



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Departamento de Eléctrica y Electrónica

Carrera de Electrónica y Automatización

MICROPROCESADORES Y MICROCONTROLADORES

Práctica 1.2

**JUEGO DE INSTRUCCIONES DE LA FÁBRICA MICROCHIP
– GAMA MEDIA**

Autor:

Iza Tipanluisa Alex Paul

Docente:

Ing. Amparo Meythaler

NRC: 4891

1) OBJETIVOS

- Consolidar el manejo del MPLAB IDE.
- Realizar una aplicación que permita realizar un corrido en pasos del juego de instrucciones de la fábrica MICROCHIP de la gama media.

2) MARCO TEORICO

JUEGO DE INSTRUCCIONES DE LA MICROCHIP – GAMA MEDIA

Los PIC16XXX son microcontroladores RISC. Esto se refleja en que tienen un repertorio reducido de 35 instrucciones ortogonales (prácticamente todas las instrucciones pueden utilizar cualquier operando), éstas son rápidas y todas tienen una longitud fija de 14 bits.

Una forma de clasificar a las instrucciones por el operador, puede ser la siguiente:

- Instrucciones de carga.
- Instrucciones de bits.
- Instrucciones aritméticas.
- Instrucciones lógicas.
- Instrucciones de salto.
- Instrucciones de manejo de subrutinas.
- Instrucciones especiales.

Algunas de las instrucciones afectan a las banderas. Las banderas de la ALU son DC (acarreo auxiliar), C (acarreo final) y Z (cero).

3) EQUIPOS Y MATERIALES

- PC con el paquete MPLAB IDE de la MICROCHIP.

4) ACTIVIDADES

a) Digite el siguiente ejercicio.

LIST P = PIC16F877

ESTADO EQU 03 ESTADO=[03]

R1 EQU 25	R1=[25]
-----------	---------

R2 EQU 37 R2=[37]

ORG 0

MOVLW A5 W=A5 C=0 CD=0 Z=0

MOVWF R1	R1=A5	C=0 CD=0 Z=0
ANDLW 6C	W=24	C=0 CD=0 Z=0
BSF ESTADO,2	PONGA 1 EN BIT 2 LOCALIDAD 03; C=1 CD=0 Z=0	
SWAPF 25,1	[25]=5A	C=1 CD=0 Z=0
MOVWF 37	W=36	C=1 CD=0 Z=0
SUBWF R1,0	W=36	C=1 CD=1 Z=0
DECF R2,1	[R2]=23	C=1 CD=1 Z=0
BTFSS 03,1	¿BIT 1 LOCALIDAD 3 ES 1? SI, SALTA	
COMF 25,0	NO REALIZA	
IORLW E7	W=F7	C=1 CD=1 Z=0
ADDWF R2,0	W=1A	C=1 CD=0 Z=0
RLF R1,1	R1=B5	C=0 CD=0 Z=0
MOVF 37,0	W=23	C=0 CD=0 Z=0
GOTO UNO	SALTA A UNO	
CLRF 43	NO REALIZA	
XORWF R1,0	NO REALIZA	
BTFSC R2,6	NO REALIZA	
MOVF 43,1	NO REALIZA	
UNO: INCF 25,1	[25]=B6	C=0 CD=0 Z=0
RRF R2,0	W=11	C=1 CD=0 Z=0
CLRW	W=0	C=1 CD=0 Z=1
BCF R1,3	PONGA CERO EN BIT 3 LOCALIDAD R1; R1=B6	
END	FIN PROGRAMA	

b) Compile el programa.

