

LABORATORIO DE MICROCONTROLADORES

MEMORIAS EEPROM Y FLASH

Barrera. Yesica, Mora. Jawer

Yesica.barrera@uptc.edu.co, Jawer.mora@uptc.edu.co

Universidad pedagógica y tecnológica de Colombia

Seccional Sogamoso – Ingeniería Electrónica

I Semestre de 2017

RESUMEN: En el presente informe se realiza un generador de señales del cual se obtienen tres señales distintas (señal seno, señal cuadrada, señal diente de sierra), esto usando la aplicación de Microchip MPLAB para la creación, simulación y programación del microcontrolador PIC18f4550. En esta aplicación se hace primordial los conocimientos con respecto a los timers, memorias EEPROM, FLASH, RAM e interrupciones, de igual modo las librerías necesarias para el funcionamiento de una pantalla LCD y una matriz 4x4, así como también la creación de funciones en C para las diferentes tareas a realizar en la aplicación.

Palabras clave: Microcontrolador, Timers, Interrupciones, Memorias EEPROM y FLASH.

1. INTRODUCCIÓN

Los microcontroladores son circuitos programables que contienen todos los elementos necesarios para desarrollar y controlar una tarea determinada. La cantidad de componentes que se integran a los microcontroladores depende del diseño de los fabricantes, sin embargo, los elementos básicos suelen ser: microprocesador, memoria RAM, memoria de programa, convertidor A/D, oscilador, puerto de comunicación, etc. Esto le ha brindado una gran versatilidad a este tipo de dispositivos y hoy en día su utilización se ha incrementado enormemente en el mundo. [1]

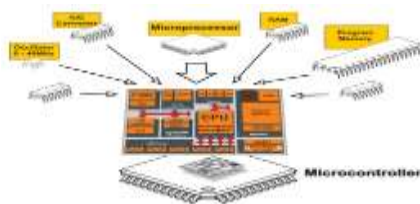


Fig. 1. Componentes de los microcontroladores [2]

Dependiendo las características de cada microcontrolador y obedeciendo a la aplicación en la cual se va a utilizar se hace necesario que el programador determine qué tipo de lenguaje se puede utilizar para cada tarea a realizar. En el caso de esta práctica se hace necesario utilizar un lenguaje de alto nivel (C) ya que se utiliza el PIC18f4550 debido a utilizar los conocimientos adquiridos en clase.

En el presente informe se especifican las tareas requeridas para las dos últimas prácticas de laboratorio del área de microcontroladores con las cuales se quiere reforzar los conocimientos en clase, acerca de: Memoria RAM, direccionamiento indirecto, set de instrucciones, visualización estroboscópica, funcionamiento y utilidad de las memorias EEPROM y FLASH, conversor Analógico a Digital, módulos de temporización por hardware, aplicaciones en lenguaje C, funcionamiento de los protocolos de comunicación SPI, I2C, RS-232, manejo adecuado dispositivos de interfaz de usuario (LCD, y Teclado matricial 4X4).

2. MARCO CONCEPTUAL

MPLAB® X IDE: Es un software que es usado para desarrollar aplicaciones para microcontroladores de microchip y control de señales digitales, estas herramientas de desarrollo comúnmente se les conoce como IDE o entorno de desarrollo integrado[1].

DIAGRAMAS DE FLUJO: La importancia de un diagrama de flujo es que hace al software más entendible, esto ayudara a visualizar de distintas formas la complejidad del problema y unas posibles soluciones, sin embargo al ofrecer este tipo de ayuda no hace comprensible un software para muchos, esto depende del alto nivel de complejidad del código como lo cita el autor[2].

LA MEMORIA FLASH [4]: Esta Memoria puede ser leída escrita y borrada durante el funcionamiento del dispositivo siempre que sea correcta la tensión de alimentación Vdd. La lectura se efectúa sobre un byte en

cada operación, la escritura se realiza en bloques de 8 bytes y el borrado de 64 bytes.

Elementos primordiales para la práctica:

CALCULO DE RETARDOS

Se programa una rutina de retardo de 2 segundos para una visualización ideal de cada dato en el puerto.

$$t = \frac{4}{f_{osc}} * (2 + (3 * M * N + 5 * M) * k) \quad [1]$$

Usando la Ec. [1] se puede determinar el retardo de 2 segundos despejando K se halla el valor a ingresar mediante el lenguaje ensamblador.

