



MICROCONTROLADORES (II)

Temario

- ▶ Tipos de instrucciones
- ▶ Instrucciones orientadas a los registros o bytes
- ▶ Instrucciones orientadas a los bits
- ▶ Operaciones con literales y de control

Tipos de instrucciones

- ▶ El PIC tiene un juego de 35 instrucciones que son la base de su funcionamiento
- ▶ Las instrucciones se dividen en tres tipos. Esta división viene dada por el tipo de datos con los que trabajan:
 1. Instrucciones orientadas a los registros o bytes (byte-oriented operations)
 2. Instrucciones orientadas a los bits (bit-oriented operations)
 3. Operaciones con literales y de control (literal and control operations)

3

Microcontroladores (II)

EC2112

Tipos de instrucciones

- ▶ Las 35 instrucciones están resumidas en las siguientes tablas donde:
 - f** representa un registro cualquiera de la memoria de datos.
 - w** representa el registro de trabajo (Working Register).
 - b** representa la dirección de un bit dentro de un registro de 8 bits (0-7).
 - l** ó **k** representa un literal o constante de 8 bits.
 - d** representa el bit de destino, 0 ó 1.
 - x** representa los bits que estén representados por este tipo de dato no tienen ninguna función y su valor lo define el compilador.

4

Microcontroladores (II)

EC2112

Instrucciones orientadas a registros

MNEMÓNICO OPERANDOS	DESCRIPCIÓN	CÓDIGO OP	BANDERAS	NCIC	NOTAS	
ADDWF	f,d	w + f → d	00 0111 dfff ffff	C, DC, Z	1	1,2
ANDWF	f,d	w AND f → d	00 0101 dfff ffff	Z	1	1,2
CLRF	f	00 h → f	00 0001 1fff ffff	Z	1	2
CLRW	-	00 h → w	00 0001 0xxx xxxx	Z	1	-
COMF	f,d	Complemento de f → d	00 1001 dfff ffff	Z	1	1,2
DECf	f,d	f - 1 → d	00 0011 dfff ffff	Z	1	1,2
DECFSZ	f,d	f - 1 → d (si es 0 salta)	00 1011 dfff ffff	Ninguna	1(2)	1,2,3
INCf	f,d	f + 1 → d	00 1010 dfff ffff	Z	1	1,2
INCFSZ	f,d	f + 1 → d (si es 0 salta)	00 1111 dfff ffff	Ninguna	1(2)	1,2,3

5

Microcontroladores (II)

EC2112

Instrucciones orientadas a registros

MNEMÓNICO OPERANDOS	DESCRIPCIÓN	CÓDIGO OP	BANDERAS	NCIC	NOTAS	
IORWF	f,d	w OR f → d	00 0100 dfff ffff	Z	1	1,2
MOVF	f,d	f → d	00 1000 dfff ffff	Z	1	1,2
MOVWF	f	w → f	00 0000 1fff ffff	Ninguna	1	-
NOP	-	No operación	00 0000 0xx0 0000	Ninguna	1	-
RLF	f,d	Rota f izq por carry → d	00 1101 dfff ffff	C	1	1,2
RRF	f,d	Rota f dcha por carry → d	00 1100 dfff ffff	C	1	1,2
SUBWF	f,d	f - w → d	00 0010 dfff ffff	C,DC,Z	1	1,2
SWAPF	f,d	Intercambia nibbles de f → d	00 1110 dfff ffff	Ninguna	1	1,2
XORWF	f,d	w XOR f → d	00 0110 dfff ffff	Z	1	1,2

6

Microcontroladores (II)

EC2112

Instrucciones orientadas a bit

MNEMÓNICO OPERANDOS		DESCRIPCIÓN	CÓDIGO OP	BANDERAS	NCIC	NOTAS
BCF	f,b	Pone a 0 bit b de registro f	01 00bb bfff ffff	Ninguna	1	1,2
BSF	f,b	Pone a 1 bit b de registro f	01 01bb bfff ffff	Ninguna	1	1,2
BTFS	f,b	Salto si bit b de reg. f es 0	01 10bb bfff ffff	Ninguna	1(2)	3
BTFS	f,b	Salto si bit b de reg. f es 1	01 11bb bfff ffff	Ninguna	1(2)	3

Instrucciones con literales y de control

MNEMÓNICO OPERANDOS		DESCRIPCIÓN	CÓDIGO OP	BANDERAS	NCIC	NOTAS
ADDLW	k	$w + k \rightarrow w$	11 111x kkkk kkkk	C,DC,Z	1	-
ANDLW	k	$w \text{ AND } k \rightarrow w$	11 1001 kkkk kkkk	Z	1	-
CALL	k	Llamada a subrutina k	10 0kkk kkkk kkkk	Ninguna	2	-
CLRWDT	-	Borra temporizador del WDT	00 0000 0110 0100	TO,PD	1	-
GOTO	k	Ir a dirección k	10 1kkk kkkk kkkk	Ninguna	2	-
IORLW	k	$w \text{ OR } k \rightarrow w$	11 1000 kkkk kkkk	Z	1	-
MOVLW	k	$k \rightarrow w$	11 00xx kkkk kkkk	Ninguna	1	-

Instrucciones con literales y de control

MNEMÓNICO OPERANDOS		DESCRIPCIÓN	CÓDIGO OP	BANDERAS	NCIC	NOTAS
RETIE	-	Retorno de una interrupción	00 0000 0000 1001	Ninguna	2	-
RETLW	k	Retorno con k en w	11 01xx kkkk kkkk	Ninguna	2	-
RETURN	-	Retorno de una subrutina	00 0000 0000 1000	Ninguna	2	-
SLEEP	-	Modo Standby	00 0000 0110 0011	TO, PD	1	-
SUBLW	k	k - w → w	11 110x kkkk kkkk	C,DC,Z	1	-
XORLW	k	w XOR k → w	11 1010 kkkk kkkk	Z	1	-

Notas

1. Al modificar un registro de E/S con una operación sobre él mismo (por ejemplo MOVF PORTB,1), el valor utilizado es el que se halle presente en los terminales del PORTB. Por ejemplo, si el biestable tiene un "1" para un terminal configurado como entrada y toma el valor de nivel bajo desde el exterior, el dato se volverá a escribir como "0".
2. Si se ejecuta esta instrucción sobre el TMR0 y d=1, se borrará el conteo de la preescala asignada (preescaler), si está asignado al TMR0, pero no se borrará la preescala asignada en OPTION_REG, que controla Timer0.
3. Si se modifica el Contador de Programa **PC** o una condición de prueba es verdadera, la instrucción requiere dos ciclos máquina. El segundo ciclo se ejecuta como un NOP.

