Imagen que contiene Icono

Descripción generada automáticamenteDibujo en blanco y negro

Descripción generada automáticamente con confianza bajaInstituto Politécnico Nacional

Escuela Superior De Computo

Instrumentación y Control

Práctica #6

Uso Del Microcontrolador “PIC16F628A”

Nombre de los Integrantes:

* García Quiroz Gustavo Ivan
* Ortiz González Alan
* Romero Hernández Oscar David

Grupo: 5CV1

Nombre Del Profesor: Cervantes De Anda Ismael

Fecha de elaboración: 29 / NOV / 2023

Fecha de entrega: 06 / DIC / 2023

INDICE

[OBJETIVO: 3](#_Toc152712776)

[MATERIAL EMPLEADO: 3](#_Toc152712777)

[INVESTIGACIÓN TEÓRICA 4](#_Toc152712778)

[*¿Que es un microcontrolador?* 4](#_Toc152712779)

[Características (PIC16F628A) 5](#_Toc152712780)

[Programación del PIC16F627A 6](#_Toc152712781)

[Requisitos de Hardware 7](#_Toc152712782)

[Modo de Programación 7](#_Toc152712783)

[DESARROLLO DE LA PRÁCTICA 8](#_Toc152712784)

[DIAGRAMA DEL CIRCUITO (EN BASE AL CODIGO) 16](#_Toc152712785)

[CIRCUITO ARMADO 17](#_Toc152712786)

[CIRCUITO SIMULADO 17](#_Toc152712787)

[CIRCUITO FUNCIONANDO 18](#_Toc152712788)

[CIRCUITO SIMULADO 18](#_Toc152712789)

[CONCLUSIONES 19](#_Toc152712790)

[Conclusión General 19](#_Toc152712791)

[Conclusión Especifica 19](#_Toc152712792)

[García Quiroz Gustavo Ivan 19](#_Toc152712793)

[Ortiz Gonzalez Alan 20](#_Toc152712794)

[Romero Hernández Oscar David 20](#_Toc152712795)

[BIBLIOGRAFÍA 21](#_Toc152712796)

# OBJETIVO:

El objetivo de estipulado para la práctica es la comprensión del funcionamiento y el uso que nos provee el microcontrolador *PIC16F628A*, a su vez que debemos comprender la forma de programar este microcontrolador mediante el software brindado por el docente en las herramientas necesarias que permiten compilar programas de lenguaje c para futura comprensión en el nivel más bajo de la programación, el lenguaje ensamblador.

# MATERIAL EMPLEADO:

* 1 Push botton
* 1 Protoboard
* 1 Fuente de VCD variable
* 1 led
* 1 Resistencia 330 Ohm
* 1 Resistencia 10k Ohm
* Alambres de conexión
* 1 PIC16F628A
* 1 PC
* Software MPLAB X IDE
* Software compilador XC8

# INVESTIGACIÓN TEÓRICA

***Antecedentes***

Durante la década de 1990, los compiladores C recién comenzaban a ser populares, pero la programación en el ensamblaje aún era necesaria en caso de querer realizar un trabajo en programación incorporada.

Por esto mismo, los microcontroladores de 8 bits han evolucionado a lo largo del tiempo, y su desarrollo ha sido parte integral de la historia de la informática embebida. Para el desarrollo de esta práctica hicimos uso del microcontrolador *PIC16F628A*.

## *¿Qué es un microcontrolador?*

Un microcontrolador es como un ordenador en pequeño: dispone de una memoria donde se guardan los programas, una memoria para almacenar datos, dispone de puertos de entrada y salida, etc. A menudo se incluyen puertos seriales (RS-232), conversores analógicos/digital, generadores de pulsos PWM para el control de motores, bus I2C, y muchas cosas más. Por supuesto, no tienen ni teclado ni monitor, aunque podemos ver el estado de teclas individuales o utilizar pantallas LCD o LED para mostrar información.

