

MÓDULO NFC (I2C)

Cristhian Ferney Pulido Garcia

Daniel Alexander Cruz.

Estudiantes, Universidad Nacional de Colombia
cfpulidog@unal.edu.co, danacruzui@unal.edu.co

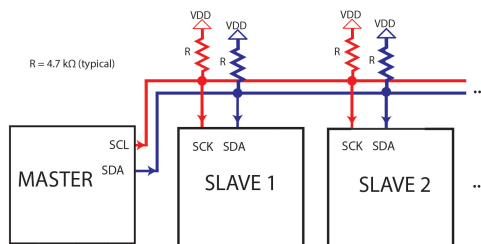
Resumen—El presente documento describe el diseño del módulo NFC (I2C), uno de los cuatro módulos que componen el proyecto Candado inteligente; este diseño se realizó a partir de la implementación del protocolo de comunicación serial I2C, el cual se caracteriza por establecer dos vías de comunicación entre el maestro y uno o más esclavos, en nuestro caso la FPGA NEXYS 4 y el controlador de comunicación de campo cercano (NFC) PN532/C1, respectivamente.

I. PROTOCOLO I2C

■ Qué es I2C

El circuito Integrado I2C, es un protocolo de comunicación serial desarrollado por Phillips Semiconductors en la década de los 80s, creado para poder comunicar varios chips al mismo tiempo dentro de los televisores. Con el protocolo I2C podemos tener a varios maestros controlando uno o múltiples esclavos. Esto puede ser de gran ayuda cuando se van a utilizar varios microcontroladores para almacenar un registro de datos hacia una sola memoria o cuando se va a mostrar información en una sola pantalla.

El protocolo I2C utiliza sólo dos vías o cables de comunicación, SDA y SCL.



SDA – Serial Data. Es la vía de comunicación entre el maestro y el esclavo para enviarse información.

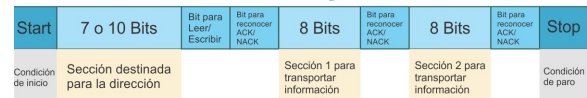
SCL – Serial Clock. Es la vía por donde viaja la señal de reloj.

■ Cómo funciona I2C

El protocolo I2C envía información a través de una sola vía de comunicación. La información es enviada bit por bit de forma coordinada en forma de mensajes. Los mensajes van divididos en tramas de datos. Cada mensaje lleva un trama con una dirección la cuál transporta la dirección binaria del esclavo al que va dirigido el mensaje, y una o más tramas que llevan la información del mensaje. El mensaje contiene condiciones de inicio

y paro, lectura y escritura de bits, y el bit ACK.

Mensaje



Init: La vía SDA cambia de un nivel de voltaje Alto a un nivel de voltaje Bajo, antes de que el canal SCL cambie de Alto a nivel Bajo.

Reset: La vía SDA ahora cambia de un nivel de voltaje Bajo a Alto, después de que la vía SCL cambia de Bajo a Alto.

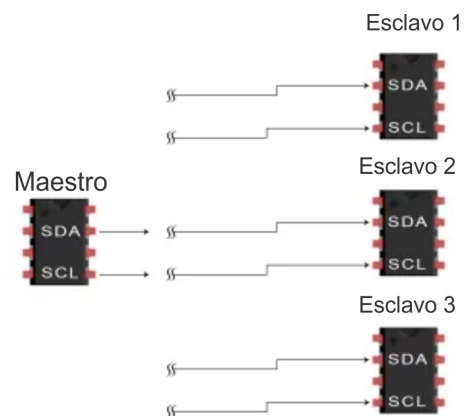
Address Frame: Es una secuencia única que va de los 7 a los 10 bits. Este sección (Frame) se envía a cada Esclavo, y va a identificar al Esclavo con el que el Maestro se quiere comunicar.

