



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, PSMC, PWM, WDT
Number of I/O	35
Program Memory Size	28KB (16K x 14)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.3V ~ 5.5V
Data Converters	A/D 14x12b; D/A 1x8b, 3x5b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-VQFN Exposed Pad
Supplier Device Package	44-QFN (8x8)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/pic16f1789-i-ml

Email: info@E-XFL.COM

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

Name	Function	Input Type	Output Type	Description		
RB4/AN11/C3IN1+/SS <sup>(1)</sup>	RB4	TTL/ST	CMOS	General purpose I/O.		
	AN11	AN		ADC Channel 11 input.		
	C3IN1+	AN		Comparator C3 positive input.		
	SS	ST	_	Slave Select input.		
RB5/AN13/C4IN2-/T1G/CCP3(1)	RB5	TTL/ST	CMOS	General purpose I/O.		
SDO <sup>(1)</sup> /C3OUT	AN13	AN		ADC Channel 13 input.		
	C4IN2-	AN		Comparator C4 negative input.		
	T1G	ST	_	Timer1 gate input.		
	CCP3	ST	CMOS	Capture/Compare/PWM3.		
	SDO	—	CMOS	SPI data output.		
	C3OUT	—	CMOS	Comparator C3 output.		
RB6/C4IN1+/TX <sup>(1)</sup> /CK <sup>(1)</sup> /SDI <sup>(1)</sup> /	RB6	TTL/ST	CMOS	General purpose I/O.		
SDA <sup>(1)</sup> /ICSPCLK	C4IN1+	AN	—	Comparator C4 positive input.		
	TX	_	CMOS	EUSART asynchronous transmit.		
	СК	ST	CMOS	EUSART synchronous clock.		
	SDI	ST	—	SPI data input.		
	SDA	l <sup>2</sup> C	OD	I <sup>2</sup> C data input/output.		
	ICSPCLK	ST	_	Serial Programming Clock.		
RB7/DAC1OUT2/DAC2OUT2/	RB7	TTL/ST	CMOS	General purpose I/O.		
	DAC10UT2	—	AN	Voltage Reference output.		
DI VSCK VSCL VICSPDAI	DAC2OUT2	—	AN	Voltage Reference output.		
	DAC3OUT2	_	AN	Voltage Reference output.		
	DAC4OUT2	—	AN	Voltage Reference output.		
	RX	ST	_	EUSART asynchronous input.		
	DT	ST	CMOS	EUSART synchronous data.		
	SCK	ST	CMOS	SPI clock.		
	SCL	l <sup>2</sup> C	OD	I <sup>2</sup> C clock.		
	ICSPDAT	ST	CMOS	ICSP™ Data I/O.		
RC0/T1OSO/T1CKI/PSMC1A	RC0	TTL/ST	CMOS	General purpose I/O.		
	T10S0	XTAL	XTAL	Timer1 Oscillator Connection.		
	T1CKI	ST	—	Timer1 clock input.		
	PSMC1A	—	CMOS	PSMC1 output A.		
RC1/T1OSI/PSMC1B/CCP2	RC1	TTL/ST	CMOS	General purpose I/O.		
	T10SI	XTAL	XTAL	Timer1 Oscillator Connection.		
	PSMC1B	—	CMOS	PSMC1 output B.		
	CCP2	ST	CMOS	Capture/Compare/PWM2.		
RC2/PSMC1C/PSMC3B/CCP1	RC2	TTL/ST	CMOS	General purpose I/O.		
	PSMC1C	—	CMOS	PSMC1 output C.		
	PSMC3B	—	CMOS	PSMC3 output B.		
	CCP1	ST	CMOS	Capture/Compare/PWM1.		

#### TABLE 1-2: PIC16(L)F1788 PINOUT DESCRIPTION (CONTINUED)

 Legend:
 AN = Analog input or output
 CMOS = CMOS compatible input or output
 OD
 = Open-Drain

 TTL = TTL compatible input
 ST
 = Schmitt Trigger input with CMOS levels
 I<sup>2</sup>C
 = Schmitt Trigger input with I<sup>2</sup>C

HV = High Voltage XTAL = Crystal levels

**Note 1:** Pin functions can be assigned to one of two locations via software. See **Register 13-1**.

**2:** All pins have interrupt-on-change functionality.

# 3.2 Data Memory Organization

The data memory is partitioned in 32 memory banks with 128 bytes in a bank. Each bank consists of (Figure 3-2):

- 12 core registers
- 20 Special Function Registers (SFR)
- Up to 80 bytes of General Purpose RAM (GPR)
- 16 bytes of common RAM

The active bank is selected by writing the bank number into the Bank Select Register (BSR). Unimplemented memory will read as '0'. All data memory can be accessed either directly (via instructions that use the file registers) or indirectly via the two File Select Registers (FSR). See **Section 3.6** "**Indirect Addressing**" for more information.

Data memory uses a 12-bit address. The upper five bits of the address define the Bank address and the lower seven bits select the registers/RAM in that bank.

#### 3.2.1 CORE REGISTERS

The core registers contain the registers that directly affect the basic operation. The core registers occupy the first 12 addresses of every data memory bank (addresses x00h/x08h through x0Bh/x8Bh). These registers are listed below in Table 3-2. For detailed information, see Table 3-11.