El *PIC16f628a* es un microcontrolador de 8 bit, posee una arquitectura RISC avanzada, así como un juego reducido de 35 instrucciones. Este microcontrolador es el remplazo del *pic16f84a*, los pines del *pic16f628a* son compatibles con el pic16f84a, así se podrían actualizar proyectos que hemos utilizado con el pic16f84a.

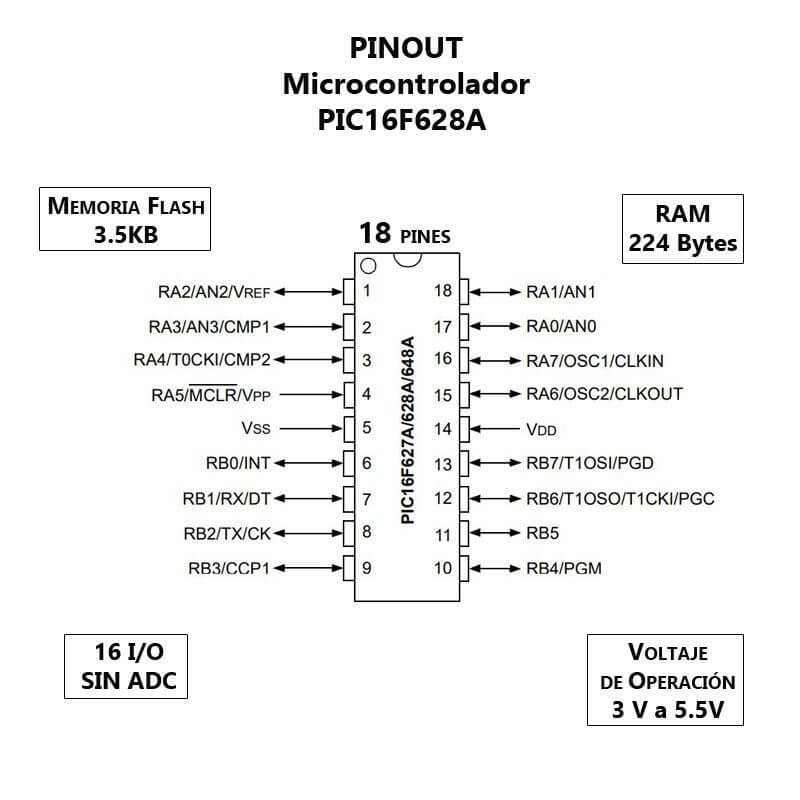
 

Imagen pinout

El ***PIC 16F628*** incorpora *tres características* importantes que son:

* Procesador tipo RISC (Procesador con un Conjunto Reducido de Instrucciones)
* Procesador segmentado
* Arquitectura HARVARD

Con estos recursos el PIC es capaz de ejecutar instrucciones solamente en un ciclo de instrucción. Con la estructura segmentada se pueden realizar simultáneamente las dos fases en que se descompone cada instrucción, ejecución de la instrucción y búsqueda de la siguiente.

La separación de los dos tipos de memoria son los pilares de la arquitectura Harvard, esto permite acceder en forma simultánea e independiente a la memoria de datos y a la de instrucciones. El tener memorias separadas permite que cada una tenga el ancho y tamaño más adecuado. Así en el PIC 16F628 el ancho de los datos es de un byte, mientras que la de las instrucciones es de 14 bits.

