

Welcome to E-XFL.COM

#### What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "<u>Embedded -</u> <u>Microcontrollers</u>"

#### Details

Product Status	Active
Core Processor	PIC
Core Size	8-Bit
Speed	32MHz
Connectivity	I <sup>2</sup> C, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	18
Program Memory Size	14KB (8K x 14)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 17x10b; D/A 1x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	20-UFQFN Exposed Pad
Supplier Device Package	20-UQFN (4x4)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f15345-i-gz

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

REGISTER	5-4: CON	FIGURATION	WORD 4: M	EMORY			
		R/W-1	U-1	R/W-1	U-1	R/W-1	R/W-1
		LVP		WRTSAF <sup>(1)</sup>	_	WRTC <sup>(1)</sup>	WRTB <sup>(1)</sup>
		bit 13	12	11	10	9	bit 8
							-
R/W-1	U-1	U-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
WRTAPP			SAFEN	BBEN	BBSIZE2	BBSIZE1	BBSIZE0
bit 7	6	5	4	3	2	1	bit 0
Legend:							
R = Readable	e bit	P = Programm	able bit	x = Bit is unkr	IOWN	U = Unimplerr read as '1'	nented bit,
'0' = Bit is cle	ared	'1' = Bit is set		W = Writable	bit	n = Value whe after Bulk Era	en blank or se
bit 13	LVP: Low Vo 1 = Low vol ignored	ltage Programm Itage programmi I.	ing Enable b <u>it</u> ng enabled. ₩	ICLR/VPP pin fu	Inction is $\overline{\text{MCL}}$	R. MCLRE Conf	iguration bit is
	0 = HV on I The LVP bit of purpose of th mode, or acc The precond	MCLR/VPP must cannot be writter is rule is to preve cidentally elimina itioned (erased)	be used for p n (to zero) whi ent the user fro ting LVP mod state for this b	rogramming. le operating fro om dropping ou e from the confi bit is critical.	m the LVP pro t of LVP mode iguration state	gramming interf while programn	ace. The ning from LVP
bit 12	Unimplemer	nted: Read as '1	3				
bit 11	WRTSAF: St	torage Area Flas	h Write Prote	ction bit			
	1 = SAF NC 0 = SAF wri Unimplemen	DT write-protecte ite-protected ted, if SAF is no	d t supported in	the device fami	ily and only ap	plicable if SAFE	<del>N</del> = 0.
bit 10	<u>Unimp</u> lemer	nted: Read as '1	,				
bit 9	WRTC: Conf 1 = Configu 0 = Configu	figuration Regist uration Register I uration Register v	er Write Prote NOT write-pro write-protected	ction bit tected d			
bit 8	WRTB: Boot	Block Write Pro	tection bit				
	1 = Boot Bl 0 = Boot Bl Only applical	ock NOT write-p ock write-protec ble if BBEN = 0.	rotected ted				
bit 7	WRTAPP: A 1 = Applica 0 = Applica	pplication Block tion Block NOT tion Block write-	Write Protection write-protected protected	on bit d			
bit 6-5	Unimplemer	nted: Read as '1	3				
bit 4	<b>SAFEN:</b> SAF 1 = SAF dis 0 = SAF en:	Enable bit abled abled					
bit 3	BBEN: Boot           1 =         Boot Blo           0 =         Boot Blo	Block Enable bi ock disabled ock enabled	t				
bit 2-0	BBSIZE[2:0]: BBSIZE is us BBSIZ bits ca	: Boot Block Siz <u>e</u> sed only when B an only be writte	Selection bits BEN = 0 n while BBEN	= 1; after BBEI	N = 0, BBSIZ i	s write-protecte	d.

**Note 1:** Bits are implemented as sticky bits. Once protection is enabled, it can only be reset through a Bulk Erase.

### 6.0 DEVICE INFORMATION AREA

The Device Information Area (DIA) is a dedicated region in the program memory space; it is a new feature in the PIC16(L)F15325/45 family of devices. The DIA contains the calibration data for the internal temperature indicator module, stores the Microchip Unique Identifier words and the Fixed Voltage Reference voltage readings measured in mV.

The complete DIA table is shown in Table 6-1: Device Information Area, followed by a description of each region and its functionality. The data is mapped from 8100h to 811Fh in the PIC16(L)F15325/45 family. These locations are read-only and cannot be erased or modified. The data is programmed into the device during manufacturing.

