

PROGRAMA FORMACIÓN DUAL UNIVERSIDAD AUTÓNOMA DE QUERÉTARO

FACULTAD DE INGENIERÍA

Continental

Comunicación SPI

Practica 1

Equipo B

Integrantes:

Cynthia Vite Gonzales
Dulce Carolina Ugalde Romero

Revisión:1.0

Fecha de Revisión: 10/25/2019

Índice

1. Objetivo	2
2. Materiales y Software	2
2.1. Materiales	2
2.2. Software	2
3. Requerimientos	3
3.1. Sección 1	3
3.1.1. Protocolo de comunicación	3
3.1.2. Velocidad de comunicación	3
3.1.3. <i>Chip-select</i>	3
3.1.4. Paridad	3
3.1.5. Dirección del esclavo	3
3.1.6. <i>Hand-shaking</i>	4
3.1.7. Numero de datos	4
3.1.8. Diccionario de mensajes	4
3.1.9. Verificación	4
3.2. Sección 2	5
4. Referencias	6
5. Anexos	6
5.1. GITRepository:	6

Índice de figuras

1. Diagrama de arquitectura	4
2. Diagrama de funcionamiento.	5

Índice de cuadros

1. Objetivo

Simular el funcionamiento “*Window Lifter*” con comunicación serial SPI en la tarjeta de desarrollo NXP.

2. Materiales y Software

2.1. Materiales

El equipo y materiales utilizado para el desarrollo de la práctica es el siguiente:

- Tarjeta de desarrollo NXP S32K144EVB.
- Cable de alimentación USB a microUSB.
- Osciloscopio a colores marca Tektronix TDS 2024B, 200[MHz], código de almacén-7138.
- Cables dupon macho-macho.

2.2. Software

Los software utilizados para el desarrollo de la práctica es el siguiente:

- S32 Design Studio for ARM version .
- Logic 1.2.18.

3. Requerimientos

3.1. Sección 1

En esta sección se definen los requerimientos generales y específicos para el desarrollo de la práctica y la identificación de cumplimiento para cada uno.

3.1.1. Protocolo de comunicación

Establecer el protocolo de comunicación SPI (Serial Peripheral Interface).

- Definir un maestro y un esclavo .
- Seleccionar el canal de comunicación del SPI

3.1.2. Velocidad de comunicación

Es necesario establecer la velocidad a la cual se realizara la comunicación entre los dispositivos.

- Para ambos, maestro y esclavo, será una velocidad establecida de 9600 bits/segundo.
- Se deberá verificar el registro en ambos dispositivos.
- Para comprobar que la velocidad fue bien establecida, se deberá verificar por medio de un osciloscopio que la oscilación del reloj interno del dispositivo coincida.

3.1.3. *Chip-select*

Establecer un chip-select (conecta o desconecta la operación del dispositivo con el que uno desea comunicarse, para multiplexeo).

3.1.4. Paridad

- Se definirá como paridad par.
- Se verificara la paridad al observar la trama de datos en un osciloscopio.
- Para comprobar que la velocidad fue bien establecida, se deberá verificar por medio de un osciloscopio que la oscilación del reloj interno del dispositivo coincida.

3.1.5. Dirección del esclavo

Identificar la dirección del esclavo. Y realizar la configuración de sus registros, para que coincidan con aquellos establecidos en el maestro.

3.1.6. *Hand-shaking*

Establecer si habrá o no hand-shaking en la trama de datos.

3.1.7. Numero de datos

El primer byte (*frame*) se utilizara para establecer el número de datos a transmitir.

3.1.8. Diccionario de mensajes

Se crea un diccionario o de mensajes dependiendo de lo que se desea realizar.

- Se establece el mismo diccionario en el esclavo y en el maestro.
- Se define le mensaje deseado de acuerdo al diccionario.

3.1.9. Verificación

Se verificara que el objetivo de la práctica se cumple y que la comunicación entre dispositivos sea la correcta mediante los siguientes puntos:

- Entrega funcional de la práctica.
- Observar la señal de *clock* en el osciloscopio.
- Observar el mensaje enviado en el bus completo.
- Observar la señal de *Hand-shaking* en el osciloscopio.