ADDLW

ADDLW	ADDLW			ADDLW
ADD Literal to w				
Operación	$w + k \rightarrow w$			
Sintaxis	[Etiqueta] ADDLW <i>k</i>			
Operadores	$0 < k < 255$			
Ciclos	1			
OPCODE	11	111x	kkkk	kkkk
Descripción	Suma el contenido del registro w al literal k , y almacena el resultado en w . Si se produce acarreo, la bandera C toma el valor de "1".			

11

Microcontroladores (II)

EC2112

ADDLW

Registro de **STATUS**

C toma el valor de 1 si se produce un acarreo desde el bit de mayor peso.

DC toma el valor de 1 si se genera un acarreo del bit 3 al bit 4.

Z toma el valor de 1 si el resultado de la operación es cero.

EJEMPLO: ADDLW 0x15

Si antes de la instrucción: $w = 10h = 0001\ 0000\ b$ y flag $Z = 0$

Al ejecutarse la instrucción:

$w = 10\ h + 15\ h = 25\ h$

$w = 0001\ 0000\ b + 0001\ 0101\ b = 0010\ 0101\ b$ y flag $Z = 0$

```

0001 0000 b
0001 0101 b
-----
0010 0101 b

```

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	X	X	X

12

Microcontroladores (II)

EC2112

ADDWF

ADDWF					ADDWF
	ADD w to f				
Operación	$w + f \rightarrow w / w + f \rightarrow f$				
Sintaxis	[Etiqueta] ADDWF f,d				
Operadores	$0 < f < 127$ $d [0,1]$				
Ciclos	1				
OPCODE	00	0111	dfff	ffff	
Descripción	Suma el contenido del registro w al contenido del registro f , y almacena el resultado en w si $d = 0$, y en el registro f si $d = 1$.				

13

Microcontroladores (II)

EC2112

ADDWF

Registro de **STATUS**

C toma el valor de 1 si se produce un acarreo desde el bit de mayor peso

DC toma el valor de 1 si se genera un acarreo del bit 3 al bit 4.

Z toma el valor de 1 si el resultado de la operación es cero

EJEMPLO: ADDWF FSR,0

Si antes de la instrucción: $w = 17 \text{ h}$ y $\text{FSR} = \text{C2 h}$ como $d=0$

Al ejecutarse:

$w = 17 \text{ h} + \text{C2 h} = \text{D9 h}$

$\text{FSR} = \text{C2 h}$

```

0001 0111 b
1100 0010 b
-----
1101 1001 b

```

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	X	X	X

14

Microcontroladores (II)

EC2112

ANDLW

ANDLW					ANDLW
AND Literal and w					
Operación	w AND k → w				
Sintaxis	[Etiqueta] ANDLW k				
Operadores	0 < f < 255				
Ciclos	1				
OPCODE	11	1001	kkkk	kkkk	
Descripción	Efectúa la operación AND lógico entre el contenido del registro w y el literal k , y almacena el resultado en w . Esta instrucción realiza la operación AND bit a bit.				

15

Microcontroladores (II)

EC2112

ANDLW

Registro de **STATUS**

Z toma el valor de 1 si el resultado de la operación es cero

EJEMPLO: ANDLW 0x5F

Si antes de la instrucción w = A3 h

Al ejecutarse:

w = 0101 1111 b AND 1010 0011 b = 0000 0011 b = 03 h

```

0101 1111 b
1010 0011 b
0000 0011 b

```

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	X	-	-

16

Microcontroladores (II)

EC2112

ANDWF

ANDWF					ANDWF
	AND w with f				
Operación	$w \text{ AND } f \rightarrow w / w \text{ AND } f \rightarrow f$				
Sintaxis	[Etiqueta] ANDWF f,d				
Operadores	$0 < f < 127 \quad d [0,1]$				
Ciclos	1				
OPCODE	00	0101	dfff	ffff	
Descripción	Efectúa la operación AND entre el contenido del registro w y el contenido del registro f , y almacena el resultado en w si d = 0 , y en f si d = 1 . Esta instrucción realiza la operación AND bit a bit.				

17

Microcontroladores (II)

EC2112

ANDWF

Registro de **STATUS**

Z toma el valor de 1 si el resultado de la operación es cero

EJEMPLO: ANDWF FSR,1

Si antes de la instrucción: $w = 17 \text{ h} = 0001 \ 0111 \text{ b}$ y

$\text{FSR} = \text{C2 h} = 1100 \ 0010 \text{ h}$

Al ejecutarse:

$w = 17 \text{ h} = 0001 \ 0111 \text{ b}$

$\text{FSR} = 0001 \ 0111 \text{ b AND } 1100 \ 0010 \text{ b} = 0000 \ 0010 \text{ b} = 02 \text{ h}$

```

0001 0111 b
1100 0010 b
-----
0000 0010 b

```

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	X	-	-

18

Microcontroladores (II)

EC2112

BCF

BCF					BCF
Bit Clear f					
Operación	$0 \rightarrow (f < b >)$				
Sintaxis	[Etiqueta] BCF f,b				
Operadores	$0 < f < 127$ $0 < b < 7$				
Ciclos	1				
OPCODE	01	00bb	bfff	ffff	
Descripción	Pone a cero el bit número b del registro f .				

