

UNIVERSIDAD SERGIO ARBOLEDA

SISTEMAS EMBEBIDOS

INGENIERÍA ELECTRÓNICA

WENDY JOHANNA CHAPARRO – CRISTOPHER RAMÍREZ RAMÍREZ

INFORME DE LABORATORIO #3

Para este laboratorio se hizo uso de la placa de desarrollo STM32F411 con el programa STM32CUBE IDE con el objetivo de decodificar las tramas enviadas por un control remoto, esto mediante el uso de un receptor infrarrojo para conocer el código (comando y dirección), representado en binario, de la tecla que se está presionando. Para comprender el código que se obtuvo fue necesario investigar sobre el protocolo de comunicación del control remoto, en este caso un control Sony, para controlar dispositivos a través de señales infrarrojas (IR) y el manejo del counter (timer) en la stm32F411.

Al obtener el código, se muestra la dirección en unos LED's, los cuales representan los bits de la dirección, y posteriormente se muestra el comando y la dirección de la tecla en dichos LED's. Finalmente se eligen 8 botones del control, los cuales serán representados por cada uno de los LED's que se encenderán cuando se presione dicha tecla del control remoto.

IMPLEMENTACIÓN DEL PROTOCOLO:

En este laboratorio se utilizó un control remoto de la marca SONY. Este control remoto utiliza el protocolo SIRC (Sony Infrared Remote Control) que permite controlar dispositivos a través de señales infrarrojas (IR). Es uno de los protocolos de control remoto más utilizados y permite que un mando a distancia envíe comandos a un dispositivo, como un televisor, reproductor de DVD, o sistema de sonido, para realizar acciones como encender, apagar, cambiar de canal, ajustar el volumen, etc. [2]

Características del protocolo SIRC:

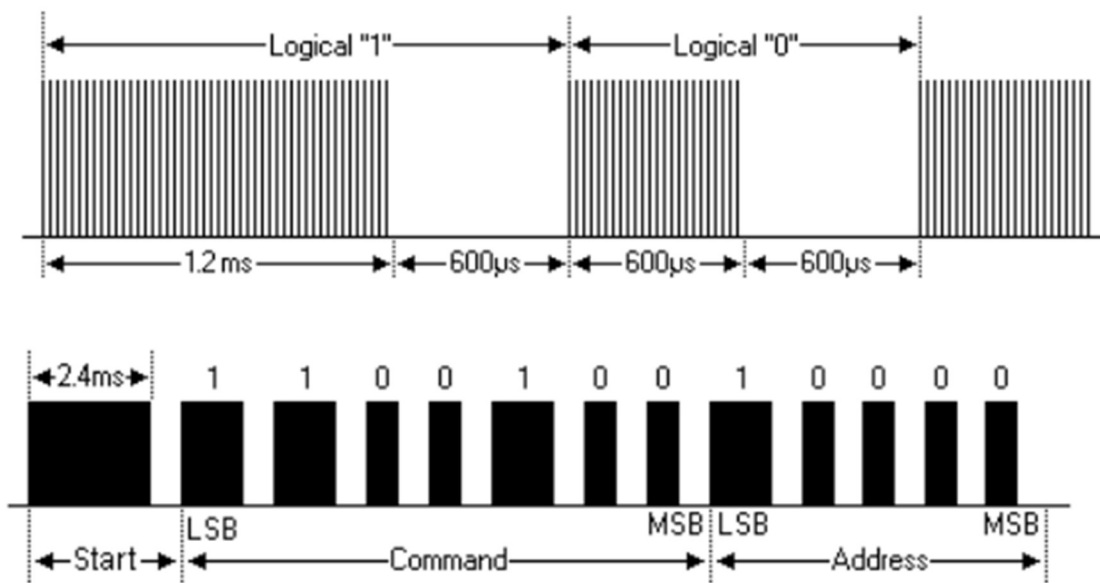
1. **Modulación:** La modulación en el protocolo SIRC es el proceso mediante el cual las señales digitales (1 y 0) se convierten en señales infrarrojas (IR) que pueden ser transmitidas por el mando a distancia y recibidas por el dispositivo. [1]
 - **Funcionamiento de la modulación en SIRC:**
 - ❖ **Frecuencia portadora:** El protocolo SIRC utiliza una frecuencia portadora de 40 kHz.
 - ❖ **Codificación de los bits:** En el protocolo SIRC, los datos digitales (1 y 0) se codifican usando una técnica conocida como PWM (Pulse Width Modulation).