CONVERSIÓN DIGITAL-ANALÓGICA R2R [5]

Los conversores digital análogo realizan sus conversiones recibiendo la información en forma serial o paralela, La decisión de emplearlos en serie o paralelo se basa en el uso que uno desea, como por ejemplo en instrumentos de medida como osciloscopios de almacenamiento digital.

Se emplea la conversión de tipo paralela y en aplicaciones del control de proceso como válvulas se puede efectuar en forma serie.

Existen dispositivos Análogos-Digitales que convierten un nivel de tensión analógico en una palabra digital correspondiente. Si n es el número de bit obtenidos de la palabra, esto significa que habrá 2n niveles de tensión diferentes. Todo convertidor de este tipo debe procurar que el conjunto de bit obtenidos a la salida sea un reflejo, lo más exacto posible del valor digital correspondiente.

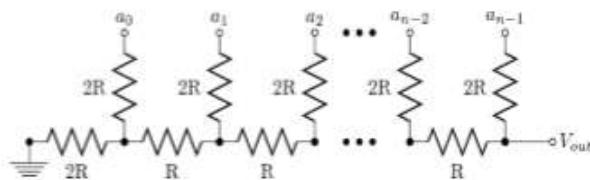


Fig. 2¹. Esquema de una red R2R

¹ Figura tomada Panama Hitek por Antony García, Kiara Navarro y su equipo creativo.

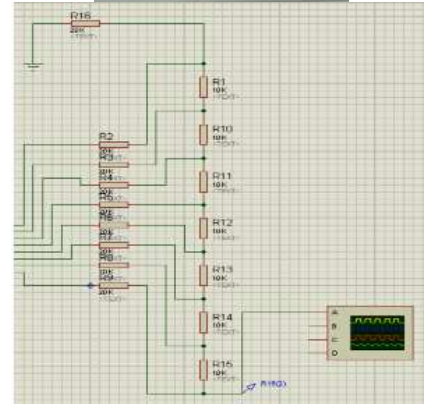


Fig. 3. Circuito implementado en protoboard y circuito utilizado en simulación, Proteus.



Fig. 4. Circuito implementado en protoboard

DESCRIPCION PROGRAMA PRINCIPAL

La aplicación desarrollada consiste en que el microcontrolador genere tres clases de señales diferentes en el dominio del tiempo las cuales son: seno, cuadrada y diente de sierra.

Para esto, el microcontrolador recibe las instrucciones sobre el tipo de señal que se quiere y el valor de frecuencia de la señal que debe generar mediante un teclado matricial. Como las salidas de datos del microcontrolador son digitales, se usa el conversor D/A R2R.

En esta aplicación se hace esencial el uso de tablas para la obtención de las señales y frecuencias requeridas para la práctica.

ESTRUCTURA DE UN PROGRAMA EN C

C es el lenguaje de programación que utilizaremos. Un programa C está compuesto por una o varias funciones, dentro de las cuales tendremos una o más instrucciones. Existe una función llamada “main” donde tendremos nuestro código principal.

Comentarios: Los comentarios son de mucha utilidad para entender lo que escribimos. Para realizar un comentario antepone el símbolo “//” y el compilador lo ignorará. También podemos comentar bloques de código utilizando “/*” para abrir y “*/” para cerrar.

Includes: utilizando “#include” incluimos en nuestro programa archivos de funciones o librerías que nos permitirán utilizar características especiales.

3.2 DESCRIPCION DEL PROGRAMA

Se diseña un generador de señales, donde se muestra una señal triangular, diente de sierra y cuadrada, con un rango de frecuencias de 1 a 255 Hz.

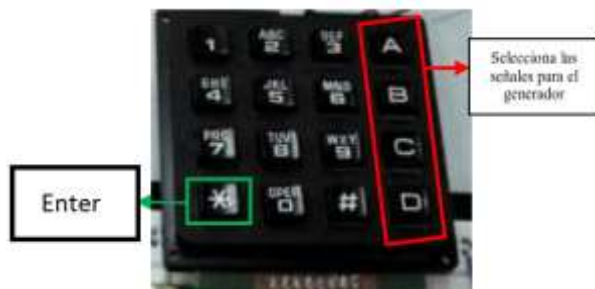


Fig 5. Teclado matricial 4X4

El funcionamiento del teclado está definido por el contacto que se ocasiona al oprimir una tecla, este dispositivo está compuesto por filas y columnas, cada vez que hay contacto eléctrico entre estas, ocasiona un cambio de estado en el puerto en el que se conecta al microcontrolador, dicho cambio de estado genera una interrupción, que hace que el micro testeé el estado del teclado para reconocer la tecla que se oprimió.