Read/Write Bit A: Es un bit de información enviado a los Esclavos. Por medio de este bit el Maestro indica si le va enviar información al Esclavo (Nivel Bajo de voltaje = Escritura), o si el Maestro quiere solicitarle información al Esclavo (Nivel Alto de Voltaje = Lectura).

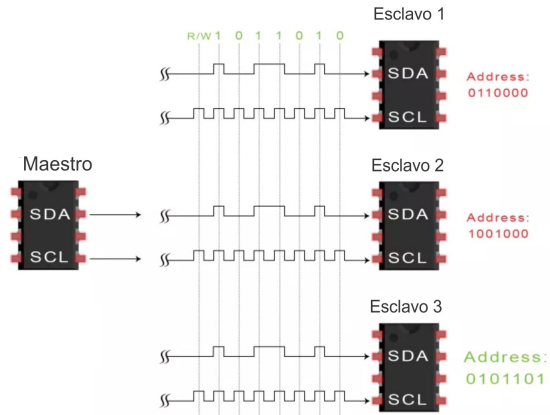
Bit ACK : Después de cada sección (Frame) de información enviada en un mensaje, podemos notar que lleva un bit acknowledge. Esto ayuda a identificar si la información fue enviada correctamente.

■ Pasos de ejecución del I2C

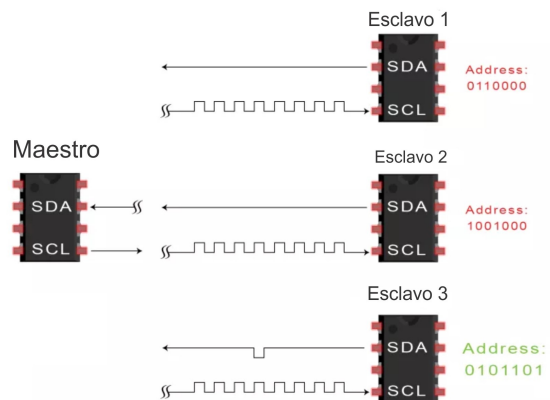
1.- El Maestro envía la condición de Inicio (Start) a cada Esclavo que esté conectado en la vía SDA, cambia el nivel de voltaje a Bajo, y deja la vía SCL en estado Alto.



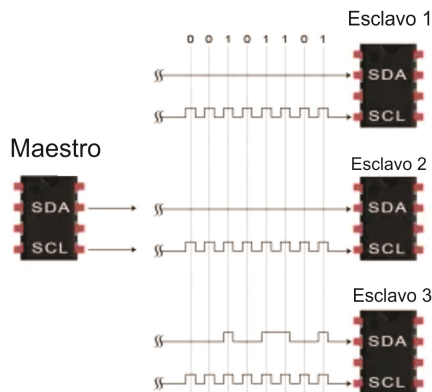
2.- El Maestro envía la dirección de 7 a 10 bits, a cada uno de los Esclavos para identificar al Esclavo con el que se quiere comunicar.



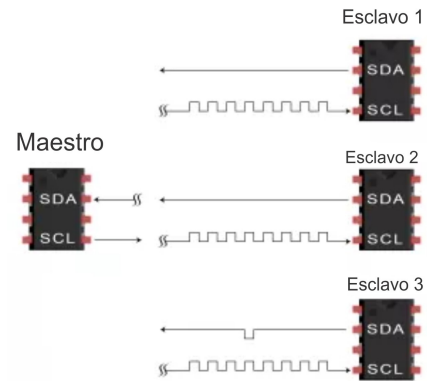
3.- Cada Esclavo recibe la dirección y la compara con su propia dirección. Si la dirección coincide, el Esclavo envía un bit ACK, y pone la vía SDA en nivel Bajo de voltaje. Si la dirección no es la misma, entonces los Esclavos no hacen nada y dejan la vía SDA en el mismo nivel de voltaje Alto.