TABLE 6-1: DEVICE INFORMATION AREA

Address Range	Name of Region	Standard Device Information
	MUI0	
	MUI1	
	MUI2	
	MUI3	
8100h-8108h	MUI4	Microchip Unique Identifier (9 Words)
	MUI5	
	MUI6	
	MUI7	
	MUI8	
8109h	MUI9	1 Word Reserved
	EUI0	
	EUI1	
	EUI2	
04046 04446	EUI3	
810An-8111n	EUI4	Unassigned (8 words)
	EUI5	
	EUI6	
	EUI7	
8112h	TSLR1	Unassigned (1 word)
8113h	TSLR2	Temperature indicator ADC reading at 90°C (low range setting)
8114h	TSLR3	Unassigned (1 word)
8115h	TSHR1	Unassigned (1 word)
8116h	TSHR2	Temperature indicator ADC reading at 90°C (high range setting)
8117h	TSHR3	Unassigned (1 Word)
8118h	FVRA1X	ADC FVR1 Output voltage for 1x setting (in mV)
8119h	FVRA2X	ADC FVR1 Output Voltage for 2x setting (in mV)
811Ah	FVRA4X <sup>(1)</sup>	ADC FVR1 Output Voltage for 4x setting (in mV)
811Bh	FVRC1X	Comparator FVR2 output voltage for 1x setting (in mV)
811Ch	FVRC2X	Comparator FVR2 output voltage for 2x setting (in mV)
811Dh	FVRC4X <sup>(1)</sup>	Comparator FVR2 output voltage for 4x setting (in mV)
811Eh-811Fh		Unassigned (1 Word)

Note 1: Value not present on LF devices.

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
OSCCON1	—		NOSC<2:0>		NDIV<3:0>				113
OSCCON2	—		COSC<2:0>		CDIV<3:0>				113
OSCCON3	CWSHOLD	SOSCPWR		ORDY	NOSCR	_	_	_	114
OSCFRQ	—	_	_	_	_	HFFRQ<2:0>			
OSCSTAT	EXTOR	HFOR	MFOR	LFOR	SOR	ADOR	—	PLLR	115
OSCTUNE	_	_		HFTUN<5:0>					
OSCEN	EXTOEN	HFOEN	MFOEN	LFOEN	SOSCEN	ADOEN	_	_	116

#### TABLE 9-3: SUMMARY OF REGISTERS ASSOCIATED WITH CLOCK SOURCES

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by clock sources.

#### TABLE 9-4: SUMMARY OF CONFIGURATION WORD WITH CLOCK SOURCES

Name	Bits	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	Register on Page
	13:8	_	_	FCMEN	_	CSWEN	_	_	CLKOUTEN	00
CONFIGT	7:0		F	RSTOSC<2:0	>	—	I	EXTOSC<2:0	>	80

Legend: — = unimplemented location, read as '0'. Shaded cells are not used by clock sources.

#### REGISTER 10-11: PIR1: PERIPHERAL INTERRUPT REQUEST REGISTER 1

R/W/HS-0/0	R/W/HS-0/0	U-0	U-0	U-0	U-0	U-0	R/W/HS-0/0		
OSFIF	CSWIF	—	—	—	—	_	ADIF		
bit 7							bit 0		
Legend:									
R = Readable	bit	W = Writable	bit	U = Unimpler	nented bit, read	as '0'			
u = Bit is unch	anged	x = Bit is unkr	nown	-n/n = Value a	at POR and BOI	R/Value at all c	other Resets		
'1' = Bit is set		'0' = Bit is clea	ared	HS = Hardwa	ire set				
bit 7	bit 7 <b>OSFIF</b> : Oscillator Fail-Safe Interrupt Flag bit 1 = Oscillator fail-safe interrupt has occurred (must be cleared in software) 0 = No oscillator fail-safe interrupt								
bit 6	<ul> <li>CSWIF: Clock Switch Complete Interrupt Flag bit</li> <li>1 = The clock switch module indicates an interrupt condition and is ready to complete the clock switch operation (must be cleared in software)</li> <li>0 = The clock switch does not indicate an interrupt condition</li> </ul>								
bit 5-1	Unimplemen	ted: Read as '	0'						
bit 0	<ul> <li>ADIF: Analog-to-Digital Converter (ADC) Interrupt Flag bit</li> <li>1 = An A/D conversion or complex operation has completed (must be cleared in software)</li> <li>0 = An A/D conversion or complex operation is not complete</li> </ul>								
Note: Interrupt flag bits are set when an interrupt condition occurs, regardless of the state of its corresponding enable bit or the Global Enable bit, GIE, of the INTCON register. User software should ensure the appropriate interrupt flag bits are clear prior to enabling an interrupt.									