TARI E 3-2-	CORF REGISTERS
IADLL J-Z.	

Addresses	BANKx
x00h or x80h	INDF0
x01h or x81h	INDF1
x02h or x82h	PCL
x03h or x83h	STATUS
x04h or x84h	FSR0L
x05h or x85h	FSR0H
x06h or x86h	FSR1L
x07h or x87h	FSR1H
x08h or x88h	BSR
x09h or x89h	WREG
k0Ah or x8Ah	PCLATH
0Bh or x8Bh	INTCON



# 5.3 Register Definitions: BOR Control

#### REGISTER 5-1: BORCON: BROWN-OUT RESET CONTROL REGISTER

R/W-1/u	R/W-0/u	U-0	U-0	U-0	U-0	U-0	R-q/u	
SBOREN	BORFS	—	—	—	—	—	BORRDY	
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	q = Value depends on condition

bit 7	SBOREN: Software Brown-out Reset Enable bit
	If BOREN <1:0> in Configuration Words ≠ 01:
	SBOREN is read/write, but has no effect on the BOR.
	If BOREN <1:0> in Configuration Words = 01:
	1 = BOR Enabled
	0 = BOR Disabled
bit 6	BORFS: Brown-out Reset Fast Start bit <sup>(1)</sup>
	<u> </u>
	BORFS is Read/Write, but has no effect.
	If BOREN <1:0> = 10 (Disabled in Sleep) or BOREN<1:0> = 01 (Under software control):
	1 = Band gap is forced on always (covers sleep/wake-up/operating cases)
	0 = Band gap operates normally, and may turn off
bit 5-1	Unimplemented: Read as '0'
bit 0	BORRDY: Brown-out Reset Circuit Ready Status bit
	1 = The Brown-out Reset circuit is active
	0 = The Brown-out Reset circuit is inactive

**Note 1:** BOREN<1:0> bits are located in Configuration Words.

# 5.13 Power Control (PCON) Register

The Power Control (PCON) register contains flag bits to differentiate between a:

- Power-on Reset (POR)
- Brown-out Reset (BOR)
- Reset Instruction Reset (RI)
- MCLR Reset (RMCLR)
- Watchdog Timer Reset (RWDT)
- Stack Underflow Reset (STKUNF)
- Stack Overflow Reset (STKOVF)

#### 5.14 Register Definitions: Power Control

#### REGISTER 5-2: PCON: POWER CONTROL REGISTER

R/W/HS-0/q	R/W/HS-0/q	U-0	R/W/HC-1/q	R/W/HC-1/q	R/W/HC-1/q	R/W/HC-q/u	R/W/HC-q/u
STKOVF	STKUNF	—	RWDT	RMCLR	RI	POR	BOR
bit 7							bit 0

Legend:		
HC = Bit is cleared by hardwa	are	HS = Bit is set by hardware
R = Readable bit	W = Writable bit	U = Unimplemented bit, read as '0'
u = Bit is unchanged	x = Bit is unknown	-m/n = Value at POR and BOR/Value at all other Resets
'1' = Bit is set	'0' = Bit is cleared	q = Value depends on condition

STKOVF: Stack Overflow Flag bit
1 = A Stack Overflow occurred
0 = A Stack Overflow has not occurred or cleared by firmware
STKUNF: Stack Underflow Flag bit
1 = A Stack Underflow occurred
0 = A Stack Underflow has not occurred or cleared by firmware
Unimplemented: Read as '0'
RWDT: Watchdog Timer Reset Flag bit
1 = A Watchdog Timer Reset has not occurred or set to '1' by firmware
0 = A Watchdog Timer Reset has occurred (cleared by hardware)
RMCLR: MCLR Reset Flag bit
1 = A $\overline{\text{MCLR}}$ Reset has not occurred or set to '1' by firmware
0 = A MCLR Reset has occurred (cleared by hardware)
RI: RESET Instruction Flag bit
1 = A RESET instruction has not been executed or set to '1' by firmware
0 = A RESET instruction has been executed (cleared by hardware)
POR: Power-on Reset Status bit
1 = No Power-on Reset occurred
0 = A Power-on Reset occurred (must be set in software after a Power-on Reset occurs)
BOR: Brown-out Reset Status bit
1 = No Brown-out Reset occurred
0 = A Brown-out Reset occurred (must be set in software after a Power-on Reset or Brown-out Reset
occurs)

The PCON register bits are shown in Register 5-2.

# PIC16(L)F1788/9

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page		
BORCON	SBOREN	BORFS			—		_	BORRDY	65		
PCON	STKOVF	STKUNF	_	RWDT	RMCLR	RI	POR	BOR	69		
STATUS	—	_		TO	PD	Z	DC	С	31		
WDTCON	—		WDTPS<4:0> SWE						114		

# TABLE 5-5: SUMMARY OF REGISTERS ASSOCIATED WITH RESETS

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

#### 6.4.2 TWO-SPEED START-UP SEQUENCE

- 1. Wake-up from Power-on Reset or Sleep.
- 2. Instructions begin execution by the internal oscillator at the frequency set in the IRCF<3:0> bits of the OSCCON register.
- 3. OST enabled to count 1024 clock cycles.
- 4. OST timed out, wait for falling edge of the internal oscillator.
- 5. OSTS is set.
- 6. System clock held low until the next falling edge of new clock (LP, XT or HS mode).
- 7. System clock is switched to external clock source.