## Características (PIC16F628A)

* Conjunto reducido de instrucciones (RISC). Sólamente 35 instrucciones que aprender a utilizar
* Oscilador interno de 4MHz
* Las instrucciones se ejecutan en un sólo ciclo de máquina excepto los saltos (*goto* y *call*), que requieren 2 ciclos. Aquí hay que especificar que un ciclo de máquina se lleva 4 ciclos de reloj, si se utiliza el reloj interno de 4MHz, los ciclos de máquina se realizarán con una frecuencia de 1MHz, es decir que cada instrucción se ejecutará en 1uS (microsegundo)
* Opera con una frecuencia de reloj de hasta 20 MHz (ciclo de máquina de 200 ns)
* Memoria de programa: 2048 locaciones de 14 bits
* Memoria de datos: Memoria RAM de 224 bytes (8 bits por registro)
* Memoria EEPROM: 128 bytes (8 bits por registro)
* Stack de 8 niveles
* 16 Terminales de I/O que soportan corrientes de hasta 25 mA
* 3 Temporizadores
* Módulos de comunicación serie, comparadores, PWM

Otra característica de los PICs es el manejo de los bancos de registros. En línea general, los registros se clasifican como de uso general (GPR) y de uso específico o de funciones especiales (SFR).

* Los registros de uso general pueden ser usados directamente por el usuario, sin existir restricciones. Pueden servir para almacenar resultados que se reciben desde el registro W (acumulador), datos que provienen de las puertas de entradas, etc.
* Los registros de uso específicos no pueden ser usados directamente por el usuario. Estos registros controlan prácticamente todo el funcionamiento del microcontrolador, pues toda la configuración necesaria para funcionamiento del microcontrolador es hecho a través de algún tipo de SFR.

## Programación del PIC16F627A

El PIC16F627A/628A/648A se programa utilizando un método serial. El modo serial permite la programación del PIC16F627A/628A/648A mientras está en el sistema del usuario. Esto proporciona una mayor flexibilidad de diseño. Esta especificación de programación se aplica a los dispositivos PIC16F627A/628A/648A en todos los encapsulados.

## Requisitos de Hardware

El PIC16F627A/628A/648A requiere una fuente de alimentación programable para VDD (2.0V a 5.5V) y un VPP de 12V a 14V, o un VPP de 4.5V a 5.5V al utilizar voltaje bajo. Ambas fuentes deben tener una resolución mínima de 0.25V.

## Modo de Programación

El modo de programación para el PIC16F627A/628A/648A permite la programación de la memoria de programa del usuario, la memoria de datos, ubicaciones especiales utilizadas para la identificación y la Palabra de Configuración.

Diagrama

Descripción generada automáticamente

Imagen PIN DIAGRAM

# DESARROLLO DE LA PRÁCTICA

Se debe realizar el siguiente circuito usando el microcontrolador PIB, para hacer prender un led con un push botón.