#### 13.3.2 NVM UNLOCK SEQUENCE

The unlock sequence is a mechanism that protects the NVM from unintended self-write programming or erasing. The sequence must be executed and completed without interruption to successfully complete any of the following operations:

- PFM Row Erase
- Load of PFM write latches
- Write of PFM write latches to PFM memory
- Write of PFM write latches to User IDs

The unlock sequence consists of the following steps and must be completed in order:

- Write 55h to NVMCON2
- Write AAh to NMVCON2
- Set the WR bit of NVMCON1

Once the WR bit is set, the processor will stall internal operations until the operation is complete and then resume with the next instruction.

Note:	The two NOP instruction	ns after settin	ig the
	WR bit that were req	uired in pre	vious
	devices are not	required	for
	PIC16(L)F15325/45	devices.	See
	Figure 13-2.		

Since the unlock sequence must not be interrupted, global interrupts should be disabled prior to the unlock sequence and re-enabled after the unlock sequence is completed.

EXAMPLE 13	3-2:	NV	M UNL	.OCK	SEQU	JEN	CE

FIGURE 13-2:	NVM UNLOCK



BCF BANKSEL	INTCON, GIE NVMCON1	; Recommended so sequence is not interrupted ;
BSF	NVMCON1, WREN	; Enable write/erase
MOVLW	55h	; Load 55h
MOVWF	NVMCON2	; Step 1: Load 55h into NVMCON2
MOVLW	AAh	; Step 2: Load W with AAh
MOVWF	NVMCON2	; Step 3: Load AAH into NVMCON2
BSF	NVMCON1, WR	; Step 4: Set WR bit to begin write/erase
BSF	INTCON, GIE	; Re-enable interrupts
Note 1:	Sequence begins whe	en NVMCON2 is written; steps 1-4 must occur in the cycle-accurate order shown.
2:	Opcodes shown are i	lustrative; any instruction that has the indicated effect may be used.



#### 13.3.8 WRERR BIT

The WRERR bit can be used to determine if a write error occurred.

WRERR will be set if one of the following conditions occurs:

- If WR is set while the NVMADRH:NMVADRL points to a write-protected address
- A Reset occurs while a self-write operation was in progress
- An unlock sequence was interrupted

The WRERR bit is normally set by hardware, but can be set by the user for test purposes. Once set, WRERR must be cleared in software.

Free	LWLO	Actions for PFM when WR = 1	Comments
1	x	Erase the 32-word row of NVMADRH:NVMADRL location. See Section 13.3.3 "NVMREG Erase of PFM"	<ul> <li>If WP is enabled, WR is cleared and WRERR is set</li> <li>All 32 words are erased</li> <li>NVMDATH:NVMDATL is ignored</li> </ul>
0	1	Copy NVMDATH:NVMDATL to the write latch corresponding to NVMADR LSBs. See Section 13.3.3 "NVMREG Erase of PFM"	<ul><li>Write protection is ignored</li><li>No memory access occurs</li></ul>
0	0	Write the write-latch data to PFM row. See Sec- tion 13.3.3 "NVMREG Erase of PFM"	<ul> <li>If WP is enabled, WR is cleared and WRERR is set</li> <li>Write latches are reset to 3FFh</li> <li>NVMDATH:NVMDATL is ignored</li> </ul>

TABLE 13-4: ACTIONS FOR PFM WHEN WR = 1

#### **REGISTER 20-4:** ADRESH: ADC RESULT REGISTER HIGH (ADRESH) ADFM = 0

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u				
	ADRES<9:2>										
bit 7							bit 0				
Legend:											
R = Readable	Readable bit W = Writable bit		bit	U = Unimplemented bit, read as '0'							
u = Bit is unchanged x = Bit is unknown			nown	-n/n = Value at POR and BOR/Value at all other Resets							

bit 7-0 **ADRES<9:2>**: ADC Result Register bits Upper eight bits of 10-bit conversion result

'1' = Bit is set

#### **REGISTER 20-5:** ADRESL: ADC RESULT REGISTER LOW (ADRESL) ADFM = 0

'0' = Bit is cleared

| R/W-x/u |
|---------|---------|---------|---------|---------|---------|---------|---------|
| ADRES   | S<1:0>  | —       | —       | —       | —       | —       | —       |
| bit 7   |         |         |         |         |         |         | bit 0   |

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	

bit 7-6 **ADRES<1:0>**: ADC Result Register bits Lower two bits of 10-bit conversion result

Lower two bits of 10-bit conversion resu

bit 5-0 Reserved: Do not use.