# 

#### FIGURE 6-8: TWO-SPEED START-UP

### 6.4.3 CHECKING TWO-SPEED CLOCK STATUS

Checking the state of the OSTS bit of the OSCSTAT register will confirm if the microcontroller is running from the external clock source, as defined by the FOSC<2:0> bits in the Configuration Words, or the internal oscillator.

R/W-0/0	R/W-0/0	R-0/0	R-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF
bit 7			•				bit 0
Legend:							
R = Readable	bit	W = Writable	bit	U = Unimpler	mented bit, read	as '0'	
u = Bit is unch	nanged	x = Bit is unkr	nown	-n/n = Value a	at POR and BOI	R/Value at all o	ther Resets
'1' = Bit is set		'0' = Bit is clea	ared				
bit 7	TMR1GIF: Tir	mer1 Gate Inte	rrupt Flag bit				
	1 = Interrupt is	s pending					
bit 6		s not pending	unt Elog hit				
Dit 0	1 = Interrunt i	s nendina	upt Flag bit				
	0 = Interrupt is	s not pending					
bit 5	RCIF: EUSAF	RT Receive Inte	errupt Flag bit				
	1 = Interrupt i	s pending					
	0 = Interrupt i	s not pending					
bit 4	TXIF: EUSAR	T Transmit Inte	errupt Flag bit				
	1 = Interrupt is	s pending					
hit 3	SSP1IE: Sync	s not pending hronous Seria	l Port (MSSP)	Interrunt Flag	bit		
bit o	1 = Interrupt is	s pendina		interrupt i lag	Sit		
	0 = Interrupt is	s not pending					
bit 2	CCP1IF: CCF	1 Interrupt Fla	g bit				
	1 = Interrupt is	s pending					
	0 = Interrupt is	s not pending					
bit 1	TMR2IF: Time	er2 to PR2 Inte	errupt Flag bit				
	1 = Interrupt is 0 = Interrupt is	s penaing s not pending					
bit 0	TMR1IF: Time	er1 Overflow In	iterrupt Flag bi	t			
	1 = Interrupt is	s pending		-			
	0 = Interrupt i	s not pending					
Note: Int	errupt flag bits a	re set when an	interrupt				
CO	ndition occurs, re	egardless of the	e state of				
its corresponding enable bit or the Global							
Enable bit, GIE, of the INTCON register.							
ap	propriate interru	upt flag bits a	are clear				
pri	or to enabling ar	n interrupt.					

# REGISTER 8-6: PIR1: PERIPHERAL INTERRUPT REQUEST REGISTER 1

#### 9.1.1 WAKE-UP USING INTERRUPTS

When global interrupts are disabled (GIE cleared) and any interrupt source has both its interrupt enable bit and interrupt flag bit set, one of the following will occur:

- If the interrupt occurs **before** the execution of a SLEEP instruction
  - SLEEP instruction will execute as a NOP.
  - WDT and WDT prescaler will not be cleared
  - TO bit of the STATUS register will not be set
  - PD bit of the STATUS register will not be cleared.

- If the interrupt occurs **during or after** the execution of a **SLEEP** instruction
  - SLEEP instruction will be completely executed
  - Device will immediately wake-up from Sleep
  - WDT and WDT prescaler will be cleared
  - TO bit of the STATUS register will be set
  - PD bit of the STATUS register will be cleared.

Even if the flag bits were checked before executing a SLEEP instruction, it may be possible for flag bits to become set before the SLEEP instruction completes. To determine whether a SLEEP instruction executed, test the PD bit. If the PD bit is set, the SLEEP instruction was executed as a NOP.

CLKIN <sup>(1</sup> CLKOUT <sup>(2</sup>	a1 a2 a3 a4 a1 a2 a3  a4 a		Q1 Q2 Q3 Q4 //_//_/	Q1 Q2 Q3 Q4 ~~_~~	Q1 Q2 Q3 Q4 /~_/~_/ //	Q1 Q2 Q3 Q4 	
Interrupt flag		/'	Interrupt Laten	CV <sup>(4)</sup>	<del>،                                     </del>		
	· · · · ·	*	I I	- ,	• •		
GIE bit	\ <u></u>	Processor in	1		· · · ·	I	
(INTCONTEG	·/i i i i i	Sleep	I I	l I	1 I	1	
	·; ; ;-	;			<u> </u> ;		
Instruction Flow					0004	0005h	
PC	$\chi - PC - \chi - PC + 1 - \chi$	PC + 2	<u>χ PC+2</u>	<u>X PC+2</u>	<u>χ 0004h</u>	<u>x 0005n</u>	
Instruction J Fetched	Inst(PC) = Sleep Inst(PC + 1)		Inst(PC + 2)	1 1	Inst(0004h)	Inst(0005h)	
Instruction J Executed	Inst(PC - 1) Sleep		Inst(PC + 1)	Forced NOP	Forced NOP	Inst(0004h)	
Note 1:	External clock. High, Medium, Low mo	de assumed.					
2:	CLKOUT is shown here for timing refer	rence.					
3:	Tost = 1024 Tosc; This delay does not apply to EC, RC and INTOSC Oscillator modes or Two-Speed Start-up (See Section 6.4 "Two-Speed Clock Start-up Mode").						
4:	GIE = 1 assumed. In this case after wa	ake-up, the processo	r calls the ISR at 0	0004h. If GIE = 0,	execution will cont	inue in-line.	