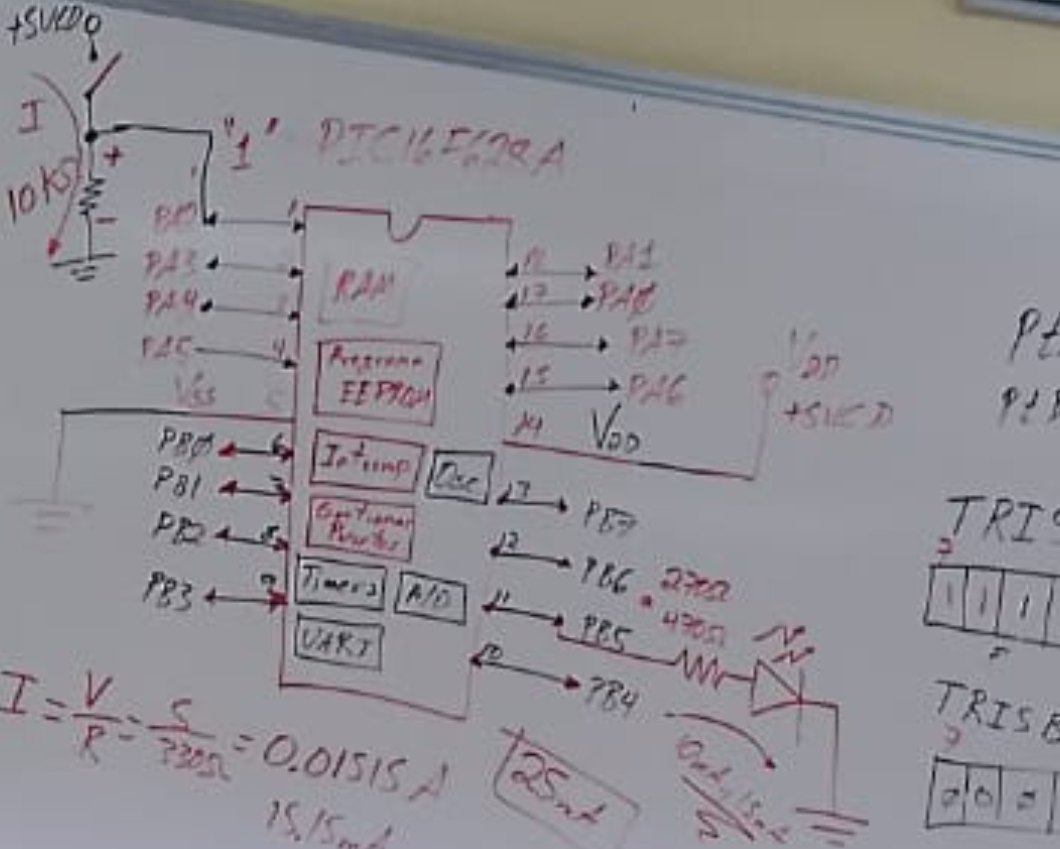


Imagen Circuito usando el PIC16F628A

Para esto se debe seguir los siguientes pasos.

1. Instalar nuestro entorno de desarrollo MPLAB X IDE y su compilador XC8 de 8bits.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Imagen MPLAB X IDE

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Imagen XC8

2. Instalar los controladores y el software para poder usar el programador.

Texto

Descripción generada automáticamente

Imagen Controladores y software programador

3. Una vez todo instalado se procede a crear un nuevo proyecto en MPLAB.

a) Seleccionar un nuevo proyecto vacio.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Imagen Nuevo proyecto

b) Seleccionar el micro a usar, en este caso “PIC16F628A”

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Imagen PIC16F628A

c) Se selecciona el compilador instalado previamente:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Imagen XC8

d) Se selecciona la ruta para guardar el proyecto junto al nombre de este y se da en finalizar

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Imagen Guardar proyecto

f) Ahora en el proyecto se da click derecho y se debe crear un nuevo archivo main en el lenguaje C.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Imagen Archivo main

g) Con el código proporcionado se compila.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Imagen Código compilado

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Imagen Output

h) Se genera el archivo .hex que será usado para meterlo al programador y programar el microcontrolador

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Imagen Archivo .hex

4. Se conecta el programador a la PC y se carga el .hex y se programa.



Imagen Programador usado

Imagen de la pantalla de un computador

Descripción generada automáticamente con confianza media

Imagen Programador

5. Se conecta el circuito en base a las conexiones del código, para eso se debe entender que hace el código:

Estas son configuraciones específicas del microcontrolador PIC16F628A. Establecen parámetros como el tipo de oscilador, si se debe habilitar o deshabilitar ciertas características de seguridad como el Watchdog Timer, y otros aspectos relacionados con la configuración del hardware.

Texto

Descripción generada automáticamente

Imagen Configuraciones específicas del microcontrolador PIC16F628A

Aquí se incluyen archivos de cabecera necesarios y se definen constantes para la frecuencia de oscilación (\_XTAL\_FREQ) y pines de entrada y salida.

Pantalla de computadora con letras

Descripción generada automáticamente con confianza baja

Imagen Pines de entrada y salida

Se define una función de interrupción vacía (VectorInterrupcion). En este caso, parece estar preparada para manejar la interrupción relacionada con la escritura en la EEPROM, aunque el cuerpo de la función está vacío.

Texto

Descripción generada automáticamente

Imagen Interrupción

La función ConfigPIC configura los puertos A y B, así como algunas configuraciones adicionales relacionadas con el puerto A, como la configuración de terminales digitales.