Tal y como se muestra en la figura 4 la configuración del funcionamiento del sistema depende de las teclas que se opriman.

A continuación se define cada configuración posible en el sistema.

El tipo de señal generada se configura con las teclas A,

B, C y D, las cuales equivalen a las señales triangular, diente de sierra, cuadrada y sinusoidal respectivamente.

Para el cambio de frecuencia en el modo generador, se emplean los valores emitidos por el teclado una vez se presiona la tecla *, estos valores dependen de la combinación de las teclas 1, 2, 3, 4, 5, 6, 7, 8, 9, 0

El cambio de frecuencia se realiza modificando la configuración del periférico TMR2 modificando el contenido de los registros PR2 y T2CON para lograr una buena resolución de las señales a cada frecuencia exigida.

En la LCD se visualiza el tipo de señal generada y la frecuencia de cada una de estas (ver figura 7).

Adicionalmente gracias a la memoria EEPROM del microcontrolador se logra guardar el nombre de la última señal y el último valor ingresado de la frecuencia ingresada cuando el circuito pierde la energía.

RESULTADOS PRACTICOS

A continuación se muestran los resultados obtenidos en la práctica:

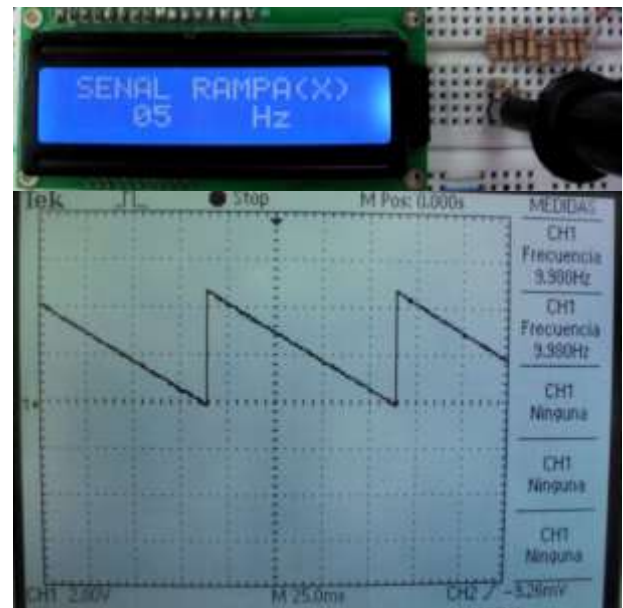


Fig. 6. Resultados obtenidos por el osciloscopio para la señal rampa

Para la señal rampa, el microcontrolador simplemente toma valor a valor de la tabla guardada en la memoria Flash. Luego la envía al puerto C del mismo conectado a su vez al R2R.

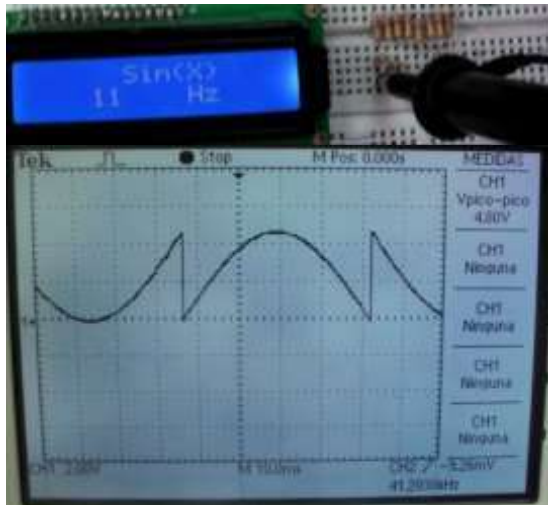


Fig.7. Resultados obtenidos por el osciloscopio para la señal Seno

En la figura anterior se ve la salida de la señal seno, en esta grafica se observa que existe un problema, esto es debido a que se generó un error en el momento de implementar la conversión digital – Analógica, ya que no toma el valor exacto de la señal.