BCF

Registro de **STATUS**

EJEMPLO: BCF FLAG_REG, 7

Si antes de la instrucción el registro:

FLAG_REG = C7 h = 1100 0111 b

Al ejecutarse la instrucción, el registro queda con el valor:

FLAG_REG = 47b = 0100 0111 b

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

BSF

BSF					BSF
Bit Set f					
Operación	1 → (f)				
Sintaxis	[Etiqueta] BSF f,b				
Operadores	0 < f < 127 0 < b < 7				
Ciclos	1				
OPCODE	01	11bb	bfff	ffff	
Descripción	Pone a 1 el bit b del registro f				

BSF

Registro de **STATUS**

EJEMPLO: BSF FLAG_REG, 7

Si antes de la instrucción el registro tiene el valor:

FLAG_REG = 0A h = 0000 1010 b

Al ejecutarse la instrucción, el registro queda con el valor:

FLAG_REG = 8A h = 1000 1010 b

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

BTFSK

BTFSK					BTFSK
Bit Test, Skip if Clear					
Operación	Salta si ($f < b$) = 0				
Sintaxis	[Etiqueta] BTFSK f,b				
Operadores	$0 < f < 127$ $0 < b < 7$				
Ciclos	1 (2)				
OPCODE	01	10bb	bfff	ffff	
Descripción	Si el bit número b del registro f es cero, la instrucción que sigue a ésta se ignora y se trata como un NOP (skip). En este caso, y sólo en este caso, la instrucción BTFSK precisa dos ciclos para ejecutarse.				

BTFSK

Registro de **STATUS**

EJEMPLO: INICIO BTFSK FLAG,1
 ES_1 GOTO PROCESO
 ES_0

Si antes de la instrucción: PC = dirección INICIO

Al ejecutarse:

if FLAG<1> = 0,

PC = dirección ES_0 y seguirá la ejecución del programa.

if FLAG<1> = 1,

PC = dirección ES_1 y el programa continuará en PROCESO

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

BTFSS

BTFSS				BTFSS
Bit Test, Skip if Set				
Operación	Salta si (f < b >) = 1			
Sintaxis	[Etiqueta] BTFSS f,b			
Operadores	$0 < f < 127$ $0 < b < 7$			
Ciclos	1 (2)			
OPCODE	01	11bb	bfff	ffff
Descripción	Si el bit número b del registro f está a 1, la instrucción que sigue a ésta se ignora y se trata como un NOP (skip). En este caso, y sólo en este caso, la instrucción BTFSS precisa dos ciclos para ejecutarse.			

25

Microcontroladores (II)

EC2112

BTFSS

Registro de **STATUS**

EJEMPLO: INICIO BTFSS FLAG,1
 ES_0 GOTO PROCESO
 ES_1

Si antes de la instrucción. PC = dirección INICIO

Al ejecutarse:

if FLAG<1> = 0,

PC = dirección ES_0 y el programa continuará en PROCESO.

if FLAG<1> = 1,

PC = dirección ES_1 y seguirá la ejecución del programa.

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

26

Microcontroladores (II)

EC2112

CALL

CALL					CALL
Subrutine Call					
Operación	PC + 1 → TOS k → PC <10:0> PCLATCH (<4:3>) → PC (<12,11>)				
Sintaxis	[Etiqueta] CALL k				
Operandos	0 = k = 2047				
Ciclos	2				
OPCODE	10	0kkk	kkkk	kkkk	kkkk
Descripción	Guarda la dirección de retorno en la pila y después llama a la subrutina situada en la dirección cargada en el PC. El modo de cálculo de la dirección efectiva difiere según la familia PIC utilizada. También hay que posicionar PA2, PA1 y PA0 (PIC 16C5X) o el registro PCLATCH (En los demás PIC) antes de ejecutarse la instrucción.				

CALL

Registro de **STATUS**

EJEMPLO: INICIO CALL SUB_1

Si antes de la instrucción:

PC = dirección INICIO

Al ejecutarse:

PC = dirección SUB_1

TOS = dirección INICIO + 1

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

CLRF

CLRF					CLRF
Clear f					
Operación	00 h → f 1 → Z				
Sintaxis	[Etiqueta] CLRF f				
Operadores	0 < f < 127				
Ciclos	1				
OPCODE	00	0001	1fff	fff	
Descripción	Se borra el contenido del registro f y la bandera Z se activa				

CLRF

Registro de **STATUS**

Z toma el valor de 1 si el resultado de la operación es cero

EJEMPLO: CLRF REG

Si antes de la instrucción:

REG = 5A h

Al ejecutarse:

REG = 00 h

flag Z = 1

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	1	-	-

CLRW

CLRW				CLRW
Clear w				
Operación	00 h → w 1 → Z			
Sintaxis	[Etiqueta] CLRW			
Operadores	No tiene			
Ciclos	1			
OPCODE	00	0001	0000	0011
Descripción	El registro de trabajo w se carga con 00h. la bandera Z toma el valor de 1			

31

Microcontroladores (II)

EC2112

CLRW

Registro de **STATUS**

Z toma el valor de 1 si el resultado de la operación es cero

EJEMPLO CLRW

Si antes de la instrucción. w = 5Ah

Al ejecutarse:

w = 00

flag Z = 1

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	1	-	-

32

Microcontroladores (II)

EC2112

CLRWDT

CLRWDT					CLRWDT
Clear watchdog Timer					
Operación	00 h → WDT 1 → TO# 1 → PD#				
Sintaxis	[Etiqueta] CLRWDT				
Operadores	No tiene				
Ciclos	1				
OPCODE	00	0000	0110	0100	
Descripción	Se borra tanto el registro WDT (watchdog) como su preescaler. Los bits TO# y PD# del registro de estado toman el valor de "1".				

