantalla.

- Una vez configurada la adquisición de datos, estos deberán actuar sin afectar el funcionamiento del sistema, por lo que el uso de un timer está fuertemente recomendado.
- Debe ser posible volver al menú principal de alguna forma.

2.4. Visualizar datos adquiridos sensores I²C

En esta pantalla, se deberán visualizar los datos adquiridos tanto por el sensor de temperatura como por el sensor lumínico. Además, se debe de llevar un registro de la cantidad de datos adquiridos por cada uno de ellos y mostrados junto con la última medición. Se debe cumplir con los siguientes puntos:

- Se debe mostrar cuál fue el último dato adquirido, junto con la cantidad de datos actuales adquiridos.
- El formato y colores para mostrarlo queda a su libre elección.



- El número máximo de la cantidad de datos a guardar, debe ser de al menos 999, pasado cierto umbral (definido por ustedes) el valor debe volver a 0.
- El número de datos adquiridos por cada sensor, debe perdurar en el tiempo, incluso en aquellos momentos en que el microcontrolador se resetee, por lo que este valor deberá ser guardado en un bloque de **memoria FLASH**.
- Debe ser posible volver al menú principal de alguna forma.

A continuación se detalla el uso de cada sensor:

2.4.1. TMP006

En el caso del primer sensor, cuyo datasheet se puede encontrar en el siguiente [link](#), se trata de un sensor de temperatura que se comunica por I2C, es de 14 bits de precisión con una resolución de 0.0315 °C por bit.

Al realizar una medición se deberá guardar el valor de temperatura en grados Celsius utilizando solo 2 decimales y se aumentará el contador de datos.

En el datasheet se encontrarán con que el sensor también mide temperaturas negativas, sin embargo estas no serán utilizadas por lo que se ignorará esta capacidad, se puede asumir que la medición siempre será positiva.

2.4.2. OPT3001

El segundo sensor, se trata de un sensor de luz ambiente, cuyo datasheet se puede encontrar en el siguiente [link](#). Este entrega el resultado de cantidad de luz ambiente en lux, tiene una respuesta muy similar al del ojo humano.

Al realizar una medición se deberá guardar el valor de luz ambiente medido en lux. Por el rango de valores que se medirá, en ningún momento se superarán los 99999 lux y siempre será un valor superior a 1. Por lo que puede asumir que no ocurrirán decimales ni valores superiores al primero.

2.5. Visualizador de orientación

Utilizando el giroscopio integrado en el Booster Pack, deberá mostrar la información de la orientación que está teniendo actualmente el microcontrolador. El intervalo de tiempo debe ser suficientemente rápido para poder verlos en “tiempo real”. Queda a su elección la forma de mostrarlo.

Al igual que en los puntos anteriores, debe ser posible volver al menú principal.



2.6. Resetear el sistema

En esta pantalla, debe ser posible resetear los contadores actuales y los guardados en memoria flash. Además, se debe volver a la configuración por defecto de los sensores.

2.7. Restricciones de implementación

Al ser un sistema que tendrá muchas acciones ocurriendo a intervalos regulables y algunos a intervalos no regulables, deberá ser optimizado el proceso y robusto ante situaciones no pensadas. Es por esto que se debe cumplir con lo siguiente:

- Se sugiera que **recepción** como **transmisión** de UART, sean implementados con **interrupciones**.
- Los timers automatizarán las tareas de adquisición de datos, deben ser realizados por interrupciones.
- Se debe tener el código separado en librerías, al menos se requiere de una carpeta que contenga los archivos relacionados con UART y otra con I2C. En caso de no cumplir esto, se le descontará puntaje. (2 décimas).
- Utilizar I²C a una velocidad de **400 kHz**.



3. Lectura recomendada y necesarias.

- Páginas iniciales del Datasheet del MSP430F5529
- Datasheet OPT300 <https://www.ti.com/lit/ds/symlink/opt3001.pdf>
- Datasheet TMP006 [https://media.digikey.com/pdf/Data%20Sheets/Texas%20Instruments%20PDFs/TMP006\(B\).pdf](https://media.digikey.com/pdf/Data%20Sheets/Texas%20Instruments%20PDFs/TMP006(B).pdf)
- Capitulo 38: I2C Mode del [MSP430x5xx and MSP430x6xx Family User's Guide](#).
- Capitulo 36: Universal Serial Communication Interface – UART Mode del [MSP430x5xx and MSP430x6xx Family User's Guide](#).
- Buen tutorial para el uso de I2C en MSP430. <http://www.simplyembedded.org/tutorials/msp430-i2c-basics/>
- [MSP430x5xx and MSP430x6xx Family User's Guide](#).
- [MSP430F552x, MSP430F551x Mixed-Signal Microcontrollers datasheet](#).
- BOOSTXL-EDUMKI User Guide <https://www.ti.com/lit/ug/slau599a/slau599a.pdf>
- Cristal LCD. <https://www.crystallfontz.com/product/cfaf128128b0145t-graphic-tft-128x128>



4. Control de salida

4.1. Consideraciones generales

- Se habilitará un control en el modulo de entrega del laboratorio.
- 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.
- Puede utilizar cualquier recurso que tenga disponible. Ya sea un block de notas, alguna parte de su código, tener una presentación de apoyo, el datasheet, etc.
- **IMPORTANTE:** se prohibirá el uso de funciones de 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.

