**Introducción**

En el desarrollo de este informe correspondiente al procedimiento del proyecto asignado ‘SD card data logger’ se detalla el proceso de programación junto al proceso de montaje de la placa STM32 para lograr la comunicación con la tarjeta SD.

El procedimiento como tal logra la comprensión de nuevos conceptos acerca de comunicación digital, en específico el protocolo SPI.

***Objetivo:***

El principal objetivo del proyecto es lograr una comunicación directa entre la placa STM32f407 con una tarjeta SD, para así poder almacenar datos dentro de ella, cada una de estas con su correspondiente programación.

***Marco teórico:***

La etapa de investigación permite conocer el protocolo de comunicación digital que se requiere para llevar a cabo la etapa de programación de la tarjeta.

El software utilizado para dicha etapa es Atollic TRUEStudio for STM32, parte importante de STMicroelectronics.

La placa utilizada en este trabajo es la versátil STM32F407VG, cuya variedad de aplicaciones y cualidades permiten lograr múltiples e interesantes proyectos a partir de ella. La conexión al software del computador es mediante un cable USD, en cambio, la conexión requerida para comunicar la tarjeta SD a la placa es mediante los pines especificados en el código y en el data sheet de la tarjeta SD genérica. Esto fue logrado mediante unos cables hembra-macho, soldando la parte macho a los pines de la tarjeta ubicados en la parte posterior de ella.

La comunicación SPI, es un protocolo síncrono principalmente utilizada para comunicar circuitos integrados a dispositivos electrónicos, en este caso un Notebook, es un bus de interfaz de periféricos serie compatible para comunicar la tarjeta SD con la placa, para así poder programar mediante Atollic. SPI controla prácticamente cualquier dispositivo electrónico digital que acepte un flujo de bit serie regulado por un reloj.

Dicho bus de comunicación utiliza el modelo Maestro-Esclavo para llevar a cabo el log de datos. Este modelo tiene un maestro (en este caso la STM32) y un esclavo (tarjeta SD). El esclavo debe recibir datos y almacenarlos. SPI utiliza tres cables principales para la transmisión de datos, y un cable o pin denominado SS, el cual puede conectar el maestro a distintos esclavos.

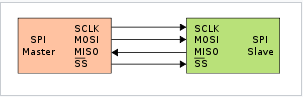


Figura – Comunicación SPI Maestro Esclavo

Posee un pin SCLK que funciona como reloj serie para la sincronización esclavo-maestro, un pin o cable MOSI (Master Out Slave In) que envía datos desde el maestro al esclavo, un pin MISO (Master In Slave Out) que envía datos desde el esclavo al maestro y un pin SS que selecciona al esclavo utilizado (Slave Select). Además, una ventaja de la comunicación SPI es que es full-duplex, es decir, puede enviar datos en ambas direcciones simultáneamente.

La idea y el fin de este proyecto es que una tarjeta SD común y corriente logre registrar secuencialmente datos en un archivo o base de datos a partir de la programación de una STM32. El proceso de detalla a continuación.

***Procedimiento***

Lo primero es comenzar la programación de la comunicación SPI mediante Atollic. Para esto se necesitan crear las librerías requeridas para SPI. Dichas librerías están todas creadas y se pueden revisar en el archivo que se adjunta a este informe. Se define el sistema de archivos FAT, utilizando la librería FATFS, módulo para pequeños sistemas embebidos, pudiendo ser incorporado a pequeños microcontroladores.

Luego de terminar el código de programación, es requerido conectar la placa STM32 con el computador vía USB, así el microcontrolador podrá comprender el mensaje enviado por Atollic. La tarjeta SD siempre admitirá el funcionamiento SPI.

Para conectar la tarjeta a la placa, se utilizan cable hembra-macho. La hembra va a los pines de la placa correspondientes a cada pin de SPI, y el macho va a la SD siendo soldada para lograr una correcta comunicación. En nuestro caso utilizamos dos adaptadores microSD a SD, uno fue soldado a la placa para utilizarlo como interfaz, ya que podemos sacar fácilmente la microSD y llevarla al otro adaptador para conectarlo directamente al computador, verificando si el almacenamiento de datos fue exitoso.

Como se aprecia en el código, cuando se inicie el proceso, se prenderá una led rojo, cuando se cree el archivo almacenador, se apagará el led rojo y se encenderá un led verde, por último cuando el archivo esté listo para escribir y leer se prenderán ambos leds.

El procedimiento y la etapa práctica fueron exitosos, llevándose a cabo al pie de la letra y funcionando correctamente.

***Bibliografía:***

<http://www.alldatasheet.com/view.jsp?Searchword=Stm32f4&gclid=CjwKCAjw1KLkBRBZEiwARzyE77Ut5yUACp4SUt4OkPGPDPLHC6VkVAu0F6csuG3ZmBxoxNN2UM2UxhoCgccQAvD_BwE>

<https://www.st.com/en/microcontrollers-microprocessors/stm32f4-series.html?querycriteria=productId=SS1577>

* Digital design and computer architecture / David Money Harris & Sarah L.Harris
* Apuntes Arquitectura de computadores 2018-2

***Conclusión:***

Una buena forma de concluir este informe es destacando todos los puntos a favor y ventajas que se pudieron aprovechar durante el proyecto. El procedimiento del programa permite comprender un nuevo mundo como es la programación, conociendo nuevos conceptos y formas de manipular un código para algún dispositivo en específico. La previa investigación de la placa, su protocolo de comunicación y el método de conexión trae consigo nuevos conocimientos sobre la transmisión de datos mediante buses de comunicación.

Nombres: **Luis San Martín**

**Nicolas Loyola Rivas**

Fecha: 12/03/2019

Profesor: **Krzysztof Herman.**

Ayudante**: Matías Andrés Max P.**