

Docentes: Ing. Jorge E. Morales
Téc. Sup. Mecatrónica Gonzalo Vera.

GRUPO NRO 8:

- Schafrik Maria Victoria
- Vera Emilio Andres
- Rojas Jorge Daniel
- Rojo Pedro Omar
- Narvaez Juan Carlos

Practico 4

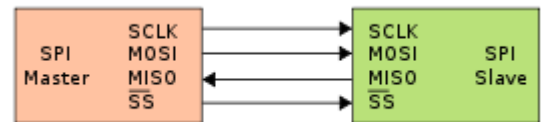
Shields v1

Ejercicio #2

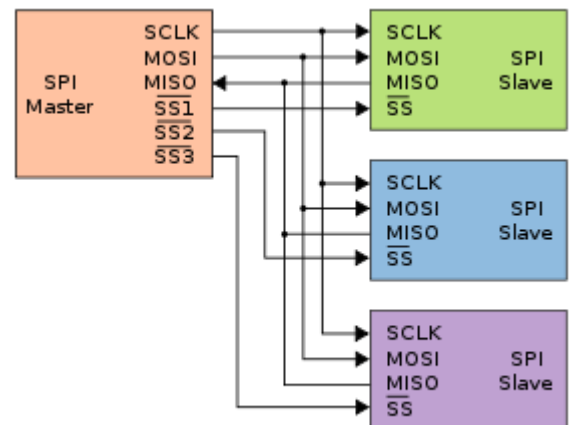
a) Que es el protocolo SPI y cuales son sus características.

El **Bus SPI** (del inglés *Serial Peripheral Interface*) es un estándar de comunicaciones, usado principalmente para la transferencia de información entre circuitos integrados en equipos electrónicos. El bus de interfaz de periféricos serie o bus SPI es un estándar para controlar casi cualquier dispositivo electrónico digital que acepte un flujo de bits serie regulado por un reloj (comunicación sincrónica).

Incluye una línea de reloj, dato entrante, dato saliente y un pin de *chip select*, que conecta o desconecta la operación del dispositivo con el que uno desea comunicarse. De esta forma, este estándar permite multiplexar las líneas de reloj.



Muchos sistemas digitales necesitan una conexión rápida con sus periféricos. La ventajas de un bus serie es que minimiza el número de conductores, pines y el tamaño del circuito integrado. Esto reduce el coste de fabricar, montar y probar la electrónica. Un bus de periféricos serie es la opción más flexible cuando se tiene tipos diferentes de periféricos serie. El hardware consiste en señales de reloj, data in, data out y chip select para cada circuito integrado que tiene que ser controlado. Casi cualquier dispositivo digital puede ser controlado con esta combinación de señales. Los dispositivos se diferencian en un número predecible de formas. Unos leen el dato cuando el reloj sube, otros cuando el reloj baja. Algunos lo leen en el flanco de subida del reloj y otros en el flanco de bajada. Escribir es casi siempre en la dirección opuesta de la dirección de movimiento del reloj. Algunos dispositivos tienen dos relojes. Uno para capturar o mostrar los datos y el otro para el dispositivo interno



Características

El SPI es un protocolo síncrono. La sincronización y la transmisión de datos se realizan por medio de 4 señales:

- **SCLK** (*Clock*): Es el pulso que marca la sincronización. Con cada pulso de este reloj, se lee o se envía un bit. También llamado TAKT (en alemán).
- **MOSI** (*Master Output Slave Input*): Salida de datos del Master y entrada de datos al Esclavo. También llamada SIMO.

- **MISO** (*Master Input Slave Output*): Salida de datos del Esclavo y entrada al Master. También conocida por SOMI.
- **SS/Select**: Para seleccionar un Esclavo, o para que el Master le diga al Esclavo que se active. También llamada SSTE.

La Cadena de bits es enviada de manera síncrona con los pulsos del reloj, es decir con cada pulso, el Master envía un bit. Para que empiece la transmisión el Master baja la señal SSTE o SS/Select a cero, con esto el Esclavo se activa y empieza la transmisión, con un pulso de reloj al mismo tiempo que el primer bit es leído. Nótese que los pulsos de reloj pueden estar programados de manera que la transmisión del bit se realice en 4 modos diferentes, a esto se llama polaridad y fase de la transmisión:

1. Con el flanco de subida sin retraso.
2. Con el flanco de subida con retraso.
3. Con el flanco de bajada sin retraso.
4. Con el flanco de bajada con retraso.

b) Cuales son las ventajas y desventajas del protocolo SPI

Ventajas

- Comunicación Full Duplex
- Mayor velocidad de transmisión que con I²C o SMBus
- Protocolo flexible en que se puede tener un control absoluto sobre los bits transmitidos
 - No está limitado a la transferencia de bloques de 8 bits
 - Elección del tamaño de la trama de bits, de su significado y propósito
- Su implementación en hardware es extremadamente simple
 - Consume menos energía que I²C o que SMBus debido que posee menos circuitos (incluyendo las resistencias *pull-up*) y estos son más simples
 - No es necesario arbitraje o mecanismo de respuesta ante fallos
 - Los dispositivos *clientes* usan el reloj que envía el *servidor*, no necesitan por tanto su propio reloj
 - No es obligatorio implementar un transceptor (emisor y receptor), un dispositivo conectado puede configurarse para que solo envíe, sólo reciba o ambas cosas a la vez
- Usa mucho menos terminales en cada chip/conector que una interfaz paralelo equivalente
- Como mucho una única señal específica para cada *cliente* (señal SS), las demás señales pueden ser compartidas

Desventajas

- Consume más pines de cada chip que I²C, incluso en la variante de 3 hilos
- El direccionamiento se hace mediante líneas específicas (señalización fuera de banda) a diferencia de lo que ocurre en I²C que se selecciona cada chip mediante una dirección de 7 bits que se envía por las mismas líneas del bus
- No hay control de flujo por hardware
- No hay señal de asentimiento. El *servidor* podría estar enviando información sin que estuviese conectado ningún *cliente* y no se daría cuenta de nada
- No permite fácilmente tener varios *servidores* conectados al bus
- Sólo funciona en las distancias cortas a diferencia de, por ejemplo, [RS-232](#), [RS-485](#), o [Bus CAN](#)