

UNIVERSIDAD AUTÓNOMA DE QUERÉTARO

PROGRAMA DE FORMACIÓN DUAL

FACULTAD DE INGENIERÍA

REPORTE DE ACTIVIDADES

*PRÁCTICA #2. COMUNICACIÓN SERIAL ENTRE DOS DISPOSITIVOS*

MENTORES:

Alejandro Rivera Garay

Continental Automotive - Querétaro Campus

Mariano Garduño

Universidad Autónoma de Querétaro

APRENDICES:

Fuentes Flores Lorena

Martínez Olvera Judith

Vite Gonzales Cynthia

Espinosa Bernal Giovanni

Ugalde Romero Dulce Carolina

Santiago de Querétaro, Qro., 10 de Noviembre de 2019.

ÍNDICE

[**I.** **INTRODUCCIÓN** 3](#_Toc24546506)

[**II.** **OBJETIVOS** 3](#_Toc24546507)

[**Objetivo general** 3](#_Toc24546508)

[**III.** **MARCO TEÓRICO** 3](#_Toc24546509)

[**IV.** **MATERIALES** Y **EQUIPO** 4](#_Toc24546510)

[**V.** **METODOLOGÍA** 5](#_Toc24546511)

[**VI.** **RESULTADOS** 5](#_Toc24546512)

[**VII.** **CONCLUSIONES** 7](#_Toc24546513)

1. **INTRODUCCIÓN**

Durante esta práctica se hace una revisión de la comunicación serial y más en específico a las características del protocolo de comunicación serial síncrona SPI necesarias para implementarlo entre una tarjeta NXP configurada como master y otra como slave (característica del protocolo), usando el software S32 Design Studio for ARM Version 2018.R1 tanto para compilar el programa como para debugear. Al final se analizan las señales de este protocolo de comunicación que existen entre las tarjetas y se comparan los resultados con las configuraciones en la tarjeta.

1. **OBJETIVOS**

**Objetivo general**

Enviar y recibir un dato hexadecimal entre dos tarjetas NXP configuradas una como master, y la otra como slave usando el protocolo de comunicación serial para verificar las configuraciones del código en cada una.

1. **MARCO TEÓRICO**

**Comunicación serial**

Este concepto se refiere a la transmisión bit por bit de cada byte y a pesar de su baja velocidad de transmisión comparada con la comunicación paralela, es más usada porque es más barata (sólo un cable), y llega a tener velocidades de transmisión considerables. Dentro de la comunicación serial existen distintos tipos por su aplicación, ejemplos de ellos son I2C, SPI, UART, cada uno con su respectivo protocolo.

**Comunicación serial**

Del acrónimo “Serial Peripheral Interface”

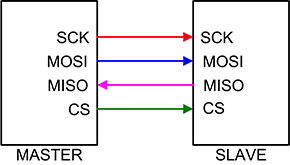
Principales características:

|  |  |
| --- | --- |
| Cables usados | 4 |
| Velocidad Máxima | Más de 10 Mbps |
| Sincronía | Síncrono |
| ¿Serial o Paralelo? | Serial |
| # máximo de maestros | 1 |
| # máximo de esclavos | No limitado en teoría |

Generalmente este protocolo de comunicación tiene sus aplicaciones en distancias pequeñas y por su rapidez, para guardar datos en memoria

**Configuración de cables**

Los cuatro cables usados en la comunicación serial entre tarjeta master y slave [fig 3.3.1] consiguen una transmisión de datos modo full dúplex, es decir, el maestro recibe y envía datos casi al mismo tiempo.