## 22.0 NUMERICALLY CONTROLLED OSCILLATOR (NCO) MODULE

The Numerically Controlled Oscillator (NCO) module is a timer that uses overflow from the addition of an increment value to divide the input frequency. The advantage of the addition method over simple counter driven timer is that the output frequency resolution does not vary with the divider value. The NCO is most useful for application that requires frequency accuracy and fine resolution at a fixed duty cycle.

Features of the NCO include:

- 20-bit Increment Function
- Fixed Duty Cycle mode (FDC) mode
- Pulse Frequency (PF) mode
- Output Pulse Width Control
- Multiple Clock Input Sources
- Output Polarity Control
- Interrupt Capability

Figure 22-1 is a simplified block diagram of the NCO module.

#### 22.2 FIXED DUTY CYCLE MODE

In Fixed Duty Cycle (FDC) mode, every time the accumulator overflows (NCO\_overflow), the output is toggled at a frequency rate half of the FOVERFLOW. This provides a 50% duty cycle, provided that the increment value remains constant. For more information, see Figure 22-2.

The FDC mode is selected by clearing the N1PFM bit in the NCO1CON register.

#### 22.3 PULSE FREQUENCY MODE

In Pulse Frequency (PF) mode, every time the Accumulator overflows, the output becomes active for one or more clock periods. Once the clock period expires, the output returns to an inactive state. This provides a pulsed output. The output becomes active on the rising clock edge immediately following the overflow event. For more information, see Figure 22-2.

The value of the active and inactive states depends on the polarity bit, N1POL in the NCO1CON register.

The PF mode is selected by setting the N1PFM bit in the NCO1CON register.

#### 22.3.1 OUTPUT PULSE WIDTH CONTROL

When operating in PF mode, the active state of the output can vary in width by multiple clock periods. Various pulse widths are selected with the N1PWS<2:0> bits in the NCO1CLK register.

When the selected pulse width is greater than the Accumulator overflow time frame, then NCO1 output does not toggle.

#### 22.4 OUTPUT POLARITY CONTROL

The last stage in the NCO module is the output polarity. The N1POL bit in the NCO1CON register selects the output polarity. Changing the polarity while the interrupts are enabled will cause an interrupt for the resulting output transition.

The NCO output signal (NCO1\_out) is available to the following peripherals:

- CLC
- CWG
- Timer1
- Timer2
- CLKR

#### 22.5 Interrupts

When the accumulator overflows (NCO\_overflow), the NCO Interrupt Flag bit, NCO1IF, of the PIR7 register is set. To enable the interrupt event (NCO\_interrupt), the following bits must be set:

- N1EN bit of the NCO1CON register
- NCO1IE bit of the PIE7 register
- PEIE bit of the INTCON register
- · GIE bit of the INTCON register

The interrupt must be cleared by software by clearing the NCO1IF bit in the Interrupt Service Routine.

#### 22.6 Effects of a Reset

All of the NCO registers are cleared to zero as the result of a Reset.

#### 22.7 Operation in Sleep

The NCO module operates independently from the system clock and will continue to run during Sleep, provided that the clock source selected remains active.

The HFINTOSC remains active during Sleep when the NCO module is enabled and the HFINTOSC is selected as the clock source, regardless of the system clock source selected.

In other words, if the HFINTOSC is simultaneously selected as the system clock and the NCO clock source, when the NCO is enabled, the CPU will go idle during Sleep, but the NCO will continue to operate and the HFINTOSC will remain active.

This will have a direct effect on the Sleep mode current.