#### FIGURE 9-1: WAKE-UP FROM SLEEP THROUGH INTERRUPT

# 10.0 LOW DROPOUT (LDO) VOLTAGE REGULATOR

The "F" devices have an internal Low Dropout Regulator (LDO) which provide operation above 3.6V. The LDO regulates a voltage for the internal device logic while permitting the VDD and I/O pins to operate at a higher voltage. There is no user enable/disable control available for the LDO, it is always active. The "LF" devices operate at a maximum VDD of 3.6V and does not incorporate an LDO.

A device I/O pin may be configured as the LDO voltage output, identified as the VCAP pin. Although not required, an external low-ESR capacitor may be connected to the VCAP pin for additional regulator stability.

The  $\overline{\text{VCAPEN}}$  bit of Configuration Words determines if which pin is assigned as the VCAP pin. Refer to Table 10-1.

VCAPEN	Pin
1	No VCAP
0	RA6

#### TABLE 10-1: VCAPEN SELECT BIT

On power-up, the external capacitor will load the LDO voltage regulator. To prevent erroneous operation, the device is held in Reset while a constant current source charges the external capacitor. After the cap is fully charged, the device is released from Reset. For more information on the constant current rate, refer to the LDO Regulator Characteristics Table in **Section 31.0 "Electrical Specifications"**.

#### TABLE 10-2: SUMMARY OF CONFIGURATION WORD WITH LDO

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			LVP	DEBUG	LPBOR	BORV	STVREN	PLLEN	<u> </u>
CONFIG2	7:0	_		VCAPEN <sup>(1)</sup>	_			WRT	<1:0>	60

**Legend:** — = unimplemented locations read as '0'. Shaded cells are not used by LDO.

Note 1: "F" devices only.

# 12.2 Using the Data EEPROM

The data EEPROM is a high-endurance, byte addressable array that has been optimized for the storage of frequently changing information (e.g., program variables or other data that are updated often). When variables in one section change frequently, while variables in another section do not change, it is possible to exceed the total number of write cycles to the EEPROM without exceeding the total number of write cycles to a single byte. Refer to **Section 31.0 "Electrical Specifications"**. If this is the case, then a refresh of the array must be performed. For this reason, variables that change infrequently (such as constants, IDs, calibration, etc.) should be stored in Flash program memory.

#### 12.2.1 READING THE DATA EEPROM MEMORY

To read a data memory location, the user must write the address to the EEADRL register, clear the EEPGD and CFGS control bits of the EECON1 register, and then set control bit RD. The data is available at the very next cycle, in the EEDATL register; therefore, it can be read in the next instruction. EEDATL will hold this value until another read or until it is written to by the user (during a write operation).

#### EXAMPLE 12-1: DATA EEPROM READ

BANKSEL	EEADRL	i
MOVLW	DATA_EE_ADDF	R ;
MOVWF	EEADRL	;Data Memory
		;Address to read
BCF	EECON1, CFGS	S ;Deselect Config space
BCF	EECON1, EEPO	GD;Point to DATA memory
BSF	EECON1, RD	;EE Read
MOVF	EEDATL, W	;W = EEDATL

# Note: Data EEPROM can be read regardless of the setting of the CPD bit.

# 12.2.2 WRITING TO THE DATA EEPROM MEMORY

To write an EEPROM data location, the user must first write the address to the EEADRL register and the data to the EEDATL register. Then the user must follow a specific sequence to initiate the write for each byte.

The write will not initiate if the above sequence is not followed exactly (write 55h to EECON2, write AAh to EECON2, then set the WR bit) for each byte. Interrupts should be disabled during this code segment.

Additionally, the WREN bit in EECON1 must be set to enable write. This mechanism prevents accidental writes to data EEPROM due to errant (unexpected) code execution (i.e., lost programs). The user should keep the WREN bit clear at all times, except when updating EEPROM. The WREN bit is not cleared by hardware.

After a write sequence has been initiated, clearing the WREN bit will not affect this write cycle. The WR bit will be inhibited from being set unless the WREN bit is set.

At the completion of the write cycle, the WR bit is cleared in hardware and the EE Write Complete Interrupt Flag bit (EEIF) is set. The user can either enable this interrupt or poll this bit. EEIF must be cleared by software.

# 12.2.3 PROTECTION AGAINST SPURIOUS WRITE

There are conditions when the user may not want to write to the data EEPROM memory. To protect against spurious EEPROM writes, various mechanisms have been built-in. On power-up, WREN is cleared. Also, the Power-up Timer (64 ms duration) prevents EEPROM write.