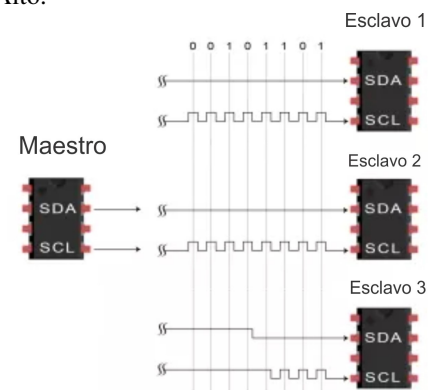
4.- El Maestro envía o recibe los Frames de información.



5.- Después de que cada Frame de información fue enviado, el dispositivo que recibe (ya sea Esclavo o Maestro) va a enviar un bit ACK al remitente, para notificarle que la información se recibió exitosamente.

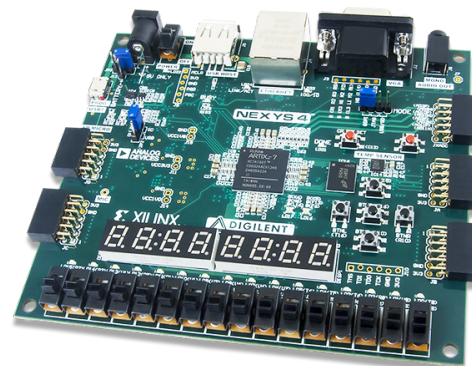


6.- Para concluir la transmisión de información, el Maestro envía al Esclavo la condición de paro (Stop) con un nivel Alto en la vía SDA, cuando cambia el estado de SCL a Alto.



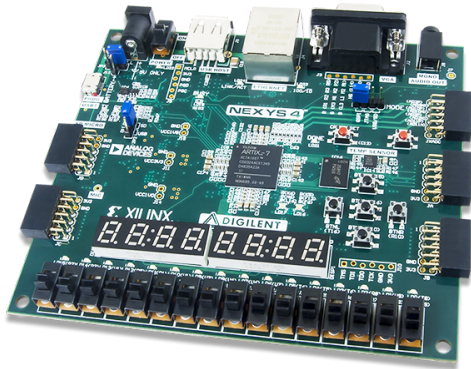
II. NEXYS 4

La NEXYS 4 es una tarjeta de desarrollo en la cual se pueden realizar una gran variedad de diseños, con la facilidad de poder probarlos con sus muchos dispositivos ya instalados. Desempeña el papel de maestro en la implementación del protocolo I2C, por lo cual se le asignan las salidas SDA y SCL a dos de sus puertos PMOD, la salida SCL corresponde al Clock de la NEXYS, el cual funciona a una frecuencia de 100 MHz. Por otro lado, se le asignan los registros *sysclk_i*, *sysrst_i*, *d_i n*, *cs*, *addr*, *rd*, *wr*, *d_o ut*, *sda*, *scl* a sus interruptores y pulsadores dando la posibilidad de generar diversas combinaciones de datos binarios.



III. NFC PN532/C1,

La NEXYS 4 es una tarjeta de desarrollo en la cual se pueden realizar una gran variedad de diseños, con la facilidad de poder probarlos con sus muchos dispositivos ya instalados. Desempeña el papel de maestro en la implementación del protocolo I2C, por lo cual se le asignan las salidas SDA y SCL a dos de sus puertos PMOD, la salida SCL corresponde al Clock de la NEXYS, el cual funciona a una frecuencia de 100 MHz. Por otro lado, se le asignan los registros sys_clk_i , sys_rst_i , $d_i n$, cs , $addr$, rd , wr , $d_o ut$, sda , scl a sus interruptores y pulsadores dando la posibilidad de generar diversas combinaciones de datos binarios.