CLRWDT

Registro de **STATUS**

TO# toma el valor de 1 cuando se ejecuta la instrucción CLRWDT o SLEEP y toma el valor de 0 si el temporizador watchdog se desborda

PD# toma el valor de 1 cuando se ejecuta la instrucción CLRWDT o SLEEP

EJEMPLO CLRWDT

Si antes de ejecutarse la instrucción
WDT = ?

Al ejecutarse: WDT = 00 h

Preescaler WDT = 0

bit de estado TO = 1

bit de estado PD = 1

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	1	1	-	-	-

COMF

COMF				COMF
Complement f				
Operación	Complemento de f → f / Complemento de f → d			
Sintaxis	[Etiqueta] COMF f,d			
Operadores	$0 < f < 127$ d [0,1]			
Ciclos	1			
OPCODE	00	1001	dfff	ffff
Descripción	Hace el complemento del contenido del registro f bit a bit. El resultado se almacena en el registro f si d=1 y en el registro w si d=0 , en este caso f no varía.			

COMF

Registro de **STATUS**

Z toma el valor de 1 si el resultado de la operación es cero

EJEMPLO: COMF REG1,0

Si antes de la instrucción:

REG1 = 13 h como d= 0

Al ejecutarse:

REG1 = 13 h = 0001 0011 b

w = EC h = 1110 1100 b

flag Z = 0

0001 0011 b
1110 1100 b

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	0	-	-

DECF

DECF				DECF
Decrement f				
Operación	$f - 1 \rightarrow d / f - 1 \rightarrow d$			
Sintaxis	[Etiqueta] DECF f,d			
Operadores	$0 < f < 127$ $d [0,1]$			
Ciclos	1			
OPCODE	00	0011	dfff	ffff
Descripción	Se decrementa el contenido del registro f en una unidad. El resultado se almacena en f si d=1 y en w si d=0 , en este caso f no varía.			

37

Microcontroladores (II)

EC2112

DECF

Registro de **STATUS**

Z toma el valor de 1 si el resultado de la operación es cero

EJEMPLO: `DECF CNT,1`

Si antes de la instrucción:

CNT = 01 h

Z = 0

Al ejecutarse:

CNT = 00 h

bit Z = 1

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	1	-	-

38

Microcontroladores (II)

EC2112

DECFSZ

DECFSZ				DECFSZ
Decrement f , Skip if 0				
Operación	f - 1 → d , salta si resultado = 0			
Sintaxis	[Etiqueta] DECFSZ f,d			
Operadores	0 < f < 127 d [0.1]			
Ciclos	1 (2)			
OPCODE	00	1011	dfff	ffff
Descripción	Decrementa el contenido del registro f en una unidad, el resultado se almacena en f si d=1 y en w si d=0 , en este caso, f no varía. Si el resultado es cero, se ignora la siguiente instrucción y, en ese caso la instrucción tiene una duración de dos ciclos.			

39

Microcontroladores (II)

EC2112

DECFSZ

Registro de **STATUS**

EJEMPLO: INICIO DECFSZ CNT,1
 GOTO LOOP
 CONTINUAR

Si antes de la instrucción: PC = dirección INICIO

Al ejecutarse: CNT = CNT -1

Si CNT = 0 entonces PC = dirección CONTINUAR

Si CNT no = 0 entonces PC = dirección INICIO + 1

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

40

Microcontroladores (II)

EC2112

GOTO

GOTO		GOTO		
Unconditional Branch				
Operación	$k \rightarrow PC <10:0>$ (PCLATH <4:3>) \rightarrow (PC <12:11>)			
Sintaxis	[Etiqueta] GOTO <i>k</i>			
Operadores	$0 < k < 2047$			
Ciclos	2			
OPCODE	10	1kkkk	kkkk	kkkk
Descripción	Salto incondicional, normalmente se utiliza para llamar a la subrutina situada en la dirección que se carga en PC. El modo de cálculo de la instrucción es cargar desde el bit 0 al 10 de la constante k en el PC y los bits 3 y 4 del registro PCLATH en los 11 y 12 del PC			

41

Microcontroladores (II)

EC2112

GOTO

Registro de **STATUS**

EJEMPLO: GOTO SEGUIR

Al ejecutarse:

PC = dirección SEGUIR

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

42

Microcontroladores (II)

EC2112

INCF

INCF				INCF
Increment f				
Operación	$f + 1 \rightarrow d / f + 1 \rightarrow w$			
Sintaxis	[Etiqueta] INCF f,d			
Operadores	$0 < f < 127$ $d [0,1]$ $f + 1 \rightarrow d$			
Ciclos	1			
OPCODE	00	1010	dfff	ffff
Descripción	Se incrementa en una unidad el contenido del registro f , si d=1 el resultado se almacena en f , si d=0 el resultado se almacena en w , en este caso el resultado de f no varía.			

43

Microcontroladores (II)

EC2112

INCF

Registro de **STATUS**

Z toma el valor de 1 si el resultado de la operación es cero al haber desbordamiento

EJEMPLO: INCF CNT,1

Si antes de la instrucción:

CNT = FF h

flag Z = 0

Al ejecutarse:

FF h + 1 h = 00 h

CNT = 00

flag Z = 1

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	1	-	-

44

Microcontroladores (II)

EC2112

INCFSZ

INCFSZ					INCFSZ
Increment f, SKIP if 0					
Operación	$f + 1 \rightarrow d$, salta si resultado = 0				
Sintaxis	[Etiqueta] INCFSZ f,d				
Operadores	$0 < f < 127$ d [0,1]				
Ciclos	1 (2)				
OPCODE	00	1111	dfff	ffff	
Descripción	Incrementa el contenido del registro f en una unidad, el resultado se almacena de nuevo en f si d=1 , y en w si d=0 , en este caso, f no varía. Si el resultado es cero, se ignora la siguiente instrucción y, en ese caso la instrucción tiene una duración de dos ciclos.				

