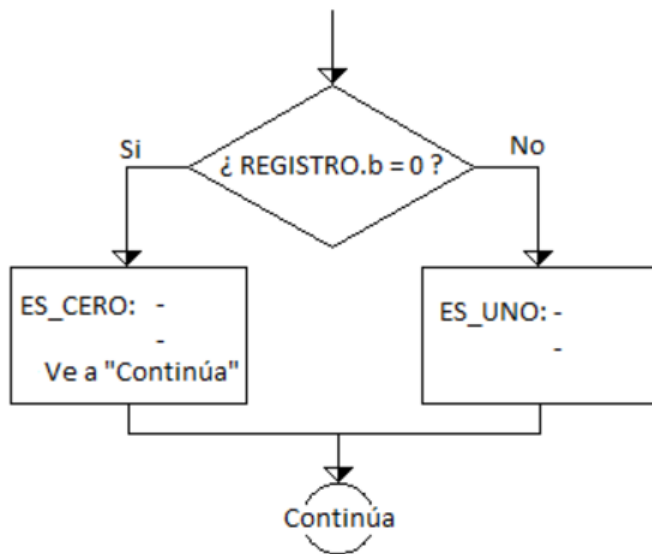


COMPARACIONES

1. COMPARACIÓN DE UN BIT DE UN REGISTRO DE 8 BITS

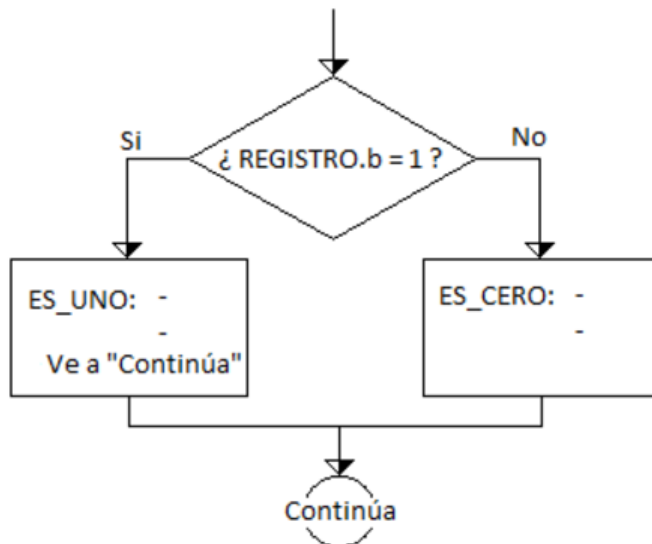
Sea "REGISTRO" cualquier localidad de memoria RAM, es decir puede ser SFR o GPR (registros de propósito específico o general), b el peso del bit, que, al ser una arquitectura de 8 bits, puede tomar los valores 0 a 7.

Probar si el bit **b** del registro de memoria RAM "REGISTRO" es 0



```
...  
BTFSC REGISTRO,b ;¿REGISTRO.b = 0?  
GOTO ES_UNO      ; No  
ES_CERO:         ; Si  
-  
-  
GOTO Continua  
ES_UNO:         -  
-  
-  
Continua:       ; Ejecuta las  
-               ; siguientes  
-               ; instrucciones  
...
```

Probar si el bit **b** del registro de memoria RAM "REGISTRO" es 1



```
...  
BTFSS REGISTRO,b ;¿REGISTRO.b = 1?  
GOTO ES_CERO      ; No  
ES_UNO:         ; Si  
-  
-  
GOTO Continua  
ES_CERO:         -  
-  
-  
Continua:       ; Ejecuta las  
-               ; siguientes  
-               ; instrucciones  
...
```

2. COMPARACIÓN DE UN REGISTRO CON UNA CONSTANTE

COMPARACIÓN: REGISTRO CONTIENE UN VALOR K



```

;PROBAR SI UN REGISTRO CONTIENE UN VALOR ESPECIFICO
PROCESSOR 16F877A
#include <P16F877A.INC>

REG EQU 20H

ORG 0          ;VECTOR DE INICIO
GOTO INICIO    ;SALTA A INICIO
ORG 5

INICIO:
  MOVF REG,W    ;W<-(REG)
  SUBLW 0X05    ;W<-(REG)-K
  BTFSS STATUS,Z ;¿REG=K?, ¿Z=1?
  GOTO NO_IGUAL ; NO
IGUALES:
  ; SI
  MOVLW 1       ;W<-1
  GOTO INICIO   ;REGRESA A INICIO
NO_IGUAL:
  CLRW          ;W<-0
  GOTO INICIO   ;REGRESA A INICIO

END
  