IV. NFC PN532/C1,

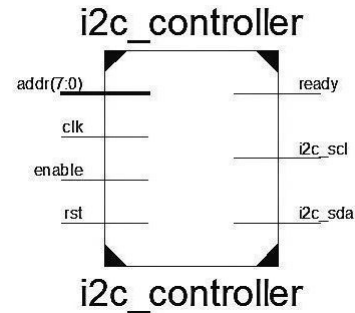
Shield RFID NFC lector PN532 es una tarjeta para leer módulos de RFID/NFC de frecuencias de 13.56Mhz. Puede trabajar por protocolo I2C, HSU o SPI. La conexión de I2C se realiza mediante dos líneas, una es SDA línea de datos y la otra es SCL/CLK línea de reloj. Las dos líneas son bidireccionales y de tipo Half Duplex que quiere decir que quiere decir que se comunican en una sola dirección a la vez. La velocidad de trabajo es de 100Kbits por segundo hasta 3.4Mbit por segundo. Las líneas de comunicación SDA y SCL deben tener resistencias pull-up de 10Kohms, esto para que en el momento de que no exista comunicación entre los dispositivos las líneas permanezcan en nivel alto.



V. CAJA NEGRA

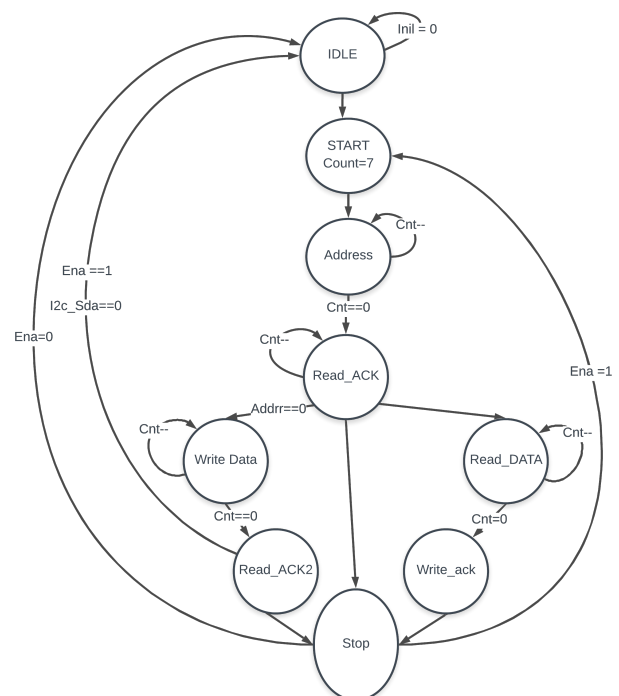
Es un elemento que se estudia desde el punto de vista de las entradas que recibe y las salidas o respuestas que produce,

sin tener en cuenta su funcionamiento interno. De una caja negra nos interesará su forma de interactuar con el medio que le rodea entendiendo qué es lo que hace, pero sin dar importancia a cómo lo hace.