<b>REGISTER 2</b>	5-1: TOCON	0: TIMER0		REGISTER 0			
R/W-0/0	U-0	R-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
TOEN	—	TOOUT	T016BIT		TOOUTI	PS<3:0>	
bit 7							bit 0
Legend:							
R = Readable	bit	W = Writable	bit	U = Unimpler	mented bit, read	l as '0'	
u = Bit is unch	anged	x = Bit is unk	nown	-n/n = Value a	at POR and BO	R/Value at all c	other Resets
'1' = Bit is set		'0' = Bit is cle	ared				
bit 7 <b>TOEN:</b> Timer0 Enable bit 1 = The module is enabled and operating 0 = The module is disabled and in the lowest power mode							
bit 6	Unimplemented: Read as '0'						
bit 5	<b>T0OUT:</b> Timer0 Output bit (read-only) Timer0 output bit						
bit 4	<b>T016BIT:</b> Timer0 Operating as 16-bit Timer Select bit 1 = Timer0 is a 16-bit timer 0 = Timer0 is an 8-bit timer						
bit 3-0	<b>TOOUTPS&lt;3:0&gt;:</b> Timer0 output postscaler (divider) select bits 1111 = 1:16 Postscaler 1110 = 1:15 Postscaler 1101 = 1:14 Postscaler 1010 = 1:13 Postscaler 1010 = 1:11 Postscaler 1001 = 1:10 Postscaler 1000 = 1:9 Postscaler 0111 = 1:8 Postscaler 0110 = 1:7 Postscaler 0101 = 1:6 Postscaler 0100 = 1:5 Postscaler 0111 = 1:4 Postscaler 0011 = 1:4 Postscaler 0011 = 1:2 Postscaler 0011 = 1:2 Postscaler 0001 = 1:2 Postscaler						



## 27.1 Timer2 Operation

Timer2 operates in three major modes:

- Free Running Period
- One-shot
- Monostable

Within each mode there are several options for starting, stopping, and reset. Table 27-1 lists the options.

In all modes, the TMR2 count register is incremented on the rising edge of the clock signal from the programmable prescaler. When TMR2 equals T2PR, a high level is output to the postscaler counter. TMR2 is cleared on the next clock input.

An external signal from hardware can also be configured to gate the timer operation or force a TMR2 count Reset. In Gate modes the counter stops when the gate is disabled and resumes when the gate is enabled. In Reset modes the TMR2 count is reset on either the level or edge from the external source.

The TMR2 and T2PR registers are both directly readable and writable. The TMR2 register is cleared and the T2PR register initializes to FFh on any device Reset. Both the prescaler and postscaler counters are cleared on the following events:

- a write to the TMR2 register
- a write to the T2CON register
- · any device Reset
- External Reset Source event that resets the timer.

Note:	TMR2	is	not	cleared	when	T2CON	is
	written.						

#### 27.1.1 FREE RUNNING PERIOD MODE

The value of TMR2 is compared to that of the Period register, T2PR, on each TMR2\_clk cycle. When the two values match, the comparator resets the value of TMR2 to 00h on the next rising TMR2\_clk edge and increments

the output postscaler counter. When the postscaler count equals the value in the OUTPS<4:0> bits of the TMRxCON1 register, a one TMR2\_clk period wide pulse occurs on the TMR2\_postscaled output, and the postscaler count is cleared.

#### 27.1.2 ONE-SHOT MODE

The One-Shot mode is identical to the Free Running Period mode except that the ON bit is cleared and the timer is stopped when TMR2 matches T2PR and will not restart until the T2ON bit is cycled off and on. Postscaler OUTPS<4:0> values other than 0 are meaningless in this mode because the timer is stopped at the first period event and the postscaler is reset when the timer is restarted.

#### 27.1.3 MONOSTABLE MODE

Monostable modes are similar to One-Shot modes except that the ON bit is not cleared and the timer can be restarted by an external Reset event.

## 27.2 Timer2 Output

The Timer2 module's primary output is TMR2\_postscaled, which pulses for a single TMR2\_clk period when the postscaler counter matches the value in the OUTPS bits of the TMR2CON register. The T2PR postscaler is incremented each time the TMR2 value matches the T2PR value. This signal can be selected as an input to several other input modules:

- The ADC module, as an Auto-conversion Trigger
- · COG, as an auto-shutdown source

In addition, the Timer2 is also used by the CCP module for pulse generation in PWM mode. Both the actual TMR2 value as well as other internal signals are sent to the CCP module to properly clock both the period and pulse width of the PWM signal. See **Section 28.0** "**Capture/Compare/PWM Modules**" for more details on setting up Timer2 for use with the CCP, as well as the timing diagrams in **Section 27.5** "**Operation Examples**" for examples of how the varying Timer2 modes affect CCP PWM output.