Cada bit de datos se representa por un tren de pulsos infrarrojos que dura un tiempo específico.

❖ Duración de los pulsos:

- En SIRC, un "1" lógico se representa por un pulso de 1.2 ms de duración de luz infrarroja encendido, seguido por un período de 0.6 ms de luz apagada.
- Un "0" lógico se representa por un pulso de 0.6 ms de duración de luz infrarroja encendida, seguido por un período de 0.6 ms de luz apagada.

❖ Bit de inicio: Antes de que se transmitan los bits de datos, se envía un bit de inicio, que es un pulso de 2.4 ms de luz infrarroja encendida seguido por un período de 0.6 ms de luz apagada. Esto le indica al receptor que va a comenzar la transmisión de un comando.



2. **Formato de señal:** SIRC utiliza una secuencia de pulsos de luz infrarroja para transmitir datos. Cada comando se compone de una serie de bits que representan diferentes parámetros, como la dirección del dispositivo (por ejemplo, TV, DVD) y el código del botón presionado en el mando. [1]
3. **Estructura de la señal:** La señal SIRC típica tiene 12, 15, o 20 bits, dependiendo del modelo del dispositivo y el mando. Esta estructura incluye un bit de inicio seguido por bits que representan el comando y la dirección del dispositivo, para este caso se encontró un protocolo de 12 bits, de los cuales los primeros 7 corresponden al comando y los otros 5 a la dirección. [1]

LIBRERIAS UTILIZADAS:

Para este laboratorio se hizo uso de las librerías HAL de stm32cube IDE, haciendo uso de las funciones para los temporizadores [3].

Funciones de la HAL para Temporizadores:

La HAL_TIM es una biblioteca en STM32 que proporciona una interfaz de alto nivel para controlar temporizadores. Sus funciones principales incluyen:

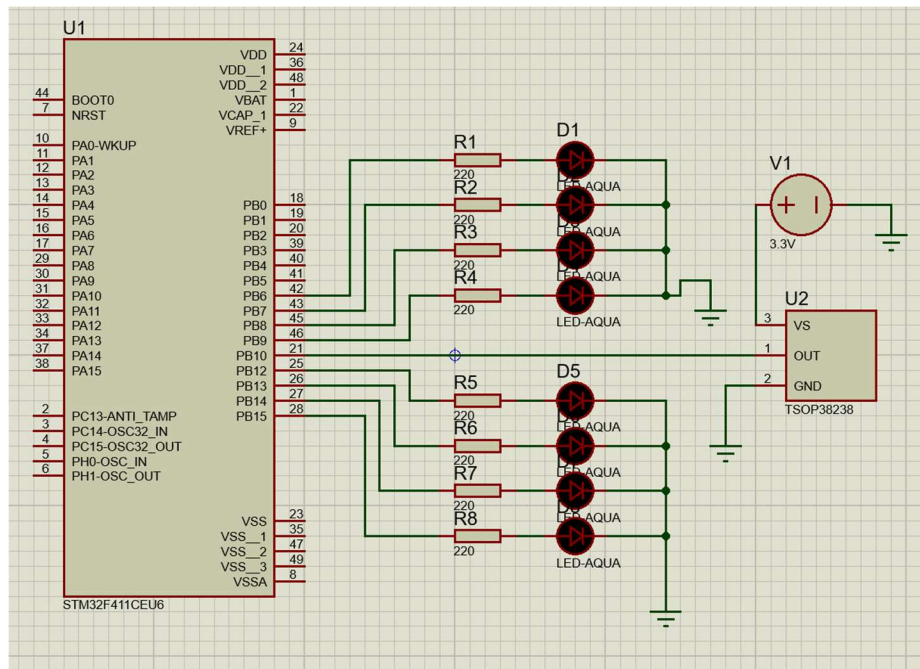
- HAL_TIM_Base_Init(): Inicializa la unidad de tiempo base del temporizador.
- HAL_TIM_Base_DeInit(): Desactiva el temporizador.
- HAL_TIM_Base_MspInit(): Realiza la configuración necesaria para el hardware específico durante la inicialización.
- HAL_TIM_Base_MspDeInit(): Revertir la configuración del hardware.
- HAL_TIM_Base_Start(): Inicia el temporizador.
- HAL_TIM_Base_Stop(): Detiene el temporizador.
- HAL_TIM_Base_Start_IT(): Inicia el temporizador en modo de interrupción.
- HAL_TIM_Base_Stop_IT(): Detiene el temporizador en modo de interrupción.
- HAL_TIM_Base_Start_DMA(): Inicia el temporizador en modo DMA.
- HAL_TIM_Base_Stop_DMA(): Detiene el temporizador en modo DMA.