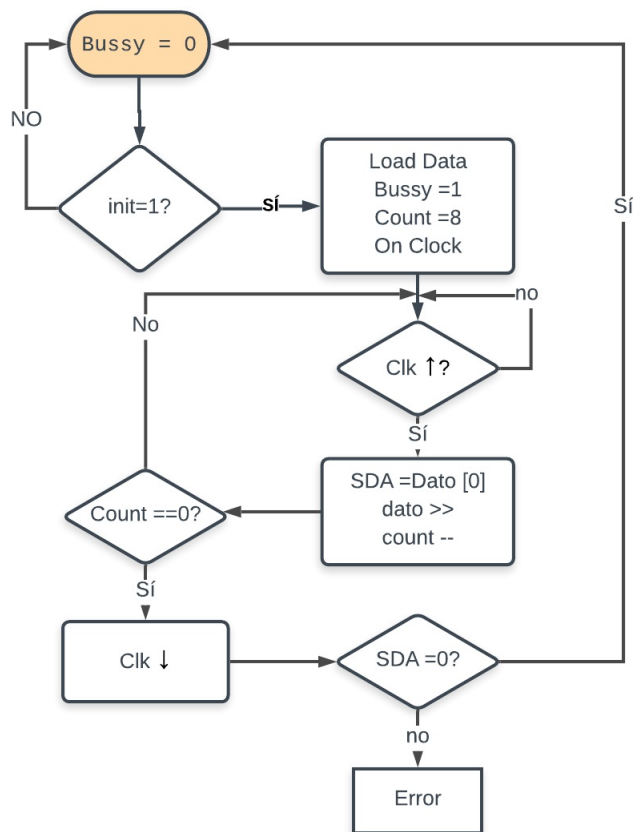


VI. MAQUINA DE ESTADOS

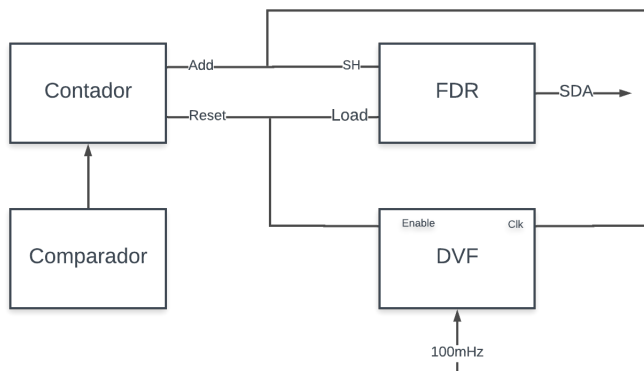
Una máquina de estados finita es un modelo usado para diseñar circuitos lógicos secuenciales. No puede estar en más de un estado por vez. El estado en el que se encuentra se denomina estado actual. El cambio de un estado a otro se denomina transición, y se dispara con el reloj del circuito. Son muy útiles en el diseño de protocolos de comunicación. En la implementación de nuestro protocolo utilizamos 9 estados, IDLE, Start, Address, Read ACK, Write Data, Read Data, Read Ack2, Write ACK y Stop.



VII. DIAGRAMA DE FLUJO

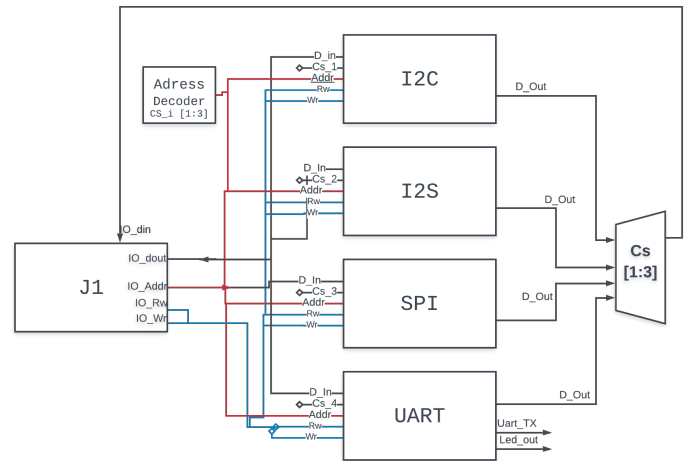


VIII. DATA PATH



IX. J1 Soc

El J1 es una central de procesamiento dedicada para ser implementada en FPGAs. Su funcionamiento se fundamenta en una arquitectura de 16 bits, basada en un procesador de tipo Von Neumann. Se le designa Soc (sistema en Silicio) a la conexión existente entre la CPU y los periféricos que se conectan a él.



X. BIBLIOGRAFIA

<https://hetpro-store.com/TUTORIALES/shield-rfid-nfc-lector-pn532-chico/>
<https://teslabem.com/nivel-intermedio/fundamentos-del-protocolo-i2c-aprende/>
<https://www.nxp.com/docs/en/nxp/data-sheets/PN532C1.pdf>
<https://sites.google.com/site/logicaprogramable/calculadoras/fpga/nexys-4-artix-7-fpga-board>
<https://www.monografias.com/docs114/maquinas-estado/maquinas-estado.shtml>
<https://sites.google.com/site/proyectodigital2/soc-Ferney-Beltran>