#### 27.3 External Reset Sources

In addition to the clock source, the Timer2 also takes in an external Reset source. This external Reset source is selected for Timer2 with the T2RST register. This source can control starting and stopping of the timer, as well as resetting the timer, depending on which mode the timer is in. The mode of the timer is controlled by the MODE<4:0> bits of the TMRxHLT register. Edge-Triggered modes require six Timer clock periods between external triggers. Level-Triggered modes require the triggering level to be at least three Timer clock periods long. External triggers are ignored while in Debug Freeze mode.

#### REGISTER 28-4: CCPRxH REGISTER: CCPx REGISTER HIGH BYTE

| R/W-x/x |
|---------|---------|---------|---------|---------|---------|---------|---------|
|         |         |         | CCPRx   | <15:8>  |         |         |         |
| bit 7   |         |         |         |         |         |         | bit 0   |
|         |         |         |         |         |         |         |         |

Legend:		
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-n/n = Value at POR and BOR/Value at all other Reset
'1' = Bit is set	'0' = Bit is cleared	

CCPxMODE = Capture mode
CCPRxH<7:0>: Captured value of TMR1H
CCPxMODE = Compare mode
CCPRxH<7:0>: MS Byte compared to TMR1H
CCPxMODE = PWM modes when CCPxFMT = 0:
CCPRxH<7:2>: Not used
CCPRxH<1:0>: Pulse-width Most Significant two bits
CCPxMODE = PWM modes when CCPxFMT = 1:
CCPRxH<7:0>: Pulse-width Most Significant eight bits

#### 30.9 CWG Steering Mode

In Steering mode (MODE = 00x), the CWG allows any combination of the CWG1x pins to be the modulated signal. The same signal can be simultaneously available on multiple pins, or a fixed-value output can be presented.

When the respective STRx bit of CWG1OCON0 is '0', the corresponding pin is held at the level defined. When the respective STRx bit of CWG1OCON0 is '1', the pin is driven by the input data signal. The user can assign the input data signal to one, two, three, or all four output pins.

The POLx bits of the CWG1CON1 register control the signal polarity only when STRx = 1.

The CWG auto-shutdown operation also applies in Steering modes as described in **Section 30.10** "**Auto-Shutdown**". An auto-shutdown event will only affect pins that have STRx = 1.

#### 30.9.1 STEERING SYNCHRONIZATION

Changing the MODE bits allows for two modes of steering, synchronous and asynchronous.

When MODE = 000, the steering event is asynchronous and will happen at the end of the instruction that writes to STRx (that is, immediately). In this case, the output signal at the output pin may be an incomplete waveform. This can be useful for immediately removing a signal from the pin.

When MODE = 001, the steering update is synchronous and occurs at the beginning of the next rising edge of the input data signal. In this case, steering the output on/off will always produce a complete waveform.

Figure 30-10 and Figure 30-11 illustrate the timing of asynchronous and synchronous steering, respectively.





#### FIGURE 30-11: EXAMPLE OF STEERING EVENT (MODE<2:0> = 001)



#### 30.10 Auto-Shutdown

Auto-shutdown is a method to immediately override the CWG output levels with specific overrides that allow for safe shutdown of the circuit. The shutdown state can be either cleared automatically or held until cleared by software. The auto-shutdown circuit is illustrated in Figure 30-12.

#### 30.10.1 SHUTDOWN

The shutdown state can be entered by either of the following two methods:

- Software generated
- External Input

#### 30.10.1.1 Software Generated Shutdown

Setting the SHUTDOWN bit of the CWG1AS0 register will force the CWG into the shutdown state.

When the auto-restart is disabled, the shutdown state will persist as long as the SHUTDOWN bit is set.

When auto-restart is enabled, the SHUTDOWN bit will clear automatically and resume operation on the next rising edge event.

#### 30.10.2 EXTERNAL INPUT SOURCE

External shutdown inputs provide the fastest way to safely suspend CWG operation in the event of a Fault condition. When any of the selected shutdown inputs goes active, the CWG outputs will immediately go to the selected override levels without software delay. Several input sources can be selected to cause a shutdown condition. All input sources are active-low. The sources are:

- Comparator C1OUT\_sync
- Comparator C2OUT\_sync
- · Timer2 TMR2\_postscaled
- CWG1IN input pin

Shutdown inputs are selected using the CWG1AS1 register (Register 30-6).