Texto

Descripción generada automáticamente

Imagen Puertos A y B

La función principal realiza un bucle infinito (while (1)) donde verifica el estado de Ent3 (un pin de entrada en el puerto A). Si Ent3 está en estado bajo (0), entonces Sal2 (un pin de salida en el puerto B) se pone en 0; de lo contrario, se pone en 1. En resumen, este código parece controlar Sal2 basándose en el estado de Ent3. Sin embargo, el bucle es infinito y no hay ninguna instrucción que pueda cambiar el estado de Ent3 dentro del bucle, por lo que esta acción se ejecutará continuamente sin cambios aparentes.

Pantalla de un video juego

Descripción generada automáticamente con confianza media

Imagen Función principal

Considerando las entradas del PIC y la configuración con el código tenemos:

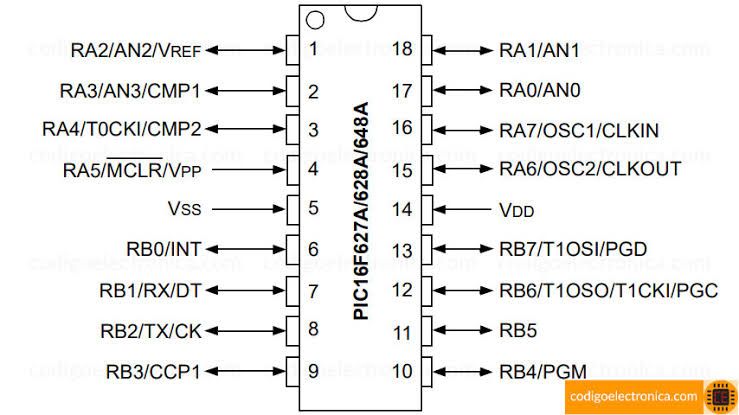


Imagen Pin out

## DIAGRAMA DEL CIRCUITO (EN BASE AL CODIGO)

**Diagrama, Esquemático

Descripción generada automáticamente**

Imagen Diagrama del circuito

## CIRCUITO ARMADO

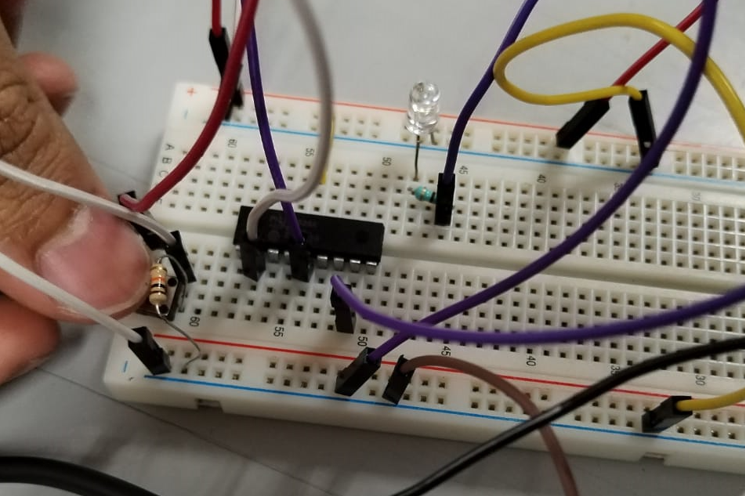
****

Imagen LED apagado

## CIRCUITO SIMULADO

**Diagrama, Esquemático

Descripción generada automáticamente**

Imagen Circuito simulado

## CIRCUITO FUNCIONANDO

**Imagen que contiene tabla, competencia de atletismo, computadora

Descripción generada automáticamente**

Imagen LED encendido

## CIRCUITO SIMULADO

**Diagrama, Esquemático

Descripción generada automáticamente**

Imagen Circuito simulado

# CONCLUSIONES

## Conclusión General

La práctica llevada a cabo con el microcontrolador PIC16F628A ha resultado ser un paso importante para entender el uso básico de este microcontrolador y con su programación. El objetivo inicial de familiarizarnos con el funcionamiento del microcontrolador y aprender a programarlo utilizando el software MPLAB X IDE y el compilador XC8 se ha alcanzado de manera satisfactoria.