```

3. COMPARACIÓN ENTRE REGISTROS

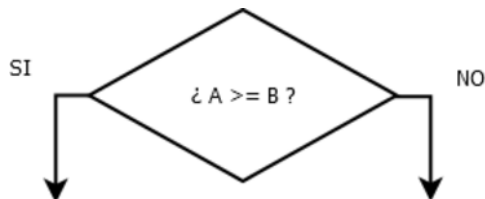
Sea:

(REGISTRO_A) = A

(REGISTRO_B) = B

(A - B)	Resultado	C	Z
A > B	Positivo	1	0
A = B	Cero	1	1
A < B	Negativo	0	0

Ejemplo:



```

MOVWF REGISTRO_B,W ;
SUBWF REGISTRO_A,W ; (REGISTRO_A) - (REGISTRO_B) --> W
BTFSS STATUS,C     ; ¿A >= B ? , ¿C = 1 ?
GOTO MENOR         ; NO, VE A ETIQUETA MENOR
MAYOR_O_IGUAL      ; SI
.
.
.
MENOR
.
.
.
  
```

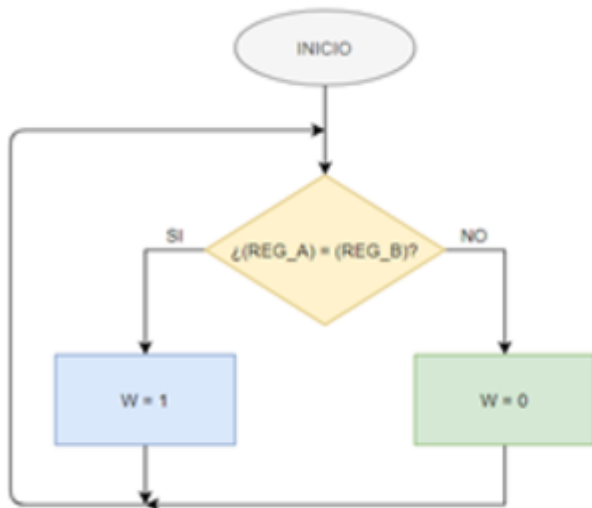


...

```

MOV F REGISTRO_B, W ; W ← (REGISTRO_B)
SUBWF REGISTRO_A, W ; (REGISTRO_A) - (REGISTRO_B) → W
BTFS STATUS, C ; ¿A >= B? , ¿C = 1?
GOTO MENOR_O_IGUAL ; NO, VE A ETIQUETA MENOR_O_IGUAL
MAYOR_O_IGUAL BTFS STATUS, Z ; SI, AHORA PRUEBA ¿A > B? , ¿Z = 0?
GOTO MENOR_O_IGUAL ; NO, VE A MENOR_O_IGUAL
MAYOR ; SI, A > B
;
;
MENOR_O_IGUAL
;
;
  
```

COMPARACIÓN: REGISTRO_A ES IGUAL QUE REGISTRO_B



```

;PROBAR SI DOS REGISTROS CONTIENEN EL MISMO VALOR
PROCESSOR 16F877A
#INCLUDE <P16F877A.INC>

REG_A EQU 20H
REG_B EQU 21H

ORG 0 ;VECTOR DE INICIO
GOTO INICIO ;SALTA A INICIO
ORG 5

INICIO:
MOV F REG_B, W ;W←(REG_B)
SUBWF REG_A, W ;W←(REG_A)-(REG_B)
BTFS STATUS, Z ;¿REG_A=REG_B?, ¿Z=1?
GOTO NO_IGUALES ; NO
IGUALES: ; SI
MOVLW 1 ;W←1
GOTO INICIO ;REGRESA A INICIO
NO_IGUALES
CLRW ;W←0
GOTO INICIO ;REGRESA A INICIO
END
  