***Figura 1.*** *Cables empleados en el protocolo SPI entre maestro y esclavo.*

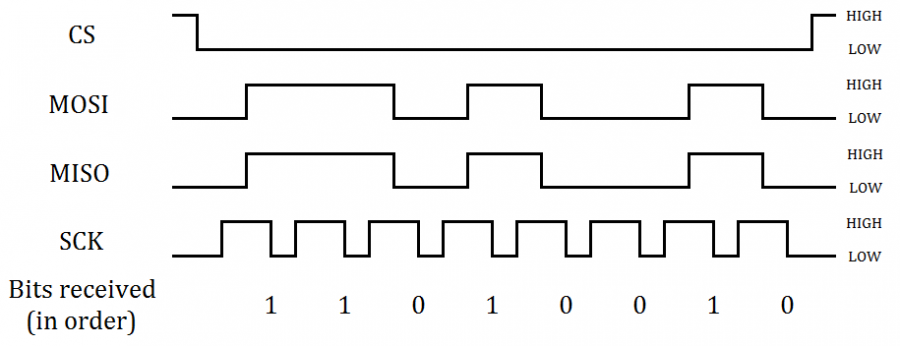
Las señales de datos, MISO y MOSI, corresponden a Master In, Slave Out y Master Out, Slave In respectivamente; SCK corresponde al reloj de sincronización y CS al Chip Select o seleccionador de esclavo ya que puede haber más esclavos.

Cabe reslatar que el Master controla las señales de control, por tnto controla la totalidad de la comunicación.

El cable CS es puesto en alto cuando el maestro desea comunicarse con el esclavo conectado a este mismo, puede haber número ilimitado de esclavos en teoría.

**Envío de datos**

El proceso con base en la figura 3.4.1. Comienza cuando el Maestro de nuestra comunicación baja la línea CS seleccionando así al esclavo correspondiente, después el maestro manda una señal de reloj consistente. Cuando el esclavo detecta un bajo en su línea de CS, comienza a enviar (Línea MISO) o recibir (Líne MOSI) datos



***Figura 2***. Datos en el bus SPI.

1. **MATERIALES Y EQUIPO**
2. Tarjeta de desarrollo NXP tipo: S32K144EVB-Q100
3. Analizador de señales SALEAE
4. Cables duponts
5. Software S32 Design Studio for ARM Version 2018.R1
6. **METODOLOGÍA**

1. Definir los materiales a usar

2. Elaborar los códigos de ambas tarjetas

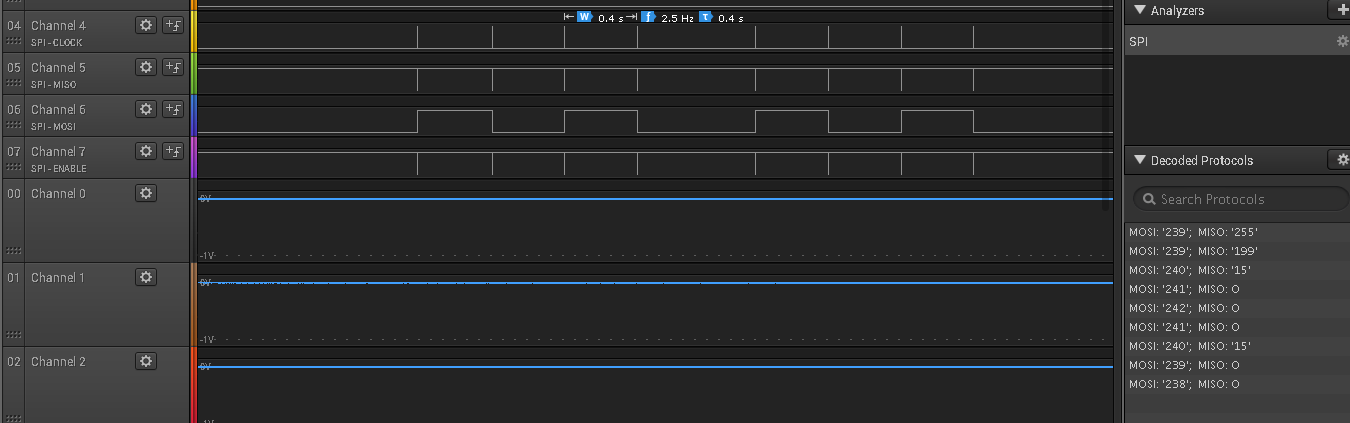
3. Poner en funcionamiento ambas tarjetas con las conexiones necesarias

4. Conectar correctamente el analizador lógico a los pines de señales de la comunicación serial

5. Realizar pruebas y analizar las características de las señales como lo son: fase, polaridad, velocidad de transmisión, frame, número de bits.