Note: Shutdown inputs are level sensitive, not edge sensitive. The shutdown state cannot be cleared, except by disabling auto-shutdown, as long as the shutdown input level persists.

#### 30.11 Operation During Sleep

The CWG module operates independently from the system clock and will continue to run during Sleep, provided that the clock and input sources selected remain active.

The HFINTOSC remains active during Sleep when all the following conditions are met:

- CWG module is enabled
- · Input source is active
- HFINTOSC is selected as the clock source, regardless of the system clock source selected.

In other words, if the HFINTOSC is simultaneously selected as the system clock and the CWG clock source, when the CWG is enabled and the input source is active, then the CPU will go idle during Sleep, but the HFINTOSC will remain active and the CWG will continue to operate. This will have a direct effect on the Sleep mode current.

#### REGISTER 31-11: CLCDATA: CLC DATA OUTPUT

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0	
—	—	—	—	MLC4OUT	MLC3OUT	MLC2OUT	MLC1OUT	
bit 7							bit 0	
Legend:								
R = Readable I	oit	W = Writable bit		U = Unimplemented bit, read as '0'				
u = Bit is uncha	anged	x = Bit is unkr	nown	-n/n = Value at POR and BOR/Value at all other Resets				
'1' = Bit is set		'0' = Bit is clea	ared					
bit 7-4 Unimplemented: Read as '0'			כ'					
bit 3	3 <b>MLC4OUT:</b> Mirror copy of LC4OUT bit							
bit 2	MLC3OUT: Mirror copy of LC3OUT bit							
bit 1	bit 1 MLC2OUT: Mirror copy of LC2OUT bit							
bit 0 MLC1OUT: Mirror copy of LC1OUT bit								

© 2016 Microchip Technology Inc.

CALL	Call Subroutine
Syntax:	[ <i>label</i> ] CALL k
Operands:	$0 \leq k \leq 2047$
Operation:	(PC)+ 1→ TOS, k → PC<10:0>, (PCLATH<6:3>) → PC<14:11>
Status Affected:	None
Description:	Call Subroutine. First, return address (PC + 1) is pushed onto the stack. The 11-bit immediate address is loaded into PC bits <10:0>. The upper bits of the PC are loaded from PCLATH. CALL is a 2-cycle instruction.

CLRWDT	Clear Watchdog Timer
Syntax:	[label] CLRWDT
Operands:	None
Operation:	$\begin{array}{l} 00h \rightarrow WDT \\ 0 \rightarrow \underline{WDT} \text{ prescaler,} \\ 1 \rightarrow \underline{TO} \\ 1 \rightarrow \underline{PD} \end{array}$
Status Affected:	TO, PD
Description:	CLRWDT instruction resets the Watch- dog Timer. It also resets the prescaler of the WDT. Status bits $\overline{\text{TO}}$ and $\overline{\text{PD}}$ are set.

CALLW	Subroutine Call With W	COMF	Complement f	
Syntax:	[ label ] CALLW	Syntax:	[ <i>label</i> ] COMF f,d	
Operands:	None	Operands:	$0 \le f \le 127$	
Operation:	(PC) +1 $\rightarrow$ TOS, (W) $\rightarrow$ PC<7:0>, (PCLATH<6:0>) $\rightarrow$ PC<14:8>	Operation: Status Affected:	$a \in [0, 1]$ ( $\overline{f}$ ) $\rightarrow$ (destination) Z	
Status Affected:	None	Description:	The contents of register 'f' are complemented. If 'd' is '0', the result is	
Description:	Subroutine call with W. First, the return address (PC + 1) is pushed onto the return stack. Then, the contents of W is loaded into PC<7:0>, and the contents of PCLATH into PC<14:8>. CALLW is a 2-cycle instruction.		stored in W. If 'd' is '1', the result is stored back in register 'f'.	

CLRF	Clear f
Syntax:	[label] CLRF f
Operands:	$0 \leq f \leq 127$
Operation:	$\begin{array}{l} 00h \rightarrow (f) \\ 1 \rightarrow Z \end{array}$
Status Affected:	Z
Description:	The contents of register 'f' are cleared and the Z bit is set.

DECF	Decrement f
Syntax:	[label] DECF f,d
Operands:	$\begin{array}{l} 0 \leq f \leq 127 \\ d  \in  [0,1] \end{array}$
Operation:	(f) - 1 $\rightarrow$ (destination)
Status Affected:	Z
Description:	Decrement register 'f'. If 'd' is '0', the result is stored in the W register. If 'd' is '1', the result is stored back in register 'f'.