La utilización de materiales básicos como un botón, un LED, resistencias y una protoboard ha permitido aplicar los conocimientos teóricos en un entorno práctico, brindando una experiencia tangible en el desarrollo de proyectos con microcontroladores. La conexión de estos elementos y la programación del PIC16F628A han proporcionado una visión integral del proceso, desde la concepción de la idea hasta la implementación práctica.

## Conclusión Especifica

## García Quiroz Gustavo Ivan

En conclusión, la práctica realizada con el microcontrolador PIC16F628A ha resultado fundamental para alcanzar una comprensión más profunda de su funcionamiento y programación a nivel básico. A través de la manipulación de componentes simples como un Push Button, un LED y resistencias, se logró establecer una conexión tangible entre la teoría y la práctica, consolidando así los conocimientos adquiridos en el aula.

El material empleado, desde la protoboard hasta el PIC16F628A, se reveló como idóneo para el propósito de la práctica, facilitando la experimentación y la conexión directa con los conceptos teóricos previamente abordados. La combinación de elementos como el botón, el LED y las resistencias no solo permitió visualizar el resultado práctico de los programas, sino que también fomentó el entendimiento de la importancia de cada componente en el diseño y ejecución de sistemas electrónicos.

## Ortiz Gonzalez Alan

El desarrollo de esta práctica, llevada desde lo teórico en las sesiones previas a la práctica, nos ha brindado la oportunidad de comprender y saber la forma en la que podemos transformas lo que nosotros, como desarrolladores de código, podemos plasmar los trabajos que lleguemos a realizar en el curso y posteriores trabajos a lo largo de la carrera.

A su vez, es impresionante la forma en la que podemos plasmar la lógica en un contexto físico, y es impresionante la forma en la que se puede convertir de un lenguaje lógico de alto nivel a uno de bajo nivel.

## Romero Hernández Oscar David

La práctica permitió una comprensión más profunda tanto del funcionamiento del PIC como de la forma de programar. La familiarización con el microcontrolador y la programación a través de MPLAB X IDE y el compilador XC8, se logró de manera satisfactoria.

La utilización de componentes simples, como botones, LEDs y resistencias, en la protoboard facilitó la aplicación de los conocimientos teóricos en un entorno práctico, brindando una valiosa experiencia en el desarrollo de proyectos con microcontroladores. En conjunto, esta práctica ha fortalecido los fundamentos necesarios para el futuro proyecto y ha proporcionado una visión integral del proceso, desde la conceptualización hasta la implementación práctica de sistemas electrónicos.

# BIBLIOGRAFÍA

[1] *Microcontrolador 8 bits - Todos los fabricantes industriales*. (s.f.). DirectIndustry - El marketplace B2B de la industria: sensores, automatismos, motores, bombas, manipulación, embalajes, etc. <https://www.directindustry.es/fabricante-industrial/microcontrolador-8-bits-210317.html>

[2] *Microcontrolador pic16f628a*. (s.f.). MV Electronica. <https://mvelectronica.com/producto/microcontrolador-pic16f628a#:~:text=El%20pic16f628a%20es%20un%20microcontrolador,hemos%20utilizado%20con%20el%20pic16f84a.>

[3] *PIC16F627A/628A/648A EEPROM Memory Programming Specification*. (s.f.). Microchip. <http://ww1.microchip.com/downloads/en/devicedoc/41196g.pdf>

[4] *MCU de 8 bits vs. de 32 bits: cómo elegir el microcontrolador adecuado para el diseño de tu PCB*. (s.f.). Altium. <https://resources.altium.com/es/p/8-bit-vs-32-bit-mcu-choosing-right-microcontroller-your-pcb-design>