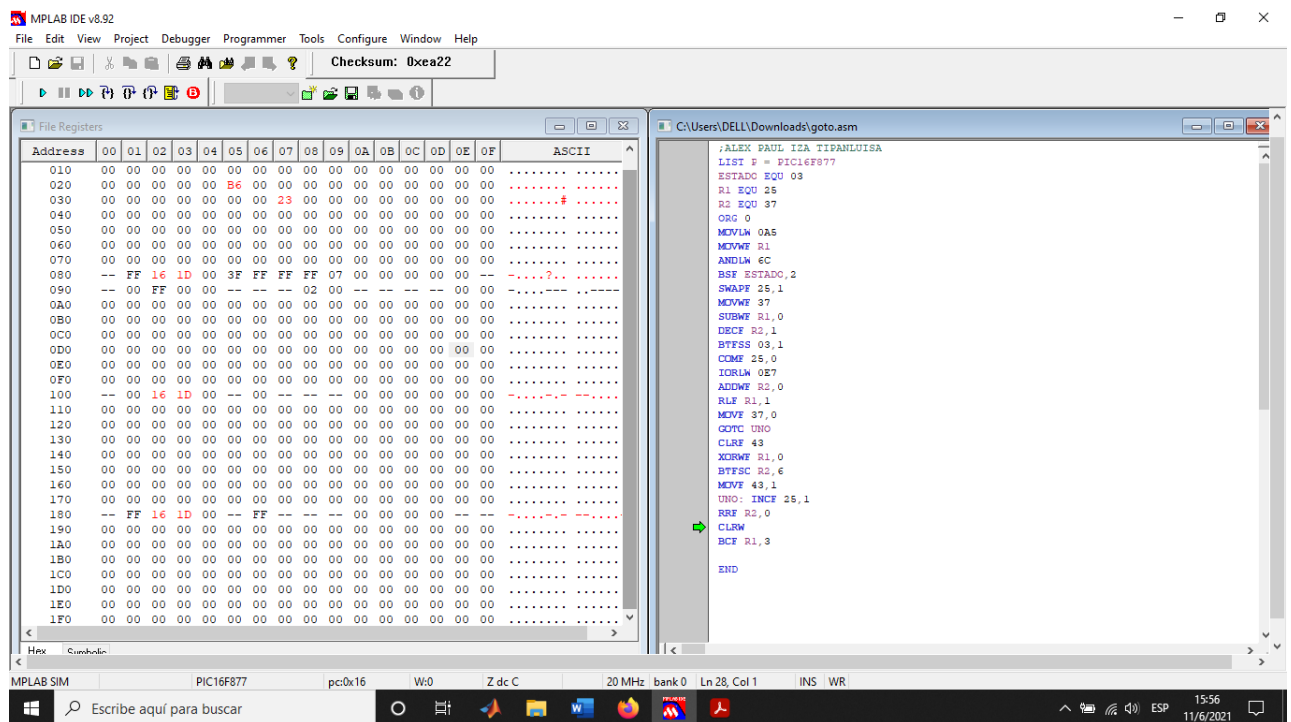
The image displays two screenshots of the MPLAB IDE v8.92 interface, showing the compilation process of a PIC16F877 program.

Top Screenshot:

- File Registers:** Shows the memory map from 010 to 1F0. The 010-01F range is filled with 00, while 020-02F contains 05 and 1F.
- Output Window:** Displays the compilation log. It shows warnings for opcodes found in columns 1, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 26, 27, 28, and 29. The log concludes with "BUILD SUCCEEDED" and "Release build of project 'C:\Users\DELL\Downloads\goto.disposable_mcp' succeeded."
- Assembler Window:** Shows the assembly code for "goto.asm". It includes a header for "ALEX PAUL IZA TIPANLUISA" and a list of instructions: `LIST P = PIC16F877`, `ESTADO EQU 03`, `R1 EQU 25`, `R2 EQU 37`, `ORG 0`, `MOVLW 0A5`, `MOVWF R1`, `ANDLW 6C`, `BSF ESTADO,2`, `SWAPF 25,1`, `MOVWF 37`, `SUBWF R1,0`, `DECF R2,1`, `BTFS 03,1`, `CMF 25,0`, `IORLW 0E7`, `ADDWF R2,0`, `RLF R1,1`, `MOVF 37,0`, `GOTC UNO`, `CLRF 43`, `XORWF R1,0`, `BTFS R2,6`, `MOVF 43,1`, `UNO: INCF 25,1`, `RRF R2,0`, `CLRW`, `BCF R1,3`, and `END`.
- StatusBar:** Shows the target device as PIC16F877, processor as pc0x1, and configuration as W:0x5 z d c c.

Bottom Screenshot:

- File Registers:** Similar to the top screenshot, showing the memory map.
- Assembler Window:** Displays the same assembly code as the top screenshot.
- StatusBar:** Shows the target device as PIC16F877, processor as pc0x5, and configuration as W:0x24 Z DC C.



5) RESULTADOS

- **¿En las instrucciones que manejan bits, explique qué significa la b y qué valor puede tener?**

La letra b significa la posición de un bit y los valores que puede tomar es del 0 hasta el 7

- **¿Cuál es la función del PC? Explique.**

La función de la PC es el de indicar a que instrucción va a realizar.

6) CONCLUSIONES

- Se ha concluido que los saltos con la instrucción BTFSC y BTFSS solo pueden realizar con datos guardados en la localidad de una memoria y puede dar solo un salto hacia adelante siempre y cuando cumpla la condición.
- Se ha concluido que la instrucción BCF y BSF nos ayudan a cambiar el bit de cero a uno o viceversa, esto dado una localidad y la posición del bit que se desea cambiar.
- Se ha concluido que se pueden dar un nombre a las localidades haciendo más fácil la interacción y la programación

7) RECOMENDACIONES

- Se recomienda revisar el lugar en el que se asignarán los valores después de realizar cada una de las instrucciones, si se asignan en la localidad de memoria o en el registro W
- Se recomienda estudiar correctamente las instrucciones y las banderas que se pueden alterar después de realizar la instrucción; tener en cuenta que en algunas instrucciones las banderas no se alteran, estas se conservan con el mismo estado lógico.
- Se recomienda revisar constantemente los cálculos del corrido de maquina ya que un dato incorrecto puede hacer la diferencia; así también revisar las banderas ya que algunas instrucciones dependen de ellas dando como resultado datos erróneos a la del compilado en el programa.

8) BIBLIOGRAFIA

Meytaler, A. (2021). Sistemas Basados en MCU. Latacunga, UFA ESPE