```

COMPARACIÓN: REGISTRO_A ES MAYOR O IGUAL QUE REGISTRO_B



```

;PROBAR SI A ES MAYOR O IGUAL A B
PROCESSOR 16F877A
#include <P16F877A.INC>

REG_A EQU 20H
REG_B EQU 21H

ORG 0          ;VECTOR DE INICIO
GOTO INICIO    ;SALTA A INICIO
ORG 5

INICIO:
  MOVF REG_B,W  ;W<-(REG_B)
  SUBWF REG_A,W ;W<-(REG_A)-(REG_B)

  BTFSS STATUS,C ;¿REG_A>=REG_B?, ¿C=1?
  GOTO A_NO_MAY_IG; NO
A_MAY_IG:      ; SI
  MOVLW 1      ;W<-1
  GOTO INICIO   ;REGRESA A INICIO
A_NO_MAY_IG:
  CLRW         ;W<-0
  GOTO INICIO   ;REGRESA A INICIO
END
  
```

COMPARACIÓN: REGISTRO_A ES MAYOR QUE REGISTRO_B



```

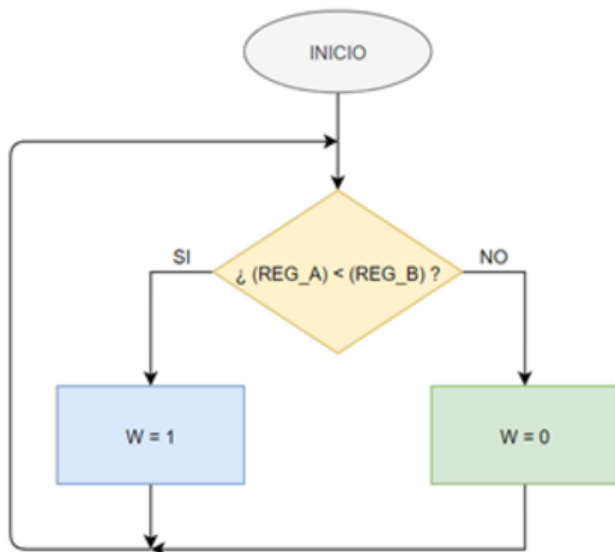
;PROBAR SI A ES MAYOR QUE B
PROCESSOR 16F877A
#include <P16F877A.INC>

REG_A EQU 20H
REG_B EQU 21H

ORG 0          ;VECTOR DE INICIO
GOTO INICIO    ;SALTA A INICIO
ORG 5

INICIO:
  MOVF REG_B,W  ;W<-(REG_B)
  SUBWF REG_A,W ;W<-(REG_A)-(REG_B)
  BTFSS STATUS,C ;¿REG_A>=REG_B?, ¿C=1?
  GOTO A_NO_MAY_IG; NO
A_MAY_IG:      ; SI
  BTFSC STATUS,Z ;¿REG_A>REG_B?, ¿Z=0?
  GOTO A_NO_MAYOR; NO
A_MAYOR:       ; SI
  MOVLW 1      ;W<-1
  GOTO INICIO   ;REGRESA A INICIO
A_NO_MAY_IG:
  CLRW         ;W<-0
  GOTO INICIO   ;REGRESA A INICIO
A_NO_MAYOR:
  CLRW         ;W<-0
  GOTO INICIO   ;REGRESA A INICIO
END
  
```

COMPARACIÓN: REGISTRO_A ES MENOR QUE REGISTRO_B



;PROBAR QUE EL CONTENIDO DEL REGISTRO A
;ES MENOR AL CONTENIDO DEL REGISTRO B

PROCESSOR 16F877A
#INCLUDE <P16F877A.INC>

REG_A EQU 20H
REG_B EQU 21H

```
ORG 0          ;VECTOR DE INICIO
GOTO INICIO    ;SALTA A INICIO
ORG 5

INICIO:
    MOVF REG_B,W    ;W<-(REG_B)
    SUBWF REG_A,W    ;W<-(REG_B)-(REG_A)
    BTFSC STATUS,C   ;¿REG_A<=REG_B?, ¿C=0?
    GOTO A_NO_MENOR  ; NO
A_MENOR:
    ; SI
    MOVLW 1          ;W<-1
    GOTO INICIO      ;REGRESA A INICIO
A_NO_MENOR:
    CLRW             ;W<-0
    GOTO INICIO      ;REGRESA A INICIO

END
```