ANÁLISIS Y DESARROLLO:

Para comenzar con la ejecución de este laboratorio se tuvo como punto de partida el análisis del comportamiento de los pulsos como si estos fueran un pulsador físico y como se haría el conteo de las pulsaciones, cuestión que permito entender un poco el funcionamiento, pero de igual manera trajo consigo inconvenientes como el hecho de que en un pulsador se debe tener en cuenta el efecto rebote.

Entendiendo la dinámica de cómo se cuenta las pulsaciones en un pulsador físico y reconociendo las diferencias con los pulsos enviados por el control, fue fundamental utilizar las librerías de HAL_TIM y comprender el protocolo para manejar el tiempo de los pulsos en la modulación de la señal enviada por el control remoto, a fin de decodificar correctamente las señales recibidas. Además, fue crucial el uso de un osciloscopio, que permitió observar las señales obtenidas por el receptor de infrarrojo, brindando una comprensión más precisa de los tiempos de duración específicos del control.

Con esta comprensión, se decidió utilizar un prescaler de 20000 para obtener un tiempo entre conteos de 200 microsegundos. Luego, se tomaron como variables el tiempo del pulso en alto y en bajo para determinar cuáles pulsos correspondían a un 1 lógico o un 0 lógico. Una vez determinado el valor de un pulso, este se almacenó en cada posición de un arreglo de 12 posiciones. De esta manera, se logró decodificar la dirección y el comando enviado por la tecla presionada en el control remoto.



CONCLUSIONES:

1. Es fundamental conocer el tipo de protocolo que se está trabajando y hacer las pruebas necesarias previas a la codificación, para optimizar al máximo lo que se desea obtener.
2. Por medio de mediciones reales en el osciloscopio se identificó que si bien las duraciones de los pulsos coinciden con lo investigado del protocolo SIRC estos estaban invertidos por la alimentación del circuito lo que fue fundamental en el desarrollo de la práctica.
3. Tener como referente el funcionamiento de un pulsador físico ayudo en la comprensión del funcionamiento y el flujo del código.
4. Es mucho mejor poder hacer uso del timer que tiene el microcontrolador que hace uso únicamente de los Delays. Las herramientas que proporciona la librería HAL para usar los timers ayudan a ajustar los tiempos a las necesidades o requerimientos que se tengan a la hora de elaborar un proyecto, optimizando recursos.
5. Esta práctica permitió conocer como es el funcionamiento de los controles remotos y el tipo de protocolos que manejan, ya que estos son los que hacen posible la

comunicación de estos controles con los dispositivos a los que se transmite la información. Además de entender que esta señal contiene un mensaje en el que se indica la dirección (la funcionalidad que se quiere controlar) y el comando (representa la tecla que se presionó).

BIBLIOGRAFÍA:

[1] S. Bergmans, «SB-Projects - IR - Sony SIRC Protocol».

<https://www.sbprojects.net/knowledge/ir/sirc.php>

[2] «SONY SIRC Protocol.» [https://faculty-](https://faculty-web.msoe.edu/johnsontimj/Common/FILES/sony_sirc_protocol.pdf)

[web.msoe.edu/johnsontimj/Common/FILES/sony_sirc_protocol.pdf](https://faculty-web.msoe.edu/johnsontimj/Common/FILES/sony_sirc_protocol.pdf)

[3] «Notas de desarrollo de la biblioteca HAL - Temporizador básico TIM - Power's Wiki».

[https://wiki-](https://wiki-power.com/es/HAL%E5%BA%93%E5%BC%80%E5%8F%91%E7%AC%94%E8%AE%B0-TIM%E5%9F%BA%E6%9C%AC%E5%AE%9A%E6%97%B6%E5%99%A8/#configuracion-basica-del-temporizador-en-el-codigo)

[power.com/es/HAL%E5%BA%93%E5%BC%80%E5%8F%91%E7%AC%94%E8%AE%B0-TIM%E5%9F%BA%E6%9C%AC%E5%AE%9A%E6%97%B6%E5%99%A8/#configuracion-basica-del-temporizador-en-el-codigo](https://wiki-power.com/es/HAL%E5%BA%93%E5%BC%80%E5%8F%91%E7%AC%94%E8%AE%B0-TIM%E5%9F%BA%E6%9C%AC%E5%AE%9A%E6%97%B6%E5%99%A8/#configuracion-basica-del-temporizador-en-el-codigo)