The write initiate sequence and the WREN bit together help prevent an accidental write during:

- Brown-out
- Power Glitch
- Software Malfunction

#### 12.2.4 DATA EEPROM OPERATION DURING CODE-PROTECT

Data memory can be code-protected by programming the  $\overline{CPD}$  bit in the Configuration Words to '0'.

When the data memory is code-protected, only the CPU is able to read and write data to the data EEPROM. It is recommended to code-protect the program memory when code-protecting data memory. This prevents anyone from replacing your program with a program that will access the contents of the data EEPROM.

# 18.0 OPERATIONAL AMPLIFIER (OPA) MODULES

The Operational Amplifier (OPA) is a standard three-terminal device requiring external feedback to operate. The OPA module has the following features:

- External connections to I/O ports
- Low leakage inputs
- · Factory Calibrated Input Offset Voltage





# 19.0 8-BIT DIGITAL-TO-ANALOG CONVERTER (DAC) MODULE

The Digital-to-Analog Converter supplies a variable voltage reference, ratiometric with the input source, with 256 selectable output levels.

The input of the DAC can be connected to:

- External VREF pins
- VDD supply voltage
- FVR (Fixed Voltage Reference)

The output of the DAC can be configured to supply a reference voltage to the following:

- Comparator positive input
- Op amp positive input
- ADC input channel
- DAC1OUT1 pin
- DAC1OUT2 pin

The Digital-to-Analog Converter (DAC) is enabled by setting the DAC1EN bit of the DAC1CON0 register.

# 19.1 Output Voltage Selection

The DAC has 256 voltage level ranges. The 256 levels are set with the DAC1R<7:0> bits of the DAC1CON1 register.

The DAC output voltage is determined by Equation 19-1:

### EQUATION 19-1: DAC OUTPUT VOLTAGE

$$\frac{IF DACxEN = 1}{VOUT} = \left( (VSOURCE+ - VSOURCE-) \times \frac{DACxR[7:0]}{2^8} \right) + VSOURCE-$$
$$VSOURCE+ = VDD, VREF, or FVR BUFFER 2$$
$$VSOURCE- = VSS$$

# 19.2 Ratiometric Output Level

The DAC output value is derived using a resistor ladder with each end of the ladder tied to a positive and negative voltage reference input source. If the voltage of either input source fluctuates, a similar fluctuation will result in the DAC output value.

The value of the individual resistors within the ladder can be found in **Section 31.0** "**Electrical Specifications**".

# 19.3 DAC Voltage Reference Output

The DAC voltage can be output to the DAC1OUT1 and DAC1OUT2 pins by setting the respective DAC1OE1 and DAC1OE2 pins of the DAC1CON0 register. Selecting the DAC reference voltage for output on either DAC1OUTx pin automatically overrides the digital output buffer and digital input threshold detector functions of that pin. Reading the DAC1OUTx pin when it has been configured for DAC reference voltage output will always return a '0'.

Due to the limited current drive capability, a buffer must be used on the DAC voltage reference output for external connections to either DAC1OUTx pin. Figure 19-2 shows an example buffering technique.

© 2013-2015 Microchip Technology Inc.

# 23.1 Timer1 Operation

The Timer1 module is a 16-bit incrementing counter which is accessed through the TMR1H:TMR1L register pair. Writes to TMR1H or TMR1L directly update the counter.

When used with an internal clock source, the module is a timer and increments on every instruction cycle. When used with an external clock source, the module can be used as either a timer or counter and increments on every selected edge of the external source.

Timer1 is enabled by configuring the TMR1ON and TMR1GE bits in the T1CON and T1GCON registers, respectively. Table 23-1 displays the Timer1 enable selections.

TABLE 23-1:	TIMER1 ENABLE
	SELECTIONS

TMR10N	TMR1GE	Timer1 Operation		
0	0	Off		
0	1	Off		
1	0	Always On		
1	1	Count Enabled		

#### 23.2 Clock Source Selection

The TMR1CS<1:0> and T1OSCEN bits of the T1CON register are used to select the clock source for Timer1. Table 23-2 displays the clock source selections.

#### 23.2.1 INTERNAL CLOCK SOURCE

When the internal clock source is selected, the TMR1H:TMR1L register pair will increment on multiples of Fosc as determined by the Timer1 prescaler.

When the Fosc internal clock source is selected, the Timer1 register value will increment by four counts every instruction clock cycle. Due to this condition, a 2 LSB error in resolution will occur when reading the Timer1 value. To utilize the full resolution of Timer1, an asynchronous input signal must be used to gate the Timer1 clock input.

The following asynchronous sources may be used:

- Asynchronous event on the T1G pin to Timer1
  gate
- C1 or C2 comparator input to Timer1 gate

#### 23.2.2 EXTERNAL CLOCK SOURCE

When the external clock source is selected, the Timer1 module may work as a timer or a counter.

When enabled to count, Timer1 is incremented on the rising edge of the external clock input T1CKI, which can be synchronized to the microcontroller system clock or can run asynchronously.

When used as a timer with a clock oscillator, an external 32.768 kHz crystal can be used in conjunction with the dedicated internal oscillator circuit.