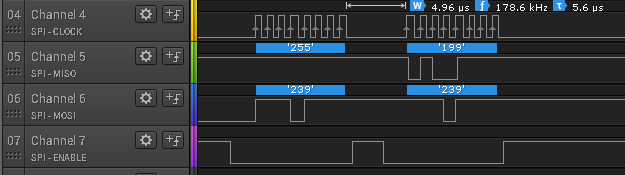
6. Evidencia de los resultados

1. **RESULTADOS**



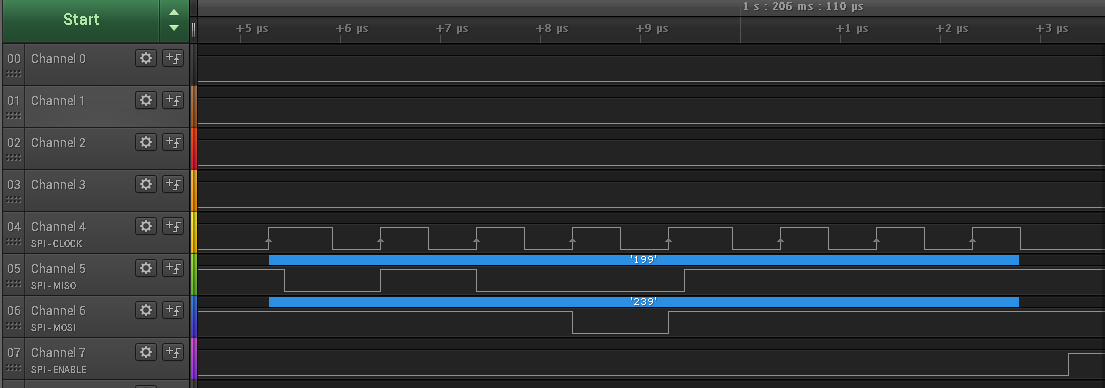
***Figura 3***. Análisis de datos en el bus SPI.

Figra 2: Señales analizadas tanto en transmisor como en receptor (MOSI, MISO) en esquina inferior izquierda



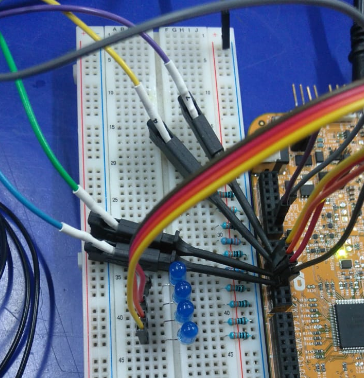
***Figura 4***. Analisis del chip select (CS).

Figura 4: Número de datos enviados por cada Frame: 1



***Figura 5***. Análisis de la señal de reloj (CLK).

Figura 5: La señal del reloj comienza en bajo y sube por lo que la polaridad es 0, además, la captura del dato ocurre medio ciclo después del comienzo del reloj por lo que es una fase de 1.



***Figura 6***. Conexión física de los duponts en la tarjeta master.

Figura 6: Conexión correcta de los duponts morado, amarillo, verde y azul en los pines del puerto b 17,14, 15 y 16 que corresponden a la interfaz del analizador lógico.

1. **CONCLUSIONES**

Se realizó una comunicación full dúplex en ambas tarjetas por lo que se cuidó que primero se transmitiera y luego se recibiera con código, ello provocó que no funcionara ninguna de las dos si todos los cables no estaban correctamente conectados. También el ordenar y encontrar los registros no fue muy sencillo al haber muchos de ellos y usar el código para configurarlos de manera correcta, también cuidamos los errores de sobre flujo al asegurarnos que nos transmitieran datos más grandes.