

GUIA DE LABORATORIO No. 1
USO INICIAL DEL LENGUAJE ENSAMBLADOR



OBJETIVO GENERAL:

Explorar las características del lenguaje ensamblador en el aprendizaje de la tecnología de microcontroladores.

OBJETIVOS ESPECIFICOS:

- Explorar el manejo de la interfaz de programación del microcontrolador.
- Reconocer el conjunto de instrucciones útiles para el microcontrolador.
- Realizar un primer acercamiento a la estructura del microcontrolador.

DEFINICIÓN DEL PROBLEMA:

Cuando se hace un acercamiento inicial a una tecnología programable se generan tropiezos principalmente por el proceso de conocer un nuevo lenguaje de programación, y mucho más si no se conoce la sintaxis de este lenguaje y las ventajas/desventajas de su uso. Como primer problema se plantea el desarrollo de programas en ensamblador que realicen operaciones aritméticas y lógicas a partir de variables y finalmente evidenciar el proceso para llegar al resultado en el entorno de simulación:

- Usando el conjunto de instrucciones dado en la Tabla 1, se realizarán varias operaciones aritméticas y lógicas basadas en el siguiente ejemplo: si se requiere trabajar con constantes en ensamblador se debe usar las instrucciones que contienen la letra “l” que indica literal o constante para el vocabulario de los PIC, a partir de esto surge la primera idea de cargar un valor constante a una variable, tal cual como se representaría en pseudocódigo (variable←constante), esto se hace con dos instrucciones, la primera tiene el fin de crear la constante en el interior del sistema, y la segunda busca llevar esta constante finalmente a la variable deseada:

movlw .5 ;Crear la constante 5 en el sistema

movwf aux1 ;Llevar la constante creada a la variable aux1

- A partir del ejemplo previo, realice un programa que cargue el valor de 9 a una variable creada previamente de nombre “var1”, luego compruebe experimentalmente cual es el valor más grande que puede cargar a esta variable.
- Para realizar operaciones aritméticas o lógicas, se debe partir de que las variables a procesar ya se encuentren previamente cargadas. Se debe tener en cuenta cuantos valores de entrada requiere una operación en específico, por ejemplo, una suma requiere dos (los sumandos) mientras que un complemento a 1 solo requiere un valor. Si por ejemplo se desea realizar la suma de una variable con un valor constante, primero se debe llevar el valor de la variable al sistema y luego sumarle la constante para finalmente llevar el valor del resultado a otra variable:

movf var1,w ;Llevar el valor de la variable var1 al sistema

addlw .5 ;Sumar la constante 5 al valor de la variable var1
movwf var2 ;Llevar el resultado de la suma del sistema hasta la variable var2

- Finalmente se debe modificar el ejemplo previo para realizar las siguientes operaciones:
 Sumar 3 a la variable aux1 cargada previamente con el valor de 7.
 Sumar las variables aux1 y aux2 previamente cargadas con los valores de 8 y 10 respectivamente.
 Restar a la constante 9 la variable aux1 cargada previamente con el valor de 5.
 Restar la variable aux1 a la variable aux2 previamente cargadas con los valores de 6 y 4 respectivamente.
 Multiplicar a la constante 4 la variable aux1 cargada previamente con el valor de 5.
 Multiplicar la variable aux1 a la variable aux2 previamente cargadas con los valores de 12 y 15 respectivamente.
 Realizar el complemento a 1 de la variable aux1 cargada previamente con el valor de 12.
 Realizar el complemento a 2 de la variable aux1 cargada previamente con el valor de 12.
 Realizar la O lógica "bitwise" entre el valor de 7 y la variable aux1 cargada previamente con el valor de 35.
 Realizar la O lógica "bitwise" entre las variables aux1 y aux2 cargadas previamente con los valores de 20 y 56 respectivamente.
 Realizar la Y lógica "bitwise" entre el valor de 15 y la variable aux1 cargada previamente con el valor de 62.
 Realizar la Y lógica "bitwise" entre las variables aux1 y aux2 cargadas previamente con los valores de 100 y 45 respectivamente.
 Realizar la O lógica exclusiva "bitwise" entre el valor de 1 y la variable aux1 cargada previamente con el valor de 120.
 Realizar la O lógica exclusiva "bitwise" entre las variables aux1 y aux2 cargadas previamente con los valores de 17 y 90 respectivamente.
 Seguido se debe realizar la siguiente operación, con aux1, aux2 y aux3 previamente cargadas con 25, 40 y 103 respectivamente:

$$aux4 \leftarrow (aux1 \text{ OR } aux2) \text{ AND } (aux3 \text{ XOR } 0xD0)$$

Finalmente se debe realizar la siguiente operación, con aux1, aux2 y aux3 previamente cargadas con 18, 60 y 16 respectivamente:

$$aux4 \leftarrow (aux1 + aux2) - 3 * (aux3 - 0b11010)$$

movlw
movwf
movf
addwf
addlw
sublw
subwf
mulwf
mullw
andwf
andlw
iorwf
iorlw
xorwf

xorlw
comf
negf
goto

Tabla 1 Instrucciones del ensamblador del PIC18

DISEÑO POR REALIZAR:

Para la solución de este problema, se debe implementar el algoritmo para un microcontrolador PIC18F4550 en lenguaje ensamblador, partiendo desde el diagrama de flujo y su posterior implementación en líneas de código en el entorno de desarrollo MPLabX para su simulación.

TIEMPO PARA EJECUCIÓN

Se contempla como tiempo adecuado para la realización de este laboratorio de 2 semanas, incluyendo la sustentación de este, correspondientes a las semanas 2 y 3 del calendario académico.

SUSTENTACIÓN

- Para la sustentación se deben presentar al docente los 16 ejercicios propuestos en esta guía desde el “Sumar 3 a la variable aux1 cargada previamente con el valor de 7”, para lo cual, deben estar en un único programa, uno detrás de otro y solo debe crear 4 variables para el desarrollo de todos los ejercicios aparte del registro W, las cuales se deben estar reusando para cada ejercicio. La explicación de los ejercicios será oral y usando el simulador de MplabX.