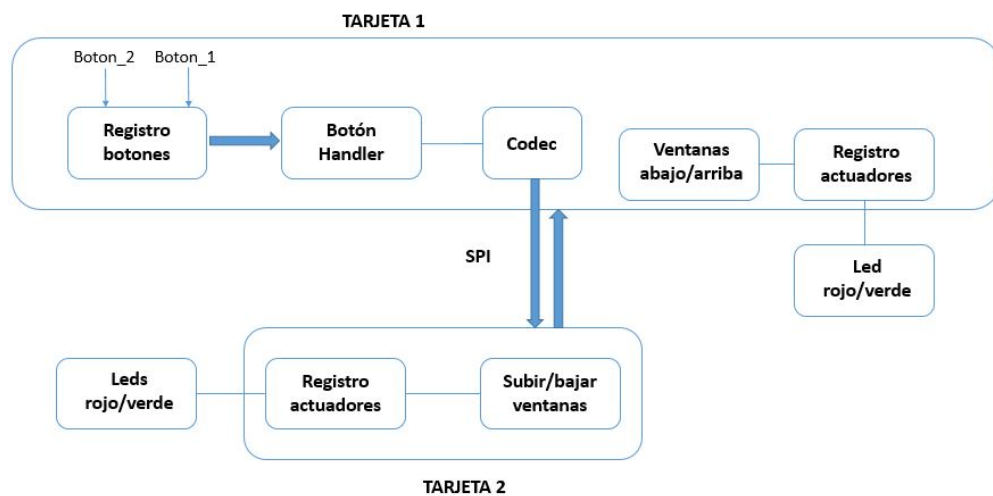


Figura 1: Diagrama de arquitectura

3.2. Sección 2

El sistema será capaz de identificar cuando, cuál y por cuánto tiempo se presionó el botón de interrupción, por cada interrupción deberá existir un evento.

- Si se oprime el boton1 se enviará el primer mensaje para subir la ventana, simulándolo con una secuencia ascendente de cuatro leds.
- Si se oprime el boton2 el mensaje enviado indicará que la ventana debe bajar, simulándolo con una secuencia descendente de cuatro leds.
- Al finalizar las secuencias ascendente y descendente, la tarjeta 2 enviará un mensaje a la tarjeta 1 para indicar el estado de la ventana.
- El led rojo en la tarjeta 1 indicará que la ventana está completamente arriba.
- El led verde en la tarjeta 1 indicará que la ventana está completamente abajo.

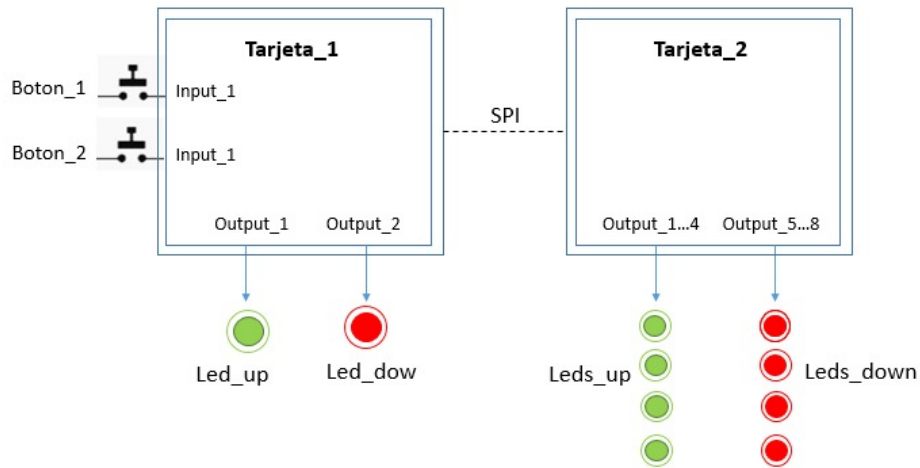


Figura 2: Diagrama de funcionamiento.

4. Referencias

- ” Próximos a añadirse”

5. Anexos

5.1. GITRepository:

- <https://github.com/CynthiaVite/Practica-1.git>