**Note:** In Counter mode, a falling edge must be registered by the counter prior to the first incrementing rising edge after any one or more of the following conditions:

- Timer1 enabled after POR
- Write to TMR1H or TMR1L
- · Timer1 is disabled
- Timer1 is disabled (TMR1ON = 0) when T1CKI is high then Timer1 is enabled (TMR1ON=1) when T1CKI is low.

TMR1CS<1:0>	T10SCEN	Clock Source				
11	x	Reserved				
10	1	mer1 Oscillator				
10	0	xternal Clocking on T1CKI Pin				
01	x	System Clock (Fosc)				
00	x	Instruction Clock (Fosc/4)				

#### TABLE 23-2: CLOCK SOURCE SELECTIONS

#### 23.6.2 TIMER1 GATE SOURCE SELECTION

Timer1 gate source selections are shown in Table 23-4. Source selection is controlled by the T1GSS bits of the T1GCON register. The polarity for each available source is also selectable. Polarity selection is controlled by the T1GPOL bit of the T1GCON register.

TABLE 23-4:	TIMER1	GATE	SOUR	CES
-------------	--------	------	------	-----

T1GSS	Timer1 Gate Source
00	Timer1 Gate Pin
01	Overflow of Timer0 (TMR0 increments from FFh to 00h)
10	Comparator 1 Output sync_C1OUT (optionally Timer1 synchronized output)
11	Comparator 2 Output sync_C2OUT (optionally Timer1 synchronized output)

#### 23.6.2.1 T1G Pin Gate Operation

The T1G pin is one source for Timer1 gate control. It can be used to supply an external source to the Timer1 gate circuitry.

#### 23.6.2.2 Timer0 Overflow Gate Operation

When Timer0 increments from FFh to 00h, a low-to-high pulse will automatically be generated and internally supplied to the Timer1 gate circuitry.

#### 23.6.2.3 Comparator C1 Gate Operation

The output resulting from a Comparator 1 operation can be selected as a source for Timer1 gate control. The Comparator 1 output (sync\_C1OUT) can be synchronized to the Timer1 clock or left asynchronous. For more information see **Section 21.4.1 "Comparator Output Synchronization**".

#### 23.6.2.4 Comparator C2 Gate Operation

The output resulting from a Comparator 2 operation can be selected as a source for Timer1 gate control. The Comparator 2 output (sync\_C2OUT) can be synchronized to the Timer1 clock or left asynchronous. For more information see **Section 21.4.1 "Comparator Output Synchronization"**.

#### 23.6.3 TIMER1 GATE TOGGLE MODE

When Timer1 Gate Toggle mode is enabled, it is possible to measure the full-cycle length of a Timer1 gate signal, as opposed to the duration of a single level pulse.

The Timer1 gate source is routed through a flip-flop that changes state on every incrementing edge of the signal. See Figure 23-4 for timing details.

Timer1 Gate Toggle mode is enabled by setting the T1GTM bit of the T1GCON register. When the T1GTM bit is cleared, the flip-flop is cleared and held clear. This is necessary in order to control which edge is measured.

Note:	Enabling Toggle mode at the same time
	as changing the gate polarity may result in
	indeterminate operation.

#### 23.6.4 TIMER1 GATE SINGLE-PULSE MODE

When Timer1 Gate Single-Pulse mode is enabled, it is possible to capture a single-pulse gate event. Timer1 Gate Single-Pulse mode is enabled by first setting the T1GSPM bit in the T1GCON register. Next, the T1GGO/DONE bit in the T1GCON register must be set. The Timer1 will be fully enabled on the next incrementing edge. On the next trailing edge of the pulse, the T1GGO/DONE bit will automatically be cleared. No other gate events will be allowed to increment Timer1 until the T1GGO/DONE bit is once again set in software. See Figure 23-5 for timing details.

If the Single-Pulse Gate mode is disabled by clearing the T1GSPM bit in the T1GCON register, the T1GGO/DONE bit should also be cleared.

Enabling the Toggle mode and the Single-Pulse mode simultaneously will permit both sections to work together. This allows the cycle times on the Timer1 gate source to be measured. See Figure 23-6 for timing details.

#### 23.6.5 TIMER1 GATE VALUE

When Timer1 Gate Value Status is utilized, it is possible to read the most current level of the gate control value. The value is accessible by reading the T1GVAL bit in the T1GCON register. The T1GVAL bit is valid even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

#### 23.6.6 TIMER1 GATE EVENT INTERRUPT

When Timer1 Gate Event Interrupt is enabled, it is possible to generate an interrupt upon the completion of a gate event. When the falling edge of T1GVAL occurs, the TMR1GIF flag bit in the PIR1 register will be set. If the TMR1GIE bit in the PIE1 register is set, then an interrupt will be recognized.

The TMR1GIF flag bit operates even when the Timer1 gate is not enabled (TMR1GE bit is cleared).