4.2. Preguntas

Las preguntas que se evaluarán en este control abarcan los siguientes temas:

- Especificaciones generales del MSP430F5529 en lo que respecta al uso del modulo de I2C.
- Especificaciones generales de cada uno de los perifericos utilizados.
- Funcionamiento del sistema del protocolo SPI en el MSP, tanto a nivel general como a nivel específico.
- Especificaciones generales del MSP430F5529 en lo que respecta a *timers*. Tales como frecuencias admitidas, elección de modos, cantidad de contadores, cantidad de *timers*, etc.
- Funcionamiento del sistema del protocolo UART en el MSP, tanto a nivel general como a nivel específico.
- Comprender **cada línea** de su programa.
- Entendimiento de *Headers Files* o archivos de extensión .h. ¿Cuál es el beneficio que tiene la creación de bibliotecas y cómo ayuda a la estructuración de código ?
- ¿Qué significan las directivas `#IFDEF` `#DEFINE` `#ENDIF` ?



5. Fecha y forma de de entrega

Este laboratorio se puede realizar en parejas o de forma individual. Para crear un repositorio compartido, debe agregar .añadir a ambos estudiantes.^al momento de ingresar al link para el repositorio. La inscripción se considerará solo de esta forma.

La entrega se debe realizar el día miércoles 01 de diciembre a las 10.00 hrs en el laboratorio asignado. El día lunes 29 de noviembre, en horario de laboratorio se realizará el control de salida.

La evaluación será 75 % código, lo que será evaluado de forma grupal. 25 % individual, que corresponde al control de salida. Cada estudiante deberá comprender el laboratorio completo, por lo que debe ser capaz de responder preguntas de todos los contenidos abordados en este laboratorio.

Deberá subir su **código completo y ordenado** a su repositorio de GitHub junto con un archivo Readme.md que indique las fuentes utilizadas y una lista de lo que no fue implementado en su código. Para esto, el tiempo máximo es las 10.00 hrs del día de la revisión.

El código entregado debe cumplir con lo siguiente, el no cumplimiento de estos puntos podría incurrir en un descuento:

- Programa compilable y sin errores. Debe seguir el orden utilizado en los laboratorios anteriores y solo subir el código utilizado.
- Código ordenado y debidamente comentado. No se exigirá ningún estándar en particular, pero se debe mantener una estructura que permita una lectura fácil. ²³
- Recursos utilizados se deben citar en el README.md de su repositorio.
- El README.md deberá mostrar de forma muy clara qué fue implementado y que no. Además puede agregar alguna breve explicación para alguna parte de su código. En caso de no realizarlo, se podrá caer en un descuento de 3 décimas.
- Su código debe tener un encabezado como el mostrado en el ejemplo de “Hello World!” del LAB 01.

²En <https://developer.gnome.org/programming-guidelines/stable/c-coding-style.html.en> pueden encontrar ejemplos de cómo mantener un buen formato de código.

³En <https://codebeautify.org/c-formatter-beautifier> pueden reformatear un código.



6. Condiciones de revisión de laboratorio

El día de la revisión, tendrá 15 min para realizar su presentación. Contando 10 min para mostrar el funcionamiento y 5 min de preguntas por parte de los ayudantes.

Para la parte de la presentación, deberá mostrar lo siguiente:

- Tener en una presentación o documento que indique lo realizado y que se mostrará. Para esto, puede usar la rúbrica que se publicará al menos 1 semana antes de la revisión.
- Programa compilado al momento de presentar. Esto con el fin de optimizar el tiempo de revisión.

Esta revisión, al ser presencial, no existirá la opción de solicitar recorección.



7. ANEXO: Recomendaciones para la realización del laboratorio

Lo siguiente, tiene como objetivo ser una ayuda para la realización del laboratorio. Más que ser una guía estricta, será una serie de recomendaciones para guiar el trabajo.

7.1. Utilizar GitHub en común

Si realizará el laboratorio en grupo, sugerimos utilizar el repositorio asignado, para que cada uno de los integrantes pueda ir aportando con desarrollos. Pueden utilizar sus Issues privadas para llevar registro de lo realizado y utilizar branch que permitan llevar desarrollos en paralelo, sin afectar la rama principal. Pueden encontrar información [aquí](#)

7.2. Pruebas con periféricos

Antes de comenzar a escribir el código del laboratorio completo, es necesario establecer una comunicación con los sensores TMP006 y OPT3001. Para esto, adjunto a este enunciado, se le entregaron 2 carpetas que contienen una estructura sugerida para la comunicación con estos periféricos. Se sugiere completar los siguientes puntos:

- Estudiar el protocolo I²C sin enfoque en el MSP. Esto es para familiarizarse con los distintos modos de operación y permitirán definir si el MSP deberá trabajar como *master* o *slave*.
- Identificar la dirección que utilizan, esta información se encuentra tanto en el datasheet del Booster Pack como en los datasheets respectivos de cada periférico.
- Definir el modo a utilizar y las velocidades disponibles. Dada las características de los sensores, se sugiere utilizar el modo a 400 kHz.
- Completar la librería I2C, en ella encontrarán una pequeña explicación de qué debe realizar cada función.
- Completar la configuración y modo de lectura de la información de cada uno de los sensores. La información necesaria para cada uno de los sensores, se encuentra en sus respectivos datasheets.
 - El sensor de temperatura, además de entregar la temperatura, es posible consultarle el voltaje medido. Identifique cuál es la instrucción necesaria para que el valor leído sea de temperatura.



- El sensor de luz, requiere de una configuración previa. La forma de realizarla está explicada en el datasheet y el valor a escribir se encuentra en el código de ejemplo.
- Una vez realizado este proceso, pruebe en códigos separados cada uno de los sensores. La librería de I2C realizada, tiene que ser compatible para cada uno de ellos. Luego, intente crear un código que envíe de forma alternada cada uno de los valores.