

# REPERTORIO DE INSTRUCCIONES - THE WELS THEORY

Descripción: Todas las instrucciones usadas en el lenguaje Assembler. Encontrarás más información en: [www.facebook.com/WelsTheory/](http://www.facebook.com/WelsTheory/) y [www.thewelstheory.com](http://www.thewelstheory.com)

## INSTRUCCIONES DE CARGA

|           |  |
|-----------|--|
| CLRW      | El contenido del registro W se borra; W se carga con b'00000000' y el flanco Z se activa en Z = 1                                    |
| CLRF F    | El contenido del registro 'F' se borra; f se carga con b'00000000' y el flanco Z se activa Z = 1                                     |
| MOVLW K   | El registro W se carga con el valor de 'K'; K = 0xF3 -> W = 0xF3   |
| MOVF F, D | El contenido del registro 'F' se carga dependiendo el valor de 'D'; Si D=0 el destino es W, si D=1 el destino es el mismo registro F |
| MOVWF F   | El contenido del registro W se lleva al registro F   |

## INSTRUCCIONES DE BIT

|          |                                       |
|----------|---------------------------------------|
| BCF F, B | Pone a cero el bit 'B' del registro F |
| BSF F, B | Pone a uno el bit 'B' del registro F  |

## INSTRUCCIÓN GOTO K

|        |  |
|--------|--|
| GOTO K | Se usa la instrucción de salto incondicional, produciendo un salto a la dirección. Eje = GOTO Salto - el programa salta hasta donde dice Salto |
|--------|--|

## INSTRUCCIONES DE SUMA

|            |  |
|------------|--|
| ADDLW K    | Suma el contenido del registro W con la constante K y el resultado se almacena en W. Si hay acarreo el Flag C se pone a 1.     |
| ADDWF F, D | Suma el contenido del registro W con el registro F. Si D=0 el resultado se almacena en W, si D=1 el resultado se almacena en F |

## INSTRUCCIONES DE RESTA

|         |   |
|---------|---|
| SUBLW K | Resta el contenido de la constante K menos el contenido del registro W y almacena el resultado en W |
|---------|---|

**SUBWF F, D** Resta el contenido del registro F con el registro W. Si D=0 el resultado se almacena en W, si D=1 el resultado se almacena en F.

## **INCREMENTAR Y DECREMENTAR**

**DECF F, D** El registro F se decrementa en una unidad. Si D=0 se almacena en W, si D=1 se almacena en F. El Flag Z se activa si el resultado llega a cero

**INCF F, D** El contenido del registro se incrementa en una unidad. Si D=0 se almacena en W, si D=1 se almacena en F. El Flag Z se activa si el resultado llega a cero es decir si hay desbordamiento.

## **INSTRUCCIONES LOGICAS**

**ANDLW K** Efectúa la operación AND lógica entre el registro W y la constante K. Almacenando el resultado en W. Flag Z=1 si el resultado es igual a Cero

**ANDWF F, D** Efectúa la operación AND lógica entre el registro W y el registro F. Si D=0 almacena el resultado en W, si D=1 se almacena en F. Flag Z=1 si el resultado es igual a Cero

**COMF F, D** Realiza un complemento del contenido del registro F, invirtiendo los valores los unos por cero y viceversa. Si D=0 almacena en W, si D=1 almacena en F. Flag Z=1 si el resultado es cero

**IORLW K** Efectúa la operación OR lógica entre el registro W y la constante K. Almacena el resultado en W. Flag Z=1 si el resultado es cero

**IORWF F, D** Efectúa la operación OR lógica entre el registro W y el registro F. Si D=0 almacena el resultado en W, si D=1 se almacena en F. Flag Z=1 si el resultado es igual a Cero

**RLF F, D** Rotación de un bit a la izquierda del registro F pasando por el bit C. Si D=0 el resultado se almacena en W, si D=1 el resultado se almacena en F.

**RRF F, D** Rotación de un bit a la derecha del registro F pasando por el bit C. Si D=0 el resultado se almacena en W, si D=1 el resultado se almacena en F.

**SWAPF F, D** Los 4 bits de más peso del registro F se intercambian con los 4 bits de menos peso. Si D=0 se almacena en W, si D=1 se almacena en F.

**XORLW K** Realiza la operación XOR lógica entre el contenido del registro W y la constante K. El resultado se almacena en W. Flag Z=1 si el resultado es cero.

**XORWF F, D** Realiza la operación XOR lógica entre el contenido del registro W y el registro F. Si D=0 el resultado se almacena en W, si D=1 el resultado se almacena en F. Flag Z=1 si el resultado es cero.

## **INSTRUCCIÓN SLEEP**

**SLEEP** Esta instrucción hace que los microcontroladores pasen al estado de bajo consumo hasta que se produzca algún acontecimiento externo.