# CLRWClear WSyntax:[ label ] CLRWOperands:NoneOperation: $00h \rightarrow (W)$ <br/> $1 \rightarrow Z$ Status Affected:Z

 Status Affected:
 Z

 Description:
 W register is cleared. Zero bit (Z) is set.

© 2016 Microchip Technology Inc.

MOVIW	Move INDFn to W			
Syntax:	[ <i>label</i> ] MOVIW ++FSRn [ <i>label</i> ] MOVIWFSRn [ <i>label</i> ] MOVIW FSRn++ [ <i>label</i> ] MOVIW FSRn [ <i>label</i> ] MOVIW k[FSRn]			
Operands:	n ∈ [0,1] mm ∈ [00,01, 10, 11] -32 ≤ k ≤ 31			
Operation:	$\begin{split} &\text{INDFn} \rightarrow W \\ &\text{Effective address is determined by} \\ &\text{FSR + 1 (preincrement)} \\ &\text{FSR - 1 (predecrement)} \\ &\text{FSR + k (relative offset)} \\ &\text{After the Move, the FSR value will be either:} \\ &\text{FSR + 1 (all increments)} \\ &\text{FSR - 1 (all decrements)} \\ &\text{Unchanged} \end{split}$			
Status Affected:	Z			

Mode	Syntax	mm
Preincrement	++FSRn	00
Predecrement	FSRn	01
Postincrement	FSRn++	10
Postdecrement	FSRn	11

Description:

This instruction is used to move data between W and one of the indirect registers (INDFn). Before/after this move, the pointer (FSRn) is updated by pre/post incrementing/decrementing it.

**Note:** The INDFn registers are not physical registers. Any instruction that accesses an INDFn register actually accesses the register at the address specified by the FSRn.

FSRn is limited to the range 0000h -FFFFh. Incrementing/decrementing it beyond these bounds will cause it to wrap-around.

#### MOVLB Move literal to BSR

Syntax:	[ <i>label</i> ] MOVLB k
Operands:	$0 \le k \le$
Operation:	$k \rightarrow BSR$
Status Affected:	None
Description:	The 6-bit literal 'k' is loaded into the Bank Select Register (BSR).

MOVLP	Move literal to PCLATH				
Syntax:	[ label ] MOVLP k				
Operands:	$0 \le k \le 127$				
Operation:	$k \rightarrow PCLATH$				
Status Affected:	None				
Description:	The 7-bit literal 'k' is loaded into the PCLATH register.				
MOVLW	Move literal to W				
Syntax:	[label] MOVLW k				
Operands:	$0 \le k \le 255$				
Operation:	$k \rightarrow (W)$				
Status Affected:	None				
Description:	The 8-bit literal 'k' is loaded into W register. The "don't cares" will assemble as '0's.				
Words:	1				
Cycles:	1				
Example:	MOVLW 0x5A				
	After Instruction W = 0x5A				
MOVWF	Move W to f				
Syntax:	[label] MOVWF f				
Operands:	$0 \leq f \leq 127$				
Operation:	$(W) \rightarrow (f)$				
Status Affected:	None				
Description:	Move data from W register to register 'f'.				
Words:	1				
Cycles:	1				
Example:	MOVWF LATA				
	Before Instruction				
	LATA = 0xFF				

W = 0x4FAfter Instruction LATA = 0x4F W = 0x4F



© 2016 Microchip Technology Inc.

## TABLE 37-17: ZERO CROSS DETECT (ZCD) SPECIFICATIONS

Standard Operating Conditions (unless otherwise stated) VDD = 3.0V, TA = 25°C							
Param. No.	Sym.	Characteristics	Min.	Тур†	Max.	Units	Comments
ZC01	VPINZC	Voltage on Zero Cross Pin	_	0.75	—	V	$\sim$
ZC02	IZCD_MAX	Maximum source or sink current	_	_	600	μΑ)	
ZC03	TRESPH	Response Time, Rising Edge		1	_	ļus	
	TRESPL	Response Time, Falling Edge	_	1	_	μs	

† Data in "Typ" column is at 3.0V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

#### FIGURE 37-12: TIMER0 AND TIMER1 EXTERNAL CLOCK TIMINGS