45

Microcontroladores (II)

EC2112

INCFSZ

Registro de **STATUS**

EJEMPLO: INICIO INCFSZ CNT,1
GOTO SALTO
CONTINUAR

Si antes de la instrucción:
PC = dirección INICIO

Al ejecutarse: CNT = CNT+1
Si CNT = 0
Entonces PC = dirección CONTINUAR
Si CNT ≠ 0
Entonces PC = dirección INICIO + 1

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

46

Microcontroladores (II)

EC2112

IORLW

IORLW					IORLW
Inclusive OR Literal with w					
Operación	$w \text{ OR } k \rightarrow w$				
Sintaxis	[Etiqueta] IORLW k				
Operadores	$0 < k < 255$				
Ciclos	1				
OPCODE	11	1000	kkkk	kkkk	
Descripción	Se realiza la operación OR entre el registro w y el literal k . El resultado se almacena en el registro w . Esta instrucción realiza la operación OR bit a bit.				

47

Microcontroladores (II)

EC2112

IORLW

Registro de **STATUS**

Z toma el valor de 1 si el resultado de la operación es cero.

EJEMPLO: IORLW 0x35

Si antes de la instrucción:

w = 9A h

flag Z = 0

Al ejecutarse:

w = 1001 1010 b + 0011 0101 b = 1011 1111 b = BF h

flag Z = 0

```

1001 1010 b
0011 0101 b
-----
1011 1111 b

```

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	0	-	-

48

Microcontroladores (II)

EC2112

IORWF

IORWF					IORWF
Inclusive OR w with f					
Operación	$w \text{ OR } f \rightarrow d / w \text{ OR } f \rightarrow w$				
Sintaxis	[Etiqueta] IORWF f,d				
Operadores	$0 < f < 127$ $d [0,1]$				
Ciclos	1				
OPCODE	00	0100	dfff	fff	
Descripción	Efectúa la operación OR entre el contenido del registro w y el contenido del registro f , y almacena el resultado en f si d=1 y en w si d=0 . Esta instrucción realiza la operación OR bit a bit.				

IORWF

Registro de **STATUS**

Z toma el valor de 1 si el resultado de la operación es cero.

EJEMPLO: IORWF RESULT,0

Si antes de la instrucción

RESULT = 13 h = 0001 0011 b

w = 91 h = 1001 0001 b

flag Z = 0

Al ejecutarse:

RESULT = 0001 0011 b OR 1001 0001 b = 1001 0011 b = 93 h

flag Z = 0

```

0001 0011 b
1001 0001 b
-----
1001 0011 b

```

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	0	-	-

MOVLW

MOVLW				MOVLW
Move literal to w				
Operación	$k \rightarrow w$			
Sintaxis	[Etiqueta] MOVLW <i>k</i>			
Operadores	$0 < f < 255$			
Ciclos	1			
OPCODE	11	00xx	kkkk	kkkk
Descripción	El registro w se carga con el valor de 8 bits del literal k			

51

Microcontroladores (II)

EC2112

MOVLW

Registro de **STATUS**

EJEMPLO: **MOVLW** 0x5A

Al ejecutarse:
w = 5A h

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	X	-	-

52

Microcontroladores (II)

EC2112

MOVF

MOVF				MOVF
Move f				
Operación	$f \rightarrow w / f \rightarrow f$			
Sintaxis	[Etiqueta] MOVF f,d			
Operadores	$0 < f < 127$ d [0,1]			
Ciclos	1			
OPCODE	00	10000	d ff f	ffff
Descripción	El contenido del registro f se carga en el registro destino dependiendo del valor de d . Si d=0 el destino es el registro w , si d=1 el destino es el propio registro f . Esta instrucción permite verificar dicho registro ya que la bandera Z queda afectada.			

53

Microcontroladores (II)

EC2112

MOVF

Registro de **STATUS**

Z toma el valor de 1 si el resultado de la operación es cero.

EJEMPLO: MOVF FSR,0

Al ejecutarse:

w = el valor del FSR

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	X	-	-

54

Microcontroladores (II)

EC2112

MOVWF

MOVWF				MOVWF
Move w to f				
Operación	w → f			
Sintaxis	[Etiqueta] MOVWF f			
Operadores	$0 < f < 127$			
Ciclos	1			
OPCODE	00	0000	1f ff	ffff
Descripción	Mueve el contenido del registro w al registro f			

55

Microcontroladores (II)

EC2112

MOVWF

Registro de **STATUS**

EJEMPLO: MOVWF OPCION

Si antes de la instrucción:

OPCION = FF h

w = 4F h

Al ejecutarse:

OPCION = 4F h

w = 4F h

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

56

Microcontroladores (II)

EC2112

NOP

NOP				NOP
No operation				
Operación	no operación			
Sintaxis	[Etiqueta] NOP			
Operadores	No tiene			
Ciclos	1			
OPCODE	00	0000	0xx0	0000
Descripción	No realiza operación alguna, pero sirve para consumir un ciclo de instrucción, equivalente a 4 ciclos de reloj.			

NOP

Registro de **STATUS**

EJEMPLO: Si usamos un cristal de cuarzo de 4 Mhz en el oscilador, podremos obtener un retardo igual a un microsegundo por cada instrucción NOP que insertemos en el código del programa:

```
RETARDO    NOP
           NOP
           NOP
           RETURN
```

Cada vez que llamemos a la subrutina RETARDO, obtendremos 3 microsegundos de demora.

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

RETFIE

RETFIE				RETFIE
Return from Interrupt				
Operación	TOS → PC 1 → GIE			
Sintaxis	[Etiqueta] RETFIE			
Operadores	No tiene			
Ciclos	2			
OPCODE	00	0000	0000	1001
Descripción	Carga el PC con el valor que se encuentra en la parte alta de la Pila, asegurando así la vuelta de la interrupción. Pone a 1 el bit GIE , con el fin de autorizar de nuevo que se tengan en cuenta las interrupciones.			