PSMCxSTR0       01h       02h       04h       08h       10h       20h         Period Event	3-Phase State	1	2	3	44	5	6
Period Event	PSMCxSTR0	01h	02h	04h	08h		20h
Rising Edge Event	Period Event						
Falling Edge Event	Rising Edge Event						
PSMCxA (1H)	Falling Edge Event						
PSMCxB (1L)	PSMCxA (1H)						
PSMCxC (2H)	PSMCxB (1L)						
PSMCxD (2L)	PSMCxC (2H)						
PSMCxE (3H)	PSMCxD (2L)						
	PSMCxE (3H)						
	PSMCxF (3L)						

# FIGURE 26-15: 3-PHASE PWM STEERING WAVEFORM (PXHSMEN = 0 AND PXLSMEN = 1)

#### 27.6.2 CLOCK ARBITRATION

Clock arbitration occurs when the master, during any receive, transmit or Repeated Start/Stop condition, releases the SCL pin (SCL allowed to float high). When the SCL pin is allowed to float high, the Baud Rate Generator (BRG) is suspended from counting until the SCL pin is actually sampled high. When the SCL pin is sampled high, the Baud Rate Generator is reloaded with the contents of SSPADD<7:0> and begins counting. This ensures that the SCL high time will always be at least one BRG rollover count in the event that the clock is held low by an external device (Figure 27-25).

#### FIGURE 27-25: BAUD RATE GENERATOR TIMING WITH CLOCK ARBITRATION



#### 27.6.3 WCOL STATUS FLAG

If the user writes the SSPBUF when a Start, Restart, Stop, Receive or Transmit sequence is in progress, the WCOL is set and the contents of the buffer are unchanged (the write does not occur). Any time the WCOL bit is set it indicates that an action on SSPBUF was attempted while the module was not idle.

Note: Because queuing of events is not allowed, writing to the lower five bits of SSPCON2 is disabled until the Start condition is complete.





#### TABLE 28-1: SUMMARY OF REGISTERS ASSOCIATED WITH ASYNCHRONOUS TRANSMISSION

Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Register on Page
APFCON1	C2OUTSEL	CC1PSEL	SDOSEL	SCKSEL	SDISEL	TXSEL	RXSEL	CCP2SEL	132
BAUDCON	ABDOVF	RCIDL		SCKP	BRG16		WUE	ABDEN	356
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	97
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	98
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	102
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	355
SPBRGL	BRG<7:0>							357	
SPBRGH	BRG<15:8>							357	
TRISC	TRISC7 TRISC6 TRISC5 TRISC4 TRISC3 TRISC2 TRISC1 TRISC0							147	
TXREG	EUSART Transmit Data Register							346*	
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	354

Legend: — = unimplemented location, read as '0'. Shaded cells are not used for asynchronous transmission.

\* Page provides register information.

CALLW	Subroutine Call With W
Syntax:	[ label ] CALLW
Operands:	None
Operation:	(PC) +1 → TOS, (W) → PC<7:0>, (PCLATH<6:0>) → PC<14:8>
Status Affected:	None
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.

CLRF	Clear f
Syntax:	[ <i>label</i> ] 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.

CLRW	Clear W
Syntax:	[label] CLRW
Operands:	None
Operation:	$\begin{array}{l} \text{00h} \rightarrow (\text{W}) \\ 1 \rightarrow \text{Z} \end{array}$
Status Affected:	Z
Description:	W register is cleared. Zero bit (Z) is set.

CLRWDT	Clear Watchdog Timer
Syntax:	[label] CLRWDT
Operands:	None
Operation:	$\begin{array}{l} \text{O0h} \rightarrow \text{WDT} \\ 0 \rightarrow \text{WDT prescaler,} \\ 1 \rightarrow \overline{\text{TO}} \\ 1 \rightarrow \overline{\text{PD}} \end{array}$
Status Affected:	TO, PD
Description:	CLRWDT instruction resets the Watchdog Timer. It also resets the prescaler of the WDT. Status bits TO and PD are set.

COMF	Complement f
Syntax:	[ <i>label</i> ] COMF f,d
Operands:	$\begin{array}{l} 0 \leq f \leq 127 \\ d \in [0,1] \end{array}$
Operation:	$(\overline{f}) \rightarrow (destination)$
Status Affected:	Z
Description:	The contents of register 'f' are complemented. If 'd' is '0', the result is stored in W. If 'd' is '1', the result is stored back in register 'f'.

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'.

# 33.2 MPLAB XC Compilers

The MPLAB XC Compilers are complete ANSI C compilers for all of Microchip's 8, 16, and 32-bit MCU and DSC devices. These compilers provide powerful integration capabilities, superior code optimization and ease of use. MPLAB XC Compilers run on Windows, Linux or MAC OS X.

For easy source level debugging, the compilers provide debug information that is optimized to the MPLAB X IDE.

The free MPLAB XC Compiler editions support all devices and commands, with no time or memory restrictions, and offer sufficient code optimization for most applications.

MPLAB XC Compilers include an assembler, linker and utilities. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. MPLAB XC Compiler uses the assembler to produce its object file. Notable features of the assembler include:

- · Support for the entire device instruction set
- Support for fixed-point and floating-point data
- Command-line interface
- · Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

#### 33.3 MPASM Assembler

The MPASM Assembler is a full-featured, universal macro assembler for PIC10/12/16/18 MCUs.

The MPASM Assembler generates relocatable object files for the MPLINK Object Linker, Intel<sup>®</sup> standard HEX files, MAP files to detail memory usage and symbol reference, absolute LST files that contain source lines and generated machine code, and COFF files for debugging.