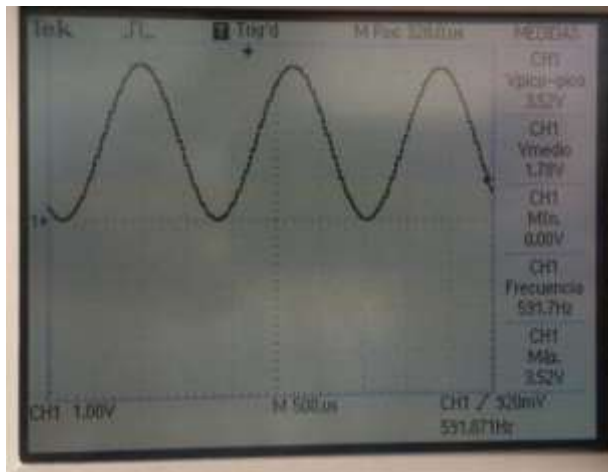


Fig.8. Resultados obtenidos por el osciloscopio para la señal Seno (Solución implementación R2R)

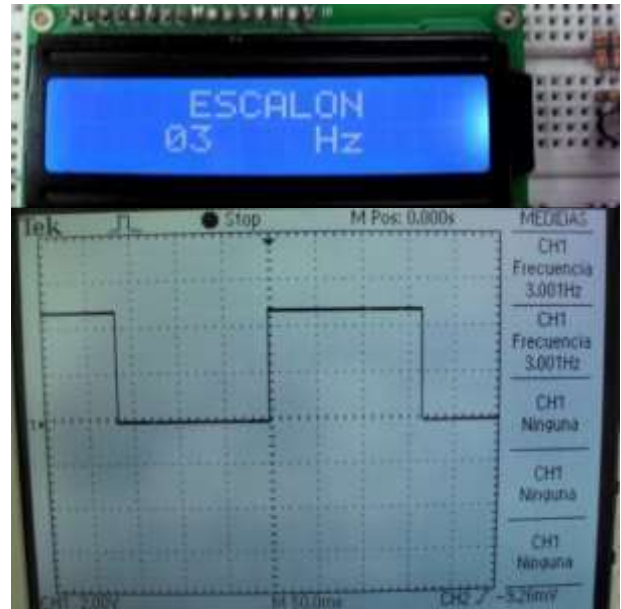


Fig.9. Resultados obtenidos por el osciloscopio para la señal cuadrada.

En la gráfica anterior se observa que no se ve afectada la señal cuadrada por la mal implementación del convertor R2R.

CONCLUSIONES

- Alimentar el circuito de una forma adecuada evitara que los dispositivos como el microcontrolador, LCD y diferentes elementos utilizados en el circuito, sufran daños irreparables, también puede ocurrir que se dañen algunas partes del microcontrolador, realizando un análisis de los componentes internos del dispositivo como la memoria EEPROM, puede suceder que internamente hayan sectores que se dañen como por ejemplo pines de los puertos ocasionando problemas de funcionamiento en real.
- El diseño de los diagramas de flujo que se necesita para cada aplicación, facilita a llevar un orden en la codificación, evitando cometer errores cuando se está implementando un lenguaje de programación y por ende en el funcionamiento.
- Se ve necesario el entendimiento del set de instrucciones, estructura interna y especificaciones del microcontrolador, debido a que esto llevara a un correcto entendimiento y funcionamiento de la aplicación a realizar.
- Los cálculos de retardo se realizan para un tiempo exacto, sin embargo, ingresar un valor que no es entero por software es algo complejo,

esto genera un porcentaje de error en el retardo y por ende en físico se verá el efecto que este error produce,

MPLINKTM Object Linker , MPLIBTM Object Librarian User ' s Guide. (2005).

REFERENCIAS

Kosower, D. a., & Lopez-Villarejo, J. J. (2015). Flowgen: Flowchart-based documentation for C<math altimg="sil.gif" display="inline" overflow="scroll" xmlns:xocs="http://www.elsevier.com/xml/xocs/td" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" x. *Computer Physics Communications*. <http://doi.org/10.1016/j.cpc.2015.05.029>