RETFIE

Registro de **STATUS**

EJEMPLO: RETFIE

Al ejecutarse:

PC = TOS

GIE = 1

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

RETLW

RETLW					RETLW
Return with Literal in w					
Operación	$k \rightarrow w;$ $TOS \rightarrow PC$				
Sintaxis	[Etiqueta] RETLW <i>k</i>				
Operadores	$0 < k < 255$				
Ciclos	2				
OPCODE	11	01xx	kkkk	kkkk	
Descripción	Carga el registro w con el literal k , y después carga el PC con el valor que se encuentra en la parte superior de la PILA, efectuando así un retorno de subrutina.				

61

Microcontroladores (II)

EC2112

RETLW

Registro de **STATUS**

EJEMPLO:

```

MOVLW    0x07    ;se carga 07 h en w
CALL     TABLA   ;tabla de valores y w
...      ...     ;contiene el valor guardado
TABLA    ADDWF   PC    ;se añade a PC el
           ;desplazamiento (offset) de w
           RETLW  k1   ;inicio de tabla
           RETLW  k2
           ...
           RETLW  kn   ;fin de tabla

```

Al ejecutarse la instrucción:

w = toma el valor de k7

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

62

Microcontroladores (II)

EC2112

RETURN

RETURN				RETURN
Return from Subroutine				
Operación	TOS – PC			
Sintaxis	[Etiqueta] RETURN			
Operadores	No tiene			
Ciclos	2			
OPCODE	00	0000	0000	1000
Descripción	Carga el PC con el valor que se encuentra en la parte superior de la PILA, efectuando así un retorno de subrutina			

63

Microcontroladores (II)

EC2112

RETURN

Registro de **STATUS**

EJEMPLO: RETURN

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

64

Microcontroladores (II)

EC2112

RLF

RLF				RLF
Rotate Left f through Carry				
Operación				
Sintaxis	[Etiqueta] RLF f,d			
Operadores	0 < f < 127 d [0,1]			
Ciclos	1			
OPCODE	00	1101	dfff	ffff
Descripción	Rotación de un bit a la izquierda del contenido del registro f , pasando por el bit de acarreo C , desde los bits menos significativos a los más significativos. El bit D7 pasa al CARRY del registro STATUS, el contenido del CARRY pasa al D0, el D0 al D1, etc. Es como si multiplicáramos por dos el contenido del registro. Si d=1 el resultado se almacena en f , si d=0 el resultado se almacena en w .			

RLF

Registro de **STATUS**

EJEMPLOS:

Si tenemos el registro VALOR = 0000 0001 b y aplicamos la instrucción **RLF VALOR,1** entonces el resultado será:

VALOR = 0000 0010 b y el bit C = 0.

Si tenemos el registro VALOR = 1110 0110 b y aplicamos la instrucción **RLF VALOR** El resultado será:

VALOR = 1100 1100 b y el bit C = 1.

Si antes de la instrucción REG1 = 1110 0110 b y flag C = 0 y aplicamos la instrucción **RLF REG1,0**, como d = 0 el resultado queda en w,

Al ejecutarse:

REG1 = 1110 0110 b

w = 1100 1100 b

flag C = 1

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	X

RRF

RRF					RRF
Rotate Right f through Carry					
Operación					
Sintaxis	[Etiqueta] RRF f,d				
Operadores	$0 < f < 127$ $d [0,1]$				
Ciclos	1				
OPCODE	00	1100	dfff	ffff	
Descripción	Rotación de un bit a la derecha del contenido del registro f , pasando por el bit de acarreo C , desde los bits más significativos a los menos significativos. El bit C del registro STATUS pasa al D7, el D0 pasa al bit C, el D1 al D0, etc. Es como si dividiéramos por dos el contenido del registro. Si d=1 el resultado se almacena en f , si d=0 el resultado se almacena en w				

67

Microcontroladores (II)

EC2112

RRF

Registro de **STATUS**

EJEMPLOS:

Si tenemos el registro VALOR = 0000 0001 b y el flag C=0, si aplicamos la instrucción **RRF VALOR,1** el resultado será:

VALOR = 0000 0000 b y el bit C = 1.

Si tenemos el registro VALOR = 1000 0000 b y el flag C=1, si aplicamos la instrucción **RRF VALOR,1** el resultado será:

VALOR = 0100 0000 b y el bit C = 0.

Si tenemos el registro REG1 = 1110 0110 b y flag C = 1, si aplicamos la instrucción **RRF REG1,0** como d = 0 el resultado queda en w, al ejecutarse:

REG1 = 1110 0110 b

w = 0111 0011 b

flag C = 0

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	X

68

Microcontroladores (II)

EC2112

SLEEP

SLEEP					SLEEP
Sleep					
Operación	00 h → WDT 0 → WDT prescaler 1 → TO# 0 → PD#				
Sintaxis	[Etiqueta] SLEEP				
Operadores	No tiene				
Ciclos	1				
OPCODE	00	0000	0110	0011	
Descripción	Pone al circuito en modo Sleep (bajo consumo) con parada del oscilador. Pone a 0 la bandera PD# (Power Down) y la bandera TO# (Timer Out) toma el valor de 1. Se puede salir de este estado por: activación de MCLR para provocar un Reset, desbordamiento del watchdog si quedó operativo en el modo reposo y generación de una interrupción que no sea TMR0 ya que ésta se desactiva con la instrucción SLEEP.				

SLEEP

Registro de **STATUS**

TO toma el valor de 1 al ejecutar la instrucción SLEEP o CLRWDT

PD toma el valor de 0 al ejecutar la instrucción SLEEP.

EJEMPLO: SLEEP

PA2	PA1	PA0	TO#	PD#	Z#	DC	C
-	-	-	1	0	-	-	-

SUBLW

SUBLW				SUBLW
Subtract w from Literal				
Operación	$k - w \rightarrow w$			
Sintaxis	[Etiqueta] SUBLW <i>k</i>			
Operadores	$0 < k < 255$			
Ciclos	1			
OPCODE	11	110x	kkkk	kkkk
Descripción	Resta el complemento a dos del contenido del literal k el contenido del registro w , y almacena el resultado en w .			