The MPASM Assembler features include:

- Integration into MPLAB X IDE projects
- User-defined macros to streamline assembly code
- Conditional assembly for multipurpose source files
- Directives that allow complete control over the assembly process

### 33.4 MPLINK Object Linker/ MPLIB Object Librarian

The MPLINK Object Linker combines relocatable objects created by the MPASM Assembler. It can link relocatable objects from precompiled libraries, using directives from a linker script.

The MPLIB Object Librarian manages the creation and modification of library files of precompiled code. When a routine from a library is called from a source file, only the modules that contain that routine will be linked in with the application. This allows large libraries to be used efficiently in many different applications.

The object linker/library features include:

- Efficient linking of single libraries instead of many smaller files
- Enhanced code maintainability by grouping related modules together
- Flexible creation of libraries with easy module listing, replacement, deletion and extraction

# 33.5 MPLAB Assembler, Linker and Librarian for Various Device Families

MPLAB Assembler produces relocatable machine code from symbolic assembly language for PIC24, PIC32 and dsPIC DSC devices. MPLAB XC Compiler uses the assembler to produce its object file. The assembler generates relocatable object files that can then be archived or linked with other relocatable object files and archives to create an executable file. Notable features of the assembler include:

- · Support for the entire device instruction set
- · Support for fixed-point and floating-point data
- Command-line interface
- · Rich directive set
- Flexible macro language
- MPLAB X IDE compatibility

#### 33.6 MPLAB X SIM Software Simulator

The MPLAB X SIM Software Simulator allows code development in a PC-hosted environment by simulating the PIC MCUs and dsPIC DSCs on an instruction level. On any given instruction, the data areas can be examined or modified and stimuli can be applied from a comprehensive stimulus controller. Registers can be logged to files for further run-time analysis. The trace buffer and logic analyzer display extend the power of the simulator to record and track program execution, actions on I/O, most peripherals and internal registers.

The MPLAB X SIM Software Simulator fully supports symbolic debugging using the MPLAB XC Compilers, and the MPASM and MPLAB Assemblers. The software simulator offers the flexibility to develop and debug code outside of the hardware laboratory environment, making it an excellent, economical software development tool.

#### 33.7 MPLAB REAL ICE In-Circuit Emulator System

The MPLAB REAL ICE In-Circuit Emulator System is Microchip's next generation high-speed emulator for Microchip Flash DSC and MCU devices. It debugs and programs all 8, 16 and 32-bit MCU, and DSC devices with the easy-to-use, powerful graphical user interface of the MPLAB X IDE.

The emulator is connected to the design engineer's PC using a high-speed USB 2.0 interface and is connected to the target with either a connector compatible with in-circuit debugger systems (RJ-11) or with the new high-speed, noise tolerant, Low-Voltage Differential Signal (LVDS) interconnection (CAT5).

The emulator is field upgradable through future firmware downloads in MPLAB X IDE. MPLAB REAL ICE offers significant advantages over competitive emulators including full-speed emulation, run-time variable watches, trace analysis, complex breakpoints, logic probes, a ruggedized probe interface and long (up to three meters) interconnection cables.

#### 33.8 MPLAB ICD 3 In-Circuit Debugger System

The MPLAB ICD 3 In-Circuit Debugger System is Microchip's most cost-effective, high-speed hardware debugger/programmer for Microchip Flash DSC and MCU devices. It debugs and programs PIC Flash microcontrollers and dsPIC DSCs with the powerful, yet easy-to-use graphical user interface of the MPLAB IDE.

The MPLAB ICD 3 In-Circuit Debugger probe is connected to the design engineer's PC using a highspeed USB 2.0 interface and is connected to the target with a connector compatible with the MPLAB ICD 2 or MPLAB REAL ICE systems (RJ-11). MPLAB ICD 3 supports all MPLAB ICD 2 headers.

# 33.9 PICkit 3 In-Circuit Debugger/ Programmer

The MPLAB PICkit 3 allows debugging and programming of PIC and dsPIC Flash microcontrollers at a most affordable price point using the powerful graphical user interface of the MPLAB IDE. The MPLAB PICkit 3 is connected to the design engineer's PC using a fullspeed USB interface and can be connected to the target via a Microchip debug (RJ-11) connector (compatible with MPLAB ICD 3 and MPLAB REAL ICE). The connector uses two device I/O pins and the Reset line to implement in-circuit debugging and In-Circuit Serial Programming<sup>™</sup> (ICSP<sup>™</sup>).

# 33.10 MPLAB PM3 Device Programmer

The MPLAB PM3 Device Programmer is a universal, CE compliant device programmer with programmable voltage verification at VDDMIN and VDDMAX for maximum reliability. It features a large LCD display (128 x 64) for menus and error messages, and a modular, detachable socket assembly to support various package types. The ICSP cable assembly is included as a standard item. In Stand-Alone mode, the MPLAB PM3 Device Programmer can read, verify and program PIC devices without a PC connection. It can also set code protection in this mode. The MPLAB PM3 connects to the host PC via an RS-232 or USB cable. The MPLAB PM3 has high-speed communications and optimized algorithms for quick programming of large memory devices, and incorporates an MMC card for file storage and data applications.