71

Microcontroladores (II)

EC2112

SUBLW

Registro de **STATUS**

Z toma el valor de 1 si el resultado de la operación es cero

DC toma el valor de 1 si se genera un acarreo del bit 3 al grupo de 4 bits superior

C toma el valor de 1 si se genera un acarreo del bit de mayor peso.

EJEMPLO: SUBLW 0x02 ;k - w → w, 02 h - w → w

Si antes de la instrucción w = 01 h y flag C = ? al ejecutarse:

02 h - 01 h = 01 h

w = 01 h

flag C = 1 ; el resultado es positivo

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	X	X	1

72

Microcontroladores (II)

EC2112

SUBLW

b) Si antes de la instrucción $w = 02\text{ h}$, flag $C = ?$ y flag $Z = ?$
 al ejecutarse: $02\text{ h} - 02\text{ h} = 00\text{ h}$
 $w = 00\text{ h}$
 flag $C = 1$
 flag $Z = 1$

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	1	X	1

c) Si antes de la instrucción $w = 03\text{ h}$ y flag $C = ?$
 al ejecutarse: $02\text{ h} - 03\text{ h} = 02\text{ h} + (C2\ 03\text{ h}) = 02\text{ h} + (C1\ 03\text{ h} + 1\text{ h})$
 $(00000010)_b + (1111\ 1100)_b + (00000001)_b = (11111111)_b = \text{FF h}$
 $w = \text{FF h}$
 flag $C = 0$

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	X	X	1

73

Microcontroladores (II)

EC2112

SUBWF

SUBWF				SUBWF
Subtract w from f				
Operación	$f - w \rightarrow d$			
Sintaxis	[Etiqueta] SUBWF f,d			
Operadores	$0 < f < 127$ $d [0,1]$			
Ciclos	1			
OPCODE	00	0010	dfff	ffff
Descripción	Resta en complemento a dos el contenido del registro f menos el contenido del registro w almacena el resultado en w si d=0 y en f si d=1 .			

74

Microcontroladores (II)

EC2112

SUBWF

Registro de **STATUS**

Z toma el valor de 1 si el resultado de la operación es cero

DC toma el valor de 1 si se genera un acarreo del bit 3 al grupo de 4 bits superior

C toma el valor de 1 si se genera un acarreo del bit de mayor peso.

EJEMPLO: SUBWF REG1,1 ; f - w → f, REG1 - w → REG1

a) Si antes de la instrucción, REG1 = 03 h, w = 02 h y flag C = ?, al

ejecutarse: 03 h - 02h = 01 h

REG1 = 01h

w = 02 h

flag C = 1 ; el resultado es positivo

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	X	X	X

75

Microcontroladores (II)

EC2112

SUBWF

b) Si antes de la instrucción, REG1 = 02 h, w = 02 h y flag C = ?, al

ejecutarse: 02 h - 02h = 00 h

REG1 = 00h

w = 02 h

flag C = 1

flag Z = 1 ; el resultado es cero

c) Si antes de la instrucción, REG1 = 01 h, w = 02 h y flag C = ?, al

ejecutarse: 01 h - 02 h = -01 h = - 0000 0001 b

C1(0000 0001 b)=1111 1110 b; 1111 1110 b + 1 b = 1111 1111 b = FF

h

REG1 = FF h

w = 02 h

flag C = 0 ; el resultado es negativo

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	X	X	X

76

Microcontroladores (II)

EC2112

SWAPF

SWAPF				SWAPF
Swap Nibbles in f				
Operación	$(f<3:0>) \rightarrow (d<7:4>)$ $(f<7:4>) \rightarrow (d<3:0>)$			
Sintaxis	[Etiqueta] SWAPF f,d			
Operadores	$0 < f < 127$ $d [0,1]$			
Ciclos	1			
OPCODE	00	1110	dfff	ffff
Descripción	Los cuatro bits de más peso del registro f se intercambian con los 4 bits de menos peso del mismo registro. Si d=0 el resultado se almacena en w , si d=1 el resultado se almacena en f .			

SWAPF

Registro de **STATUS**

EJEMPLO: SWAPF REG1,0

Si antes de la instrucción:

REG1 = A5 h = 1010 0101 h

Como d=0 el resultado se almacenará en w

Al ejecutarse la instrucción:

REG1 = A5 h = 1010 0101 b

w = 5A h = 0101 1010 b

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

XORLW

XORLW					XORLW
Exclusive OR Literal with k					
Operación	$w \text{ XOR } k \rightarrow w$				
Sintaxis	[Etiqueta] XORLW <i>k</i>				
Operadores	$0 < f < 255$				
Ciclos	1				
OPCODE	11	1010	kkkk	kkkk	
Descripción	Realiza la función OR-Exclusiva entre el contenido del registro w y la constante k de 8 bits. El resultado se almacena en w . Esta instrucción realiza la operación EXOR bit a bit.				

XORLW

Registro de **STATUS**

Z toma el valor de 1 si el resultado de la última operación es cero.

EJEMPLO: XORLW 0xAF

Si antes de la instrucción:

$w = (10110101)_b = B5 \text{ h}$

Al ejecutarse la instrucción:

$w = (10110101)_b \rightarrow (10101111)_b = (00011010)_b = 1A \text{ h}$

$$\begin{array}{r} (10110101)_b \\ \underline{(10101111)_b} \\ (00011010)_b \end{array}$$

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	1	-	-

XORWF

XORWF	XORWF
Exclusive OR w with f	
Operación	$w \text{ XOR } f \rightarrow d$
Sintaxis	[Etiqueta] XORWF f,d
Operadores	$0 < f < 127$ $d [0,1]$
Ciclos	1
Descripción	Realiza la función OR-Exclusiva entre el contenido del registro w y el contenido del registro f , y almacena el resultado en f si d=1 y en w si d=0 . Esta instrucción realiza la operación EXOR bit a bit.

81

Microcontroladores (II)

EC2112

XORWF

Registro de **STATUS**

EJEMPLO: XORWFREG1,1

Si antes de la instrucción:

REG1 = AF h = 1010 1111 b

w = B5 h = 1011 0101 b

Como d=1, el resultado se almacena en REG1

Al ejecutarse:

REG1 = 1010 1111 \rightarrow 1011 0101 \oplus 0001 1010 = 1A h

w = B5 h

```

1010 1111
1011 0101
-----
0001 1010

```

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	1	-	-

82

Microcontroladores (II)

EC2112

OPTION

OPTION				OPTION
Guarda el valor del acumulador en el registro OPTION				
Operación	w → OPTION			
Sintaxis	[Etiqueta] OPTION			
Operadores	No tiene			
Ciclos	1			
OPCODE	00	0000	0110	0010
Descripción	Esta instrucción guarda en el registro especial OPTION el valor contenido en el acumulador w . No modifica ningún bit de estado.			

83

Microcontroladores (II)

EC2112

OPTION

Registro de **STATUS**

EJEMPLO: `MOVLW 10H` ; carga el acumulador con el valor 10h.
 `OPTION` ; carga el registro OPTION con el valor
 ; contenido en el acumulador

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

84

Microcontroladores (II)

EC2112

TRIS

TRIS				TRIS
Guarda el acumulador en uno de los registros de TRIS .				
Operación	w → f(TRISA ó TRISB)			
Sintaxis	[Etiqueta] TRIS f			
Operadores	No tiene			
Ciclos	1			
OPCODE	00	0000	0110	1111
Descripción	Esta instrucción guarda el valor del acumulador w en uno de los registros especiales TRIS (TRISA o TRISB) que indicamos en el parámetro f . No modifica ningún bit de estado. Los registros TRIS determinan el funcionamiento como entrada y salida de las líneas I/O del PIC.			

85

Microcontroladores (II)

EC2112

TRIS

Registro de **STATUS**

EJEMPLO: MOVLW 16h ; carga el acumulador W con el valor 16h
 TRIS PORTA ; carga el registro PORTA con el acumulador

PA2	PA1	PA0	TO#	PD#	Z	DC	C
-	-	-	-	-	-	-	-

86

Microcontroladores (II)

EC2112

Instrucciones especiales

- ▶ Existe un conjunto de instrucciones especiales diseñadas para facilitar las operaciones a la hora de diseñar los algoritmos
- ▶ Estas instrucciones pueden ser implementadas con una, dos o tres de las instrucciones de la gama media. La mayoría de ellas se basa en las operaciones con los acarrees y con los bits del registro status en general

Instrucciones especiales

Mnemónico Parámetros	Descripción	Traducción	Operación Equivalente	Flag	
ADDCF	f, d	Add Carry to File	Sumar acarreo a f	BTFSC 3,0 INCF f,d	Z
ADDDCF	f, d	Add Digit Carry to File	Sumar acarreo de dígito a f	BTFSC 3,1 INCF f,d	Z
B	K	Branch	Saltar a una etiqueta	GOTO k	-
BC	K	Branch on Carry	Saltar a una etiqueta si hay acarreo	BTFSC 3,0 GOTO k	-
BDC	K	Branch on Digit Carry	Saltar a una etiqueta si hay acarreo de dígito	BTFSC 3,1 GOTO k	-
BNC	K	Branch on No Carry	Saltar a una etiqueta si no hay acarreo	BTFSS 3,0 GOTO k	-
BNDC	K	Branch on No Digit Carry	Saltar a una etiqueta si no hay acarreo de dígito	BTFSS 3,1 GOTO k	-
BNZ	K	Branch on No Zero	Saltar a una etiqueta si no hay cero	BTFSS 3,2 GOTO k	-
BZ	K	Branch on Zero	Saltar a una etiqueta si hay cero	BTFSC 3,2 GOTO k	-

Instrucciones especiales

Mnemónico Parámetros	Descripción	Traducción	Operación Equivalente	Flag	
CLRC		Clear Carry	Poner a cero acarreo	BCF 3,0	-
CLRDC		Clear Digit Carry	Poner a cero acarreo de dígito	BCF 3,1	-
CLRZ		Clear Zero	Poner a cero la bandera Zero	BCF 3,2	-
LCALL	K	Long CALL	Llamada larga a una etiqueta	BSF/BCF 0A,3 BSF/BCF 0A,4 CALL k	-
LGOTO	K	Long GOTO	Salto largo a una etiqueta	BSF/BCF 0A,3 BSF/BCF 0A,4 GOTO k	-
MOVFW	F	Move File to W	Mover registro a W	MOVF r,0	Z
NEGF	f, d	Negate File	Negar un registro	COMF f,1 INCF f,d	Z
SETC		Set Carry	Poner a uno el acarreo	BSF 3,0	-
SETDC		Set Digit Carry	Poner a uno el acarreo de dígito	BSF 3,1	-

Instrucciones especiales

Mnemónico Parámetros	Descripción	Traducción	Operación Equivalente	Flag	
SETZ		Set Zero	Poner a uno el Zero	BSF 3,2	-
SKPC		Skip on Carry	Saltar si hay acarreo	BTFSS 3,0	-
SKPDC		Skip on Digit Carry	Saltar si hay acarreo de dígito	BTFSS 3,1	-
SKPNC		Skip on No Carry	Saltar si no hay acarreo	BTFSC 3,0	-
SKPNDC		Skip on No Digit Carry	Saltar si no hay acarreo de dígito	BTFSC 3,1	-
SKPNZ		Skip on Non Zero	Saltar si no hay Zero	BTFSC 3,2	-
SKPZ		Skip on Zero	Saltar si hay Zero	BTFSS 3,2	-
SUBCF	f,d	Subtract Carry from File	Restar acarreo		